A three-dimensional transonic, potential flow computer program: its conversion to IBM fortran and utilization

Paschall, Jack.

https://hdl.handle.net/10945/20012

Downloaded from NPS Archive: Calhoun
NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A THREE-DIMENSIONAL TRANSONIC, POTENTIAL FLOW COMPUTER PROGRAM, ITS CONVERSION TO IBM FORTRAN AND UTILIZATION

by

Jack Paschall III

December 1983

Thesis Advisor: R. D. Zucker

Approved for Public Release, Distribution Unlimited
**REPORT DOCUMENTATION PAGE**

<table>
<thead>
<tr>
<th>1. REPORT NUMBER</th>
<th>2. GOVT ACCESSION NO.</th>
<th>3. RECIPIENT'S CATALOG NUMBER</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>4. TITLE (and Subtitle)</th>
<th>5. TYPE OF REPORT &amp; PERIOD COVERED</th>
</tr>
</thead>
<tbody>
<tr>
<td>A Three-Dimensional Transonic, Potential Flow Computer Program, Its Conversion to IBM Fortran and Utilization</td>
<td>Master's Thesis; December 1983</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>7. AUTHOR(s)</th>
<th>8. CONTRACT OR GRANT NUMBER(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jack Paschall III</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>9. PERFORMING ORGANIZATION NAME AND ADDRESS</th>
<th>10. PROGRAM ELEMENT, PROJECT, TASK AREA &amp; WORK UNIT NUMBERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Naval Postgraduate School Monterey, California 93943</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>11. CONTROLLING OFFICE NAME AND ADDRESS</th>
<th>12. REPORT DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Naval Postgraduate School Monterey, California 93943</td>
<td>December 1983</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>14. MONITORING AGENCY NAME &amp; ADDRESS (if different from Controlling Office)</th>
<th>13. NUMBER OF PAGES</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>152</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>15. SECURITY CLASS. (of this report)</th>
<th>16. DISTRIBUTION STATEMENT (of this Report)</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNCLASSIFIED</td>
<td>Approved for Public Release, Distribution Unlimited</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>18. SUPPLEMENTARY NOTES</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>19. KEY WORDS (Continue on reverse side if necessary and identify by block number)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer Program; Inviscid, Three-Dimensional Transonic Potential Flow Over Wings or Wing-Body Combinations</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>20. ABSTRACT (Continue on reverse side if necessary and identify by block number)</th>
</tr>
</thead>
</table>

This thesis describes the conversion of a computer program from Fortran IV for the NOS 1.2 operating system of the CYBER 175 or CDC 6600 computer to Fortran IV compatible with the Naval Postgraduate School IBM 3033 system. The converted program, called FLO27, calculates the inviscid, three-dimensional transonic potential flow over wings or wing-body combinations. The data input to FLO27 is
extensive; therefore, an interactive program was developed to aid the user in building the required input data file.
Approved for Public Release, Distribution Unlimited

A Three-Dimensional Transonic, Potential Flow Computer Program, Its Conversion to IBM Fortran and Utilization

by

Jack Faschall III
Commander, United States Navy
B.S., Oregon State University, 1965

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1983
This thesis describes the conversion of a computer program from Fortran IV for the NOS 1.2 operating system of the CYBER 175 or CDC 6600 computer to Fortran IV compatible with the Naval Postgraduate School IBM 3033 system. The converted program, called FLO27, calculates the inviscid, three-dimensional transonic potential flow over wings or wing-body combinations. The data input to FLO27 is extensive; therefore, an interactive program was developed to aid the user in building the required input data file.
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I.</td>
<td>INTRODUCTION</td>
<td>9</td>
</tr>
<tr>
<td>A.</td>
<td>BACKGROUND</td>
<td>9</td>
</tr>
<tr>
<td>B.</td>
<td>VISCOUS/INVISCID SYSTEM OF PROGRAMS</td>
<td>10</td>
</tr>
<tr>
<td>II.</td>
<td>POTENTIAL FLOW PROGRAM FLO27</td>
<td>14</td>
</tr>
<tr>
<td>A.</td>
<td>RE-FROGRAMING</td>
<td>14</td>
</tr>
<tr>
<td>B.</td>
<td>PROGRAM DESCRIPTION</td>
<td>19</td>
</tr>
<tr>
<td>1.</td>
<td>Program Input</td>
<td>22</td>
</tr>
<tr>
<td>2.</td>
<td>Program Output</td>
<td>32</td>
</tr>
<tr>
<td>III.</td>
<td>INTERACTIVE INPUT PROGRAM FLO27IN</td>
<td>35</td>
</tr>
<tr>
<td>IV.</td>
<td>FLO27 BATCH SYSTEM EXECUTION</td>
<td>38</td>
</tr>
<tr>
<td>V.</td>
<td>PROGRAM TEST RESULTS</td>
<td>40</td>
</tr>
<tr>
<td>A.</td>
<td>ACCEPTANCE TEST DATA</td>
<td>40</td>
</tr>
<tr>
<td>B.</td>
<td>COMPARISON WITH OTHER PROGRAMS</td>
<td>41</td>
</tr>
<tr>
<td>C.</td>
<td>AE-4501 CLASS PROJECT</td>
<td>41</td>
</tr>
<tr>
<td>APPENDIX A</td>
<td></td>
<td>44</td>
</tr>
<tr>
<td>APPENDIX B</td>
<td></td>
<td>46</td>
</tr>
<tr>
<td>APPENDIX C</td>
<td></td>
<td>47</td>
</tr>
<tr>
<td>APPENDIX D</td>
<td></td>
<td>48</td>
</tr>
<tr>
<td>APPENDIX E</td>
<td></td>
<td>70</td>
</tr>
<tr>
<td>APPENDIX F</td>
<td></td>
<td>132</td>
</tr>
<tr>
<td>LIST OF REFERENCES</td>
<td></td>
<td>151</td>
</tr>
<tr>
<td>INITIAL DISTRIBUTION LIST</td>
<td></td>
<td>152</td>
</tr>
</tbody>
</table>
**LIST OF TABLES**

<table>
<thead>
<tr>
<th></th>
<th>Table Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I.</td>
<td>Viscous/Inviscid Wing System of Programs</td>
<td>11</td>
</tr>
<tr>
<td>II.</td>
<td>CDC Magnetic Tape Files</td>
<td>15</td>
</tr>
<tr>
<td>III.</td>
<td>FL027 Re-Programing Changes</td>
<td>16</td>
</tr>
<tr>
<td>IV.</td>
<td>Initialized Input Variables</td>
<td>18</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

1.1. Viscous/Inviscid Interaction Procedure ........ 12
2.1. Dihedral Angle ....................... 27
2.2. Wing Defining Geometry ................. 28
2.3. Section Defining Geometry ............... 31
2.4. Versatec Plot of C\textsubscript{p} vs. X/C ........ 34
5.1. Program Calculated and Wind Tunnel Data .... 42
ACKNOWLEDGEMENT

I would like to acknowledge the most valuable assistance of the W. R. Church computer center personnel in the successful completion of this thesis. In particular, the guidance of Mr. Dennis Mar was instrumental in directing my initial efforts. I am grateful to Dr. Robert Zucker for providing the constant attention and encouragement which kept me working at an even pace. In addition, I am deeply appreciative of the love and support provided by my wife Tricia and our two children. Most of all I thank God for the strength, determination and peace which has sustained me and brought this project to a successful completion.
I. INTRODUCTION

In the Aeronautical Engineering curriculum graduate level aerodynamics course, AE-4501, the students are exposed to two computer programs. One of these, prepared by the Douglas Aircraft Company, analyzes the potential flow around three-dimensional wings but is limited to incompressible flow [Ref. 1]. The other program, prepared by Cebeci, calculates the friction drag for two dimensional incompressible flow over airfoils [Ref. 2]. A serious defect of these programs is that they are not state-of-the-art computer programs. The Douglas program does not consider the effects of compressibility and the boundary layer program, in addition to being restricted to incompressible flow, does not predict the laminar to turbulent transition point.

A. BACKGROUND

In 1980 the Department of Aeronautics at the Naval Postgraduate School acquired an intricate computer program recently developed by the Boeing Commercial Airplane Company. This state-of-the-art program calculates three-dimensional transonic flow over wings and bodies in
both the outer-inviscid flow region governed by the
transonic potential equation and the thin layer in which the
first order, compressible boundary layer equations are
assumed to be valid.

The Boeing program as received was designed to be executed on a CDC 6600 or a CYBER 175 computer and was written using CDC FORTRAN IV extended language. This thesis therefore was primarily concerned with the conversion of the program to FORTRAN IV extended compatible with the Naval Postgraduate School's (NPS) IBM 3033 system. The large modular program was divided so that the potential flow analysis portion could be run separately. Simplified instructions for use of the program were also prepared.

3. VISCIOUS/INVISCID SYSTEM OF PROGRAMS

The Viscous/Inviscid Wing System (VIWS) of programs calculates three-dimensional transonic flow over wings and wing body combinations including details of the laminar or turbulent flow in the three-dimensional viscous boundary layer. The flow field is calculated in two overlapping regions: an outer inviscid flow region governed by the transonic potential equation, and a thin boundary layer in which the first order, three-dimensional, compressible
boundary layer equations are assumed to hold and in which the effects of surface heat and mass transfer can be computed. A list of the VIWS of programs is presented in Table I.

**TABLE I**

Viscous/Inviscid Wing System of Programs

<table>
<thead>
<tr>
<th>Program Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FI027</td>
<td>Jameson-Caughey inviscid, transonic wing code</td>
</tr>
<tr>
<td>A411IN</td>
<td>Reads geometry &amp; velocity data, constructs coordinate system</td>
</tr>
<tr>
<td>VWIN</td>
<td>Potential flow boundary layer interface</td>
</tr>
<tr>
<td>A411AC1</td>
<td>Three-dimensional boundary layer program</td>
</tr>
<tr>
<td>INTERP</td>
<td>Boundary layer potential flow interface</td>
</tr>
<tr>
<td>A411F1, A411F2, A411FS</td>
<td>Graphics display programs</td>
</tr>
</tbody>
</table>

The basic sequence of calculations used by the VIWS to obtain matched viscous and inviscid solutions consists of an iterative loop in which the inviscid outer flow analysis and the boundary layer analysis are performed sequentially. The iterative sequence is continued until either convergence (satisfactory matching) is achieved, or the maximum number of iterations specified by the user has been performed. The VIWS programing sequence is shown schematically in Fig. 1.1.
The potential flow is calculated for the bare wing during the first iteration. In subsequent iterations, the effect of the boundary layer flow on the outer inviscid flow is
felt as a modification to the wing shape through the addition of the boundary layer displacement thickness. Convergence is recognized, and the iterations are stopped, when the maximum change between the new and old displacement thickness, expressed as a fraction of the maximum displacement thickness, is less than the convergence tolerance chosen by the user.

The VIWS utilizes the Jameson-Caughey transonic inviscid wing program FLO27, to carry out the potential flow analysis. The boundary layer analysis is performed by a finite difference boundary layer prediction program developed by the Boeing Commercial Airplane Company. The basic theory behind the boundary layer program is contained in [Ref. 3]. A detailed description of the VIWS of programs (excluding the potential flow program FLO27) is contained in [Ref. 4]. A basic guide to the use of the VIWS of programs is contained in [Ref. 5].
II. POTENTIAL FLOW PROGRAM FLO27

Because of the extensive length and number of program modules in the VIWS, the Potential Flow Program, FLO27, was singled out for conversion. It was anticipated that FLO27 would be run separately at first and recombined with the other program modules at some later date when these modules were themselves translated for execution on the IBM 3033 computer.

A. RE-PROGRAMING

The Potential Flow Program, hereafter called FLO27, was received on magnetic tape and loaded into the IBM 3033 mass storage system using the Job Control Language (JCL) routines presented in Appendix A. The magnetic tape contained twenty (20) total files in which the format was 9 track, 1600 CPI, unlabeled. The card image format for the sixteen (16) program files is 80 characters per record and the four (4) output files contain 150 characters per record. The program and output files on the original CDC tape are listed in Table II.

The FLO27 program was converted to FORTRAN IV extended suitable for execution on the IBM computer using the NPS CDC
TABLE II
CDC Magnetic Tape Files

<table>
<thead>
<tr>
<th>File/Records</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 /2356</td>
<td>FLO27</td>
<td>Potential Flow Program</td>
</tr>
<tr>
<td>2 /3194</td>
<td>A411IN</td>
<td>Reads geometry &amp; velocity data constructs coordinate system</td>
</tr>
<tr>
<td>3 / 378</td>
<td>VWIN</td>
<td>Potential flow boundary-layer interface</td>
</tr>
<tr>
<td>4 /6611</td>
<td>A411A01</td>
<td>Three-dimensional boundary layer program</td>
</tr>
<tr>
<td>5 /1977</td>
<td>INTERP</td>
<td>Boundary-layer potential flow interface</td>
</tr>
<tr>
<td>6 / 668</td>
<td>A411PS</td>
<td>Streamline plots</td>
</tr>
<tr>
<td>7 / 211</td>
<td>A411P1</td>
<td>One-dimensional plots</td>
</tr>
<tr>
<td>8 / 586</td>
<td>A411P2</td>
<td>Contour plots</td>
</tr>
<tr>
<td>9 / 70</td>
<td>COUPLE</td>
<td>Procedure files</td>
</tr>
<tr>
<td>10 /158</td>
<td>ITER</td>
<td></td>
</tr>
<tr>
<td>11 / 7</td>
<td>DATAIN</td>
<td></td>
</tr>
<tr>
<td>12 / 78</td>
<td>FINAL</td>
<td></td>
</tr>
<tr>
<td>13 / 434</td>
<td>BOEB1</td>
<td>Boeing McLean computer program</td>
</tr>
<tr>
<td>14 / 36</td>
<td>CONTPLT</td>
<td>Contour plots</td>
</tr>
<tr>
<td>15 / 17</td>
<td>CORDPLT</td>
<td>One-dimensional plots</td>
</tr>
<tr>
<td>16 / 40</td>
<td>STREPLT</td>
<td>Streamwise plots</td>
</tr>
<tr>
<td>17</td>
<td>OUTF27</td>
<td>Output from FLO27</td>
</tr>
<tr>
<td>18</td>
<td>OUTIPC</td>
<td>Output from VWIN</td>
</tr>
<tr>
<td>19</td>
<td>OUT411L</td>
<td>Output from boundary-layer, lower surface</td>
</tr>
<tr>
<td>20</td>
<td>OUT411U</td>
<td>Output from boundary-layer, upper surface</td>
</tr>
</tbody>
</table>

to IFM conversion guide [Ref. 6]. The first step taken consisted of program compilation using the WATFIV compiler with its extended error messages. The listing which was
produced flagged all areas of the program which required revision. Program changes were accomplished utilizing this WATFIV listing. Some of the more general and repetitive changes are listed in Table III.

**TABLE III**

**FLO27 Re-Programming Changes**

<table>
<thead>
<tr>
<th>CDC Code</th>
<th>IBM Code Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variables: FREAD, FREAD, FWRITE, FWRITE, IREAD, IREAD, IWRITE, IWRITE</td>
<td>Eliminated from program</td>
</tr>
<tr>
<td>WRITE(IWRITE, 600)</td>
<td>WRITE(6, 600)</td>
</tr>
<tr>
<td>READ(IREAD, 500)</td>
<td>READ(5, 500)</td>
</tr>
<tr>
<td>READ 7, WRITE 7 or REWIND 7</td>
<td>Changed to READ 14, WRITE 14 or REWIND 14</td>
</tr>
<tr>
<td>Call SECCNE(T)</td>
<td>Step eliminated</td>
</tr>
<tr>
<td>Call SSWITCH(1, ISTCP)</td>
<td>Call SLITET(1, ISTOP)</td>
</tr>
<tr>
<td>Delimiter of form *</td>
<td>Replaced by `</td>
</tr>
<tr>
<td>Comment cards with *</td>
<td>Replaced by C</td>
</tr>
<tr>
<td>LEVEL statement</td>
<td>Step eliminated</td>
</tr>
<tr>
<td>If(UNIT(N).GT.0.) GO TC</td>
<td>All of this type eliminated</td>
</tr>
</tbody>
</table>

The most difficult change to make occurred with the CDC Buffer IN or Buffer OUT statements which were used in the program to transfer portions of a three-dimensional array into and out of main memory. The Buffer routines reduce the memory size required to execute the program. This statement type occurred in the main program and several of the subroutines.
The change required to translate this statement is presented below with the CDC code preceding the IBM FORTRAN.

\begin{verbatim}
BUFFER OUT (N3,1) (G(1,1,1),G(MX,MY,1)) changed to
WRITE(N3) ((G(I,J,1),I=1,MX),J=1,MY) and
BUFFER IN (N1,1) (G(1,1,M),G(MX,MY,M)) changed to
READ(N1,ERR= ) ((G(I,J,M),I=1,MX),J=1,MY)
\end{verbatim}

The variable ERR was assigned the GO TO statement number of the UNIT statement immediately following the BUFFER IN line of code. As an example, if the UNIT statement following the BUFFER IN code was - If(UNIT(N1).GT.0.) GO TO 151, then the number 151 was assigned to variable ERR following the equal sign. All CDC UNIT statements were eliminated from the FLO27 source code per Table II.

In addition to the program changes required to run FLO27 on the IBM computer, several lines of code were added to modify the output format to a more usable form. A subroutine, VERTEC, which calls the Versatec plotter was also added to enhance program usefulness. This plotting routine is user controlled through an input variable and is explained in the next section. The modified FLO27 program source code is presented in Appendix E.
To facilitate program data entry several input variables which had recommended values were initialized to these values within the Main program and the subroutine GEOM. The initialized input variables and their values are presented in Table IV.

**TABLE IV**

Initialized Input Variables

<table>
<thead>
<tr>
<th>AREA</th>
<th>VARIABLE NAME</th>
<th>INITIALIZED VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAIN Prgm.</td>
<td>XSCAL</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>PSCAL</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>FCONT</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>P20</td>
<td>0.7</td>
</tr>
<tr>
<td></td>
<td>P30</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>FSMCO</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>PTMAP</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>BLCP</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>WEIG</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>PTCK</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>FIX</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>YSYM</td>
<td>0.0</td>
</tr>
<tr>
<td>Subrt. GEOM</td>
<td>FNB</td>
<td>2.0</td>
</tr>
<tr>
<td></td>
<td>PX</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>PZ</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>TRL</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>SLT</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>XSING</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>YSING</td>
<td>0.0</td>
</tr>
</tbody>
</table>
A complete description of each input variable in Table IV can be found on pages 19 through 23 of [Ref. 5].

B. PROGRAM DESCRIPTION

The FLC27 program is a computer code written to analyze the transonic flow over a wing alone or a wing on a cylindrical fuselage. It uses a finite-volume formulation to solve the exact potential flow equation in conservative form. In the development of the equations, the basic assumptions are: steady flow, no heat or work transfer, isentropic flow, irrotational flow, no body forces and a perfect gas. The velocity vector in cartesian coordinates is

\[ \mathbf{V} = u \mathbf{i} + v \mathbf{j} + w \mathbf{k} \]  \hspace{1cm} (2.1)

where u, v and w are the velocity components. The continuity equation, assuming steady flow, is

\[ \frac{\partial}{\partial x} (\rho u) + \frac{\partial}{\partial y} (\rho v) + \frac{\partial}{\partial z} (\rho w) = 0 \]  \hspace{1cm} (2.2)

Next a velocity potential is introduced such that the velocity components are calculated as the gradient of this potential.

\[ u = \phi_x , \quad v = \phi_y , \quad w = \phi_z \]  \hspace{1cm} (2.3)
With the introduction of the velocity potential, the continuity equation 2.2 becomes

\[ \frac{\partial}{\partial x}(\rho \phi_x) + \frac{\partial}{\partial y}(\rho \phi_y) + \frac{\partial}{\partial z}(\rho \phi) = 0 \]  \hspace{1cm} (2.4)

Assuming no heat or work transfer, the energy equation can be written as

\[ T \left[ 1 + \frac{(\gamma - 1)}{2} \frac{M^2}{M} \right] = T_{\infty} \left[ 1 + \frac{(\gamma - 1)}{2} \frac{M^2}{M_{\infty}} \right] \]  \hspace{1cm} (2.5)

The flow is assumed to be uniform in the far field. On the surface of the body, the normal velocity component is zero. The velocities and densities of the near field are normalized using the free stream velocity and density, thus \( v_{\infty} = 1 \) and \( \rho_{\infty} = 1 \). Using the assumptions that the flow is isentropic and a perfect gas, the energy equation 2.5 can be shown to be

\[ \rho = \left[ 1 + \frac{(\gamma - 1)}{2} \frac{M^2}{M_{\infty}} (1 - v^2) \right] \frac{1}{\gamma - 1} \]  \hspace{1cm} (2.6)

With equations 2.5 and 2.6 there are only two unknowns, \( \phi \) and \( \rho \). They can be solved, subject to the boundary condition of flow tangency, using a finite volume technique. The basic numerical scheme for the solution is the
construction of a mesh from small volume elements (cubes) which are packed around the wing or wing-body configuration. The cubes in the computational domain are separately mapped to distorted cubes in the physical domain by independent transformations from local coordinates \( X, Y \) and \( Z \) to Cartesian coordinates \( x, y \) and \( z \). The mesh points are the vertices (corners) of the mapped cubes. The velocity potential and density are calculated at each vertex in the mesh. The pressure distribution can then be calculated from

\[
P = \frac{\varphi}{\gamma M_{\infty}^2}
\]  

(2.7)

In the event that the local flow velocity becomes supersonic and shocks occur, these are handled in the usual manner by insuring that:

1) The tangential velocity components are equal on each side of the shock.

2) Continuity is maintained by keeping the product of \( \rho U_n \) constant across the shock (where \( U_n \) is the normal velocity component).

3) Discontinuous expansions (corresponding to an "expansion shock") are excluded from the flow field.
The assumption of isentropic flow along with the existence of shocks presents a contradiction which can only be resolved by limiting the flow to very weak shocks for which entropy and vorticity generation may be ignored. Thus, solutions will be valid only for subsonic free stream velocities.

The main three-dimensional array containing the potential function data is stored on disk, and special unformatted input/output statements are used to bring planes of data into central computer memory and to store updated planes of data back on the disk. In the construction of the computational coordinate system, a Joukowski transformation is used to transform the cylindrical fuselage to a vertical slit and then a sheared parabolic transformation is used in planes containing the airfoil sections. A detailed mathematical formulation of the potential flow analysis is contained in [Ref. 7].

1. **Program Input**

The input to FLO27 consists of variables which are read with an *F10.6* FORMAT. Each input card has a title card which precedes it. This title card contains the input variable name and effectively labels the input data for easy
reference. The title for each input data is placed in the same column as the input data it labels. The title cards are read with a 20A4 FORMAT. All numerical input values are real numbers. The following data deck, listed card by card, is the minimum input data required for a simple wing analysis. Each "card" can be interpreted as one line of data on your terminal. A complete sample data set is presented in Appendix C.

CARD 1 The Run Title (64 characters maximum)

CARD 2 Title card for the input variables

FNX, FNY, FNZ, FMESH and PPLOT

CARD 3

Cols. 1-10 FNX - Number of computational cells in the chordwise direction for the initial mesh.

\[ \text{MAX} = \frac{160}{2^n}, \quad \text{where} \quad n = \text{FMESH} - 1. \]  
(See Cols. 31-40 for FMESH)

Cols. 11-20 FNY - Number of computational cells in the normal direction from the airfoil surface for the initial mesh.

\[ \text{MAX} = \frac{16}{2^n}, \quad \text{where} \quad n = \text{FMESH} - 1. \]

Cols. 21-30 FNZ - Number of computational cells in the spanwise direction for the initial mesh.

\[ \text{MAX} = \frac{32}{2^n}, \quad \text{where} \quad n = \text{FMESH} - 1. \]
Cols. 31-40 FMESH - Determines the number of times a program generated computational mesh is refined. Enter only 1.0, 2.0 or 3.0 for coarse, medium or fine mesh. If 3.0 is selected the program will calculate flow over the wing for the coarse mesh then half the mesh size (medium), recalculate, then half the mesh again (fine) and do a final potential flow calculation. Output parameters are printed for each mesh size for which calculations were performed.

Cols. 41-50 FFLOT - Output flag

0.0 = Normal output without printer-plot of Cp
1.0 = Normal output with printer-plot of Cp at each computational mesh point for each wing section.
2.0 = Normal output with Versatec plots of Cp versus X/C for each wing section of the final mesh.

CARD 4 Title card for the input variables
FIT, COVO and P10

CARD 5-M One card for each computational mesh. Total number of cards equal to M = FMESH.
Cols. 1-10 FIT - A parameter which fixes the maximum number of iterations the program will use to converge the velocity potential to a specified tolerance (COVO). This parameter must be repeated for each mesh refinement.

Cols. 11-20 COVO - Velocity potential convergence criteria. This input variable is also entered for each selected mesh. A value of 0.000001 is recommended.

Cols. 21-30 P10 - This parameter determines the subsonic point relaxation factor for the specified mesh size. A value of less than 2.0 must be entered for each designated mesh. Recommended values are: 1.6 for coarse, 1.3 for medium and 1.2 for the fine mesh.

CARD 6 Title card for the input variables
FMACH, YA, AL and CDO

CARD 7
Cols. 1-10 FMACH - Free stream Mach number
Cols. 11-20 YA - Yaw angle in degrees
Cols. 21-30 AL - Angle of attack in degrees
Cols. 31-40 CD0 - Drag coefficient due to skin friction. Unless known, an estimated value of 0.01 is recommended.

CARD 8  Title card for the input variables
ZSYM, FNS, SWEEP, DIHED and FUS

CARD 9

Cols. 1-10 ZSYM - The wing planform symmetry trigger.

0.0 = Yawed wing, has no spanwise symmetry
1.0 = Swept wing, has spanwise symmetry

Cols. 11-20 FNS - This input variable tells the program the total number of wing sections you have selected to define the wing half span. The number must be at least three (3) but not more than eleven (11) sections.

Cols. 21-30 SWEEP - Leading edge sweep angle in degrees.

Cols. 31-40 DIHED - Dihedral angle in degrees. See Fig. 2.1.

Cols. 41-50 FUS - Input the fuselage radius. Enter 0.0 for a wing-alone case.
Data input cards from 10 through 15 are used for defining wing planforms and section geometrics. For the first wing section, all data cards from card 10 through card 15 must be used. For the second and subsequent sections there is an option for skipping the wing section defining data (cards 12 through 15) and copying the data from that of the previous section. This option is controlled by the input variable FSEC. If this option is not used, data cards from 10 through 15 must be repeated for each wing defining section. The number of wing sections which are defined is input with the variable FNS. Remember, up to 11 sections may be defined, and a minimum of 3 sections must be defined. All wing planform and section defining geometrics must be in consistent units. Wing planform and section defining quantities are presented in Fig. 2.2.
Figure 2.2. Wing Defining Geometry

CARD 10 Title card for the input variables

ZS, XI, YL, CHORD, THICK, AT and FSEC

CARD 11

Cols. 1-10 ZS - The section spanwise coordinate

(Start at the centerline and work outboard)

Cols. 11-20 XL - Section leading edge X coordinate

Cols. 21-30 YL - Section leading edge Y coordinate

Cols. 31-40 CHORD - Section chord length
Ccols. 41-50 THICK - The thickness scaling factor can be used to scale all Y coordinates of the wing section. Thus percent thickness and camber are increased (or decreased) accordingly. Use 1.0 if no scaling is desired.

Ccols. 51-60 AT - The twist angle of each section (geometric twist) measured from the X axis to the chord line. A positive twist angle reduces the section angle of attack and gives "washout". Use 0.0 for no twist.

Ccols. 61-70 FSEC - This is a flag which determines whether or not the program reads wing section defining geometry from a previous wing section or from new defining geometry. For the first section defined you must set FSEC to 1.0. Following the first section, if you define new section geometry then use FSEC = 1.0. If you want the program to read the section geometry defined from the previous section then set FSEC = 0.0.

