PSU-AERO-R-79/80-25

(NASA-CR-162816) COMPUTER PREDICTION OF THREE-DIMENSIONAL POTENTIAL FLOW FIELDS IN WHICH AIRCRAFT PROPELLERS OPERATE: COMPUTER PROGRAM DESCRIPTION AND USERS MANUAL (Pennsylvania State Univ.) 258 p

N80-17994

Unclas
G3/02 47313

COMPUTER PREDICTION OF THREE-DIMENSIONAL POTENTIAL FLOW FIELDS IN WHICH AIRCRAFT PROPELLERS OPERATE
(COMPUTER PROGRAM DESCRIPTION AND USERS MANUAL)

by

Stephen J. Jumper

August 1979

THE PENNSYLVANIA STATE UNIVERSITY
DEPARTMENT OF AEROSPACE ENGINEERING

80 N 17994

Errata Sheet

for

COMPUTER PREDICTION OF THREE-DIMENSIONAL POTENTIAL FLOW FIELDS IN WHICH AIRCRAFT PROPELLERS OPERATE (COMPUTER PROGRAM DESCRIPTION AND USERS MANUAL)

August 1979

by

Stephen J. Jumper

The Pennsylvania State University
Department of Aerospace Engineering

March 1980

The subroutine named VORTEX which is listed in the users manual has been modified to eliminate a defect which can cause errors when twin-engine aircraft configurations are modeled.

Due to this modification, the following seven corrections must be made to the users manual.

1. On page 21 in Table 2.1, the list of subroutine names in which Common Block /WING1/ appears must be changed. Remove the word VORTEX. Change that group of names to read

> MAIN PROGRAM, INPUT, COEFIC,
> COFSYM, WGEOM, WINGV, VELOCI,
> and VPROPS

2. On page 25 under item d concerning Common Statement /WING1/, delete the last line which reads: " -card VOR 2250 in subroutine VORTEX."

3. On page 32 in line 17, change the number 3991 to 3975.

4. On page 34 in line 8, change the number 363,782 to 363,766.

5. On page 34 in line 26, change the number 25,266 to 25,250.

6. In the Appendix A description of subroutine VORTEX found on pages 105 and 106, page 106 is correct. However, page 105 must be changed and should be replaced by the following attached page.

7. Finally, the listing of subroutine VORTEX found on pages 225 through 228 in Appendix B of the manual is no longer used. Instead, those four pages must be replaced with the listing of the newly modified subroutine VORTEX which is also enclosed on the last four pages of this errata.

The sample case presented in Appendix C of the users manual is not affected by the subroutine modification and no corrections are needed there.

ACKNOWLEDGMENT

This work was prepared under the National Aeronautics and Space Administration Grant NSG 1308.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

A method has been developed for predicting the potential flow velocity field at the plane of a propeller operating under the influence of a wing-fuselage-cowl or nacelle combination. The method is applicable to configurations moving at a constant low subsonic velocity. Additionally, the method predicts the potential flow on the body surface. The details of the theory are given in reference 1. A computer program has been written which predicts this three-dimensional potential flow field. The purpose of this manual is to describe the contents of the program, its input data, and its output results. The propeller plane flow field output is intended to be used as input to the propeller performance prediction program described in references 2 and 3.

METHOD OF SOLUTION

Several cases of body angle of attack and side slip may be solved together during one run of the program. The wing, if present, is represented by a single horseshoe vortex bound at the one quarter chord line. The trailing vortices are separated by a distance of $\pi/4$ times the span. A wing lift coefficient is specified by the user for each angle of attack so the wing vortex strength is not an unknown in the problem. The influence of the wing on the body is considered, but the body is assumed not to influence the wing. The body may be of any arbitrary shape. The body surface is represented by a mesh of triangular and quadrilateral panels. Certain panels may be specified as being relaxed boundaries through which some specified percentage of free stream velocity is specified. Such panels model inlet and outlet regions. All other panels are solid boundaries through which no velocity may pass.

A constant distribution of source strength acts on the panel under consideration. At other panels, the source distribution is lumped as a point source at the panel control points. At the panel under consideration, a summation is made of the normal velocity induced by the panel source on itself, by the other panel point sources, by the wing and the free stream velocity. To generate an equation the appropriate boundary condition is applied to this panel, zero normal velocity (if a solid panel) or normal velocity equal to the specified amount (if a relaxed boundary panel). Repeating this process for each panel produces a system of linear algebraic equations. The unknowns of the system are the set of body panel source strengths. One set of unknowns corresponds to one of the input body orientations. The equations are solved using the Gauss-Seidel iteration method. Once the panel source strengths are solved, the entire body flow field is calculable. The body surface velocities and pressure coefficients are computed. Lastly, the velocity components at specified points in the propeller plane are computed. These propeller plane velocity results may be punched on computer cards if desired. Format of card output is such that the cards may be used directly as part of the input data to the propeller performance prediction program given in references 2 and 3.

BRIEF PROGRAM DESCRIPTION

This program was written in single precision FORTRAN for use on an IBM 370 computer using an OS/370 operating system and using built-in library functions. Two auxiliary sequential scratch files are required by the program. These files were stored on an IBM 3330 Disk Pack. Details on allocating space for these files is given in Section 2 of this manual.

A listing of the program is given in Appendix B. This particular version of the program is dimensioned to a large size capable of processing up to 2596 panels and 6 body orientations. However, the program dimensions may be enlarged to enable handling of very long bodies, very wide bodies, or bodies requiring more panels or the program dimensions can be reduced to enable the program to fit in a smaller computer. Detailed procedures for altering program dimensions are given in Section 2 of this manual. Also the computer storage requirements as a function of program dimensions is explained in Section 2.

Section 2 of this manual gives an in-depth description of the mechanics of this program too detailed to mention in this brief description.

SECTION 1 BODY PANELING CONSIDERATIONS

1.1 Panel Distribution and Size Factors

The quality of and success in obtaining results from this three-dimensional potential flow program depends on the number of body panels used, and the distribution and size of panels.

Two factors should be kept in mind when paneling the configuration. The first factor, as discussed in reference 4, is concerned with the distribution of panels. The panel mesh must be finer (more and smaller panels) on regions of the body where abrupt surface curvature and rapid change in cross sectional shape occurs. Examples of such regions are nacelle inlet lips, locations where fairings or other protrusions are attached to the body, and the region where the canopy or cabin intersects the fuselage. Nearby geometry has the strongest influence on the propeller plane flow field. Since the main purpose of this program is to calculate the flow field in the propeller plane, it is desirable to most accurately define the regions of the body nearest to the propeller plane. Therefore, special attempts should be made to accurately panel the cowl or nacelle inlet face and foreward body regions using as fine a paneling mesh as possible. The second factor to consider deals with panel sizes. For best results, gradual changes in panel size must be made to blend paneling in regions of dense panel concentrations with panels in regions of sparse concentration. Reference 5 states that the characteristic dimensions of a panel should not differ by more than 50 percent from the dimensions of adjacent panels. Although this rule is difficult to obey with every panel on all configurations, it gives a guideline for the user to follow.

The two factors of panel mesh size and distribution give the user basic guidelines to follow in paneling the body. The source strength solutions will generally converge to a solution if the two guidelines are followed. The accuracy of the solutions obtained may depend somewhat upon how closely the guidelines are followed. If the body paneling strays too far from the above guidelines, the matrix system of equations may become so poorly behaved (not diagonally dominant) that the iterative solution process will diverge giving no results. On complex configurations, especially multiple body types as found in twin engine aircraft, the paneling mesh may lead to divergent solutions. In some situations, the user must use trial and error in modifying the paneling arrangement to get convergent solutions. Reference 5 contains a general discussion on the numerical behavior of the system of equations characteristic of this paneling method. Other geometry paneling factors affecting solution convergence are explained in sections 1.2 and 1.3 below.

1.2 Single Body Paneling Input Techniques

1.2.1 Cross Section and Periphery Point Input Rules--General

Examples of configurations considered as a single body are single engine aircraft cowl-fusslages without spinners, or the cowl-fuselage with the spinner attached to the cowling without a physical separation. Such a body is modeled by inputting the discrete point description of the body cross sections. See also section 3, card set 3 in this manual.

The method of paneling closely resembles the method used in reference 4. The body is cut into many cross sections perpendicular or nearly perpendicular to the body reference x axis. Cross sections need not always be purely coplanar cuts through the body. Rather, the cross section cuts may weave or curve forward and aft in an irregular fashion. See Figure 1.1. Such "quasi" cross sections may be needed to allow shifting of points forward and aft at awkwardly contoured regions on the body.

Cross section descriptions must be input in sequence starting at the front of the body and moving aft. See Figure 1.2. Note that a cross section may simply be a single point repeated several times. This occurs in the example in Appendix C. An exception to the front to aft ordering occurs in paneling the inside of inlets (as at the front of a cowl). In this situation, see Figure 1.3, the first cross section must be the one which defines the inlet contour and is farthest aft.

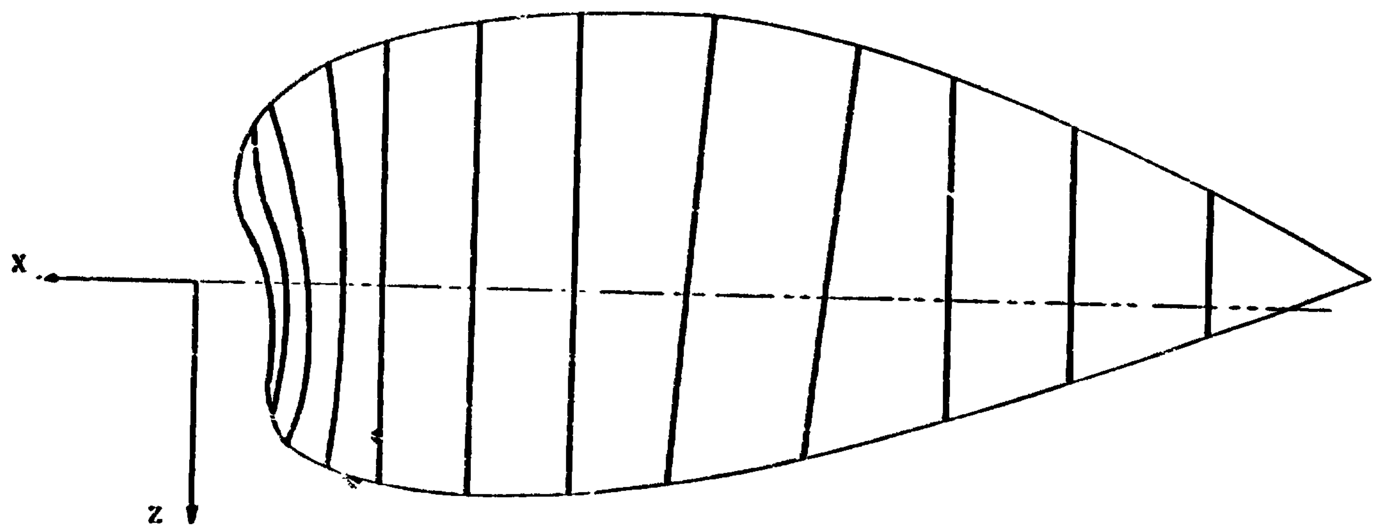


Figure 1.1 Body left side view showing plane cross section cuts and irregular cross section cuts which may be used to panel the body.

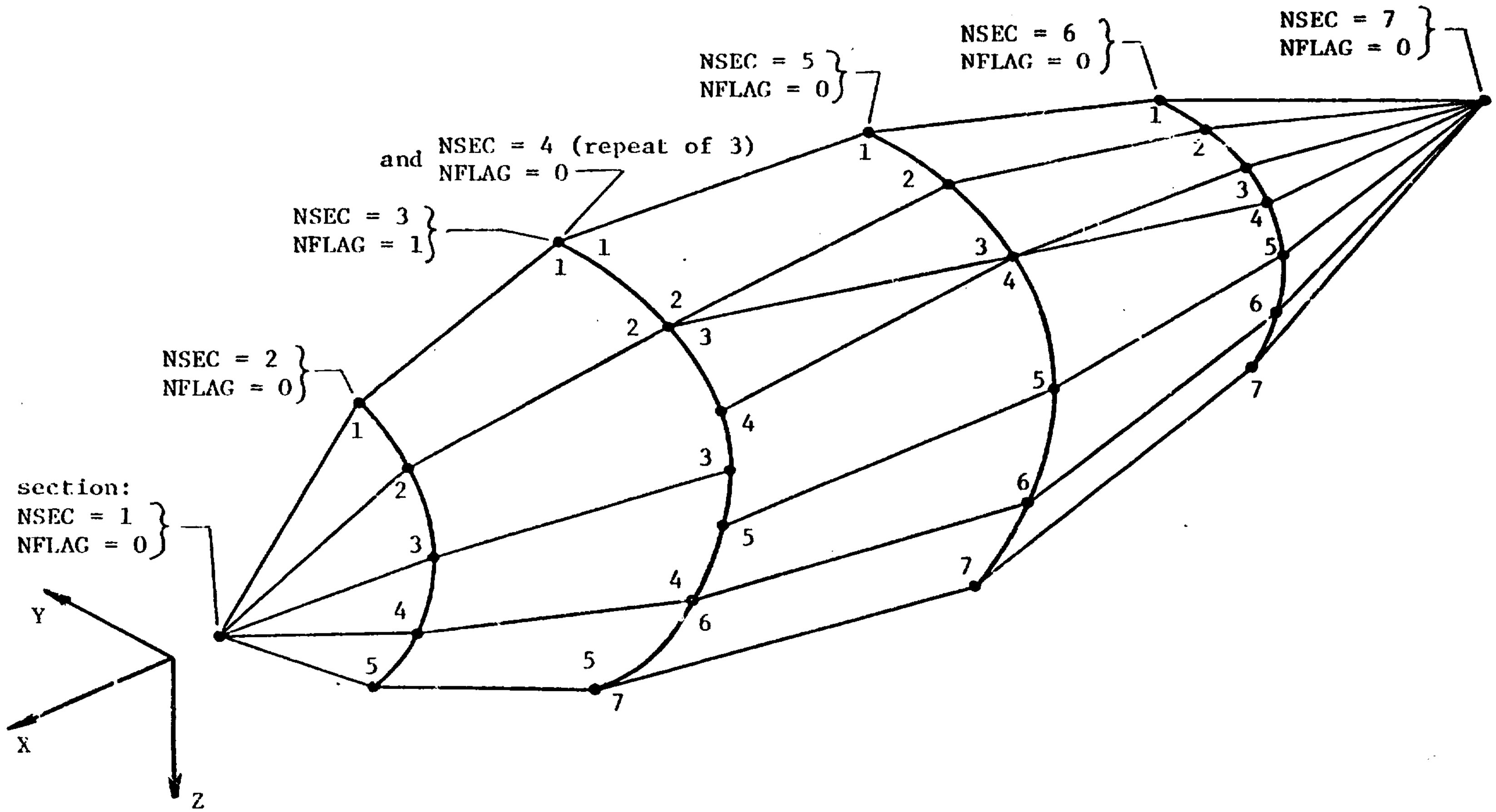


Figure 1.2 Sample body cross section input showing a repeated cross section description and indicating the input sequence of periphery points.

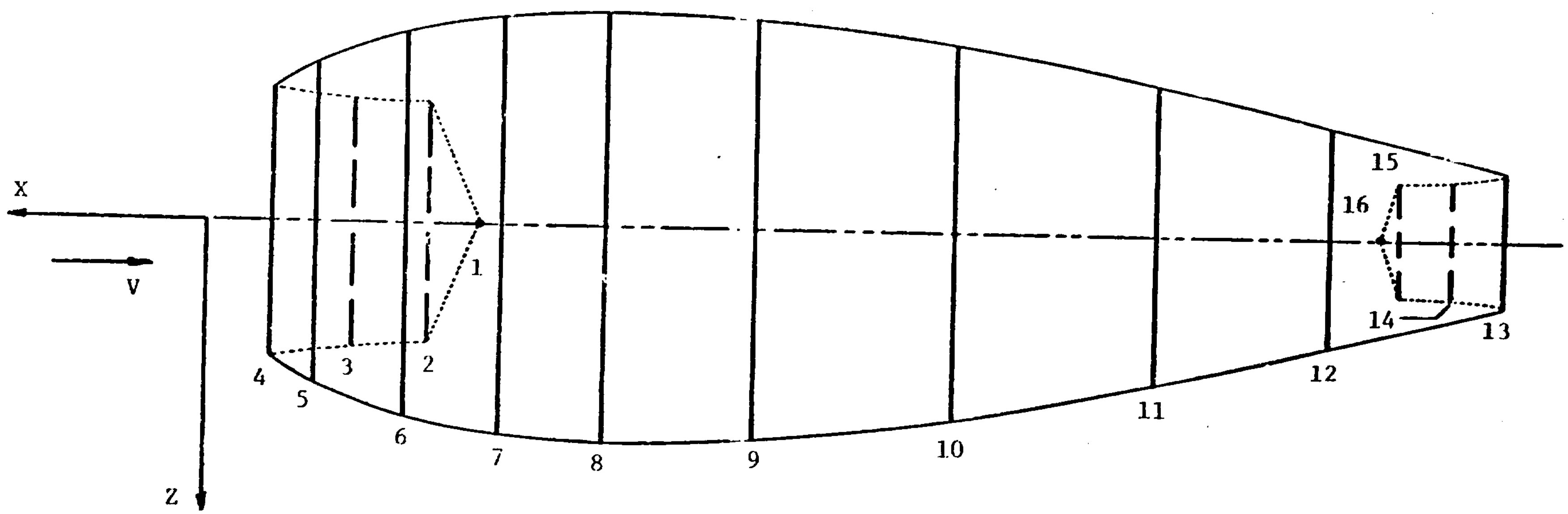


Figure 1.3 Left side view of a body showing the required input sequence of cross sections if the interior walls of inlet and exhaust ducts are paneled.

The inlet wall cross sections are input sequentially from aft to front to the inlet lip. Then the outer contours are specified in the normal front to aft sequence.

Each cross section is defined by specifying the coordinates of the periphery points on the section in the sequential order indicated in Figures 1.2 and 3.2.

Between two cross sections, the program generates a ring of panels using the input points for corners. See Figure 1.2. The ordering of the section points is important as the first point on one section is paired with the first point in the next section, etc. By specifying a point twice or more, that point will become the corner of one or more triangular panels. Otherwise, quadrilateral panels will be generated.

The input order of cross sections and periphery points shown in Figures 1.2, 1.3, and 3.2 must be followed to ensure the panels will have outward drawn normal vectors.

Refer to Figure 1.2. Panel rings will be generated between cross sections I and I+1 unless the flag signal, NFLAG, is given a value of 1 on section I. NFLAG=1 for section I indicates that section I will be repeated as section I+1 using a different number or distribution of periphery points. Thus no panels will be generated between sections I and I+1 in this situation. Use of NFLAG=1 is the method for increasing, decreasing, or altering the distribution of panels on the rings back along the body. It is essential that the number of points, NIP, on section I equal the same number of points on section I+1 if panels are generated between these sections.

Cross sections, between which panels will be placed, must never be allowed to intersect or coincide. That is, no point on the one section can be the same as a point on the other section. This would trigger an error in the program. However, cross sections may be concentric. This might be necessary in paneling the flat front face of a cylinder, for example.

It should be stated that although it is possible to panel concave body contours, such as the inside of an inlet, it has been found that in such cases, the equation solutions will usually diverge and no results will be obtained. It is not recommended that concave shapes such as the interior walls of inlets be paneled. Section 1.2.3 below presents an alternative technique for modeling inlets.

1.2.2 Use of Symmetric and Non-Symmetric Input Options

The input of cross section periphery points completely around the section, as explained in section 1.2.1, is the general or non-symmetric input option method (see Figure 3.2.b). This input generates panels on both sides of the body and can always be used regardless of body symmetry.

As is also outlined in section 3 of this manual, the symmetric body input option can be used if special conditions apply. These conditions are:

1. The body cross sections must all have left-right symmetry about a common (x-z) plane of symmetry. This plane of symmetry must contain the y=0 coordinate.

2. The root of the wing (if present) must be on this plane of symmetry.

3. Inlets and outlets (panels) must be symmetrically arranged about the plane of symmetry.

4. All body sideslip angles must be zero. The body may be oriented at angle of attack only.

When all these conditions are met, then only points on the left (negative y) half of each cross section need be input (see Figure 3.2.b). This is the symmetric input option. If this special input option is used, then the right side panels are mirror images of those on the left and they are automat ally generated by the program. The important benefit of the symmetric input option is that equations need only be created for the left side panels. The size of the system of equations is half of that produced when non-symmetric input is employed. Thus computing time and costs are greatly reduced. It is recommended that the symmetric input option be used whenever possible.

Typically, a single engine aricraft configuration can be paneled using the symmetry option if the non-symmetrical small ducts and other small protrusions are ignored, and if the spinner is attached to the fuselage symmetrically. If the spinner is yawed with respect to the fuselage (it typically is built this way), then the symmetric input option can not be used with this configuration unless the spinner is not modeled.

1.2.3 Relaxed (Inlet/Outlet) Panel Input Considerations

The body must always be paneled as a closed body. So if the front of the body is actually an open inlet, the inside of the inlet must be paneled as a closed cup shape. However this paneling method is not recommended as the solutions will fail to converge. Instead, the inlet face should be covered with "relaxed" panels which allow some flow to pass through them. To do this, panel the opening of the inlet as if it were closed off by a plate. Now in the inlet panel data cards (set 5 of input data--see section 3) the panel numbers of the inlet covering panels are specified. Assign a through flow velocity ratio to these panels. This velocity ratio will simulate the flow into the mouth of the inlet. The configuration will appear as a convex closed body with inlet inflow modeled. This should prevent solution divergence problems caused by concave paneled surfaces.

If the user is uncertain of the inlet/outlet panel numbers to specify, he should first run the configuration using the geometry test run option, NCALC = 1. To do this, set NCALC = 1 on card set 4 of input. Do not specify any inlet/outlet panels in card set 5 of input. Submit the data. The program will generate the body paneling but will not solve the flow equations. Take the output panel geometry from the run and find the numbers of the panels which should be inlets or outlets. Go back to the input data and specify these inlet/outlet panels in card set 5. Change the NCALC = 0 on card set 4. Then the data is resubmitted and a normal run will be made. All flow calculations will be made. Refer also to section 3 of this manual for use of the test run, NCALC = 1, option.

Any panel on the configuration may be an inlet or outlet panel. However the total number of these special panels must not exceed a certain limit (see MAXINF in section 2.1 and refer to section 3 of this manual).

If the symmetric input option is used, the inlet or outlet panels must be symmetrically positioned about the plane of symmetry, and only the inlet/outlet panels on the left side of the body are specified by the user in card set 5 of the input.

1.3 Multiple Body Paneling Input Techniques

1.3.1 Body Geometry Input Rules--General

The paneled configuration may consist of several discrete closed bodies. An example of this is a single engine aircraft fuselage-spinner combination in which the spinner is paneled as one isolated closed body and the fuselage is a second isolated closed body. A second example treated as a multiple body is a twin engine aircraft configuration with wing mounted nacelles. Each nacelle is a separate closed body. The fuselage itself is a third closed body and is not physically connected to the nacelles.

Multiple body configurations are paneled by considering one body at a time using the techniques given in section 1.2.1. All bodies are referenced to one common coordinate system. It does not matter which discrete body is defined first. For example, a nacelle can be paneled first, then the fuselage, and lastly another nacelle. Or the fuselage could be paneled first. If, on a twin nacelle aircraft, the left propeller plane is considered for the predictions, it may be convenient to input the left nacelle body first.

Input the first discrete body cross sections according to section 1.2.1 rules. Use the NFLAG = 1 parameter if a cross section repeat is desired. To distinguish the end of the first discrete body from the start of the second discrete body, it is necessary that NFLAG = 1 be specified on the last input section of the first body. In this use of NFLAG = 1, the last cross section of the first body will not be repeated. Instead, the flag signifies that no panels will be placed between this last cross section of the first body and the first cross section of the second body. Now input the second discrete body using the rules of section 1.2.1. As before, if a third body follows the second, use NFLAG = 1 on the last cross section of the second body. Panel the third discrete body etc. Panel the last discrete body using the rules of section 1.2.1. However, since the last discrete body will have none following it, do not specify NFLAG = 1 on the last section of the last body. The result is several closed bodies, each paneled according to single body rules with none of the bodies physically connected.

Through experience in paneling nacelle-fuselage-nacelle configurations, it has been found that this panel geometry often produces a system of equations which will diverge when solved iteratively. To obtain results, it may be necessary to ignore some regions of the configuration. For example, if only the left nacelle and foreward portion of the fuselage are paneled, the equations may be well behaved and will converge. Then if flow calculations are made on the left propeller plane, the results are still valid since the left nacelle and front fuselage have a dominant influence on this propeller plane. The right nacelle and rear fuselage are remote from the left propeller plane and it is safe to ignore these components in paneling the configuration.

1.3.2 Use of Symmetric and Non-Symmetric Input Options

Each discrete body of a multi-body configuration may always be input using the non-symmetric input option. To use this option, follow the procedures in sections 1.3.1 and 1.2.1 (see Figures 1.2 and 3.2.a).

The same conditions as given in section 1.2.2 must hold before the symmetric body input option may be used on multi-body configurations. These conditions are repeated here and have new meaning when applied to multiple body configurations:

1. All cross sections of all discrete bodies must be symmetric (left and right) about one common plane of symmetry. This x-z plane of symmetry must contain the y = 0 coordinate.

2. The root coordinate of the wing (if present) must lie on this plane of symmetry.

3. Inlets and outlets (panels) must be symmetrically arranged about the plane of symmetry.

4. All body sideslip angles must be zero. The configuration may be oriented at angle of attack only.

All of these four conditions must exist before the symmetric input option can be used. In using this symmetric option, only the points on the left (negative y) side of each body cross section are input (see Figure 3.2.b). The right side mirror image panels will be automatically created by the program. However, equations need only be written for the left side panels. Thus, the system of equations is only half as large as would occur if the non-symmetric option were used. Savings in computer time and cost is the result.

Note that rule 1 above requires that all discrete bodies of a multiple body configuration must lie in tandem on the common x axis. This is the only manner by which all cross sections of all bodies could be symmetric about one plane of symmetry. A multiple body configuration which could use the symmetric input option is a single engine fuselage-spinner combination, provided the spinner is not yawed with respect to the fuselage.

Although typical twin engine nacelle-fuselage-nacelle configurations would seem to be symmetric, they can never be paneled using the symmetry option as defined for this program, because each nacelle lies completely to one side of the plane of symmetry and does not obey rule 1.

Other examples of configurations which can not use the symmetry option are:

- nacelle-nacelle bodies side by side (rule 1 does not hold)
- fuselage-yawed spinner (rule 1 does not hold)
- fuselage-nacelle (rule 1 does not hold)
- wing mounted nacelle and wing. This is a single body which cannot use the symmetry option because the wing root does not lie on the nacelle plane of symmetry.

Unfortunately, as seen above, the symmetric input option has limited application when paneling mulciple body configurations.

1.3.3 Relaxed (Inlet/Outlet) Panel Input Considerations

Each discrete body of the multiple body configuration must be paneled as a closed body. If the bodies contain inlet or outlet openings, the use of relaxed inlet/outlet panels can be made. These relaxed panels can be used to model inlet openings without paneling the actual concave inner wall shape. The discussion of these relaxed panels given in section 1.2.3 applies equally to multiple body configurations.

1.4 Summary

The formulas used for calculating panel characteristics from discrete cross section periphery point input can be found in reference 1. This section of the manual should be used in conjunction with section 3 when making the input data deck.

Although any arbitrary body geometry can be paneled successfully, the system of equations obtained from the paneling network may not be well behaved numerically. As a result, possible solution divergence may occur during the iterative solution process. Thus, it may be necessary to use trial and error in getting a paneling network which accurately represents the body and also converges to a solution. Basically, the quality of the paneling depends upon the judgement and experience of the user.

SECTION 2 OVERALL PROGRAM DETAILS

2.1 Program Structure

The 3-D potential flow computer program consists of a main program used in setting up array dimensions, establishing program size, and calling the subroutines. Of the 14 subroutines used in the program, 10 are called directly by the main program. The overall structure of the program is illustrated by the flow charg on Figure 2.1. Also a description of the main program and each subroutine is given in Appendix A.

A copy of the program is listed in Appendix B. This particular version has been dimensioned to a fairly large size capable of handling up to 2596 body panels and up to six body orientations. The size of the program in Appendix B is one of the largest which can be compiled and executed on the IBM 370 computer facility at the Pennsylvania State University.

2.2 Program Dimension Size--Introduction

As the program dimension size determines the size of the configuration which may be handled, core storage and auxiliary file storage requirements, it may be necessary to change the size of the program given in Appendix B. Some or all of the dimensions of the program may be decreased to enable use of the program in a smaller computer installation, or to enable the program to be run in a smaller storage-higher priority run category at the installation. Conversely, some or all dimensions of the program may be increased to take advantage of a larger computer facility. This would enable configurations with more panels to be handled.

In the remaining portions of this section is a description of the variables which must be changed by the user to alter the program dimensions. Also described are the COMMON statements and DIMENSION statements needed, auxiliary files, and step-by-step procedures to be followed in altering the dimension size of the program. Then instructions for allocation of auxiliary file storage are given along with

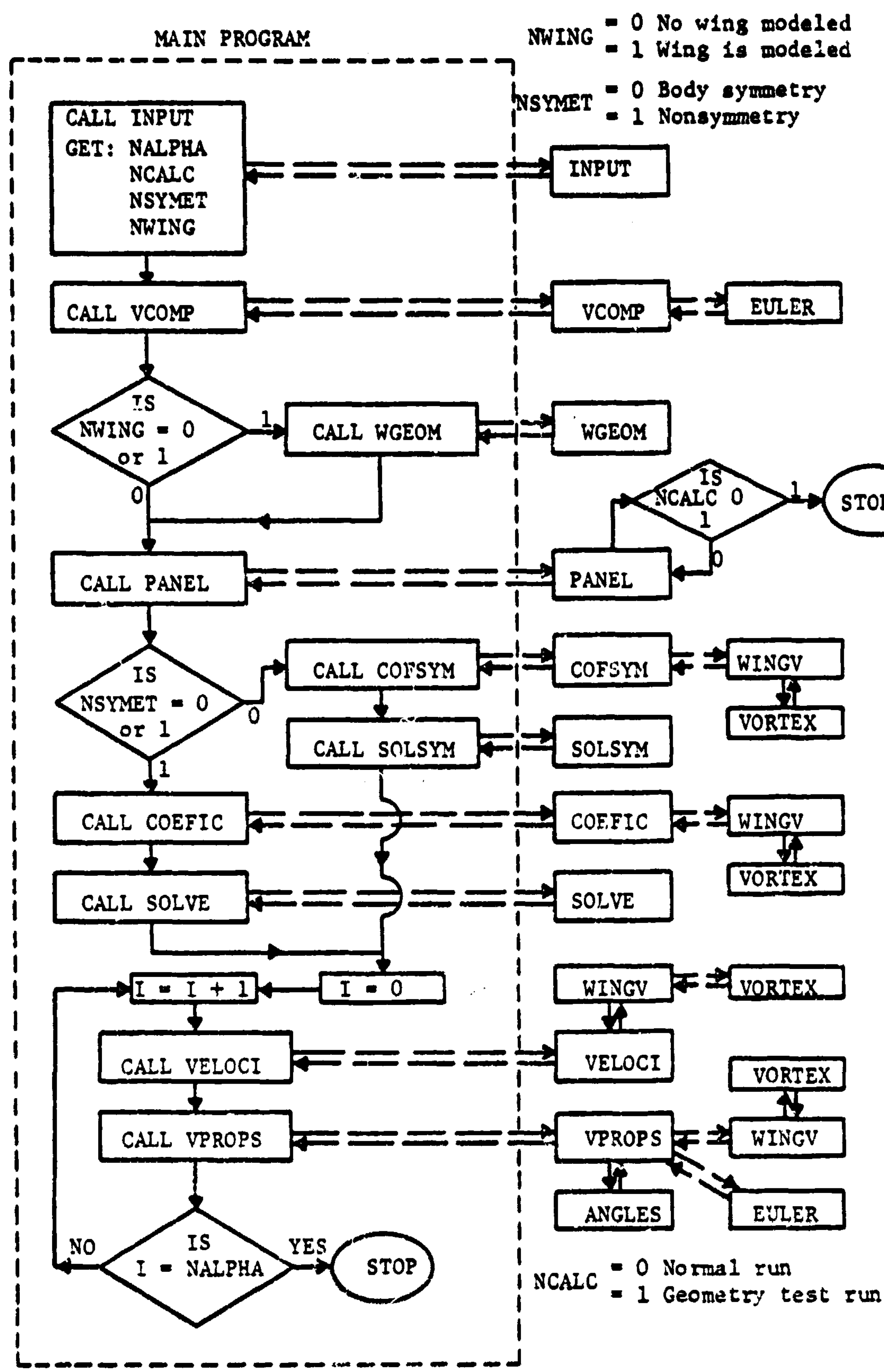


Figure 2.1 Organizational structure of 3-D potential flow program.

estimations of output record requirements and core storage requirements for the given program dimensions. Execution time is discussed, and instructions are given for creating an object program deck to enable the program to execute faster. Lastly, a section gives instructions for operating the program, including all required job control cards needed to run the program on the Penn State computer facility.

2.3 Program Dimension Size Controlling Variable Descriptions

The user has control of the program dimensions by specifying values of the following four variables on four cards in the main program. The main purpose of these is to permit execution time dimensioning of certain arrays in the subroutines:

1. NSECMA Upper limit on the number of body cross section descriptions which may be input to define the body.
   Rule: NSECMA must be integer value of 3 or more.

2. NIPMAX Upper limit on the number of periphery points allowed around a complete cross section. (First and last point are the same but count as two points.)
   Rule: NIPMAX must be an odd integer with a value of 5 or more.

3. MAXALP Maximum number of body orientations (pairs of body angle of attack and sideslip) which may be input in one run of the program.
   Rule: MAXALP must be an integer value of 1 or more.

4. MAXINF Maximum number of body panels over the entire configuration which may be specified as being inlet or outlet panels.
   Rule: MAXINF must be an even integer value of 2 or more.

If the user breaks any of the rules above, the program will detect them and stop. Refer to section 2.8 for instructions on changing these values, and refer to the listing in Appendix B as an example.

Based on the user specified size control variables above, three other variables are computed by the program and are also used for execution time dimensioning of certain arrays in subroutines. However, the user must also manually calculate the values as he must supply the numbers in the main program DIMENSION statement (section 2.5) and in the COMMON statements (section 2.6). The three additional variables are:

5\. LLL — LLL = (NSECMA - 1)*(NIPMAX - 1). LLL is the maximum possible number of body panels generated. This amount can occur only if NSECMA cross sections have been input with NIPMAX points on all sections. Also the paneling input would have to be completely efficient, that is, no repeated descriptions of a cross sections and no triangular panel input. Generally, with complex bodies, cross sections are often repeated with two descriptions and section points are repeated several times to create triangular panels. So LLL panels will not usually be generated, but could be.
<u>Note</u>: LLL will always be an <u>even</u> integer of size 8 or more.

6\. MD — MD = (LLL + MAXALP). MD is the maximum number of terms (right and left side coefficients) in each equation for panel source strengths. MD also is the length of logical records which must be specified in auxiliary file 9 (see sections 2.7 and 2.9).

7\. LLLHAF — LLLHAF = (LLL/2). It is the maximum possible number of panels on the left half of a symmetrically input body.

See section 2.8 for the procedures involving variables LLL, MD, and LLLHAF.

2.4 $DPSI_{min}$ and $NRAD_{max}$--Definitions

Two input data values control the number and distribution of flow survey points in the propeller plane. The minimum value of one and maximum value of the other control the size of auxiliary file 50 and the amount of output records:

1. DPSI Azimuthal angular increment between radial spokes of survey points on the propeller plane. $DPSI_{min}$ is the smallest increment allowed and must be more than 0.0 but not more than 360. A $DPSI_{min}$ value of 1.0 is a good choice and is used in the program version in Appendix B.

2. NRAD Number of survey points positioned radially from hub to tip at each azimuth increment on the propeller plane. It must be a positive integer but not more than $NRAD_{max}$. An $NRAD_{max}$ value of 51 is a good choice and has been used in the program of Appendix B.

The choices of $DPSI_{min}$ and $NRAD_{max}$ determine the maximum number of survey points generated. Thus they affect the number of output records produced. More importantly, they determine the number of records which must be provided in auxiliary file 50 (see sections 2.7 and 2.9).

The user must know what these values are in the program version being used so he can properly allocate file 50 storage. Instructions for changing $DPSI_{min}$ and $NRAD_{max}$ values in the program are found in section 2.8.

2.5 Description of Array DIMENSION Statement in Main Program

The following array names are found in a DIMENSION statement at the beginning of the main program. Refer to Appendix B. The dimensions of these arrays must be changed by the user in this statement when the program dimension size is altered (see section 2.8, step 3). These arrays are the only ones which are automatically dimensioned within the subroutines during execution. The DIMENSION statement and arrays are given below (without subscripts):

```
DIMENSION A, SIGMA, NCOUNT, NFLAG, PX, PY, PZ, STOX1, STOY1,
STOZ1, STOX2, STOY2, STOZ2, ANVX, ANVY, ANVZ, S, XC, YC, ZC,
BIG, VARIAB, SUM, SIGSAV
```

2.6 COMMON Block Descriptions

A total of eight different COMMON blocks are used throughout the program. Some subroutines use most of the COMMON statements and others use but a few. In most cases these COMMON statements contain both subscripted (array) and unsubscripted variables. The array variable dimensions in the COMMON statements must be changed by the user when the program dimension sizes have been changed. Dimension changes must be made to every COMMON statement card found in the program. The procedure for making the changes is found in section 2.8, step 4.

Table 2.1 describes each COMMON block name and indicates the type of data stored and the locations at which the COMMON block cards must be present.

2.7 Auxiliary File Descriptions

File 5 is a formatted card data input file. File 6 is a formatted output file written on a standard 132 character-per-line printer. File 7 refers to a formatted data output file punched on standard 80 column computer cards. These three file numbers are the default values for the computer facility at the Pennsylvania State University.

Additionally, two auxiliary scratch files are needed by the program. They are files 9 and 50 and may be stored on magnetic tape or disk. These files are described below:

File 9: This is a sequential scratch file used only to store the left and right side coefficients of the system of linear equations for the unknown source strengths. Each logical record of this file contains the coefficients of one equation. The

Table 2.1 COMMON block descriptions

COMMON BLOCK NAME	DATA STORED	SUBROUTINES IN WHICH THE COMMON STATEMENT APPEARS
/INLET/	INLET OR OUTLET PANEL INDICES AND THROUGH FLOW RATIOS	INPUT, COEFIC, COFSYM, VELOCI, and VPROPS
/INPUTS/	BODY ORIENTATIONS AND FREE STREAM VELOCITY COMPONENTS	MAIN PROGRAM, INPUT, VCOMP, COEFIC, COFSYM, SOLVE, SOLSYM, VELOCI, VPROPS, WGEOM, and VORTEX
/OPTION/	PROGRAM OPTION CONTROL VARIABLES	INPUT, PANEL, and VPROPS
/PROP/	PROPELLER PLANE GEOMETRY	INPUT and VPROPS
/SYMTRE/	SYMMETRIC CONFIGURATION PANEL BOOKKEEPING ARRAYS	MAIN PROGRAM, INPUT, PANEL COFSYM, and SOLSYM
/TITLE/	INPUT DATA HEADING	INPUT and VPROPS
/WING1/	INPUT WING GEOMETRY	MAIN PROGRAM, INPUT, COEFIC, COFSYM, WGEOM, WINGV, VELOCI, VPROPS, and VORTEX
/WING2/	CALCULATED WING VORTEX GEOMETRY	WGEOM and WINGV

records save the equations in sequence starting with the first equation or row of the system.

This file is written during execution of subroutine COEFIC if the symmetric body input option is not used. Then one record (equation) is written for each body panel. If the symmetric body input option (see section 3) is used, the file is written during execution of subroutine COFSYM. In this case, a record (equation) is written only for the panels on the left side of the symmetric body.

The file is read during the iterative solution for panel source strengths during execution of either subroutine SOLVE or SOLSYM.

On the Penn State computer facility this file is stored on an IBM 3330 Disk Pack.

File 50: This is also a sequential scratch file which temporarily saves propeller plane flow field prediction output. This allows the output to be sequentially printed on propeller plane output table part 2 (see section 4). The file is written and read only during execution of subroutine VPROPS.