CARD 12 Title card for the input variable FN
CARD 13

Ccls. 1-10 FN - This variable contains the number of points which define the upper and lower surface of the section. A maximum of 161 points may be used.

CARD 14

Title card for the input variables

XF(I) and YP(I)

CARDS 15-1 to 15-N Total number of cards equals N,

where N = integer part of \((FN+2)/3\).
The X and Y coordinates at each point are entered in pairs, three points to a card. (See Appendix C for sample input)

Ccls. 1-10 XP(I) - X coordinate of the wing section point

11-20 YP(I) - Y coordinate of the wing section point

21-30 defining X coordinate for next point

31-40 defining Y coordinate for next point

41-50 defining X coordinate for following point
defining $Y$ coordinate for following point

The $X$ and $Y$ coordinates of the wing section defining points must be entered starting with the upper surface trailing edge point and proceeding along the upper surface to the leading edge, and returning along the lower surface to the lower surface trailing edge point. It is very important to define the section leading edge with a large number of closely spaced points. Suggest at least 0.05 spacing or less between $X$ coordinate values from $0.1 \, X/C$ to the leading edge, $X/C = 0.0$. Each $X$ and $Y$ coordinate point is normalized using the chord length for that section. Section defining geometrics are illustrated in Fig. 2.3.

![Figure 2.3. Section Defining Geometry](image-url)
CARD 16  Title card containing the words in Cols. 1-90

END OF CALCULATIONS

CARD 17  Title card for the input variable

FNX

CARD 18

Cols. 1-10 FNX - This variable indicates the end of a set of calculations and must be set equal to 0.0. Its purpose is to indicate that the program has run to completion.

2. **Program Output**

Output from the PLO27 program varies with the value of the input variable FPLCT. When FPLCT is set equal to 0.0 a normal output is produced. This normal output contains (in order of occurrence): refined input geometry data including trailing edge slope and angle calculations; iterative solution of the potential flow mesh; section characteristics and wing characteristics. The iterative solution, section and wing characteristic data are repeated for each mesh refinement requested. Thus, if the input variable FMESH is set equal to 3, these data are calculated and output three times. The last data in the normal output consists of the non-dimensionalized chord (X/C) and pressure
coefficient \((C_p)\) data at each computational mesh point for each wing section calculated during the final mesh. A sample of the normal output data is presented in Appendix D and represents the output data from Appendix C input data.

If variable \(F\)PLOT is set equal to 1.0, the output data is increased considerably. This output contains the normal output plus a line printer-plot of the pressure coefficient at each computational mesh point for each wing section. The line printer-plot is produced for each wing section of each mesh refinement. The length of the output data with \(F\)PLOT set equal to 1.0 can approach 6000 records depending on the number of mesh refinements requested. These plots are of questionable value and, therefore, an alternate plotting program was developed.

When the variable \(F\)PLOT is set equal to 2.0, the normal output data is produced plus a Versatec plotting subroutine (VERTEC) is called. The subroutine outputs, via the Versatec plotter, plots of \(C_p\) versus \(X/C\) for each wing section of the final mesh calculations. This routine is simply putting into plot form the \(C_p\) and \(X/C\) numerical data contained in the normal output. A sample of the Versatec plot is presented in Fig. 2.4.
**SECTION CP DATA**

- * = UPPER SURFACE
- + = LOWER SURFACE

---

**Figure 2.4.** Versatec Plot of Cp vs. X/C

- SPAN STATION = 2.550
- MACH 0.800  YAW 0.000  AOA 2.000
- CL 0.34675  CD 0.02234  CM 0.00123

---

34
III. **INTERACTIVE INPUT PROGRAM FLO27IN**

The input data file required for the FLO27 program is extensive. Errors in input data FORMAT will cause program errors at execution time. In order to eliminate these errors and reduce the input data workload, a computer terminal interactive program was written. This interactive program, called FLO27IN, is a user-friendly way of creating an input data file for the potential flow wing analysis program FLO27. The FLO27IN program source code is presented in Appendix F.

The interactive program, FLO27IN, when executed asks questions of the user in order to construct and write to the user's "A" disk the required FLO27 input data file. The following presents the step-by-step procedure for executing the interactive program FLO27IN.

**STEP #1---**Log on to any IBM 3033 interactive terminal with your user number and password.

**STEP #2---**Once logged on and in the CMS operation mode type:

```
CP LINK 0247P 191 120 RR then hit ENTER
```

**STEP #3---**The word PASSWORD will appear, Type and ENTER

AERO

35
STEP #4---Type and ENTER

ACC 120 D

STEP #5---Type and ENTER

LOAD FLO27IN (START)

The screen will display the header for the interactive program. Answer each question presented. At the end of each question in parenthesis is the input data variable associated with that question and whether the input parameter is a real number (R) or an integer (I). Example:

==> Enter the free stream Mach number (FMACH): (R). FMACH is the input data variable for the question. As you proceed through the FLC27IN program, opportunities to review and change input data will be presented. Should it become necessary to change your input data after completing the FLO27IN program, you can simply XEDIT the created data file.

The FLO27IN program also incorporates a library which contains the wing-section defining data for a number of current wing shapes. A copy of this library is presented in Appendix B. This feature will be displayed during program execution by the use of a menu from which the user can select a pre-defined wing section or define his own.
Upon completion of user inputs to the interactive program three additional data lines are automatically written to the bottom of the input file. They are:

END OF CALCULATION

PNX

0.0

In addition, Job Control Language (JCL) cards are written to the top and bottom of the file. All JCL cards start with a // format. After FLO27IN has run to completion type and enter RELEASE 191 to release the aero disk which was linked while executing the FLO27IN program. The created data file is written to the user's "A" disk with <filename> <filetype> of FLO27 DATAIN. Additional changes can be made simply by entering the XEDIT mode and editing the file.
IV. **FLO27 BATCH SYSTEM EXECUTION**

The potential flow program FLO27 can be executed after the input data file has been created. The batch processor is required for FLO27 execution because of the extensive CPU time needed to run the program. While in the XEDIT mode, a standard JOB card must be added to the top of the FLO27 DATAIN file prior to submission for job execution. The JOB card has the form:

```
//j0name JOB (nnnn, pppp),'ident',CLASS=J
```

`j0name` may contain up to 8 alphanumeric characters, the first of which must be alphabetic.

`nnnn` = your user number

`pppp` = project number, assigned by professor

`'ident'` = contains the user's own identification information. A maximum of 20 characters may be contained within the single quotation marks.

After adding the JOB card to your data file, you are ready to execute the program. Type `SUBMIT FLO27 DATAIN` and press ENTER. Batch runs are normally not worth waiting for. To inquire about the status of the job, enter `INQ` and the job name used on the JOB card or "logoff". If the system is busy and the maximum mesh size was selected, it may be several hours before your job is run.

38
When the job is run the output will be spooled to the batch printer located next to the VM printer in the main computer building. The title at the top of the printout for batch jobs is the name entered on the JOB card. If it is desired to have the program output data spooled directly to the terminal, it will be necessary to add one additional JCL card to the input data set. This card must be placed immediately following the JOB card and has the form:

```//*MAIN ORG=NPGVM1.nnnnP```

where nnnn = your user number

Inserting this card in the input data will cause all program output to be spooled to the user's virtual reader where it may be looked at, printed or transferred to his "A" disk. To enquire as to whether information is in the reader simply type RDR and hit enter, then follow the instructions on the screen.
V. PROGRAM TEST RESULTS

The FLC27 program was tested in three stages; (1) during the reprogramming phase for conversion completeness, (2) after successful conversion with suitable wing data for program accuracy and (3) during an AE-4501 class project.

A. ACCEPTANCE TEST DATA

To test and ensure that the FLO27 program was converted to IEM compatible Fortran without error, an acceptance test data set was used. The acceptance test input and output data was supplied with the original CDC program source code. After conversion of the FLO27 program to Fortran suitable for the NPS IBM system, the acceptance test input data were run and the output results compared to the output generated by the CDC system.

Both output data sets were numerically exact when the FLO27 program was run in double precision on the IBM system. If the program was run in single precision, the numerical output values were exact to the third decimal place. The difference in single precision accuracy occurs because the CDC system uses a 64 bit word length while the IBM system word length in single precision is only 32 bits. It was
decided that the IBM single precision accuracy was satisfactory.

B. COMPARISON WITH OTHER PROGRAMS

The FLO27 program was also tested for accuracy by using the wing planform and section data from a NACA 572 wing. The data were run on both the FLO27 program and the Douglas potential flow program [Ref. 1]. The data generated by both programs was compared to wind tunnel data for the NACA 572 wing [Ref. 8]. The results are presented in Fig. 5.1 as plots of lift coefficient versus angle-of-attack. The results show that for the NACA 572 wing the FLO27 program more accurately predicts the wing lift coefficient than does the Douglas program.

C. AE-4501 CLASS PROJECT

The final test phase was conducted by introducing the FLO27 program into the AE-4501 course as a class project. This was accomplished to determine student problems/comments concerning the data input program FLO27IN and to test an additional wing shape. The wing chosen for study was that of the A-7 airplane. The A-7 wing has a distinct leading edge notch at the approximate mid-span. When the planform geometry was run with the notch included the FLO27 program
The program ran to completion but gave negative values for section and total induced drag coefficient. The value for the lift coefficient was low for the freestream Mach and angle-of-attack used. It was found that if the notch was...
excluded from the wing geometry input data the program results were satisfactory both for induced drag and lift coefficient.

From the AE-4501 class experience it was determined that sharp wing planform discontinuities cannot be handled by the program. If however, the changes in shape are gradual, such as a wing glove, the program output appears to be satisfactory. Such was the case with the acceptance test case data where the wing geometry was that of the F-8 supercritical wing which incorporates a wing glove.
APPENDIX A

C This JCL routine allocates sufficient space on the mass storage system to store the entire tape contents.
//JACK JOB (3266,0178), "PASCHALL-2759", CLASS=A
/* MAIN ORG=NEGVM1.3266P */
EXEC PGM=IEFBR14
DD1 DD UNIT=3330V,MSVGP=PUB4C DISP=(NEW,CATLG),/* DSN=MSS.S3266.WFLOW.DATA,SPACE=(CYL,(16,4,2))*/

C This JCL routine is used to transfer all tape files to a partitioned data set in the mass storage system.
//JACK JOB (3266,0178), "PASCHALL-2759", CLASS=J
/* MAIN ORG=NEGVM1.3266P */
EXEC PGM=IEBGENER
SYSPRINT DD SYSOUT=A
SYSIN DD DUMMY
SYSUT1 DD UNIT=3400-6,VCL=SER=WFLOW,DISP=(OLD,PASS),/Label=(&FILE,BLP,,IN)
DCB=(RECFM=F,BLKSZ=80,DEN=3,OPTCD=Q)
SYST2 DD DISP=(OLD,KEEP),DSN=MSS.S3266.WFLOW.SOURCE(&MEM),/DCB=(RECFM=FB,LRECL=80,BLKSZ=6400)
PEND
EXEC COPY,FILE=1, MEM=FL027
EXEC COPY,FILE=2, MEM=A411N
EXEC COPY,FILE=3, MEM=WIN
EXEC COPY,FILE=4, MEM=A411A01
EXEC COPY,FILE=5, MEM=INTER
EXEC COPY,FILE=6, MEM=A411PS
EXEC COPY,FILE=7, MEM=A411P1
EXEC COPY,FILE=8, MEM=A411P2
EXEC COPY,FILE=9, MEM=COUPLE
EXEC COPY,FILE=10, MEM=ITER
EXEC COPY,FILE=11, MEM=DATAIN
EXEC COPY,FILE=12, MEM=FINAL
EXEC COPY,FILE=13, MEM=BOEB1
EXEC COPY,FILE=14, MEM=CONTPLT
EXEC COPY,FILE=15, MEM=CORDPLT
EXEC COPY,FILE=16, MEM=STREPLT
COPY2 PROC FILE=,MEM=,LRECL=80,BLK=6400
EXEC PGM=IEBGENER
SYSPRINT DD SYSOUT=A
SYSIN DD DUMMY
SYSUT1 DD UNIT=3400-6,VOL=SER=WFLOW,DISP=(OLD,PASS),/LABEL=(&FILE,BLP,,IN)
DCB=(RECFM=F,BLKSZ=&LRECL,DEN=3,OPTCD=Q)
SYST2 DD DISP=(OLD,KEEP),DSN=MSS.S3266.WFLOW.DATA(&MEM),/DCB=(RECFM=FB,LRECL=&LRECL, BLKSZ=&BLK)
PEND
EXEC COPY2,FILE=17,LRECL=150,BLK=6000,MEM=OUTF27
EXEC COPY2,FILE=18,LRECL=150,BLK=6000,MEM=OUTIFC
EXEC COPY2,FILE=19,LRECL=150,BLK=6000,MEM=OUT411L
EXEC COPY2,FILE=20,LRECL=150,BLK=6000,MEM=OUT411U
*/
This JCL routine moves all source code files from mass storage to the MVS 004 disk which can be accessed by entering GET MVS then following the screen instructions to move source files to your disk. If you want to move the data files to MVS 004 then change the word SOURCE to DATA in the JCL program below.

```
//JACK JOB (3266.0178), 'FASCHALL=2759', CLASS=A
//*MAIN ORG=NPGM1.3266P
// EXEC PGM=IEBCOPY
// SYSPRINT DD SYSOUT=A
// FROM DD DISP=SHR, DSN=MSS.S3266.WPLLOW.SOURCE
// INTO DD UNIT=3350, VOL=SER=MVS.004, DISP=(NEW,KEEP),
// SPACE=(CYL,(16,4,10),RLSE), DSN=S3266.SOURCE
// SYSUT3 DD UNIT=SYSDA, SPACE=(CYL,(2,2))
// SYSUT4 DD UNIT=SYSDA, SPACE=(CYL,(2,2))
// SYSIN DD *
// COPY OUTDD=INTC, INDD=FFCM
/*
*/
APPENDIX B

LIBRARY OF AIRFOIL SECTION GEOMETRIES

0 = user input section coordinate data
1 = flat plate data
2 = symmetrical wing (11% thickness at 30% chord)
3 = supercritical wing (cambered, 12% thickness at 32% chord)
4 = NACA 24-30-0 (cambered, 12% thickness at 30% chord)
5 = F-14 wing (cambered, 9.5% thickness at 37% chord)
6 = A-7 wing (7 deg droop at 20% chord, 7% thickness at 43% chord)
7 = LISSAMAN 7769 Airfoil (cambered, 11% thickness at 30% chord)
8 = NACA 0010 (symmetrical, 10% thickness at 30% chord)
9 = NACA 0010-34 (symmetrical, 10% thickness at 40% chord)
10 = NACA 0010-35 (symmetrical, 10% thickness at 50% chord)
11 = NACA 0010-64 (symmetrical, 10% thickness at 40% chord)
12 = NACA 0010-66 (symmetrical, 10% thickness at 60% chord)
13 = NACA 16-009 (symmetrical, 9% thickness at 50% chord)
14 = NACA 63-010 (symmetrical, 10% thickness at 35% chord)
15 = NACA 63A010 (symmetrical, 10% thickness at 35% chord)
16 = NACA 64-010 (symmetrical, 10% thickness at 40% chord)
17 = NACA 64A010 (symmetrical, 10% thickness at 40% chord)
18 = NACA 65-010 (symmetrical, 10% thickness at 40% chord)
19 = NACA 65A010 (symmetrical, 10% thickness at 40% chord)
20 = NACA 66-010 (symmetrical, 10% thickness at 45% chord)
APPENDIX C

THIS APPENDIX PRESENTS A COMPLETE INPUT DATA SET INCLUDING THE JCL 
CARDS REQUIRED TO EXECUTE THE PROGRAM FLC27

// (STANDARD JOB CARD - SEE MVS USER'S GUIDE NO. MVS-01)
// EXEC FLC27
// GO SYSLN CC *

SAMPLE DATA (NACA 572 WING SECTION)

<table>
<thead>
<tr>
<th>FNX</th>
<th>FNY</th>
<th>FZ</th>
<th>FMESH</th>
<th>FFLUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>40.0</td>
<td>4.0</td>
<td>6.0</td>
<td>3.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

FIT

<table>
<thead>
<tr>
<th>COVC</th>
<th>P10</th>
</tr>
</thead>
<tbody>
<tr>
<td>100.0</td>
<td>0.990001</td>
</tr>
<tr>
<td>20.0</td>
<td>0.990001</td>
</tr>
<tr>
<td>5.0</td>
<td>0.990001</td>
</tr>
</tbody>
</table>

FMACF

<table>
<thead>
<tr>
<th>YA</th>
<th>AL</th>
<th>CD0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.10</td>
<td>0.0</td>
<td>2.0</td>
</tr>
</tbody>
</table>

ZSYM

<table>
<thead>
<tr>
<th>FNS</th>
<th>SHEEP</th>
<th>DIFED</th>
<th>FUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>3.0</td>
<td>32.3275</td>
<td>0.0</td>
</tr>
</tbody>
</table>

ZS

<table>
<thead>
<tr>
<th>XL</th>
<th>YL</th>
<th>CHORD</th>
<th>THICK</th>
<th>AT</th>
<th>FSEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>8.6000</td>
<td>1.0000</td>
<td>0.0</td>
<td>1.0</td>
</tr>
</tbody>
</table>

FN

<table>
<thead>
<tr>
<th>XP(I)</th>
<th>YP(I)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0000000</td>
<td>0.0013000</td>
</tr>
<tr>
<td>0.8000000</td>
<td>0.3760000</td>
</tr>
<tr>
<td>0.5000000</td>
<td>0.5360000</td>
</tr>
<tr>
<td>0.2500000</td>
<td>0.7660000</td>
</tr>
<tr>
<td>0.1000000</td>
<td>0.9660000</td>
</tr>
<tr>
<td>0.0500000</td>
<td>0.9800000</td>
</tr>
<tr>
<td>0.0250000</td>
<td>0.9900000</td>
</tr>
<tr>
<td>0.0125000</td>
<td>0.9960000</td>
</tr>
<tr>
<td>0.0062500</td>
<td>0.9980000</td>
</tr>
<tr>
<td>0.0031250</td>
<td>0.9990000</td>
</tr>
<tr>
<td>0.0015625</td>
<td>0.9999000</td>
</tr>
</tbody>
</table>

ZS

<table>
<thead>
<tr>
<th>XL</th>
<th>YL</th>
<th>CHORD</th>
<th>THICK</th>
<th>AT</th>
<th>FSEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0675</td>
<td>6.1307</td>
<td>0.0</td>
<td>0.4500</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
</tbody>
</table>

ZS

<table>
<thead>
<tr>
<th>XL</th>
<th>YL</th>
<th>CHORD</th>
<th>THICK</th>
<th>AT</th>
<th>FSEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>19.3750</td>
<td>12.2614</td>
<td>0.0</td>
<td>0.3000</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
</tbody>
</table>

END OF CALCULATION

FNX

| 0.0 |

/*
//
APPENDIX D

THIS APPENDIX PRESENTS THE FLG27 OUTPUT DATA PRODUCED FROM THE INPUT DATA OF THE PREVIOUS APPENDIX.

A46C MODIFIED FROM FLG27 OF ANTONY JAMESC, CURANT INSTITUTE THREE DIMENSIONAL WING ANALYSIS IN TRANSONIC FLOW USING FINITE VOLUME SCHEME NACA 572 WING SECTION FUSELAGE RAC

SWEEP LIHED
0.0 0.0

PROFILE AT Z = 0.0
TE ANGLE TE SLOPE X SING Y SING
16.0833 -0.0720 0.0024 -0.0000

NL = 21, XF(NL) = 0.0

(XP,YP)

<table>
<thead>
<tr>
<th>XP</th>
<th>YP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0000</td>
<td>-0.00130</td>
</tr>
<tr>
<td>0.6000</td>
<td>-0.00260</td>
</tr>
<tr>
<td>0.2000</td>
<td>0.00422</td>
</tr>
<tr>
<td>0.0300</td>
<td>0.00227</td>
</tr>
<tr>
<td>0.0300</td>
<td>0.02990</td>
</tr>
<tr>
<td>0.2000</td>
<td>0.07260</td>
</tr>
<tr>
<td>0.6000</td>
<td>0.66420</td>
</tr>
<tr>
<td>1.0000</td>
<td>0.00130</td>
</tr>
</tbody>
</table>

SECTION DEFINITION AT Z = 0.0

<table>
<thead>
<tr>
<th>XLE</th>
<th>YLE</th>
<th>CHORC</th>
<th>THICKNESS RATIO</th>
<th>TWIST</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>8.6000</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>YMIN</th>
<th>JMNX</th>
<th>JMAX</th>
<th>YDIF</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.4220E-01</td>
<td>10</td>
<td>0.7880E-01</td>
<td>.1210</td>
</tr>
</tbody>
</table>

SECTION DEFINITION AT Z = 9.68750

<table>
<thead>
<tr>
<th>XLE</th>
<th>YLE</th>
<th>CHORC</th>
<th>THICKNESS RATIO</th>
<th>TWIST</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1307</td>
<td>0.0</td>
<td>6.4500</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>YMIN</th>
<th>JMNX</th>
<th>JMAX</th>
<th>YDIF</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.4220E-01</td>
<td>10</td>
<td>0.7880E-01</td>
<td>.1210</td>
</tr>
</tbody>
</table>

SECTION DEFINITION AT Z = 19.37500

<table>
<thead>
<tr>
<th>XLE</th>
<th>YLE</th>
<th>CHORC</th>
<th>THICKNESS RATIO</th>
<th>TWIST</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.2614</td>
<td>0.0</td>
<td>4.3000</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>YMIN</th>
<th>JMNX</th>
<th>JMAX</th>
<th>YDIF</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.4220E-01</td>
<td>10</td>
<td>0.7880E-01</td>
<td>.1210</td>
</tr>
</tbody>
</table>
### SECTION CHARACTERISTICS

<table>
<thead>
<tr>
<th>MACH NO</th>
<th>YAW</th>
<th>ANG OF ATTACK</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.10000</td>
<td>0</td>
<td>2.0000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SPAN STATION</th>
<th>CL</th>
<th>CD</th>
<th>CM</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>.20238</td>
<td>.71088E-02</td>
<td>-9.1575E-01</td>
</tr>
<tr>
<td>3.87499</td>
<td>.24033</td>
<td>-2.4623E-02</td>
<td>-5.1798E-01</td>
</tr>
<tr>
<td>7.74599</td>
<td>.26515</td>
<td>-4.8115E-02</td>
<td>-1.6506</td>
</tr>
<tr>
<td>11.62498</td>
<td>.28127</td>
<td>-5.7074E-02</td>
<td>-1.6920</td>
</tr>
<tr>
<td>15.49998</td>
<td>.28695</td>
<td>-6.5451E-02</td>
<td>-1.6935</td>
</tr>
<tr>
<td>19.37497</td>
<td>.24168</td>
<td>-7.0588E-02</td>
<td>-8.6038E-01</td>
</tr>
</tbody>
</table>

### WING CHARACTERISTICS

<table>
<thead>
<tr>
<th>MACH NO</th>
<th>YAW</th>
<th>ANG OF ATTACK</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.10000</td>
<td>0</td>
<td>2.0000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MACH NO</th>
<th>YAW</th>
<th>ANG OF ATTACK</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.25575</td>
<td>-.32534E-02</td>
<td>.10000E-01</td>
</tr>
<tr>
<td>CM YAW</td>
<td>CM ROLL</td>
<td>CM PITCH</td>
</tr>
<tr>
<td>-.00467</td>
<td>.23919</td>
<td>-.33140</td>
</tr>
<tr>
<td>ITERATIVE SOLUTION</td>
<td>ANG CF ATTACK</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------</td>
<td></td>
</tr>
<tr>
<td>MACH NO</td>
<td>YAW</td>
<td>2.0000</td>
</tr>
<tr>
<td>NX</td>
<td>80</td>
<td>16</td>
</tr>
<tr>
<td>RELAX FCT 1</td>
<td>RELAX FCT 2</td>
<td>RELAX FCT 3</td>
</tr>
<tr>
<td>1.30000</td>
<td>1.00000</td>
<td></td>
</tr>
<tr>
<td>MAX CORRREC</td>
<td>AVG CORRREC</td>
<td>MAX RESIDAL</td>
</tr>
<tr>
<td>ITERATION</td>
<td>I</td>
<td>J</td>
</tr>
<tr>
<td>1</td>
<td>0.1442E-02</td>
<td>64</td>
</tr>
<tr>
<td>2</td>
<td>-0.1217E-02</td>
<td>22</td>
</tr>
<tr>
<td>3</td>
<td>0.5853E-03</td>
<td>22</td>
</tr>
<tr>
<td>4</td>
<td>-0.3149E-03</td>
<td>22</td>
</tr>
<tr>
<td>5</td>
<td>0.2258E-03</td>
<td>22</td>
</tr>
<tr>
<td>6</td>
<td>-0.1848E-03</td>
<td>22</td>
</tr>
<tr>
<td>7</td>
<td>-0.1431E-03</td>
<td>22</td>
</tr>
<tr>
<td>8</td>
<td>-0.1491E-03</td>
<td>22</td>
</tr>
<tr>
<td>9</td>
<td>-0.1368E-03</td>
<td>22</td>
</tr>
<tr>
<td>10</td>
<td>-0.1268E-03</td>
<td>22</td>
</tr>
<tr>
<td>11</td>
<td>-0.1174E-03</td>
<td>22</td>
</tr>
<tr>
<td>12</td>
<td>-0.1083E-03</td>
<td>22</td>
</tr>
<tr>
<td>13</td>
<td>-0.0999E-03</td>
<td>22</td>
</tr>
<tr>
<td>14</td>
<td>-0.0936E-03</td>
<td>22</td>
</tr>
<tr>
<td>15</td>
<td>-0.0851E-03</td>
<td>22</td>
</tr>
<tr>
<td>16</td>
<td>-0.0789E-03</td>
<td>22</td>
</tr>
<tr>
<td>17</td>
<td>-0.0736E-03</td>
<td>22</td>
</tr>
<tr>
<td>18</td>
<td>-0.0654E-03</td>
<td>22</td>
</tr>
<tr>
<td>19</td>
<td>-0.0659E-03</td>
<td>22</td>
</tr>
<tr>
<td>20</td>
<td>-0.0619E-03</td>
<td>22</td>
</tr>
</tbody>
</table>

MAX RESIDAL 1 | MAX RESIDAL 2 | WORK | RELUCTCYCLE | CONV | TOLERANCE |
| -0.1646E-02 | -0.2872E-03 | 19.0000 | 0.9122 | 0.1000E-05 |
### SECTION CHARACTERISTICS

<table>
<thead>
<tr>
<th>SPAN STATION</th>
<th>YAW</th>
<th>CL</th>
<th>CM</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>0</td>
<td>22.093</td>
<td>.18167E-01</td>
</tr>
<tr>
<td>1.93750</td>
<td>0</td>
<td>24.319</td>
<td>.53303E-02</td>
</tr>
<tr>
<td>3.87499</td>
<td>0</td>
<td>26.053</td>
<td>.14154E-02</td>
</tr>
<tr>
<td>5.81249</td>
<td>0</td>
<td>27.508</td>
<td>.44517E-05</td>
</tr>
<tr>
<td>7.74995</td>
<td>0</td>
<td>28.683</td>
<td>-.72064E-03</td>
</tr>
<tr>
<td>9.68749</td>
<td>0</td>
<td>29.643</td>
<td>-.11954E-02</td>
</tr>
<tr>
<td>11.62499</td>
<td>0</td>
<td>30.359</td>
<td>-.16262E-02</td>
</tr>
<tr>
<td>13.56248</td>
<td>0</td>
<td>30.863</td>
<td>-.20637E-02</td>
</tr>
<tr>
<td>15.49998</td>
<td>0</td>
<td>30.746</td>
<td>-.27385E-02</td>
</tr>
<tr>
<td>17.43747</td>
<td>0</td>
<td>29.141</td>
<td>-.41055E-02</td>
</tr>
<tr>
<td>19.37497</td>
<td>0</td>
<td>21.866</td>
<td>-.42033E-02</td>
</tr>
</tbody>
</table>