Each logical record of file 50 contains flow output information for one survey point in the propeller plane. Each record of the file corresponds to one line of the printed output table. Data written on each record consists of nine values including radial and azimuthal location of the point, three velocity components and four flow angles.

On the Penn State computer facility, file 50 is stored on an IBM 3330 Disk Pack.

The details of allocating file 9 and file 50 storage space is given in section 2.9. Also, that section explains how to write the job control cards for these files for use on the Penn State facility.

2.8 User Procedure for Altering Program Dimension Size

If the user wishes to alter the dimensions of the program from those of the version listed in Appendix B, the following steps must be performed. In what follows refer to sections 2.3 to 2.6 for the definitions of dimensioning variables, dimensioned arrays and COMMON blocks. Refer to the program listing in Appendix B when a specific card number is mentioned in the steps. Here are the nine steps involved in modifying the program dimensions:

Step 1. Choose the desired values of dimensioning variables NSECMA, NIPMAX, MAXALP, and MAXINF. Ensure they conform to the rules given for them in section 2.3.

Step 2. Before changing the program cards, check the new core storage requirements of the program corresponding to the values chosen in Step 1. Use the procedure outlined in section 2.11 for this. If it is found the program will require more storage than is available, it will be necessary to repeat Step 1 with new values until the program core storage required is acceptable. Then perform the modifications given in the remaining steps 3 to 9.

Step 3. Physically replace cards MAN 5850, MAN 5900, MAN 5950, and MAN 6000 of the main program with cards having the new values of NSECMA, NIPMAX, MAXALP, and MAXINF.

Step 4. Calculate the values of the other three sizing variables LLL, MD, and LLLHAF using the values of Step 1 in the following formulas:

a. LLL = (NSECMA - 1)*(NIPMAX - 1)

b. MD = LLL + MAXALP

c. LLLHAF = (LLL/2)

These values are automatically generated by the program, but the user must know the values so the DIMENSION statement in the main program can be modified in the next step.

Step 5. **Modify the DIMENSION statement in the main program by physically changing cards MAN 5200 to MAN 5350. Change the array dimensions to values compatible with the chosen dimension sizes of Steps 1 and 4. To do this, use the following pattern DIMENSION statement which shows the array dimensions arbitrarily in terms of the dimensioning variables. Repunch cards MAN 5200 to MAN 5350 using the pattern below. (Insert the actual number values of the array subscripts, however. Punch the characters in the columns indicated. Also punch the card identifier numbers in columns 73 to 80.) (Note: only four cards should be needed for this statement.)**

Card Column:

789 72

```
DIMENSION A(MD),SIGMA(LLL,MAXALP),NCOUNT(NSECMA),
NFLAG(NSECMA),PX(NSECMA,NIPMAX),PY(NSECMA,NIPMAX),
PZ(NSECMA,NIPMAX),STOX1(NIPMAX),STOY1(NIPMAX),STOZ1(NIPMAX),
STOX2(NIPMAX),STOY2(NIPMAX),STOZ2(NIPMAX),ANVX(LLL),
ANVY(LLL),ANVZ(LLL),S(LLL),XC(LLL),YC(LLL),ZC(LLL),
BIG(MAXALP),VARIAB(MAXALP),SUM(MAXALP),SIGSAV(LLLHAF)
```

Step 6. Five of the COMMON blocks contain subscripted array variables which may be changed. Modify the dimensions of these COMMON statement arrays affected by the changed dimension size variables of Steps 1 and 4. Every affected COMMON card in the main program and subroutines must be changed by using the new numerical values of the dimension size variables. The COMMON blocks which must be changed are given below and the cards to be changed are also given. (Note: where the dimension size variable names appear in parentheses, they are to be replaced with the actual numerical value). Also the COMMON statements must be punched with the variables <u>exactly</u> in the sequence shown:

a. COMMON /INLET/ FLRATO(MAXINF),INDEX(MAXINF),NINFLO

-card INP25450 in subroutine INPUT
-card COF 1300 in subroutine COEFIC
-card CSY 900 in subroutine COFSYM
-card VEL 1150 in subroutine VELOCI
-card VPR 2450 in subroutine VPROPS

b. COMMON /INPUTS/ ALPHA(MAXALP),BETA(MAXALP), VX(MAXALP),VY(MAXALP),VZ(MAXALP),V,NALPHA

-card MAN 5400 in main program
-card INP25200 in subroutine INPUT
-card VCM 1100 in subroutine VCOMP
-card COF 1250 in subroutine COEFIC
-card SOL 1200 in subroutine SOLVE
-card CSY 850 in subroutine COFSYM
-card SOS 1350 in subroutine SOLSYM
-card VEL 1100 in subroutine VELOCI
-card VPR 2200 in subroutine VPROPS
-card WGE 2400 in subroutine WGEOM
-card VOR 2200 in subroutine VORTEX

c. COMMON /SYMTRE/ NPNSYM(LLL),INSOLV(LLLHAF),NSYMET

-card MAN 5450 in main program
-card INP25400 in subroutine INPUT
-card PAN 1300 in subroutine PANEL
-card CSY 950 in subroutine COFSYM
-card SOS 1400 in subroutine SCLSYM

d. COMMON /WING1/ CL(MAXALP),CHORD,DIHED,SPAN,SWEEP, XQR,YQR,ZQR,NWING

-card MAN 5500 in main program
-card INP25500 in subroutine INPUT
-card COF 1350 in subroutine COEFIC
-card CSY 1000 in subroutine COFSYM
-card WGE 2450 in subroutine WGEOM
-card VEL 1200 in subroutine VELOCI
-card VPR 2400 in subroutine VPROPS
-card WGV 2800 in subroutine WINGV
-card VOR 2250 in subroutine VORTEX

e. COMMON /WING2/ GAMMA(MAXALP),XTRALL(MAXALP),YTRALL (MAXALP),ZTRALL(MAXALP),XTRALR(MAXALP),YTRALR(MAXALP), ZTRALR(MAXALP),XBTIPL,YBTIPL,ZBTIPL,XBTIPR,YBTIPR,ZBTIPR

-card WGE 2500 and WGE 2550 in subroutine WGEOM
-card WGV 2850 and WGV 2900 in subroutine WINGV

Step 7. Modify the following cards in the main program and subroutine INPUT as explained below:

<u>Main Program</u>

a. card MAN 4400: insert new values of NSECMA,NIPMAX, MAXALP, and MAXINF

b. card MAN 4550: insert new value of LLL after the words "UP TO LLL ="

c. card MAN 4700: insert value of MAXALP after the words "UP TO"

d. card MAN 4800: insert value of MAXINF after the words "UP TO"

<u>Subroutine INPUT</u>

e. card INP 1850: "C . . . CONTAINS 3 TO N CARDS . . ." insert value of N where N = (MAXALP + 2)

f. card INP 2300: "C . . . (VALUE FROM 1 to N . . ." insert value of N, where N = MAXALP

g. card INP 4050: "C . . . CONTAINS 11 TO N CARDS . . ." insert value of N, where N = (NIPMAX * NSECMA) + NSECMA + 2

h. card INP 5650: "C . . . MAX OF N) . . ." insert value of N, where N = NSECMA

i. card INP 6650: "C . . . N). IF . . ." insert value of N, where N = (NIPMAX + 1)/2

j. card INP 6900: "C . . . MAXIMUM OF N . . ." insert value of N, where N = NIPMAX

k. card INP11200: "C . . . CONTAINS 1 TO N CARDS) . . ." insert value of N, where N = (MAXINF + 1)

l. card INP11900: "C . . . VALUE N . . ." insert value of N, where N = (MAXINF/2)

m. card INP12000: "C . . . VALUE OF N IF . . ." insert value of N, where N = MAXINF

n. card INP14050: "C . . . N CARDS) . . ." insert value of N, where N = MAXALP + 2

Step 8. If desired, change either or both the values of $DPSI_{min}$ and $NRAD_{max}$. Ensure they obey the rules for these values given in section 2.4. The recommended values of $DPSI_{min}$ = 1.0 degree and $NRAD_{max}$ = 51 have been used in the program listed in Appendix B. If these values are changed, perform the following:

a. If $DPSI_{min}$ is changed, modify the following card found in subroutine INPUT

-card INP20300: "C . . . MUST BE AT LEAST N AND . . ."
insert value of N, where
N = $DPSI_{min}$ value

b. If $NRAD_{max}$ is changed, modify the two following cards found in subroutine INPUT:

-card INP21100: "C . . . FROM 1 TO N . . ."
insert value of N, where
N = $NRAD_{max}$ value

-card INP36450: ". . .20 IF (NRAD.GT.N) GO TO 76"
insert value of N, where
N = $NRAD_{max}$ value

Step 9. The last step in modifying program dimensions is this. Calculate the amount of auxiliary storage needed for files 9 and 50. Write the job control language cards to give the new storage allocation. The rules for this are given in section 2.9.

The version of the program given in Appendix B has been dimensioned using the following dimension size variable values:

NSECMA = 60
NIPMAX = 45
MAXALP = 6
MAXINF = 500
LLL = 2596
MD = 2602
LLLHAF = 1298

and

$NRAD_{max}$ = 51

$DPSI_{min}$ = 1.0

2.9 Determining Auxiliary File Storage Allocation Based on Program Dimension Size

Storage for auxiliary file 9 and file 50 must be allocated on either magnetic tape or disk, and job control cards must be made to perform the allocation. The maximum dimension of the program version being used determines how many records must be available on the two files. This was explained in section 2.7.

In the following parts 1 and 2 below is given the formulas for calculating record and disk pack requirements for files 9 and 50, respectively. Also given is the job control language needed for establishing these files on the Penn State computer facility.

2.9.1 File 9 Storage Allocation Formulas and JCL

File 9 is a sequential file requiring variably spanned physical records[6]. Enough tape or disk space must be provided to store up to LLL (section 2.3) logical records, each containing MD (section 2.3) single precision numbers of 4 bytes each.

On the Penn State University IBM 370 computer system, file 9 is stored on an IBM 3330 Disk Pack[6]. Since this file is relatively large, the disk storage is allotted by requesting numbers of cylinders of space, CYL.

For this disk pack, the user should specify the following parameters in the job control cards when allocating space for file 9:

RECFM = VS (i.e., variable spanned records)
BLKSIZE = 3120 (bytes)
LRECL = (BLKSIZE - 4) = 3116 (bytes)
CYL = number of disk cylinders requested (calculated below)

Number of cylinders, CYL, required is a function of the size variables, LLL and MD. Use the formula 1 or 2 below (whichever applies):

1. If [(4 * MD) > (BLKSIZE - 8)] use:

$$CYL = \left[\frac{LLL * \left[\left(\frac{MD * 4}{BLKSIZE - 8} \right)_{\text{rounded to higher integer}} \right]}{19 * \left[\frac{13165}{BLKSIZE + 135} \right]_{\text{rounded to lower integer}}} \right]_{\text{rounded to higher integer}}$$

-or-

2. If [(4 * MD) $\leq$ (BLKSIZE - 8)] use:

$$CYL = \left[\frac{LLL}{19 * \left(\frac{13165}{BLKSIZE + 135} \right)_{\text{rounded to lower integer}}} \right]_{\text{rounded to higher integer}}$$

Note that CYL is the maximum number of cylinders which would be needed if LLL body panels were generated. So CYL must be available, but usually, not all of the space is used, since most configurations seldom generate the full amount of LLL panels.

The following example specifies file 9 storage and job control cards for the program version listed in Appendix B. This example applied to the Penn State University computer facility. This program has:

LLL = 2596 and MD = 2602

The job control parameters are:

```
BLKSIZE = 3120
LRECL   = 3116
CYL     =  137  calculated using formula 1 above.
```

Two job control cards punched exactly as shown below, both starting in column 1, will establish 137 cylinders of space for file 9:

```
//DATA.FT09F001 DD UNIT=SYSDA,SPACE=(CYL,(123,1),RLSE),
// DCB=(RECFM=VS,LRECL=3116,BLKSIZE=3120)
```

Here the cylinders have been requested using a primary allocation of 123 cylinders with a secondary allocation of one cylinder (repeatable up to 14 times) which is made when the space is needed. This gives 137 cylinders available. Reference 6 contains details on the disk pack and job control language for the Penn State computer system.

2.9.2 File 50 Storage Allocation Formulas and JCL

File 50 is a sequential file requiring variably spanned physical records[6]. Enough tape or disk space must be provided to store up to R logical records, where R is given by:

$$R = NRAD_{max} * \left(\frac{360}{DPSI_{min}}\right) \text{ rounded to lower integer}$$

$NRAD_{max}$ and $DPSI_{min}$ are defined in section 2.4.

Each record contains nine single precision numbers of 4 bytes each.

On the Penn State University IBM 370 computer system, file 50 is stored on an IBM 3330 Disk Pack[6]. The disk storage for this file should be allotted by requesting numbers of cylinders of space, CYL.

For this disk pack, the user should specify the following parameters in the job control cards when allocating space for file 50:

RECFM = VS (i.e., variable spanned records)
BLKSIZE = 44 (bytes)
LRECL = (BLKSIZE - 4) = 40 (bytes)
CYL = number of disk cylinders requested (calculated below.

Number of cylinders, CYL, required is a function of the propeller plane input point limits, $NRAD_{max}$ and $DPSI_{min}$, of the program version being used. Use the following formula to calculate CYL for file 50:

$$\text{CYL} = \left[\frac{\text{NRAD}_{max} * \left(\frac{360}{\text{DPSI}_{min}}\right) \text{rounded to lower integer}}{19 * \left(\frac{13165}{\text{BLKSIZE} + 135}\right) \text{rounded to lower integer}} \right] \text{rounded to higher integer}$$

The following example specifies file 50 storage allocation and job control cards for the program version listed in Appendix B. This example applies to the Penn State University computer facility. This program has the following propeller plane survey point limits:

$$\text{DPSI}_{min} = 1.0 \quad \text{and} \quad \text{NRAD}_{max} = 51$$

The job control parameters are:

BLKSIZE = 44
LRECL = 40
CYL = 14 calculated using the above formula.

Two job control cards punched exactly as shown below, both starting in column 1, will establish 14 cylinders of space for file 50:

```
//DATA,FT50F001 DD UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE),
// DCB=(RECFM=VS,LRECL=40,BLKSIZE=44)
```

Here the 14 cylinders have been requested using a primary allocation of one cylinder with a secondary allocation of one cylinder (repeatable up to 14 times) which is made when the space is needed. This gives 15 cylinders available. Reference 6 contains details on the disk pack and job control language for the Penn State computer system.

2.10 Output Record Considerations Based on Program Dimension Size

For some computer installations such as the one at Penn State, there is a limit on the number of output records which may be produced during a single run. At Penn State, this output limit is 30,000 records. If more records will be produced, special handling must be made to have the records written to a tape so it can be printed later in smaller portions.

The maximum number of output records from this computer program is a function of the dimension size variables (section 2.3) and propeller plane input limits (section 2.4). The four primary quantities affecting length of output are LLL, MAXALP, $NRAD_{max}$ and $DPSI_{min}$.

An approximate formula for estimating the maximum number of output records produced is given below. The formula is a function of the dimension size variables defined earlier in sections 2.2 and 2.4. Also used are certain input variables defined in section 3. Note the formula assumes printed output is written on a standard 132 character-per-line printer, and the punched output is on 80 column cards. Refer also to section 4:

$$R_{tot} \cong R_1 + R_2 + R_3 + R_3 + R_5 + MAXALP * (R_6 + R_7 + R_8 + R_9)$$

where R_{tot} is the total maximum possible number of output records produced by a given size program version.

The R_i terms in the formula are each given by formulas below:

$$R_1 \cong \begin{bmatrix} 0, \text{ if program listing is not printed} \\ 3991, \text{ if program listing is printed} \end{bmatrix} \quad : \text{(source listing)}$$

$$R_2 \cong \begin{bmatrix} MAXALP + 1, \text{ if } NWING = 1 \\ 0, \text{ if } NWING = 0 \end{bmatrix} + \begin{bmatrix} 2, \text{ if } NPOINT = 1 \\ 0, \text{ if } NPOINT = 0 \end{bmatrix} + MAXALP$$
$$+ MAXINF + NSECMA * (NIPMAX + 1) + 27:$$

(This is list of input.)

$$R_3 = \begin{bmatrix} 0, \text{ if } NWING = 0 \\ (MAXALP + 26), \text{ if } NWING = 1 \end{bmatrix} \quad : \text{(This is wing geometry table.)}$$

$$R_4 = \begin{bmatrix} 1, \text{ if } NLIST = 1 \text{ and } NCALC = 0 \\ (5 * LLL) + 16, \text{ if } NLIST = 0 \text{ or } NCALC = 1 \end{bmatrix} \quad : \text{(This is body geometry table.)}$$

$$R_5 = \begin{bmatrix} 0, \text{ if } NCALC = 1 \\ (3 * ITMAX) + 4, \text{ if } NCALC = 0 \end{bmatrix} \quad : \text{(This is solution iteration table.)}$$

$$R_6 = \begin{bmatrix} 0, \text{ if } NCALC = 1 \\ (LLL + 14), \text{ if } NCALC = 0 \end{bmatrix} \quad : \text{(Surface velocity table.)}$$

$$R_7 = \begin{cases} 0, \text{ if NCALC} = 1 \\ \text{MAXALP, if NPOINT} = 0 \text{ and NCALC} = 0 \\ [(360/\text{DPSI}_{min})_{\substack{\text{rounded to} \\ \text{lower integer}}} * \text{NRAD}_{max}] + 29, \text{ if NPOINT} = 1 \text{ and NCALC} = 0 \end{cases} :$$

(This is propeller plane output table part 1.)

$$R_8 = \begin{cases} 0, \text{ if NCALC} = 1 \\ \text{MAXALP, if NPOINT} = 0 \text{ and NCALC} = 0 \\ [(360/\text{DPSI}_{min})_{\substack{\text{rounded to} \\ \text{lower integer}}} * \text{NRAD}_{max}] + 34, \text{ if NPOINT} = 1 \text{ and NCALC} = 0 \end{cases} :$$

(This is propeller plane output table part 2.)

$$R_9 = \begin{cases} 0, \text{ if NPUNCH} = 0 \\ [(360/\text{DPSI}_{min})_{\substack{\text{rounded to} \\ \text{lower integer}}} * \text{NRAD}_{max}] + 3, \text{ if NPUNCH} = 1 \end{cases} :$$

(This is punched output.)

It must be stated that R_{tot} is the maximum possible number of output records which would be produced by the program version, only if all input options were used and only if the configuration used LLL panels and only if MAXALP orientations were input.

Normally the configuration will not have LLL panels but will have NP panels instead. Also the number of orientations used is often less than MAXALP and will be NALPHA instead.

Thus to get a better estimate of the number of output records which will be produced when running a specific configuration, do the following.

Use the R_{tot} expression given above, but make the following changes:

-replace LLL value by the value of NP for the body (best estimate should be used here.)

-replace MAXALP value by value of NALPHA (section 3)

-replace MAXINF value by value of NINFLO (section 3)

-in the R_2 formula, replace the following:

replace [NSECMA * (NIPMAX + 1)] by a value equal to the number of cards in set 3 of the input card deck (refer to section 3).

-replace $NRAD_{max}$ by value of NRAD

-replace $DPSI_{min}$ by value of DPSI

As an example of the preceding descriptions, use the program listed in Appendix B. This program has dimensions:

LLL = 2596, MAXALP = 6, MAXINF = 500, NSECMA = 60, NIPMAX = 45, $NRAD_{max}$ = 51, and $DPSI_{min}$ = 1.0

Using these values in the R_{tot} formula gives the maximum possible number of output records by this program version:

$$R_{tot} = 363,782 \text{ records}$$

This far exceeds the 30,000 record limit imposed at the Penn State computer facility. However, a typical configuration run with this version of the program might consist of:

NP = 968 panels
NALPHA = 6
NINFLO = 88
NRAD = 21
DPSI = 15.0
ITMAX = 20
NSYMET = 0
NCALC = 0
NWING = 1
NPOINT = 0
NPUNCH = 1
with 877 cards in the set 3 of the input card deck.

Using the formula for R_{tot} with the above numerical substitutions, gives the following:

R_{tot} = 25,266 output records actually produced.

This specific configuration could be run at Penn State since it meets the 30,000 record limit requirement.

Although the program dimensions are such that it fits in the computer core storage and executes properly, the output records may be excessive if the configuration uses most of the available program storage. As a result, it may be necessary to make tradeoffs on the input to the program. These tradeoffs might involve reducing the number of orientations submitted at one time, or attempting to reduce the number of body panels.

2.11 Core Storage Requirements Based on Program Dimension Size

Core storage requirements for the source statements is constant. However, the core storage needed to store arrays and COMMON blocks is a function of the dimension size variable values for the program version being used. The program is run in single precision. For the IBM 370 computer, this means each number in the arrays and COMMON statements occupies 4 bytes of storage.

The maximum storage available to users at Penn State is 560 K bytes core storage (K equals 1024). However, not all of the 560 K bytes are available directly for program storage. Some of the storage is taken by the computer for compiler operation and is not available for program array storage.

An estimate of core storage required by the computer program on the Penn State computer facility is obtained by the following formula (refer to definitions in section 2.3):

$$S_{tot} = (S_{obj} + S_{arrays}) \quad : \text{ K bytes, where K} = 1024$$

where S_{tot} is total core storage needed by the program

S_{obj} is core storage needed to store the program source statements

S_{arrays} is core storage needed to store arrays and other variables in COMMON blocks.

S_{obj} is a constant for this program:

$$S_{obj} = \frac{118656}{1024} \text{ K bytes}$$

S_{arrays} is a function of the program dimension size variables:

$$S_{arrays} = \frac{4}{1024} * [(\text{MAXALP} + 10) * \text{LLL} + (\text{NIPMAX} * \text{NSECMA} * 3) + (17 * \text{MAXALP}) + (6 * \text{NIPMAX}) + (2 * \text{NSECMA}) + (2 * \text{MAXINF}) + 281] \text{ K bytes}$$

Estimation of S_{tot} gives an approximate idea of how much core storage to request. However, the total storage requested must also provide enough space to satisfy S_{tot} as well as storage required by the computer compiler.

For example, the program version listed in Appendix B is dimensioned as follows:

LLL = 2596, MAXALP = 6, NSECMA = 60, NIPMAX = 45, and MAXINF = 500

From the formula above compute:

$$S_{tot} = 316.69 \text{ K bytes}$$

It has been found that S_{tot} plus the compiler storage requirement does not exceed 560 K bytes. Thus the program version of Appendix B can be run in the 560 K storage region (only as a category 5 run) on the Penn State computer facility.

For a second example, the same program but dimensioned to a smaller size (listing not included in this manual) has been written using:

LLL = 2450, MAXALP = 1, NSECMA = 50, NIPMAX = 51, and MAXINF = 500

From the storage formula compute:

$$S_{tot} = 257.69 \text{ K bytes}$$

It has been found that with this smaller size version of the program, the source deck must be compiled into an object deck using the Fortran H compiler, Optimization level 2 feature on the Penn State system and needs 560 K for this. However, once this object deck has been made, it can be executed using only 280 K storage. The 280 K storage required means this version of the program can be run in higher priority job categories at Penn State and will have faster turn-around time. Refer to sections 2.13 and 2.14 on compiling and using an object deck for the program.

2.12 Execution Time and Object Program Considerations

The iterative solutions of hundreds of simultaneous equations by the computer program is the process requiring most of the total execution time. The required execution time increases rapidly as the number of body panels (equations) increases. System time used in reading equations from the auxiliary file storage device increases as the number of equations increases and as the number of solution iterations increases. Also more time is needed as the number of body orientation cases increases, and the number of survey points in the propeller plane increases.

No quantitative relationships have been made between run time and the above program factors. However, Table 2.2 lists several cases run on the Penn State IBM 370 computer, and these give some insight into the run time requirements as a function of number of equations, number of iterations, number of orientations, and number of survey points. The time limit imposed on runs made on the Penn State facility is 2000 seconds. This is the maximum amount of time ever allowed for any run. The last run on Table 2.2 required more than 2000 seconds and failed to give the complete solution.

Some obvious points can be made regarding run time. When possible, a configuration should be submitted using the symmetric input option (see sections 1 and 3). This will halve the number of equations which must be solved simultaneously and will greatly reduce run time. It is likely that a configuration, run on the Appendix B size program, which uses most available program storage (i.e. has nearly LLL equations, MAXALP orientations, and maximum number of propeller plane survey points) will require excessive run time on the Penn State computer and will not be completed. However, no configuration this large has been run and the actual time requirement in this situation is not certain. To keep run time within limits, it may be necessary to compromise on runs with configurations having very many panels. For example, reducing the number of orientations submitted and the number of propeller plane survey points may be necessary when the configuration has many panels.

From experience, due to the lengthy execution times required on typical configurations having 1000 or more panels, it is highly recommended that the source program deck (i.e. as it is listed in Appendix B) be compiled using the Fortran H compiler, Optimization level = 2 feature available on the Penn State computer system. The resulting object program deck may be punched on cards or placed on

Table 2.2 Comparison of computing time requirements of several run cases. (IBM 370 computer and IBM 3330 disk pack used.)

CONFIGURATION (INPUT OPTION)	TOTAL PANELS (EQUATIONS)	NUMBER OF BODY ORIENTATIONS	NUMBER OF PROPELLER PLANE SURVEY POINTS	ITERATIONS TO SOLUTION ERROR<10**-4	TYPE OF PROGRAM DECK USED	CPU TIME USED - SECONDS	TOTAL SYSTEM TIME USED - SECONDS. (2000 AVAILABLE)
SPHERE (SYMMETRIC)	120 (60)	1	372	8	OBJECT (H, OPT = 2)*	7	53
PIPER CHEROKEE PA-28-180 (SYMMETRIC)	968 (484)	6	504	18	OBJECT (H, OPT = 2)*	465	655
PIPER CHEROKEE PA-28-180 (NON-SYMMETRIC)	846 (846)	3	168	19	OBJECT (H, OPT = 2)*	695	1171
WING - FUSELAGE, LEFT NACELLE OF REFERENCE 7 (NON-SYMMETRIC)	424 (424)	6	168	10	OBJECT (H, OPT = 2)*	116	215
PIPER CHEROKEE PA-28-180 (NON-SYMMETRIC)	786 (786)	1	528	13	SOURCE	1210	1406
PIPER CHEROKEE PA-28-180 (NON-SYMMETRIC)	1000 (1000) APPROXIMATE	1	504	—	SOURCE	2000 +	2000 + (TIME LIMIT EXCEEDED)

* FORTRAN H compiled - optimized machine language deck (Penn State feature). See sections 2.13 and 2.14

magnetic storage (BAT) files. Runs are then made by submitting and executing this highly efficient object deck. The resulting decrease in run time with the object deck has been on the order of 50 percent. In some configurations, use of the object deck has enabled solutions to be obtained whereas the same configuration run using the source deck would have failed to finish within the 2000 second time limit. Refer to sections 2.13 and 2.14 for the details of compiling the object deck from the source deck and submitting runs with the object deck.

2.13 Object Program--Considerations and Procedures for Compiling It

The procedures and discussion of this section pertain specifically to the Penn State University computer facility which uses an IBM 370 computer.

It is highly recommended that the FORTRAN source program (as it appears in Appendix B) be first compiled using the FORTRAN H compiler, under Optimization level 2. The output is a very efficient machine language object program deck. This object deck output should be punched on cards or written to the magnetic disk files (BAT files on the Penn State facility).

When submitting data and running the program, the object program deck is submitted directly with data. Time is not lost in compilation since the object program has been compiled before submission. Most importantly, the H, opt = 2 object program is efficient and executes much faster than occurs when the original source deck is submitted for compilation and execution. As pointed out in section 2.12, experience has shown that as much as 50 percent execution time savings is attained by using the object deck directly with the data. Time saved in using the object deck may allow large jobs to complete within time limits whereas the same case performed by submitting the source program for compilation and execution would have failed to complete within the time limits.

The following two subsections give specific instructions for compiling an object deck and storing it on cards and disk files, respectively.

2.13.1 Instructions for Compiling the Object Deck and Punching it on Cards

To compile the source program using FORTRAN H, Optimization level 2, and punching the object deck on cards, do the following. Submit the following cards exactly as shown (all cards below start in column 1 unless otherwise noted):

```
// job card (use  category 5, specify T=2000, R=30000, S=280 K)
// EXEC FHC,PARM.SOURCE='DECK,OPT=2'
//SOURCE.INPUT DD *
      .
      .
      .   FORTRAN source deck version of the program
      .   (cards as they appear in Appendix B, or the BAT
      .   files containing these card images.  This is done by
/*        placing cards here of the form:
          /*INCLUDE Userid.$ source file name 1
          /*INCLUDE Userid.$ source file name 2
          etc., until all source files are given)
```

Note: Userid is the identifying number a user must have for using the Penn State BAT file system.

2.13.2 Instructions for Compiling the Object Deck and Writing it on BAT Files

To compile the source program using FORTRAN H, Optimization level = 2, and writing the object deck on BAT files, do the following. Submit the following cards exactly as shown (all cards below start in column 1 unless otherwise noted):

```
// job card (use category 5, specify T=2000, R=30000, S=560 K)
/*USERID  card for using the Penn State BAT file system
// EXEC FHC,PARM.SOURCE='DECK,OPT=2'
//SOURCE.SYSPUNCH DD UNIT=BAT,FILES=($filename 1, ...,$filename 4)
//SOURCE.INPUT DD *
       .
       .
       . FORTRAN source deck version of the program
       . (cards as they appear in Appendix B, or the BAT files con-
       . taining these card images.  This is done by placing cards
       . here of the form:  /*INCLUDE Userid.$source file name 1
       .                    /*INCLUDE Userid.$source file name 2
       .                    etc. until all source files are given)
       .
       .
/*
```

Note: $filename 1, $filename 2, $filename 3, and $filename 4 are the names of the four BAT files which sequentially store the output object program deck. These names must fit the naming conventions for BAT files. (See references 6 and 8.)

The potential flow program requires four BAT files to store its object deck. Four files are required regardless of the dimension size of the program being compiled.

Instructions for running the object program with data are given in section 2.14 of this manual.

2.14 Operating Instructions--Required Job Control Cards

The following instructions and job control cards apply specifically to the source program version listed in Appendix B or its corresponding object deck. The instructions will apply to any size version of the program if the job control cards establishing file 9 and file 50 storage are modified according to the rules of section 2.9. Also the job control cards apply specifically to the Penn State IBM 370 computer facility.

The program in Appendix B has been dimensioned as follows: (See sections 2.3 and 2.4 for definitions.)

NSECMA	=	60
NIPMAX	=	45
MAXALP	=	6
MAXINF	=	500
LLL	=	2596
MD	=	2602
LLLHAF	=	1298
$DPSI_{min}$	=	1.00
$NRAD_{max}$	=	51

Subsections 2.14.1 and 2.14.2 below give operating instructions for running the program from the source deck and object deck, respectively. These are applicable to the computer system at Penn State.

2.14.1 Running Program by Submitting Source Deck and Data

This is the method of using the program which is simpler but less efficient. The source program runs slower, and it is not the recommended method. However, this may be the only way the program can be used at computer facilities different from that at Penn State.

The job control cards given in the following allocate file 9 and file 50 disk storage corresponding to the above program dimension sizes. For using the program with different dimensions, these job control cards must be modified using the instructions in section 2.9.

Supply the following cards, in the order shown, to compile and execute the source program and process data. Punch these cards exactly as shown starting in card column 1 unless otherwise noted:

```
// job card (use category 5, specify T=2000, R=30000, S=360 K)
// EXEC FHCG
//SYSIN DD *
     .
     .
     .   source program:  (cards as they appear in Appendix B or
     .                    the BAT files containing these card
     .                    images. This is done by placing cards
     .                    here of the form:
     .                         /*INCLUDE Userid.$source file name 1
     .                         /*INCLUDE Userid.$source file name 2
     .                         etc. until all source files are given.)

//DATA.FT09F001 DD UNIT=SYSDA,SPACE=(CYL,(123,1),RLSE),
// DCB=(RECFM=VS,LRECL=3116,BLKSIZE=3120)
//DATA.FT50F001 DD UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE),
// DCB=(RECFM=VS,LRECL=40,BLKSIZE=44)
//DATA.INPUT DD *
     .
     .
     .   input data cards punched according to format of section 3
     .   of this manual. Also may have data on BAT files included
     .   here by punching the following cards here of the form:
     .        /*INCLUDE Userid.$data file name
     .   Include enough of these cards to supply all necessary
     .   data BAT files. Cards and BAT file include cards may be
     .   intermixed.
     .
```

-- This concludes the operating procedure. --

2.14.2 Running Program by Submitting Object Deck and Data

This second operating method is recommended. It is more complex, however, because the object program deck must be obtained using the methods of section 2.13. With the object deck available on cards or BAT files, jobs may be run using the procedure given in this subsection.

As in section 2.14.1, the job control cards given here establish file 9 and file 50 storage space to meet the needs of the program in Appendix B whose dimensions were listed above. These job control cards may be modified to suit a program of different dimensions. This is done by following the procedures in section 2.9.

Supply the following cards, in the order shown, to execute the object program deck and process data. Punch these cards exactly as shown starting in card column 1 unless otherwise noted:

```
// job card (use category 5, specify T = 2000, R = 30000, S = 560 K)
// EXEC FHG
//DATA.DECK DD *
        .
        .
        .  object program:  (card deck or images of cards from BAT
        .                   files as produced by section 2.13 instruc-
        .                   tions.  To submit program from BAT files,
        .                   place the following four cards here of the
        .                   form:  /*INCLUDE Userid.$object file name 1
        .                          /*INCLUDE Userid.$object file name 2
        .                          /*INCLUDE Userid.$object file name 3
        .                          /*INCLUDE Userid.$object file name 4
        .                   where the above file names are the same
        .                   as those used on the cards when following
        .                   procedures of section 2.13.2.)
        .
        .
        .
//DATA.FT09F001 DD UNIT=SYSDA,SPACE=(CYL,(123,1),RLSE),
// DCB=(RECFM=VS,LRECL=3116,BLKSIZE=3120)
//DATA.FT50F001 DD UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE),
// DCB=(RECFM=VS,LRECL=40,BLKSIZE=44)
//DATA.INPUT DD *
        .
        .
        .  input data cards punched according to format of section 3
        .  of this manual.  Also may have data on BAT files included
        .  here by punching the following cards here of the form:
        .       /*INCLUDE Userid.$data file name
        .  Include enough of these cards to supply all necessary data
        .  BAT files.  Cards and BAT file include cards may be
        .  intermixed.
        .
```

-- This concludes the operating procedure. --

This concludes the detailed descriptions of the program. In the next sections of the manual are descriptions of input data and output data organization.

SECTION 3 INPUT DATA DESCRIPTION

The input to the program consists of eight card sets, each set containing one or more cards. No card set is ever omitted. Some input options are available and are explained below and in the card descriptions:

1. Symmetric body input option: This may be used if body is symmetric about the y = 0, x-z plane. No closed separate bodies such as tip tanks or nacelles can be present. The wing, if present, must be symmetric, and no body sideslip angles may be specified. Then only the left side of the body geometry need be provided by the user in card set 3.

2. Regular body input option: If any condition in (1) above is not met then the regular input option must be used. The user must provide the complete body surface geometry in card set 3.

3. Test run option: This is used only to generate and print the body panel geometry. No flow calculations are made. Two purposes of this option are:
   a. To check for errors in the geometry input.
   b. If user is not certain which index number will be assigned to a certain panel by the program, this option allows the user to identify each panel by index number (as is required when specifying inlet/outlet panels in card set 5). In using this option, all card sets are input as instructed except no inlet/outlet panels may be specified (NINFLO = 0 in card set 5). After making the test run, the proper index numbers of inlet panels are identified from the output panel geometry and can be input in card set 5. Then the card deck may be run as a normal flow prediction run.

4. Printing and punched output options: User may select to print-out the panel geometry or not to print it. Note for a test run this geometry is automatically printed (see output description in this manual). Also, user may elect to punch or not punch certain propeller plane flow output on cards.

This card input description, below, is written in general form applicable to any dimension size version of the program which the user has available. That is, the value limits on the input variables in the card descriptions are expressed in terms of the following program dimension size variables: NSECMA, NIPMAX, MAXALP, MAXINF, LLL, MD, LLLHAF, $DPSI_{min}$, and $NRAD_{max}$. These are defined in detail in the main program and govern the dimensioning of arrays. They are explained in detail in section 2 of this manual on changing program dimensions. The definitions of the above size variables are:

NSECMA - Maximum number of body cross sections which may be input.

NIPMAX - Maximum number of periphery points allowed to be input around any complete cross section.

MAXALP - Maximum number of input body orientations allowed.

MAXINF - Maximum number of inlet/outlet panels which may be present over the entire body.

LLL - =(NSECMA-1)*(NIPMAX-1), Maximum possible number of body panels on entire body.

MD - =(LLL + MAXALP)

LLLHAF - =(LLL/2)

$DPSI_{min}$ - Minimum allowed azimuth angle spacing for points on propeller plane.

$NRAD_{max}$ - Maximum allowed number of points along each azimuth in the propeller plane.

Keep in mind that if the input is being used in the program version listed in Appendix B then the dimension variables have the following values:

NSECMA	=	60
NIPMAX	=	45
MAXALP	=	6
MAXINF	=	500
LLL	=	2596
MD	=	2602
LLLHAF	=	1298
$DPSI_{min}$	=	1.0
$NRAD_{max}$	=	51

Use any consistent family of units for the velocities and coordinates throughout the data. Dimensionless quantities are so noted. All angles are in degrees.

Integers are right justified in card column fields and are identified as integers in the description.

Floating point numbers are punched anywhere in the given column field and must include the decimal point. These type numbers are identified as floating point in the card description.

Card Set 1: Data or Run Identification (2 cards)

Card 1.1 Title Card 1

Notes	Variable	Columns	Description
(1)	SYMBOL(I), I=1,80	1 - 80	First part of the input title.

Card 1.2 Title Card 2

Notes	Variable	Columns	Description
(2)	SYMBOL(I), I=81,160	1 - 80	Second part of the input title.

Notes for Card Set 1

(1) Contains first part of a title containing any desired identifying information in letters, numbers, or symbols. Card must always be present in deck even if totally blank.

(2) Second card for continuation of the identifying information. Contains any letters, numbers, or symbols. Card must always be present in deck even if totally blank.

Card Set 2: Free Stream Velocity--Body Orientations

Set specifies the body orientations for which flow solutions are to be obtained. (Set contains 3 to MAXALP + 2 cards.)

Card 2.1 Free Stream Velocity

Notes	Variable	Columns	Description
(1)	V	1 - 10	Floating Point. Magnitude of free stream velocity. (1.0 is convenient)

Card 2.2 Number of Orientations

Notes	Variable	Column	Description
none	NALPHA	1	Integer. Number of body orientations (pairs of angle of attack and sideslip) this run. Value from 1 to MAXALP.

Cards 2.3 Body Orientations (NALPHA cards. See note (2).)

Notes	Variable	Columns	Description
(3)	ALPHA(I)	1 - 10	Floating point. Body angle of attack in degrees for orientation case I. (Value of 0 to 360., or 0 to -360.)
(4, 5)	BETA(I)	20 - 29	Floating point. Body sideslip angle in degrees for orientation case I. (Value of 0 to 360., or 0 to -360.)

Notes for Card Set 2

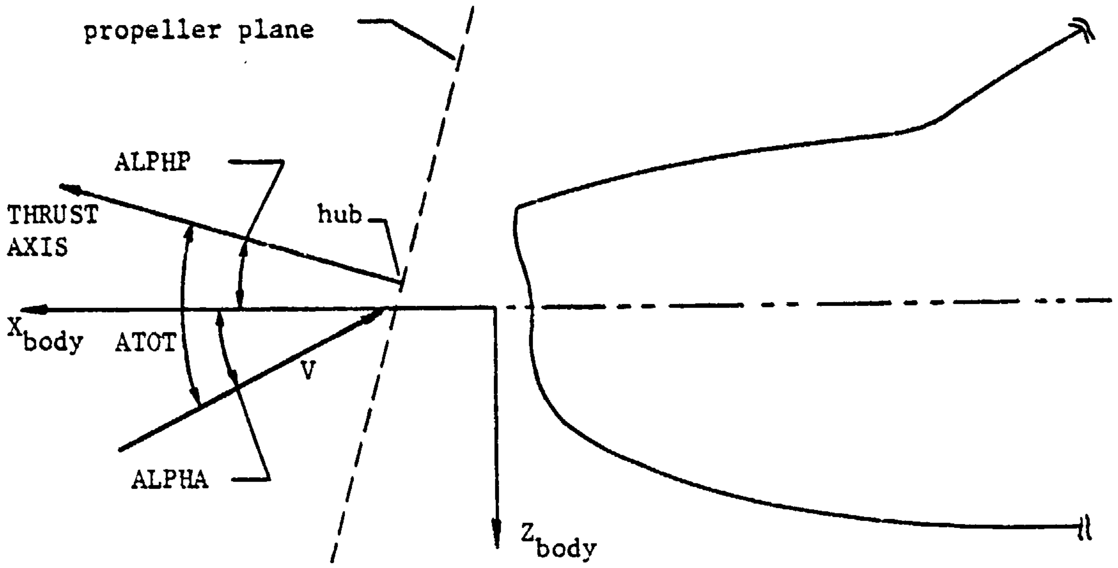
(1) All output velocity quantities are nondimensionalized with respect to V so the actual magnitude of V is not important and a value of 1.0 is recommended for convenience.