### WING CHARACTERISTICS

<table>
<thead>
<tr>
<th>MACH NO</th>
<th>YAW</th>
<th>CL</th>
<th>CD FORM</th>
<th>CD FRICTION</th>
<th>CD</th>
<th>L/D FORM</th>
<th>L/D</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.10000</td>
<td>0</td>
<td>87151E-03</td>
<td>1.0000E-01</td>
<td>1.0E72E-01</td>
<td>317.42</td>
<td>25.00</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MACH NO</th>
<th>YAW</th>
<th>CM RCLL</th>
<th>CM PITCH</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.00145</td>
<td>0.25615</td>
<td>-0.35963</td>
<td></td>
</tr>
</tbody>
</table>
## Iterative Solution

<table>
<thead>
<tr>
<th>Mach No</th>
<th>Yaw</th>
<th>Angle of Attack</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.10000</td>
<td>0</td>
<td>2.0000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Nx</th>
<th>Ny</th>
<th>Relax Fct 1</th>
<th>Relax Fct 2</th>
<th>Relax Fct 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>160</td>
<td>16</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Max CORRECN 1</th>
<th>J</th>
<th>K</th>
<th>Avg CORRECN</th>
<th>Max Residual 1</th>
<th>J</th>
<th>K</th>
<th>Work</th>
<th>RELUCTN/CYCLE</th>
<th>CONV TOLERANCE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.95314E-03</td>
<td>17</td>
<td>24</td>
<td>C.20565E-04</td>
<td>-0.13855E-02</td>
<td>2</td>
<td>16</td>
<td>24</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-0.96428E-03</td>
<td>17</td>
<td>23</td>
<td>C.12764E-04</td>
<td>0.98673E-03</td>
<td>16</td>
<td>15</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.42692E-03</td>
<td>17</td>
<td>23</td>
<td>0.96247E-05</td>
<td>0.87373E-03</td>
<td>16</td>
<td>15</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>-0.24824E-03</td>
<td>15</td>
<td>15</td>
<td>C.75284E-05</td>
<td>0.73407E-03</td>
<td>16</td>
<td>15</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.21125E-03</td>
<td>15</td>
<td>16</td>
<td>0.71493E-05</td>
<td>0.62240E-03</td>
<td>16</td>
<td>15</td>
<td>3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Max Residual 1 | Max Residual 2 | Work | Relaxn/Cycle | Conv Tolerance
-0.1385E-02 | 0.6224E-03 | 4.0000 | 0.8197 | 0.1000E-05

5
<table>
<thead>
<tr>
<th>SPAN STATION</th>
<th>C1</th>
<th>CD</th>
<th>ANG OF ATTACK</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>0.22536</td>
<td>0.25222E-01</td>
<td>CM</td>
</tr>
<tr>
<td>0.956875</td>
<td>0.23679</td>
<td>0.13934E-01</td>
<td>11051</td>
</tr>
<tr>
<td>1.93750</td>
<td>0.24748</td>
<td>0.81939E-02</td>
<td>16917</td>
</tr>
<tr>
<td>2.90625</td>
<td>0.25893</td>
<td>0.57553E-02</td>
<td>10974</td>
</tr>
<tr>
<td>3.87499</td>
<td>0.26262</td>
<td>0.42533E-02</td>
<td>11099</td>
</tr>
<tr>
<td>4.84374</td>
<td>0.27275</td>
<td>0.34226E-02</td>
<td>11244</td>
</tr>
<tr>
<td>5.81249</td>
<td>0.27957</td>
<td>0.26896E-02</td>
<td>11393</td>
</tr>
<tr>
<td>6.78124</td>
<td>0.28574</td>
<td>0.24530E-02</td>
<td>11546</td>
</tr>
<tr>
<td>7.74999</td>
<td>0.29139</td>
<td>0.21550E-02</td>
<td>11686</td>
</tr>
<tr>
<td>8.71874</td>
<td>0.29653</td>
<td>0.19061E-02</td>
<td>11821</td>
</tr>
<tr>
<td>9.68749</td>
<td>0.30116</td>
<td>0.16846E-02</td>
<td>11944</td>
</tr>
<tr>
<td>10.65624</td>
<td>0.30522</td>
<td>0.14920E-02</td>
<td>12056</td>
</tr>
<tr>
<td>11.62499</td>
<td>0.30877</td>
<td>0.12848E-02</td>
<td>12147</td>
</tr>
<tr>
<td>12.55373</td>
<td>0.31151</td>
<td>0.10850E-02</td>
<td>12225</td>
</tr>
<tr>
<td>13.56248</td>
<td>0.31326</td>
<td>0.83827E-03</td>
<td>12276</td>
</tr>
<tr>
<td>14.53123</td>
<td>0.31355</td>
<td>0.58364E-03</td>
<td>12287</td>
</tr>
<tr>
<td>15.49998</td>
<td>0.31168</td>
<td>0.12256E-03</td>
<td>12287</td>
</tr>
<tr>
<td>16.46872</td>
<td>0.30555</td>
<td>0.30467E-03</td>
<td>1235</td>
</tr>
<tr>
<td>17.43747</td>
<td>0.29369</td>
<td>-.11776</td>
<td>-</td>
</tr>
<tr>
<td>18.40622</td>
<td>0.26619</td>
<td>-.49561E-03</td>
<td>-</td>
</tr>
<tr>
<td>19.37497</td>
<td>0.20055</td>
<td>-.68273E-04</td>
<td>-.76850E-01</td>
</tr>
</tbody>
</table>

WING CHARACTERISTICS

<table>
<thead>
<tr>
<th>MACH NO</th>
<th>YAW</th>
<th>ANG OF ATTACK</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.10000</td>
<td>0</td>
<td>2.0000</td>
</tr>
</tbody>
</table>

CL | CD | FCRM | CD FRICTION | CD | L/D FORM | L/D |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.28117</td>
<td>0.38318E-02</td>
<td>0.10030E-01</td>
<td>0.13832E-01</td>
<td>73.379</td>
<td>20.</td>
<td></td>
</tr>
</tbody>
</table>

CH YAW | CH RCLL | CM PITCH | MACH NO | YAW | ANG OF ATTACK |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00116</td>
<td>0.25975</td>
<td>-</td>
<td>0.10000</td>
<td>0</td>
<td>2.0000</td>
</tr>
</tbody>
</table>

A46C MODIFIED FROM FLC27 OF ANTCY JAMESON, CURRANT INSTITUTE.
THREE DIMENSIONAL WING ANALYSIS IN TRANSONIC FLOW USING FINITE VOLUME SCHEME.
BOEING VERSION, PREPARED BY DR. HAI-CHOW CHEN STANDARD BOEING INPUT FORMAT.
<table>
<thead>
<tr>
<th>X/C</th>
<th>1.000000</th>
<th>0.235694</th>
<th>0.560492</th>
<th>0.266130</th>
<th>0.521777</th>
<th>0.159264</th>
<th>0.883857</th>
<th>0.12588</th>
</tr>
</thead>
<tbody>
<tr>
<td>CP</td>
<td>0.846730</td>
<td>0.100136</td>
<td>0.103096</td>
<td>0.077248</td>
<td>0.744854</td>
<td>0.055858</td>
<td>0.740105</td>
<td>0.02678</td>
</tr>
<tr>
<td></td>
<td>0.706152</td>
<td>0.024387</td>
<td>0.725658</td>
<td>0.021524</td>
<td>0.640641</td>
<td>0.023433</td>
<td>0.609077</td>
<td>0.02888</td>
</tr>
<tr>
<td></td>
<td>0.578298</td>
<td>0.013896</td>
<td>0.548030</td>
<td>0.013896</td>
<td>0.519055</td>
<td>0.018980</td>
<td>0.490694</td>
<td>0.00780</td>
</tr>
<tr>
<td></td>
<td>0.463650</td>
<td>0.027180</td>
<td>0.362828</td>
<td>-0.022496</td>
<td>0.410261</td>
<td>-0.021841</td>
<td>0.385015</td>
<td>-0.01202</td>
</tr>
<tr>
<td></td>
<td>0.360542</td>
<td>0.050332</td>
<td>0.336848</td>
<td>-0.074966</td>
<td>0.213400</td>
<td>-0.095895</td>
<td>0.291831</td>
<td>-0.11564</td>
</tr>
<tr>
<td></td>
<td>0.270527</td>
<td>0.123109</td>
<td>-0.210941</td>
<td>-0.123109</td>
<td>0.007504</td>
<td>-0.120546</td>
<td>0.121358</td>
<td>-0.14243</td>
</tr>
<tr>
<td></td>
<td>0.128543</td>
<td>0.192727</td>
<td>0.114441</td>
<td>0.170904</td>
<td>0.107166</td>
<td>0.164057</td>
<td>0.088561</td>
<td>0.11892</td>
</tr>
<tr>
<td></td>
<td>0.076658</td>
<td>0.214849</td>
<td>0.065638</td>
<td>-0.239943</td>
<td>0.055858</td>
<td>-0.231641</td>
<td>0.046334</td>
<td>0.10132</td>
</tr>
<tr>
<td></td>
<td>0.038050</td>
<td>0.105613</td>
<td>0.030625</td>
<td>0.011946</td>
<td>0.021313</td>
<td>0.115395</td>
<td>0.018260</td>
<td>0.26272</td>
</tr>
<tr>
<td></td>
<td>0.013378</td>
<td>0.236647</td>
<td>0.009185</td>
<td>0.297724</td>
<td>0.006189</td>
<td>0.417649</td>
<td>0.003566</td>
<td>0.51652</td>
</tr>
<tr>
<td></td>
<td>0.001527</td>
<td>0.565799</td>
<td>0.003362</td>
<td>0.655856</td>
<td>0.000000</td>
<td>0.680651</td>
<td>0.003063</td>
<td>0.62774</td>
</tr>
<tr>
<td></td>
<td>0.001528</td>
<td>0.565625</td>
<td>0.003356</td>
<td>0.514167</td>
<td>0.006187</td>
<td>0.448722</td>
<td>0.009017</td>
<td>0.34531</td>
</tr>
<tr>
<td></td>
<td>0.012560</td>
<td>0.242914</td>
<td>0.017041</td>
<td>0.142642</td>
<td>0.022359</td>
<td>0.050850</td>
<td>0.028618</td>
<td>0.02137</td>
</tr>
<tr>
<td></td>
<td>0.035734</td>
<td>0.120334</td>
<td>0.034353</td>
<td>0.245997</td>
<td>0.092058</td>
<td>0.409033</td>
<td>0.061694</td>
<td>0.51362</td>
</tr>
<tr>
<td></td>
<td>0.072512</td>
<td>0.523090</td>
<td>0.084319</td>
<td>0.476948</td>
<td>0.096972</td>
<td>0.426525</td>
<td>0.110346</td>
<td>0.46552</td>
</tr>
<tr>
<td></td>
<td>0.124444</td>
<td>0.416798</td>
<td>0.139316</td>
<td>0.442701</td>
<td>0.185014</td>
<td>0.461411</td>
<td>0.171584</td>
<td>0.47243</td>
</tr>
<tr>
<td></td>
<td>0.186546</td>
<td>0.472179</td>
<td>0.207137</td>
<td>0.471177</td>
<td>0.262132</td>
<td>0.474036</td>
<td>0.245939</td>
<td>0.47567</td>
</tr>
<tr>
<td></td>
<td>0.266567</td>
<td>0.472580</td>
<td>0.288015</td>
<td>0.464286</td>
<td>0.310273</td>
<td>0.452544</td>
<td>0.333338</td>
<td>0.43992</td>
</tr>
<tr>
<td></td>
<td>0.357204</td>
<td>0.422491</td>
<td>0.381869</td>
<td>0.405424</td>
<td>0.407326</td>
<td>0.395000</td>
<td>0.435577</td>
<td>0.36128</td>
</tr>
<tr>
<td></td>
<td>0.460617</td>
<td>0.366813</td>
<td>0.488450</td>
<td>0.355193</td>
<td>0.517091</td>
<td>0.341901</td>
<td>0.546485</td>
<td>0.32663</td>
</tr>
<tr>
<td></td>
<td>0.576652</td>
<td>0.306066</td>
<td>0.607654</td>
<td>0.279410</td>
<td>0.639466</td>
<td>0.246108</td>
<td>0.672056</td>
<td>0.21228</td>
</tr>
<tr>
<td></td>
<td>0.705395</td>
<td>0.183974</td>
<td>0.739503</td>
<td>0.161077</td>
<td>0.779388</td>
<td>0.135022</td>
<td>0.810059</td>
<td>0.09599</td>
</tr>
<tr>
<td></td>
<td>0.846511</td>
<td>0.044331</td>
<td>0.883730</td>
<td>0.008174</td>
<td>0.912712</td>
<td>0.006243</td>
<td>0.960471</td>
<td>0.14359</td>
</tr>
</tbody>
</table>

SPAN STATION = 2.90625
NO. CF DATA POINTS = 101

<table>
<thead>
<tr>
<th>X/C</th>
<th>1.000000</th>
<th>0.236239</th>
<th>0.560492</th>
<th>0.207048</th>
<th>0.521777</th>
<th>0.161171</th>
<th>0.808557</th>
<th>0.12633</th>
</tr>
</thead>
<tbody>
<tr>
<td>CP</td>
<td>0.846730</td>
<td>0.102997</td>
<td>0.810356</td>
<td>0.080654</td>
<td>0.774854</td>
<td>0.059673</td>
<td>0.740105</td>
<td>0.04100</td>
</tr>
<tr>
<td></td>
<td>0.706152</td>
<td>0.251555</td>
<td>0.672598</td>
<td>0.025745</td>
<td>0.640641</td>
<td>0.027657</td>
<td>0.609077</td>
<td>0.02724</td>
</tr>
</tbody>
</table>
### SPAN STATION = 7.74959
**NO. OF DATA POINTS = 101**

<table>
<thead>
<tr>
<th>X/C</th>
<th>CP</th>
<th>X/C</th>
<th>CP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000000</td>
<td>0.237601</td>
<td>0.500490</td>
<td>0.254070</td>
</tr>
<tr>
<td>0.846773</td>
<td>0.108310</td>
<td>0.517634</td>
<td>0.672058</td>
</tr>
<tr>
<td>0.706152</td>
<td>0.037378</td>
<td>0.578298</td>
<td>0.632015</td>
</tr>
<tr>
<td>0.578298</td>
<td>0.011112</td>
<td>0.636042</td>
<td>0.500490</td>
</tr>
<tr>
<td>0.463054</td>
<td>0.031119</td>
<td>0.320572</td>
<td>0.265629</td>
</tr>
<tr>
<td>0.270527</td>
<td>0.125679</td>
<td>0.153325</td>
<td>0.178323</td>
</tr>
<tr>
<td>0.153325</td>
<td>0.195973</td>
<td>0.128524</td>
<td>0.180557</td>
</tr>
<tr>
<td>0.076658</td>
<td>0.032246</td>
<td>0.036566</td>
<td>0.027828</td>
</tr>
<tr>
<td>0.011527</td>
<td>0.017012</td>
<td>0.001528</td>
<td>0.017012</td>
</tr>
<tr>
<td>0.017012</td>
<td>0.070056</td>
<td>0.012560</td>
<td>0.125679</td>
</tr>
</tbody>
</table>

### SPAN STATION = 8.71874
**NO. CF DATA FCINTS = 101**

<table>
<thead>
<tr>
<th>X/C</th>
<th>CP</th>
<th>X/C</th>
<th>CP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000000</td>
<td>0.237601</td>
<td>0.500490</td>
<td>0.254070</td>
</tr>
<tr>
<td>0.846773</td>
<td>0.108310</td>
<td>0.517634</td>
<td>0.672058</td>
</tr>
<tr>
<td>0.706152</td>
<td>0.037378</td>
<td>0.578298</td>
<td>0.632015</td>
</tr>
<tr>
<td>0.578298</td>
<td>0.011112</td>
<td>0.636042</td>
<td>0.500490</td>
</tr>
<tr>
<td>0.463054</td>
<td>0.031119</td>
<td>0.320572</td>
<td>0.265629</td>
</tr>
<tr>
<td>0.270527</td>
<td>0.125679</td>
<td>0.153325</td>
<td>0.178323</td>
</tr>
<tr>
<td>0.153325</td>
<td>0.195973</td>
<td>0.128524</td>
<td>0.180557</td>
</tr>
<tr>
<td>0.076658</td>
<td>0.032246</td>
<td>0.036566</td>
<td>0.027828</td>
</tr>
<tr>
<td>0.011527</td>
<td>0.017012</td>
<td>0.001528</td>
<td>0.017012</td>
</tr>
<tr>
<td>0.017012</td>
<td>0.070056</td>
<td>0.012560</td>
<td>0.125679</td>
</tr>
<tr>
<td>X/C</td>
<td>CP</td>
<td>X/C</td>
<td>CP</td>
</tr>
<tr>
<td>-----</td>
<td>--------</td>
<td>-----</td>
<td>--------</td>
</tr>
<tr>
<td>0.000000</td>
<td>0.237601</td>
<td>0.960492</td>
<td>0.2940C</td>
</tr>
<tr>
<td>0.846730</td>
<td>0.106719</td>
<td>0.610366</td>
<td>0.877388</td>
</tr>
<tr>
<td>0.706152</td>
<td>0.351012</td>
<td>0.672558</td>
<td>0.036244</td>
</tr>
<tr>
<td>0.100000</td>
<td>0.219890</td>
<td>0.883750</td>
<td>0.016212</td>
</tr>
</tbody>
</table>

SPAN STATION = 10.65624
NO. OF DATA POINTS = 101

### SPAN STATION = S.68749
NO. OF DATA POINTS = 101

<table>
<thead>
<tr>
<th>X/C</th>
<th>CP</th>
<th>X/C</th>
<th>CP</th>
<th>X/C</th>
<th>CP</th>
<th>X/C</th>
<th>CP</th>
<th>X/C</th>
<th>CP</th>
<th>X/C</th>
<th>CP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000000</td>
<td>0.237601</td>
<td>0.960492</td>
<td>0.2940C</td>
<td>0.521777</td>
<td>0.164577</td>
<td>0.883857</td>
<td>0.154610</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.846730</td>
<td>0.106719</td>
<td>0.610366</td>
<td>0.877388</td>
<td>0.774854</td>
<td>0.067711</td>
<td>0.740105</td>
<td>0.045599</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.706152</td>
<td>0.351012</td>
<td>0.672558</td>
<td>0.036244</td>
<td>0.640642</td>
<td>0.038924</td>
<td>0.690707</td>
<td>0.074164</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000</td>
<td>0.219890</td>
<td>0.883750</td>
<td>0.016212</td>
<td>0.521712</td>
<td>0.065616</td>
<td>0.960470</td>
<td>0.145310</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X/C</td>
<td>C</td>
<td>X/C</td>
<td>C</td>
<td>X/C</td>
<td>C</td>
<td>X/C</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<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>1.000000</td>
<td>0.238146</td>
<td>0.960492</td>
<td>0.209406</td>
<td>0.521777</td>
<td>0.164577</td>
<td>0.863857</td>
<td>0.123103</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.846730</td>
<td>0.108719</td>
<td>0.810376</td>
<td>0.087738</td>
<td>0.774854</td>
<td>0.039101</td>
<td>0.590974</td>
<td>0.035120</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.706152</td>
<td>0.335101</td>
<td>0.672558</td>
<td>0.033675</td>
<td>0.640642</td>
<td>0.000324</td>
<td>0.490528</td>
<td>0.027166</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.578461</td>
<td>0.196212</td>
<td>0.503614</td>
<td>0.032724</td>
<td>0.459059</td>
<td>0.007285</td>
<td>0.385013</td>
<td>0.016713</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.450168</td>
<td>0.073367</td>
<td>0.362828</td>
<td>0.045613</td>
<td>0.382621</td>
<td>0.000135</td>
<td>0.336265</td>
<td>0.002233</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.360543</td>
<td>0.028517</td>
<td>0.336848</td>
<td>0.051941</td>
<td>0.323940</td>
<td>0.007434</td>
<td>0.291831</td>
<td>0.005601</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.270527</td>
<td>0.069512</td>
<td>0.250026</td>
<td>0.056296</td>
<td>0.230321</td>
<td>0.009307</td>
<td>0.211358</td>
<td>0.005669</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.193235</td>
<td>0.112116</td>
<td>0.175822</td>
<td>0.144715</td>
<td>0.159225</td>
<td>0.165479</td>
<td>0.143458</td>
<td>0.165832</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.128543</td>
<td>0.152026</td>
<td>0.114441</td>
<td>0.128261</td>
<td>0.101166</td>
<td>0.118009</td>
<td>0.088552</td>
<td>0.193191</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.076658</td>
<td>0.155210</td>
<td>0.065398</td>
<td>0.171380</td>
<td>0.055586</td>
<td>0.155210</td>
<td>0.046334</td>
<td>0.055472</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.038050</td>
<td>0.023425</td>
<td>0.036335</td>
<td>0.074932</td>
<td>0.024313</td>
<td>0.158909</td>
<td>0.018626</td>
<td>0.284669</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.013578</td>
<td>0.335012</td>
<td>0.051948</td>
<td>0.385012</td>
<td>0.006188</td>
<td>0.360369</td>
<td>0.003586</td>
<td>0.553531</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.001527</td>
<td>0.652445</td>
<td>0.000362</td>
<td>0.666626</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.001502</td>
<td>0.959640</td>
<td>0.003625</td>
<td>0.860367</td>
<td>0.001017</td>
<td>0.309260</td>
<td>0.004000</td>
<td>0.712000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.012592</td>
<td>0.281619</td>
<td>0.728539</td>
<td>0.152490</td>
<td>0.022358</td>
<td>0.623121</td>
<td>0.062818</td>
<td>0.212133</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.072512</td>
<td>0.707771</td>
<td>0.084319</td>
<td>0.642368</td>
<td>0.052008</td>
<td>0.062491</td>
<td>0.061964</td>
<td>0.710444</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.124444</td>
<td>0.542824</td>
<td>0.139316</td>
<td>0.565205</td>
<td>0.155015</td>
<td>0.581367</td>
<td>0.110348</td>
<td>0.543060</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.188946</td>
<td>0.575313</td>
<td>0.207137</td>
<td>0.567615</td>
<td>0.226132</td>
<td>0.563094</td>
<td>0.245939</td>
<td>0.557986</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.266567</td>
<td>0.547937</td>
<td>0.288015</td>
<td>0.532397</td>
<td>0.310274</td>
<td>0.513341</td>
<td>0.333333</td>
<td>0.452500</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.357205</td>
<td>0.470191</td>
<td>0.381869</td>
<td>0.449334</td>
<td>0.407327</td>
<td>0.428549</td>
<td>0.433577</td>
<td>0.448321</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.460617</td>
<td>0.390227</td>
<td>0.488450</td>
<td>0.371822</td>
<td>0.517017</td>
<td>0.353651</td>
<td>0.546485</td>
<td>0.333333</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.576693</td>
<td>0.305046</td>
<td>0.607694</td>
<td>0.278575</td>
<td>0.639646</td>
<td>0.241050</td>
<td>0.672086</td>
<td>0.245250</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.705355</td>
<td>0.174957</td>
<td>0.739503</td>
<td>0.150774</td>
<td>0.774368</td>
<td>0.123995</td>
<td>0.810597</td>
<td>0.184763</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.846911</td>
<td>0.034338</td>
<td>0.883730</td>
<td>0.011686</td>
<td>0.921712</td>
<td>0.070572</td>
<td>0.960411</td>
<td>0.151630</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**SPAN STATION** = 14.53123

**NO. CF DATA POINTS** = 101
APPENDIX E

THIS APPENDIX PRESENTS THE SOURCE CODE FOR THE POTENTIAL FLOW PROGRAM FLO27.

*........................FLG27........................9/12/83..............................**
C**
C** THREE DIMENSIONAL WING ANALYSIS IN TRANSONIC FLOW USING
C** FINITE VOLUME SCHEME WITH SHEARED PARABOLIC COORDINATES
C** PROGRAMMED BY ANTONY JAMESON, JANUARY-APRIL 1977
C**
C**________________________________________________________________________**
C** THE BOEING VERSION OF FLO27 WAS PREPARED BY CR. HAI-CHOW
C** CHEN WITH THE FOLLOWING MODIFICATIONS
C** 1) TEMPORARY STORAGE OF THE LARGE CORE MEMORY REQUIREMENTS
C** HAS BEEN IMPLEMENTED TO REDUCE THE COMPUTING COSTS
C** BY BUFFERING DATA IN AND OUT OF CORE.
C**
C** 2) STANDARD BOEING INPUT FORMAT FOR THE WING SECTION
C** HAS BEEN USED.
C**
C** 3) SUBPROGRAM BLIN HAS BEEN IMPLEMENTED TO ADD THE
C** DISPLACEMENT THICKNESS TO THE ORIGINAL WING SECTIONS
C**
C** 4) WING SECTION LEADING EDGE SINGULAR POINT IS FOUND BY
C** COMPUTING THE FOCUS OF A PARABOLA BY 2N+1 POINTS
C** LEAST-SQUARE FIT CENTERED AT THE LEADING EDGE POINT.
C** N IS SUPPLIED BY THE USER THROUGH INPUT CARD.
C**
C** 5) TRAILING EDGE CLOSURE ANGLE AND BISECTOR SLOPE ARE
C** COMPUTED BASED ON BACKWARD DIFFERENCE.
C**
C** 6) SUBRoutines FOR PRINTER-Plotting OF THE UNWRAPPED
C** WING SECTIONS IS AVAILABLE
C**
C**________________________________________________________________________**
C
THE FOLLOWING FILES ARE USED TO EXECUTE FLO27. SOME OF THESE
FILES ARE USED SUBSEQUENTLY IN OTHER MODULES OF THE VISCOSI
INVISCO INTERACTIVE WING SYSTEM

FILE1 IS USED TO BUFFER DATA IN AND OUT OF CORE
FILE2 IS USED TO BUFFER DATA IN AND OUT OF CORE
FILE3 IS USED TO BUFFER DATA IN AND OUT OF CORE
FILE4 IS USED TO READ IN THE VELOCITY POTENTIAL GENERATED PREVIOUSLY.

FILE8 IS WRITTEN FOR DATA TRANSFER TO BEING TURBULENT BOUNDARY LAYER PROGRAM A411.

FILES IS USED TO SAVE SECTION SURFACE PRESSURE AGAINST X/C.

FILE10 IS USED TO SAVE THE SECTION X, Y, Z LOCATION FOR A411IN WHICH CALCULATES THE CORRESPONDING DISPLACEMENT THICKNESS WHERE X, Y, Z SHOULD BE THE WING SURFACE LOCATION FOR THE CURRENT RUN.

FILE11 IS USED TO READ IN THE DISPLACEMENT THICKNESS FROM A411IN.

FILE12 IS USED TO REPLACE PART OF THE INPUT CARDS BY CARD IMAGES

TAPE13 IS USED TO REPLACE PART OF THE CUTPLT SKIPPED FROM THE LINE PRINTER

TAPE14 IS USED TO SAVE THE VELOCITY POTENTIAL FOR FUTURE USE.

******************************************************************************

COMMON G (161, 18, 3), SO (161, 35), VCRT (115), ZV (115),
1 IV (161, 35), ITE (35), ITEL (35),
2 AO (161), BO (18), XO (35), YC (35), ZO (35), SCAL (35),
3 NX, NY, N2, KTE1, KTE2, SYM, KS, SYM, FUS,
4 YAW, CYAW, SYAW, ALPHA, CA, SA, FMACH, N1, N2, N3, IO.

COMMON /CPF/ NL.
COMMON /PCKR/ PTK.
COMMON /FLO/ P1, P2, P3, FRES, JRES, KRES, ARES, CG, LG, JG, LG, AG, NSUP.
COMMON /PRT3/ XI (161), YI (161), ZI (161),
1 UT3 (161), VT3 (161), WT3 (161), AO1, Z.