(2) Card 2.3 is repeated NALPHA times, each with an angle of attack and associated side slip angle. The first card 2.3 is orientation case 1, the second card is case 2, etc.

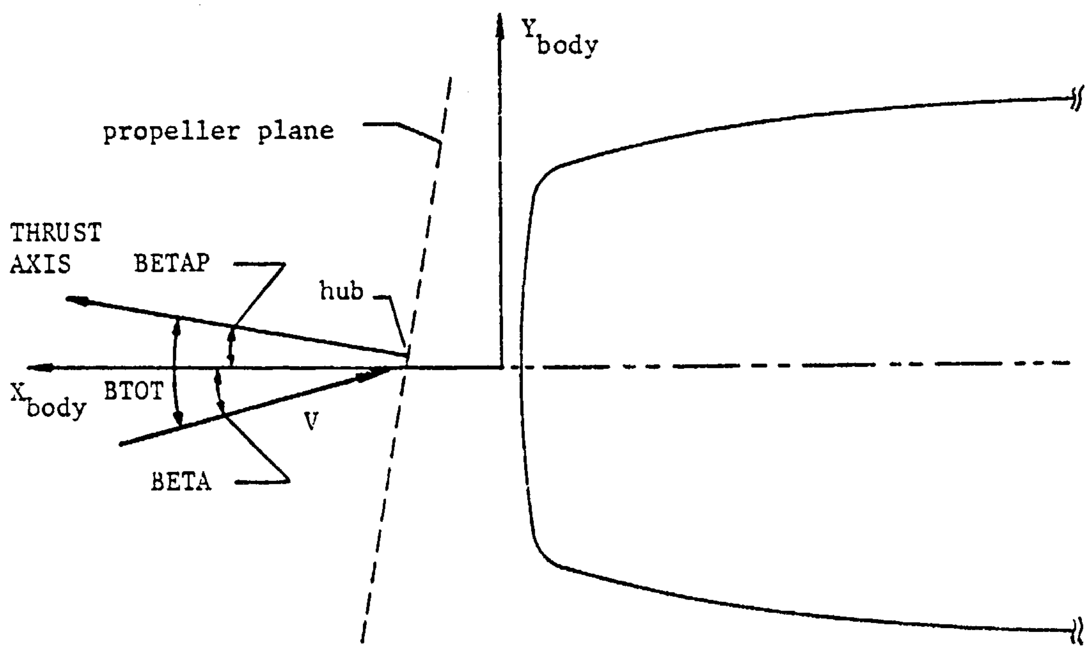
(3) ALPHA is the angle of attack between the free stream velocity (projected onto the reference body x-z plane) and the reference body x axis. It is positive for body nose upward. See Figure 3.1.

(4) BETA is the angle between the free stream velocity (projected onto the reference body x-y plane) and the reference body x axis. It is positive for body nose to the right. See Figure 3.1.

(5) All BETA(I), I=1, NALPHA must be zero if the symmetric body input option (NSYMET = 0) is specified in card set 3 below.



(a) Side view



(b) Top view

Figure 3.1 Body and propell r plane orientation angles.
(All angles shown positive.)

Card Set 3: Symmetry Option and Body Paneling Input

This set of cards specifies the use of symmetry or non-symmetry in panel input. The body cross section geometry points are given here. The left side of the geometry is given if symmetry is used. Otherwise both sides of the geometry must be given. All geometry is in terms of the reference body aircraft axis system (x foreward, y right, and z down). The origin may be anywhere desired except if symmetry is used. If symmetry is used, the origin must be on the plane of symmetry. See also section 1 in the manual which explains body paneling. (Set contains 11 to [(NSECMA * NIPMAX) + NSECMA + 2] cards.)

Card 3.1 Symmetric Body Input Option Choice

Notes	Variable	Column	Description
(1,2,3)	NSYMET	1	Integer=0 if symmetric option used. Integer=1 if option is not used.

Card 3.2 Number of Body Cross Sections

Notes	Variable	Columns	Description
(4)	NSECTO	1 - 2	Integer. Total number of body cross section descriptions given below. ($3 \leq$ NSECTO $\leq$ NSECMA)

Card Groups 3.3 Cross Section Descriptions

NSECTO card groups 3.3 are given in sequence. Each group is associated with a body cross section description. Each group is composed of one card 3.3.A followed by several cards 3.3.B as shown below. Groups are in sequence from 1 (first and front section) to NSECTO (last section). Refer also to section 1 of this manual.

Card 3.3.A Cross Section Identifier

Notes	Variable	Columns	Description
(5)	NSEC	1 - 2	Integer. Cross section sequence number. ($1 \leq$ NSEC $\leq$ NSECTO)
(6)	NIP or NCOUNT(I)	10 - 11	Integer. Number of periphery points on this I th = NSEC th cross section. ($3 \leq$ NIP $\leq$ [NIPMAX + 1]/2) if NSYMET=0 ($4 \leq$ NIP $\leq$ NIPMAX) if NSYMET=1
(7)	NEND or NFLAG(I)	30	Integer. Section repeat signal for this I th = NSEC th section. Equals 0 or 1 as explained in note.

Cards 3.3.B Section Periphery Points (NIP of these cards given for section NSEC identified in card 3.3.A above.)

Notes	Variable	Columns	Description
(8)	X or PX(I,J)	1 - 10	Floating point coordinates of the J th point around the periphery of the I th = NSEC th cross section. Coordinates relative to the reference aircraft coordinate system.
(8)	Y of PY(I,J)	20 - 29	
(8)	Z or PZ(I,J)	40 - 49	

Notes for Card Set 3

(1) Symmetric input option (NSYMET=0) should be used if possible to reduce solution time and cost.

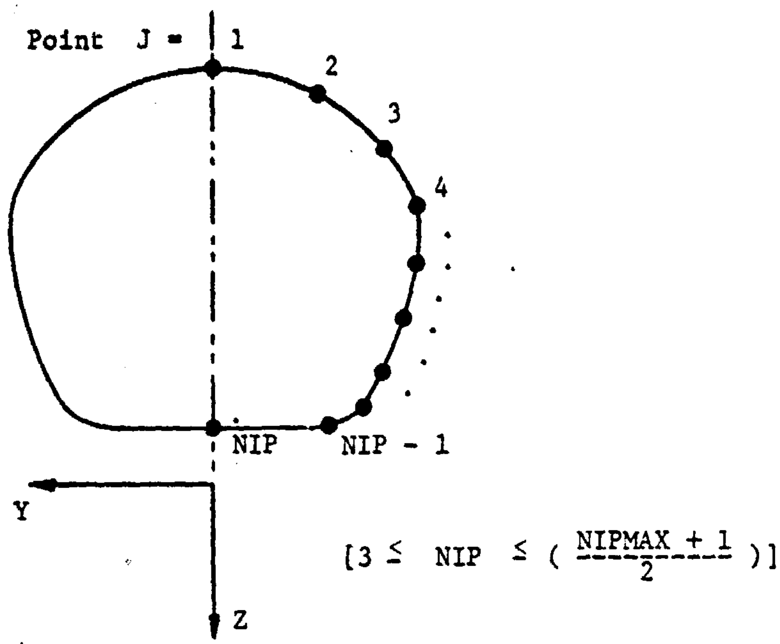
(2) Symmetry option (NSYMET=0) is possible only if all the following rules are met.
- All BETA (cards 2.3) must be zero.
- Origin of the body reference aircraft coordinate system must be on the body plane of symmetry with the plane containing the x and z axes.
- Configuration may not have discrete closed bodies to the left or right of the plane of symmetry even if such bodies have a counterpart on the other side of the plane of symmetry. Example is engine nacelles on the wings. So twin engine aircraft cannot use the symmetry option.
- Only the inlet/outlet panels on the left (-y) side are specified in card set 5 and such panels have symmetric counterparts on the right side of the body which will be generated automatically by the program.
- If a wing is to be modeled (card set 6 below) the wing must be symmetric with the wing root located on the body plane of symmetry.

If any rule above is not obeyed then must specify NSYMET=1.

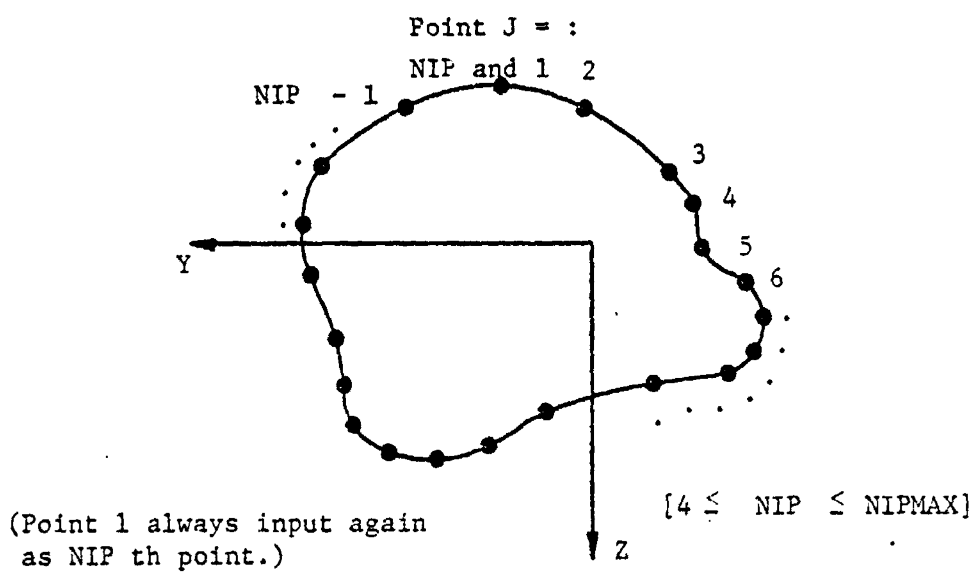
(3) If the symmetry criteria are met and NSYMET=0 is used then read in only the left (-y) side of the body geometry points on cards 3.3.B. See Figure 3.2.a.

If the general non-symmetric input option (NSYMET=1) is used then input all points on a cross section on cards 3.3.B. See Figure 3.2.b.

(4) A body cross section is at a constant x or nearly constant x location on the body. A physical cross section may be repeated with a new point description, but the repeat is counted as an additional section and contributes to the NSECTO amount of sections.



(a) Symmetric input option



(b) General or nonsymmetric input option

Figure 3.2 Input of periphery points around a cross section.

(5) Cross sections are input sequentially so NSEC must be consecutively increasing each time a card 3.3.A is punched. The first NSEC must 1 then 2, 3 ... NSECTO.

(6) NIP value is read from cards but renamed and inserted in array entry NCOUNT(NSEC). If symmetry option is not used (NSYMET = 1) then NIP is the number of points completely around the section and the first (top) point is repeated as the last NIP th point. See figure 3.2.b. Refer also to section 1.2.1.

If symmetry input option is used (NSYMET = 0), then NIP is the number of points around the left half of the cross section with the first point on the top centerline and the NIP th point on the lower centerline. See Figure 3.2.a.

NIP on section NSEC must equal NIP on adjacent sections between which panels will be generated.

(7) NEND is read in but program stores it under a new name in array entry NFLAG(NSEC). Use NEND = 0 if this section NSEC is not going to be repeated with a second description numbered NSEC + 1.

Or use NEND = 0 if this is the last input section of the entire configuration. (NEND = 0 signals the program to generate panels between sections NSEC and NSEC + 1.) See section 1.2.1 of this manual.

Use NEND = 1 if this section NSEC, having NIP points, will be repeated as section NSEC + 1 with a new description using a different number of periphery points or the same number of, but different, points.

Or use NEND = 1 if the following applies: This section NSEC is the last input section of a discrete body in a multibody configuration and another body follows this one. The next section NSEC + 1 is the start of a new discrete body panel network. For example, NSEC is the last section on the left nacelle (NFLAG = 1 for this), and section NSEC + 1 is the first section at fuselage nose. (NEND = 1 signals program not to generate panels between sections NSEC and NSEC + 1.) NEND = 1 can never appear in two consecutive cross section inputs.

(8) The coordinates X, Y, and Z are read from input and the program saves them under new names respectively in the array entries PX(NSEC,J), PY(NSEC,J), and PZ(NSEC,J) where J is the J th point around section NSEC as by the following rules.

The cards 3.3.B (points X,Y,Z) must be input in the sequential order illustrated by Figures 3.2.a and 3.2.b:

- For NSYMET = 0 the first card must be the top centerline point. Points are input consecutively clockwise around the left side of the section. The last card of the section is the NIP th point and it must be on the lower centerline of the section.
- For NSYMET = 1 the first card must be the upper point on the section. Points are input consecutively clockwise around the left side to the right side of the section. The last card is the NIP th point and must be an exact repeat of the first point.

Card Set 4: Test Run Option and Panel Geometry Print Options (Set always contains 2 cards.)

Card 4.1 Panel Geometry Print Option

Notes	Variable	Column	Description
(1)	NLIST	1	Integer = 0 if printout of panel geometry table is desired. Integer = 1 if printout is not desired.

Card 4.2 Geometry Check Run Option

Notes	Variable	Column	Description
(2)	NCALC	1	Integer = 0 if this is normal run for complete flow solutions. Integer = 1 if this is a test run to check generated body panel geometry.

Notes for Card Set 4

(1) Generated body panel geometry table printed using NLIST = 0 contains the following:

- Body panel index numbers (in sequence generated).
- Input corner points used to make panel.
- Computed control point coordinates of all panels.
- Surface area of each panel.

(2) Check run option (NCALC = 1) is used to get a listing of generated body panels. The listing helps check user geometry input and enables user to identify the index numbers of panels he wishes to specify as inlet or outlet panels if these are not known.

NCALC = 1 stops program after panel generation and does not allow flow solution to continue.

NCALC = 1 overrides the NLIST option and always causes the geometry to be printed.

Use of NCALC = 1 assumes that user does not know the index numbers of any panels which must be specified as inlet or outlet panels for the flow solution. Therefore the data deck must be organized as follows if a test run is to be made.

- Set NCALC = 1 on card 4.2.
- Specify NINFLO = 0 on card 5.1 below.
- All other card sets contain data as if this were a normal run.

After making the test run, the geometry output will allow the inlet and outlet panels to be identified by index number.

- Modify Card Set 5 by specifying the inlet and outlet panels as found by the test run.
- Change to NCALC = 0 on card 4.2.
- Resubmit data as a normal run to get flow solutions.

<u>Card Set 5: Inlet and Outlet Panel Specifications</u> (Set contains 1 to MAXINF + 1 cards.)

<u>Card 5.1 Number of Inlet and Outlet Panels</u>

Notes	Variable	Columns	Description
(1)	NINFLO	1 - 4	Integer. Number of inlet and outlet body panels. Specified on cards 5.2 below. ($0 \leq$ NINFLO $\leq$ MAXINF) if NSYMET = 1 [$0 \leq$ NINFLO $\leq$ (MAXINF/2)] if NSYMET = 0

<u>Cards 5.2 Inlet/Outlet Panel Index Numbers--Velocity Ratios</u> (Omit this card of NINFLO = 0, otherwise provide NINFLO of these cards.)

Notes	Variable	Columns	Description
(2)	INDEX(I)	1 - 4	Integer. Panel index number of the I th inlet or outlet panel as generated by program.
(3)	FLRATO(I)	20 - 29	Floating point. Inlet or outlet velocity ratio on panel INDEX(I). = 0.0 if panel is solid wall (not an inlet or outlet). > 0.0 if inflow velocity (INDEX(I) is an inlet panel). < 0.0 if outflow velocity (INDEX(I) is an outlet panel).

Notes for Card Set 5

(1) For symmetric option (NSYMET = 0), NINFLO is the number of inlet and outlet panels only on the left (-y) side of the body.

If the symmetry option is not used (NSYMET = 1), then NINFLO is the total number of inlet and outlet panels on the entire configuration.

(2) If NSYMET = 0, then the INDEX(I) of the left (-y) side inlet/outlet panels are specified. The ones on the right side of the body are automatically taken care of by the program.

If NSYMET = 1, then the INDEX(I) values of all inlet/outlet panels of the configuration must be given by the user.

In either of the above situations, the INDEX(I) values on cards 5.2 may be input in any order desired and do not have to be in any special sequence.

If user does not know which panel index numbers correspond to panels which are on inlets and outlets (as may occur with a new untried input geometry in card set 3), it will be necessary to make a test run to get a listing of generated panels. To do this set NCALC = 1 on card 4.2, set NINFLO to 0, and follow the test run procedures given in note (2) at the end of card set 4.

(3) FLRATO is a non-dimensional velocity (velocity/V) which is specified normal to the panel surface along the inward normal (positive value) or along the outward normal (negative value). The solid wall boundary condition will be relaxed in the equation for this panel number INDEX(I).

If NSYMET = 0, each FLRATO(I) value will correspond to left (-y) side panels INDEX(I). Program automatically assigns the same FLRATO(I) values to the right side panels.

FLRATO(I) = 0.0 is valid but should not be used. It is better to delete this panel INDEX(I) from the card set 5.

Card Set 6: Wing Input Data (Set contains 1 to MAXALP + 2 cards.)

Card 6.1 Wing Input Option

Notes	Variable	Column	Description
none	NWING	1	Integer = 0 if wing is not modeled. Integer = 1 if wing is modeled as given in cards 6.2 and 6.3 below.

Card 6.2 Wing Geometry (Delete this card if NWING = 0)

Notes	Variable	Columns	Description
(1)	XQR	1 - 10	Floating point coordinates of the wing root quarter chord location. In the reference aircraft coordinate system.
(1)	YQR	11 - 20	
(1)	ZQR	21 - 30	
(2)	SPAN	31 - 40	Floating point. Wing span as seen in planform view.
(2)	CHORD	41 - 50	Floating point. Wing root chord length.
(2)	DIHED	51 - 60	Floating point. Wing dihedral angle in degrees. Positive for wing tips upward from horizontal.
(2)	SWEEP	61 - 70	Floating point. Sweep angle of wing quarter chord line in degrees. Positive for aft sweep.

Card 6.3 Wing Lift Coefficients (Delete these cards if NWING = 0)
(If NWING = 1, repeat card 6.3 NALPHA times.)

Notes	Variable	Columns	Description
(3)	CL(I)	1 - 10	Floating point. Value of wing lift coefficient corresponding to ALPHA(I) of the I th orientation case on card 2.3.

Notes for Card Set 6

(1) Wing root is the location midway between wing tips; it is not the location of the wing - fuselage interface. See figure 3.3.

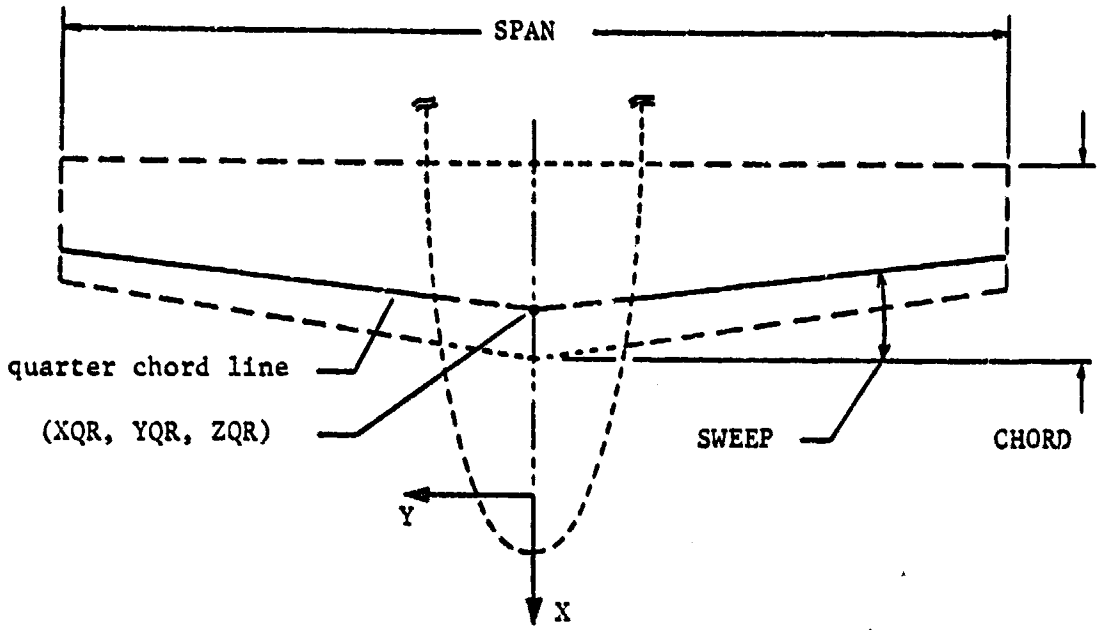
If the symmetric body input option (NSYMET = 0) has been used in card set 3, then the following must be met for the wing:

- The wing must be symmetric about the body plane of symmetry. (i.e. YQR = 0.0 is required.)

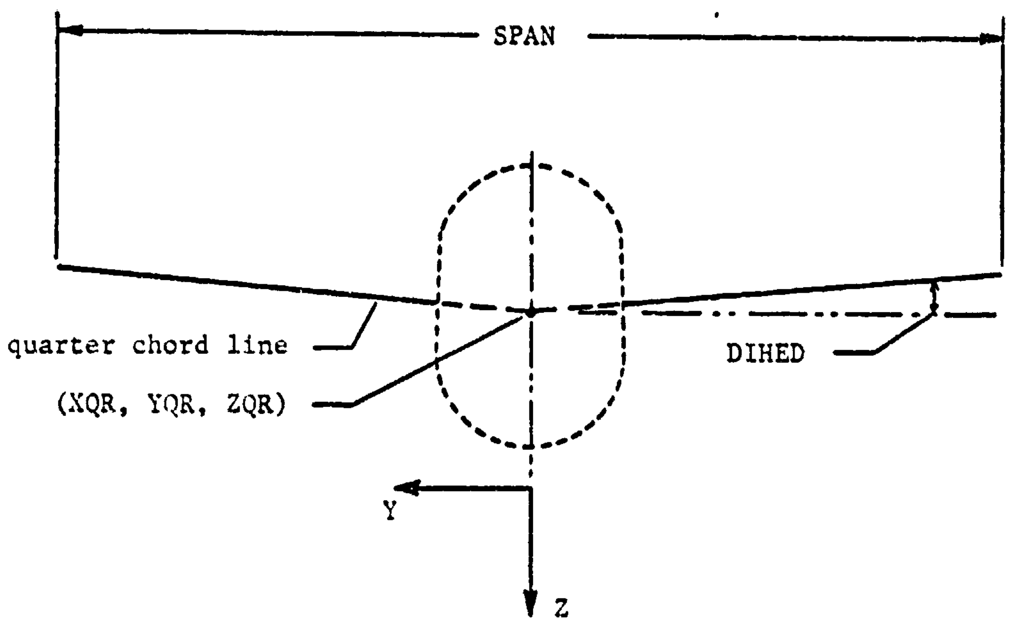
(2) See figure 3.3 for definition of angles.

(3) Put one CL value on each of the NALPHA cards 6.3. These cards must be in sequence corresponding to cards 2.3 such that the first card 6.3 has CL(1) corresponding to ALPHA(1) on the first card 2.3 ... etc.

CL(I) = 0.0 is valid and has the same effect as not modeling the wing for orientation case I.



(a) Plan view



(b) Front view

Figure 3.3 Input wing geometry

Card Set 7: Propeller Plane Description (Set has either 1 or 4 cards.)

Card 7.1 Propeller Plane Specification Option

Notes	Variable	Column	Description
none	NPOINT	1	Integer = 0 if no propeller plane is defined. Integer = 1 if propeller plane is modeled as defined on cards 7.2 and 7.3. Propeller plane flowfield predictions will be made.

Card 7.2 Propeller Plane Location and Orientation (Delete this card if NPOINT = 0)

Notes	Variable	Columns	Description
(1)	XHUB	1 - 10	Floating point coordinates of the hub or center point location of the propeller plane. In the reference aircraft coordinate system.
(1)	YHUB	11 - 20	
(1)	ZHUB	21 - 30	
(2)	ALPHP	31 - 40	Floating point. Built-in vertical tilt angle of propeller plane in degrees. (Value of 0 to 360. or 0 to -360.)
(3)	BETAP	41 - 50	Floating point. Built-in sidward tilt angle of propeller plane in degrees. (Value of 0 to 360. or 0 to -360.)
(4)	RADIUS	51 - 60	Floating point. Reference propeller plane radius. (Propeller blade length.)

Card 7.3 Distribution of Flow Prediction Points on Propeller Plane (Delete this card if NPOINT = 0)

Notes	Variable	Columns	Description
(5,7)	DPSI	1 - 10	Floating point. Azimuth angle increment (degrees) for spacing of radial rows of points around the propeller plane. Value must be positive in the range from $DPSI_{min}$ to 360.)
(6,7)	DRADUS	11 - 20	Floating point. Fractional percentage of RADIUS (card 7.2) giving the radial spacing of points along a constant azimuth in the propeller plane. Dimensionless value of 0.0 or any positive number.

Card 7.3 (continued)

Notes	Variable	Columns	Description
(6,7)	NRAD	21 - 22	Integer. Number of points to be distributed radially along each azimuth position. Value from 1 to $NRAD_{max}$.

Card 7.4 Propeller Pland Flow Field Punched Card Output Option (Delete this card if NPOINT = 0)

Notes	Variable	Column	Description
(7)	NPUNCH	1	Integer = 0 if no punched card output desired. Integer = 1 if desire punched card output.

Notes for Card Set 7

(1) Propeller plane may be positioned at any location desired. (See figure 3.1.) It need not be positioned on the plane of symmetry, even if NSYMET = 0. (i.e., if NSYMET = 0, YHUB = 0.0 is not required.)

Also the hub may be located within the body. For example, the hub inside of a spinner may be modeled. However, flow predicted at propeller plane regions inside the body will have no physical meaning.

(2) ALPHP is a fixed angle of pitch between the propeller thrust axis and the body reference x axis. It is the angle as seen projected on the body x-z plane. (i.e. as seen in the side view of the body drawing.) Angle is positive if propeller thrust axis is tilted upward. See figure 3.1.

(3) BETAP is a fixed yaw angle between the propeller thrust axis and the body reference x axis. It is the angle as seen projected on the body x-y plane. (i.e., as seen in the top view of the body drawing.) A yawed propeller plane is permitted even whem the symmetry option has been specified. (i.e., if NSYMET = 0, BETAP need not be zero.) BETAP is positive for thrust axis tilt toward the right (+y) side of the body. See figure 3.1.

(4) RADIUS is used as a scale length for nondimensionalizing the radial positions of points in the propeller plane. Also the RADIUS length determines how much propeller plane area is mapped in conjunction with values of DRADUS and NRAD given in card 7.3. Choice of larger RADIUS will move propeller points farther out from the hub. See figure 3.4.

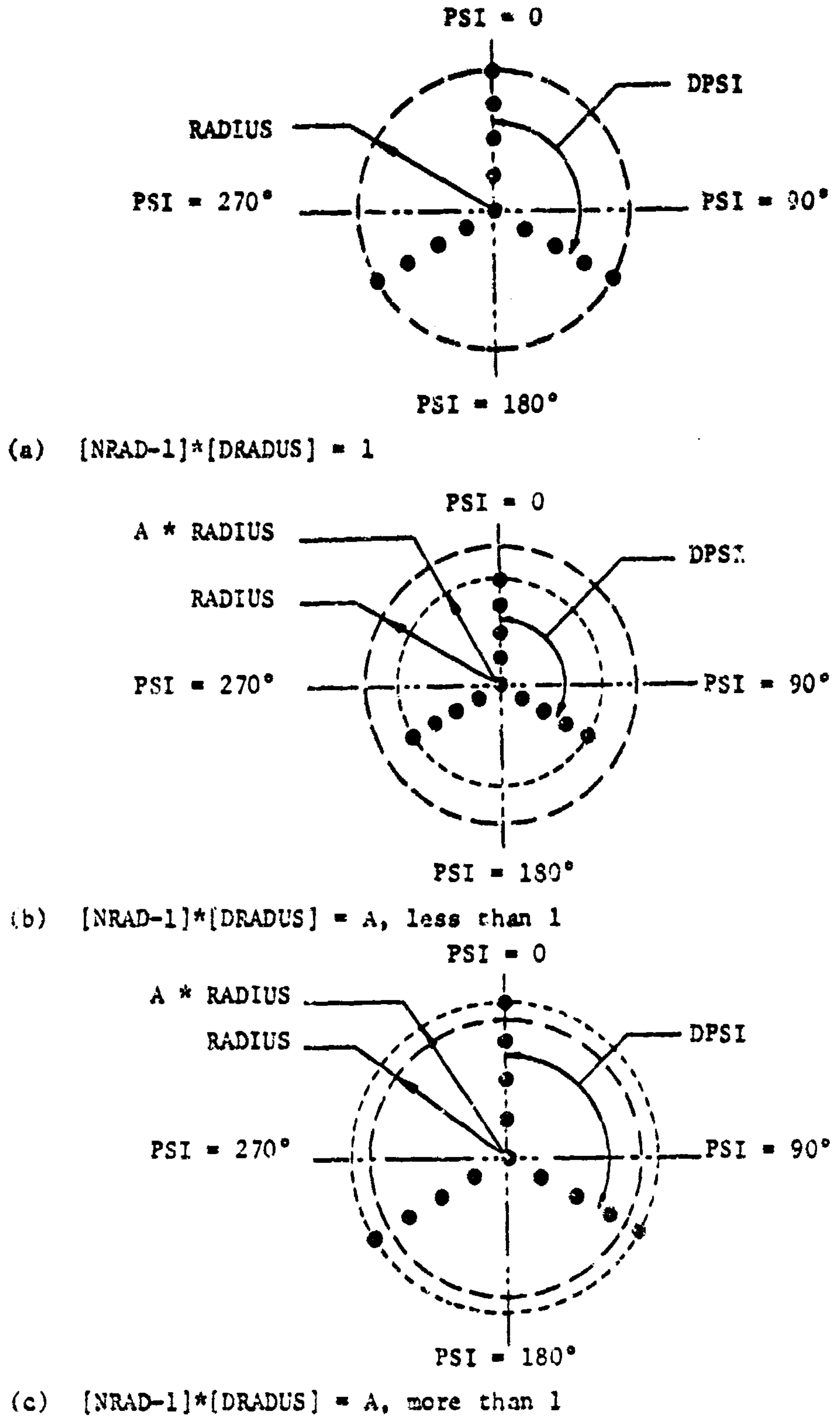


Figure 3.4 Distribution of survey points on propeller plane viewed in thrust direction.

(5) DPSI should be chosen such that it divides 360.0 into whole parts. (i.e. (360/DPSI) = integer). This ensures an even azimuthal spacing or mapping of points around the propeller plane. See figure 3.4.

(6) NRAD always includes the hub point. The flow at the hub will be calculated repeatedly as each azimuthal row of points is examined.

The choices of NRAD and DRADUS, in combination, will determine the radial spread of points over the propeller plane of RADIUS size. The point spread is determined by one of the following: (see also figure 3.4)

- if [(NRAD - 1)*DRADUS] = A, less than 1: Points will be spread evenly from hub to a distance A*RADIUS, inside of the RADIUS circle (Figure 3.4.b)
- if [(NRAD - 1)*DRADUS] = A, more than 1: Points will be spread evenly from hub to a distance A*RADIUS, outside of the RADIUS circle (Figure 3.4.c)
- if [(NRAD - 1)*DRADUS] = 1.0: Points will be spread evenly from hub to the edge of the RADIUS circle. (Figure 3.4.a)

(7) If card punched output is made (NPUNCH = 1), and if the punched output is to be input to the propeller analysis program of reference 3, then the following conditions must all be met:

- DPSI must be set equal to 15.0,
- DRADUS must be set equal to 0.05, and
- NRAD must be set equal to 21

Card Set 8: Solution Iteration Control Card (1 card)

Card 8.1

Notes	Variable	Columns	Description
none	ITMAX	1 - 2	Integer. Maximum allowed iterations in Gauss Seidel Solution for source strengths. Value from 1 to 99, but recommended values are 15 to 20.
(1,2)	ERR	11 - 20	Floating point. Exponent on the value of the largest permitted relative error. Defines solution convergence limit. Recommended value is -4.00.

Notes for Card Set 8

(1) Target solution error residual is given as:
error = 10.0^{ERR} Thus ERR should be a negative number to get useful solutions.

(2) All body orientation cases, [ALPHA(I), BETA(I)], for I = 1 to NALPHA, are solved together during each iteration. All orientation cases must converge to the error limit within ITMAX iterations. If one or more cases fail to converge, all solutions which did converge will also be lost. Program will abort and no solutions will be given.

This concludes the input data descriptions.

Appendix C describes a sample run case and shows an example input data deck.

SECTION 4 OUTPUT DATA DESCRIPTION

All output is written to a standard 132 character-per-line printer. Additionally, if NPUNCH = 1 is specified, certain parts of the propeller plane flow field prediction output is punched on standard 80 column punch cards. These cards are intended for use as input to the propeller aerodynamic analysis computer program described in references 2 and 3.

The total number of printed records depends primarily upon the number of panels, NP, on the configuration and on the number of body orientations run. Refer to section 2.10 of this manual for output record requirements.

The number of punched output records depends primarily upon the number of points specified on the propeller plane, and the number of body orientations run. Again, see section 2.10.

In the remainder of this section, the printed output is described followed by a detailed description of the contents and format of the punched card output data.

4.1 Printed Output Description

Printed output can be categorized into six major parts. Depending on the choice of print options and configuration, some of these parts may not always be printed. The conditions required for printing of a part are given with the description. The following six output parts are described in the sequence in which they appear on the printout.

Part 1: Input Data Card Listing (Always Printed)

This table lists all of the input data card variables in the order read in. The values are identified by their variable name. This input listing is printed during execution of subroutine INPUT. If any detectable error is discovered in reading a data card, that incorrect data is printed with an error message. The output will stop at that point, the remaining data will not be evaluated, and the program will abort. If all data is read successfully, this is indicated at the end of the table.

<u>Part 2: Wing Horseshoe Vortex Strengths and Geometry Table</u> (Printed only if NWING = 1)

This table is printed by subroutine WGEOM only if a wing has been modeled. The physical wing input parameters are printed first. Next the geometry of the wing bound vortex, along the quarter chord, is printed. This is constant regardless of body orientation. Lastly the table gives the geometry of the two trailing vortex filaments. This geometry varies with body orientation, so each orientation is given along with the input body angle of attack, wing lift coefficient, and computed strength of the horseshoe vortex.

<u>Part 3: Listing of Generated Body Panel Geometry</u> (See note 1 below.) (Printed only if test run, NCALC = 1 with NLIST = 0 or 1, has been made. Or printed if a normal run, NCALC = 0 with NLIST = 1, has been made.)

This tabular listing gives the geometry of the body paneling produced by subroutine PANEL from the input cross section geometry. The first column gives the panel index number. Knowledge of these index numbers is needed for specifying inlet and outlet panels on input card set 5. The next columns give the four input corner points of the panel. Triangular panels may be recognized by a repetition of one of the corner points. Next is printed the computed control point coordinates of the panel, relative to the body reference axes. Lastly, is printed on the surface area of the panel. The end of this table is denoted by a completion message and a tally of the total number of panels, NP, on the configuration.

Note 1: For a test run (NCALC = 1) part 3 above is the last output data obtained.

The following printed parts and punched data are obtained if a normal run (NCALC = 0) was made:

<u>Part 4: Solution of Source Strengths, SIGMA, Table</u> (Always printed when NCALC = 0)

This table gives a summary of the progress during the iterative solution of equations for the panel source strengths. The iteration and target error limits are given. Then for each iteration performed, the body orientation case, body angle of attack, and sideslip angle are printed along with the resultant solution error for that iteration. As an orientation case converges, this is indicated by a message.

If all orientations converged within the iteration limits, this is stated in a message at the end of this output. Then the remaining parts 5 and 6 will be printed.

If one or more orientations fail to converge, an error message is printed. All converged solutions are lost and the program is aborted.

Part 5: Body Surface Velocities and Pressures Table (See note 2 after part 6 description.) (Always printed when NCALC = 0)

One of these tables is produced for each body orientation case. This output is printed by subroutine VELOCI. The table gives the orientation case number, body angle of attack, and sideslip angle. Free stream velocity magnitude and components are printed as well as the total number of body panels. Each panel index is listed with its control point. The nondimensionalized cartesian velocity components computed at the control point are printed. These components are relative to the body reference coordinates system. Lastly the pressure coefficient, CP, at the panel is given.

Part 6: Propeller Plane Flow Field Output Table (See note 2 below.) (Printed when both NCALC = 0 and NPOINT = 1)

This output gives the flow field predictions in the propeller plane for one body orientation case. Part 6 is divided into two subparts 6.A and 6.B described below. Part 6 output is produced by subroutine VPROPS.

Subpart 6.A Propeller Plane Output (Part 1)

This table prints the orientation case number as well as body angle of attack, ALPHA, and sideslip, BETA. Free stream velocity magnitude, V, and components are given along with the input propeller reference radius, RADIUS. Refer to figure 4.1.

The table proceeds point-by-point in the plane. The point location in cylindrical coordinates (R, PSI) and the cartesian coordinates of the location, in the body reference coordinate system, are printed. Lastly, the nondimensionalized cartesian velocity components, VX_{prop}, VY_{prop} and

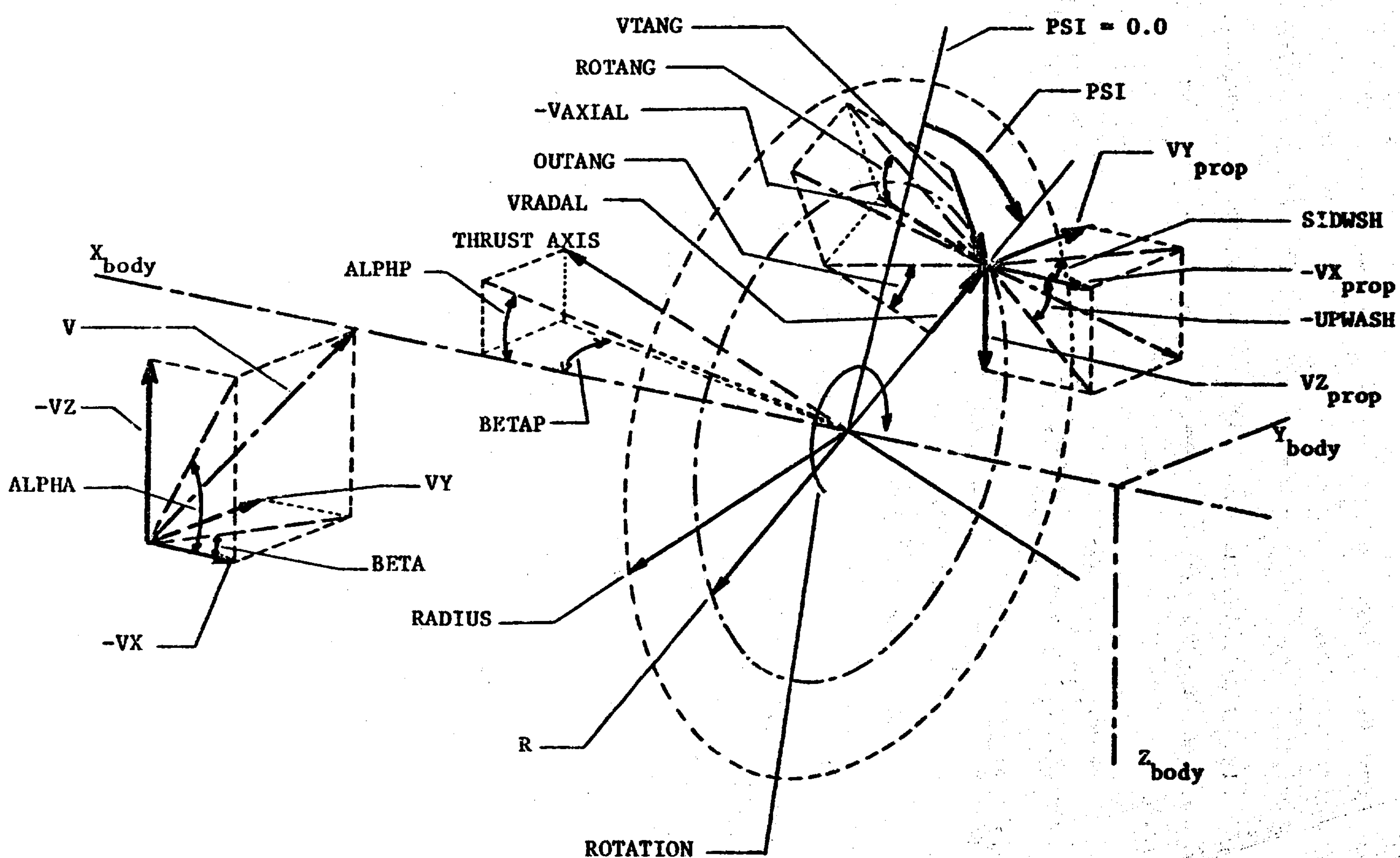


Figure 4.1 Propeller plane orientation, body orientation, and flow velocities and angles at a propeller plane survey point.