DIMENSION XS (161, 11), YS (161, 11), ZS (11), XE (11), YE (11),
1 SLOF (11), TRLI (11), NP (11), EL (11), E2 (11), E3 (11),
2 DI (161), D2 (161), D3 (161), SN (161),
3 SV (161), S2 (161), CP (161), XP (161), YP (161),
4 YP (35), ZP (35), XMAX (35), YMIN (35), YMAX (35), YM (35),
5 CHO (35), SCL (35), SOC (35), SCC (35), TITLE (20),
1 RES (201), CCUN (201), PIMA (3),
1
2 DUMX(161), DUMY(161), DELR(161)
C******************************************************************************
C** INITIALIZE INPUT PARAMETERS WHICH HAVE RECOMMENDED PROGRAM VALUES **
C******************************************************************************
C**
C**
C****************************************************************************

XSCAL = 0.0
PSCL = 0.0
FSCNT = 0.0
BLCP = 0.0
WEIG = 1.0
PTCK = 0.0

ND = 161
NE = 161
KPLCT = 0
ILCPT = 1
ISTCP = 2
N1 = 1
N2 = 2
N3 = 3
REWIND 1
REWIND 2
REWIND 4
REWIND 10
REWIND 13
REWIND 14
JO = 0
RAD = 57.29579513082

1 WRITE (6,600)
WRITE (6,612)
2 FORMAT ('CA469 MODIFIED FROM FLC27 CF ANTONY JAMESON, ',
'COURANT INSTITUTE /
'FOR THREE DIMENSIONAL WING ANALYSIS IN TRANSONIC FLOW,
'USING FINITE VOLUME SCHEME /
'OBEOING VERSION, PREPARED BY DR. HAI-CHOW CHEN',
'STANDARD BOEING INPUT FORMAT FOR WING SECTION DATA IS USED')
READ (5,530) TITLE
WRITE (6,630) TITLE
READ (5,500)
READ (5,510) FNX, FNY, FNZ, FMESH, FPLCT
NX = FNX
NY = FNY
NZ = FNZ
MMESH = FMESH
IF (NX.LT.1) GO TO 301
KPLCT = ABS(FPLCT)
READ (5,500)
DO 12 NP=1, NMESH
C**************************************************************************
C** INITIALIZE INPUT PARAMETERS WHICH HAVE RECOMMENDED PROGRAM VALUES **
C**************************************************************************
P2G(NM) = 0.7
P3G(NM) = 1.0
FSMG0(NM) = 0.0
PTMAP(NM) = 0.0
C**************************************************************************
12 READ (5,510) FIT(NM), CCVO(NM), P10(NM)
READ (5,500)
READ (5,510) FMACH, YA, AL, CDO
CALL GECH (ND, NS, NP, XS, YS, ZS, XLE, YLE, SLOPT, TRAIL, XP, YP,
1 FUS, XTEG, CHORDO, ZTIP, SWEEP, CHED,
2 FIX, PX, PZ, ISYM0, KSYM)
ALPHA = AL/RAD
IF (BLCF .LE. 0.) GO TO 44
IF (PTCK .GE. 1.) WRITE (6,600)
READ (11) (TITLE(I), I=1,8), FMACH, ALPHA, NS
DO 40 K = 1, NS
READ (11) NFCK
NP(K) = NFCK
NPCK1 = NFCK + 1
READ (11) (DUMX(NFCK1-I), DUMZ, DUMY(NFCK1-I), CELR(NFCK1-I),
1 I=1, NFCK)
IF (PTCK .LE. 0.) GC TC 30
WRITE (6,52) K, NPCK, NPCK1
WRITE (6,54) (DUMX(NFCK1-I), DUMZ, DUMY(NFCK1-I),
1 I=1, NFCK)
WRITE (6,54) (XS(I,K), YS(I,K), I=1, NFCK)
30 CONTINUE
CALL BLIN (XS(1,K), YS(1,K), DELR, WEIG, NFCK, NL)
40 CONTINUE
CALL BLIN 
44 CONTINUE
IF (PTCK .LT. 0.) GO TO 56
IF (PTCK .GE. 1.) WRITE (6,600)
WRITE (10) (TITLE(I), I=1,8), FMACH, ALPHA, NS
DO 50 K = 1, NS
NFCK = NP(K)
NPCK1 = NFCK + 1
DO 48 I = 1, NPCK
DUMX(NFCK1-I) = XS(I,K) + XLE(K)
DUMY(NFCK1-I) = YS(I,K) + YLE(K)
48 CONTINUE
WRITE (10) NPCK
WRITE (10) (DUMX(I), ZS(K), DUMY(I), I=1, NPCK)
IF (PTCK .LE. 0.) GC TO 49
WRITE (6,52) K, NPCs, ZS(K)
WRITE (6,54) (DUMX(I), ZS(K), DUMY(I), I = 1, NPCs)
CONTINUE
ENDFILE 10
FORMAT (1HO, 5X, 31S, F11.4)
FORMAT (12F11.4)
CONTINUE
IF (KSMP NE. 0) YA = 0.
ISYM = ISYM
IF (ALNE.0) ISYM = 0
YA = YA/4AD
CYAh = COS(YA)
SYAh = SIN(YA)
CA = CYAh*COS(ALPHA)
SA = CYAh*SIN(ALPHA)
IF (FCCTLT. 1.) GC TC 91
REAL 4
NX, NY, NZ, NM, K1, K2, NIT
MX = NX + 1
MY = NY + 2
MZ = NZ + 3
DO 62 K = 1, MZ
READ 4 ((G(I, J, I), I = 1, MX), J = 1, MY)
WRITE (N3) ((G(I, J, I), I = 1, MX), J = 1, MY)
IF (UNIT(N3), GT. 0.) GO TO 1
C BUFFER CLT(N3, 1) (G(I, I, I), G(MX, MY, 1))
WRITE (N3) ((G(I, J, I), I = 1, MX), J = 1, MY)
C IF (UNIT(N1), GT. 0.) GO TO 1
C CONTINUE
READ 4 (VORT(K), K = K1, K2)
REWIND N3
REWIND N1
REWIND 4
91 CALL CCROD (NX, NY, NZ, KSYM, ZTIP, XLIM, ZLIM, 1 SY, AX, AY, AZ, PX, PZ, AC, BC, ZO)
CALL SINGLE (NS, NZ, KSYM, KTE1, KTE2, FUS, CCR1, D, ZS, XLE, YLE, 1 SWEET, CIHED, X0, Y0, Z0, ZP, Z1, Z2, EL, E2, E3, IND)
CALL SUFF (ND, NE, NS, NX, NZ, ISYM, KSYM, KTE1, KTE2, 1 YW, XE, XLIM, FIX, XP, XS, YS, ZS, SLOP1, TRAIL, 2 AO, AX, ZC, SO, SCAL, ZV, IV, ITE1, ITE2, 3 XP, YP, SN, D1, D2, D3, IND)
IF (INC. GE. 0) GC TC 291
IF (FCCAT. GE. 1.) GO TO 101
N = 1
NIT = 0
CALL ESTIM
IF (IGNE. 0) GO TO 1
REWIND N1
REWIND N2

101 IF (PTCK .GE. 1.) WRITE (6,600)
   FCNT = 0,
   COV = COVG(NM)
   P1 = P10(NM)
   P2 = P20(NM)
   P3 = P30(NM)
   MIT = FIT(NM) + NIT
   KIT = MIT
   IF (KIT.LT.2) KIT=2
   JIT = NIT
   LRES = 0
   MRRES = (MIT - NIT - 2)/200 + 2
   NRES = 0
   MX = NX + 1
   MY = NY + 2
   MZ = NZ + 3
   KY = NY + 1
   K1 = 1
   K2 = NZ + 1
   IF (KSYM.EQ.0) GO TO 103
   K1 = 3
   K2 = NZ + 3
   L2 = NZ/2 + 1
103 IF (KSYM.NE.0) L2 = 3
104 FORMAT(48H0INDICATION OF LOCATION OF WING AND VORTEX SHEET,
1 27H IN COORDINATE PLANE Y = 0./
2 27H0((1V(I,K),K=K1,K2),I=1,MX))
   DO 106 I=1,MX
106 WRITE (6,650) (1V(I,K),K=K1,K2)
   CONTINUE
108 IMAP = FTMAP(NM)
   IF (IMAP .EQ. 0) GO TO 830
   WRITE (6,600)
   WRITE (6,112)
112 FORMAT(4H0CHORDWISE CELL DISTRIBUTION IN SQUARE ROOT PLANE,
1 54H AND MAPPED SURFACE COORDINATES AT CENTER LINE AND TIP)
   DO 820 ISEC = L2,KTEZ,IMAP
5H0SEC, 2PO(ISEC)
   WRITE (6,812) ISEC,2PO(ISEC)
812 FORMAT (15H0 1 X ,20H SECTION PROFILE NO.,
1 12,5X,=I,'G13.4) 820 CALL PFPY (2,NX,AO,SO(I,1,ISEC))
   CONTINUE
   IF (PTCK .LE. 0.) GO TO 130
   WRITE (6,116)
116 FORMAT(15H0 TE LOCATION ,15H POWER LAW )
WRITE (6,610) XLM,A X
WRITE (6,600)
WRITE (6,118)
116 FORMAT(46H0 NORMAL CELL DISTRIBUTION IN SQUARE ROOT PLANE/ 1 15H0 Y )
DO 120 K=1,K
120 WRITE (6,610) BC(J)
WRITE (6,122)
122 FORMAT(15H0 SCALE FACTOR,15H POWER LAW )
WRITE (6,610) SY,A Y
WRITE (6,600)
WRITE (6,124)
124 FORMAT(45H0 SPANWISE CELL DISTRIBUTION AND SINGULAR LINE/ 1 15H0 Z ,15H X SING ,15H Y SING )
DO 126 K=K1,K2
126 WRITE (6,610) ZO(K),X0(K),YO(K)
WRITE (6,128)
128 FORMAT(15H0 TIP LOCATION,15H POWER LAW )
WRITE (6,610) ZLIM,A Z
130 CONTINUE
WRITE (6,600)
WRITE (6,132)
132 FORMAT(15H0 ITERATIVE SOLUTION)
WRITE (6,134)
134 FORMAT(15H0 MACH NO ,15H YAW ,15H ANG OF ATTACK )
WRITE (6,610) FMACH,YA,AL
WRITE (6,136)
136 FORMAT(15H0 NX ,15H NY ,15H NZ )
WRITE (6,640) NX,NY,NZ
WRITE (6,138)
138 FORMAT(15H0 RELAX FCT 1 ,15H RELAX FCT 2 ,15H RELAX FCT 3 )
WRITE (6,610) P10(NM),P20(NM),P30(NM)
WRITE (6,140)
140 FORMAT(15H0 ITERATION,
1 15H MAX CORRECTN ,4H I ,4F J ,4H K ,15H AVG CORRECTN ;
2 15H MAX RESIDAL ,4H I ,4F J ,4H K ,15H AVG RESIDAL ;
3 15H CIRCULATION,15H SONIC PSI)
141 NIT = NIT +1
JIT = JIT +1
CALL MIXFLG
IF (IO,EC.0) GO TO 151
151 JC = 0
REIND A1
REIND A2
N = N1
N1 = N2
N2 = N3
N3 = N
WRITE (6,660) NIT, DG, IG, JG, KG, AG, FRES, IFES, JRES, KRES, ARES,
1, VCR(1LZ), SUP
LRES = LRES + 1
IF (LRES.EQ.MRES) LRES = 1
IF (LRES.NE.1) GO TO 143
NRES = NRES + 1
COUNT(NRES) = NIT - 1
RES(NRES) = FRES
143 IF (JIT.GE.KIT) GO TO 251
IF (NM.LE.1 .OR. NM.LT.MMESH) GO TO 148
IF (ABS(DG).LE. COV) GO TO 251
148 CONTINUE
IF (NIT.LT.MIT.AND.ABS(DG).GT.COY.AND.ABS(DG).LT.10.) GO TO 141
GO TO 161
151 IF (JO.EQ.1) GO TO 1
REIND N1
REIND N2
K0 = 1
N0 = N0
N1 = N1
N2 = N2
N3 = N3
GO TO 141
161 RATE = 0.
IF (NRES.GT.1) RATE = (ABS(RES(NRES)/RES(1)))
***(1./(COUNT(NRES) - COUNT(1)))
WRITE (6,162)
162 FORMAT(15HO MAX RESIDAL 1,15H MAX RESIDAL 2,15H WORK ,
1 15H REDUCTN/CYCLE, 15H CONV TOLERANCE)
WRITE (6,670) RES(1), RES(NRES), COUNT(NRES), RATE
1, CV
WRITE (6,600)
DO 164 K = 1,3
C BUFFER IN (N1,1) (G(1,1,M),G(MX,MY,M))
REAL (N1,ERR=151) (G(I,J,M),I=1,MX),J=1,MY)
C IF (UNIT(N1).GT.0.) GO TO 151
164 CONTINUE
LX = NX/2 + 1
K = 2
KKK = 0
C WRITE HEADER ON TAPE 8
IF (NM.LT.MMESH) GO TO 17C
REIND B
REIND C
REIND E
NRC = KIE2 - KTE1 + 1
WRITE (E) (TITLE(I),I=1,8), FMACH, ALPHA, NRC
17C CONTINUE
171 K = K + 1
    IF (K.EQ.MZ) GO TO 191
  DO 172 J = 1, MY
  DO 172 I = 1, MX
     G(I,J,1) = G(I,J,2)
  172 G(I,J,2) = G(I,J,3)
C BUFFER IN (NI, NJ (G(I,1,3), G(MX,MY,3))
C READ (NI, ERR=151) ((G(I,J,3), I=1,MX), J=1,MY)
C IF (UNIT(NI).GT.0) GO TO 151
C IF (K.LT.KTE1 OR K.GT.KTE2) GO TO 171
C CALL VELG (K, SV, SM, CP, XP, YP, XMAX(K), XMIN(K), YMAX(K), YMIN(K))
11   = ITEL(K)
I2   = ITE2(K)
CHCRD(K) = XP(I1) - XP(LX)
C ALL FCRCF (I1, I2, XP, YP, CP, AL, CHCRC(K), X0(K), YPO(K),
  1 SCL(K), SCP(K), SCM(K))
KKK = KKK + 1
IF (KPLCT.GT.1 AND K.GT.KTE1) GC TC 185
IF (KPLCT.EQ.0 AND KKK.GT.1) GO TO 185
WRITE (6,600)
WRITE (6,182)
C 182 FORMAT (24HOSECTION CHARACTERISTICS/)
1     15HO MACH NO , 15H YAW , 15H ANG OF ATTACK)
1     WRITE (6,610) FMACH,YA,AL
1     WRITE (6,184)
1 184 FORMAT (15HOSPAN STATIC , 15H X , 15H CD ,
1     15H CM )
1     WRITE (6,161) ZPO(K),SCL(K),SCC(K),SCM(K)
1     Z = ZPC(K)
1     IF (MAT.LE.0) GC TC 850
1     IF (KPLCT.LE.2) CALL CPHLET (2, NX, FMACH, XP, YP, CP, SM, I1, I2, KPLCT)
1     CONTINUE
C WRITE CNE FILE ON TAPE 8
1     IF (NM.LT.MMES) GO TO 186
1     WRITE (E) NC
1     WRITE (E) (XT3(I), ZT3(I), YT3(I), UT3(I), HT3(I), VT3(I), I=1,NGI)
1     NR = I2 - I1 + 1
C WRITE CP VS X/C SECTION DATA FOR FINAL MESH ON TAPE 9
1     WRITE (9,900) ZPO(K)
1     WRITE (9,910) NRD
1     WRITE (9,920)
1     WRITE (9,950) (XOCD(J),CP(J), J=I1, I2)
900 FORMAT (1,HOSPAN STATION =,F12.5)
910 FORMAT (1X,20HNC, CF DATA POINTS =,I5)
920 FORMAT (1X,6H X/C,8X,2HCP,7X,3HX/C,8X,2HCP,
1    17X,3HX/C,8X,2HCP,3X)
950 FORMAT (8F10.6)
186 CONTINUE
C WHEN KPLOT = 2 CALL SUBROUTINE VERTEC WHICH PLOTS CP VS X/C
C FOR EACH SECTION OF THE FINAL MESH
C
IF (KPLCT.EQ.2.AND.NM.EQ.MMESH) CALL VERTEC(11,12,XOCO,CP,ARD,
ZPO,FMACH,YA,AL,SCL,SCD,SCM,K)
GO TO 171
191 CONTINUE
IF(NM.LT.MMESH) GO TO 200
ENCFILE 8
REXD 8
ENCFILE 8
REXD 8
20C CONTINUE
CALL TCFOR (KTE1,KTE2,CHORD,SCL,SCD,SCM,XC,YPO,ZPO,
CL,CD1,CD0,CMR,CNY)
CD1 = CAYCD1
CD = CD0 + CD1
VLC) = 0
IF (ABS(CD1).GT.1.E-6) VLD1 = CL/CD1
VLC = 0
IF (ABS(CD).GT.1.E-6) VLD = CL/CD
WRITE (6,600)
WRITE (6,192)
192 FORMAT(21HOWING CHARACTERISTICS/)
1 15H MACH NO.,15H YAW ,15H ANG OF ATTACK)
15H FMACH,YA,AL
WRITE (6,610)
WRITE (6,154)
194 FORMAT(15H CL ,15H CD FORM ,15H CD FRICTION )
15H L/C FORM ,15H L/D ,15H CD FORM ,15H CM PITCH ,)
15H CM YAW ,15H L/C FORM ,15H CM PITCH ,)
WRITE (6,610)
WRITE (6,196)
196 FORMAT(15H CM YAW ,15H CM PITCH ,)
WRITE (6,610)
REXD 81
IF (KPLCT.LT.1) GO TO 201
C CALL RPLT (IPLCT,NRES,RES,COUNT,TITLE,FMACH,YA,AL,NX,NY,NZ)
C CALL DRAW (IPLCT,XMAX,XMIN,YMAX,YMIN,ZPO,FUS,TITLE,NZ,KTE1,KTE2)
C CALL TFEED (IPLCT,SF,SF,CP,XP,YP,ZPO,TITLE,YA,AL,
VLC,CL,CD,CHORD0,XSCAL,PSCAL)
GO TO 151
201 IF (1STCP.EQ.1) GO TO 151
IF (NM.LT.MMESH) GC TC 203
GO TO 1
203 CONTINUE
NX = NX +NX
NY = NY +NY
NZ = NZ +NZ
CALL CCRDC (NX, NY, NZ, KSYM, ZTIP, XLIM, ZLIM,
1 SY, AX, AY, AZ, PX, PY, PZ, AO, BC, ZO)
CALL SINGL (NS, NZ, KSYM, KTE1, KTE2, FUS, CHORDO, ZS, XLE, YLE,
1 SWEEP, DIHED, XO, YO, Z0, YFC, ZPC, E1, E2, E3, IND)
CALL SUFF (NO, NE, NS, NX, NZ, ISYM, KSYM, KTE1, KTE2,
1 YAW, XTE0, XLIM, FIX, NP, XS, YS, ZS, SLOP, TRAIL,
2 AO, XO, Z0, SO, SCAL, ZV, IV, ITE1, ITE2,
3 XP, YP, SN, D1, D2, D3, IND)
IF (INC.EQ.0) GC TO 291
CALL REFIN
IF (IO.EQ.0) GO TO 221
REWIND N1
REWIND N2
NSMCC = FSMCC(NM)
IF (NSMCC.LT.1) GO TO 211
DO 202 K=1,NSMCC
CALL SMCC
IF (IO.EQ.0) GO TO 221
REWIND N1
REWIND N2
211 N = N1
N1 = N2
N2 = N3
N3 = N
NM = NM + 1
NIT = 0
GO TO 101
221 NX = NX/2
NY = NY/2
NZ = NZ/2
CALL CCRDC (NX, NY, NZ, KSYM, ZTIP, XLIM, ZLIM,
1 SY, AX, AY, AX, PX, PY, PZ, AO, BC, ZO)
CALL SINGL (NS, NZ, KSYM, KTE1, KTE2, FUS, CHORDO, ZS, XLE, YLE,
1 SWEEP, DIHED, XO, YO, ZO, YFC, ZPC, E1, E2, E3, IND)
CALL SUFF (NO, NE, NS, NX, NZ, ISYM, KSYM, KTE1, KTE2,
1 YAW, XTE0, XLIM, FIX, NP, XS, YS, ZS, SLOP, TRAIL,
2 AO, XO, ZO, SO, SCAL, ZV, IV, ITE1, ITE2,
3 XP, YP, SN, D1, D2, D3, IND)
IF (INC.EQ.0) GC TO 291
GO TO 151
251 K1 = KTE1 - 1
K2 = KTE2 + ITE2(KTE2) - NX/2
WRITE (14) NX, NY, NZ, NM, K1, K2, NIT
DO 262 K=1,NZ
C BUFFER IA (N,1) (G(I,1,J),GIMX,MY,1))
READ (N) IE=281 ((G(I,J,1), I=1,MX), J=1,MY)
C IF (UNIT(N).GT.0.) GO TO 281
262 WRITE (14) ((G(I,J,1), I=1,MX), J=1,MY)
REMIND  1
  WRITE (14) (VORT(1),K=K1,K2)
ENDFILE  14
REMIND  14
CALL  SETCH(K,ISTOP)
CALL  SLIET(K,ISTOP)
IF  (ISTCP.EQ.1)  GO  TO  161
  IF  (ITP.LT.MIT.AND.ABS(DG).GT.COVA.AND.AES(DG).LT.10.)  GO  TO  141
  GO  TO  161
281  REMIND  4
  GO  TO  151
291  WRITE  (6,600)
  WRITE  (6,292)
292  FORMAT(4H:OBA: DATA, SPLINE FAILURE)
  GO  TO  1
301  CONTINUE
  REMIND  10
  REMIND  11
  IF  (KPLCT.GT.0)  CALL  PLOT(0.,0.,955)
STCP
50C   FORMAT(1X)
51C   FORMAT(8F10.6)
53C   FORMAT(2CA4)
60C   FORMAT(1H1)
61C   FORMAT(12E5.5,7G15.5)
63C   FORMAT(1H0,20A4)
64C   FORMAT(18,7I15)
65C   FORMAT(1X,33I3)
66C   FORMAT(110,E15.5,3I4,2E15.5,3I4,E15.5,F10.5,I10,F10.3)
67C   FORMAT (2E15.4,2F15.4,E15.4)
END
**SUBROUTINE BLIN**

SUBROUTINE BLIN (XT, YT, DELR, WEIG, NL)

SUBPROGRAM FOR NORMALLY ADDING THE DISPLACEMENT THICKNESS TO THE ORIGINAL WING SECTIONS

XT: CONTAIN THE X COORDINATES OF THE ORIGINAL WING SECTION WHEN CALLED

YT: CONTAIN THE Y COORDINATES OF THE DISPLACED WING SECTION WHEN CALLED

DEL: THE DISPLACEMENT THICKNESS

COMMON /FKK/ PTCK
DIMENSION XT(1), YT(1), DELR(1)
WRITE (6,1000)
I = 1
X2 = XT(I)
Y2 = YT(I)
20C IF (I.EQ.N) GO TO 300
X3 = XT(I+1)
Y3 = YT(I+1)
30C IF (I.EQ.1) GO TO 400
X1 = 2*X2 - X3
Y1 = 2*Y2 - Y3
40C IF (I.EQ.N) GO TO 500
X3 = 2*X2 - X1
Y3 = 2*Y2 - Y1
50C CONTINUE
X12 = X1 - X2
X23 = X2 - X3
X31 = X3 - X1
IF (ABS(X31).LE.1.E-6) GO TO 600
DYX = -X23/(X12*X31)*Y1
DYX = DLYX + (X12-X23)/(X12*X23)*Y2
DYX = DLYX + X12/(X31*X23)*Y3
IF (ABS(DYX).LE.1.E-6) GO TO 820
DYXX = -1./DYX
GO TO 700
60C DLYX = C
70C CONTINUE
IF (I.EQ.NL) GO TO 800
800 CONTINUE
SI = 1.
IF (I.I.L.NL) S1 = -1.
GO TO 850
820 CONTINUE
S = 0.
DL = DELR(I)
DX = 0.
DY = DL * S1
GO TO 850
850 CONTINUE
S = 1. + DYN * 2
S1 = SQR(S)
F = 1.
IF (DYN * S1.LT.0.) F = -1.
DL = DELR(I)
DX = S * F
DY = ABS(DYN) * S * S1
DX = DX * DL
DY = DY * DL
XT(I) = X2 - DX * WEIG
YT(I) = Y2 + DY * WEIG
880 CONTINUE
IF (PTCK .LE. 1.) GO TO 890
WRITE (6, 1000) I, F, S, DL, DX, DY, CYXN, XT(I), YT(I), CYX, WEIG
890 CONTINUE
IF (I .EQ. N) GO TO 900
X1 = X2
Y1 = Y2
X2 = X3
Y2 = Y3
I = I + 1
GO TO 2CO
900 CONTINUE
RETURN
1000 FORMAT (1H, I5, F7.2, 9G13.5)
END
**SUBROUTINE PPXY**

SUBROUTINE PPXY (I1, I2, X, Y)

COMMON /PCKR/ PCK
COMMON /SFARE/ LINE(1GC)
DIMENSION X(I1), Y(I1)
DATA I8 /1H /, IP /1H+/ , KMAX /100/ , ACC /1.5/ ,
1 IZ /1F* , ICONST /0/
DO 10 I =1,100
LINE(I) = I8
1C CONTINUE
YMAX = -1.0E35
YMIN = -YMAX
WICH = KMAX - 5
DO 20 I = 11, I2
YMAX = AMAX1(YMAX, Y(I))
YMIN = AMIN1(YMIN, Y(I))
2C CONTINUE
VAL = ABS(YMAX) + ABS(YMIN)
S = WICH/VAL
KK = 0
IF (ICCAST LE YMAX .AND. ICONST GE YMIN) KK = S*(YMAX-ICONT)+ACC
IF (KK .NE. 0) LINE(KK) = IZ
DO 30 I = 11, I2
K = S*(YMAX-Y(I)) + ACC
IF (K LT 1) K = 1
IF (K GE KMAX) K = KMAX
LINE(K) = IP
WRITE (*,10C) I, X(I), Y(I), LINE
LINE(K) = IZ
30 CONTINUE
RETURN
10C FORMAT (1X,13,12F10.4,4X,100AL)
END
C**SLBROUTINE LSQR********************************************************************
SUBROUTINE LSQR (NL, NB, XP, YP, XSING, YSING)
C
SUBPROGRAM FOR WING SECTION LEADING EDGE SINGULAR POINT
CALCULATION BY MEANS OF COMPUTING THE FOCUS OF A
PARABOLA BY NB*2+1 POINTS LEAST-SQUARE FIT CENTERED AT
THE LEADING EDGE
C
NB: SUPPLY BY THE CALLING PROGRAM GEOM
C
COMMON /FCKR/ FTCK
DIMENSION XP(1), YP(1)
NL = NL - NB
N2 = NL + NB
N = N2 - N1 + 1
A1 = 1
B1 = 0.
C1 = 0.
A2 = 0.
B2 = 0.
C2 = 0.
A3 = 0.
B3 = 0.
C3 = 0.
D1 = 0.
D2 = 0.
D3 = 0.
SCALE = 100.
SCALE2 = 500.
DO 300 I = NL, N2
YY = (YP(I) - YP(NL))*SCALE
B1 = B1 + YY
YP2 = YY + YY
C1 = C1 + YP2
YP3 = YP2*YY
C2 = C2 + YP3
YP4 = YP3*YY
C3 = C3 + YP4
XX = XP(I)*SCALE2
D1 = D1 + XX
YX = YX + XX
D2 = D2 + YX
Y2X = YP2*XX
300
D3 = D2 + Y2X
A2 = B1
B2 = C1
A3 = C2
B3 = C2
FA1 = B2*C3 - B3*C2
FA2 = A3*C2 - A2*C3
FA3 = A2*B3 - A3*B2
DET = A1*FA1 + E1*FA2 + C1*FA3
DI = 1./DET
FA1 = FA1*DI
FA2 = FA2*DI
FA3 = FA3*DI
FB1 = (E3*C1 - E1*C3)*DI
FB2 = (E1*C3 - A3*C1)*DI
FB3 = (A3*B1 - A1*B3)*DI
FC1 = (E1*C2 - E2*C1)*DI
FC2 = (A2*C1 - A1*C2)*DI
FC3 = (A1*B2 - A2*B1)*DI
A = FA1*DI + FB1*DI + FC1*DI
B = FA2*DI + FB2*DI + FC2*DI
C = FA3*DI + FB3*DI + FC3*DI
X = A*(1.* - E**2)*.25/C
Y = B*3/C + YP(NL)
IF (PTCK.LT.0.) GO TO 520
WRITE (6,400) N
400 FORMAT (1H0,5X,'LEAST SQUARE MATRIX FOR LEADING EDGE',
1 '13,' 'FCINTS LEAST-SQUARE FIT')/
WRITE (6,500) A1,B1,C1,D1
WRITE (6,500) A2,B2,C2,D2
WRITE (6,500) A3,B3,C3,D3
WRITE (6,500) FA1,FB1,FC1
WRITE (6,500) FA2,FB2,FC2
WRITE (6,500) FA3,FB3,FC3
WRITE (6,500) X,Y,A,B,C,DET,DI
500 FORMAT (15D13.5)
520 CONTINUE
R2 = 0.
DXAM = C.
DX2M = C.
DO 650 I = N1,N2
Y = YP(I) - YP(NL)
X = A*Y + B*Y**2 + C*Y**3
DX = X - XP(I)
DX2 = C*X*DX
R2 = R2 + DX2
DXA = ABS(DX)
IF (DXAM.GE.DXA) GC TO 600
DXAM = CXA
DXAM = CXA
DX2M = CXAM**2

60C CONTINUE
IF (PTCK .LE. 0.) GC TC 65C
WRITE (6,70C) 1,X,Y,DX,DX2,R2,CXAM,DX2M,XP(1),VP(1)

65C CONTINUE
IF (PTCK .LE. 0.) GC TC 75C
RA = R2/A1
WRITE (6,70C) N,RA,DXAM

70C FORMAT (113,9G13.5)
75C CONTINUE
RETURN
ENTRY LIC
IF (DXAM .LE. 1.E-4) RETURN
WRITE (6,80C) DXAM

80C FORMAT (1F0.5X,'WARNING ??? DEVIATION OF THE LEADING EDGE POINTS',
1 ' FROM PARABOLA IS GREATER THAN 0.0001',/6X,'DXAM =',G13.4/)
RETURN
END
**SUBROUTINE GEOM**

**GEOMETRIC DEFINITION OF WING**

STANDARD BOEING INPUT FORMAT FOR WING SECTION DATA IS USED

OPTION FOR WING SECTION TRAILING EDGE CLOSURE ANGLE
AND BISECTOR SLCR BE AUTOMATIC COMPUTED IS AVAILABLE

LEADING EDGE SINGULAR POINT CAN BE AUTOMATIC COMPUTED BY INCLUDING THE OPTION TO CALL LSQR

**COMMON /PCKR/ PICK**
**COMMON /CPP/ NL**

DIMENSION XS(ND,1),YS(ND,1),ZS(1),XLE(1),YLE(1),
SLOP(1),TRAIL(1),XP(ND),YP(ND),NP(1)

**INITIALIZE INPUT PARAMETERS WHICH HAVE RECOMMENDED PROGRAM VALUES**

```
FN= 0.0
PX = 0.0
PZ = 0.0
FIX = 0.0
TRL = 0.0
SLT = 0.0
XSING = 0.0
YSING = 0.0
YSYM = 0.0
```

**INITIALIZE INPUT DATA**

```
RAD = 57.295779513082
READ (5,50C)
READ (5,51O) ZSYM,FNS,SWEEP,DIHED,FUS
IF (FNS.LT.3) RETURN
KSYM = ZSYM
IF (FUS.GE.0.) KSYM = 1
NS = FNS
WRITE (6,2)
2 FORMAT(6H0 ,FUSELAGE RAD )
WRITE (6,61O) FLS
WRITE (6,4)
4 FORMAT(6H0 ,SWEEP ,15H ,CIHEL)
WRITE (6,61O) SWEEP,DIHED
SWEEP = SWEEP/RAD
DIHED = DIHED/RAD
```
ISYMO = 1
XTEC = 0
CHORDO = 0
K = 1

11 READ (5,500)
READ (5,510) ZSI(K),XL,YL,CHORD,THICK,AL,FSEC
IF (K.EQ.1) ZS1 = ZS(1)
ALPHA = AL/RAD
IF (K.EQ.1.AND.FSEC.EQ.0.) GO TO 31
READ (5,500)
READ (5,510) FN
N = FA
READ (5,500)
N1 = N + 1
READ (5,520) (XP(N1-I),YP(N1-I),I=1,N)

52C FORMAT (6F14.0)
DO 26 I=1,N
IF (XP(I+1).LT.XP(I)) GC TC 26
NL = 1
GO TO 860
26 CONTINUE
860 CONTINUE
IF (FNE.LT.0.) GC TO 21
DYL = YF(1) - YF(2)
DXL = XP(1) - XP(2)
DYU = YF(N) - YP(N-1)
DXU = XP(N) - Xp(N-1)
TSU = EYL/DXU
TSL = EYL/DXL
TRL = ATAN2(DYL,DXL) - ATAN2(DYU,DXU)
SLT = (TSL+TSU)*.5
NB = FAE
CALL LSER (N,NE,XP,YP,XSING,YSING)
21 WRITE (6,600)
WRITE (6,22) ZS(K)

22 FORMAT (160 PROFILE AT Z = ,F10.5/
1 15H0 TE ANGLE ,15H0 TE SLOPE ,15H0 X SING , 2
15H0 Y SING )
WRITE (6,610) TRL,SLT,XSING,YSING
27 WRITE (6,620) NL,XP(NL),XP(I),YP(I),I=1,N
62C FORMAT (/// NL = ,I3,\, XP(NL) = ,F10.5 /// (XP,YP)///
1 (2X,5(2F10.5,4X)))
31 SCALE = CHORD/(XP(1) - XP(NL))
XLE(K) = XL + (XSING - XP(NL))*THICK*SCALE
YLE(K) = YL + (YSING - YP(NL))*THICK*SCALE
XX = XP(NL) + (XSING - XP(NL))*THICK
YY = YP(NL) + (YSING - YP(NL))*THICK
CA = COS(ALPHA)
SA = SIN(ALPHA)
DO 32 I=1,N
32 XS(I,K) = SCALE*{(XP(I) - XX)*CA + THICK*(YP(I) - YY)*SA}
YS(I,K) = SCALE*{(THICK*(YP(I) - YY)*CA - (XP(I) - XX)*SA)
SLOPT(K) = THICK*SLT - TAN(ALPHA)
TRAIL(K) = THICK*TRL/RAD
NP(K) = N
CHORDO = AMAX1(CHORDO,CHORD)
IF (YS(K) .LE. 0. .OR. ALPHA .NE. 0.) ISYM = C
WRITE (6,42) ZS(K)
42 FORMAT ('FSECTION DEFINITION AT Z = ',F10.5/, 1, 15H XLE, 15H YLE, 15H CHORD, 2, 15H THICKNESS, RATIO, 15H TWIST)
WRITE (6,610) XL,YL,CHORD,THICK,AL
YMIN = YP(NL)
YMAX = YMIN
DO 44 I = 1,N
IF (YP(I) .GE. YMIN) GC TO 43
JMIN = I
YMIN = YP(I)
43 IF (YP(I) .LE. YMAX) GG TO 44
JMAX = I
YMAX = YP(I)
44 CONTINUE
YDIFF = YMAX - YMIN
NN = N - 1
SUM = C.
DO 46 I = NL,NN
SUM = SUM + .5*(YP(I)+YP(I+1))*(XP(I+1)-XP(I))
46 CONTINUE
NM = NL - 1
DO 48 I = 1,NM
SUM = SUM + .5*(YP(I)+YP(I+1))*(XP(I+1)-XP(I))
48 CONTINUE
WRITE (6,300)
300 FORMAT (15H YMIN ,15H JMIN ,15H YMAX , ,15H JMAX ,15H YDIFF,15H AREA)
WRITE (6,320) YMIN,JMIN,YMAX,JMAX,YDIFF,SUM
320 FORMAT (1H 'G12.4,111,G19.4,111,G19.4,G15.4)
CALL LSC
IF (FSY .LE. 0.) GO TO 61
R = AMAX1(0.,(FSL*2 - YLE(I)**2))
Z = ZS(K) - LS1 + SCRT(R)
R = FSY**2/(YLE(K)**2 + Z**2)
ZS(K) = Z*(1. - R)
YLE(K) = YLE(K)*(1. + R)
S = R*XS(NL,K)
XLE(K) = XLE(K) - S
DO 52 J = 1, N
XNa(I,K) = XNa(I,K) + S
52 YS(I,K) = YS(I,K)*(1. + R)
61 K = K + 1
IF (K .LE. NS) GO TO 11
Z0 = 0.5*(ZS(I) + ZS(NS))
IF (K SYS .NE. 0) Z0 = ZS(I)
DO 62 K = 1, NS
XTE0 = AMAX1(XTE0, XNa(I,K))
62 ZS(K) = ZS(K) - Z0
ZTIP = ZS(NS)
RETURN
500 FORMAT(1X)
510 FORMAT(E10.6)
60C FORMAT(1H1)
61C FORMAT(F12.4, 7F15.4)
END
C**SLBROUTINE COORD*******************************************************************************
SUBROUTINE COORD (NX, NY, NZ, KSYM, ZLIM, XLIM, ZLIM, 
1 SY, AX, AY, AZ, PX, PZ, AO, BO, ZO)
C SETS UP STRETCHED PARABOLIC AND SPANWISE COORDINATES
DIMENSION AO(1), BO(1), ZO(1)
PI = 3.1415926535898
BOUND = .99
AX = .5
AY = .5
AZ = .5
XLIM = .625*BOUND
ZLIM = .625*BOUND
SY = .5
SCALZ = ZLIM/(1.000001*ZLIM)
LX = NX/2 +1
MX = NX +1
DX = 2.*EQUAC/NX
QR = PI/XLIM
R = PX/C
DO 12 I=1,MX
D = (I -LX)*DX
D = D +R*SIN(C*D)
IF (ABS(D) .LE. XLIM) GO TO 12
B = 1.
IF (D.LT.0.) B = -1.
A = 1. -((C -B*Xlim)*E)**2
C = A**AX
D = B*Xlim +(D -B*Xlim)/C
12 AO(I) = D
KY = NY +1
DY = BOUND/NY
DO 22 J=1,KY
D = (KY -J)*DY
D = 1. -D*D
C = A**AY
22 BO(J) = SY*D/C
LZ = NZ/2 +1
K1 = 1
K2 = NZ +1
DZ = 2.*EQUAC/NZ
QR = PI/ZLIM
R = PZ/C
IF (KSYM.EQ.0) GO TO 31
LZ = 3
K1 = 2
K2 = NZ +3
DZ = BOUND/NZ
31 RE  
32 DD K=K1,K2  
      = 1.*/(1. +R*SIN(Q) -ZLIM)  
      = (K -LZ)*OZ  
      = O* +R*SIN(Q*O)  
      IF (ABS(C) .LE. ZLIM) GC TO 32  
      B = I.  
      IF (D .LT. 0.) B = -1.  
      A = 1. - (C -B*ZLIM)*E)**2  
      C = A**OZ  
      D = B*ZLIM + (D -B*ZLIM)/C  
32 ZO(K) = SCALZ*C  
RETURN  
END
C**SLBROUTINE SINGLE
SUBROUTINE SINGLE (NS, NZ, KSYM, KTE1, KTE2, FUS, CHORDO, ZS, XLE, YLE,
   1 S, DIHED, XO, YO, ZC, YPO, ZPO, E1, E2, E3, IND)
C
GENERATES SINGULAR LINE FOR SQUARE ROOT TRANSFORMATION
DIMENSION ZS(1), XLE(1), YLE(1), XO(1), YC(1), ZO(1), YPO(1), ZPO(1),
   1 E1(1), E2(1), E3(1)
K1 = 1
K2 = NZ + 1
IF (KSYM*EC*0) GO TO 11
K1 = 2
K2 = NZ + 3
KTE1 = 3
11 DO 12 K=K1,K2
   IF (Z0(K).LT.ZS(1)) KTE1 = K + 1
   IF (ZC(K).LE.ZS(NS)) KTE2 = K
CONTINUE
12 CALL SFLIF (1, NS, ZS, XLE, E1, E2, E3, 2, 0, 2, 0, 0, 0, IND)
CALL INTPL (KTE1, KTE2, Z0, XO, 1, NS, ZS, XLE, E1, E2, E3, 0)
   S = CHORDO*TAN(SWEEP)
   S1 = CHORDO*E1(1)
   S2 = CHORDO*E1(NS)
CALL SFLIF (1, NS, ZS, YLE, E1, E2, E3, 2, 0, 2, 0, 0, 0, IND)
CALL INTPL (KTE1, KTE2, ZC, YC, 1, NS, ZS, YLE, E1, E2, E3, 0)
   T = CHORDO*TAN(DIHED)
   T1 = CHORDO*E1(1)
   T2 = CHORDO*E1(NS)
   XO(KTE1-1) = XO(KTE1) + XO(KTE1) - XO(KTE1+1)
   Y0(KTE1-1) = YC(KTE1) + Y0(KTE1) - YO(KTE1+1)
IF (KSYM*NE*0) GO TO 31
N = KTE1 - 1
22 DO 22 K=K1,N
   ZZ = (Z0(K) - Z0(KTE1))/CHORDO
   A = EXP(ZZ)
   X0(K) = XO(KTE1) + S*ZZ - (S1 - S)*T(1: -A)
   Y0(K) = YO(KTE1) + T*ZZ - (T1 - T)*T(1: -A)
22 N = KTE2 + 1
31 DO 32 K=N,K2
   ZZ = (Z0(K) - Z0(KTE2))/CHORDO
   A = EXP(-ZZ)
   X0(K) = XO(KTE2) + S*ZZ + (S2 - S)*T(1: -A)
   Y0(K) = YO(KTE2) + T*ZZ + (T2 - T)*T(1: -A)
32 DO 42 K=K1,K2
   YPO(K) = YO(K)
   ZPO(K) = ZO(K)
   IF (FUS.LE.0.0) GO TO 42
   A = *.5*(Z0(K))*2 - YO(K)*2 + FUS**2
   B = *.5*Z0(K)*Y0(K)
   S = SQRT(A**2 + B**2)
IF (S.GT.0.) T = 5*ATAN2(B,A)  
S = SQRT(S)  
YP0(K) = 5*Y0(K) + S*SIN(T)  
ZP0(K) = 5*Z0(K) + S*COS(T)  
CONTINUE  
RETURN  
END
C**SUBROUTINE SURF******************************************************************************
SUBROUTINE SURF (NC,NE,NS,NX,NZ,ISYM,KSYM,KTE1,KTE2,
  1  YAW,XTE0,XLIM,FIX,NP,XS,YS,ZS,SLCT,TRAIL,
  2  AO,XO,ZO,SO,SCAL,ZV,IV,ITE1,ITE2,
  3  XP,YP,SN,D1,D2,E3,ITE3)
C INTERPOLATES MAPPED WING SURFACE AT MESH POINTS
DIMENSION SO(NE,1),XS(NS,1),YS(NC,1),ZS(1),SLCT(1),TRAIL(1),
  1  AD(11),XO(11),ZO(11),SCAL(1),ZV(1),
  2  XP(1),YP(1),SN(1),D1(1),D2(1),D3(1),
  3  IVINE(1),NP(1),ITE1(1),ITE2(1)
PI  = 3.1415926535898
TYAW = TAN(YAW)
SS0  = XTE0/XLIM**2
DX   = 2. / NX
LX   = NX / 2  + 1
MX   = NX  + 1
MZ   = NZ  + 3
IVO  = 1  -ISYM -ISYM -ISYM
IV1  = -1  -ISYM
DO 2 K=1,NZ
ITE1(K) = MX
ITE2(K) = MX
DO 2 I=1,NX
IV(I,K) = -2
2  SO(I,K) = 0.
K   = KTE1
K2   = K2  + 1
K1   = K2  - 1
IF (ZS(K2) -Z0(K)) 21,25,23
R2   = (Z0(K) -ZS(K1))/(ZS(K2) -ZS(K1))
20  R1   = 1  -R2
I2   = (I3*NX)/16  + 1
I1   = NX  + 2  -12
IF (FIX.EQ.GC) GO TO 31
C = R1*XG(1,K1) +R2*XS(1,K2)
CC  = SQRT(C/SS0)
DO 26 I=2,NX
IF ((AC(I))  + 5*DX*I.T-ECC) I1 = I  + 1
IF ((AC(I))  - 5*DX*I.T.ECC) I2 = I
26 CONTINUE
31 KK   = K1
RR   = R1
41 N   = NP(KK)
ANGL  = PI  +PI
U    = 1.
V    = 0.
DO 42 I=1,N
R  = SQRT(XS(I,KK)**2 +YS(I,KK)**2)
IF (R.EQ.0) GO TO 43
ANGL  = ANGL + ATAN2((U*YS(I,KK) - V*XS(I,KK)),
                    (U*XS(I,KK) + V*YS(I,KK)))
U  = XS(I,KK)
V  = YS(I,KK)
R  = SQRT(R**2 +R)
XP(I) = R*CCS(.5*ANGL)
YP(I) = R*SIN(.5*ANGL)
GO TO 42
43 ANGL  = PI
U  = -1.
V  = 0.
XP(I) = 0.
YP(I) = 0.
42 CONTINUE
S  = AO(I2)/AMIN1(ABS(XP(I)),ABS(XP(N)))
SS  = .5/S**2
DO 44 I=1,N
XP(I) = S*XP(I)
YP(I) = S*YP(I)
ANGL  = ATAN(SLQPT(KK))
ANGL1 = ATAN(YS(1,KK)/XS(1,KK))
ANGL2 = ATAN(YS(N,KK)/XS(N,KK))
ANGL1 = ANGL - .5*(ANGL1 -TRAIL(KK))
ANGL2 = ANGL - .5*(ANGL2 +TRAIL(KK))
T1  = TAN(ANGL1)
T2  = TAN(ANGL2)
CALL SFLIF (I,N,XP,YP,D1,D2,D3,1,T1,1,T2,0,0,IND)
IF (IND.EQ.0) WRITE (6,500) KK,K1,K2,N,FR,R1,R2,ZS(KK)
500 FORMAT (12HOBAC MAPPING,4I10,4G13.4)
CALL INTPL (I1,I2,A0,SA,1,A,XP,YP,C1,D2,D3,C)
X1  = .25*XS(1,KK)
A  = SLQPT(KK)*(XS(1,KK) -X1)
B  = 1./(XS(1,KK) -X1)
ANGL  = PI +PI
U  = 1.
V  = 1.
M  = I1 -1
DO 52 I=2,M
XX  = SS*A0(I)**2
X  = B*(XX -X1)
Y = YS(I,KK) +A*ALOG(D)/D
R  = SQRT((XX)**2 +YY**2)
ANGL  = ANGL + ATAN2((U*YY - V*XX), (U*X) +V*YY))
U  = XX
V  = YY
```
R
52 SN(I) = R*SCRT(R +R)
A = R*SIN(.5*ANGL)
B = SLCFT(KK)*(XS(N,KK) -X1)
E = 1./(XS(N,KK) -X1)
ANGL = 0.
U = 1.*
V = 0.*
M = 12 +1
DO 54 I=M,NX
XX = SS*AO(I)**2
YY = BS*(XX -X1)
YY = YS(N,KK) +A*ATLOG(D)/D
R = SQRT(XX**2 +YY**2)
ANGL = ANGL +ATAN2((U*YY -V*XX),(U*XX +V*YY))
U = XX
V = YY
R = R*SCRT(R +R)
DO 52 I=2,NX
IF (KK.EQ.K2) GC TC 71
KK = K2
RR = R2
GO TO 41
71 S0 = SS0
IF (FIX.EQ.00)
1 S0 = (K1*XS(I,K1) +R2*XS(I,K2))/(AO(I)**2 -S0(I1,K)**2)
SCAL(K) = SS +S0
ITE1(K) = I1
ITE2(K) = I2
ZV(K) = Z0(K) -TYAW*(X0(K) +S0*AC(I1)*AO(I1))
DO 72 I=11,12
72 IV(I,K) = 2
M = I1 -1
DO 74 I=1,M
ZI = Z0(K) -TYAW*(X0(K) +S0*AO(I)*AO(I))
IF (ZI.GE.ZV(KTE1)) IV(I,K) = IV0
74 CONTINUE
M = 12 +1
DO 76 I=M,NX
ZI = Z0(K) -TYAW*(X0(K) +S0*AO(I)*AO(I))
IF (ZI.GE.ZV(KTE1)) IV(I,K) = IV0
76 CONTINUE
K2 = K2 -1
K = K +1
IF (K.LE.KTE2) GC TC 21
K1 = 2
K2 = N2
```
IF (KSYP.GE.0) GC TC 81
K1 = 3
K2 = NZ +2
81 SCAL(K) = SCAL(KTE2)
DO 82 1=1, MX
ZZ = Z0(K) - TYAW*(X0(K) + SS*A0(I)*AO(I))
IF (ZZ.LE.ZS(NS).AND.ZZ.GE.ZV(KTE1)) IV(I,K) = IV0
82 CONTINUE
K = K +1
IF (K.LE.K2) GO TO 81
SCAL(K) = SCAL(KTE2)
N = KTE2
IF (YAH.LE.0) GO TO 93
I0 = ITE1(KTE2) +1
DO 92 I=10, LX
N = N +1
92 ZV(N) = Z0(KTE2) - TYAW*(X0(KTE2) + SS*A0(I)*A0(I))
93 I = ITE1(KTE1)
ZV(KTE1-I) = Z0(KTE1-I) - TYAW*(X0(KTE1) + SS*A0(I)*A0(I))
ZV(N+1) = Z0(KTE2+1)
DO 102 K=K1,K2
DO 104 I=2,NX
IF (IV(I,K).GT.0) GC TC 104
IF (IV(I+1,K+1).GT.C.0.0) IV(I-I,K+1).GT.C) IV(I,K) = IV1
IF (IV(I+1,K-1).GT.C.0.0) IV(I-I,K-1).GT.C) IV(I,K) = IV1
104 CONTINUE
102 IF (SOL(LX,K).LT.1.E-05) IV(LX,K) = 0
IF (KSYP.NE.0) RETURN
N = KTE1 -1
112 CONTINUE
112 SCAL(K) = SCAL(KTE1)
RETURN
END
C**SBLROUTINE ESTIM**

**SBLROUTINE ESTIM**

C

**INITIAL ESTIMATE OF REDUCED POTENTIAL**

COMMON G(161,18,3),SO(161,35),VORT(115),ZV(115),
IV(161,35),ITE1(35),ITE2(35),
AO(161),BO(18),VO(35),VC(35),ZC(35),SCAL(35),
NX,NY,NZ,KTE1,KTE2,ISYM,KSYM,FUS,
YAW,SYAW,ALPHA,CA,SA,FMACH,N1,N2,N3,IC

MX = NX +1
MY = NY +2
MZ = NZ +3

DO 12 1=1,161
DO 12 J=1,18
DO 12 K=1,3

12 G(I,J,K) = 0.

DO 22 K=1,MZ
WRITE (13) ((G(I,J,1),I=1,MX),J=1,MY)
WRITE (11) ((G(I,J,1),I=1,MX),J=1,MY)

22 CONTINUE

K1 = KTE1 -1
K2 = KTE2 +ITE2(KTE2) -NX/2
DO 32 K=K1,K2

32 VORT(K) = 0.

IO = 1
RETURN
END
C**SUBROUTINE MIXFLO**

SUBROUTINE MIXFLO

C SOLUTION OF EQUATIONS FOR MIXED SUBSONIC AND SUPERSONIC FLOW

C USING FINITE VOLUME SCHEME

COMMON G(161,18,3),SU(161,35),V0(115),ZV(115),
IV(161,35),ITE1(35),ITE2(35),
A0(161),B0(18),X0(35),YC(35),Z0(35),SCAL(35),
NX,NY,NZ,KTE1,KTE2,ISYM,KSYM,FUS,
YAW,SYAW,SYAW,ALPHA,CA,SA,FMACH,N1,N2,N3,IO
COM/M/SPA/ GL(161,18),QQL(161,18),FL(161,18),
UL(161,18),VL(161,18),WL(161,18),
AL(161,18),BL(161,18),CL(161,18),
RESL(161,18)
COM/M/FLC/ P1,P2,P3,FRES,RES,FRES,FRES,FRES,ARES,DG,IG,JG,KG,AG,NSLUP
COM/M/FLC/ K,NL,LM,KY,MY,J1,K1,FMACH2,AAO,Q1,Q2,RV,TYAW,TOT
LX = NX/2 +1
MX = NX +1
KY = NY +1
MY = NY +2
J1 = 2
IF (FMACH.GE.1.) J1 = 3
TYAW = SYAW/CYAW
FMACH2 = FMACH**2
AAO = 1./FMACH**2 +2
G1 = 2./P1
Q2 = 1./F2 -1.
TOT = 0.
FRES = 0.
ARES = 0.
DG = 0.
AG = 0.
NSLUP = 0.
K1 = 3
K2 = NZ +2
IF (KSYM.EQ.1) GO TO TG 1
K1 = 2
IF (FMACH.GE.1.) K1 = 3
K2 = NZ
1 DO 2 M=2,3
READ (N1,ERR=101) ((G(I,J,M),I=1,MX),J=1,MY)
2 CONTINUE
K = 1
NV = KTE1 -1
RV = 2.
DO 12 I=1,MX
DO 12 J=1,MY
G(I,J,1) = G(I,J,2)
GL(I,J) = G(I,J,2)
QQL(I,J) = 0.
SL(I,J) = 0.
UL(I,J) = 0.
VL(I,J) = 0.
WH(I,J) = 0.
AL(I,J) = 0.
BL(I,J) = 0.
CL(I,J) = 0.