VZ_{prop}, at the propeller plane point are given. These velocities are relative to the body reference coordinate system. They represent the contribution of velocities induced by solid panels, inlet/outlet panels, the wing, and the free stream velocity. Again refer to figure 4.1.

Subpart 6.B Propeller Plane Output (Part 2)

In this table, each point in the propeller plane is identified by its radial-azimuthal position only. The nondimensionalized axial, tangential, and radial induced velocity components are printed. These components are relative to the cylindrical coordinate system attached to the propeller plane (see Figure 4.1). As in subpart 6.A these velocities represent the contributions of solid panels, inlet/outlet panels, the wing, and the free stream velocity. Lastly, the flow angles corresponding to the velocity components at a point are printed. These angles are rotational flow angle, outflow angle, upwash angle, and sidewash angle. Also these angles are named ROTANG, OUTANG, UPWASH, and SIDWSH, respectively, in the program and are shown on Figure 4.1. See also reference 1 for the definition of these angles.

Note 2: Output parts 5 and 6 are printed in pairs for a single body orientation case. The first pair of tables is printed for orientation case 1 followed by a second pair for orientation case 2, etc.

This concludes the description of the printed output. A sample of printed output for an example case is shown in Appendix C.

4.2 Punched Output Description and Format

This punched output card data is intended for use as part of the input cards to the propeller performance prediction program of reference 3. These cards may be directly used in the reference 3 program without modification.

Punched output on 80 column cards is obtained only if NPUNCH = 1 with NCALC = 0. This data is collected on auxiliary file 50 in subroutine VPROPS. Then the contents of file 50 are punched on the cards.

For each body orientation case a separate group of output cards is punched. The output card groups are punched in the order matching the input order of the body orientations (input cards 2.3). Therefore, there will be NALPHA card groups punched. The contents of a card group is given in detail below. Note a single card group (below) represents the complete card deck needed for input data set III of the propeller performance prediction program in reference 3. See this reference for details.

Columns indicates the card column field in which a value is punched. F format represents floating point numbers. I format denotes integer values which are always punched right justified in their column fields.

Each card group, I: I = 1 to NALPHA, contains three sets (types) of cards as follows:

Card Set 1: Title Cards (always 2 cards)

Notes	Variable	Columns	Format	Description
none	SYMBOL(I), I = 1, 160	1 - 80	Alpha-numeric	Input title supplied by user on cards 1.1 and 1.2 of input data.

Card Set 2: Overall Propeller Plane Orientation for Group I (1 card)

Notes	Variable	Columns	Format	Description
(1)	NPSIS	1 - 5	I5	Number of azimuth positions.
(2)	NRAD	11 - 15	I5	Number of radial positions.
(3)	RADIUS	21 - 30	F10.4	Reference propeller plane radius.
(4)	ATOT	31 - 40	F10.4	Total propeller angle of attack. (Degrees)
(5)	BTOT	41 - 50	F10.4	Total propeller plane side-slip angle. (Degrees)
(6)	ALPHA(I)	51 - 60	F10.4	Body angle of attack for orientation case I. (Degrees)
(7)	BETA(I)	61 - 70	F10.4	Body sideslip angle for orientation case I. (Degrees)

Notes for Card Set 2

(1) This is calculated by program from input value of DPSI (input card 7.3) as follows:

NPSIS = (360./DPSI) rounded to nearest lower integer

(2) NRAD is the same value specified on input card 7.3.

(3) RADIUS is the same value specified on input card 7.2 (see Figure 4.1).

(4) For output card group I (body orientation case I), ATOT is the net propeller plane angle of attack between the propeller thrust axis and the free stream velocity as seen projected onto the body x-z plane. It is given by the input angles:

ATOT = ALPHA(I) + ALPHP (see Figures 3.1.a and 4.1)

where ALPHA(I) was given on the I th card 2.3 of input.
ALPHP is the angle given on input card 7.2.

(5) For output card group I (body orientation case I), BTOT is the net propeller plane sideslip angle between the propeller thrust axis and the free stream velocity as seen projected on the body x-y plane. It is given by the input angles:

BTOT = BETA(I) + BETAP (see Figures 3.1.b and 4.1)

where BETA(I) was given on the I th card 2.3 of input.
BETAP is the angle given on input card 7.2.

(6) ALPHA(I) is the same value specified on input cards 2.3.

(7) BETA(I) is the same value specified on input cards 2.3.

Card Set 3: Propeller Plane Point--Velocity Component Cards (For each card group there are NPSIS * NRAD of these cards punched.)

Notes	Variable	Columns	Format	Description
(1)	PSI	1 - 10	F10.4	Azimuth position of point. (Degrees)
(2)	RRATIO	21 - 30	F10.4	Radial position of point. (Nondimensionalized)
(3)	VAXIAL	41 - 50	F10.4	Axial component of velocity at point. (Nondimensionalized)
(4)	VTANG	61 - 70	F10.4	Tangential component of velocity at point. (Nondimensionalized)

One card for each point is punched. The card order is such that RRATIO increases most rapidly:

For example--Assume the points are spaced at every 15 degrees of azimuth with 11 points radially spaced at 10 percent of RADIUS.

The card set 3 order will be:

first NRAD cards at PSI = 0.00 with RRATIO = 0.0 to 1.0
second NRAD cards at PSI = 15.0 with RRATIO = 0.0 to 1.0

.

.

NPSIS th NRAD cards at PSI = 345.0 with RRATIO = 0.0 to 1.0

Notes for Card Set 3

(1) Aximuth angles are clockwise increasing around the propeller plane disk as viewed in the direction of thrust. The first azimuth is always 0.0. Each azimuth position is spaced DPSI degrees from the others. DPSI was input on data card 7.3 (see Figures 3.4 and 4.1).

(2) RRATIO is the dimensionless distance outward from the propeller plane hub to the point. It is a fraction of the reference propeller plane radius as follows:

RRATIO = R/RADIUS

where R is the dimensional radial distance from hub to point (Figure 4.1).

(3) VAXIAL is nondimensionalized by dividing by the free stream velocity magnitude, V (on input card 2.1). VAXIAL is positive in the direction of propeller thrust (right hand propeller rotation assumed). (See Figure 4.1)

(4) VTANG is nondimensionalized by dividing by the free stream velocity magnitude, V (given on input card 2.1). VTANG is positive if it follows the local direction of blade rotation (right hand propeller rotation assumed). A positive value of VTANG, as seen by an observer positioned on a blade section, would subtract from or reduce the net onset velocity the blade section would see (see Figure 4.1).

This concludes the description of the program output data.

REFERENCES

1. Jumper, Stephen J., "Computer Prediction of Three-Dimensional Potential Flow Fields in Which Aircraft Propeller Operate," M.S. Thesis, The Pennsylvania State University, 1980.

2. Aljabri, Abdullah S., "Prediction of Propeller Performance and Loading in Uniform and Nonuniform FlowFields," M.S. Thesis, The Pennsylvania State University, November 1978.

3. Aljabri, A. S., "Prediction of Propeller Performance and Loading in Uniform and Nonuniform Flow Fields (Computer Program Description and Users Manual)," The Pennsylvania State University, 1979.

4. Woodward, F. A., Dvorak, F. A., and Geller, E. W., "A Computer Program for Three-Dimensional Lifting Bodies in Subsonic Inviscid Flow," Report USAAMRDL-TR-74-18, April 1974.

5. Hess, J. L. and Smith, A. M. O., "Calculation of Potential Flow About Arbitrary Bodies," Progress in Aeronautical Sciences, Pergamon Press, Vol. 8, 1967.

6. "JCL Techniques," The Pennsylvania State University Computation Center, revised January 1976.

7. Roberts, J. C. and Yaggy, P. F., "A Survey of the Flow at the Plane of the Propeller of a Twin-Engine Airplane," NACA TN 2192, September 1950.

8. "The Batch and Terminal (BAT) File System," The Pennsylvania State University Computation Center, revised December 1977.

APPENDIX A

Main Program and Subroutine Descriptions

(Main program given first followed by subroutines in alphabetical order)

Each of the following descriptions summarizes the purpose, method, and use of the subroutines. Also given is the name of common blocks required in each. The principal input variables are listed, generally in order of first appearance. Similarly, the output variables are listed.

The parts of the computer program described are:

MAIN PROGRAM
Subroutine ANGLES
Subroutine COEFIC
Subroutine COFSYM
Subroutine EULER
Subroutine INPUT
Subroutine PANEL
Subroutine SOLSYM
Subroutine SOLVE
Subroutine VCOMP
Subroutine VELOCI
Subroutine VORTEX
Subroutine VPROPS
Subroutine WGEOM
Subroutine WINGV

MAIN PROGRAM

Purpose: To establish program array dimension sizes and define values of execution time dimensioning variables. Also to link subroutines together and ensure they are executed in the proper sequence.

Method: User specifies the values of four dimension size control variables and properly dimensions arrays in the dimension statement at the start of the program. Refer to section 2 of this manual for details.

The main program takes the user supplied sizing variables and from these, calculates other sizing variables. All of these quantities are checked to ensure they have valid values as defined by the rules in section 2 of this manual.

If all the dimension sizing variables are valid, the main program executes the subroutines and uses the variables for automatic execution time dimensioning of all subroutine dimension statements.

Arrays in the COMMON statements of subroutines are not automatically dimensioned by the main program. These must be dimensioned by user by actually changing the COMMON cards in the subroutines if a change in program dimensioning variables is made. See section 2 of this manual.

Finally, if any of the dimension size variable values is invalid, the main program prints an error message identifying the problem. Program then aborts.

Common Blocks Required: INPUTS, SYMTRE, WING1

Input: (User changes the following cards if program dimensions are to be modified).

NSECMA Maximum limit on number of body cross section descriptions permitted in defining geometry.

NIPMAX Maximum limit on number of points specified around any complete body cross section.

MAXALP Maximum number of body orientations which may be solved in one run.

MAXINF Maximum total number of inlet and outlet panels which may be specified over the entire body.

Other Input: (obtained from data input)

NWING Wing input option = 0 if no wing modeled.
= 1 if wing is modeled.

NSYMET Symmetric body input option
= 0 if symmetry used.
= 1 if symmetry not used.

NALPHA Number of body orientations input.

Output: (Dimension size variables generated by program based on user supplied ones)

LLL Maximum total number of body panels which can possibly be generated consistent with geometry input limits of NSECMA and NIPMAX.

MD Sum of LLL and MAXALP, will equal number of coefficients in each equation for unknown source strengths.

LLLHAF LLL/2, represents half the total maximum number of body panels which may be generated.

Other Output:

ISTOP Dimensioning variable error detect signal
= 0 if no such errors found.
= 1 if one or more errors detected.

Subroutines Called: INPUT, VCOMP, WGEOM (if NWING = 1), PANEL, COEFIC (if NSYMET = 1), SOLVE (if NSYMET = 1), COFSYM (if NSYMET = 0), SOLSYM (if NSYMET = 0), VELOCI, and VPROPS

Error Stops: If ISTOP = 1
Print error message about dimensioning variables, run stopped.

Subroutine ANGLES

Purpose: To compute the rotational flow angle, outflow angle, upwash angle, and sidewash angle at a point on the propeller plane

Method: The rotational flow and outflow angles are computed in the propeller plane cylindrical coordinate system as defined in reference 1 and Figure 4.1 of this manual. Upwash and sidewash angles are computed in the reference body coordinate system as defined in reference 1 and Figure 4.1 of this manual.

Equations solved are inverse tangent functions and required logic is used which ensures proper sign and angle quadrant are maintained.

All angles are returned in degrees.

Use: CALL ANGLES(VTOTX,VTOTY,VTOTZ,VAXIAL,VRADAL, VTANG, PSI,ROTANG,OUTANG,UPWASH,SIDWSH)

Common Blocks Required: None

Note: All velocities given are values normalized by free stream velocity magnitude.

Inputs:

VAXIAL	Axial velocity component in propeller plane coordinate system at point.
VRADAL	Radial velocity component in propeller plane coordinate system at point.
VTANG	Tangential velocity component in propeller plane coordinate system at point.
VTOTX, VTOTY, VTOTZ	x, y, and z components of velocity at the propeller plane point relative to reference body coordinate system.
PSI	Azimuth position angle of propeller plane point.

<u>Output</u>:

OUTANG	Outflow angle.
ROTANG	Rotational flow angle.
UPWASH	Angle of upwash.
SIDWSH	Angle of sidewash.

Subroutines
Called: None

Error Returns: None

Error Stops: None

Subroutine COEFIC

Purpose: To calculate the matrix of coefficients of the full size system of linear equations. The equations are later solved for the unknown body panel source strengths.

Method: This subroutine is used only if the symmetric input option has not been used.

An equation is generated for every panel in the configuration.

The equations (rows of matrix) are created and numbered sequentially starting with panel 1 then panel 2, etc.

To generate an equation, the left side coefficients are obtained first. Then the right side coefficients are obtained. See reference 1 for description of the equations. The equation is then written on one logical record of the sequential scratch file 9.

The subroutine returns with one equation written for each panel and stored sequentially on file 9.

Use: CALL COEFIC(LLL,MD,NP,XC,YC,ZC,S,ANVX,ANVY,ANVZ,A)

Common Blocks Required: INPUTS, INLET, and WING1

Input:

LLL Program dimension size variable specified in main program which equals the maximum allowed number of body panels which may be generated. Used here for execution time dimensioning of arrays.

MD Program dimension size variable sepcified in main program which is sum of LLL and maximum allowed number of body orientations. Used here for execution time dimensioning of arrays.

NP Total number of body panels on configuration. Same as the number of equations generated.

XC,YC, ZC Arrays of coordinates of each body panel control point relative to reference body coordinate system.

S Array of body panel surface areas.

ANVX, ANVY, ANVZ	Arrays of components of the outward drawn normal unit vectors located at each panel control point. Relative to reference body coordinate system.
NINFLO	Total number of body panels specified as being inlet or outlet panels.
INDEX	Array of panel index numbers of body panels which have been specified as inlet or outlet panels.
NALPHA	Total number of body orientations. Same as the number of right side coefficients on each equation being generated.
NWING	Wing signal variable: = 0 if no wing is modeled. = 1 if wing is modeled.
FLRATO	Array of specified velocity ratios assigned to the inlet and outlet body panels.

<u>Output</u>: (Written on file 9)

A	Buffer array containing left and right side coefficients of an equation. Used to write each equation on the scratch file 9.

Subroutines Called: WINGV (only if wing is modeled on configuration)

Error Returns: None

Error Stops: None

Subroutine COFSYM

Purpose: To calculate the matrix of coefficients of the half size system of linear equations. The equations are solved later for the unknown body panel source strengths.

Method: This subroutine is used only if the symmetric input option has been used. (Body symmetry exists.)

An equation is generated only for the panels on the left side of the body. The right side image panels are automatically accounted for.

The equations (rows of matrix) are created and numbered sequentially starting with first left side panel then the second left side panel, etc.

To generate an equation, the coefficients on the left side of the equation are obtained first. Then the right side coefficients are obtained. See reference 1 for a description of these modified symmetric body equations. The equation is then written on one logical record of the sequential scratch file 9.

The subroutine returns with one equation written for each panel on the left side of the body. The equations are stored sequentially on file 9.

Use: CALL COFSYM(LLL,MD,NP,XC,YC,ZC,S,ANVX,ANVY,ANVZ,A)

Common Blocks Required: INPUTS, INLET, SYMTRE, and WING1

Input:

LLL Program dimension size variable specified in main program which equals the maximum allowed number of body panels which may be generated. Used here for execution time dimensioning of arrays.

MD Program dimension size variable specified in main program which is sum of LLL and maximum allowed number of body orientations. Used here for execution time dimensioning of arrays.

NPNSYM	Bookkeeping array which designates left side panels and records corresponding right side panels as follows:
	if NPNSYM(I) = 0, then panel whose index number is I is on the right side of the body.
	if NPNSYM(I) = (N $\neq$ 0), then panel whose index number is I is on the left side of the body and the corresponding panel on the right side of the body has index number N.
NP	Total number of body panels on configuration. Equals twice the number of equations generated.
XC, YC, ZC	Arrays of coordinates of body panel control points. (Only those on left side panels are used.)
S	Array of body panel surface areas.
ANVX, ANVY, ANVZ	Arrays of components of the outward drawn normal unit vector at each panel control point. Components relative to reference body coordinate system.
NINFLO	Number of body panels which are specified as being inlet or outlet panels.
INDEX	Array of panel index numbers of body panels which have been specified as inlet or outlet panels.
NALPHA	Total number of body orientations. Same as the number of coefficients on the right side of each equation.
NWING	Wing signal variable: = 0 if no wing is modeled. = 1 if wing is modeled.
FLRATO	Array of specified velocity ratios assigned to the inlet and outlet body panels.

Output:

INSOLV	Bookkeeping array which stores the index numbers only of the panels on the left side of the body. Index numbers are stored in sequence such that INSOLV(1) stores the index of the first left side panel, INSOLV(2) stores the index number of the second left side panel, etc.

Other Output: (Written on file 9)

A	Buffer array containing left and right side coefficients of an equation. Used to write each equation on the scratch file 9.

Subroutines Called:	WINGV (only if wing is modeled on configuration)
Error Returns:	None
Error Stops:	None

Subroutine EULER

Purpose: To calculate value of an euler angle used in coordinate transformation. The transformation being from the reference body coordinate system to either the wind axis or propeller fixed axis systems.

Method: Solves the inverse tangent function expression for the euler angle as given in reference 1. Logic is used to keep track of sign so proper quadrant value of angle is not lost.

Use: CALL EULER(A,B,ANADA)

Common Blocks Required: None

Input:

A	Value of either a body angle of attack or angle of attack of the propeller plane relative to the body (Degrees)
B	Is either the body sideslip angle or the yaw angle of the propeller plane relative to the body (Degrees).

Output:

ANADA	Euler angle used in coordinate transformation.

Subroutines Called: None

Error Returns: None

Error Stops: None

Subroutine INPUT

Purpose: To read in all input data to the computer program.

Method: Eight sets of data are read. (Refer to section 3 of this manual.) As each data card is read, the values are also printed along with the variable name.

The quantities read are checked for detectable errors which will be explained below in the section on error stops. When any error is detected in a value, a message describing the error is printed. The subroutine will stop processing data and will cause the entire program to terminate. If no errors are detected in the data, the input will be saved in arrays and variables. Then a listing will be printed which summarizes what further output tables to expect from the remainder of the program.

Control is then returned to the main program.

Use: CALL INPUT(LLL,MAXALP,MAXINF,NIPMAX,NSECMA,NSECTO, NCOUNT,NFLAG,PX,PY,PZ,ITMAX,ERR)

Common Blocks Required: TITLE, INPUTS, OPTION, PROP, SYMTRE, INLET, and WING1

Input: (Supplied from main program.)

NSECMA Program dimension size variable spceified in main program which equals the maximum number of body cross sections which may be used to describe the geometry. Used here for error checks and execution time dimensioning of arrays.

NIPMAX Program dimension size variable specified in main program which equals maximum number of periphery points which may be input around any cross section. Used here for error checks and execution time dimensioning of arrays.

MAXALP Program dimension size variable specified in main program which equals the maximum allowed number of body orientations. Used here for error checks and execution time dimensioning of arrays.

LLL Program dimension size variable specified in main program which equals the maximum possible number of body panels which may be generated by the input. Used here for error checks and execution time dimensioning of arrays.

MAXINF Program dimension size variable specified in main program which equals the maximum number of inlet and outlet panels that may be prescribed over the entire body. Used here for error checks and execution time dimensioning of arrays.

Remaining Input: (Supplied from input card deck.)

SYMBOL Array of alphanumeric characters making up the input heading.

V Magnitude of free stream velocity.

NALPHA Number of body orientations to be run.

ALPHA Array of body angles of attack.

BETA Array of body side slip angles.

NSYMET Symmetric body input option:
= 0 if body symmetry is used.
= 1 if body symmetry is not used.

NSECTO Total number of body cross sections used to define body.

NSEC Cross section sequence number.

NIP Number of input points on the cross section.

NEND Cross section flag signal:
= 0 if cross section is not repeated or not the end of a panel network. Place panels between this section and the next.

= 1 if cross section is repeated or is the end of a panel network. Do not place panels between this section and the next one.

X,Y,Z Coordinates of a cross section periphery point relative to the reference body coordinate system.

NLIST Panel geometry table print option:
= 0 if all body panel geometry is to be printed in table during execution of paneling subroutine.

= 1 if geometry is not to be printed.

NCALC	Geometry check--test run option: = 0 for a normal run to obtain flow solutions. = 1 for test run to obtain only panel geometry.
NINFLO	Number of panels specified as being inlet or outlet panels.
INDEX	Array of panel index numbers to identify body panels.
FLRATO	Array of specified inlet or outlet flow velocity ratios.
NWING	Wing modeling signal variable: = 0 if no wing is modeled on this configuration. = 1 if wing is modeled.
XQR, YQR, ZQR	Coordinates of the wing root quarter chord location relative to reference body coordinate system.
SPAN	Physical wing span.
CHORD	Wing root chord length.
DIHED	Wing dihedral angle.
SWEEP	Sweep angle of wing quarter chord line.
CL	Array of wing lift coefficients.
NPOINT	Propeller plane signal variable: = 0 if no propeller plane is modeled and no propeller plane flow field predictions are made. = 1 if propeller plane is modeled and flow field predictions will be made.
XHUB, YHUB, ZHUB	Coordinates of the hub or center of the propeller plane disk relative to the reference coordinate system.
ALPHP	Angle of attack of propeller plane relative to body longitudinal axis.
BETAP	Yaw angle of propeller plane relative to body longitudinal axis.
RADIUS	Reference scale radius of propeller plane.
DPSI	Azimuth angle spacing between radial rows of points in the propeller plane.

DRADUS	Radial spacing between points in the propeller plane as fraction of reference radius.
NRAD	Number of points positioned radially along the propeller plane.
NPUNCH	Propeller plane flow--punched card output option: = 0 if no punched card output is desired. = 1 if desire punched card output.
ITMAX	Maximum number of iterations allowed for obtaining source strength solutions.
ERR	Power of ten of the value of desired solution error residual.

Output: (This output is printed only but not saved.)

J	Body orientation case number.

Remaining Output: (All input card data is printed, but some of above input values are renamed as new output quantities given below.)

NIPMAX	Same value as defined in input section above unless symmetry option NSYMET = 0 was used. If symmetry was used, this variable takes the value of the maximum number of periphery points allowed to be specified on the left half of any cross section.
NCOUNT	Array whose i th value stores the NIP value associated with the i th input cross section.
NFLAG	Array whose i th value stores the value of NEND associated with the i th input cross section.
PX, PY, PZ	Two dimensional arrays containing coordinates of all points (X, Y, Z) (defined in input above). First array index corresponds to the cross section number. Second array index corresponds to the order in which points are placed around the cross section.

Subroutines Called: None

Error Returns: None

Error Stops: Forty-two detectable errors will cause run termination. In general these errors occur if a value is out of its permissible range. Other errors can be detected if some geometry input rules have been violated.

The detectable errors are given below:

Stop Number	Error Description
1	V ≤ 0
2	NALPHA > MAXALP
3	NALPHA < 1
4	NSYMET ≠ 0 and ≠ 1 is invalid.
5	A BETA angle is nonzero but symmetry option has been specified.
6	NSECTO < 3
7	NSECTO > NSECMA
8	NSEC not in increasing sequence starting with 1.
9	NIP < 3 with symmetry option chosen.
10	NIP < 4 with symmetry option not chosen.
11	NIP > NIPMAX for this section--with symmetry option used.
12	NIP > NIPMAX for this section--with symmetry option not used.
13	NEND = NFLAG(1) = 1 for first section is invalid.
14	NEND ≠ 0 on the first section is invalid.
15	First input point on the cross section is not repeated as last point of section. This is required for non symmetry option.
16	Symmetry option used but first point on section is not on upper centerline of contour.
17	Symmetry option chosen but last point on section is not on lower centerline of contour.
18	Symmetry option chosen but a point on the right side of the body has been input.
19	NEND ≠ 0 and ≠ 1 is invalid.
20	NEND (i.e. NFLAG) = 1 specified on adjacent cross sections--invalid.
21	On two cross sections between which panels are to be generated, the number of points on one section is not the same as number on the other section--invalid.
22	Error in two sections between which panels are to be generated. The cross sections intersect or overlap since a point on one section has been also placed on the other.
23	NLIST ≠ 1 and ≠ 0 is invalid.
24	NCALC ≠ 1 and ≠ 0 is invalid.
25	Test run specified (NCALC = 1) but inflow panels have been specified (NINFLO ≠ 0)--invalid.
26	NINFLO is negative.
27	Symmetric option chosen but NINFLO value is incorrect (NINFLO > MAXINF/2 was used.)
28	Non-symmetry option chosen but NINFLO value is incorrect (NINFLO > MAXINF was used.)

29 INDEX < 1 or > LLL was used and is invalid.
30 NWING $\neq$ 0 and $\neq$ 1 is invalid.
31 Symmetric option was used but wing root is not located on the plane of symmetry. (i.e. YQR $\neq$ 0 was given and is invalid.)
32 SPAN $\leq$ 0 is invalid.
33 CHORD $\leq$ 0 is invalid.
34 NPOINT $\neq$ 0 and $\neq$ 1 is invalid.
35 RADIUS $\leq$ 0 is invalid.
36 DPSI $\leq$ 0 is invalid.
37 DPSI > 360.0 is invalid.
38 DRADUS < 0 is invalid.
39 NRAD $\leq$ 0 is invalid.
40 NRAD > $NRAD_{max}$ is invalid.
41 NPUNCH $\neq$ 0 and $\neq$ 1 is invalid.
42 ITMAX $\leq$ 0 is invalid.

Warning Messages: Two possible warning messages, if printed, do not terminate program but indicate possible problem:

1. Warning due to DRADUS = 0
2. Warning due to ERR > 0 which gives solution error residual greater than 1.0

Subroutine PANEL

Purpose: To generate the body surface panels from the input cross section geometry. Information produced includes panel index numbers, surface areas, outward normal unit vector components, control points, and a total of the number of panels generated.

Method: Rings of Panels are produced around the surface between adjacent body cross sections. The first ring is placed between input sections one and two, etc. The panels are assigned consecutive identifying index numbers starting with the first panel on the first ring and increasing back on the configuration. Within each ring of panels, the top left side panel is produced first and has the smallest index number found on that ring. Then panels are generated and numbered in sequence on the ring around the left side of the body, and up the right side with the last panel of the ring being on the top right side.

Panels generated may be either triangular or four-cornered. A triangular panel is made when an input cross section point has been repeated. Otherwise, the panel will have four different corner points. The subroutine checks that a panel of zero area has not been input. This would happen if corresponding points on both cross sections were both repeated. Such errors cause complete program termination. The logic used by this subroutine for triangular panels differs from that used with four-cornered panels.

With triangular panels, the input corner points are coplanar and need not be shifted to produce a flat panel. The outward normal unit vector for this panel is obtained by taking the vector product of the two adjacent sides of the triangle. Then surface area is computed using the formula for the area of a triangle. Lastly, the control point is located at the area centroid of the triangular panel.

Logic required with four-cornered panels is more involved since the input corner points are usually not coplanar. First the unit normal vector is found by taking the vector product of the diagonals between opposite corner points. Now the input corner points are shifted to make them coplanar. This is done by projecting the input points onto a plane which contains the average of the input corner points and which is orthogonal to the unit normal vector. The new projected points are then coplanar and define a flat quardilateral panel. Now the surface area is obtained using using the area formula for any general plane quadrilateral. Lastly, the control point is placed at the area centroid of the quadrilateral panel.

If the symmetric body input option, NSYMET = 0, is used, only the left side of each cross section was input. However, the subroutine computes the right side body panels automatically using symmetry and follows the same panel numbering sequence as would exist if the geometry had been input without using the symmetry option.

For the symmetric case only, a bookkeeping system array is generated which identifies a panel as being on the left side or on the right side of the body. This bookkeeping array is not used if the non-symmetric input option, NSYMET = 1, has been used.

At the end of subroutine execution, the total number of panels generated, NP, is stored and printed out.

Finally, this subroutine will print a detailed panel geometry listing if requested by the, NLIST = 0, parameter. This listing is automatically printed if the test run option, NCALC = 1, has been input. This listing identifies all panels of the configuration. The panel index number, surface area, control point, and input corner points are listed. Additionally, for symmetric configurations, a message denotes panels which are on the right side of the body and states the index number of the corresponding left side panels.

Use: CALL PANEL(NIPMAX,NSECMA,NSECTO,NFLAG,NP,NCOUNT,LLL,PX,PY,PZ,STOX1,STOY1,STOZ1,STOX2,STOY2,STOZ2,S,ANVX,ANVY,ANVZ,XC,YC,ZC)

Common Blocks Required: SYMTRE and OPTION

Input: (All quantities are relative to the reference body coordinate system.)

NSECMA Program dimension size variable specified in main program which is the maximum number of body cross sections which may be used to define the configuration. Used here for execution time dimensioning of arrays.

NIPMAX Program dimension size variable specified in main program which equals the maximum number of points that may be used to define any body cross section. Used here for execution time dimensioning of arrays.

C - 2

and define a flat quardilateral panel. Now the surface area is obtained using using the area formula for any general plane quadrilateral. Lastly, the control point is placed at the area centroid of the quadrilateral panel.

If the symmetric body input option, NSYMET = 0, is used, only the left side of each cross section was input. However, the subroutine computes the right side body panels automatically using symmetry and follows the same panel numbering sequence as would exist if the geometry had been input without using the symmetry option.

For the symmetric case only, a bookkeeping system array is generated which identifies a panel as being on the left side or on the right side of the body. This bookkeeping array is not used if the non-symmetric input option, NSYMET = 1, has been used.

At the end of subroutine execution, the total number of panels generated, NP, is stored and printed out.

Finally, this subroutine will print a detailed panel geometry listing if requested by the, NLIST = 0, parameter. This listing is automatically printed if the test run option, NCALC = 1, has been input. This listing identifies all panels of the configuration. The panel index number, surface area, control point, and input corner points are listed. Additionally, for symmetric configurations, a message denotes panels which are on the right side of the body and states the index number of the corresponding left side panels.

Use: CALL PANEL(NIPMAX,NSECMA,NSECTO,NFLAG,NP,NCOUNT,LLL,PX,PY,PZ,STOX1,STOY1,STOZ1,STOX2,STOY2,STOZ2,S,ANVX,ANVY,ANVZ,XC,YC,ZC)

Common Blocks Required: SYMTRE and OPTION

Input: (All quantities are relative to the reference body coordinate system.)

NSECMA — Program dimension size variable specified in main program which is the maximum number of body cross sections which may be used to define the configuration. Used here for execution time dimensioning of arrays.

NIPMAX — Program dimension size variable specified in main program which equals the maximum number of points that may be used to define any body cross section. Used here for execution time dimensioning of arrays.

C-2

LLL	Program dimension size variable specified in main program which equals the maximum number of panels which may possibly be generated on any configuration. Used here for execution time dimensioning of arrays.
NSYMET	Symmetric input option variable: = 0 if body symmetric input option used. = 1 if body symmetric input not used.
NCALC	Test run option: = 0 if run is normal run for flow calculations. = 1 if test run is made for panel geometry only.
NLIST	Panel geometry print option: = 0 if desire printing of geometry table. = 1 if do not want geometry table printed.
NSECTO	Total number of body cross sections input on this configuration.
NFLAG	Array of input signal variables assigned to each input cross section. I th value is associated with the I th cross section: = 0 if cross section is not end of body panel network, or if this section is not repeated. = 1 if cross section is end of network, or if this section is repeated.
NCOUNT	Array whose I th value equals total number of input points specified on the I th cross section. This is array of NIP values for each section.
STOX1, STOY1, STOZ1	Intermediate storage arrays containing coordinates of input points to the first of two cross sections between which a ring of panels is being generated.
STOX2, STOY2, STOZ2	Intermediate storage arrays containing coordinates of input points to the second of two cross sections between which a ring of panels is being generated.
PX, PY, PZ	Two dimensional arrays containing original input cross section points. First array index equals cross section number of point, second index gives the sequence number of the point around its cross section.

Output: (All coordinates and components relative to reference body coordinate system.)

NPNSYM — Bookkeeping array established only if NSYMET = 0 to record right side panels:

if NPSYM(I) = 0, body panel number I is an image panel on the right side of the symmetric body.

if NPNSYM(I) = ($N \neq 0$), body panel I is on the left side of the body and its right side image panel is number N.

ANVX, ANVY, ANVZ — Arrays of components of the outward drawn unit normal vector of each panel. I th value is for I th panel.

S — Array of body panel surface areas. I th value is area of I th panel.

XC, YC, ZC — Arrays of coordinates of body panel control points. I th value is for I th panel.

I — Do loop variable which equals index number of panel currently under consideration. Used only for printing geometry. Value not saved or passed to remainder or program.

STOX1, STOY1, STOZ1, STOX2, STOY2, STOZ2 — Same as described in input. These original input corner points are printed on the output geometry table. The values are not saved after printing or after subroutine has completed execution.

PRX1, PRY1, PRZ1 — Temporary printing output variables containing coordinates of the first corner point of a right side image panel of a symmetric configuration. Not saved after printing on geometry table.

PRX2, PRY2, PRZ2 — Same as above but second corner point of panel.

PRX3, PRY3, PRZ3 — Same as above but third corner point of panel.

<u>Output</u>: (All coordinates and components relative to reference body coordinate system.)

NPNSYM	Bookkeeping array established only if NSYMET = 0 to record right side panels: if NPSYM(I) = 0, body panel number I is an image panel on the right side of the symmetric body. if NPNSYM(I) = (N $\neq$ 0), body panel I is on the left side of the body and its right side image panel is number N.
ANVX, ANVY, ANVZ	Arrays of components of the outward drawn unit normal vector of each panel. I th value is for I th panel.
S	Array of body panel surface areas. I th value is area of I th panel.
XC, YC, ZC	Arrays of coordinates of body panel control points. I th value is for I th panel.
I	Do loop variable which equals index number of panel currently under consideration. Used only for printing geometry. Value not saved or passed to remainder or program.
STOX1, STOY1, STOZ1, STOX2, STOY2, STOZ2	Same as described in input. These original input corner points are printed on the output geometry table. The values are not saved after printing or after subroutine has completed execution.
PRX1, PRY1, PRZ1	Temporary printing output variables containing coordinates of the first corner point of a right side image panel of a symmetric configuration. Not saved after printing on geometry table.
PRX2, PRY2, PRZ2	Same as above but second corner point of panel.
PRX3, PRY3, PRZ3	Same as above but third corner point of panel.

PRX4, PRY4, PRZ4	Same as above but the fourth corner point of panel.
NP	Total number of body panels generated by subroutine from user input cross section geometry.
PX, PY, PZ	Two dimensional arrays containing all corner points of panels. Values in these output arrays are the modified values of input corner points if points are on quadrilateral panels. (Note these values are not original input corner points and are not printed on geometry table.)
STOPER	Error signal = 0.0 if no fatal paneling errors found. = 1.0 if no one or more fatal paneling errors were found. See error stops below.

Subroutines Called: None

Error Returns: None

Error Stops: If fatal error is detected, STOPER is set = 1.0. Subroutine continues generating panels. After all panel geometry has been handled, and if STOPER = 1.0, the subroutine prints fatal error message and number of panels. Subroutine aborts entire program.

Types of fatal errors detected are one of three:

1. Zero area panels have been generated due to repeat of both corresponding points on adjacent cross sections.

2. Roundoff error in a very small panel has led to negative panel area or other impossible numerical value.

3. After modification of four-cornered panel corner points, the resulting corner points are on a line or so close as to produce a quadrilateral having zero or imaginary surface area. This is a roundoff error also.

PRX4, PRY4, PRZ4	Same as above but the fourth corner point of panel.
NP	Total number of body panels generated by subroutine from user input cross section geometry.
PX, PY, PZ	Two dimensional arrays containing all corner points of panels. Values in these output arrays are the modified values of input corner points if points are on quadrilateral panels. (Note these values are not original input corner points and are not printed on geometry table.)
STOPER	Error signal = 0.0 if no fatal paneling errors found. = 1.0 if no one or more fatal paneling errors were found. See error stops below.

Subroutines Called: None

Error Returns: None

Error Stops: If fatal error is detected, STOPER is set = 1.0. Subroutine continues generating panels. After all panel geometry has been handled, and if STOPER = 1.0, the subroutine prints fatal error message and number of panels. Subroutine aborts entire program.

Types of fatal errors detected are one of three:

1. Zero area panels have been generated due to repeat of both corresponding points on adjacent cross sections.
2. Roundoff error in a very small panel has led to negative panel area or other impossible numerical value.
3. After modification of four-cornered panel corner points, the resulting corner points are on a line or so close as to produce a quadrilateral having zero or imaginary surface area. This is a roundoff error also.

Subroutine SOLSYM

Purpose: To solve the half size system of simultaneous linear equations for the unknown body panel source strengths. System was generated using left side panels as the body geometry has been input as a symmetric body.

Method: This subroutine is called only if the symmetric body input option was used. The method of Gauss-Seidel iteration is used. The system of equations is read from auxiliary file 9.

The unknown source strengths on the left side panels are solved, and sets of unknowns for each body orientation are obtained. During the solution process, all input body orientations are being solved together during one execution of the subroutine.

Solution progress is printed at the end of each iteration. Residual solution error is indicated and a message indicates if the solution set for a body orientation has converged.

Iterations continue until convergence is achieved for all separate sets of source strength unknowns. Should one or more sets of unknowns fail to converge within the allotted amount of iterations, the subroutine stops the program and all solutions are lost.

After the source strengths have been solved for the left side body panels, the proper value of source strength is assigned to each mirror image panel on the right side of the body. This is done using a panel bookkeeping array which keeps track of which panels are on the left and which are on the right side of the body.

The solution array contains sets of source strengths, one set for each body orientation. Each set has one source strength for each panel on the body.

Use: CALL SOLSYM(LLL,LLLHAF,MD,MAXALP,NP,A,BIG,VARIAB,SUM, ERR,ITMAX,SIGMA,SIGSAV)

Common Blocks Required: INPUTS and SYMTRE

Subroutine SOLSYM

Purpose: To solve the half size system of simultaneous linear equations for the unknown body panel source strengths. System was generated using left side panels as the body geometry has been input as a symmetric body.

Method: This subroutine is called only if the symmetric body input option was used. The method of Gauss-Seidel iteration is used. The system of equations is read from auxiliary file 9.

The unknown source strengths on the left side panels are solved, and sets of unknowns for each body orientation are obtained. During the solution process, all input body orientations are being solved together during one execution of the subroutine.

Solution progress is printed at the end of each iteration. Residual solution error is indicated and a message indicates if the solution set for a body orientation has converged.

Iterations continue until convergence is achieved for all separate sets of source strength unknowns. Should one or more sets of unknowns fail to converge within the allotted amount of iterations, the subroutine stops the program and all solutions are lost.

After the source strengths have been solved for the left side body panels, the proper value of source strength is assigned to each mirror image panel on the right side of the body. This is done using a panel bookkeeping array which keeps track of which panels are on the left and which are on the right side of the body.

The solution array contains sets of source strengths, one set for each body orientation. Each set has one source strength for each panel on the body.

Use: CALL SOLSYM(LLL,LLLHAF,MD,MAXALP,NP,A,BIG,VARIAB,SUM, ERR,ITMAX,SIGMA,SIGSAV)

Common Blocks Required: INPUTS and SYMTRE

Input:

LLL	Program dimension specified in main program which is maximum allowed number of body panels. Used here for execution time dimensioning of arrays.
LLLHAF	Program dimension specified in main program equal to half the maximum allowed number of body panels. Used here for execution time dimensioning of arrays.
MAXALP	Program dimension specified in main program equal to maximum number of body orientations allowed. Used for execution time dimensioning of arrays.
MD	Program dimension specified in main program which is sum of LLL and MAXALP. Used for execution time dimensioning of arrays.
NP	Total number of body surface panels generated over the entire configuration. Is twice the number of equations being solved.
ERR	Power of ten of the desired solution error.
ITMAX	Maximum iterations allowed in which to obtain solution convergence.
NALPHA	Number of body orientations (sets of unknowns).
SUM	Array of intermediate sums used in solution process.
A	Buffer array containing left and right side coefficients of one equation in the system. Contents taken from one logical record of file 9.
VARIAB	Vector containing intermediate source strength solution value from each set of unknowns. Used to test for solution convergence.
INSOLV	Array containing the index numbers of the body panels on the left side of the configuration as generated by the paneling subroutine.

Input:

LLL — Program dimension specified in main program which is maximum allowed number of body panels. Used here for execution time dimensioning of arrays.

LLLHAF — Program dimension specified in main program equal to half the maximum allowed number of body panels. Used here for execution time dimensioning of arrays.

MAXALP — Program dimension specified in main program equal to maximum number of body orientations allowed. Used for execution time dimensioning of arrays.

[illegible] — Program dimension specified in main program which is sum of LLL and MAXALP. Used for execution time dimensioning of arrays.

NP — Total number of body surface panels generated over the entire configuration. Is twice the number of equations being solved.

ERR — Power of ten of the desired solution error.

ITMAX — Maximum iterations allowed in which to obtain solution convergence.

NALPHA — Number of body orientations (sets of unknowns).

SUM — Array of intermediate sums used in solution process.

A — Buffer array containing left and right side coefficients of one equation in the system. Contents taken from one logical record of file 9.

VARIAB — Vector containing intermediate source strength solution value from each set of unknowns. Used to test for solution convergence.

INSOLV — Array containing the index numbers of the body panels on the left side of the configuration as generated by the paneling subroutine.

NPNSYM Bookkeeping array which matches index number of a left side body panel with the index number of its corresponding right side panel:

if NPNSYM(I) = 0, the body panel whose index number is I is a right side panel.

if NPNSYM(I) = ($N \neq 0$), the panel whose index is I is a left side panel and its right side image panel is panel number N.

<u>Output</u>: (This output is also printed.)

ITMAX Same as quantity defined in input section above.

ERROR Specified maximum permitted solution residual error:

$ERROR = 10.0^{ERR}$

ITER Number of iteration being done.

K Body orientation case number corresponding to a set of source strength solutions being found (This variable is not saved after printing.)

ALPHA Array of input body angles of attack, each value corresponding to an orientation case for which a set of solutions is being found.

BETA Array of input body sideslip angles, each value corresponding to an orientation case for which a set of solutions is being found.

<u>Other Output</u>: (This output not printed by this subroutine.)

BIG Vector containing the maximum error residual obtained for each set of unknowns during one iteration.

SIGSAV Bookkeeping vector containing only one set of solutions for left side panels only. Temporary storage for the unknowns until source strengths are assigned to the right side panels.

SIGMA Array containing final NALPHA sets of source strength solutions for all NP body panels in the configuration.

NSIGNL Convergence point signal:
= 0 if all solution sets have converged.
= 1 if one or more sets of solutions have failed to converge this iteration (See below).

NPNSYM	Bookkeeping array which matches index number of a left side body panel with the index number of its corresponding right side panel:
	if NPNSYM(I) = 0, the body panel whose index number is I is a right side panel.
	if NPNSYM(I) = (N $\neq$ 0), the panel whose index is I is a left side panel and its right side image panel is panel number N.

<u>Output</u>: (This output is also printed.)

ITMAX	Same as quantity defined in input section above.
ERROR	Specified maximum permitted solution residual error:
	ERROR = 10.0^{ERR}
ITER	Number of iteration being done.
K	Body orientation case number corresponding to a set of source strength solutions being found (This variable is not saved after printing.)
ALPHA	Array of input body angles of attack, each value corresponding to an orientation case for which a set of solutions is being found.
BETA	Array of input body sideslip angles, each value corresponding to an orientation case for which a set of solutions is being found.

<u>Other Output</u>: (This output not printed by this subroutine.)

BIG	Vector containing the maximum error residual obtained for each set of unknowns during one iteration.
SIGSAV	Bookkeeping vector containing only one set of solutions for left side panels only. Temporary storage for the unknowns until source strengths are assigned to the right side panels.
SIGMA	Array containing final NALPHA sets of source strength solutions for all NP body panels in the configuration.
NSIGNL	Convergence print signal: = 0 if all solution sets have converged. = 1 if one or more sets of solutions have failed to converge this iteration (See below).

Subroutines Called:	None
Error Returns:	None
Error Stops:	If one or more solution sets fail to converge after ITMAX iterations (i.e. NSIGNL = 1), print solution failure message. Subroutine stops entire program.

Subroutines Called:	None
Error Returns:	None
Error Stops:	If one or more solution sets fail to converge after ITMAX iterations (i.e. NSIGNL = 1), print solution failure message. Subroutine stops entire program.

Subroutine SOLVE

Purpose: To solve the full size system of linear equations for the unknown body panel source strengths. System was generated using all body panels because body geometry was input without using the symmetry option.

Method: This subroutine is called only if the body has been input as a nonsymmetric configuration. The method of Gauss-Seidel iteration is used. The system of equations is read from auxiliary file 9.

The unknown source strengths for all panels are solved, and sets of unknowns for each body orientation are obtained. During the solution process, all input body orientations are being solved together during one execution of the subroutine.

Solution progress is printed at the end of each iteration. Residual solution error is indicated and a message indicates if the solution set for a body orientation has converged.

Iterations continue until convergence is achieved for all separate sets of source strength unknowns. Should one or more sets of unknowns fail to converge within the allotted amount of itérations, the subroutine stops the program and all solutions are lost.

If all solutions converge, the solution array is returned. It contains sets of source strengths, one set for each body orientation. Each set has one source strength for each panel on the body.

Use: CALL SOLVE(LLL,MD,MAXALP,NP,A,BIG,VARIAB,SUM,ERR, ITMAX,SIGMA)

Common Blocks Required: INPUTS

Input:

LLL Program dimension variable specified in main program which equals maximum allowed number of body panels. Used here for execution time dimensioning of arrays.

Subroutine SOLVE

Purpose: To solve the full size system of linear equations for the unknown body panel source strengths. System was generated using all body panels because body geometry was input without using the symmetry option.

Method: This subroutine is called only if the body has been input as a nonsymmetric configuration. The method of Gauss-Seidel iteration is used. The system of equations is read from auxiliary file 9.

The unknown source strengths for all panels are solved, and sets of unknowns for each body orientation are obtained. During the solution process, all input body orientations are being solved together during one execution of the subroutine.

Solution progress is printed at the end of each iteration. Residual solution error is indicated and a message indicates if the solution set for a body orientation has converged.

Iterations continue until convergence is achieved for all separate sets of source strength unknowns. Should one or more sets of unknowns fail to converge within the allotted amount of iterations, the subroutine stops the program and all solutions are lost.

If all solutions converge, the solution array is returned. It contains sets of source strengths, one set for each body orientation. Each set has one source strength for each panel on the body.

Use: CALL SOLVE(LLL,MD,MAXALP,NP,A,BIG,VARIAB,SUM,ERR,ITMAX,SIGMA)

Common Blocks Required: INPUTS

Input:

LLL Program dimension variable specified in main program which equals maximum allowed number of body panels. Used here for execution time dimensioning of arrays.

MAXALP Program dimension variable specified in main program which equals the maximum number of body orientation cases allowed. Used here for execution time dimensioning of arrays.

MD Program dimension variable specified in main program which is the sum of LLL and MAXALP. Used here for execution time dimensioning of arrays.

NP Total number of body panels in the configuration which equals the number of equations in the system.

NALPHA Number of body orientation cases (sets of unknowns).

ERR Power of ten of the desired solution error residual.

ITMAX Maximum iterations allowed in which to obtain solution convergence.

SUM Array of intermediate sums used in solution process.

A Buffer array containing left and right side coefficients of one equation in the system. Contents read from one logical record of file 9.

VARIAB Vector containing intermediate source strength solution values from each set of unknowns. Used to check solution convergence.

Output: (The following output is also printed out.)

ITMAX Same as quantity defined in input section above.

ERROR Specified maximum permitted solution residual error:

$ERROR = 10.0^{ERR}$

ITER Number of iteration being done.

K Body orientation case number corresponding to a set of source strength solutions being found (This variable is not saved after printing.)

MAXALP	Program dimension variable specified in main program which equals the maximum number of body orientation cases allowed. Used here for execution time dimensioning of arrays.
MD	Program dimension variable specified in main program which is the sum of LLL and MAXALP. Used here for execution time dimensioning of arrays.
NP	Total number of body panels in the configuration which equals the number of equations in the system.
NALPHA	Number of body orientation cases (sets of unknowns).
ERR	Power of ten of the desired solution error residual.
ITMAX	Maximum iterations allowed in which to obtain solution convergence.
SUM	Array of intermediate sums used in solution process.
A	Buffer array containing left and right side coefficients of one equation in the system. Contents read from one logical record of file 9.
VARIAB	Vector containing intermediate source strength solution values from each set of unknowns. Used to check solution convergence.

<u>Output</u>: (The following output is also printed out.)

ITMAX	Same as quantity defined in input section above.
ERROR	Specified maximum permitted solution residual error: $ERROR = 10.0^{ERR}$
ITER	Number of iteration being done.
K	Body orientation case number corresponding to a set of source strength solutions being found (This variable is not saved after printing.)

ALPHA Array of input body angles of attack, each value corresponding to an orientation case for which a set of solutions is being found.

BETA Array of input body sideslip angles, each value corresponding to an orientation case for which a set of solutions is being found.

BIG Vector containing the maximum error residual obtained for each set of unknowns during one iteration.

Other Output: (The following is not printed by this subroutine.)

SIGMA Array containing final NALPHA sets of source strength solutions for all NP body panels in the configuration.

NSIGNL Convergence print signal:
= 0 if all solution sets have converged
= 1 if one or more sets of solutions have failed to converge this iteration (see below).

Subroutines Called: None

Error Returns: None

Error Stops: If one or more solution sets fail to converge after ITMAX iterations (i.e. NSIGNL = 1), print solution failure message. Subroutine stops entire program.

ALPHA	Array of input body angles of attack, each value corresponding to an orientation case for which a set of solutions is being found.
BETA	Array of input body sideslip angles, each value corresponding to an orientation case for which a set of solutions is being found.
BIG	Vector containing the maximum error residual obtained for each set of unknowns during one iteration.

<u>Other Output</u>: (The following is not printed by this subroutine.)

SIGMA	Array containing final NALPHA sets of source strength solutions for all NP body panels in the configuration.
NSIGNL	Convergence print signal: = 0 if all solution sets have converged = 1 if one or more sets of solutions have failed to converge this iteration (see below).

Subroutines Called: None

Error Returns: None

Error Stops: If one or more solution sets fail to converge after ITMAX iterations (i.e. NSIGNL = 1), print solution failure message. Subroutine stops entire program.

Subroutine VCOMP

Purpose: To calculate sets of components of free stream velocity relative to the reference body coordinate system. Each set of components corresponds to a separate body orientation.

Method: Makes a conversion from the wind axis system (aligned with the free stream velocity direction) to the reference body cartesian coordinate system. the euler angle for the transformation is obtained using subroutine EULER. Reference 1 contains the transformation formulas.

Use: CALL VCOMP

Common Block Required: INPUTS

Input:

NALPHA	Number of body orientations for which sets of velocity components are to be found.
ANADA	Euler angle required in axis transformation (obtained from subroutine EULER).
ALPHA	Array of body angles of attack, each value corresponding to a body orientation case.
BETA	Array of sideslip angles, each value corresponding to a body orientation case.

Output:

VX, VY, VZ	Arrays of free stream velocity components in the reference body coordinate system. Arrays contain sets of components, each associated with one body orientation.

Subroutine Called: EULER

Error Returns: None

Error Stops: None

Subroutine VCOMP

Purpose: To calculate sets of components of free stream velocity relative to the reference body coordinate system. Each set of components corresponds to a separate body orientation.

Method: Makes a conversion from the wind axis system (aligned with the free stream velocity direction) to the reference body cartesian coordinate system. the euler angle for the transformation is obtained using subroutine EULER. Reference 1 contains the transformation formulas.

Use: CALL VCOMP

Common Block Required: INPUTS

Input:

NALPHA	Number of body orientations for which sets of velocity components are to be found.
ANADA	Euler angle required in axis transformation (obtained from subroutine EULER).
ALPHA	Array of body angles of attack, each value corresponding to a body orientation case.
BETA	Array of sideslip angles, each value corresponding to a body orientation case.

Output:

VX, VY, VZ	Arrays of free stream velocity components in the reference body coordinate system. Arrays contain sets of components, each associated with one body orientation.

Subroutine Called: EULER

Error Returns: None

Error Stops: None

Subroutine VELOCI

Purpose: To calculate and print out the body surface velocities and pressure coefficients.

Method: This subroutine is called for each body orientation case. One body orientation is being evaluated at a time.

The flow velocity on the surface of each body panel is calculated. The control point is the representative location of the panel.

At panel point I under consideration, the velocity components induced by panel I are first obtained. To these components are added all the velocities induced at panel point I by the sources at all other body panels J. Then if a wing is present, the induced velocity components induced at panel I by the wing are added. Finally, the components of the free stream velocity are added. The resulting sums contain the x, y, and z components of velocity on the surface of panel I at its control point.

Using the velocity components above, the pressure coefficient at panel point I is obtained.

The velocity and pressure results at panel I are printed on the output table along with the panel index number and control point coordinates. Also are printed the body panel source strength solutions.

The above process is repeated until results at every body panel have been obtained.

The velocity components and pressure coefficients are not saved after they have been printed.

See reference 1 for a description of the formulas for velocity induced by sources and vortices.

Use: CALL VELOCI(LLL,MD,MAXALP,ICASE,NP,ANVX,ANVY,ANVZ,S, SIGMA,XC,YC,ZC)

Common Blocks Required: INPUTS, INLET, and WING1

Input:

LLL Program dimension size variable specified in main program which equals the maximum allowed number of body panels which may be generated. Used here for execution time dimensioning of arrays.

Subroutine VELOCI

Purpose: To calculate and print out the body surface velocities and pressure coefficients.

Method: This subroutine is called for each body orientation case. One body orientation is being evaluated at a time.

The flow velocity on the surface of each body panel is calculated. The control point is the representative location of the panel.

At panel point I under consideration, the velocity components induced by panel I are first obtained. To these components are added all the velocities induced at panel point I by the sources at all other body panels J. Then if a wing is present, the induced velocity components induced at panel I by the wing are added. Finally, the components of the free stream velocity are added. The resulting sums contain the x, y, and z components of velocity on the surface of panel I at its control point.

Using the velocity components above, the pressure coefficient at panel point I is obtained.

The velocity and pressure results at panel I are printed on the output table along with the panel index number and control point coordinates. Also are printed the body panel source strength solutions.

The above process is repeated until results at every body panel have been obtained.

The velocity components and pressure coefficients are not saved after they have been printed.

See reference 1 for a description of the formulas for velocity induced by sources and vortices.

Use: CALL VELOCI(LLL,MD,MAXALP,ICASE,NP,ANVX,ANVY,ANVZ,S, SIGMA,XC,YC,ZC)

Common Blocks Required: INPUTS, INLET, and WING1

Input:

LLL — Program dimension size variable specified in main program which equals the maximum allowed number of body panels which may be generated. Used here for execution time dimensioning of arrays.

MAXALP	Program dimension size variable specified in main program which equals the maximum allowed number of body orientations. Used here for execution time dimensioning of arrays.
MD	Program dimension size variable specified in main program which is sum of LLL and MAXALP. Used here for execution time dimensioning of arrays.
ICASE	Body orientation case being considered during this call of the subroutine. Value is also printed out.
ALPHA	Array of body angles of attack. The ICASE th value in the array is used by subroutine. This value is also printed out.
BETA	Array of body sideslip angles. The ICASE th value in the array is used by subroutine. This value is also printed out.
V	Magnitude of free stream velocity. Value is also printed out.
VX, VY, VZ	Arrays of components of free stream velocity relative to the reference body coordinate system. The ICASE th value of each array is used and printed out.
NP	Total number of body panels on this configuration. This is also printed out.
XC, YC, ZC	Arrays of coordinates of body panel control points. These values appear on the print out also. Values relative to reference body coordinate system.
S	Array of body panel surface areas.
SIGMA	Array of body panel source strengths. The ICASE th set of the strengths is being used. These are the solutions of the system of equations and they are printed out on the output table.
ANVX, ANVY, ANVZ	Arrays of components of outward drawn normal unit vectors acting at the panel control points. Components are relative to reference body coordinate system.

MAXALP	Program dimension size variable specified in main program which equals the maximum allowed number of body orientations. Used here for execution time dimensioning of arrays.
MD	Program dimension size variable specified in main program which is sum of LLL and MAXALP. Used here for execution time dimensioning of arrays.
ICASE	Body orientation case being considered during this call of the subroutine. Value is also printed out.
ALPHA	Array of body angles of attack. The ICASE th value in the array is used by subroutine. This value is also printed out.
BETA	Array of body sideslip angles. The ICASE th value in the array is used by subroutine. This value is also printed out.
V	Magnitude of free stream velocity. Value is also printed out.
VX, VY, VZ	Arrays of components of free stream velocity relative to the reference body coordinate system. The ICASE th value of each array is used and printed out.
NP	Total number of body panels on this configuration. This is also printed out.
XC, YC, ZC	Arrays of coordinates of body panel control points. These values appear on the print out also. Values relative to reference body coordinate system.
S	Array of body panel surface areas.
SIGMA	Array of body panel source strengths. The ICASE th set of the strengths is being used. These are the solutions of the system of equations and they are printed out on the output table.
ANVX, ANVY, ANVZ	Arrays of components of outward drawn normal unit vectors acting at the panel control points. Components are relative to reference body coordinate system.

NWING	Wing signal variable = 0 if no wing is modeled on this configuration. = 1 if wing is modeled.

<u>Output</u>: (In addition to the input values above which were printed, all the following are printed but not saved after printing.)

VRX, VRY, VRZ	Normalized surface velocity components at a panel control point. Values have been made nondimensional by dividing by free stream velocity magnitude, V. The components are relative to the reference body coordinate system.
VR	Total magnitude of the nondimensional velocity at the panel control point.
CP	Pressure coefficient at panel control point.

Subroutine Called: WINGV (Called only if wing is modeled on configuration.)

Error Returns: None

Error Stops: None

NWING	Wing signal variable = 0 if no wing is modeled on this configuration. = 1 if wing is modeled.

Output: (In addition to the input values above which were printed, all the following are printed but not saved after printing.)

VRX, VRY, VRZ	Normalized surface velocity components at a panel control point. Values have been made nondimensional by dividing by free stream velocity magnitude, V. The components are relative to the reference body coordinate system.
VR	Total magnitude of the nondimensional velocity at the panel control point.
CP	Pressure coefficient at panel control point.

Subroutine Called: WINGV (Called only if wing is modeled on configuration.)

Error Returns: None

Error Stops: None

Subroutine VORTEX

Purpose:	To calculate the x, y, and z components of velocity at point (x,y,z) induced by a finite length vortex filament of constant strength.
Method:	Given the starting and ending points of the filament, in the vector sense, the Biot-Savart law for a straight vortex filament is used to compute the three velocity components at the desired location. If the point lies on the filament, the velocity components returned are set to zero to prevent numerical difficulties inherent in the Biot-Savart law. A maximum limit on the induced velocity magnitude is used which equals 20 percent of the free stream velocity. If the velocity induced at the point, using the Biot-Savart law, is more than the maximum limit, the induced magnitude is set equal to 20 percent of the free stream velocity. The velocity components returned are adjusted accordingly.
Use:	CALL VORTEX(X,Y,Z,X1,Y1,Z1,X2,Y2,Z2,GAMMA,VVX,VVY,VVZ)
Common Block Required:	INPUTS

Note: All coordinates and velocity components below are relative to the reference body coordinate system.

Input:

X1,Y1, Z1	Coordinates of the starting end of the vortex filament.
X2,Y2, Z2	Coordinates of the other endpoint of the vortex filament. (Sense of the circulation is from the starting end to this end point.)
X, Y, Z	Coordinates of the point at which induced velocity is to be calculated.
GAMMA	Strength of the vortex filament.
V	Magnitude of the free stream velocity.

Subroutine VORTEX

Purpose: To calculate the x, y, and z components of velocity at point (x,y,z) induced by a finite length vortex filament of constant strength.

Method: Given the starting and ending points of the filament, in the vector sense, the Biot-Savart law for a straight vortex filament is used to compute the three velocity components at the desired location.

If the point lies on the filament, the velocity components returned are set to zero to prevent numerical difficulties inherent in the Biot-Savart law.

A maximum limit on the induced velocity magnitude is used which equals 20 percent of the free stream velocity. If the velocity induced at the point, using the Biot-Savart law, is more than the maximum limit, the induced magnitude is set equal to 20 percent of the free stream velocity. The velocity components returned are adjusted accordingly.

Use: CALL VORTEX(X,Y,Z,X1,Y1,Z1,X2,Y2,Z2,GAMMA,VVX,VVY,VVZ)

Common Block Required: INPUTS

Note: All coordinates and velocity components below are relative to the reference body coordinate system.

Input:

X1,Y1,Z1 — Coordinates of the starting end of the vortex filament.

X2,Y2,Z2 — Coordinates of the other endpoint of the vortex filament. (Sense of the circulation is from the starting end to this end point.)

X, Y, Z — Coordinates of the point at which induced velocity is to be calculated.

GAMMA — Strength of the vortex filament.

V — Magnitude of the free stream velocity.

<u>Output</u>:

VVX, VVY, VVZ	Three vortex induced velocity components at the point.

Subroutines Called: None

Error Returns: None

Error Stops: None

<u>Output</u>:

VVX, VVY, VVZ	Three vortex induced velocity components at the point.

Subroutines Called: None

Error Returns: None

Error Stops: None

Subroutine VPROPS

Purpose: To calculate the flow field, velocity components and flow angles, at all points in the propeller plane.

Method: Subroutine returns control immediately to main program if no propeller plane has been input, NPOINT = 0.

Otherwise, during one call of this subroutine, the propeller flow field at one input body orientation is found.

Point locations in the propeller plane are defined in the propeller plane cylindrical coordinate system according to the azimuthal and radial spacings specified by the input data.

For each point, in turn, the following is performed. The point's position in terms of the reference body cartesian coordinate system is computed. The point's radial position is nondimensionalized by dividing by the reference propeller plane radius value obtained from common block PROP.

Next, the velocity components are found in terms of the reference body coordinate system. The velocity component contributions induced at the point by the body panel sources are summed. Then, if a wing is present, the velocity induced at the point by the wing horseshoe vortex is added. Finally, the free stream velocity components are added. The total flow velocity components and magnitude at the point are normalized by dividing by the free stream velocity magnitude.

The point location in both coordinate systems and normalized velocity components in the reference body coordinate system are printed out in propeller output flow field table part 1 for this body orientation.

Next, the reference cartesian velocity components are transformed to normalized axial, radial, and tangential components in the propeller plane cylindrical axis system. Using the velocity components in both coordinate systems, the upwash and side wash angles at the point are found. Also the rotational flow and outflow angles, as defined in reference 1 and Figure 4.1 of this manual, are calculated.

The point's dimensionless radial and azimuthal position, axial, tangential, and radial velocity components, as well as all flow angles are written to one record of scratch file 30. This enables printing these quantities in a second table later during this subroutine execution.

Subroutine VPROPS

Purpose: To calculate the flow field, velocity components and flow angles, at all points in the propeller plane.

Method: Subroutine returns control immediately to main program if no propeller plane has been input, NPOINT = 0.

Otherwise, during one call of this subroutine, the propeller flow field at one input body orientation is found.

Po[illegible]cations in the propeller plane are defined in [illegible]ller plane cylindrical coordinate system [illegible]g to the azimuthal and radial spacings specified by the input data.

For each point, in turn, the following is performed. The point's position in terms of the reference body cartesian coordinate system is computed. The point's radial position is nondimensionalized by dividing by the reference propeller plane radius value obtained from common block PROP.

Next, the velocity components are found in terms of the reference body coordinate system. The velocity component contributions induced at the point by the body panel sources are summed. Then, if a wing is present, the velocity induced at the point by the wing horseshoe vortex is added. Finally, the free stream velocity components are added. The total flow velocity components and magnitude at the point are normalized by dividing by the free stream velocity magnitude.

The point location in both coordinate systems and normalized velocity components in the reference body coordinate system are printed out in propeller output flow field table part 1 for this body orientation.

Next, the reference cartesian velocity components are transformed to normalized axial, radial, and tangential components in the propeller plane cylindrical axis system. Using the velocity components in both coordinate systems, the upwash and side wash angles at the point are found. Also the rotational flow and outflow angles, as defined in reference 1 and Figure 4.1 of this manual, are calculated.

The point's dimensionless radial and azimuthal position, axial, tangential, and radial velocity components, as well as all flow angles are written to one record of scratch file 50. This enables printing these quantities in a second table later during this subroutine execution.

All of the above process is repeated until all points in the propeller plane have been considered.

After all points have been considered, the contents of file 50 is printed in the propeller plane flow field output table part 2.

Lastly, this subroutine produces card punched output if the punch option, NPUNCH = 1, was specified. This punched output consists of a card for each propeller point giving point location and the calculated axial and tangential velocity components at the point. See the output description section of this manual for more details on the punched output.

The formulas for transformation of coordinate systems, formulas for induced velocity by sources and vortices, and formulas for the flow angles are all given in reference 1.

Note that all newly calculated output quantities from this subroutine are printed out or punched on cards but are not saved after printing or punching.

Use: CALL VPROPS(LLL,MD,MAXALP,ICASE,NP,ANVX,ANVY,ANVZ,S, SIGMA,XC,YC,ZC)

Common Blocks Required: TITLE, INPUTS, PROP, OPTION, WING1, and INLET

<u>Input</u>:

LLL — Program dimension size variable specified in main program which equals the maximum number of panels which may possibly be generated on any configuration. Used here for execution time dimensioning of arrays.

MAXALP — Program dimension size variable specified in main program which equals the maximum number of body orientations which may be input. Used here for execution time dimensioning of arrays.

MD — Program dimension size variable specified in main program which equals the sum of LLL and MAXALP. Used here for execution time dimensioning of arrays.

NPOINT — Propeller plane signal variable:
= 0 if no propeller plane is specified.
= 1 if propeller plane is specified.

All of the above process is repeated until all points in the propeller plane have been considered.

After all points have been considered, the contents of file 50 is printed in the propeller plane flow field output table part 2.

Lastly, this subroutine produces card punched output if the punch option, NPUNCH = 1, was specified. This punched output consists of a card for each propeller point giving point location and the calculated axial and tangential velocity components at the point. See the output description section of this manual for more details on the punched output.

The formulas for transformation of coordinate systems, formulas for induced velocity by sources and vortices, and formulas for the flow angles are all given in reference 1.

Note that all newly calculated output quantities from this subroutine are printed out or punched on cards but are not saved after printing or punching.

Use: CALL VPROPS(LLL,MD,MAXALP,ICASE,NP,ANVX,ANVY,ANVZ,S, SIGMA,XC,YC,ZC)

Common Blocks Required: TITLE, INPUTS, PROP, OPTION, WING1, and INLET

Input:

LLL — Program dimension size variable specified in main program which equals the maximum number of panels which may possibly be generated on any configuration. Used here for execution time dimensioning of arrays.

MAXALP — Program dimension size variable specified in main program which equals the maximum number of body orientations which may be input. Used here for execution time dimensioning of arrays.

MD — Program dimension size variable specified in main program which equals the sum of LLL and MAXALP. Used here for execution time dimensioning of arrays.

NPOINT — Propeller plane signal variable:
= 0 if no propeller plane is specified.
= 1 if propeller plane is specified.

ICASE	Body orientation--lift case number of the orientation for which the subroutine is now getting propeller plane flow field output.
ALPHA	Body angle of attack corresponding to orientation case ICASE.
BETA	Body sideslip angle corresponding to orientation case ICASE.
NWING	Wing input signal variable = 0 if wing is not modeled on this configuration. = 1 if wing is modeled.
CL	Array of lift coefficients of wing (if present) Value used is the one corresponding to orientation case ICASE.
V	Free stream velocity magnitude
VX, VY, VZ	Arrays of components of the free stream velocity. The values used are the ones corresponding to orientation case ICASE. Components are relative to the reference body coordinate system.
NINFLO	Number of body panels specified as having inflow or outflow through their surface.
XHUB, YHUB, ZHUB	Position coordinates of center of the propeller plane relative to reference body coordinate system.
RADIUS	Reference propeller plane (blade) radius.
ALPHP	Pitch angle of propeller plane thrust axis relative to reference body x axis.
BETAP	Yaw angle of propeller plane thrust axis relative to reference body x axis.
DPSI	Azimuth angle spacing of points around propeller plane.
NPUNCH	Punched card output option variable: = 0 if no output is to be punched on cards = 1 if desire punched card output.
SYMBOL	Alphanumeric array containing the heading supplied in the input data.

ICASE	Body orientation — lift case number of the orientation for which the subroutine is now getting propeller plane flow field output.
ALPHA	Body angle of attack corresponding to orientation case ICASE.
BETA	Body sideslip angle corresponding to orientation case ICASE.
NWING	Wing input signal variable = 0 if wing is not modeled on this configuration = 1 if wing is modeled.
CL	Array of lift coefficients of wing (if present) Value used is the one corresponding to orientation case ICASE.
V	Free stream velocity magnitude
VX, VY, VZ	Arrays of components of the free stream velocity. The values used are the ones corresponding to orientation case ICASE. Components are relative to the reference body coordinate system.
NINFLO	Number of body panels specified as having inflow or outflow through their surface.
XHUB, YHUB, ZHUB	Position coordinates of center of the propeller plane relative to reference body coordinate system.
RADIUS	Reference propeller plane (blade) radius.
ALPHP	Pitch angle of propeller plane thrust axis relative to reference body x axis.
BETAP	Yaw angle of propeller plane thrust axis relative to reference body x axis.
DPSI	Azimuth angle spacing of points around propeller plane.
NPUNCH	Punched card output option variable: = 0 if no output is to be punched on cards = 1 if desire punched card output.
SYMBOL	Alphanumeric array containing the heading supplied in the input data.

DRADUS Radial spacing of points in the propeller plane in fraction of reference propeller plane radius.

NRAD Number of points located radially along each azimuth position in the propeller plane.

NP Total number of body surface panels in the configuration.

XC, YC, ZC Array of body panel control point coordinates in terms of reference body coordinate system.

S Array of body panel surface areas.

SIGMA Array of body panel source strengths associated with the body orientation case, ICASE.

ANVX, ANVY, ANVZ Arrays of components of the outward drawn unit normal vector at each body panel in terms of the reference body coordinate system.

Output: (All output printed and/or punched. All new output below is lost after printing and after the subroutine has completed.)

Old input quantities which are also printed and saved:

ICASE Same as input definition.

ALPHA Same as input definition.

BETA Same as input definition.

CL Same as input definition.

V Same as input definition.

VX, VY, VZ Same as input definition.

XHUB, YHUB, ZHUB Same as input definition.

RADIUS Same as input definition.

ALPHP Same as input definition.

BETAP Same as input definition.

DRADUS Radial spacing of points in the propeller plane in fraction of reference propeller plane radius.

NRAD Number of points located radially along each azimuth position in the propeller plane.

NP Total number of body surface panels in the configuration.

XC, YC, ZC Array of body panel control point coordinates in terms of reference body coordinate system.

S Array of body panel surface areas.

SIGMA Array of body panel source strengths associated with the body orientation case, ICASE.

ANVX, ANVY, ANVZ Arrays of components of the outward drawn unit normal vector at each body panel in terms of the reference body coordinate system.

Output: (All output printed and/or punched. All new output below is lost after printing and after the subroutine has completed.)

Old input quantities which are also printed and saved:

ICASE Same as input definition.

ALPHA Same as input definition.

BETA Same as input definition.

CL Same as input definition.

V Same as input definition.

VX, VY, VZ Same as input definition.

XHUB, YHUB, ZHUB Same as input definition.

RADIUS Same as input definition.

ALPHP Same as input definition.

BETAP Same as input definition.

FLRATO — Through flow velocity assigned to first inlet of outlet panel (if any) specified in the program input. Quantity is fraction of free stream velocity magnitude.

New output not saved:

RRATIO — Dimensionless radial position of point in propeller plane in the propeller plane cylindrical axis system. Is ratio of radius to reference propeller plane radius.

PSI — Azimuth angular position of a point in propeller plane. In the propeller plane cylindrical axis system.

POINTX, POINTY, POINTZ — Position of a point in the propeller plane relative to reference body coordinate system.

VTOTX, VTOTY, VTOTZ — Normalized components of complete flow velocity at a point in the propeller plane relative to the reference body coordinate system.

VTOT — Normalized resultant magnitude of the induced velocity at a point in the propeller plane.

Further output printed using file 50:

VAXIAL, VRADAL, VTANG — Normalized axial, radial and tangential velocity components at a point in the propeller plane relative to the propeller plane cylindrical axis system.

ROTANG — Rotational flow angle at a point in propeller plane.

OUTANG — Outflow angle at a point in propeller plane.

UPWASH — Angle of upwash at a point in propeller plane.

SIDWSH — Angle of sidewash at a point in propeller plane.

Punched card output: (If requested)

SYMBOL — Same as input definition.

NRAD — Same as input definition.

RADIUS — Same as input definition.

FLRATO Through flow velocity assigned to first inlet of outlet panel (if any) specified in the program input. Quantity is fraction of free stream velocity magnitude.

New output not saved:

RRATIO Dimensionless radial position of point in propeller plane in the propeller plane cylindrical axis system. Is ratio of radius to reference propeller plane radius.

PSI Azimuth angular position of a point in propeller plane. In the propeller plane cylindrical axis system.

POINTX, POINTY, POINTZ Position of a point in the propeller plane relative to reference body coordinate system.

VTOTX, VTOTY, VTOTZ Normalized components of complete flow velocity at a point in the propeller plane relative to the reference body coordinate system.

VTOT Normalized resultant magnitude of the induced velocity at a point in the propeller plane.

Further output printed using file 50:

VAXIAL, VRADAL, VTANG Normalized axial, radial and tangential velocity components at a point in the propeller plane relative to the propeller plane cylindrical axis system.

ROTANG Rotational flow angle at a point in propeller plane.

OUTANG Outflow angle at a point in propeller plane.

UPWASH Angle of upwash at a point in propeller plane.

SIDWSH Angle of sidewash at a point in propeller plane.

Punched card output: (If requested)

SYMBOL Same as input definition.

NRAD Same as input definition.

RADIUS Same as input definition.

ALPHA	Same as input definition.
BETA	Same as input definition.
PSI	Same as defined earlier in output section.
RRATIO	Same as defined earlier in output section.
VAXIAL, VTANG	Same as defined earlier in output section.
NPSIS	Total number of azimuth locations at which points are situated on the propeller plane.
ATOT	Propeller plane angle of attack between propeller plane thrust axis and free stream velocity direction.
BTOT	Propeller plane angle of sideslip between propeller plane thrust axis and free stream velocity direction.

Subroutines Called: EULER, WINGV (if NWING = 1), and ANGLES

Warning Messages: If 360 is not an integer multiple of DPSI, DPSI is printed with a warning stating uneven azimuthal spacing of propeller plane points exists. Subroutine continues to execute normally.

Error Returns: None

Error Stops: None

ALPHA   Same as input definition.

BETA   Same as input definition.

PSI   Same as defined earlier in output section.

RRATIO   Same as defined earlier in output section.

VAXIAL, VTANG   Same as defined earlier in output section.

NPSIS   Total number of azimuth locations at which points are situated on the propeller plane.

ATOT   Propeller plane angle of attack between propeller plane thrust axis and free stream velocity direction.

BTOT   Propeller plane angle of sideslip between propeller plane thrust axis and free stream velocity direction.

Subroutines Called: EULER, WINGV (if NWING = 1), and ANGLES

Warning Messages: If 360 is not an integer multiple of DPSI, DPSI is printed with a warning stating uneven azimuthal spacing of propeller plane points exists. Subroutine continues to execute normally.

Error Returns: None

Error Stops: None

Subroutine WGEOM

Purpose: To set up the wing horseshoe vortex geometry and strengths corresponding to each input body orientation-lift case.

Method: This subroutine is called only if a wing has been modeled. For each input body orientation, a horseshoe vortex is generated. The horseshoe vortex has a span equal to (0.25 * π * physical wing span). The trailing vortices are shed parallel to the free stream velocity and are extended aft a distance of one hundred wing chord lengths.

The end point coordinates of the four finite parts of each horseshoe are calculated relative to the reference body coordinate system. These four parts are the straight line vortex filaments consisting of the left side trailing vortex, the right side trailing vortex, the left semispan bound vortex (on the quarter chord line), and the right semispan bound vortex (also on the quarter chord line).

For each body orientation, a different vortex strength is calculated as a function of the input lift coefficients.

The output horseshoe vortex geometry and strength is printed out on a table, and the values are saved in common blocks WING1 and WING2.

Use: CALL WGEOM (Only if wing has been modeled).

Common Blocks Required: INPUTS, WING1, and WING2

Input:

V	Magnitude of free stream velocity.
SPAN	Physical wing span.
CHORD	Mean or root chord of wing.
XQR, YQR, ZQR	Coordinates of wing root quarter chord location. Also the inboard end points of each bound semi-span vortex filament. Relative to the reference body coordinate system.
DIHED	Wing and bound vortex dihedral angle.

Subroutine WGEOM

Purpose: To set up the wing horseshoe vortex geometry and strength corresponding to each input body orientation.

Method: This subroutine is called only if a wing has been modeled. For each input body orientation, a horseshoe vortex is generated. The horseshoe vortex has a span equal to ($\pi/4$ of a physical wing span). The trailing vortices are shed parallel to the free stream velocity and are extended aft a distance of one hundred wing chord lengths.

The end point coordinates of the four finite parts of each horseshoe are calculated relative to the reference body coordinate system. These four parts are the straight line vortex filaments consisting of the left side trailing vortex, the right side trailing vortex, the left semispan bound vortex (on the quarter chord line), and the right semispan bound vortex (also on the quarter chord line).

For each body orientation, a different vortex strength is calculated as a function of the input lift coefficients.

The output horseshoe vortex geometry and strength is printed out on a table, and the values are saved in common blocks WING1 and WING2.

Use: CALL WGEOM (Only if wing has been modeled).

Common Blocks Required: INPUTS, WING1, and WING2

<u>Input</u>:

V	Magnitude of free stream velocity.
SPAN	Physical wing span.
CHORD	Mean or root chord of wing.
XQR, YQR, ZQR	Coordinates of wing root quarter chord location. Also the inboard end points of each bound semi-span vortex filament. Relative to the reference body coordinate system.
DIHED	Wing and bound vortex dihedral angle.

SWEEP Wing quarter chord and bound vortex sweep angle.

NALPHA Number of input body orientations.

CL Array of wing lift coefficients. Each value corresponds to a different body orientation.

ALPHA Array of body angles of attack. Each value corresponds to a different body orientation.

BETA Array of body sideslip angles. Each value corresponds to a different body orientation.

Output: (All of the above input values as well as output, are printed out. All coordinates below are relative to the reference body coordinate system.)

SPAN Horseshoe vortex span. (Output value replaces the original input value of physical wing span.)

XBTIPL, YBTIPL, ZBTIPL Coordinates of the outboard end of the left side bound vortex filament. Also the upwind end of the left trailing vortex filament.

XBTIPR, YBTIPR, ZBTIPR Coordinates of the outboard end of the right side bound vortex filament. Also the upwind end of the right trailing vortex filament.

GAMMA Array of horseshoe vortex strengths. Each value corresponds to a different body orientation.

XTRALL, YTRALL, ZTRALL Arrays of coordinates of the downwind end of the left trailing vortex filament. Each set of coordinates corresponds to a different body orientation.