12 RESL(I,J) = 0.
IF (K_STSNE.EQ.0) GO TO 21
CALL YSLEEP
RV = 1.
GO TO 51
21 DO 22 J=1,MY
DO 22 I=1,NX
G(I,J,2) = G(I,J,3)
22 GL(I,J) = G(I,J,3)
REAL (A1,ERR=101) ((G(I,J,3), I=1,MX), J=1,MY)
WRITE (A2) ((G(I,J,1), I=1,MX), J=1,MY)
K = K + 1
GO TO 51
31 CALL YSLEEP
RV = 1.
IF (K_STSNE.EQ.2) GO TO 51
IO = ITE1(K) + 1
DO 42 I=IO,LX
M = NX + 2 -1
V = G(M,KY,1) - G(I,KY,1)
NV = NV + 1
42 VORT(NV) = VORT(NV) + P3*(V - VORT(NV))
51 IF (K_STSNE.EQ.2) GO TO 61
DO 52 J=1,MY
DO 52 I=1,MX
G(I,J,1) = G(I,J,2)
52 G(I,J,2) = G(I,J,3)
REAL (A1,ERR=101) ((G(I,J,3), I=1,MX), J=1,MY)
WRITE (A2) ((G(I,J,1), I=1,MX), J=1,MY)
K = K + 1
GO TO 21
61 DO 62 M=2,3
WRITE (A2) ((G(I,J,M), I=1,MX), J=1,MY)
62 CONTINUE
FRES = FRES/64.
ARES = ARES/(64.*TOT)
AG = AG/TOT
10 RETURN
101 IO = 0
C**SUBROUTINE YSWEEP
SUBROUTINE YSWEEP
C
ROW RELAXATION
C
FINITE VOLUME SCHEME
C
COMMON

G(161,18,3),SO(161,35),VORT(115),ZV(115),
IV(161,35),ITE1(35),ITE2(35),
AO(161),BO(18),XO(35),YO(35),Z0(35),SCAL(35),
NX,NY,Nz,KTE1,KTE2,ISYM,KSYM,FUS,
YAW,CYAW,SYAW,ALPHA,CA,SA,FMACH,N1,N2,N3,I C

COMMON/SPA/

GL(161,18),QL(161,18),FL(161,18),
UL(161,18),VL(161,18),WL(161,18),
AL(161,18),BL(161,18),CL(161,18),
RESL(161,18)

COMMON/FLG/
P1,P2,P3,FRES,FRES,IFRES,IFRES,IFRES,ARES,ARES,ARES,ARES,ARES,ARES
COMMON/SWP/

K,NV,LY,MX,KY,MY,J1,K1,FMACH2,AAO,Q1,Q2,RV,TFAW,TOT

DIMENSION

XR(161),YR(161),ZR(161),XRM(161),YRM(161),ZRM(161),
QM(161),QCP(161),DP(161),FP(161),
UM(161),UP(161),VM(161),VP(161),WM(161),WP(161),
AM(161),AP(161),BM(161),BP(161),CM(161),CP(161),

DIMENSION

ABM(161),ABP(161),BCM(161),BPM(161),
CAP(161),COP(161),CAP(161),COP(161),
RES(161),D(161),E(161),C(161)

PI = 3.1415926535898
BV = .125*RV
CV = .125*(2.-RV)
D(1) = 0.
E(1) = 0.
J2 = NX
J = 1
XS = X0(K)/SCAL(KTE1)
YS = Y0(K)/SCAL(KTE1)
ZS = Z0(K)/SCAL(KTE1)
XR = X0(K+1)/SCAL(KTE1)
YR = Y0(K+1)/SCAL(KTE1)
ZR = Z0(K+1)/SCAL(KTE1)
S2 = SCAL(K)/SCAL(KTE1)
S1 = .5*S2
SR = SCAL(K+1)/SCAL(KTE1)
SR1 = .5*SR2
FS = FUS/SCAL(KTE1)

DO 12 I=1,MAX
RESO(I) = 1.
QA(I) = 0.
P(I) = 0.
Q(1) = 0.

12

104
R(I) = 0.
QP(I) = 0.
FU(I) = 0.
FW(I) = 0.
CG(I) = 0.
XM(I) = XS + S1*(AO(I)**2 - (B0(J) + SO(I,K))**2)
YM(I) = YS + S2*AO(I)*(B0(J) + SO(I,K))
ZM(I) = ZS
XR(I) = XRS + SR1*(AO(I)**2 - (E0(J) + SO(I,K+1))**2)
YR(I) = YRS + SR2*AO(I)*(B0(J) + SO(I,K+1))
ZR(I) = ZRS

IF (FUS.GE.C.) GO TO 21
DO 14 I = 1, MX
   A = .25*(ZM(I)**2 - YM(I)**2) + FS**2
   B = .5*ZM(I)*YM(I)
   S = SQRT(A**2 + B**2)
   IF (S.GT.0.) T = 5*ATAN2(B,A)
   IF (B.EQ.0.) AND YM(I).GT.(FS + FS) T = .5*PI
   IF (B.EQ.0.) AND YM(I).LT.-(FS + FS) T = -.5*PI
   YM(I) = .5*YM(I) + S*SIN(T)
   ZM(I) = .5*ZM(I) + S*COS(T)
   A = .25*(ZR(I)**2 - YR(I)**2) + FS**2
   B = .5*ZR(I)*YR(I)
   T = S
   S = SQRT(A**2 + B**2)
   IF (S.GT.0.) T = 5*ATAN2(B,A)
   IF (S.EQ.0.) T = 5*ATAN2(B,A)
   S = SQRT(S)
   YR(I) = .5*YR(I) + S*SIN(T)
   ZR(I) = .5*ZR(I) + S*COS(T)

DO 22 I = 1, MX
   X(I) = XM(I)
   Y(I) = YM(I)
   Z(I) = ZM(I)
   XR(I) = XRS + SR1*(AO(I)**2 - (B0(J+1) + SO(I,K))**2)
   YR(I) = YRS + SR2*AO(I)*(B0(J+1) + SO(I,K+1))

IF (FUS.LE.C.) GO TO 31
DO 24 I = 1, MX
   A = .25*(ZM(I)**2 - YM(I)**2) + FS**2
B = .5 * ZM(I) * YM(I)
S = SQRT(A**2 + B**2)
T = 0.
IF (S < 0.0) T = .5 * ATAN2(B, A)
IF (B < 0.0) AND (YM(I) < 0.0) GT (FS + FS)) T = .5 * F1
IF (B < 0.0) AND (YM(I) > 0.0) LT -(FS + FS)) T = -.5 * P1
S = SQRT(S)
YM(I) = .5 * YM(I) + S * SIN(T)
ZM(I) = .5 * ZM(I) + S * COS(T)
A = .25 * (ZM(I)**2) - YRM(I)**2) + FS**2
B = .5 * ZM(I) * YRM(I)
T = 0.
S = SQRT(A**2 + B**2)
IF (S < 0.0) T = .5 * ATAN2(B, A)
S = SQRT(S)
YRM(I) = .5 * YRM(I) + S * SIN(I)
ZRM(I) = .5 * ZRM(I) + S * COS(T)
DO 32 I=1,NX
XX = XR(I+1) - XR(I) + XRM(I+1) - XRM(I)
XY = XR(I+1) - XR(I) + XM(I+1) - XM(I)
1 XZ = XR(I+1) + XR(I) - XRM(I+1) - XRM(I)
1 YX = YR(I+1) - YR(I) + YRM(I+1) - YRM(I)
1 YY = YR(I+1) + YR(I) - YRM(I+1) - YRM(I)
1 YZ = YR(I+1) + YR(I) + YRM(I+1) + YRM(I)
1 ZX = ZR(I+1) - ZR(I) - ZRM(I+1) + ZRM(I)
1 ZY = ZR(I+1) + ZR(I) + ZRM(I+1) + ZRM(I)
1 ZZ = ZR(I+1) + ZR(I) - ZRM(I+1) - ZRM(I)
1 FXX = YY*ZZ - YZ*ZY
FYX = YZ*XZ - YX*ZZ
FZK = YK*ZY - YX*ZX
FXY = ZY*XZ - ZZ*XY
FYY = ZZ*XX - XZ*XZ
FXY = ZX*XY - ZY*XX
FXZ = XY*YZ - XZ*YY
FYZ = XZ*YY - XX*YZ
FZZ = XX*YY - XY*YX
FM(I) = FXX*XX + FYX*XY + FZX*XZ
A = 1. / FM(I)
GX = G(I+1, J, 2) - G(I, J, 2) + G(I+1, J+1, 2) - G(I, J+1, 2)
GXR = G(I+1,J,3) - G(I,J,3) + G(I+1,J+1,3) - G(I,J+1,3)
GY = G(I+1,J,2) + G(I,J,2) - G(I+1,J+1,2) - G(I,J+1,2)
GXY = G(I+1,J,3) + G(I,J,3) - G(I+1,J+1,3) - G(I,J+1,3)
GXYR = G(I+1,J,3) - G(I,J,3) - G(I+1,J+1,3) + G(I,J+1,3)
GZ = G(I+1,J,3) + G(I,J,3) + G(I+1,J+1,3) + G(I,J+1,3)
1 = -(G(I+1,J,2) + G(I,J,2) + G(I+1,J+1,2) + G(I,J+1,2))
ABM(I) = GXYR * GXY
BCM(I) = GYR - GY
CAM(I) = GXR - GX
ABCM(I) = GXYR - GXY
GX = GXR + GX
GY = GYR +GY
U = (FX*XG + FYY*GY + FZX*GZ)*A + CA
V = (FX*XG + FYY*GY + FZY*GZ)*A + SA
W = (FX*XG + FYY*GY + FZZ*GZ)*A + SYW
QQM(I) = U*U + V*V + W*W
AA = AA0 - 2*QQM(I)
DM(I) = ABS(FMACH2*AA)
UM(I) = FXX*U + FXY*V + FZ*W
VM(I) = FXX*U + FYY*V + FZ*W
WM(I) = FXX*U + FXY*V + FZ*W
AM(I) = (FX*X2 + FYY*Y2 + FZ*Z2) - UM(I)*2/AA*A
BM(I) = (FX*X2 + FYY*Y2 + FZ*Z2) - VM(I)*2/AA*A
CM(I) = (FX*X2 + FYY*Y2 + FZ*Z2) - WM(I)*2/AA*A
32 DO 34 I=1,NX
34 DM(I) = DM(I)*CM(I)*SQRT(DM(I))
36 DO 36 I=1,NX
36 UM(I) = DM(I)*UM(I)
VM(I) = DM(I)*VM(I)
WM(I) = DM(I)*WM(I)
AM(I) = DM(I)*AM(I)
BM(I) = DM(I)*BM(I)
CM(I) = DM(I)*CM(I)
FM(I) = DM(I)*FM(I)
ABM(I) = (AM(I) + BM(I)) * ABM(I)
BCM(I) = (BM(I) + CM(I)) * BCM(I)
CAM(I) = (CM(I) + AM(I)) * CAM(I)
36 ABCM(I) = (AM(I) + BM(I) + CM(I)) * ABCM(I)
IF (J.LT.2) GO TO 71
41 DO 42 I=2,NX
42 QQR = QQP(I) + QCP(I-1) + QQP(I) + QQP(I-1)
FR = FP(I) + FP(I-1) + FM(I) + FM(I-1)
UR = UP(I) + UP(I-1) + UM(I) + UM(I-1)
VR = VP(I) + VP(I-1) + VM(I) + VM(I-1)
WR = WP(I) + WP(I-1) + WM(I) + WM(I-1)
QQ = RESC(I)*BV*(QQR + QQL(I,J))
F = BV*(FR + FL(I,J))
2
3

\[ \begin{align*}
B & = WP(I) + WP(I-1) + WM(I) + WM(I-1) \\
\frac{1}{2} & = -BCP(I) - BCP(I-1) + BCM(I) + BCM(I-1) \\
3 & = +ABCP(I) + ABCP(I-1) - ABCM(I) + ABM(I-1) + R(I) \\
RES(I) & = AV\times(A + B + PF - QP(I) - Q(I)) + RESL(I,J) \\
RESL(I,J) & = A - B \\
AR & = AP(I) + AP(I-1) + AM(I) + AM(I-1) \\
BR & = BP(I) + BP(I-1) + BM(I) + BM(I-1) \\
CR & = CP(I) + CP(I-1) + CM(I) + CM(I-1) \\
A & = AV + AR + AL(I,J) \\
B & = AV + ER + BL(I,J) \\
C & = AV + CR + GL(I,J) \\
AL(I,J) & = AR \\
BL(I,J) & = BR \\
CL(I,J) & = CR \\
A & = A + A \\
B & = A + B \\
C & = C + C \\
TP & = A + Q1*(B + C) + Q2*(ABS(FU(I)) + FV(I) + Fm(I)) \\
ST & = (B + Q2*FV(I) + CG(I)) + (C + Q2*Fm(I))*G(I,J,1) - GL(I,J) \\
F & = 0. \\
IF (QA(I) \leq E \cdot I) \text{ GC TG 53} \\
F & = FUU(I) \\
T & = TP + F \\
S & = T + S*Q-(F + FV(I) + Fm(I)) + F \\
TM & = T + S*Q-(FV(I) + CG(I) + Fm(I)) + G(I,J,1) - GL(I,J)) \\
IF (FU(I) \leq O \cdot Q) \text{ TM = TM} + Q2*FU(I) + F + F \\
TP & = TP + Q2*FRES(I) + F + F \\
TM & = TM + T*RESO(I-1) \\
B & = 1/(I - TM*E(I-1)) \\
D(I) & = B*(RES(I) + S + TM*L(I-1)) \\
E(I) & = B*TP \\
IF (J,L \leq J1, OR.K \leq L \cdot K1) \text{ GO TO 71} \\
I & = I2 \\
CG(I+1) & = 0. \\
DO 62 I = 2, I2 \\
GL(I,J) & = G(I,J,2) \\
CG(I) & = DI(I) + E(I)*CG(I+1) \\
G(I,J,2) & = G(I,J,2) + CG(I) \\
TO1 & = TOT + 1. \\
ARES & = ARES \times ABS(RES(I)) \\
IF (ABS(RES(I)) \leq E \cdot ABS(FRES)) \text{ GO TO 63} \\
FRES & = RES(I) \\
IRES & = I
FM(I) = FP(I)
UM(I) = UP(I)
VM(I) = -VP(I)
WM(I) = AP(I)
BM(I) = BP(I)
CM(I) = CP(I)
ABM(I) = -ABP(I)
BCM(I) = -BCF(I)
CAM(I) = -CAP(I)
ABCM(I) = -ABCP(I)

82 CONTINUE
DO 92 I = 2, NX
IF (IAES(I, IV(I, K))) GT 1 GO TO 92
RESL(I) = 0.
S = 1.
A = AMAX0 (0, IABS (IV(I, K)))
RESL(I, J) = S + A
AL(I, J) = A
CL(I, J) = 0.
92 CONTINUE
GO TO 41

101 S1 = .5*SCAL(K)
I = NX + 2 - 12
I1 = ITE1(K)
N = NV
IF (I1 GE I) OR (ISYM .EQ. 1) GO TO 103
V = G(I2, KY, 2) - G(I1, KY, 2)
VOR(T(NV)) = VOR(T(NV)) + 3*(V - VOR(T(NV)))
N = NV
103 I = I - 1
V = 0.
IF (IV(I, K) NE 1) GO TO 109
ZZ = ZO(K) - TVAK*(XO(K) + S1*A0(1)*A0(1))
105 IF (ZZ GE ZV(N-1)) GO TO 107
N = N - 1
GO TO 105
107 A = (ZZ - 2ZV(N-1))/(ZV(N) - 2ZV(N-1))
V = A*VGR(T(N)) + (1. - A)*VCFT(N+1)
109 M = NX + 2 - 1
G(I, KY+1, 2) = G(M, KY-1, 2) - V
G(I, KY+1, 2) = G(I, KY-1, 2) + V
G(M, KY+1, 2) = G(I, KY, 2) + V
IF (I1 GE 1) GO TO 103
G(I, KY, 2) = -.5*V
C**SLBROLTINE VELG

SUBROUTINE VELG (K, SV, SM, CP, XP, YP, XMAX, XMIN, YMAX, YMIN)

CALCULATES SURFACE VELOCITY

COMMON G(161,18), SO(161,35), VORT(115), ZV(115),
    IV(161,35), ITE1(35), ITE2(35),
    AO(161), BO(181), XQ(35), VC(35), ZC(35), SCAL(35),
    NX, NY, N2, KTE1, KTE2, ISYP, KSYM, FUS,
    YAW,CYAN, SYAW, ALPHA, CA, SA, FMACH, N1, N2, N3, IG

COMMON/SPA/
    XL(161), XL(161,18), YL(161,18), ZL(161,18),
    X(161,18), Y(161,18), Z(161,18),
    XR(161,18), YR(161,18), ZR(161,18),
    RESL(161,18)

COMMON /UVW/
    UU(161), UV(161), WW(161)

DIMENSION SV(1), SM(1), CP(1), XP(1), YP(1)

PI = 3.1415926535898
Q1 = .2*FMACH**2
TI = 1.0/(.7*FMACH**2)
MX = NX + 1
NMAX = 1
AV = .5
IF (KSYM.EQ.1 .AND. K.EQ.KTE1) GOTO 1
NMAX = 2
AV = .25
NMAX = 2
IF (K.NE.KTE1) GOTO 11
N = KTE1 - 2 + KSYM
DO 2 J=1,2
DO 2 I=1,MX
XL(I,J) = 0.
YL(I,J) = 0.
ZL(I,J) = 0.
XR(I,J) = 0.
YR(I,J) = 0.
ZR(I,J) = 0.

2 N = N + 1
XR = XQ(N)/SCAL(KTE1)
YR = YQ(N)/SCAL(KTE1)
ZR = ZQ(N)/SCAL(KTE1)
SZ = SCAL(N)/SCAL(KTE1)
FS = .5*S2
DO 12 J=1,2
M = NY + 2 - J
DO 12 I=1,MX
XL(I,J) = X(I,J)
YL(I,J) = Y(I,J)
ZL(I,J) = Z(I,J)
XR(I,J) = XR(I,J)
Y(I,J) = YR(I,J)
Z(I,J) = ZR(I,J)
XR(I,J) = XRS + S1*(AO(I)*2 - EQ(M) + SC(I,N))**2
YR(I,J) = YRS + S2*AO(I)*(BO(M) + SO(I,N))
ZR(I,J) = ZRS
IF (FUS.LE.0.) GC TC 21
DO 14 J=1,NX
  DO 14 I=1,NX
    A = 2.5*(ZR(I,J)**2 - YR(I,J)**2) + FS**2
    B = 2.5*ZR(I,J)*YR(I,J)
    S = SQRT(A**2 + B**2)
    T = .5*ATAN2(B,A)
    IF (B.EQ.0. .AND. YR(I,J).GT.(FS + FS)) T = .5*PI
    IF (B.EQ.0. .AND. YR(I,J).LT.(FS + FS)) T = -.5*PI
    S = SQRT(S)
    YR(I,J) = 2.5*YR(I,J) + S*SIN(T)
  14 ZR(I,J) = 2.5*ZR(I,J) + S*COS(T)
11 IF (N.LE.K) GO TO 11
10 DO 22 I=2,NX
  U(I) = 0.
  V(I) = 0.
  W(I) = 0.
  XX = X(I+1,1) - X(I,1)
  XY = X(I,2) - X(I,1)
  XZ = XR(I,1) - X(I,1)
  YX = Y(I+1,1) - Y(I,1)
  YY = Y(I,2) - Y(I,1)
  YZ = YR(I,1) - Y(I,1)
  ZX = Z(I+1,1) - Z(I,1)
  ZY = Z(I,2) - Z(I,1)
  ZZ = ZR(I,1) - Z(I,1)
  GX = G(I+1,J,2) - G(I,J,2)
  GY = G(I,J,-1,2) - G(I,J,2)
  GZ = G(I,J,3) - G(I,J,2)
23 FXX = YY*ZZ - YZ*ZY
  FYX = YZ*XZ - YX*ZZ
  FZX = YX*ZY - YZ*XX
  FXY = ZY*XZ - ZZ*XX
  FYY = ZZ*XX - ZX*XZ
  FXY = ZX*YY - ZY*XX
  FYZ = XY*YY - XZ*YY
  FYZ = XZ*YY - XX*YZ
  FZZ = XX*YY - XY*XX
  F = 1./((FXX*XX + FYX*XY + FZX*XZ)
U = U + (F*X*G) + (F*Y*G) + (F*Z*G) * F + C
V = V + (F*X*G) + (F*Y*G) + (F*Z*G) * F + S
W = W + (F*Z*G) + (F*Y*G) + (F*Z*G) * F + S

IF (M.EQ.2) GO TO 25
M = 2
XX = X(I,1) - X(I-1,1)
YY = Y(I,1) - Y(I-1,1)
ZZ = Z(I,1) - Z(I-1,1)
GX = G(I,J,2) - G(I-1,J,2)

GO TO 25

IF (N.EQ.NMAX) GO TO 27
M = 1
N = 2
XX = X(I+1,1) - X(I,1)
YY = Y(I+1,1) - Y(I,1)
ZZ = Z(I+1,1) - Z(I,1)
GX = G(I+1,J,2) - G(I,J,2)

GO TO 23

UU(I) = U + V + W
VV(I) = V
WW(I) = W

Q = SQRT(Q)
SV(I) = AMAX(0.0, (1.0 + Q1 * (1.0 - Q)))
SM(I) = FMACH * SV(I) / SQRT(Q)
CP(I) = T1 * (Q * 0.5 - 1.0)
XP(I) = SCAL(KTE) * X(I,1)

DO 22 I1 = 1, I2
22 XP(I) = SCAL(KTE) * X(I,1)

DO 32 I1 = 1, I2
XM1 = AMAX1(XMAX, XP(I))
XM2 = AMIN1(XMIN, XP(I))
YM1 = YM1 = AMAX1(YM1, YP(I))
YM2 = AMIN1(YMIN, YP(I))

RETURN
END
**SUBROUTINE CPLCT**

SUBROUTINE CPLCT (I3, I4, FMACH, XP, YP, CP, SM, I1, I2, KPLT)

C PLCTS CF AT COMPUTATIONAL MESH POINTS

COMMON /PCKR/ PTCK
COMMON /PARMT3/ XT3(161), YT3(161), ZT3(161),
1 COMMON /UVW/ UU(161), VV(161), WW(161)
COMMON /PIS/ XOC(161)
COMMON /SPARE/ LINE(90), DUMY(10)

DIMENSION KDE(1), XP(1), YP(1), CP(1), SM(1)

DATA KDE/1H, 1H*/
DATA IST/1H*/
NOI = C
IMIN = I1 + (I2 - I1)/2
CHC = XP(IMIN) - XP(I1)

2 FORMAT(40H0)PLOT CF CP AT COMPUTATIONAL MESH POINTS/

1 CH0 X , 1CH Y , 7HMACH NO. 8H CP
2 7H XCC : 2X, 5F-CP* = F8 - 4, 2X, 7HC-GRD = F1C . 4)
CPC = 1.0 - (+2*FMACH**2)**3.5 - 1.0/(7*FMACH**2)

DO 12 I = 1, 90
12 LINE(1) = KDE(1)
CPS = (((5. + FMACH**21/6.)**3 - 5 - 1.)/(.7*FMACH**2)
IF (KPLT .EQ. 0 .OR. KPLT .GT. 1) GO TO 15
WRITE (6,2) CPS, CHC

15 CONTINUE
KS = 3C .*(CP0 - CPS) + 4.5
IF (KS .LE. 90) LINE(KS) = IST
DO 22 I = 13, 14
K = 30 .*(CP0 - CP(I)) + 4.5
K = MIND(9C, K)
LINE(K) = KDE(2)
XCC = (XP(I) - XP(IMIN))/CHC
XOC(I) = XCC
IF (KPLT .EQ. 0 .OR. KPLT .GT. 1) GO TO 20
WRITE (6,1) XCC

20 CONTINUE
LINE(K) = KDE(1)
IF (I .EQ. 11 .OR. I .GT. 12) GO TO 22
NOI = AC1 + 1
XT3(NO1) = XP(I)
YT3(NO1) = YP(I)
ZT3(NO1) = Z
UT3(NO1) = LU(I)
VT3(NO1) = WV(I)
WT3(NO1) = W(I)

IF (K .EQ. KS) LINE(KS) = IST
IF (KPLT .EQ. 0 .OR. KPLT .GT. 1) GO TO 25
CALL IMAKT6
SUBROUTINE INVR6
COMMON /FCKR/ FTCK
COMMON /PARMT3/ XT3(161), YT3(161), ZT3(161), UT3(161), V13(161), W13(161), NO1, Z
DIMENSION P(161,6), TEMP(161)
EQUIVALENCE (XT3(1), P(1,1))
DO 30 J=1,6
DO 10 I=1,NCI
M=NO1-I+1
TEMP(M)=P(I,J)
10 CONTINUE
DO 20 I=1,NCI
P(I,J)=TEMP(I)
20 CONTINUE
WRITE (*,100) (I,XT3(I),ZT3(I),YT3(I),UT3(I), W13(I), V13(I), I=1,NO1)
WRITE (*,100) (I,XT3(I),ZT3(I),YT3(I),UT3(I), W13(I), V13(I), I=1,NO1)
RETURN
10C FORMAT(4H0 1,1X,6HXT3(I),4X,6HZT3(I),4X,6HYT3(I),4X,6HUT3(I),
  1 4X,6HT3(I),4X,6HT3(I),4X,6HT3(I),4X,6HT3(I),4X,6HT3(I),4X,6HT3(I),
  2 4X,6HT3(I),4X,6HT3(I),4X,6HT3(I),4X,6HT3(I),4X,6HT3(I),4X,6HT3(I),
  3 4X,6HT3(I),4X,6HT3(I),4X,6HT3(I),4X,6HT3(I),4X,6HT3(I),4X,6HT3(I),
  4 (I4,1X,6G10.3,2X,I4,1X,6G10.3))
C**SUBROUTINE FORCF**************************************************************************
SUBROUTINE FORCF (II,12,XP,YP,CP,AL,CHORD,XM,YM,CL,CD,CM)
C
CALCULATES SECTION FORCE COEFFICIENTS
DIMENSION XP(II),YP(II),CP(II)
RAD     = 57.295779513082
ALPHA   = AL/RAD
CL      = 0.
CD      = 0.
CM      = 0.
N       = 12 -1
DO 12 I=11,1
   DX = (XP(I+1) -XP(I))/CHORD
   DY = (YP(I+1) -YP(I))/CHORD
   XA = (.5*(XF(I+1) +XP(I)) -XM)/CHORD
   YA = (.5*(YP(I+1) +YP(I)) -YM)/CHORD
   CPA = .5*(CP(I+1) +CP(I))
   DCL = -CPA*DX
   DCC = CPA*DY
   CL = CL +DCL
   CD = CD +DCC
12
CM = CM +DCD*YA -DCL*XA
CCL = CL*COS(ALPHA) -CD*SIN(ALPHA)
CD = CL*COS(ALPHA) +CD*COS(ALPHA)
CL = DCL
RETURN
END
SUBROUTINE TCFOR(KTE1, KTE2, CHORD, SCL, SCD, SCM, X0, YPO, ZPO, 1
   CL, CD, CMP, CMR, CMY)
C
CALCULATES TOTAL FORCE COEFFICIENTS
DIMENSION CHORD(1), SCL(1), SCD(1), SCM(1), X0(1), YPO(1), ZPO(1)
SPAN = ZPO(KTE2) - ZPO(KTE1)
CL = 0.
CD = 0.
CMP = 0.
CMR = 0.
CMY = 0.
S = 0.