XTRALR, YTRALR, ZTRALR Arrays of coordinates of the downwind end of the right trailing vortex filament. Each set of coordinates corresponds to a different body orientation.

Subroutines Called: None

Error Returns: None

Error Stops: None

Subroutine WINGV

Purpose: To calculate the velocity components induced at a specified point by the wing horseshoe vortex of constant strength.

Method: The horseshoe vortex is broken into four finite length straight vortex filaments all of the same strength. The four parts are the left side trailing filament, left semispan bound filament, right semispan bound filament, and right side trailing filament. The end points of the four filaments are obtained from common blocks WING1 and WING2. At the desired point, the induced velocity components due to each of the four filaments are computed, in turn, by the Biot-Savart law. Then the component contributions due to all filaments are summed to give the resulting velocity components due to the horseshoe vortex as a whole.

Use: CALL WINGV(ICASE,X,Y,Z,VELX,VELY,VELZ)

Common Blocks Required: WING1 and WING2

Note: All coordinates and velocity components below are relative to the reference body coordinate system.

Input:

X, Y, Z — Coordinates of the point at which the vortex induced velocities are to be found.

ICASE — Body orientation--lift case (angle of attack, sideslip, and lift coefficient) being considered.

XTRALL, YTRALL, ZTRALL — Arrays of coordinates of the downstream starting end of the left side trailing vortex filament for the body orientation case ICASE.

XBTIPL, YBTIPL, ZBTIPL — Coordinates of the point at which left side trailing vortex joins the left semispan bound vortex filament.

GAMMA — Constant strength of the horseshoe vortex and its four straight line filament parts. Value corresponds to the orientation case ICASE.

XQR, YQR, ZQR	Coordinates of the inboard or root ends of the left and right semispan bound vortex filaments.
XBTIPR, YBTIPR, ZBTIPR	Coordinates of the point at which the right side trailing vortex filament joins the right semispan bound vortex filament.
XTRALR, YTRALR, ZTRALR	Array of coordinates of the downstream end of the right side trailing vortex filament.

<u>Output</u>:

VELX, VELY, VELZ	Components of velocity induced at the input point by the complete wing horseshoe vortex.

Subroutines
Called: VORTEX

Error Returns: None

Error Stops: None

APPENDIX B

Source Program Listing

This version of the program is dimensioned using the following values of the dimension size variables specified in the main program. (Quantities are explained in sections 2.2 and 2.3 of this manual.)

NSECMA = 60

NIPMAX = 45

MAXALP = 6

MAXINF = 500

LLL = 2596

MD = 2602

LLLHAF = 1298

and:

$DPSI_{min}$ = 1.0

$NRAD_{max}$ = 51

```
C     ****************************************************************MAN   50
C     *     3-D BODY POTENTIAL FLOW SURFACE FLOW PREDICTIONS         *MAN  100
C     *       AND PROPELLER PLANE FLOW FIELD PREDICTION              *MAN  150
C     *                    (MAIN PROGRAM)                            *MAN  200
C     ****************************************************************MAN  250
C                                                                     MAN  300
C     ****************************************************************MAN  350
C     ****************************************************************MAN  400
C     *** SEE SUBROUTINE INPUT LISTING FOR COMPLETE DESCRIPTION OF DATA MAN  450
C     TO BE SUPPLIED BY USER ***                                      MAN  500
C     ----------------------------------------------------------------MAN  550
C                                                                     MAN  600
C                    FEATURES/CHARACTERISTICS                         MAN  650
C                    ------------------------                         MAN  700
C     SUBROUTINES REQUIRED: INPUT, VCOMP, EULER, WGEOM, PANEL, COEFIC, MAN  750
C                           SOLVE, COFSYM, SOLSYM, VELOCI, WINGV,     MAN  800
C                           VORTEX, VPROPS, AND ANGLES.               MAN  850
C     ----------------------------------------------------------------MAN  900
C     DIMENSIONS MAY BE ADJUSTED TO PERMIT SOLUTION OF SEVERAL BODY   MAN  950
C     ORIENTATIONS (ALPHA-BETA) DURING A SINGLE RUN.                  MAN 1000
C     ----------------------------------------------------------------MAN 1050
C     SYMMETRIC BODY OPTION MAY BE USED IF BODY IS SYMMETRIC ABOUT X-Z MAN 1100
C     PLANE, AND WING (IF PRESENT) IS ALSO SYMMETRIC ABOUT X-Z PLANE, MAN 1150
C     AND ALL INPUT BODY SIDESLIP ANGLES ARE ZERO.  THEN ONLY THE LEFT MAN 1200
C     SIDE BODY GEOMETRY MUST BE INPUT BY USER. THE EQUATION SYSTEM   MAN 1250
C     SIZE IS CUT IN HALF THUS GREATLY REDUCING EXECUTION TIME.       MAN 1300
C     ----------------------------------------------------------------MAN 1350
C     PROPELLER PLANE FLOW ANGLES AND VELOCITIES OUTPUT MAY BE PUNCHED MAN 1400
C     ON CARDS IF PUNCH OPTION IS SPECIFIED IN INPUT DATA.            MAN 1450
C     ----------------------------------------------------------------MAN 1500
C     TWO SEQUENTIAL SCRATCH FILES (MAGNETIC DISK OR TAPE) MUST BE    MAN 1550
C     PROVIDED- (SEE USERS MANUAL TO DETERMINE FILE SIZES).           MAN 1600
C      FILE  9: CONTAINS SYSTEM OF SIMULTANEOUS LINEAR EQUATIONS (USED MAN 1650
C               IN SUBROUTINES COEFIC, COFSYM, SOLVE, AND SOLSYM).    MAN 1700
C      FILE 50: USED IN SUBROUTINE VPROPS TO PRINT FLOW VELOCITIES AND MAN 1750
C               FLOW ANGLES.                                          MAN 1800
```

```
C     (NOTE DATA ON THESE FILES ARE LOST AFTER PROGRAM COMPLETION.)     MAN 1850
C     ------------------------------------------------------------------MAN 1900
C     AUTOMATIC EXECUTION TIME DIMENSIONING OF SUBROUTINE DIMENSION     MAN 1950
C     STATEMENTS IS USED. COMMON BLOCK ARRAY DIMENSIONS ARE NOT         MAN 2000
C     AUTOMATICALLY DIMENSIONED. - SEE BELOW.                           MAN 2050
C     ------------------------------------------------------------------MAN 2100
C     USER MAY CHANGE PROGRAM DIMENSION SIZE BY CHANGING FOUR SIZING    MAN 2150
C     VARIABLES:                                                        MAN 2200
C      NSECMA - MAXIMUM ALLOWED INPUT BODY SECTION CONTOURS. (MAY BE    MAN 2250
C               ANY INTEGER WITH VALUE OF 3 OR MORE.)                   MAN 2300
C      NIPMAX - MAXIMUM ALLOWED NUMBER OF INPUT POINTS ON ANY COMPLETE  MAN 2350
C               CONTOUR (TOP POINT COUNTED TWICE). (MUST BE AN ODD      MAN 2400
C               INTEGER HAVING A VALUE OF 5 OR MORE.)                   MAN 2450
C      MAXALP - MAXIMUM ALLOWED NUMBER OF INPUT BODY ORENTATIONS (ALPHA MAN 2500
C               -BETA) INPUT IN ONE RUN. (MUST BE AN INTEGER WITH A     MAN 2550
C               VALUE OF 1 OR MORE.)                                    MAN 2600
C      MAXINF - MAXIMUM ALLOWED NUMBER OF THROUGH FLOW PANELS SPECIFIED MAN 2650
C               OVER ENTIRE BODY. (MUST BE AN EVEN INTEGER WITH A VALUE MAN 2700
C               OF 2 OR MORE.)                                          MAN 2750
C     ***NOTE: QUANTITY LLL=((NSECMA-1)*(NIPMAX-1)) IS THE MAXIMUM      MAN 2800
C              NUMBER OF BODY PANELS WHICH MAY BE GENERATED BY THIS     MAN 2850
C              PROGRAM AND WILL ALWAYS BE AN EVEN NUMBER. ***           MAN 2900
C     VALUES ARE GIVEN TO THESE VARIABLES IN MAIN PROGRAM BELOW, THEN   MAN 2950
C     THE DIMENSION STATEMENT IN MAIN PROGRAM MUST BE CHANGED           MAN 3000
C     ACCORDINGLY - SEE USERS MANUAL. FINALLY EVERY AFFECTED COMMON     MAN 3050
C     CARD IN EACH SUBROUTINE MUST BE CHANGED ACCORDING TO THE SIZE     MAN 3100
C     VARIABLES - SEE USERS MANUAL.                                     MAN 3150
C     ------------------------------------------------------------------MAN 3200
C     EIGHT COMMON BLOCKS ARE USED AMONG SUBROUTINES AND MAIN PROGRAM   MAN 3250
C     (SEE ALSO USERS MANUAL):                                          MAN 3300
C     BLOCK NAME:       DATA STORED:            SUBROUTINES CONTAINING IT: MAN 3350
C      /INLET/   BODY THROUGH FLOW PANELS    INPUT, COEFIC, COFSYM,     MAN 3400
C                                            VELOCI, AND VPROPS         MAN 3450
C      /INPUTS/  BODY ORIENTATIONS, FREE     MAIN PROG., INPUT, VCOMP,  MAN 3500
C                STREAM COMPONENTS.          COEFIC, COFSYM, SOLVE,     MAN 3550
C                                            SOLSYM, VELOCI, VPROPS,    MAN 3600
```

```
C                                             WGEOM, AND VORTEX.          MAN 3650
C     /OPTION/  OUTPUT OPTION CONTROL         INPUT, PANEL, VPROPS        MAN 3700
C               VARIABLES                                                 MAN 3750
C     /PROP/    PROP PLANE GEOMETRY           INPUT, VPROPS               MAN 3800
C     /SYMTRE/  SYMMETRIC BODY PANEL          MAIN PROG., INPUT, PANEL,   MAN 3850
C               RECORDING VARAIBLES           COFSYM, AND SOLSYM.         MAN 3900
C     /TITLE/   INPUT DATA HEADING            INPUT, AND VPROPS           MAN 3950
C     /WING1/   INPUT WING GEOMETRY           MAIN PROG., INPUT, COEFIC,  MAN 4000
C                                             COFSYM, WGEOM, WINGV,       MAN 4050
C                                             VELOCI, VPROPS, VORTEX.     MAN 4100
C     /WING2/   GENERATED WING VORTEX         WGEOM, AND WINGV            MAN 4150
C               GEOMETRY                                                  MAN 4200
C                                                                         MAN 4250
C     ////////////////////////////////////////////////////////////////////MAN 4300
C     CURRENT VERSION OF PROGRAM SIZED USING :                            MAN 4350
C     NSECMA=60   , NIPMAX=45    , MAXALP=6   , AND MAXINF= 500           MAN 4400
C                                                                         MAN 4450
C     GIVES THE FOLLOWING CAPABILITIES:                                   MAN 4500
C     * MAY PANEL AT LEAST LLL= 8 AND UP TO LLL= 2596  BODY PANELS OVER   MAN 4550
C       ENTIRE SURFACE OR HALF THESE AMOUNTS OVER ONE SIDE OF BODY IF     MAN 4600
C       SYMMETRY OPTION IS USED.                                          MAN 4650
C     * SOLVES UP TO 6 BODY ORIENTATION AND INPUT LIFT COEFFICIENT        MAN 4700
C       CASES SIMULTANEOUSLY IN ONE RUN.                                  MAN 4750
C     * MAY SPECIFY UP TO 500  PANELS OVER ENTIRE BODY OR HALF THAT MANYMAN 4800
C       OVER ONE SIDE OF BODY (IF SYMMETRIC) WHICH ARE INLET AND OUTLET   MAN 4850
C       OUTLET PANELS HAVING THROUGH FLOW.                                MAN 4900
C     * MUST BE RUN IN 560K STORAGE REGION -CATEGORY 5 ONLY- USING        MAN 4950
C       THE WATIV, G, OR H COMPILERS ON THE IBM 370 AT PENN STATE.        MAN 5000
C     ********************************************************************MAN 5050
C     ********************************************************************MAN 5100
C                                                                         MAN 5150
      DIMENSION A(2602),SIGMA(2596,6),NCOUNT(60),NFLAG(60),PX(60,45),     MAN 5200
     1PY(60,45),PZ(60,45),STOX1(45),STOY1(45),STOZ1(45),STOX2(45),        MAN 5250
     2STOY2(45),STOZ2(45),ANVX(2596),ANVY(2596),ANVZ(2596),S(2596),       MAN 5300
     3XC(2596),YC(2596),ZC(2596),BIG(6),VARIAB(6),SUM(6),SIGSAV(1298)     MAN 5350
      COMMON /INPUTS/ ALPHA(6),BETA(6),VX(6),VY(6),VZ(6),V,NALPHA         MAN 5400
```

```
      COMMON /SYMTRE/ NPNSYM(2596),INSOLV(1298),NSYMET                  MAN 5450
      COMMON /WING1/ CL(6),CHORD,DIHED,SPAN,SWEEP,XQR,YQR,ZQR,NWING     MAN 5500
C                                                                       MAN 5550
C     FOUR USER SPECIFIED PROGRAM SIZE CONTROL VARIABLES FOR SIZING OF  MAN 5600
C     THE ABOVE DIMENSION STATEMENT, COMMON STATEMENT, AND ALL          MAN 5650
C     SUBROUTINE COMMON STATEMENTS. THEY MUST BE INSERTED NEXT:         MAN 5700
C                                                                       MAN 5750
C     IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIMAN 5800
      NSECMA=60                                                         MAN 5850
      NIPMAX=45                                                         MAN 5900
      MAXALP=6                                                          MAN 5950
      MAXINF=500                                                        MAN 6000
C     IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIMAN 6050
C                                                                       MAN 6100
      LLL=(NSECMA-1)*(NIPMAX-1)                                         MAN 6150
      ND=LLL+MAXALP                                                     MAN 6200
      LLLHAF=LLL/2                                                      MAN 6250
C                                                                       MAN 6300
C     CHECK VALIDITY OF THE ABOVE DIMENSIONING VARIABLES (STOP PROGRAM  MAN 6350
C     ISTOP = 1) IF ONE OR MORE ERRORS EXIST IN THESE VALUES:           MAN 6400
      ISTOP=0                                                           MAN 6450
      IF(NSECMA.GE.3) GO TO 1                                           MAN 6500
      ISTOP=1                                                           MAN 6550
      WRITE(6,100)                                                      MAN 6600
    1 IF(NIPMAX.GE.5) GO TO 2                                           MAN 6650
      ISTOP=1                                                           MAN 6700
      WRITE(6,200)                                                      MAN 6750
C     CHECK THAT NIPMAX IS ODD:                                         MAN 6800
    2 BTEST=(FLOAT(NIPMAX)+1.0)/2.0                                     MAN 6850
      IF(IFIX(BTEST).EQ.BTEST) GO TO 3                                  MAN 6900
      ISTOP=1                                                           MAN 6950
      WRITE(6,300)                                                      MAN 7000
    3 IF(MAXALP.GE.1) GO TO 4                                           MAN 7050
      ISTOP=1                                                           MAN 7100
      WRITE(6,400)                                                      MAN 7150
    4 IF(MAXINF.GE.2) GO TO 5                                           MAN 7200
```

```
      ISTOP=1
      WRITE(6,500)
C     CHECK THAT MAXINF IS EVEN:
    5 BTEST=(FLOAT(MAXINF)+1.0)/2.0
      IF(IFIX(BTEST).NE.BTEST) GO TO 6
      ISTOP=1
      WRITE(6,600)
    6 IF(ISTOP.EQ.0) GO TO 7
      WRITE(6,700)
      STOP
C
    7 CONTINUE
C
C     READ IN, CHECK FOR DATA ERRORS, AND PRINT LIST OF ALL INPUT DATA:
C
      CALL INPUT(LLL,MAXALP,MAXINF,NIPMAX,NSECMA,NSECTO,NCOUNT,
     1NFLAG,PX,PY,PZ,ITMAX,ERR)
C
C     COMPUTE COMPONENTS OF FREESTREAM VELOCITY FOR EACH OF THE INPUT
C     BODY (ALPHA-BETA) ORIENTATIONS. SAVE IN COMMON /INPUTS/:
C
      CALL VCOMP
C
C     IF WING IS INPUT ON THE BODY, GENERATE THE WING VORTEX MODEL.
C     GENERATE VORTEX STRENGTHS, AND GEOMETRY CORRESPONDING TO EACH
C     INPUT BODY (ALPHA-BETA) ORIENTATION. SAVE IN COMMON /WING2/:
C
      IF(NWING.EQ.1) CALL WGEOM
C
C     GENERATE THE BODY SURFACE PANEL GEOMETRY - TOTAL NUMBER OF
C     PANELS, PANEL AREAS, PANEL CONTROL POINTS, NORMAL UNIT VECTOR
C     COMPONENTS, AND RECORD OF SYMMETRIC PANEL PAIRS (IF SYMMETRY
C     OPTION WAS USED), AND PRINT GEOMETRY LIST IF INPUT REQUESTED IT.
C
      CALL PANEL(NIPMAX,NSECMA,NSECTO,NFLAG,NP,NCOUNT,LLL,PX,PY,PZ,
     1STOX1,STOY1,STOZ1,STOX2,STOY2,STOZ2,S,ANVX,ANVY,ANVZ,XC,YC,ZC)
```

```
C
C
C     SET UP MATRIX OF SIMULTANEOUS LINEAR EQUATIONS THEN SOLVE THE
C     SYSTEM, ITERATIVELY, FOR THE UNKNOWN BODY PANEL SOURCE STRENGTHS.
C     IF BODY WAS NOT INPUT AS SYMMETRIC, USE FULL SIZE SYSTEM OF
C     EQUATIONS ON FILE 9 USING COEFIC AND SOLVE. IF BODY IS SYMMETRIC,
C     USE THE HALF SIZE SYSTEM OF EQUATIONS ON FILE 9 USING SUBROUTINES
C     COFSYM AND SOLSYM:
C
      IF(NSYMET.EQ.0) GO TO 8
      CALL COEFIC(LLL,MD,NP,XC,YC,ZC,S,ANVX,ANVY,ANVZ,A)
      CALL SOLVE(LLL,MD,MAXALP,NP,A,BIG,VARIAB,SUM,ERR,ITMAX,SIGMA)
      GO TO 9
    8 CALL COFSYM(LLL,MD,NP,XC,YC,ZC,S,ANVX,ANVY,ANVZ,A)
      CALL SOLSYM(LLL,LLLHAF,MD,MAXALP,NP,A,BIG,VARIAB,SUM,ERR,ITMAX,
     1SIGMA,SIGSAV)
    9 CONTINUE
C
C     FOR EACH INPUT BODY ORIENTATION (ALPHA(ICASE), BETA(ICASE)) PAIR
C     (IN TURN) CALCULATE BODY SURFACE VELOCITIES AND PRESSURES - PRINT
C     LIST OF VALUES. ALSO IF PROPELLER PLANE FLOW IS DESIRED,
C     PRINT OUT LIST OF VALUES, AND PUNCH VALUES ON CARDS IF PUNCH
C     OPTION WAS SPECIFIED BY USER:
C
      DO 10 ICASE=1,NALPHA
      CALL VELOCI(LLL,MD,MAXALP,ICASE,NP,ANVX,ANVY,ANVZ,S,SIGMA,
     1XC,YC,ZC)
      CALL VPROPS(LLL,MD,MAXALP,ICASE,NP,ANVX,ANVY,ANVZ,S,SIGMA,
     1XC,YC,ZC)
   10 CONTINUE
      WRITE(6,800)
      STOP
C
C     FORMATS
C
  100 FORMAT('0','*** ERROR *** : NSECMA DIMENSIONING VARIABLE GIVEN IN
```

```
     1MAIN PROGRAM IS LESS THAN 3 - INVALID. RECHECK DIMENSIONING OF PROMAN10850
     2GRAM')                                                              MAN10900
 200 FORMAT('0','*** ERROR *** : NIPMAX DIMENSIONING VARIABLE GIVEN IN MAN10950
     1MAIN PROGRAM IS LESS THAN 5 - INVALID. RECHECK DIMENSIONING OF PROMAN11000
     2GRAM')                                                              MAN11050
 300 FORMAT('0','*** ERROR *** : NIPMAX DIMENSIONING VARIABLE GIVEN IN MAN11100
     1MAIN PROGRAM IS EVEN VALUE - INVALID. (MUST BE ODD INTEGER).'/,' 'MAN11150
     2,T17,'RECHECK DIMENSIONING OF PROGRAM.')                            MAN11200
 400 FORMAT('0','*** ERROR *** : MAXALP DIMENSIONING VARIABLE GIVEN IN MAN11250
     1MAIN PROGRAM IS ZERO OR NEGATIVE - INVALID. RECHECK DIMENSIONING OMAN11300
     2F PROGRAM')                                                         MAN11350
 500 FORMAT('0','*** ERROR *** : MAXINF DIMENSIONING VARIABLE GIVEN IN MAN11400
     1MAIN PROGRAM IS LESS THAN 2  - INVALID. RECHECK DIMENSIONING OF PRMAN11450
     2OGRAM.')                                                            MAN11500
 600 FORMAT('0','*** ERROR *** : MAXINF DIMENSIONING VARIABLE GIVEN IN MAN11550
     1MAIN PROGRAM IS ODD VALUE. - INVALID. (MUST BE EVEN INTEGER).'/,' MAN11600
     2',T17,'RECHECK DIMENSIONING OF PROGRAM.')                           MAN11650
 700 FORMAT('0','*** PROGRAM STOPPED DUE TO INVALID DIMENSIONING VARIABMAN11700
     1LES USED IN MAIN PROGRAM ***')                                      MAN11750
 800 FORMAT('0',T40,'*** END OF JOB - ALL PARTS COMPLETED SUCCESSFULLY MAN11800
     1 ***')                                                              MAN11850
      END                                                                 MAN11900
C                                                                         MAN11950
C                                                                         MAN12000
C     ****************************************************************MAN12050
C                                                                         MAN12100
C                                                                         MAN12150
C                                                                         MAN12200
```

```
C     ****************************************************************INP
C                                                                     INP
C     SUBROUTINE INPUT(LLL,MAXALP,MAXINF,NIPMAX,NSECMA,NSECTO,NCOUNT,   INP
C    INFLAG,PX,PY,PZ,ITMAX,ERR)                                        INP
C                                                                     INP
C     ****************************************************************INP
C     THIS ROUTINE READS IN AND PRINTS OUT ALL USER SUPPLIED INPUT DATA INP
C     THE INPUT DATA DECK CONSISTS OF 8 GROUPS OR CARD SETS. THE ROUTINEINP
C     CHECKS FOR CERTAIN ERRORS IN THE INPUT AND WILL STOP THE PROGRAM  INP
C     WITH A PRINTED MESSAGE WHEN THE FIRST SUCH ERROR IS FOUND.        INP
C                                                                     INP
C     **************************************                          INP
C     * DESCRIPTION OF INPUT DATA CARD SETS *                         INP
C     **************************************                          INP
C                                                                     INP
C     // VELOCITIES, COORDINATES AND LENGTHS ALL IN SAME CONSISTENT     INP
C        FAMILY ON UNITS.                                              INP
C                                                                     INP
C     // PUNCH ALL INTEGERS RIGHT JUSTIFIED IN COLUMN FIELDS AND ALL    INP
C        FLOATING POINT NUMBERS ARE SINGLE PRECISION (INCLUDE DECIMAL   INP
C        POINT).                                                       INP
C                                                                     INP
C     ****************************************************************INP
C-----CARD SET 1 : TITLE CARDS (2 CARDS)                              INP
C     ****************************************************************INP
C                                                                     INP
C     CARD VARIABLE COLUMN(S) NUMBER TYPE DESCRIPTION                  INP
C     1.1  SYMBOL    1 TO 80    CHARACTERS  ANY DESIRED KEYPUNCH SYMBOLS INP
C                                          OR BLANKS WHICH IDENTIFY THE  INP
C     1.2  SYMBOL    1 TO 80    CHARACTERS  DATA DECK.                  INP
C                                                                     INP
C     (CARDS 1.1 AND 1.2 MUST BOTH BE USED EVEN IF BOTH ARE BLANK).     INP
C                                                                     INP
C                                                                     INP
C     ****************************************************************INP
C-----CARD SET 2 : FREE STREAM VELOCITY - BODY ORIENTATION INFORMATION  INP
```

```
C                (CONTAINS 3 TO 8 CARDS)                                  INP 1850
C     *****************************************************************INP 1900
C     CARD VARIABLE COLUMN(S) NUMBER TYPE DESCRIPTION                     INP 1950
C     2.1    V        1 TO 10    FLOAT POINT MAGNITUDE OF FREE STREAM     INP 2000
C                                            VELOCITY (UNITY IS CONVENIENT)INP 2050
C                                            MUST BE GREATER THAN ZERO.    INP 2100
C                                                                          INP 2150
C     2.2  NALPHA   1            INTEGER     NUMBER OF ORIENTATIONS (PAIRS INP 2200
C                                            OF ALPHA AND BETA) THIS RUN.  INP 2250
C                                            (VALUE FROM 1 TO 6).          INP 2300
C                                                                          INP 2350
C     2.3  ALPHA    1 TO 10    FLOAT POINT BODY ANGLE OF ATTACK (DEGREES)INP 2400
C                                            NOSE UP POSITIVE. IT IS ANGLE INP 2450
C                                            BETWEEN FREE STREAM AND BODY  INP 2500
C                                            LONGITUDINAL (X) AXIS AS SEEN INP 2550
C                                            IN THE (X-Z) PLANE. (VALUES OFINP 2600
C                                            0 TO 360 AND 0 TO -360).      INP 2650
C                                                                          INP 2700
C          BETA     20 TO 29   FLOAT POINT BODY SIDESLIP ANGLE (DEGREES) INP 2750
C                                            NOSE RIGHT IS POSITIVE. IT IS INP 2800
C                                            ANGLE BETWEEN FREE STREAM AND INP 2850
C                                            LONGITUDINAL (X) AXIS         INP 2900
C                                            AS SEEN PROJECTED ON A PLANE  INP 2950
C                                            WHICH WOULD CONTAIN THE BODY  INP 3000
C                                            X AND Y AXES IF BODY WERE AT  INP 3050
C                                            ZERO ALPHA. (IE. AS SEEN      INP 3100
C                                            LOOKING VERTICALLY DOWN).     INP 3150
C                                            (VALUES 0 TO 360 AND 0 TO     INP 3200
C                                            -360).                        INP 3250
C                                                                          INP 3300
C     (CARD 2.3 REPEATED NALPHA TIMES - ONE ANGLE PAIR PER CARD)          INP 3350
C                                                                          INP 3400
C     NOTE : (ALL BETA VALUES MUST BE ZERO IF SYMMETRIC BODY INPUT,       INP 3450
C            NSYMET =0, IS SPECIFIED IN CARD SET 3 BELOW).                 INP 3500
C                                                                          INP 3550
C                                                                          INP 3600
```

```
C      ****************************************************************INP 3650
C-----CARD SET 3 : SYMMETRIC BODY INPUT OPTION AND BODY SURFACE INPUT    INP 3700
C                  GEOMETRY FOR PANELING. BODY FIXED REFERENCE AXES      INP 3750
C                  ASSUMED WITH LONGITUDINAL X AXIS POSITIVE FOREWARD,   INP 3800
C                  Y POSITIVE TOWARD RIGHT, AND Z POSITIVE DOWN. ORIGIN  INP 3850
C                  OF AXES MAY BE LOCATED ANYWHERE BUT ROTATE WITH BODY  INP 3900
C                  WHEN BODY HAS ANGLES OF ATTACK AND SIDESLIP. (USUALLY INP 3950
C                  ORIGIN LOCATED ON AIRCRAFT CENTERLINE IS CONVENIENT). INP 4000
C                  (SET CONTAINS 11 TO 2762 CARDS).                      INP 4050
C      ****************************************************************INP 4100
C                                                                        INP 4150
C      CARD VARIABLE COLUMN(S) NUMBER TYPE DESCRIPTION                   INP 4200
C      3.1  NSYMET    1          INTEGER     =0 SYMMETRY USED. MAY READ IN INP 4250
C                                               ONLY LEFT SIDE (-Y) SIDE OFINP 4300
C                                               BODY GEOMETRY IF BODY AND  INP 4350
C                                               WING IS SYMMETRIC ABOUT THEINP 4400
C                                               X-Z PLANE AND ALL SIDESLIP INP 4450
C                                               ANGLES, BETA, ARE ZERO.    INP 4500
C                                               ALLOWS FASTER SOLUTION. SEEINP 4550
C                                               ALSO NOTE 1 OF CARD SET 6. INP 4600
C                                              -OR-                        INP 4650
C                                            =1 NONSYMMETRY USED. MAY USE  INP 4700
C                                               ALWAYS, BUT MUST BE USED IFINP 4750
C                                               BODY IS NOT SYMMETRIC      INP 4800
C                                               ABOUT THE X-Z PLANE OR IF  INP 4850
C                                               ANY INPUT BETA IS NONZERO. INP 4900
C                                               MUST ALSO BE USED IF BODY  INP 4950
C                                               IS SYMMETRIC BUT THE WING  INP 5000
C                                               IS NOT. (SEE NOTE 1 OF CARDINP 5050
C                                               SET 6 BELOW). ENTIRE BODY  INP 5100
C                                               SURFACE MUST BE INPUT BELOWINP 5150
C                                                                        INP 5200
C      3.2  NSECTO    1 AND 2    INTEGER        TOTAL NUMBER OF BODY CROSS-  INP 5250
C                                            SECTIONS IN THIS BODY. CROSS- INP 5300
C                                            SECTIONS ARE TAKEN AT CONSTANTINP 5350
C                                            OR NEARLY CONSTANT X. A CROSS-INP 5400
```

```
C                                      SECTION MAY BE REPEATED WITH  INP 5450
C                                      A NEW POINT DESCRIPTION, BUT  INP 5500
C                                      THE REPEAT IS COUNTED AS      INP 5550
C                                      AN ADDITIONAL SECTION. (MUST  INP 5600
C                                      BE MORE THAN 2, MAX OF 60).   INP 5650
C                                                                    INP 5700
C    3.3 - CONSISTS OF NSECTO GROUPS OF CARDS. EACH GROUP IS ASSOCIATEDINP 5750
C          WITH ONE CROSS-SECTION AND CONSISTS OF ONE CARD ,3.3.A, PLUSINP 5800
C          SEVERAL CARDS, 3.3.B. GROUPS ARE IN SEQUENCE BEGINNING      INP 5850
C          WITH THAT FOR THE MOST FOREWARD BODY CROSS-SECTION. GROUP   INP 5900
C          CARDS ARE SHOWN BELOW --                                    INP 5950
C                                                                    INP 6000
C    CARD VARIABLE COLUMN(S) NUMBER TYPE DESCRIPTION                 INP 6050
C    3.3.A NSEC     1 AND 2   INTEGER     SEQUENCE NUMBER OF THIS CROSS-INP 6100
C                                      SECTION.                      INP 6150
C                                                                    INP 6200
C          NIP      10 AND 11 INTEGER     NUMBER OF POINTS SPECIFIED ON INP 6250
C                                      THIS CROSS-SECTION PERIPHERY. INP 6300
C                                      IF SYMMETRY OPTION USED, NIP  INP 6350
C                                      IS NUMBER OF POINTS ON LEFT   INP 6400
C                                      HALF OF CROSS SECTION         INP 6450
C                                      (INCLUDING UPPER AND LOWER    INP 6500
C                                      CENTERLINE POINTS)--(HERE NIP INP 6550
C                                      MUST BE MORE THAN 2, MAX OF   INP 6600
C                                      23). IF NONSYMMETRY IS USED,  INP 6650
C                                      NIP IS NUMBER OF POINTS ON    INP 6700
C                                      ENTIRE CROSS SECTION (WITH    INP 6750
C                                      FIRST POINT COUNTED TWICE)--  INP 6800
C                                      (HERE NIP MUST BE MORE THAN 4,INP 6850
C                                      MAXIMUM OF 45).               INP 6900
C                                                                    INP 6950
C          NEND     30        INTEGER     =0 IF THIS SECTION IS NOT     INP 7000
C                                         REPEATED. PANELING         INP 7050
C                                         CONTINUES BETWEEN THIS     INP 7100
C                                         SECTION AND THE NEXT ONE.  INP 7150
C                                        -OR-                        INP 7200
```

```
C                                            =1 IF THIS IS LAST SECTION     INP 7250
C                                               OF A BODY SEGMENT AND A NEWINP 7300
C                                               SEGMENT SECTION FOLLOWS. ORINP 7350
C                                               IF THIS SECTION WILL BE     INP 7400
C                                               REPEATED WITH DIFFERENT     INP 7450
C                                               POINT DESCRIPTIONS. PANELS INP 7500
C                                               WILL NOT BE CONSTRUCTED     INP 7550
C                                               BETWEEN THIS INPUT SECTION INP 7600
C                                               AND THE NEXT ONE.           INP 7650
C                                                                           INP 7700
C     3.3.B  X      1 TO 10   FLOAT POINT X,Y,AND Z COORDINATES OF A        INP 7750
C            Y     20 TO 29   FLOAT POINT PERIPHERY POINT ON CROSS-         INP 7800
C            Z     40 TO 49   FLOAT POINT SECTION NSEC. ONE POINT PER       INP 7850
C                                         CARD.                             INP 7900
C                                                                           INP 7950
C     FOR EACH CARD 3.3.A THE CARD 3.3.B IS PUNCHED NIP TIMES, EACH CARDINP 8000
C     WITH A DIFFERENT POINT. FOR THE GENERAL, NONSYMMETRY, OPTION,         INP 8050
C     POINTS ARE INPUT STARTING WITH POINT ON TOP OF CROSS SECTION          INP 8100
C     THEN MOVING SEQUENTIALLY CLOCKWISE AROUND PERIPHERY (AS SEEN          INP 8150
C     LOOKING AFT ALONG -X DIRECTION). NIP'TH POINT IS ALWAYS A REPEAT      INP 8200
C     OF THE FIRST POINT.  FOR SPECIAL SYMMETRY OPTION, POINTS ARE INPUTINP 8250
C     STARTING WITH THE POINT ON THE TOP AT SECTION LINE OF SYMMETRY.       INP 8300
C     OTHER POINTS ARE INPUT SEQUENTIALLY CLOCKWISE AROUND LEFT (-Y)        INP 8350
C     HALF OF THE CROSS SECTION. NIP'TH, LAST, POINT IS THE POINT AT        INP 8400
C     THE BOTTOM ON THE LINE OF SYMMETRY OF THE SECTION. (NOTE :            INP 8450
C     REGARDLESS OF CHOICE OF SYMMETRY OR NONSYMMETRY, ANY PERIPHERY        INP 8500
C     POINT MAY BE REPEATED TO CREATE A TRIANGULAR PANEL. USE AS MANY       INP 8550
C     REPEATS OF THE POINT AS THERE ARE SEPARATE TRIANGULAR PANELS          INP 8600
C     CONTAINING THIS CORNER POINT).                                        INP 8650
C     BODY INPUT GROUP SEQUENCE IS FIRST CARD 3.3.A THEN ITS NIP CARDS      INP 8700
C     3.3.B, SECOND CARD 3.3.A THEN ITS NIP CARDS 3.3.B ETC....             INP 8750
C                                                                           INP 8800
C                                                                           INP 8850
C     ******************************************************************INP 8900
C-----CARD SET 4 :  TEST RUN OPTION AND BODY PANEL GEOMETRY LIST            INP 8950
C                   OPTION. TEST RUN OPTION, IF CHOSEN, ALLOWS READING      INP 9000
```

```
C                   AND PRINTING OF ALL EIGHT INPUT DATA CARD SETS. BODYINP 9050
C                   SURFACE PANELS ARE GENERATED WITH LISTING OF EACH   INP 9100
C                   PANEL WITH ITS INDEX NUMBER, INPUT POINTS IT USES,  INP 9150
C                   AND PANEL AREA. PROGRAM STOPS AT THIS TIME WITHOUT  INP 9200
C                   SETTING UP OR SOLVING FLOW EQUATIONS. THIS OPTION ISINP 9250
C                   USED WHEN THE PANELING MUST BE CHECKED TO IDENTIFY  INP 9300
C                   PANELS AND INDEX NUMBERS TO BE USED FOR THROUGH     INP 9350
C                   FLOW PANELS. (TEST RUN IS VERY FAST AND INEXPENSIVE INP 9400
C                   SINCE NO EQUATION SOLVING IS DONE).                 INP 9450
C                   THE BODY PANEL GEOMETRY LIST OPTION, WHEN SPECIFIED,INP 9500
C                   WILL CAUSE THE PRINTING OF A TABLE OF PANEL         INP 9550
C                   INDEX NUMBERS, INPUT POINTS USED, AND AREAS FOR     INP 9600
C                   JOBS WHERE PROGRAM IS CARRIED TO COMPLETION.        INP 9650
C                   (NOTE : WHEN TEST RUN IS SPECIFIED, CHOICE OF PANEL INP 9700
C                   LIST OPTION IS IGNORED AND A PANEL LIST IS ALWAYS   INP 9750
C                   MADE. IF NORMAL COMPLETE SOLUTION RUN IS MADE, THE  INP 9800
C                   PANEL GEOMETRY LIST OPTION CHOICE WILL DETERMINE IF INP 9850
C                   PANEL LIST IS PRINTED.  (SET CONTAINS 2 CARDS).     INP 9900
C     ******************************************************************INP 9950
C                                                                       INP10000
C     CARD VARIABLE COLUMN(S) NUMBER TYPE DESCRIPTION                   INP10050
C     4.1  NLIST    1             INTEGER     =0 IF DESIRE PRINTOUT OF BODY INP10100
C                                                PANEL LIST DURING NORMAL   INP10150
C                                                RUN.                       INP10200
C                                               -OR-                        INP10250
C                                             =1 IF DO NOT WANT PANEL LIST  INP10300
C                                                PRINTED.                   INP10350
C                                                                           INP10400
C     4.2  NCALC    1             INTEGER     =0 IF THIS IS NORMAL RUN IN   INP10450
C                                                WHICH COMPLETE FLOW        INP10500
C                                                SOLUTION IS FOUND.         INP10550
C                                               -OR-                        INP10600
C                                             =1 IF TEST RUN IS TO BE MADE. INP10650
C                                                NO FLOW PREDICTED. PANEL   INP10700
C                                                LIST PRINTED. NLIST IGNOREDINP10750
C                                                (NCALC =1 WILL DEMAND      INP10800
```

```
C                                      SETTING NINFLO OF CARD SET INP10850
C                                      5 TO 0. SEE NOTES 1 AND 2  INP10900
C                                      OF CARD SET 5 BELOW).      INP10950
C                                                                 INP11000
C                                                                 INP11050
C     ************************************************************INP11100
C-----CARD SET 5 : INLET PANEL SPECIFICATION - INLET FLOW RATIO   INP11150
C             (SET CONTAINS 1 TO 501 CARDS).                      INP11200
C     ************************************************************INP11250
C                                                                 INP11300
C     CARD VARIABLE COLUMN(S) NUMBER TYPE DESCRIPTION             INP11350
C     5.1  NINFLO    1 TO 4     INTEGER     NUMBER OF PANELS, LISTED  INP11400
C                                      BELOW, WHICH ARE SPECIFIED AS INP11450
C                                      INLET OR OUTLET PANELS HAVING INP11500
C                                      A FLOW VELOCITY NORMAL THROUGHINP11550
C                                      THE SURFACE. NINFLO =0 IF NO  INP11600
C                                      SUCH PANELS SPECIFIED OR IF   INP11650
C                                      NCALC =1. SEE NOTE 1 BELOW. IFINP11700
C                                      SYMMETRIC BODY INPUT IS USED, INP11750
C                                      NINFLO IS AMOUNT OF SPECIAL   INP11800
C                                      PANELS ON THE LEFT HALF OF    INP11850
C                                      THE BODY. (MAXIMUM VALUE 250  INP11900
C                                      IF SYMMETRY OPTION CHOSEN).   INP11950
C                                      (MAXIMUM VALUE OF 500 IF      INP12000
C                                      NONSYMMETRY BODY INPUT USED). INP12050
C                                                                 INP12100
C     NOTE 1: FOR NCALC =1, TEST RUN IS BEING RUN TO IDENTIFY THE PANELSINP12150
C             AND INDEX NUMBERS. IT IS ASSUMED THAT DESIRED THROUGH-    INP12200
C             FLOW PANELS ARE UNKNOWN AT THIS TIME THUS NINFLO MUST BE  INP12250
C             SET EQUAL TO ZERO. ONLY WHEN NCALC =0 FOR NORMAL RUN CAN  INP12300
C             THE THROUGH FLOW PANELS BE SPECIFIED AND LISTED ON CARDS  INP12350
C             5.2 WHICH FOLLOW:                                   INP12400
C                                                                 INP12450
C     CARD VARIABLE COLUMN(S) NUMBER TYPE DESCRIPTION             INP12500
C     5.2  INDEX     1 TO 4     INTEGER     BODY PANEL INDEX IDENTIFYING  INP12550
C                                      NUMBER (AS GENERATED BY THE   INP12600
```

```
C                                                PROGRAM) OF EACH PANEL FOR     INP12650
C                                                WHICH A THROUGH FLOW IS        INP12700
C                                                DESIRED. IF SYMMETRIC BODY     INP12750
C                                                INPUT USED, SEE NOTE 2 BELOW.  INP12800
C                                                                               INP12850
C          FLRATO    20 TO 29  FLOAT POINT SPECIFIED THROUGH FLOW RATIO         INP12900
C                                                FOR PANEL, INDEX. IS DECIMAL   INP12950
C                                                FRACTION OF FREE STREAM        INP13000
C                                                VELOCITY, V.                   INP13050
C                                                NEGATIVE VALUE IS FOR OUTFLOW  INP13100
C                                                ON EXHAUST PANELS.             INP13150
C                                                POSITIVE VALUE IS FOR INFLOW   INP13200
C                                                ON INLET PANELS.               INP13250
C                                                ZERO VALUE (VALID) GIVES SOLIDINP13300
C                                                PANEL WITH NO THROUGH FLOW.    INP13350
C                                                (FLRATO MAY BE MORE THAN +1.0 INP13400
C                                                AND LESS THAN -1.0).           INP13450
C                                                                               INP13500
C     (CARD 5.2 IS REPEATED NINFLO TIMES AND DELETED IF NINFLO =0).             INP13550
C                                                                               INP13600
C     NOTE 2 : IF NSYMET = 0, SYMMETRY OF THROUGH FLOW PANELS IS ALSO           INP13650
C              REQUIRED.  WITH SYMMETRIC BODY INPUT, CARDS 5.2 LIST ONLYINP13700
C              THE THROUGH FLOW PANELS FOR THE LEFT (-Y) SIDE OF THE            INP13750
C              BODY.                                                            INP13800
C                                                                               INP13850
C                                                                               INP13900
C     ******************************************************************INP13950
C-----CARD SET 6 : WING INPUT DATA  (CONTAINS 1 CARD OR FROM 3 TO               INP14000
C                  8 CARDS).                                                    INP14050
C     ******************************************************************INP14100
C                                                                               INP14150
C     CARD VARIABLE COLUMN(S) NUMBER TYPE DESCRIPTION                           INP14200
C     6.1  NWING     1          INTEGER     =0 IF NO WING IS MODELED.           INP14250
C                                              -OR-                             INP14300
C                                           =1 IF WING IS MODELED (WING         INP14350
C                                              DESCRIBED ON CARDS 6.2 AND INP14400
```

```
C                                                6.3).                  INP14450
C                                                                       INP14500
C     6.2   XQR       1 TO 10   FLOAT POINT X,Y, AND Z COORDINATES OF   INP14550
C           YQR      11 TO 20   FLOAT POINT WING ROOT SECTION QUARTER   INP14600
C           ZQR      21 TO 30   FLOAT POINT CHORD LOCATION. (WING ROOT  INP14650
C                                           ASSUMED TO LIE INSIDE FUSELAGEINP14700
C                                           WITH Y COORDINATE ON FUSELAGE INP14750
C                                           CENTERLINE). COORDINATES ARE  INP14800
C                                           RELATIVE TO BODY AXIS SYSTEM. INP14850
C                                           YQR MUST BE 0.0 IF SYMMETRIC  INP14900
C                                           BODY INPUT (HENCE SYMMETRIC   INP14950
C                                           WING) OPTION USED. SEE NOTE 1 INP15000
C                                           BELOW.                        INP15050
C                                                                       INP15100
C           SPAN     31 TO 40   FLOAT POINT ACTUAL WING SPAN MEASURED IN  INP15150
C                                           PLANFORM VIEW AND PARALLEL TO INP15200
C                                           THE Y AXIS.                   INP15250
C                                                                       INP15300
C           CHORD    41 TO 50   FLOAT POINT ROOT WING CHORD               INP15350
C                                                                       INP15400
C                                                                       INP15450
C           DIHED    51 TO 60   FLOAT POINT WING DIHEDRAL ANGLE (DEGREES) INP15500
C                                           MEASURED FROM (X-Y) PLANE     INP15550
C                                           POSITIVE FOR WING TIPS UPWARD INP15600
C                                         . TOWARD NEGATIVE Z AXIS.       INP15650
C                                                                       INP15700
C           SWEEP    61 TO 70   FLOAT POINT SWEEP ANGLE (DEGREES) BETWEEN INP15750
C                                           WING QUARTER CHORD LINES AND  INP15800
C                                           THE Y AXIS.  AFT SWEEP IS     INP15850
C                                           POSITIVE VALUE.               INP15900
C                                                                       INP15950
C     (CARD 6.2 IS DELETED IF NWING =0).                                INP16000
C                                                                       INP16050
C     CARD VARIABLE COLUMN(S) NUMBER TYPE DESCRIPTION                   INP16100
C     6.3   CL        1 TO 10   FLOAT POINT WING LIFT COEFFICIENT AT      INP16150
C                                           INPUT BODY ANGLES OF ATTACK.  INP16200
```

```
C                                                                       INP16250
C     (CARD 6.3 IS REPEATED NALPHA TIMES. CL CARDS MUST BE IN THE       INP16300
C     SAME SEQUENCE AS THE BODY ALPHA-BETA CARDS, 2.3, SO THAT THE FIRSTINP16350
C     CL CORRESPONDS TO THE FIRST ALPHA ETC... CARD 6.3 DELETED IF      INP16400
C     NWING =0).                                                        INP16450
C                                                                       INP16500
C     NOTE 1 : SYMMETRIC BODY CONSIDERATIONS -- IF BODY SYMMETRIC INPUT INP16550
C              IS USED, THE WING MUST ALSO BE SYMMETRIC ABOUT THE BODY. INP16600
C              IF BODY IS SYMMETRIC BUT WING IS NOT SYMMETRIC ABOUT BODYINP16650
C              (EXAMPLE ISOLATED NACELLE ON LEFT WING - PLUS WING),     INP16700
C              THEN THE WING CAN NOT BE MODELED IF THIS BODY HAS BEEN   INP16750
C              INPUT USING THE SYMMETRIC OPTION. IN SUCH CASES, THE     INP16800
C              BODY MUST BE INPUT AS UNSYMMETRIC SO THAT WING MAY ALSO  INP16850
C              BE MODELED.                                              INP16900
C                                                                       INP16950
C                                                                       INP17000
C     ******************************************************************INP17050
C-----CARD SET 7 : PROPELLER PLANE DATA AND DATA DEFINING POINTS ON THE INP17100
C                  PLANE AT WHICH VELOCITIES AND FLOW ANGLES ARE TO BE  INP17150
C                  CALCULATED (CONTAINS 1 TO 4 CARDS)                   INP17200
C     ******************************************************************INP17250
C                                                                       INP17300
C     CARD VARIABLE COLUMN(S) NUMBER TYPE DESCRIPTION                   INP17350
C     7.1  NPOINT   1         INTEGER     =0 IF NO PROPELLER PLANE IS   INP17400
C                                            DEFINED AND NO CALCULATION INP17450
C                                            OF VELOCITIES IN A         INP17500
C                                            PROPELLER PLANE IS DONE.   INP17550
C                                          -OR-                         INP17600
C                                         =1 IF PROPELLER PLANE FLOW    INP17650
C                                            FIELD CALCULATIONS ARE     INP17700
C                                            MADE. DESCRIPTION OF       INP17750
C                                            PROPELLER PLANE AND POINT  INP17800
C                                            LOCATIONS FOLLOWS ON NEXT  INP17850
C                                            CARDS.                     INP17900
C                                                                       INP17950
C     7.2  XHUB     1 TO 10   FLOAT POINT X,Y, AND Z COORDINATES OF     INP18000
```

```
C        YHUB      11 TO 20   FLOAT POINT LOCATION OF PROPELLER PLANE   INP18050
C        ZHUB      21 TO 30   FLOAT POINT CENTER OR HUB IN TERMS OF BODYINP18100
C                                         REFERENCE AXES.               INP18150
C                                                                       INP18200
C        ALPHP     31 TO 40   FLOAT POINT BUILT-IN VERTICAL ANGULAR TILTINP18250
C                                         OF PROPELLER THRUST AXIS OR   INP18300
C                                         SHAFT (DEGREES) . IS THE ANGLEINP18350
C                                         BETWEEN BODY X AXIS AND THRUSTINP18400
C                                         AXIS AS SEEN PROJECTED ON BODYINP18450
C                                         X-Z PLANE. POSITIVE FOR TILT  INP18500
C                                         UPWARD TOWARD  -Z AXIS. (VALUEINP18550
C                                         OF 0.0 TO 360. OR 0.0 TO      INP18600
C                                         -360.).                       INP18650
C                                                                       INP18700
C        BETAP     41 TO 50   FLOAT POINT BUILT-IN SIDEWARD ANGULAR TILTINP18750
C                                         OF PROPELLER SHAFT OR THRUST  INP18800
C                                         AXIS (DEGREES). IS THE ANGLE  INP18850
C                                         BETWEEN BODY X AXIS AND THRUSTINP18900
C                                         AXIS AS SEEN PROJECTED ON A   INP18950
C                                         PLANE WHICH WOULD CONTAIN THE INP19000
C                                         BODY X AND Y AXES IF BODY     INP19050
C                                         WERE AT ZERO ALPHA. (IE. AS   INP19100
C                                         SEEN LOOKING VERTICALLY DOWN).INP19150
C                                         IT IS POSITIVE FOR THRUST     INP19200
C                                         AXIS ROTATION TOWARD RIGHT    INP19250
C                                         (+Y) AXIS. (VALUE OF 0.0 TO   INP19300
C                                         + 360. OR 0.0 TO -360.).      INP19350
C                                                                       INP19400
C        RADIUS    51 TO 60   FLOAT POINT REFERENCE PROPELLER PLANE     INP19450
C                                         RADIUS OR BLADE RADIUS.       INP19500
C                                                                       INP19550
C   (DELETE CARD 7.2 IF NPOINT =0).                                     INP19600
C                                                                       INP19650
C   CARD VARIABLE COLUMN(S) NUMBER TYPE DESCRIPTION                     INP19700
C   7.3  DPSI       1 TO 10   FLOAT POINT AZIMUTH ANGLE INCREMENT IN    INP19750
C                                         DEGREES BETWEEN SUCCESSIVE    INP19800
```

```
C                                          BLADE AZIMUTH POSITIONS ALONG INP19850
C                                          WHICH POINTS FOR FLOW         INP19900
C                                          PREDICTION ARE DEFINED.       INP19950
C                                          (FOR COMPLETE - EVEN AZIMUTH  INP20000
C                                          SPACING, THE VALUE OF DPSI    INP20050
C                                          MULTIPLIED BY AN INTEGER MUST INP20100
C                                          EQUAL 360.0 OTHERWISE         INP20150
C                                          UNEVEN AZIMUTHAL SPACING OF   INP20200
C                                          POINTS OCCURS). (DPSI VALUE   INP20250
C                                          MUST BE AT LEAST 1.0 AND      INP20300
C                                          NOT MORE THAN 360.).          INP20350
C                                                                        INP20400
C      DRADUS    11 TO 20  FLOAT POINT SPACING BETWEEN PROPELLER      INP20450
C                                          PLANE POINTS ALONG A CONSTANT INP20500
C                                          AZIMUTH LINE. ( IN FRACTION   INP20550
C                                          OF REFERENCE RADIUS VALUE.)   INP20600
C                                          DRADUS MUST BE ZERO OR        INP20650
C                                          POSITIVE VALUE. IT IS         INP20700
C                                          DIMENSIONLESS.                INP20750
C                                                                        INP20800
C      NRAD      21 AND 22 INTEGER         NUMBER OF PROPELLER PLANE     INP20850
C                                          POINTS LOCATED RADIALLY ALONG INP20900
C                                          EACH CONSTANT AZIMUTH.  FOR   INP20950
C                                          EACH AZIMUTH, FIRST POINT     INP21000
C                                          IS ALWAYS PLACED ON HUB.      INP21050
C                                          (NRAD HAS VALUE FROM 1 TO 51) INP21100
C                                          SEE CHOICE OF NRAD BELOW--    INP21150
C                                                                        INP21200
C     CHOICE OF NRAD :                                                   INP21250
C                                                                        INP21300
C     IF ((NRAD-1)*DRADUS=1            : PROPELLER PLANE POINTS ARE     INP21350
C                                        SPACED EVENLY FROM HUB OVER ONEINP21400
C                                        REFERENCE BLADE RADIUS.         INP21450
C     IF ((NRAD-1)*DRADUS) MORE THAN 1 : SPACES POINTS EVENLY FROM HUB  INP21500
C                                        TO SOME DISTANCE BEYOND THE     INP21550
C                                        REFERENCE BLADE RADIUS.         INP21600
```

```
C                                                  BLADE AZIMUTH POSITIONS ALONG INP19850
C                                                  WHICH POINTS FOR FLOW         INP19900
C                                                  PREDICTION ARE DEFINED.       INP19950
C                                                  (FOR COMPLETE - EVEN AZIMUTH  INP20000
C                                                  SPACING, THE VALUE OF DPSI    INP20050
C                                                  MULTIPLIED BY AN INTEGER MUST INP20100
C                                                  EQUAL 360.0 OTHERWISE         INP20150
C                                                  UNEVEN AZIMUTHAL SPACING OF   INP20200
C                                                  POINTS OCCURS). (DPSI VALUE   INP20250
C                                                  MUST BE AT LEAST 1.0 AND      INP20300
C                                                  NOT MORE THAN 360.).          INP20350
C                                                                                INP20400
C       DRADUS    11 TO 20   FLOAT POINT SPACING BETWEEN PROPELLER     INP20450
C                                                  PLANE POINTS ALONG A CONSTANT INP20500
C                                                  AZIMUTH LINE. ( IN FRACTION   INP20550
C                                                  OF REFERENCE RADIUS VALUE.)   INP20600
C                                                  DRADUS MUST BE ZERO OR        INP20650
C                                                  POSITIVE VALUE. IT IS         INP20700
C                                                  DIMENSIONLESS.                INP20750
C                                                                                INP20800
C       NRAD      21 AND 22 INTEGER      NUMBER OF PROPELLER PLANE     INP20850
C                                                  POINTS LOCATED RADIALLY ALONG INP20900
C                                                  EACH CONSTANT AZIMUTH.  FOR   INP20950
C                                                  EACH AZIMUTH, FIRST POINT     INP21000
C                                                  IS ALWAYS PLACED ON HUB.      INP21050
C                                                  (NRAD HAS VALUE FROM 1 TO 51) INP21100
C                                                  SEE CHOICE OF NRAD BELOW--    INP21150
C                                                                                INP21200
C     CHOICE OF NRAD :                                                           INP21250
C                                                                                INP21300
C     IF ((NRAD-1)*DRADUS=1            : PROPELLER PLANE POINTS ARE    INP21350
C                                        SPACED EVENLY FROM HUB OVER ONEINP21400
C                                        REFERENCE BLADE RADIUS.       INP21450
C     IF ((NRAD-1)*DRADUS) MORE THAN 1 : SPACES POINTS EVENLY FROM HUB INP21500
C                                        TO SOME DISTANCE BEYOND THE   INP21550
C                                        REFERENCE BLADE RADIUS.       INP21600
```

```
C     IF ((NRAD-1)*DRADUS) LESS THAN 1 : SPACES POINTS EVENLY FROM HUB  INP21650
C                                        TO A RADIAL POSITION INSIDE OF INP21700
C                                        THE REFERENCE BLADE RADIUS.    INP21750
C                                                                       INP21800
C     ( DELETE CARD 7.3 IF NPOINT =0).                                  INP21850
C                                                                       INP21900
C     CARD VARIABLE COLUMN(S) NUMBER TYPE DESCRIPTION                   INP21950
C     7.4  NPUNCH    1          INTEGER     PROPELLER PLANE FLOW FIELD  INP22000
C                                           PUNCHED OUTPUT OPTION       INP22050
C                                           =0 IF NO OUTPUT DATA IS TO BE INP22100
C                                              PUNCHED ON CARDS.        INP22150
C                                             -OR-                      INP22200
C                                           =1 IF PROPELLER PLANE OUTPUT INP22250
C                                              DATA WILL BE PUNCHED. THE INP22300
C                                              PUNCHED OUTPUT CONSISTS OF INP22350
C                                              LOCAL AXIAL AND TANGENTIAL INP22400
C                                              VELOCITIES, ANGLE OF     INP22450
C                                              ROTATIONAL FLOW AND THE  INP22500
C                                              RADIAL AND AZIMUTHAL     INP22550
C                                              LOCATION OF THE POINT.   INP22600
C                                              ONE CARD HAS ALL DATA FOR INP22650
C                                              A SINGLE POINT.          INP22700
C                                                                       INP22750
C     (DELETE CARD 7.4 IF NPOINT =0).                                   INP22800
C                                                                       INP22850
C     NOTE : CHOICE OF BODY SYMMETRY INPUT OPTION HAS NO INFLUENCE      INP22900
C            ON THE PROPELLER PLANE INPUT DATA. A PROPELLER PLANE OF ANYINP22950
C            SIZE LOCATION AND ORIENTATION MAY BE SPECIFIED REGARDLESS  INP23000
C            OF SYMMETRY OPTION CHOICE.                                 INP23050
C                                                                       INP23100
C                                                                       INP23150
C     ******************************************************************INP23200
C-----CARD SET 8 : ITERATION CONTROL CARD FOR SOLUTION OF BODY SURFACE  INP23250
C                  SOURCE STRENGTHS.  ( 1 CARD)                         INP23300
C     ******************************************************************INP23350
C                                                                       INP23400
```

```
C     CARD VARIABLE COLUMN(S) NUMBER TYPE DESCRIPTION                    INP23450
C     8.1  ITMAX     1 AND 2   INTEGER     MAXIMUM ALLOWED ITERATIONS IN INP23500
C                                          MATRIX SOLUTION PROCESS.      INP23550
C                                          ( ALLOWED VALUE FROM 1 TO 99).INP23600
C                                          RECOMMEND VALUE BETWEEN       INP23650
C                                          15 AND 20                     INP23700
C                                                                        INP23750
C          ERR       11 TO 20  FLOAT POINT EXPONENT ON THE VALUE OF      INP23800
C                                          LARGEST PERMITTED RELATIVE    INP23850
C                                          ERROR BETWEEN SOLUTION        INP23900
C                                          ITERATIONS. DEFINES CONVERG-  INP23950
C                                          ENCE LIMIT. (ERROR=10.**ERR)  INP24000
C                                          RECOMMEND VALUE ERR= -4.0.    INP24050
C                                                                        INP24100
C     (CARD 8.1 IS ALWAYS LAST CARD OF INPUT DATA DECK.)                 INP24150
C                                                                        INP24200
C     NOTE : ALL BODY ALPHA- BETA ORIENTATION CASES ARE SOLVED           INP24250
C            TOGETHER DURING ITERATIONS. SHOULD THE SOLUTION FOR ONE OR  INP24300
C            MORE ORIENTATIONS FAIL TO CONVERGE WITHIN THE ITERATION ANDINP24350
C            ERROR LIMITS, THE SUCCESSFUL SOLUTIONS FOR THE REMAINING    INP24400
C            ORIENTATIONS WILL ALSO BE REJECTED. THE PROGRAM WILL ABORT  INP24450
C            AND NO SOLUTIONS WILL BE GIVEN.                             INP24500
C                                                                        INP24550
C     ******************************************************************INP24600
C                                                                        INP24650
C     *******************************************                        INP24700
C     */////  END OF DATA INPUT DESCRIPTION /////*                       INP24750
C     *******************************************                        INP24800
C     //////////////////////////////////////////////////////////////////INP24850
C                                                                        INP24900
      SUBROUTINE INPUT(LLL,MAXALP,MAXINF,NIPMAX,NSECMA,NSECTO,NCOUNT,    INP24950
     +NFLAG,PX,PY,PZ,ITMAX,ERR)                                          INP25000
      DIMENSION NCOUNT(NSECMA),NFLAG(NSECMA),PX(NSECMA,NIPMAX),          INP25050
     1PY(NSECMA,NIPMAX),PZ(NSECMA,NIPMAX)                                INP25100
      COMMON /TITLE/ SYMBOL(160)                                         INP25150
      COMMON /INPUTS/ ALPHA(6),BETA(6),VX(6),VY(6),VZ(6),V,NALPHA        INP25200
```

```
      COMMON /OPTION/ NCALC,NLIST,NPUNCH                                INP25250
      COMMON /PROP/ ALPHP,BETAP,XHUB,YHUB,ZHUB,DPSI,DRADUS,RADIUS,NRAD, INP25300
     +NPOINT                                                            INP25350
      COMMON /SYMTRE/ NPNSYM(2596),INSOLV(1298),NSYMET                  INP25400
      COMMON /INLET/ FLRATO(500),INDEX(500),NINFLO                      INP25450
      COMMON /WING1/ CL(6),CHORD,DIHED,SPAN,SWEEP,XQR,YQR,ZQR,NWING     INP25500
      INTEGER STARS(17),STAR/'****'/                                    INP25550
C-----------------------------------------------------------------------INP25600
C                                                                       INP25650
C                                                                       INP25700
C     READ IN AND PRINT CARD SET 1 DATA :                               INP25750