1
K = KTE1
CL = SCL(K) * CHORD(K)
CD = SCD(K) * CHORD(K)
QM = CHCFO(K) * (SCM(K) * CHORD(K) - SCL(K) * X0(K) + SCD(K) * YPO(K))

1
DZ = .5 * (ZPO(K+1) - ZPO(K))
AZ = .5 * (ZPO(K+1) + ZPO(K))
PL = SCL(K+1) * CHORD(K+1)
PD = SCD(K+1) * CHORD(K+1)
PM = CHCFO(K+1) * (SCM(K+1) * CHORD(K+1) - SCL(K+1) * X0(K+1) + SCD(K+1) * YPO(K+1))

CL = DZ * (PL + CL)
CD = DZ * (PD + CD)
CL = CL + CLA
CD = CD + CEA
CMP = CMP + DZ * (PK + QM)
CMR = CMR + AZ * CLA
CMY = CMY + AZ * CEA
S = S + DZ * (CHORD(K+1) + CHORD(K))

IF (K .LTE. KTE2) CC TC 11
CL = CL / S
CD = CD / S
CMP = CMP * SPAN / S ** 2
CMR = (CMR + CMR) / (S * SPAN)
CMY = (CMY + CMY) / (S * SPAN)
RETURN
END
C**SLBROLTNE REFIN**************************************************************************

SUBROUTINE REFIN

C

HALVES MESH

COMMON

G(161,18,3),SO(161,35),VORT(115),ZV(115),

IV(161,35),ITE(35),ITE2(35),

AQ(161),BO(18),X0(35),Y0(35),Z0(35),SCAL(35),

NX,NY,NZ,KTE1,KTE2,KSYM,FUS,

YAW,CYAW,SYAW,ALPHA,CA,SA,FMACH,N1,N2,N3,IO

MX = NX + 1
KY = NY + 1
MY = NY + 2
MZ = NZ + 3
MXO = NX/2 + 1
MYC = NY/2 + 2
MZO = NZ/2 + 1
K = 1
IF (KSYM.EQ.0) GO TO 11
MZO = NZ/2 + 3
REAC (N1,ERR=401) ((G(I,J,1),I=1,MX0),J=1,MYO)
K = 2
11 REAC (N1,ERR=401) ((G(I,J,1),I=1,MXC),J=1,MYO)
J = NY/2 + 1
JJ = KY
21 IF (I.GT.0) GO TO 31
J = J - 1
J = J - 1
31 G(I,J,J,1) = G(I,J,1)
II = I - 1
II = II - 2
IF (I.GT.0) GO TO 31
J = J - 1
J = J - 2
DO 42 J = 1,KY,2
DO 42 J = 1,NX,2
42 G(I,J,1) = 5*(G(I+1,J,1) +G(I-1,J,1))
DO 52 I = 1,MX
DO 54 J = 1,MY
52 G(I,MY,1) = 0.
G(I,NY,1) = 0.
G(MX,NY,1) = 0.
WRITE (N2) ((G(I,J,1),I=1,MX),J=1,MY)
K = K + 1
IF (K.LE.MZO) GO TO 11
REIND N1
REIND N2
READ (N2,ERR=401) ((G(I,J,1),I=1,MX),J=1,MY)
READ (N2,ERR=401) ((G(I,J,3),I=1,MX),J=1,MY)
WRITE (N1) ((G(I,J,1),I=1,MX),J=1,MY)
K = 1
IF (KSYP.NE.0) K = 2
111 DO 112 1 = 1,MY
DO 112 I = 1,MX
112 G(I,J,2) = -5*(G(I,J,1) + G(I,J,3))
122 WRITE (N1) ((G(I,J,M),I=1,MX),J=1,MY)
122 CONTINUE
IF (K.EQ.MZC) GC TC 201
DO 132 J = 1,MY
DO 132 I = 1,MX
132 G(I,J,1) = G(I,J,3)
REAL (K2,ERR=401) ((G(I,J,3),I=1,MX),J=1,MY)
201 GO TO 111
201 REWIND A1
REWIND A2
DO 202 K = 1,3
202 REAL (K2,ERR=401) ((G(I,J,M),I=1,MX),J=1,MY)
CONTINUE
WRITE (N2) ((G(I,J,1),I=1,MX),J=1,MY)
TYAw = SYAw/CYWw
NV = KTE1 - 1
VORT(NV) = 0.
K = 2
IF (KSYP.NE.0) GO TO TC 251
211 S1 = -5*SCAL(K)
N = NV
I = MX0 + 1
IF (K.LT.KTE1 OR K.GT.KTE2) GO TO 231
I1 = ITE1(K)
I2 = ITE2(K)
DO 212 I = I1,12
212 G(I,KY+1,2) = G(I,KY,2) + G(I,KY,2) - G(I,KY-1,2)
NV = NV + 1
VORT(NV) = G(I2,KY,2) - G(I1,KY,2)
N = NV
I = I1
IF (K.EQ.KTE2 OR YAw.LE.0.) GO TO 231
221 I = I + 1
M = NX + 2 - 1
NV = NV + 1
VORT(NV) = G(M,KY,2) - G(I,KY,2)
IF (I.LT.MX0) GC TO 221
I = I1 - 1
V = 0.
IF (IV(I,K) .NE. 1) GO TO 237
ZZ = ZO(K) - TIAH*(X0(K) + S1*AC(I)*AO(I))
233 IF (ZZ .GE. Z(V(N-1))) GO TO 235
N = N - 1
GO TO 233
235 A = (ZZ - Z(V(N-1)))/(Z(V(N)) - Z(V(N-1))
V = A*UQRT(N) + (1.0 - A)*UQRT(N-1)
237 M = NX + 2 - 1
G(I,KY+1,2) = G(M,KY-1,2) - V
G(M,KY+1,2) = G(I,KY-1,2) + V
IF (IV(I,K) .NE. -1) GO TO 241
G(I,KY,3) = .5*(G(I,KY,1) + .25*(G(I,KY,3) + G(M,KY,3))
IF (IV(I,K+1) .LT. 1)
1G(I,KY,3) = .5*(G(I,KY,3) + .25*(G(I,KY,1) + G(M,KY,1))
G(M,KY,3) = G(I,KY,3)
G(I,KY-1,2) = .5*(G(I,KY,2) + G(I,KY-2,2))
G(M,KY-1,2) = .5*(G(M,KY,2) + G(M,KY-2,2))
241 IF (I.GT.1) GO TO 231
G(I,KY,2) = -.5*V
G(M,KY,2) = -.5*V
251 K = K + 1
IF (K .GE. MZ) GO TO 261
DO 252 = 1, KY
DO 252 = 1, KX
G(I,J,1) = G(I,J,1)
252 G(I,J,2) = G(I,J,2)
WRITE (N2) ((G(I,J), I=1, KX), J=1, MY)
READ (N1), ERR=401' ((G(I,J,1), I=1, KX), J=1, MY)
GO TO 211
261 VORT(N+1) = 0.
DO 262 = 1, MZ
WRITE (N2) ((G(I,J,M), I=1, KX), J=1, MY)
262 CONTINUE
REWRIND N1
REWRIND N2
DO 302 = 1, MZ
READ (N2), ERR=401' ((G(I,J), I=1, KX), J=1, MY)
WRITE (N1) ((G(I,J,1), I=1, KX), J=1, MY)
302 CONTINUE
IO = 1
RETURN
401 IO = 0
RETURN
C**SUBROUTINE SMCC**

SUBROUTINE SMCC

COMMON

G(161,18,3),SO(161,35),QRT(115),ZV(115),
11 IV(161,35),ITE1(35),ITE2(35),
2 AO(161),BO(18),DO(35),Y(35),Z(35),SCAL(35),
3 NX, NY,NZ,KTE1,KTE2,ISYM,KSHP,FUS,
4 YAW, CYAW, SYAW, ALPHA, CA, SA, FMACH, N1, N2, N3, IO

MX = NX +1
KY = NY +1
MY = NY +2
MZ = NZ +3
K1 = 2
K2 = NZ +2

IF (KSHP.EQ.0) GO TO 1
K1 = 3
K2 = NZ +2

1 PX = 1.0 / 6.
PY = 1.0 / 6.
PZ = 1.0 / 6.

DO 2 L=1,3
READ (K),ERR=511 ((G(I,J,L),I=1,MX),J=1,MY)
2 CONTINUE
WRITE (A2) ((G(I,J,1),I=1,MX),J=1,MY)
K = K1
11 K = K +1

DO 12 J=3,NY
DO 14 I=2,NX
14 G(I,J,1) = (1.0 - FX - PY - PZ) * G(I,J,2)
1 + 5 * PX * (G(I+1,J,2) + G(I-1,J,2))
2 + 5 * PY * (G(I,J+1,2) + G(I,J-1,2))
3 + 5 * PZ * (G(I,J,3) + G(I,J,1))

12 G(MX,J,1) = G(MX,J,2)

DO 16 I=1,MX
G(I,1,1) = G(I,1,2)
G(I,2,1) = G(I,2,2)
G(I,KY,1) = G(I,KY,2)
16 G(I,MY,1) = G(I,MY,2)
WRITE (A2) ((G(I,J,1),I=1,MX),J=1,MY)
IF (K.EQ.K2) GO TO 31
DO 22 J=1,MY
DO 22 I=1,MX
G(I,J,1) = G(I,J,2)
22 G(I,J,2) = G(I,J,3)
READ (K),ERR=511 ((G(I,J,3),I=1,MX),J=1,MY)
GO TO 11
31 WRITE (A2) ((G(I,J,3),I=1,MX),J=1,MY)
*SUBROUTINE SPLIF**************************************************************************
SUBROUTINE SPLIF(M,N,S,F,FP,FPP,FPPP,KM,VM,KN,VN,MODE,FQM,IND)
C SPLINE FIT - JAMES R
C INTEGRAL PLACED IN FPPP IF MODE GREATER THAN 0
C IND SET TO ZERO IF DATA ILLEGAL
DIMENSION S(1),F(1),FP(1),FPP(1),FPPP(1)

IND = 0
K = IABS(N - M)
IF (K - 1) E1, E1, 1
1 K = (N - M)/K
J = M + K
DS = S(J) - S(I)
D = DS
IF (DS) I1, I1, I1
I1 DF = (F(J) - F(I))/DS
IF (K - 2) 12, 13, 14
12 U = .5
V = 3.*(DF - VM)/DS
13 GO TO 25
14 U = VM
GO TO 25
14 V = -1. *
V = -DS*VM
21 J = J
J = J + K
DS = S(J) - S(I)
IF (DS) I1, I1, 23
23 DF = (F(J) - F(I))/DS
B = 1. /(DS + DS + U)
B = B*DS
B = B*(6.*DF - V)
25 FP(1) = U
FPP(1) = V
V = (2.*U)*DS
V = 6.*DF + DS*V
IF (J - N) 21, 31, 21
31 IF (KN - 2) 32, 33, 34
32 V = (6.*VN - VI)/U
GO TO 35
32 V = VN
GO TO 35
33 V = (DS*VN + FPP(1))/(1. + FP(I))
34 B = V
D = DS
41 CS = S(J) - S(I)
L = FPP(I) - FPP(I)*V
FPP(I) = (V - U)/DS
FPP(I) = U
F(I) = (F(J) - F(I))/CS - CS*(V + U)/6.
V = U
J = I
IF (J - K) = 41, 51, 41
51 I = N - K
FPP(N) = FPPF(I)
FPP(N) = B
FP(N) = DF + D*(FPP(I) + B + B)/6.
INC = 1
IF (MODE) = 81, 81, 61
61 FPFP(J) = FQM
V = FPP(J)
71 I = J
J = J + K
DS = S(J) - S(I)
U = FPP(J)
FPPP(J) = FPPP(I) + .5*DS*(F(I) + F(J)) - CS*DS*(U + V)/12.
V = U
IF (J = N) = 71, 81, 71
81 IF (INC, EC, 1) GC TO 90
WRITE (6, 85) INC, MCDE, I, J, K, M, S(I), S(J), DS, L
85 FORMAT (6, HOCHECK, 6, 110, 4, GL3, 4,)
WRITE (6, 86) (S(I), F(I), I = M, N)
86 FORMAT (10GL13.4)
90 RETURN
ENC
C**SUBROUTINE INTPL **********************************************************
SUBROUTINE INTPL(F1,NI,S1,F,S1,F,FF,FPP,FPPP,MODE)
C INTERPCLATION USING TAYLOR SERIES - JAMESCN
C ADCC CORRECTION FOR PIECEWISE CONSTANT FOURTH DERIVATIVE
C IF MODE GREATER THAN 0
DIMENSION SI(1),F1(1),S(1),F(1),FP(1),FPP(1),FPPP(1)
K = IABS(N - M)
I = (N - M)/K
MIN = M
NIN = NI
D = S(N) - S(M)
IF (D*(SI(NI) - SI(MI))) 11,13,13
11 MIN = NI
NIN = MI
13 K = IABS(NIN - MIN)
IF (KI) 21,21,15
15 KI = (NIN - MIN)/KI
21 II = MIN - KI
C IF (MODE) 31,31,23
23 C I = 1
31 II = II + KI
33 I = I + K
IF (I - N) 25,37,35
35 IF (D*(S(I) - SS)) 33,33,37
37 J = I
I = I - K
SS = S(S - S(I))
FPPP = C*(FPPP(I) - FPPP(I))/S(J) - S(I))
FF = S(FPPP(I) + .25*SS*FPPP)
FF = FPP(1) + SS*FF/3.
FF = FPP(I) + .5*SS*FF
FI(I) = F(I) + SS*FF
IF (II - NIN) 31,41,31
41 RETURN
ENC
**SUBROUTINE VERTEC**

SUBROUTINE VERTEC(II, I2, XCCD, CP, NRD, ZPC, FMACH, YA, AL, 1
SCL, SCD, SCM, K)

C SUBROUTINE FOR VERSATEC PLOTTERING OF THE PRESSURE COEFFICIENT
C VS NON-DIMENSIONAL CHORD (X/C) FOR EACH SECTION OF THE FINAL
C MESH

C REAL PX(C(165), PCP(165), XCCD(85), XCLP(85), CPLO(85), CPUP(85),
1 XCCD(161), CP(161), ZP0(35), SCL(35), SCD(35), SCM(35),
2 FMACH, YA, AL
INTEGER I, J, NUM, NUM1, I1, I2, NRD, K

C INITIALIZE ARRAYS AND DATA TO ZERO.
NUM = C*O
NUM1 = C*O
DO 10 I=1, 165
   PX(C(I)) = 0.0
   PCP(I) = 0.0
10 CONTINUE
DO 20 J=1, 85
   XCCD(J) = 0.0
   CP(J) = 0.0
   ZP(J) = 0.0
   CPUP(J) = 0.0
20 CONTINUE

C READ IN X/C AND CP DATA INTO NEW ARRAY STARTING AT ARRAY
C ELEMENT NUMBER 1
DO 30 I=I1, I2
   PX(C(I-11)+1) = XCCD(I)
   CP(I-11)+1 = CP(I)
30 CONTINUE
PX(C(NNC I) = 1.0

C PUT THE DATA INTO TWO ARRAYS: ONE FOR THE LOWER SURFACE
C AND ONE FOR THE UPPER SURFACE
NUM = (NRD-1)/2
NUM1 = NUM + 1
DO 40 I=1, NUM1
   XCLP(I) = PX(C I)
   CPLC(I) = CP(I)
40 CONTINUE
DO 50 = NUM1, NRC
   PCUP(J-NLM) = PX(C(J)
   CPUP(J-NUM) = CP(J)
50 CONTINUE

C INITIALIZE THE VERSATEC PLOTTER SYSTEM
CALL PCLS (0.0,0.0,0.0)
CALL THE DATA TO AN 5.0 X 7.0 INCH SPACE
CALL SCALE (PXC,5.0,NRD+1)
CALL SCALE (PCP,7.0,NRD-1)

DRAW THE X AND Y AXIS
CALL AXIS (1.0,2.0,'X/C','-',3.5,0.0,C,PXC(NRD+1),PXC(NRD+2))
CALL AXIS (1.0,2.0,'PRESSURE COEFFICIENT (CF)','25,')

PUT SCALE FACTORS INTO TWO ARRAYS FOR UPPER AND LOWER SURFACE
XCLC(NL+1) = PXC(NRD+1)
XLCO(NL+1) = PXC(NRD+2)
CPLC(NL+1) = PCP(NRD+1)
CPLO(NL+1) = PCP(NRD+2)

PLCT THE DATA POINTS
CALL NEWPEN (2)
CALL PLCT (1.0,2.0,'-3')
CALL LINE (XCLG,CPG,NUM1,1,-1,3)
CALL LINE (XCLP,CPG,NUM1,1,-1,11)

PLACE TITLE AT TOP CF PAGE
CALL NEWPEN (3)
CALL SYMBCOL (1.25,7.5,0.2,'SECTION CP DATA ','0.0,16)
CALL NEWPEN (1)
CALL SYMBCOL (1.25,7.25,0.1,'** = UPPER SURFACE ','0.0,18)
CALL SYMBCOL (1.25,7.0,0.1,'+ = LOWER SURFACE ','0.0,18)

PLACE THE EFFECTING INFORMATION ON PLOT
CALL NEWPEN (2)
CALL SYMBCOL (0.0,-.75,0.1,'SPAN STATION = ','0.0,15)
CALL NUMBER (999,999,0.1,'ZPG(K) ','0.0,23)
CALL SYMBCOL (0.0,-1.0,0.1,'MACF ','0.0,6)
CALL NUMBER (999,999,0.1,'FMACH ','0.0,23)
CALL SYMBCOL (2.0,-1.0,0.1,'YAW ','0.0,5)
CALL NUMBER (999,999,0.1,'YA ','0.0,3)
CALL SYMBCOL (4.0,-1.0,0.1,'ADA ','0.0,5)
CALL NUMBER (999,999,0.1,'AL ','0.0,3)
CALL SYMBCOL (0.0,-1.25,0.1,'CL ','0.0,6)
CALL NUMBER (999,999,0.1,'SCL (K) ','0.0,5)
CALL SYMBCOL (2.0,-1.25,0.1,'CD ','0.0,5)
APPENDIX F

THIS APPENDIX PRESENTS THE SOURCE CODE FOR THE INTERACTIVE PROGRAM FLO27IN

C**/* FLO27IN FORTRAN-1 C/O 05/83 JACK PASCHALL AERO ENGINEERING
C A PROGRAM TO INTERACTIVELY DEFINE A POTENTIAL FLOW PROBLEM.
C THIS PROGRAM WRITES AN INPUT FILE "FLO27 DATA" TO THE USERS
C A DISK WHICH CAN BE SUBSEQUENTLY EXEEDED FOR JOB CARD ENTRY AND
C BATCH SYSTEM EXECUTION.
C**/* REAL T1(E1), T2(E1), T3(E1), T4(E1), T5(E1), T6(E1), T7(E1), T8(E1),
C 1 FMACH, YA, AL, CD, SYM, FNS, SWEEP, DIF, FUS, ZS, XL, YL, CHORC
C 2 TFICK, AT, FSEC, FN, YSYM, XP(161), YP(161), ZERG, SCWFN, SCWXP(67),
C 3 SCWYP(67), SYMFN, SYMP(59), SYMY(55), N572FN, N572XP(41),
C 4 N572YP(41), FPYN, FXP(31), FYP(31), F14FN, F14XP(47), F14YP(47),
C 5 A7FN, A7YP(31), A7YXP(31), LI SFN, LI XP(49), LI YP(49), N10FN,
C 6 N1CXP(37), N10YP(37), N34FN, N34XP(45), N34YP(45), N35FN,
C 7 N35XP(45), N35YP(45), N64FN, N64XP(45), N64YP(45), N66FN,
C 8 N66XP(45), N66YP(45), N6FN, N6XP(45), N6YP(45), N16YP(45), N16YXP(51),
C 9 N16YXP(51), N16XP(51), N65FN, N65XP(51), N65YP(51), N60FN,
C 10 N60XP(51), N60YP(51), N66XP(51), N66YP(51), NIN, NUM, NUM, NUM,
C 11 JC, JI, JN, N1, FLAG, NUM, SCWLN, SYMM, N57NUM, FPYNUM,
C 12 N16NUM, N16N, L15NUM, N10NUM, N34NUM, N35NUM, N64NUM, N66NUM,
C**/* DATA ZER0/0.0, SCWFN/67.0, SCWXM/67.0/
C SCWP/1.0, 9.393, 9.683, 5522, 9361, 5038, 8713, 8382,
C 1 8043, 7694, 7335, 6962, 6525, 6475, 6174, 5742, 3835,
C 2 4293, 3683, 2863, 2163, 1727, 1216, 942, 648, 635,
C 3 4093, 3483, 2933, 2313, 1763, 1313, 972, 676, 661,
C 4 5093, 4483, 3933, 3413, 2863, 2413, 1972, 1674, 1621,
C 5 6962, 7335, 7694, 8043, 8382, 8713, 9361, 9522,
C 6 9633, 9853, 9937, 1.07,
C SCWYP/363526, 0.374, 0.381, 0.414, 0.438, 0.461, 0.506, 0.544, 0.579,
C 1 0.611, 0.641, 0.669, 0.695, 0.725, 0.751, 0.780, 0.805, 0.827,
C 2 0.847, 0.859, 0.880, 0.893, 0.926, 0.972, 0.764, 0.823, 0.854,
C 3 0.607, 0.581, 0.545, 0.507, 0.471, 0.420, 0.378, 0.336, 0.284,
C 4 0.296, 0.266, 0.216, 0.166, 0.080, 0.000, 0.094, 0.073, -0.010,
C 5 -0.030, -0.010, -0.012, -0.013, -0.007, -0.004, -0.001, 0.033,
C 6 0.081, 0.125, 0.178, 0.222, 0.273, 0.317, 0.334, 0.347, 0.394,
C 7 0.339, 0.329, 0.308, 0.302, 0.2937
### NACA 16-005 Section Data

<table>
<thead>
<tr>
<th>C</th>
<th>16X</th>
<th>16Y</th>
<th>N16XF</th>
<th>N16YF</th>
<th>N16</th>
<th>N16NF</th>
<th>N16NF</th>
<th>N16NF</th>
<th>N16NF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td>0000</td>
<td>0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td>9500</td>
<td>0950</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td>800</td>
<td>0800</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td>700</td>
<td>0700</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td>600</td>
<td>0600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td>500</td>
<td>0500</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### NACA 63-010 Section Data

<table>
<thead>
<tr>
<th>C</th>
<th>63X</th>
<th>63Y</th>
<th>N63XF</th>
<th>N63YF</th>
<th>N63</th>
<th>N63NF</th>
<th>N63NF</th>
<th>N63NF</th>
<th>N63NF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td>0000</td>
<td>0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td>9500</td>
<td>0950</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td>800</td>
<td>0800</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td>700</td>
<td>0700</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td>600</td>
<td>0600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td>500</td>
<td>0500</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### NACA 63A010 Section Data

<table>
<thead>
<tr>
<th>C</th>
<th>63A</th>
<th>63AN</th>
<th>N63AXF</th>
<th>N63AYF</th>
<th>N63A</th>
<th>N63AN</th>
<th>N63AN</th>
<th>N63AN</th>
<th>N63AN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td>0000</td>
<td>0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td>9500</td>
<td>0950</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td>800</td>
<td>0800</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td>700</td>
<td>0700</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td>600</td>
<td>0600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td>500</td>
<td>0500</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### NACA 64-01C Section Data

| N63AYF | 1.0000, 55.00, 52.00, 85.00, 45.00, 1000.00, 550.00, 650.00, 850.00, 900.00, 350.00 | 1.0000, 55.00, 52.00, 85.00, 45.00, 1000.00, 550.00, 650.00, 850.00, 900.00, 350.00 |
| N63AYF | 0.0021, 0.0025, 0.0003, 0.0025, 0.0025, 0.0025, 0.0025, 0.0025, 0.0025, 0.0025, 0.0025 |

### NACA 64010 Section Data

| N64AFN/51 | 1.0000, 65.00, 63.00, 65.00, 65.00, 65.00, 65.00, 65.00, 65.00, 65.00, 65.00 |
| N64AFN/51 | 0.0021, 0.0025, 0.0025, 0.0025, 0.0025, 0.0025, 0.0025, 0.0025, 0.0025, 0.0025, 0.0025 |

### NACA 65-010 Section Data

| N65OXF | 1.0000, 55.00, 52.00, 85.00, 45.00, 1000.00, 550.00, 650.00, 850.00, 900.00, 350.00 | 1.0000, 55.00, 52.00, 85.00, 45.00, 1000.00, 550.00, 650.00, 850.00, 900.00, 350.00 |
| N65OXF | 0.0021, 0.0025, 0.0003, 0.0025, 0.0025, 0.0025, 0.0025, 0.0025, 0.0025, 0.0025, 0.0025 |
C******************************************************************************NACA 65A010 SECTION DATA******************************************************************************

1  N65AXP/ 1.0000,  55.00,  6500,  8000,  7500,  7000,  8500,  7500,  7000,  6500,  6000,
2   5500,  5000,  5000,  5000,  4500,  4000,  3500,  3000,  2500,  2000,  1500,
3   1000,  0750,  0500,  0250,  0125,  0075,  0050,  0050,  0050,  0050,  0050,
4   0075,  0125,  0250,  0500,  0750,  1000,  1500,  2000,  2500,  3000,  3500,
5   3000,  3500,  4000,  4500,  5000,  5500,  6000,  6500,  7000,  7500,  8000,
6   8500,  9000,  9500,  1000,  1000,  1000,  1000,  1000,  1000,  1000,  1000,
7   1000,  1000,  1000,  1000,  1000,  1000,  1000,  1000,  1000,  1000,  1000,
C******************************************************************************NACA 66-010 SECTION DATA******************************************************************************

1  N66XNP/ 1.0000,  9500,  9000,  8500,  8000,  7500,  7000,  6500,  6000,  5500,  5000,
2   5500,  5000,  4500,  4000,  3500,  3000,  2500,  2000,  1500,  1000,  5000,
3   1000,  0750,  0500,  0250,  0125,  0075,  0050,  0050,  0050,  0050,  0050,
4   0075,  0125,  0250,  0500,  0750,  1000,  1500,  2000,  2500,  3000,  3500,
5   3000,  3500,  4000,  4500,  5000,  5500,  6000,  6500,  7000,  7500,  8000,
6   8500,  9000,  9500,  1000,  1000,  1000,  1000,  1000,  1000,  1000,  1000,
7   1000,  1000,  1000,  1000,  1000,  1000,  1000,  1000,  1000,  1000,  1000,
C******************************************************************************

FILE DEFINITIONS
CALL FRICHS ('FILE.CEF', '5', 'TERM 1')
CALL FRICHS ('FILE.CEF', '8', 'DISK 1', 'FLO27', 'CATAIN', 'A')

1
C TITLE PAGE AND INSTRUCTIONS
CALL FRICMS('CLRSCRN ')
WRITE (6,410)
WRITE (6,410)
WRITE (6,420)
WRITE (6,430)
WRITE (6,410)
WRITE (6,410)
WRITE (6,410)
WRITE (6,410)

C FIRST LINE INPUT DATA--DEFINE COMPUTATIONAL GRID
WRITE (6,450)
WRITE (6,460)
READ (5,1000) TITLE
CONTINUE
WRITE (6,470)
READ (5,*) FMESH
WRITE (6,480)
READ (5,*) FNX
WRITE (6,490)
READ (5,*) FNY
CALL FRICMS('CLRSCRN ')
WRITE (6,500)
READ (5,*) FNZ
WRITE (6,510)
READ (5,*) FPLT

C SUMMARY OF FIRST LINE INPUT DATA
CALL FRICMS('CLRSCRN ')
WRITE (6,520)
READ (5,*) ANS
IF (ANS.GE.2) GC TC 20
WRITE (6,521)
WRITE (6,522) FNX,FNY,FNZ,FMESH,FPLT
WRITE (6,530)
READ (5,*) ANS
IF (ANS.EQ.1) GO TO 10
CONTINUE

C SECOND, THIRD AND FORTH LINES INPUT DATA--ITERATION AND CONVERGENCE
TOLERANCE FOR GRID. NUMBER OF LINES EQUAL TO M = FMESH
M = IFI)(FMESH)
CALL FRICMS('CLRSCRN ')

C
WRITE (6,450)
DO 30 I=1,M
IF (I.EQ. 1) WRITE (6,541)
IF (I.EQ. 2) WRITE (6,542)
IF (I.EQ. 3) WRITE (6,543)
WRITE (6,550)
REAL (5,*) FIT(I)
WRITE (6,560)
REAL (5,*) COVO(I)
WRITE (6,570)
REAL (5,*) P10(I)
CALL FRICMS ('CLRSCRN ')
30 CONTINUE

C ---------------------------------------------------------------
C SUMMARY OF SECOND, THIRD AND FOURTH LINES INPUT DATA
C ---------------------------------------------------------------
CALL FRICMS ('CLRSCRN ')
WRITE (6,580)
READ (5,*) ANS
IF (ANS.GE.2) GC TO 40
WRITE (6,581)
WRITE (6,582) (FIT(I),COVO(I),P10(I)),I=1,M
WRITE (6,590)
REAL (5,*) ANS
IF (ANS.EQ.1) GO TO 20
40 CONTINUE

C ---------------------------------------------------------------
C FIFTH LINE INPUT DATA--MACH NO., YAW ANGLE, AOA, SKIN FRICTION DRAG
C ---------------------------------------------------------------
CALL FRICMS ('CLRSCRN ')
WRITE (6,450)
WRITE (6,600)
READ (5,*) FMACH
WRITE (6,610)
READ (5,*) YA
WRITE (6,620)
READ (5,*) AL
WRITE (6,630)
REAL (5,*) CD0

C ---------------------------------------------------------------
C SUMMARY OF FIFTH LINE INPUT DATA
C ---------------------------------------------------------------
CALL FRICMS ('CLRSCRN ')
WRITE (6,640)
READ (5,*) ANS
IF (ANS.GE.2) GC TO 50
WRITE (6,641)
WRITE (6,642) FMACH,YA,AL,CD0
WRITE (6,650)
READ (5,*) ANS
IF (ANS.EQ.1) Go TO 40

50 CONTINUE

C SIXTH LINE INPUT DATA--WING PLANFORM SYMMETRY, NUMBER OF SECTIONS,
Sweep, DIhedral Angle and Fuselage radius

CALL FRCMS ('CLRSCRN *')
WRITE (6,450)
WRITE (6,660)
READ (5,*) ZSYM
WRITE (6,670)
READ (5,*) FNS
WRITE (6,680)
READ (5,*) SWEEP
WRITE (6,690)
READ (5,*) DlHED
WRITE (6,700)
READ (5,*) FUS

C SUMMARY OF SIXTH LINE INPUT DATA

CALL FRCMS ('CLRSCRN *')
WRITE (6,710)
READ (5,*) ANS
IF (ANS.GE.21) Go TO 60
WRITE (6,711)
WRITE (6,712) ZSYM,FNS,SWEEP,DIHED,FLS
WRITE (6,720)
READ (5,*) ANS
IF (ANS.EQ.1) Go TO 50

60 CONTINUE

C WRITE JCL CARDS TO TOP OF FILE ON USER'S "A" DISK

WRITE (6,1200)
WRITE (6,1210)

C WRITE FIRST SIX LINES OF DATA TO FILE ON USER'S "A" DISK

WRITE (6,1010) TITLE
WRITE (6,1020)
WRITE (6,1030) FNX,FNY,FNZ,FMESH,FFLCT
WRITE (6,1040)
WRITE (6,1050) ((FIT(I),COVO(I),PI0(I)),I=1,M)
WRITE (6,1060)
WRITE (6,1070) FMACH, YA, AL, CMD
WRITE (6,1080)
WRITE (6,1090) ZSYM,FNS,SWEEP,CIHEC,FUS

C SECTION INFLT DATA

N = IFIX(FNS)
CALL FRTCMS ('CLRSRN ')
WRITE (6,450)
WRITE (6,730)
WRITE (6,410)
DO 200 I=1,N
WRITE (6,760) I
WRITE (6,770)
READ (5,*) ZS
WRITE (6,780)
READ (5,*) XL
WRITE (6,790)
READ (5,*) YL
WRITE (6,800)
READ (5,*) CHRL
WRITE (6,810)
READ (5,*) THICK
WRITE (6,820)
READ (5,*) AT
CALL FRTCMS ('CLRSRN ')
WRITE (6,830)
READ (5,*) FSEC
WRITE (8,1100)
WRITE (8,1110) ZS,XL,YL,CHORD,THICK,AT,FSEC
IF (FSEC .EQ. 0.0) GO TO 190
CALL FRTCMS ('CLRSRN ')
WRITE (6,550)
WRITE (6,740)
WRITE (6,741)
WRITE (6,750)
READ (5,*) ANS
FLAG = ANS
IF (FLAG .GT. 0) GO TO 70
WRITE (6,840)
READ (5,*) YSYM
CALL FRTCMS ('CLRSRN ')
WRITE (6,850)
READ (5,*) FN

C USER INPUT X AND Y COORDINATES FOR WING SECTION DEFINING GEOMETRY

NUM = IFIX(FN)
CALL FRTCMS ('CLRSRN ')
WRITE (6,450)
WRITE (6,860) I
DO 180 J=1,NUM
   WRITE (6,870)
   READ (5,4) XP(J),YP(J)
180 CONTINUE
IF (YSYM .EQ. 0.0) GC TC 185
   N1 = NUM + NUM
   J1 = NUM
   DO 182 IA=1,J1
      XF(N1-IA) = XP(IA)
      YF(N1-IA) = -YP(IA)
   183 CONTINUE
   NUM = N1 - 1
   FN = FN + (FN - 1.0)
185 CONTINUE
WRITE (8,1120)
WRITE (8,1130) FN
WRITE (8,1140)
WRITE (8,1150) ((XP(J),YP(J)),J=1,NUM)
GO TO 190
70 CONTINUE
C--------------------------------------------------------
C MENU INPUT X AND Y COORDINATES FOR WING SECTION DEFINING GEOMETRY
C--------------------------------------------------------
IF (FLAG .EQ. 1) GO TO 81
IF (FLAG .EQ. 2) GO TO 82
IF (FLAG .EQ. 3) GO TO 83
IF (FLAG .EQ. 4) GO TO 84
IF (FLAG .EQ. 5) GO TO 85
IF (FLAG .EQ. 6) GO TO 86
IF (FLAG .EQ. 7) GO TO 87
IF (FLAG .EQ. 8) GO TO 88
IF (FLAG .EQ. 9) GO TO 89
IF (FLAG .EQ. 10) GO TO 90
IF (FLAG .EQ. 11) GO TO 91
IF (FLAG .EQ. 12) GO TO 92
IF (FLAG .EQ. 13) GO TO 93
IF (FLAG .EQ. 14) GO TO 94
IF (FLAG .EQ. 15) GO TO 95
IF (FLAG .EQ. 16) GO TO 96
IF (FLAG .EQ. 17) GO TO 97
IF (FLAG .EQ. 18) GO TO 98
IF (FLAG .EQ. 19) GO TO 99
IF (FLAG .EQ. 20) GO TO 100
81 CONTINUE
WRITE (8,1120)
WRITE (8,1130) FP FN
WRITE (8, 1140)
WRITE (8, 1150) ((FPXP(J), FPYP(J)), J=1, FPNUM)
CONTINUE
WRITE (8, 1120)
WRITE (8, 1130) SYMFN
WRITE (8, 1140)
WRITE (8, 1150) ((SYMXP(J), SYMYP(J)), J=1, SYMNUM)
CONTINUE
WRITE (8, 1120)
WRITE (8, 1130) SCWFN
WRITE (8, 1140)
WRITE (8, 1150) ((SCWXP(J), SCWYP(J)), J=1, SCWNUM)
CONTINUE
WRITE (8, 1120)
WRITE (8, 1130) N572FN
WRITE (8, 1140)
WRITE (8, 1150) ((N572XP(J), N572YP(J)), J=1, N57NUM)
CONTINUE
WRITE (8, 1120)
WRITE (8, 1130) F14FN
WRITE (8, 1140)
WRITE (8, 1150) ((F14XP(J), F14YP(J)), J=1, F14NUM)
CONTINUE
WRITE (8, 1120)
WRITE (8, 1130) A7FN
WRITE (8, 1140)
WRITE (8, 1150) ((A7XP(J), A7YP(J)), J=1, A7NUM)
CONTINUE
WRITE (8, 1120)
WRITE (8, 1130) LISFN
WRITE (8, 1140)
WRITE (8, 1150) ((LISXP(J), LISYP(J)), J=1, LISNUM)
CONTINUE
WRITE (8, 1120)
WRITE (8, 1130) N1CFN
WRITE (8, 1140)
WRITE (8, 1150) ((N10XP(J), N10YP(J)), J=1, N10NUM)
CONTINUE
WRITE (8, 1120)
WRITE (8, 1130) N34FN
WRITE (8, 114C)
WRITE (8, 1150) ((N34XP(J),N34YP(J)),J=1,N34NUM)
GC TO 190 CONTINUE
WRITE (8, 1120)
WRITE (8, 1130) N35FN
WRITE (8, 1140)
WRITE (8, 1150) ((N35XP(J),N35YP(J)),J=1,N35NUM)
GC TO 190 CONTINUE
WRITE (8, 1120)
WRITE (8, 1130) N64FN
WRITE (8, 1140)
WRITE (8, 1150) ((N64XP(J),N64YP(J)),J=1,N64NUM)
GC TO 190 CONTINUE
WRITE (8, 1120)
WRITE (8, 1130) N66FN
WRITE (8, 1140)
WRITE (8, 1150) ((N66XP(J),N66YP(J)),J=1,N66NUM)
GC TO 190 CONTINUE
WRITE (8, 1120)
WRITE (8, 1130) N16FN
WRITE (8, 1140)
WRITE (8, 1150) ((N16XP(J),N16YP(J)),J=1,N16NUM)
GC TO 190 CONTINUE
WRITE (8, 1120)
WRITE (8, 1130) N63FN
WRITE (8, 1140)
WRITE (8, 1150) ((N63XP(J),N63YP(J)),J=1,N63NUM)
GC TO 190 CONTINUE
WRITE (8, 1120)
WRITE (8, 1130) N63AFN
WRITE (8, 1140)
WRITE (8, 1150) ((N63AXP(J),N63AYP(J)),J=1,N63AN)
GC TO 190 CONTINUE
WRITE (8, 1120)
WRITE (8, 1130) N64OFN
WRITE (8, 1140)
WRITE (8, 1150) ((N64OFP(J),N64OYP(J)),J=1,N64ON)
GC TO 190 CONTINUE
WRITE (8, 1120)
WRITE (8, 1130) N64AFN
WRITE (8, 1140)
WRITE (8, 1150) ((N64AXP(J), N64AYP(J)), J=1, N64AN)
CONTINUE
98 WRITE (8, 1120)
WRITE (8, 1130) N650FN
WRITE (8, 1140)
WRITE (8, 1150) ((N650AXP(J), N650AYP(J)), J=1, N650N)
CONTINUE
99 WRITE (8, 1120)
WRITE (8, 1130) N65AFN
WRITE (8, 1140)
WRITE (8, 1150) ((N65AXP(J), N65AYP(J)), J=1, N65AN)
CONTINUE
100 WRITE (8, 1120)
WRITE (8, 1130) N660FN
WRITE (8, 1140)
WRITE (8, 1150) ((N660AXP(J), N660AYP(J)), J=1, N660N)
CONTINUE
190 CALL FRICMS ('CLRSCRN ')
200 CONTINUE

C WRITE THREE LINES TO USER'S FILE INDICATING END OF CALCULATION

C WRITE (E, 1160)
WRITE (E, 1170)
WRITE (E, 1180) ZERC

C WRITE JCL CARDS TO BOTTOM OF FILE CN USER'S "A" DISK

C WRITE (E, 1220)
WRITE (E, 1230)

C INDICATE TO USER THAT INPUT IS COMPLETE

C CALL FRICMS ('CLRSCRN ')
WRITE (E, 450)
WRITE (E, 480)
WRITE (E, 410)
WRITE (E, 890)

C FORMAT STATEMENTS

C RETURN

410 FORMAT (1X,79H================================================================================
1====================================================================================)
FORMAT (1X,66H12FLO27 DATA INPUT PROGRAM, 33X,7HAE-4501//)
430 FORMAT (6X,66H THIS PROGRAM INTERACTIVELY WRITES A DATA FILE TO YOU
1R A DISK WHICH, 8X,66H CAN SUBSEQUENTLY BE USED TO RUN THE POTENTI
2AL FLOW PROGRAM //FLC27//)
440 FORMAT (1X,8H ENTER DATA FOR THE POTENTIAL FLOW PROGRAM IN FREE
1 FORMAT //, 8X,5H AFTER EACH QUESTION THE FORMAT IS GIVEN: (R) - RE
2AL //, 8X,14H (I) - INTEGER //, 8X,45H EXAMPLE: (R),8,J INPUT 2.5,6.789
30 OR (I) INPUT 5, //)
450 FORMAT (1X,79H**********************************************************************
1 FLO27 INPUT PARAMETERS
**********************************************************************)
460 FORMAT (1X,28H => ENTER THE PROBLEM TITLE: //, 5X,20H (64 LETTERS MAX
1 (M))
470 FORMAT (1X, 76H => ENTER THE NUMBER OF TIMES THE COMPUTATIONAL MESH
1 IS REFINED (FMESH): (R),//, 1X,22H1.0 = COARSE MESH ONLY, //, 1X,27H2.
20 = REFINED TO MEDIUM MESH, //, 1X,25H3.0 = REFINED TO FINE MESH)
480 FORMAT (1X, 56H => ENTER THE NUMBER CF COMPUTATIONAL CELLS IN CHORD
1 WISE //, 5X,41H DIRECTION FOR THE INITIAL MESH (FNX): (R),//, 5X,47H MA
2X MUM NC. = 160/2**N, WHERE N = FMESH - 1.0)
490 FORMAT (1X, 63H => ENTER THE NUMBER OF COMPUTATIONAL CELLS IN NORMAL
1 DIRECTION //, 5X,52H FRCM AIRFOIL SURFACE FOR THE INITIAL MESH (FNY
2): (R),//, 5X, 46H MAXIMUM NO. = 16/2**N, WHERE N = FMESH - 1.0)
500 FORMAT (1X, 55H => ENTER THE NUMBER OF COMPUTATIONAL CELLS IN SPAN W
1ISE //, 5X, 41H DIRECTION FOR THE INITIAL MESH (FN2): (R),//, 5X, 46H MA
2X MUM NC. = 32/2**N, WHERE N = FMESH - 1.0)
510 FORMAT (1X, 54H => ENTER FLAG FOR PRINTER-PLOTTING OF CP (FPLT): (1R
1),//, 1X, 5CH0.0 = PRINTER OUTPUT OF CP WITHU CL CP PRINTER-PLOT, //, 1X
2, 47H 1. C = PRINTER OUTPUT OF CP WITH CP PRINTER-PLOT, //, 1X, 57H2. 0 =
3PRINTER OUTPUT OF CP AND VERSATEC PLOT CF CP VS X/C, //, 7X, 34H FOR EA
4CH SECTION OF THE FINAL MESH)
520 FORMAT (1X, 33H SUMMARY OF FIRST LINE INPUT DATA?, //, 1X,25H => ENTER
11 = YES; 2 = NO)
521 FORMAT (1X, 3HFN2, 7X, 3HFN2, 7X, 3HFN2, 7X, 5HF MESH, 5X, 5HF PLT)
522 FORMAT (1X, F5.1, 15F, F4.1, 16X, F4.1, 16X, F3.1, 17X, F3.1)
530 FORMAT (1X, 29HCANGE FIRST LINE INPUT DATA?, //, 1X, 25H => ENTER 1 =
1YES; 2 = NO)
540 FORMAT (26X, 28H**** COARSE MESH PARAMETERS****)
542 FORMAT (26X, 28H**** MEDIUM MESH PARAMETERS****)
543 FORMAT (26X, 26H**** FINE MESH PARAMETERS****)
550 FORMAT (1X, 58H => ENTER MAXIMUM NUMBER CF ITERATIONS FOR MESH (FIT
1 1): (R),//, 5X, 41H RECOMMEND: 100 ITERATIONS FC COARSE MESH, //, 17X, 29H
220 ITERATIONS FOR MESH, //, 17X, 27H10 ITERATIONS FOR MESH 3)
560 FORMAT (1X, 66H => ENTER CONVERGENCE TOLERANCE FOR VELOCITY POTENTI
1AL (CGVC): (R),//, 5X, 18H RECOMMEND: .003001)
570 FORMAT (1X, 62H => ENTER SUBSONIC POINT RELEXATION FACTOR FOR MESH
26 FOR COARSE MESH, //, 16X, 19H1.3 FOR MEDIUM MESH, //, 16X, 17H1.2 FOR FIN
3 MESH)
FORMAT (1X,55HSUMMARY CF SECONC, THIRD AND FCRTH LINES OF INPUT DATA
1TA? ,/ ,1X,25H=> ENTER 1 = YES; 2 = NO)
581 FORMAT (1X,3HHIT,7X,4HCQVO,6X,3HP10)
582 FORMAT (1X,F5.1,5X,F7.3,F5.1)
590 FORMAT (//,1X,48HCHANGE THIRD AND FCRTH LINES INPUT DATA?, /,1X,25H=>
ENTER 1 = YES; 2 = NO)
600 FORMAT (1X,46H=> ENTER FREE STREAM MACF NUMBER (FMACH): (R))
610 FORMAT (1X,40H=> ENTER YAW ANGLE IN DEGREES (YA): (R))
620 FORMAT (1X,46H=> ENTER ANGLE OF ATTACK IN DEGREES (AL): (R))
630 FORMAT (1X,58H=> ENTER DRAG COEFFICIENT DUE TO SKIN FRICTION (CDO 1): (R),/ ,5X,48H(UNLESS OTHERWISE AVAILABLE 0.01 IS RECOMMENDED))
640 FORMAT (1X,33HSUMMARY CF FIFTH LINE INPUT DATA?, /,1X,25H=> ENTER
11 = YES; 2 = NO)
641 FORMAT (1X,5HFMAC,5X,2HYA,8X,2HAL,8X,3HCD)
642 FORMAT (1X,F4.2,2,EX,F3.1,7X,F3.1,7X,F8.6) / /
650 FORMAT (1X,29HCHANGE FIFTH LINE INPUT DATA?, //,1X,25H=> ENTER 1 =
YES; 2 = NO)
660 FORMAT (1X,52H=> ENTER WING PLANFORM SYMMETRY TRIGGER (ZSYM): (R)
1, / ,1X,41H=0 = YAWED WING HAS NO SPANWISE SYMMETRY, /,1X,38H=0 = S
2HEPT WING HAS SPANWISE SYMMETRY)
670 FORMAT (1X,78H=> ENTER NUMBER OF SECTIONS WHERE WING SECTION GEOM
ENTRY IS DEFINED (FNS): (R),/ ,5X,53H(VALUE MUST BE > OR = 3.0, BUT
2NOT GREATER THAN 11.0))
680 FORMAT (1X,58H=> ENTER LEADING EDGE SWEET ANGLE IN DEGREES (SWEEP
1): (R))
690 FORMAT (1X,48H=> ENTER DIHEDRAL ANGLE IN DEGREES (DIJED): (R))
700 FORMAT (1X,36H=> ENTER FUSELAGE RADIUS (FUS): (R),/ ,5X,33HNOTE: U
1SE 0.0 FOR WING-ALONE CASE)
710 FORMAT (1X,33HSUMMARY OF SIXTH LINE INPUT DATA?, /,1X,25H=> ENTER
11 = YES; 2 = NO)
711 FORMAT (1X,4HSYM,6X,3HFNS,7X,5HSWEEP,5X,5HCED,5X,3HFUS)
712 FORMAT (1X,F3.1,7X,F4.1,6X,F6.3,4X,F6.3,4X,F10.6) / /
720 FORMAT (1X,29HCHANGE SIXTH LINE INPUT DATA?, //,1X,25H=> ENTER 1 =
YES; 2 = NO)
730 FORMAT (5X,65HTHE NEXT SET OF INPUT DATA WILL BE REPEATED FOR EACH
1 WING SECTION?, /,5X,63HX AND Y COORDINATES DEFINING THE WING SECT
2ON SHAPE ARE ENTERED, /,5X,67HSTARTING WITH THE UPPER SURFACE TRAIL
3ING EDGE AND PROCEEDING AROUND, /,5X,35HTHE LOWER SURFACE TRAIL
4ING EDGE, /,5X,57HX AND Y COORDINATES ARE NORMALIZED WITH THE CHORD
5LENGTH, /,5X,46H WING SECTION DEFINING COORDINATES CAN BE INPUT, /,
6,5X,36HBY THE USER OR SELECTED FROM A MENU.)
740 FORMAT (28X,23H***WING SECTION MENU***, /,6X,34H0 = USER INPLT SEC
1TION COORD DATA, /,6X,19H1 = FLAT PLATE DATA, /,6X,49H2 = SYMMETRICA
2L WING (1% THICKNESS AT 30% CHORD), /,6X,61H3 = SUPERCRITICAL WING
3 (CAMBERED), 12% THICKNESS AT 30% CHORD), /,6X,51H4 = NASA 24-30 (CA
4MBER, 12% THICKNESS AT 30% CHORD), /,6X,52H5 = F14 WING (CAMBERED,
59, 5% THICKNESS AT 37% CHORD), /,6X,64H6 = A-7 WING (7 DEG DRCP AT
620% CHORD, 7% THICKNESS AT 43% CHORD), /,6X,64H7 = LISSAMAN 7769 A1
RFCL1 (CAMBERED, 11% THICKNESS AT 30% CHORD),//,6X,55H8 = NACA 0010
8 (SYMMETRICAL, 10% THICKNESS AT 30% CHORD),//,6X,58H9 = NACA 0010-3
94 (SYMMETRICAL, 10% THICKNESS AT 40% CHORD),//,5X,59H10 = NACA 00
25 (SYMMETRICAL, 10% THICKNESS AT 50% CHORD),//,5X,59H11 = NACA 00
110-64 (SYMMETRICAL, 10% THICKNESS AT 60% CHORD),//,5X,59H12 = NACA
200 (SYMMETRICAL, 10% THICKNESS AT 60% CHORD),//,5X,59H13 = NAC
3A 12-0C5 (SYMMETRICAL, 9% THICKNESS AT 50% CHORD),//,5X,58H14 = NAC
4A 63-01C (SYMMETRICAL, 10% THICKNESS AT 35% CHORD),//,5X,58H15 = NAC
5CA 63A010 (SYMMETRICAL, 10% THICKNESS AT 35% CHORD),//,5X,58H16 = NAC
6CA 64-010 (SYMMETRICAL, 10% THICKNESS AT 40% CHORD),//,5X,58H17 = NAC
7CA 64-010 (SYMMETRICAL, 10% THICKNESS AT 40% CHORD),//,5X,58H18 = NAC
6E-010 (SYMMETRICAL, 10% THICKNESS AT 40% CHORD),//,5X,58H19 = NAC

FORMAT (1X,35H=> ENTER DESIRED NUMBER FROM MENU.)

FORMAT (1X,22H**WING SECTION NUMBER,12,1X,13HPARAMETERS***,//,5X,
142HNOTE: ALL DIMENSIONS MUST BE IN SAME UNITS,//)

FORMAT (1X,40H=> ENTER THE SPANWISE COORDINATE FOR THIS SECTION (XL): (R))

FORMAT (1X,53H=> ENTER SECTION LEADING EDGE X COORDINATE (XL): (R))

FORMAT (1X,53H=> ENTER SECTION LEADING EDGE Y COORDINATE (YL): (R))

FORMAT (1X,43H=> ENTER SECTION CHORD LENGTH (CHORD): (R))

FORMAT (1X,55H=> ENTER SECTION THICKNESS SCALING FACTOR (THICK): (R))

FORMAT (1X,50H=> ENTER SECTION TWIST ANGLE IN DEGREES (AT): (R))

FORMAT (1X,65H=> ENTER FLAG FOR DEFINING NEW WING SECTION GEOMETRY
1Y (FSEC): (R),//,1X,40H1.0 = DEFINE A NEW WING SECTION GEOMETRY,//,1
2X,6H0. = COPY THE WING SECTION DEFINING GEOMETRY FROM PREVIOUS S
3ECTION,//,7X,38HNOTE: FOR FIRST SECTION MUST ENTER 1.0)

FORMAT (1X,44H=> ENTER FLAG FOR INDICATING WHETHER OR NOT,//,5X43H
1THE WING SECTION IS SYMMETRICAL (YSYM): (R),//,1X,20H0.0 = NCSYMME
2TRICAL,//,1X,29H1.0 = SYMMETRICAL SECTION,//,1X,53HNOTE: IF SYMMETRI
3CAL SELECTED, YOU ONLY HAVE TO INPUT,//,1X,46HDEFINING PCNTS FOR T
4E SECTION UPPER SURFACE,)

FORMAT (1X,58H=> ENTER NUMBER OF WING SECTION DEFINING POINTS (FN
1): (R),//,1X,38HNOTE: MAXIMUM NUMBER OF POINTS IS 161)

FORMAT (17X,22H***WING SECTION NUMBER,12,1X,22HX AND Y COORDIN
1ES***,//)

FORMAT (1X,67H=> ENTER WING SECTION DEFINING POINT X AND Y COORD.
1 (XP,YP): (R,R))

FORMAT (5X,64HTHREE ADDITIONAL DATA LINES WILL BE AUTOMATICALLY WR
1ITTEN TO THE,/5X,43HBOTTOM OF YOUR INPUT FILE. THESE LINES ARE:/
2,5X,18HEND OF CALCULATION,//,5X,3HFAXx/,//,5X,3HFy)

FORMAT (5X,70HTHIS IS COMPLETE THE INPUT DATA. A FILE WITH <FILENAME
1> <FILETYPE> FLG27,/,5X,69HDATAIN HAS BEEN WRITTEN TO YOUR "A" DIS
2K. IF YOU WISH TO MAKE FURTHER /, 5X, 62+ CHANGES TO YOUR INPUT DATA
3SIMPLY XEDIT THE CREATED DATA FILE. /, 5X, 62HTOP RUN THE POTENTIAL
4FLC Program (FLC27) USING YOUR DATA FILE. /, 5X, 61HXEDIT THE FILE
5AND ENTER THE ADDITIONAL CARDS (JOB CARD ETC.). /, 5X, 60HAS GUILINED
6IN THE INSTRUCTIONS, THEN SUBMIT THE FILE TO THE, /, 5X, 16HBATCH PR
7COSSOR. /, 1X, 4HBYE.)

1000 FORMAT (16(A4))
1010 FORMAT (1X,16(A4))
1020 FORMAT (2HFNX, 7X, 2HFNY, 7X, 2HFNZ, 7X, HEMESH, 5X, 2HFLCT)
1030 FORMAT (F5.1, 5X, F4.1, 6X, 2HF, 5X, 2F3.1, 7X, F3.1)
1040 FORMAT (3HFIT, 7X, 4HC0VC, 6X, 3HP10)
1050 FORMAT (F5.1, 5X, F7.6, 3X, F3.1)
1060 FORMAT (5HEMACH, 5X, 2HYA, 8X, 2HAL, 8X, 3HC0C)
1070 FORMAT (F4.1, 5X, F7.6, 3X, F3.1, 7X, F8.6)
1080 FORMAT (4H2SYM, 6X, 3HFNS, 7X, 3SHSHEEP, 5X, 5HDIHED, 5X, 3HFUS)
1090 FORMAT (F3.1, 7X, F4.1, 6X, 2F6.3, 4X, F6.3, 4X, F10.6)
1100 FORMAT (2H2S, 8X, 2HXL, 8X, 2HYL, 8X, 2HCHCRD, 5X, 5HTHICK, 5X,
12HAT, 8X, 4F FSEC)
1110 FORMAT (6(F8.4, 2X), F3.1)
1120 FORMAT (2FFN)
1130 FORMAT (F5.1)
1140 FORMAT (5HXP(I), 5X, 5HYP(I))
1150 FORMAT (6F10.7)
1160 FORMAT (1X, 18HEAD OF CALCULATION)
1170 FORMAT (2FFNX)
1180 FORMAT (F3.1)
1200 FORMAT (16H EXEC FLC27)
1210 FORMAT (17F//GO SYSIN DD *)
1220 FORMAT (2H*)
1230 FORMAT (2H/*)
ENC
LIST OF REFERENCES


### INITIAL DISTRIBUTION LIST

<table>
<thead>
<tr>
<th>No.</th>
<th>Copies</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1.  | 2      | Defense Technical Information Center  
Cameron Station  
Alexandria, Virginia 22314 |
| 2.  | 2      | Library, Code 0142  
Naval Postgraduate School  
Monterey, California 93943 |
| 3.  | 1      | Department Chairman, Code 67  
Department of Aeronautical Engineering  
Naval Postgraduate School  
Monterey, California 93943 |
| 4.  | 5      | Dr. Robert D. Zucker, Code 67Zu  
Department of Aeronautical Engineering  
Naval Postgraduate School  
Monterey, California 93943 |
| 5.  | 2      | Dr. Oscar Bitlarz, Code 67Bi  
Department of Aeronautical Engineering  
Naval Postgraduate School  
Monterey, California 93943 |
| 6.  | 2      | LCDR Lowell D. Boaz, Code 67Bo  
Department of Aeronautical Engineering  
Naval Postgraduate School  
Monterey, California 93943 |
| 7.  | 2      | CDR Jack Paschall III, Code 3107  
Naval Weapons Center  
China Lake, California 93555 |
A three-dimensional transonic, potential flow computer program, its conversion to IBM Fortran and utilization.