C                                                                       INP25800
      DO 1 K=1,17                                                       INP25850
      STARS(K)=STAR                                                     INP25900
    1 CONTINUE                                                          INP25950
      READ(5,100)(SYMBOL(K),K=1,80)                                     INP26000
      READ(5,100)(SYMBOL(K),K=81,160)                                   INP26050
      WRITE(6,101)(STARS(J),J=1,17)                                     INP26100
      WRITE(6,102)                                                      INP26150
      WRITE(6,103)(STARS(J),J=1,17)                                     INP26200
      WRITE(6,104)(SYMBOL(J),J=1,80)                                    INP26250
      WRITE(6,104)(SYMBOL(J),J=81,160)                                  INP26300
C-----------------------------------------------------------------------INP26350
C                                                                       INP26400
C                                                                       INP26450
C     READ IN, CHECK VALIDITY, AND PRINT CARD SET 2 DATA :              INP26500
C                                                                       INP26550
      READ(5,200) V                                                     INP26600
      WRITE(6,201) V                                                    INP26650
      IF(V.LE.0.0) GO TO 35                                             INP26700
      READ(5,202) NALPHA                                                INP26750
      WRITE(6,203) NALPHA                                               INP26800
      IF(NALPHA.GT.MAXALP) GO TO 36                                     INP26850
      IF(NALPHA.LT.1) GO TO 37                                          INP26900
      DO 2 J=1,NALPHA                                                   INP26950
      READ(5,204) ALPHA(J),BETA(J)                                      INP27000
```

```
      WRITE(6,205) J,ALPHA(J),J,BETA(J)                                 INP27050
    2 CONTINUE                                                          INP27100
C-----------------------------------------------------------------------INP27150
C                                                                       INP27200
C                                                                       INP27250
C     READ IN, CHECK VALIDITY, AND PRINT CARD SET 3 DATA :              INP27300
C                                                                       INP27350
      READ(5,300) NSYMET                                                INP27400
      WRITE(6,301) NSYMET                                               INP27450
      IF(NSYMET.EQ.1) GO TO 4                                           INP27500
      IF(NSYMET.NE.0) GO TO 38                                          INP27550
      DO 3 J=1,NALPHA                                                   INP27600
      IF(BETA(J).EQ.0.0) GO TO 3                                        INP27650
      GO TO 39                                                          INP27700
    3 CONTINUE                                                          INP27750
C                                                                       INP27800
C     SYMMETRIC BODY INPUT IS USED THIS RUN. ADJUST MAX ALLOWED POINTS  INP27850
C     PER CROSS SECTION TO MAX ALLOWED POINTS ON THE LEFT HALF OF       INP27900
C     CROSS SECTION.                                                    INP27950
C                                                                       INP28000
      NIPMAX=(NIPMAX+1)/2                                               INP28050
C                                                                       INP28100
    4 READ(5,302) NSECTO                                                INP28150
      WRITE(6,303) NSECTO                                               INP28200
      IF(NSECTO.LT.3) GO TO 40                                          INP28250
      IF(NSECTO.GT.NSECMA) GO TO 41                                     INP28300
C                                                                       INP28350
C     CARDS 3.3.A AND 3.3.B FOR FIRST SECTION ARE READ :                INP28400
C                                                                       INP28450
      L=1                                                               INP28500
      READ(5,304) NSEC,NIP,NEND                                         INP28550
      NCOUNT(L)= NIP                                                    INP28600
      NFLAG(L)=NEND                                                     INP28650
      WRITE(6,305) NSEC,NIP,NEND                                        INP28700
      IF(NSEC.NE.L) GO TO 42                                            INP28750
      IF((NSYMET.EQ.0).AND.(NIP.LT.3)) GO TO 43                         INP28800
```

```
      IF((NSYMET.EQ.1).AND.(NIP.LT.4)) GO TO 44                         INP28850
      IF(NIP.GT.NIPMAX) GO TO 45                                        INP28900
      IF(NEND.EQ.1) GO TO 47                                            INP28950
      IF(NEND NE.0) GO TO 48                                            INP29000
      DO 5 K=1,NIP                                                      INP29050
      READ(5,306) X,Y,Z                                                 INP29100
      WRITE(6,307)X,Y,Z                                                 INP29150
      PX(L,K)=X                                                         INP29200
      PY(L,K)=Y                                                         INP29250
      PZ(L,K)=Z                                                         INP29300
    5 CONTINUE                                                          INP29350
C                                                                       INP29400
C     FOR SYMMETRIC INPUT CHECK THAT ALL POINTS ARE ON LEFT HALF AND    INP29450
C     CENTERLINE OF CROSS SECTION. FOR NONSYMMETRIC INPUT CHECK THAT    INP29500
C     LAST INPUT POINT OF SECTION IS REPEAT OF FIRST POINT ON SECTION.  INP29550
C                                                                       INP29600
      IF(NSYMET.EQ.0) GO TO 6                                           INP29650
      IF((PX(L,1).NE.PX(L,NIP)).OR.(PY(L,1).NE.PY(L,NIP)).OR.           INP29700
     1(PZ(L,1).NE.PZ(L,NIP))) GO TO 49                                  INP29750
      GO TO 8                                                           INP29800
    6 IF(PY(L,1).NE.0.0) GO TO 50                                       INP29850
      IF(PY(L,NIP).NE.0.0) GO TO 51                                     INP29900
      NIPLO=NIP-1                                                       INP29950
      DO 7 K=2,NIPLO                                                    INP30000
      IF(PY(L,K).GT.0.0) GO TO 52                                       INP30050
    7 CONTINUE                                                          INP30100
    8 CONTINUE                                                          INP30150
C                                                                       INP30200
C     READ CARDS 3.3.A AND 3.3.B FOR ALL REMAINING CROSS SECTIONS.      INP30250
C                                                                       INP30300
      DO 15 L=2,NSECTO                                                  INP30350
      READ(5,304) NSEC,NIP,NEND                                         INP30400
      NCOUNT(L)=NIP                                                     INP30450
      NFLAG(L)=NEND                                                     INP30500
      WRITE(6,305) NSEC,NIP,NEND                                        INP30550
      IF(NSEC.NE.L) GO TO 42                                            INP30600
```

```
      IF((NSYMET.EQ.0).AND.(NIP.LT.3)) GO TO 43                         INP30650
      IF((NSYMET.EQ.1).AND.(NIP.LT.4)) GO TO 44                         INP30700
      IF(NIP.GT.NIPMAX) GO TO 45                                        INP30750
      IF((NEND.NE.0).AND.(NEND.NE.1)) GO TO 53                          INP30800
      DO 9 K=1,NIP                                                      INP30850
      READ(5,306)X,Y,Z                                                  INP30900
      WRITE(6,307)X,Y,Z                                                 INP30950
      PX(L,K)=X                                                         INP31000
      PY(L,K)=Y                                                         INP31050
      PZ(L,K)=Z                                                         INP31100
    9 CONTINUE                                                          INP31150
C                                                                       INP31200
C     FOR SYMMETRIC INPUT CHECK THAT ALL POINTS ARE ON LEFT HALF OR     INP31250
C     CENTERLINE OF SECTION. ALSO FOR NONSYMMETRIC INPUT CHECK THAT     INP31300
C     LAST INPUT POINT IS A REPEAT IF FIRST INPUT POINT ON SECTION.     INP31350
C                                                                       INP31400
      IF(NSYMET.EQ.0) GO TO 10                                          INP31450
      IF((PX(L,1).NE.PX(L,NIP)).OR.(PY(L,1).NE.PY(L,NIP)).OR.           INP31500
     +(PZ(L,1).NE.PZ(L,NIP))) GO TO 49                                  INP31550
      GO TO 12                                                          INP31600
   10 IF(PY(L,1).NE.0.0) GO TO 50                                       INP31650
      IF(PY(L,NIP).NE.0.0) GO TO 51                                     INP31700
      NIPLO=NIP-1                                                       INP31750
      DO 11 K=2,NIPLO                                                   INP31800
      IF(PY(L,K).GT.0.0) GO TO 52                                       INP31850
   11 CONTINUE                                                          INP31900
   12 CONTINUE                                                          INP31950
C                                                                       INP32000
C     CROSS CHECK NIP,NEND AND ALL X,Y,Z POINTS OF CURRENT L TH INPUT   INP32050
C     SECTION TO VERIFY THEY ARE CONSISTENT WITH THE NIP, NEND AND      INP32100
C     X,Y,Z VALUES OF THE PREVIOUS INPUT SECTION (L-1).                 INP32150
C                                                                       INP32200
      IF((NFLAG(L-1).EQ.1).AND.(NFLAG(L).EQ.1)) GO TO 54                INP32250
      IF( NFLAG(L-1).EQ.1) GO TO 14                                     INP32300
C                                                                       INP32350
C     FOR NEND ON PREVIOUS SECTION OF ZERO, PANELS TO BE PLACED BETWEEN INP32400
```

```
C     CURRENT INPUT SECTION AND PREVIOUS ONE. SO CURRENT SECTION NIP    INP32450
C     VALUE MUST BE SAME AS NIP ON PREVIOUS SECTION. ALSO CHECK THAT NO INP32500
C     K TH SECTION POINT ON CURRENT INPUT SECTION IS IDENTICAL TO ITS   INP32550
C     CORRESPONDING K TH POINT ON THE PREVIOUS INPUT SECTION.           INP32600
C                                                                       INP32650
      IF(NCOUNT(L).NE.NCOUNT(L-1)) GO TO 55                             INP32700
      DO 13 K=1,NIP                                                     INP32750
      IF((PX(L,K).EQ.PX(L-1,K)).AND.(PY(L,K).EQ.PY(L-1,K)).AND.         INP32800
     +(PZ(L,K).EQ.PZ(L-1,K))) GO TO 56                                  INP32850
   13 CONTINUE                                                          INP32900
   14 CONTINUE                                                          INP32950
   15 CONTINUE                                                          INP33000
C-----------------------------------------------------------------------INP33050
C                                                                       INP33100
C                                                                       INP33150
C     READ IN, CHECK VALIDITY, AND PRINT CARD SET 4 DATA :              INP33200
C                                                                       INP33250
      READ(5,400) NLIST                                                 INP33300
      WRITE(6,401) NLIST                                                INP33350
      IF((NLIST.NE.0).AND.(NLIST.NE.1)) GO TO 57                        INP33400
      READ(5,400) NCALC                                                 INP33450
      WRITE(6,402) NCALC                                                INP33500
      IF((NCALC.NE.0).AND.(NCALC.NE.1)) GO TO 58                        INP33550
C-----------------------------------------------------------------------INP33600
C                                                                       INP33650
C                                                                       INP33700
C     READ IN, CHECK VALIDITY, AND PRINT CARD SET 5 DATA :              INP33750
C                                                                       INP33800
      READ(5,500) NINFLO                                                INP33850
      WRITE(6,501) NINFLO                                               INP33900
      IF((NCALC.EQ.1).AND.(NINFLO.NE.0)) GO TO 59                       INP33950
      IF(NINFLO.LT.0) GO TO 60                                          INP34000
      IF((NSYMET.EQ.0).AND.(NINFLO.GT.(MAXINF/2))) GO TO 61             INP34050
      IF(NINFLO.GT.MAXINF) GO TO 62                                     INP34100
      IF(NINFLO.EQ.0) GO TO 17                                          INP34150
      DO 16 J=1,NINFLO                                                  INP34200
```

```
      READ(5,502) INDEX(J),FLRATO(J)                                    INP34250
      WRITE(6,503) J,INDEX(J),J,FLRATO(J)                               INP34300
      IF((INDEX(J).LT.1).OR.(INDEX(J).GT.LLL)) GO TO 63                 INP34350
   16 CONTINUE                                                          INP34400
   17 CONTINUE                                                          INP34450
C-----------------------------------------------------------------------INP34500
C                                                                       INP34550
C                                                                       INP34600
C     READ IN, CHECK VALIDITY, AND PRINT CARD SET 6 DATA :              INP34650
C                                                                       INP34700
      READ(5,600) NWING                                                 INP34750
      WRITE(6,601) NWING                                                INP34800
      IF((NWING.NE.0).AND.(NWING.NE.1)) GO TO 64                        INP34850
      IF(NWING.EQ.0) GO TO 19                                           INP34900
      READ(5,602) XQR,YQR,ZQR,SPAN,CHORD,DIHED,SWEEP                    INP34950
      WRITE(6,603) XQR,YQR,ZQR,SPAN,CHORD,DIHED,SWEEP                   INP35000
      IF((NSYMET.EQ.0).AND.(YQR.NE.0.0)) GO TO 73                       INP35050
      IF(SPAN.LE.0.0) GO TO 65                                          INP35100
      IF(CHORD.LE.0.0) GO TO 74                                         INP35150
      DO 18 J=1,NALPHA                                                  INP35200
      READ(5,604) CL(J)                                                 INP35250
      WRITE(6,605) J,CL(J)                                              INP35300
   18 CONTINUE                                                          INP35350
   19 CONTINUE                                                          INP35400
C-----------------------------------------------------------------------INP35450
C                                                                       INP35500
C                                                                       INP35550
C     READ IN, CHECK VALIDITY, AND PRINT CARD SET 7 DATA :              INP35600
C                                                                       INP35650
      READ(5,700) NPOINT                                                INP35700
      WRITE(6,701) NPOINT                                               INP35750
      IF((NPOINT.NE.0).AND.(NPOINT.NE.1)) GO TO 66                      INP35800
      IF(NPOINT.EQ.0) GO TO 21                                          INP35850
      READ(5,702) XHUB,YHUB,ZHUB,ALPHP,BETAP,RADIUS                     INP35900
      WRITE(6,703) XHUB,YHUB,ZHUB,ALPHP,BETAP,RADIUS                    INP35950
      IF(RADIUS.LE.0.0) GO TO 67                                        INP36000
```

```
      READ(5,704) DPSI,DRADUS,NRAD                                      INP36050
      WRITE(6,705) DPSI,DRADUS,NRAD                                     INP36100
      IF(DPSI.LE.0.0) GO TO 68                                          INP36150
      IF(DPSI.GT.360.) GO TO 75                                         INP36200
      IF(DRADUS.LT.0.0) GO TO 69                                        INP36250
      IF(DRADUS.NE.0.0) GO TO 20                                        INP36300
      WRITE(6,706)                                                      INP36350
   20 IF(NRAD.LE.0) GO TO 70                                            INP36400
      IF(NRAD.GT.51) GO TO 76                                           INP36450
      READ(5,707) NPUNCH                                                INP36500
      WRITE(6,708) NPUNCH                                               INP36550
      IF((NPUNCH.NE.0).AND.(NPUNCH.NE.1)) GO TO 71                      INP36600
   21 CONTINUE                                                          INP36650
C-----------------------------------------------------------------------INP36700
C                                                                       INP36750
C                                                                       INP36800
C     READ IN, CHECK VALIDITY, AND PRINT CARD SET 8 DATA :              INP36850
C                                                                       INP36900
      READ(5,800) ITMAX,ERR                                             INP36950
      WRITE(6,801) ITMAX,ERR                                            INP37000
      IF(ITMAX.LE.0) GO TO 72                                           INP37050
      IF(ERR.LT.0.0) GO TO 22                                           INP37100
      WRITE(6,802)                                                      INP37150
   22 WRITE(6,803)                                                      INP37200
      WRITE(6,804)(STARS(J),STARS(J),J=1,15)                            INP37250
C-----------------------------------------------------------------------INP37300
C                                                                       INP37350
C     WRITE THE OPTIONS SPECIFIED BY INPUT FOR THIS JOB.                INP37400
C                                                                       INP37450
      WRITE(6,1000)                                                     INP37500
      WRITE(6,1001)                                                     INP37550
      WRITE(6,1002)                                                     INP37600
      IF(NCALC.EQ.0) GO TO 24                                           INP37650
      WRITE(6,1003)                                                     INP37700
      IF(NSYMET.EQ.1) GO TO 23                                          INP37750
      WRITE(6,1004)                                                     INP37800
```

```
      GO TO 34                                                          INP37850
   23 WRITE(6,1005)                                                     INP37900
      GO TO 34                                                          INP37950
   24 WRITE(6,1006)                                                     INP38000
      WRITE(6,1007) NALPHA                                              INP38050
      IF(NSYMET.EQ.1) GO TO 25                                          INP38100
      WRITE(6,1008)                                                     INP38150
      GO TO 26                                                          INP38200
   25 WRITE(6,1009)                                                     INP38250
   26 IF(NINFLO.NE.0) GO TO 27                                          INP38300
      WRITE(6,1010)                                                     INP38350
      GO TO 29                                                          INP38400
   27 IF(NSYMET.EQ.1) GO TO 28                                          INP38450
      WRITE(6,1011) NINFLO                                              INP38500
      GO TO 29                                                          INP38550
   28 WRITE(6,1012) NINFLO                                              INP38600
   29 IF(NWING.EQ.1) GO TO 30                                           INP38650
      WRITE(6,1013)                                                     INP38700
      GO TO 31                                                          INP38750
   30 WRITE(6,1014)                                                     INP38800
   31 IF(NPOINT.EQ.1) GO TO 32                                          INP38850
      WRITE(6,1015)                                                     INP38900
      GO TO 34                                                          INP38950
   32 WRITE(6,1016)                                                     INP39000
      IF(NPUNCH.EQ.1) GO TO 33                                          INP39050
      WRITE(6,1017)                                                     INP39100
      GO TO 34                                                          INP39150
   33 WRITE(6,1018)                                                     INP39200
   34 WRITE(6,1019)(STARS(J),STARS(J),J=1,16)                           INP39250
      RETURN                                                            INP39300
C-----------------------------------------------------------------------INP39350
C                                                                       INP39400
C     INPUT ERROR CHECKS AND PROGRAM STOPS                              INP39450
C                                                                       INP39500
   35 WRITE(6,900) V                                                    INP39550
      STOP                                                              INP39600
```

```
36 WRITE(6,901) NALPHA, MAXALP                                      INP39650
   STOP                                                             INP39700
37 WRITE(6,902) NALPHA                                              INP39750
   STOP                                                             INP39800
38 WRITE(6,903) NSYMET                                              INP39850
   STOP                                                             INP39900
39 WRITE(6,904)                                                     INP39950
   STOP                                                             INP40000
40 WRITE(6,905) NSECTO                                              INP40050
   STOP                                                             INP40100
41 WRITE(6,906) NSECTO,NSECMA                                       INP40150
   STOP                                                             INP40200
42 WRITE(6,907) NSEC                                                INP40250
   STOP                                                             INP40300
43 WRITE(6,908) NIP                                                 INP40350
   STOP                                                             INP40400
44 WRITE(6,909) NIP                                                 INP40450
   STOP                                                             INP40500
45 IF(NSYMET.EQ.1) GO TO 46                                         INP40550
   WRITE(6,910) NIP,NIPMAX                                          INP40600
   STOP                                                             INP40650
46 WRITE(6,911) NIP,NIPMAX                                          INP40700
   STOP                                                             INP40750
47 WRITE(6,912) NEND                                                INP40800
   STOP                                                             INP40850
48 WRITE(6,913) NEND                                                INP40900
   STOP                                                             INP40950
49 WRITE(6,914)                                                     INP41000
   STOP                                                             INP41050
50 WRITE(6,915)                                                     INP41100
   STOP                                                             INP41150
51 WRITE(6,916)                                                     INP41200
   STOP                                                             INP41250
52 WRITE(6,917)                                                     INP41300
   STOP                                                             INP41350
53 WRITE(6,918) NEND                                                INP41400
```

```
      STOP                                                              INP41450
   54 WRITE(6,919)                                                      INP41500
      STOP                                                              INP41550
   55 WRITE(6,920)                                                      INP41600
      STOP                                                              INP41650
   56 WRITE(6,921)                                                      INP41700
      STOP                                                              INP41750
   57 WRITE(6,922)                                                      INP41800
      STOP                                                              INP41850
   58 WRITE(6,923)                                                      INP41900
      STOP                                                              INP41950
   59 WRITE(6,924)                                                      INP42000
      STOP                                                              INP42050
   60 WRITE(6,925)                                                      INP42100
      STOP                                                              INP42150
   61 MAXIN=MAXINF/2                                                    INP42200
      WRITE(6,926) NINFLO,MAXIN                                         INP42250
      STOP                                                              INP42300
   62 WRITE(6,927) NINFLO,MAXINF                                        INP42350
      STOP                                                              INP42400
   63 WRITE(6,928) LLL                                                  INP42450
      STOP                                                              INP42500
   64 WRITE(6,929)                                                      INP42550
      STOP                                                              INP42600
   65 WRITE(6,930)                                                      INP42650
      STOP                                                              INP42700
   66 WRITE(6,931)                                                      INP42750
      STOP                                                              INP42800
   67 WRITE(6,932)                                                      INP42850
      STOP                                                              INP42900
   68 WRITE(6,933)                                                      INP42950
      STOP                                                              INP43000
   69 WRITE(6,934)                                                      INP43050
      STOP                                                              INP43100
   70 WRITE(6,935)                                                      INP43150
      STOP                                                              INP43200
```

```
   71 WRITE(6,936)                                                      INP43250
      STOP                                                              INP43300
   72 WRITE(6,937)                                                      INP43350
      STOP                                                              INP43400
   73 WRITE(6,938)                                                      INP43450
      STOP                                                              INP43500
   74 WRITE(6,939)                                                      INP43550
      STOP                                                              INP43600
   75 WRITE(6,940)                                                      INP43650
      STOP                                                              INP43700
   76 WRITE(6,941)                                                      INP43750
      STOP                                                              INP43800
C-----------------------------------------------------------------------INP43850
C                                                                       INP43900
C     FORMATS                                                           INP43950
C                                                                       INP44000
  100 FORMAT(80A1)                                                      INP44050
  101 FORMAT('1',T33,17A4)                                              INP44100
  102 FORMAT(' ',T33,'* LISTING OF ALL INPUT DATA CARDS TO ALLOW USER CHINP44150
     1ECKING OF ERRORS *')                                              INP44200
  103 FORMAT(' ',T33,17A4//)                                            INP44250
  104 FORMAT(' ',80A1)                                                  INP44300
  200 FORMAT(F10.2)                                                     INP44350
  201 FORMAT('0','V= ',F13.7)                                           INP44400
  202 FORMAT(I1)                                                        INP44450
  203 FORMAT(' ','NALPHA= ',I2)                                         INP44500
  204 FORMAT(F10.2,T20,F10.2)                                           INP44550
  205 FORMAT(' ' 'ALPHA(',I2,')= ',F13.7,5X,'BETA(',I2,')= ',F13.7)     INP44600
  300 FORMAT(I1)                                                        INP44650
  301 FORMAT('0' 'NSYMET= ',I2)                                         INP44700
  302 FORMAT(I2)                                                        INP44750
  303 FORMAT('0','NSECTO= ',I2)                                         INP44800
  304 FORMAT(I2,T10,I2,T30,I1)                                          INP44850
  305 FORMAT('0','NSEC= ',I2,5X,'NIP= ',I2,5X,'NEND= ',I1)              INP44900
  306 FORMAT(F10.2,T20,F10.2,T40,F10.2)                                 INP44950
  307 FORMAT(' ','X= ',F13.7,10X,'Y= ',F13.7,10X,'Z= ',F13.7)           INP45000
```

```
400 FORMAT(I1)                                                         INP45050
401 FORMAT('0','NLIST= ',I1)                                           INP45100
402 FORMAT(' ','NCALC= ',I1)                                           INP45150
500 FORMAT(I4)                                                         INP45200
501 FORMAT('0','NINFLO= ',I5)                                          INP45250
502 FORMAT(I4,T20,F10.2)                                               INP45300
503 FORMAT(' ','INDEX(',I4,')= ',I5,5X,'FLRATO(',I4,')= ',F13.7)       INP45350
600 FORMAT(I1)                                                         INP45400
601 FORMAT('0','NWINC= ',I2)                                           INP45450
602 FORMAT(7F10.2)                                                     INP45500
603 FORMAT(' ','XQR=',F13.7,1X,'YQR=',F13.7,1X,'ZQR=',F13.7,1X,'SPAN='INP45550
   1,F13.7,1X,'CHORD=',F13.7,1X,'DIHED=',F11.5,1X,'SWEEP=',F11.5)      INP45600
604 FORMAT(F10.2)                                                      INP45650
605 FORMAT(' ','CL(',I2,')= ',F13.7)                                   INP45700
700 FORMAT(I1)                                                         INP45750
701 FORMAT('0','NPOINT= ',I2)                                          INP45800
702 FORMAT(6F10.2)                                                     INP45850
703 FORMAT(' ','XHUB= ',F13.7,2X,'YHUB= ',F13.7,2X,'ZHUB= ',F13.7,2X, INP45900
   1'ALPHP= ',F13.7,2X,'BETAP= ',F13.7,2X,'RADIUS= ',F13.7)            INP45950
704 FORMAT(2F10.2,T21,I2)                                              INP46000
705 FORMAT(' ','DPSI= ',F13.7,5X,'DRADUS= ',F13.7,5X,'NRAD= ',I4)      INP46050
706 FORMAT('0','---WARNING--- DRADUS= 0 SO ONLY PROP HUB POINT WILL BEINP46100
   + PREDICTED (SEVERAL TIMES)  -CONTINUE.')                           INP46150
707 FORMAT(I1)                                                         INP46200
708 FORMAT(' ','NPUNCH= ',I2)                                          INP46250
800 FORMAT(I2,T11,F10.2)                                               INP46300
801 FORMAT('0','ITMAX= ',I3,5X,'ERR= ',F13.7)                          INP46350
802 FORMAT('0','---WARNING--- VALUE OF ERR POSITIVE OR ZERO GIVES RELAINP46400
   1TIVE ERROR IN SOLUTION GREATER THAN UNITY.'/,' ',T15,'VALUE OF ERRINP46450
   2 SHOULD BE MADE NEGATIVE. -- RUN CONTINUES AS IS , HOWEVER.')      INP46500
803 FORMAT('0','////END OF INPUT DATA -- ALL DATA READ SUCCESSFULLY///INP46550
   1/')                                                                INP46600
804 FORMAT('0',30A4/)                                                  INP46650
900 FORMAT('0',5X,'*** ERROR *** : V= ',F13.7,' ZERO OR NEGATIVE - RUNINP46700
   1 STOPPED.')                                                        INP46750
901 FORMAT('0',5X,'*** ERROR *** : NALPHA= ',I2,' IS GREATER THAN MAX INP46800
```

```
     1VALUE= ',I2,'ALLOWED - RUN STOPPED.')                          INP46850
 902 FORMAT('0',5X,'*** ERROR *** : NALPHA= ',I2,' IS ZERO OR NEGATIVE INP46900
     1- INVALID. RUN STOPPED.')                                          INP46950
 903 FORMAT('0',5X,'*** ERROR *** : NSYMET= ',I2,' IS NOT 0 OR 1 - INVAINP47000
     1LID. RUN STOPPED.')                                                INP47050
 904 FORMAT('0',5X,'*** ERROR *** : ATTEMPT TO USE SYMMETRIC INPUT OPTIINP47100
     1ON, BUT ONE OR MORE BETAS ARE NONZERO - INVALID. RUN STOPPED.')    INP47150
 905 FORMAT('0',5X,'*** ERROR *** : NUMBER OF SECTIONS, NSECTO= ',I2,    INP47200
     1', IS LESS THAN 3 - INVALID. RUN STOPPED.')                        INP47250
 906 FORMAT('0',5X,'*** ERROR *** : NUMBER OF SECTIONS, NSEC= ',I3,      INP47300
     1', IS GREATER THAN MAX VALUE= ',I3,' ALLOWED. RUN STOPPED.')       INP47350
 907 FORMAT('0',5X,'*** ERROR *** : SECTION NUMBER, NSEC= ',I2,', IS NOINP47400
     1T IN PROPER ASCENDING SEQUENCE ORDER - RUN STOPPED.')              INP47450
 908 FORMAT('0',5X,'*** ERROR *** : FOR SYMMETRIC BODY INPUT, NUMBER OFINP47500
     1 HALF SECTION POINTS, NIP= ',I2,', IS LESS THAN 3 - INVALID. RUN SINP47550
     2TOPPED.')                                                          INP47600
 909 FORMAT('0',5X,'*** ERROR *** : FOR NONSYMMETRIC BODY INPUT, NUMBERINP47650
     1 OF SECTION POINTS, NIP= ',I2,', IS LESS THAN 4 - INVALID RUN STOPINP47700
     2PED.')                                                             INP47750
 910 FORMAT('0',5X,'*** ERROR *** : FOR SYMMETRIC BODY INPUT, NUMBER OFINP47800
     1 HALF SECTION POINTS, NIP= ',I2,', IS MORE THAN MAX,',I3,', ALLOWEINP47850
     2D. STOP.')                                                         INP47900
 911 FORMAT('0',5X,'*** ERROR *** : FOR NONSYMMETRIC BODY INPUT, NUMBERINP47950
     1 OF SECTION POINTS, NIP= ',I2,', IS MORE THAN MAX,',I3,', ALLOWED.INP48000
     2 STOP.')                                                           INP48050
 912 FORMAT('0',5X,'*** ERROR *** : NEND= ',I2,' FOR THE FIRST INPUT BOINP48100
     1DY SECTION IS INVALID - RUN STOPPED.')                             INP48150
 913 FORMAT('0',5X,'*** ERROR *** : FOR FIRST INPUT BODY SECTION, NEND=INP48200
     1 ',I2,' IS NOT THE REQUIRED ZERO. RUN STOPPED.')                   INP48250
 914 FORMAT('0' 5X,'*** ERROR *** : FOR NONSYMMETRIC BODY INPUT OF THISINP48300
     1 WHOLE SECTION''S POINTS, THE LAST POINT OF SECTION'/,' ',T22,     INP48350
     2'HAS NOT BEEN MADE A REPEAT OF THE FIRST POINT - INVALID. RUN STOPINP48400
     3PED.')                                                             INP48450
 915 FORMAT('0',5X,'*** ERROR *** : FOR SYMMETRIC BODY INPUT OF THE HALINP48500
     1F SECTION, THIS FIRST POINT IS NOT'/,' ',T22,'ON THE SECTION CENTEINP48550
     2RLINE (IE. Y=0.) - POINT IS INVALID. STOP.')                       INP48600
```

```
916 FORMAT('0',5X,'*** ERROR *** : FOR SYMMETRIC BODY INPUT OF THIS HAINP48650
   1LF SECTION, THIS LAST POINT IS NOT'/,' ',T22,'ON THE SECTION CENTEINP48700
   2RLINE (IE. Y=0.) - POINT IS INVALID. STOP.')                      INP48750
917 FORMAT('0',5X,'*** ERROR *** : FOR SYMMETRIC (LEFT SIDE ONLY) SECTINP48800
   1ION INPUT, ONE OR MORE POINTS LYING ON'/,' ',T22,'THE RIGHT OR (+YINP48850
   2) SIDE OF SECTION HAVE BEEN LISTED. THESE POINTS INVALID. STOP.') INP48900
918 FORMAT('0',5X,'*** ERROR *** : ON THIS SECTION, NEND= ',I2,       INP48950
   1' IS INVALID - MUST BE 0 OR 1. RUN STOPPED.')                     INP49000
919 FORMAT('0',5X,'*** ERROR *** : TWO CONSECUTIVE CROSS SECTIONS HAVEINP49050
   1 NEND= 1. - INVALID. RUN STOPPED.')                               INP49100
920 FORMAT('0',5X,'*** ERROR *** : NIP VALUE ON THIS SECTION MUST EQUAINP49150
   1L NIP OF PRIOR SECTION AND DOES NOT - INVALID. RUN STOPPED.')     INP49200
921 FORMAT('0',5X,'*** ERROR *** : NEND= 0 ON PREVIOUS SECTION SO PANEINP49250
   1LING DESIRED BETWEEN PREVIOUS SECTION AND THIS SECTION, BUT'/,' ',INP49300
   2T22,'ONE OR MORE POINTS HAVE BEEN REPEATED IDENTICALLY ON THESE TWINP49350
   3O SECTIONS - INVALID POINTS. RUN STOPPED.')                       INP49400
922 FORMAT('0',5X,'*** ERROR *** : NLIST IS NOT 0 OR 1 - INVALID. RUN INP49450
   1STOPPED.')                                                        INP49500
923 FORMAT('0' 5X,'*** ERROR *** : NCALC IS NOT 0 OR 1 - INVALID. RUN INP49550
   1STOPPED.')                                                        INP49600
924 FORMAT('0',5X,'*** ERROR *** : NINFLO MUST BE 0 BECAUSE TEST CASE INP49650
   1OPTION (NCALC= 1) WAS SPECIFIED. RUN STOPPED.')                   INP49700
925 FORMAT('0' 5X,'*** ERROR *** : NEGATIVE NINFLO VALUE IS INVALID. RINP49750
   1UN STOPPED.')                                                     INP49800
926 FORMAT('0' 5X,'*** ERROR *** : NINFLO= ',I4,' IS MORE THAN MAX VALINP49850
   1UE ',I4,', ALLOWED ON LEFT SIDE OF BODY FOR SYMMETRY OPTION. RUN SINP49900
   2TOPPED.')                                                         INP49950
927 FORMAT('0' 5X,'*** ERROR *** : NINFLO= ',I4,' IS MORE THAN MAX VALINP50000
   1UE,',I4,', ALLOWED WHEN NONSYMMETRIC BODY OPTION USED. RUN STOPPEDINP50050
   2.')                                                               INP50100
928 FORMAT('0',5X,'*** ERROR *** : INDEX VALUE IS NEGATIVE, ZERO, OR MINP50150
   1ORE THAN THE HIGHEST POSSIBLE PANEL NUMBER OF,',I5,'. RUN STOPPED.INP50200
   2')                                                                INP50250
929 FORMAT('0',5X,'*** ERROR *** : NWING IS NOT 0 OR 1 - INVALID. RUN INP50300
   1STOPPED.')                                                        INP50350
930 FORMAT('0',5X,'*** ERROR *** : SPAN IS ZERO OR NEGATIVE - INVALID.INP50400
```

```
    1 RUN STOPPED.')                                                    INP50450
  931 FORMAT('0',5X,'*** ERROR *** : NPOINT IS NOT 0 OR 1 - INVALID. RUNINP50500
    1 STOPPED.')                                                        INP50550
  932 FORMAT('0',5X,'*** ERROR *** : REFERENCE RADIUS IS ZERO OR NEGATIVINP50600
    1E - INVALID. RUN STOPPED.')                                        INP50650
  933 FORMAT('0',5X,'*** ERROR *** : DPSI VALUE IS ZERO OR NEGATIVE - ININP50700
    1VALID. RUN STOPPED.')                                              INP50750
  934 FORMAT('0',5X,'*** ERROR *** : DRADUS VALUE IS NEGATIVE - INVALID.INP50800
    1 RUN STOPPED.')                                                    INP50850
  935 FORMAT('0',5X,'*** ERROR *** : NRAD IS ZERO OR NEGATIVE - INVALID.INP50900
    1 RUN STOPPED.')                                                    INP50950
  936 FORMAT('0' 5X,'*** ERROR *** : NPUNCH IS NOT 0 OR 1 - INVALID. RUNINP51000
    1 STOPPED.')                                                        INP51050
  937 FORMAT('0',5X,'*** ERROR *** : ITMAX IS ZERO OR NEGATIVE - INVALIDINP51100
    1. RUN STOPPED.')                                                   INP51150
  938 FORMAT('0',5X,'*** ERROR *** : YQR MUST BE 0.0 THIS RUN BECAUSE SYINP51200
    1MMETRIC BODY INPUT WAS USED REQUIRING THAT WING MUST'/,' ',T22,    INP51250
    2'ALSO BE SYMMETRIC IF INPUT. SO WING ROOT MUST LIE ON AXIS OF SYMMINP51300
    3ETRY. YQR INVALID. RUN STOPPED.')                                  INP51350
  939 FORMAT('0',5X,' *** ERROR *** : CHORD VALUE IS ZERO OR NEGATIVE - INP51400
    1INVALID. RUN STOPPED.')                                            INP51450
  940 FORMAT('0',5X,'*** ERROR *** : DPSI VALUE MORE THAN 360.0 IS INVALINP51500
    1ID - RUN STOPPED.')                                                INP51550
  941 FORMAT('0',5X,'*** ERROR *** : VALUE OF NRAD IS TOO LARGE - RUN STINP51600
    1OPPED.')                                                           INP51650
 1000 FORMAT('0',T42,12A4)                                              INP51700
 1001 FORMAT(' ' T42,'*  OPTIONS REQUESTED AND CONTENTS OF THIS RUN  *')INP51750
 1002 FORMAT(' ',T42,12A4/)                                             INP51800
 1003 FORMAT('0',T20,'1) TEST RUN OPTION USED - ONLY BODY PANEL GEOMETRYINP51850
    1 FOUND AND LISTED. NO FLOW CALCULATIONS.')                         INP51900
 1004 FORMAT('0',T20,'2) SYMMETRIC BODY (LEFT SIDE) INPUT OPTION USED. EINP51950
    1NTIRE BODY PANELED AND ALL PANELS LISTED.'/)                       INP52000
 1005 FORMAT('0',T20,'2) NONSYMMETRIC BODY INPUT OPTION USED.'/)        INP52050
 1006 FORMAT('0',T20,'1) NORMAL RUN OPTION USED - BODY PANELING GENERATEINP52100
    1D. EQUATIONS SOLVED, AND FLOW PREDICTIONS MADE :')                 INP52150
 1007 FORMAT('0',T20,'2) SURFACE FLOW CALCULATIONS MADE FOR ALL',I3,' ININP52200
```

```
     1PUT BODY (ALPHA-BETA) ORIENTATIONS.')                        INP52250
 1008 FORMAT('0',T20,'3) SYMMETRIC BODY (LEFT SIDE) INPUT OPTION USED.')INP52300
 1009 FORMAT('0',T20,'3) NONSYMMETRIC BODY (BOTH SIDES) INPUT OPTION USEINP52350
     1D.')                                                              INP52400
 1010 FORMAT('0',T20,'4) NO INLET OR OUTLET PANELS SPECIFIED.')         INP52450
 1011 FORMAT('0',T20,'4) ',I4,' INLET AND/OR OUTLET PANELS ON LEFT HALF INP52500
     1OF SYMMETRIC BODY ARE SPECIFIED.')                                INP52550
 1012 FORMAT('0',T20,'4) ' I4,' INLET AND/OR OUTLET PANELS SPECIFIED OVEINP52600
     1R THE COMPLETE BODY.')                                            INP52650
 1013 FORMAT('0',T20,'5) NO WING (OR CL) IS MODELED.')                  INP52700
 1014 FORMAT('0',T20,'5) WING (WITH PROPER CL FOR EACH ALPHA) IS MODELEDINP52750
     1.')                                                               INP52800
 1015 FORMAT('0',T20,'6) NO PROPELLER PLANE FLOW CALCULATIONS MADE ON THINP52850
     1IS JOB.'/)                                                        INP52900
 1016 FORMAT('0',T20,'6) PROPELLER PLANE FLOW CALCULATION MADE FOR ALL BINP52950
     1ODY (ALPHA-BETA) ORIENTATIONS.')                                  INP53000
 1017 FORMAT('0' T20,'7) NO PUNCHED PROPELLER PLANE FLOW OUTPUT.'/)     INP53050
 1018 FORMAT('0',T20,'7) PUNCHED CARD OUTPUT GIVEN ON PROPELLER PLANE FLINP53100
     1OW FOR ALL BODY (ALPHA-BETA) ORIENTATIONS.'/)                     INP53150
 1019 FORMAT('0',32A4/)                                                 INP53200
C                                                                       INP53250
      END                                                               INP53300
C                                                                       INP53350
C                                                                       INP53400
C     ******************************************************************INP53450
C                                                                       INP53500
C                                                                       INP53550
```

```
      SUBROUTINE VCOMP
C
C     ******************************************************************
C     ******************************************************************
C     COMPUTES X,Y,AND Z COMPONENTS OF FREESTREAM VELOCITY : VX(I),
C     VY(I),AND VZ(I) AS SEEN IN THE BODY REFERENCE AXIS SYSTEM. A SET
C     OF COMPONENTS IS FOUND FOR EACH INPUT I TH PAIR OF BODY ANGLES OF
C     ATTACK, ALPHA(I), AND SIDESLIP, BETA(I). EXPRESSIONS WERE
C     DERIVED FROM THE FOLLOWING AXIS ROTATIONS.
C     1- INERTIAL WIND AXES AND FIXED BODY AXES COINCIDE INITIALLY
C     2- ROTATE ALPHA DEGREES ABOUT Y(BODY) AXIS.
C     3- ROTATE IN RIGHT HAND SENSE ABOUT Z(BODY) AXIS BY EULER
C        ANGLE, ANADA.
C     ANADA MUST BE OF SUFFICIENT AMOUNT TO PRODUCE THE DESIRED
C     SIDESLIP ANGLE, BETA.
C
C     ANADA IS CALCULATED IN SUBROUTINE EULER USING THE GIVEN ALPHA AND
C     BETA.
C     ******************************************************************
C     ******************************************************************
C
      COMMON /INPUTS/ ALPHA(6),BETA(6),VX(6),VY(6),VZ(6),V,NALPHA
      PI=3.141593
      DO 11 I=1,NALPHA
      CALL EULER(ALPHA(I),BETA(I),ANADA)
      VX(I)=-V*COS(ALPHA(I)*PI/180.)*COS(ANADA*PI/180.)
      VY(I)=V*COS(ALPHA(I)*PI/180.)*SIN(ANADA*PI/180.)
      VZ(I)=-V*SIN(ALPHA(I)*PI/180.)
   11 CONTINUE
      RETURN
      END
C
C     ******************************************************************
C
C
C
```

```
      SUBROUTINE EULER (A,B,ANADA)
C
C     ********************************************************************
C     ********************************************************************
C     TAKES INPUT ANGLE OF ATTACK ,A(DEGREES) AND SIDESLIP, B(DEGREES).
C     CALCULATES AND RETURNS EULER ANGLE, ANADA, IN DEGREES. ANADA IS
C     ROTATION ABOUT THE (ROTATED SYSTEM'S Z AXIS) IN RIGHT HAND SENSE.
C      THE ROTATION A ABOUT ORIGINAL Y AXIS THEN ROTATION ANADA ABOUT
C     THE MOVED Z AXIS CAUSES FINAL MOVED AXIS SYSTEM TO BE AT ANGLE OF
C     ATTACK A AND SIDESLIP B RELATIVE TO ORIGINAL AXIS SYSTEM.
C     ANADA IS USED IN TWO CASES
C     1- ROTATION OF BODY FIXED AXES RELATIVE TO INERTIALLY FIXED
C        WIND AXES ( TO GET FREESTREAM VELOCITY COMPONENTS FOR A BODY
C        AT ANGLE OF ATTACK AND SIDESLIP.).
C     2- ROTATION OF PROPELLER PLANE AXES RELATIVE TO BODY FIXED AXES
C        ( TO CALCULATE COORDINATES OF POINTS IN AN ANGLED AND
C        SIDESLIPPED PROPELLER PLANE IN TERMS OF THE BODY FIXED AXES.).
C
C     EXPRESSION USED TO GET ANADA IS:
C     ANADA=ARCTAN(TANB*COSA)
C     DUE TO LOGIC OF ARCTAN FUNCTION WHICH WORKS ONLY IN THE FIRST AND
C     FOURTH QUADRANTS, THE QUADRANT LOCATION OF ANGLE ANADA IS LOST.
C     THE TRUE VALUE OF ANADA IS THUS UNCERTAIN. SO THE PROPER
C     QUADRANT OF ANADA IS DERERMINED BELOW, BASED ON INPUT A AND B.
C     ********************************************************************
C     ********************************************************************
C
      PI=3.141593
      IF((COS(B*PI/180.).EQ.1.0).OR.( COS(B*PI/180.).EQ.0.0)) GO TO 1
      GO TO 2
    1 ANADA=B
      RETURN
    2 IF((COS(A*PI/180.).EQ.1.0).OR.(COS(A*PI/180.).EQ.0.0)) GO TO 3
      GO TO 4
    3 ANADA=B
      RETURN
```

```
C
    4 ANADA=(ATAN(COS(A*PI/180.)*TAN(B*PI/180.)))*180./PI
C
C     DETERMINE QUADRANT OF B. THEN TAKE CALCULATED ANADA VALUE OF
C     STATEMENT 4 AND CONVERT IT INTO ITS CORRESPONDING ANGLE IN THE
C     QUADRANT OF B.
C
C     CHECK IF B IS IN 1 ST QUADRANT.
      IF((COS(B*PI/180.).GT.0.0).AND.(SIN(B*PI/180.).GT.0.0)) GO TO 5
C
C     CHECK IF B IS IN 2 ND QUADRANT.
      IF((COS(B*PI/180.).LT.0.0).AND.(SIN(B*PI/180.).GT.0.0)) GO TO 6
C
C     CHECK IF B IS IN 3 RD QUADRANT.
      IF((COS(B*PI/180.).LT.0.0).AND (SIN(B*PI/180.).LT.0.0)) GO TO 7
C
C     THEN B IS IN 4 TH QUADRANT; PUT ANADA IN 4 TH QUADRANT.
      IF(ANADA.LT.0.0) RETURN
      ANADA=-ANADA
      RETURN
C
    5 CONTINUE
C
C     PUT ANADA IN 1 ST QUADRANT
      IF(ANADA GT.0.0) RETURN
      ANADA=-ANADA
      RETURN
C
    6 CONTINUE
C
C     PUT ANADA IN SECOND QUADRANT
      IF(ANADA.GT.0.0) GO TO 9
      ANADA=ANADA+180.0
      RETURN
    9 ANADA=180.0 - ANADA
      RETURN
```

```
C                                                                       EUL 3650
    7 CONTINUE                                                          EUL 3700
C                                                                       EUL 3750
C     PUT ANADA IN 3 RD QUADRANT                                        EUL 3800
      IF(ANADA.GT.0.0) GO TO 10                                         EUL 3850
      ANADA=180.0 - ANADA                                               EUL 3900
      RETURN                                                            EUL 3950
   10 ANADA=ANADA + 180.0                                               EUL 4000
      RETURN                                                            EUL 4050
      END                                                               EUL 4100
C                                                                       EUL 4150
C     ******************************************************************EUL 4200
C                                                                       EUL 4250
C                                                                       EUL 4300
C                                                                       EUL 4350
```

```
C     SUBROUTINE WGEOM                                                  WGE   50
C                                                                       WGE  100
C     ******************************************************************WGE  150
C     ******************************************************************WGE  200
C     ESTABLISHES GEOMETRY AND STRENGTH OF WING HORSESHOE VORTEX        WGE  250
C     MODEL IF WING INPUT WAS GIVEN. HORSESHOE VORTEX IS MODELLED ON THEWGE  300
C     ASSUMPTION THAT THE TRAILING VORTICES ROLL UP WITH A SPAN OF      WGE  350
C     SEPARATION EQUAL TO (PI/4) * PHYSICAL WING SPAN. TRAILING VORTEX  WGE  400
C     FILAMENTS ARE EXTENDED AFT BY 100 CHORD LENGTHS AND ASSUMED TO    WGE  450
C     LIE PARALLEL TO THE FREE STREAM VELOCITY DIRECTION REGARDLESS OF  WGE  500
C     BODY ANGLE OF ATTACK AND SIDESLIP.                                WGE  550
C     INPUT PARAMETERS:  NALPHA, V, ALPHA(I)'S, AND BETA(I)'S FROM      WGE  600
C                        COMMON BLOCK /INPUTS/.  AND CL(I)'S, CHORD,    WGE  650
C                        DIHED, SPAN, SWEEP, XQR, YQR, AND ZQR FROM     WGE  700
C                        COMMON BLOCK /WING1/.                          WGE  750
C     OUTPUT VALUES - STORED IN COMMON BLOCK /WING2/ :                  WGE  800
C                                       STRENGTH OF HORSESHOE VORTEX    WGE  850
C                                       ASSOSIATED WITH CL(I) AND       WGE  900
C                                       ALPHA(I).                       WGE  950
C      XTRALL(I), YTRALL(I), ZTRALL(I): X,Y,AND Z COORDIATAES OF AFT    WGE 1000
C                                       ''STARTING'' POINT OF LEFT SIDE WGE 1050
C                                       TRAILING VORTEX FILAMENT        WGE 1100
C                                       ASSOCIATED WITH BODY ORIENTATIONWGE 1150
C                                       ALPHA(I) AND BETA(I).           WGE 1200
C      XTRALR(I), YTRALR(I), ZTRALR(I): X,Y,AND Z COORDINATES OF AFT    WGE 1250
C                                       ''ENDING'' POINT OF RIGHT SIDE  WGE 1300
C                                       TRAILING VORTEX FILAMENT        WGE 1350
C                                       ASSOCIATED WITH BODY ORIENTATIONWGE 1400
C                                       ALPHA(I) AND BETA(I).           WGE 1450
C      XBTIPL, YBTIPL, ZBTIPL:          COORDINATES OF LEFT END OF      WGE 1500
C                                       BOUND VORTEX FILAMENT AND       WGE 1550
C                                       ''END'' POINT OF LEFT TRAILING  WGE 1600
C                                       VORTEX FILAMENT.                WGE 1650
C      XBTIPR, YBTIPR, ZBTIPR:          COORDINATES OF RIGHT END OF     WGE 1700
C                                       BOUND VORTEX FILAMENT AND       WGE 1750
C                                       ''STARTING'' POINT OF RIGHT SIDEWGE 1800
```

```
C                                          TRAILING VORTEX FILAMENT.       WGE 1850
C     SPAN:                                SPAN OR SPACING BETWEEN TRAILINGWGE 1900
C                                          VORTICES = (PI/4)*INPUT SPAN.   WGE 1950
C    NOTE:(THIS NEW SPAN IS SAVED IN COMMON /WING1/ OLD VALUE IS LOST.)WGE 2000
C                                                                      WGE 2050
C    PRINT OUT IS MADE OF INPUT QUANTITIES AND THE COMPUTED HORSESHOE  WGE 2100
C    VORTEX CHARACTERISTICS FOR EACH INPUT BODY ANGLE OF ATTACK ALPHA. WGE 2150
C    *****************************************************************WGE 2200
C    *****************************************************************WGE 2250
C                                                                      WGE 2300
     SUBROUTINE WGEOM                                                  WGE 2350
     COMMON /INPUTS/ ALPHA(6),BETA(6),VX(6),VY(6),VZ(6),V,NALPHA       WGE 2400
     COMMON /WING1/ CL(6),CHORD,DIHED,SPAN,SWEEP,XQR,YQR,ZQR,NWING     WGE 2450
     COMMON /WING2/ GAMMA(6),XTRALL(6),YTRALL(6),ZTRALL(6),XTRALR(6),  WGE 2500
    1YTRALR(6),ZTRALR(6),XBTIPL,YBTIPL,ZBTIPL,XBTIPR,YBTIPR,ZBTIPR     WGE 2550
     INTEGER STAR/'****'/,STARS(11),DASH/'----'/,DASHES(11)            WGE 2600
     PI=3.141593                                                       WGE 2650
     DO 1 K=1,11                                                       WGE 2700
     STARS(K)=STAR                                                     WGE 2750
     DASHES(K)=DASH                                                    WGE 2800
   1 CONTINUE                                                          WGE 2850
C                                                                      WGE 2900
C    PRINT TABLE HEADING AND ASSUMPTIONS:                              WGE 2950
C                                                                      WGE 3000
     WRITE(6,100)STAR,(STARS(K),K=1,11)                                WGE 3050
     WRITE(6,101)                                                      WGE 3100
     WRITE(6,102)                                                      WGE 3150
     WRITE(6,103)STAR,(STARS(K),K=1,11)                                WGE 3200
     WRITE(6,200)                                                      WGE 3250
     WRITE(6,201)                                                      WGE 3300
     WRITE(6,202)                                                      WGE 3350
     WRITE(6,203)                                                      WGE 3400
     WRITE(6,204)                                                      WGE 3450
C    ----------------------------------------------------------------- WGE 3500
C                                                                      WGE 3550
C    PRINT  INPUT PARAMETERS:                                          WGE 3600
```

```
C                                                                       WGE 3650
      WRITE(6,300)                                                      WGE 3700
      WRITE(6,301)                                                      WGE 3750
      WRITE(6,302)(DASHES(K),DASHES(K),DASHES(K),K=1,11)                WGE 3800
      WRITE(6,303)V,SPAN,CHORD,XQR,YQR,ZQR,DIHED,SWEEP                  WGE 3850
C     ------------------------------------------------------------------WGE 3900
C                                                                       WGE 3950
C     CALCULATE AND PRINT BOUND VORTEX GEOMETRY WHICH IS CONSTANT       WGE 4000
C     FOR ALL ORIENTATIONS (ALPHA - BETA):                              WGE 4050
C                                                                       WGE 4100
      WRITE(6,400)                                                      WGE 4150
      WRITE(6,401)                                                      WGE 4200
      WRITE(6,302)(DASHES(K),DASHES(K),DASHES(K),K=1,11)                WGE 4250
      SPAN=(PI/4.0)*SPAN                                                WGE 4300
      XBTIPL=XQR-((SPAN/2.0)*TAN((SWEEP*PI)/180.))                      WGE 4350
      XBTIPR=XBTIPL                                                     WGE 4400
      YBTIPL=YQR-(SPAN/2.0)                                             WGE 4450
      YBTIPR=YQR+(SPAN/2.0)                                             WGE 4500
      ZBTIPL=ZQR-((SPAN/2.0)*TAN((DIHED*PI)/180.))                      WGE 4550
      ZBTIPR=ZBTIPL                                                     WGE 4600
      WRITE(6,402)XBTIPL,YBTIPL,ZBTIPL,XBTIPR,YBTIPR,ZBTIPR,SPAN,DIHED  WGE 4650
     1SWEEP                                                             WGE 4700
C     ------------------------------------------------------------------WGE 4750
C                                                                       WGE 4800
C     LASTLY - CALCULATE AND PRINT HORSESHOE STRENGTH AND TRAILING      WGE 4850
C     VORTEX GEOMETRY WHICH VARIES WITH BODY ORIENTATION:               WGE 4900
C                                                                       WGE 4950
      WRITE(6,500)                                                      WGE 5000
      WRITE(6,501)                                                      WGE 5050
      WRITE(6,302)(DASHES(K),DASHES(K),DASHES(K),K=1,11)                WGE 5100
      DO 2 ICASE=1,NALPHA                                               WGE 5150
      GAMMA(ICASE)=(CHORD*CL(ICASE)*V)/2.0                              WGE 5200
      XTRALL(ICASE)=XBTIPL-(100.0*CHORD)                                WGE 5250
      XTRALR(ICASE)=XTRALL(ICASE)                                       WGE 5300
      YTRALL(ICASE)=YBTIPL+(100.0*CHORD*COS((ALPHA(ICASE)*PI)/180.)*    WGE 5350
     1TAN((BETA(ICASE)*PI)/180.))                                       WGE 5400
```

```
      YTRALR(ICASE)=YBTIPR+(100.0*CHORD*COS((ALPHA(ICASE)*PI)/180.)*
     1TAN((BETA(ICASE)*PI)/180.))
      ZTRALL(ICASE)=ZBTIPL-(100.0*CHORD*COS((ALPHA(ICASE)*PI)/180.)*
     1SIN((ALPHA(ICASE)*PI)/180.))
      ZTRALR(ICASE)=ZTRALL(ICASE)
      WRITE(6,502)ICASE,ALPHA(ICASE),CL(ICASE),GAMMA(ICASE),
     1XTRALL(ICASE),YTRALL(ICASE),ZTRALL(ICASE),XTRALR(ICASE),
     2YTRALR(ICASE),ZTRALR(ICASE)
    2 CONTINUE
      WRITE(6,600)(STARS(K),STARS(K),STARS(K),K=1,11)
      RETURN
C     ----------------------------------------------------------------
C
C     FORMATS:
C
  100 FORMAT('1',T43,12A4)
  101 FORMAT(' ',T43,'* WING HORSESHOE VORTEX STRENGTHS AND GEOMETRY *')
  102 FORMAT(' ',T43,'*  (COORDINATES RELATIVE TO BODY AXIS SYSTEM)  *')
  103 FORMAT(' ',T43,12A4/)
  200 FORMAT('0',T57,'*** ASSUMPTIONS ***'/)
  201 FORMAT(' ',T23,'- TRAILING VORTICES SEPARATED BY SPAN OF (PI/4) TI
     1MES THE PHYSICAL WING SPAN.')
  202 FORMAT(' ',T23,'- TRAILING VORTICES EXTEND AFT 100 CHORD LENGTHS A
     1ND TRAIL PARALLEL TO FREE STREAM VELICITY.')
  203 FORMAT(' ',T23,'- BOUND VORTEX FILAMENT COMPOSED OF TWO SEMISPANS
     1- EACH WITH DIHEDRAL AND SWEEP.')
  204 FORMAT(' ',T23,'- WING INPUT CL(I) ASSUMED EQUAL TO WING ROOT SECT
     1ION CL(I).')
  300 FORMAT(/,'0',T46,'*** INPUT GEOMETRY AND FLIGHT CONDITION ***'/)
  301 FORMAT(' ',T59,'WING ROOT QUARTER CHORD LOCATION',/' ',T24,'WING',
     1T40,'ROOT',T55,'(COORDINATES OF BOUND VORTEX MID POINT)',T105,
     2'DIHEDRAL',T123,'SWEEP',/' ',T7,'V',T24,'SPAN',T40,'CHORD',T58,
     3'XQR',T74,'YQR',T90,'ZQR',T104,'(DEGREES)' T121,'(DEGREES)')
  302 FORMAT(' ',33A4)
  303 FORMAT(' ',T2,F13.6,T19,F13.6,T36,F13.6,T53,F13.6,T69,F13.6,T85,
     1F13.6,T102,F13.6,T119,F13.6)
```

```
  400 FORMAT(/,'0',T31,'*** BOUND VORTEX GEOMETRY - CONSTANT FOR ALL BODWGE 7250
     1Y ALPHA ORIENTATIONS ***'/)                                       WGE 7300
  401 FORMAT(' ',T6,'LEFT BOUND END POINT COORDINATES',T48,'RIGHT BOUND WGE 7350
     1END POINT COORDINATES',/' ',T6,'(LEFT TRAILING VORTEX END POINT)',WGE 7400
     2T46,'(RIGHT TRAILING VORTEX STARTING POINT)',T107,                WGE 7450
     3'DIHEDRAL',T124,'SWEEP',/' ',T5,'XBTIPL',T19,'YBTIPL',T33,        WGE 7500
     4'ZBTIPL',T48,'XBTIPR',T62,'YBTIPR',T76,'ZBTIPR',T93,'SPAN',T106,  WGE 7550
     5'(DEGREES)' T122,'(DEGREES)')                                     WGE 7600
  402 FORMAT(' ',T1,F13.6,T15,F13.6,T29,F13.6,T44,F13.6,T58,F13.6,T72,  WGE 7650
     1F13.6,T88,F13.6,T104,F13.6,T120,F13.6)                            WGE 7700
  500 FORMAT(/,'0',T33,'*** HORSESHOE VORTEX STRENGTH AND GEOMETRY - DEPWGE 7750
     1ENDENT ON BODY ORIENTATION ***'/)                                 WGE 7800
  501 FORMAT(' ',T9,'INPUT',T23,'INPUT',T37,'STRENGTH',T51,'LEFT TRAILINWGE 7850
     1G VORTEX AFT END LOCATION',T94,'RIGHT TRAILING VORTEX AFT END LOCAWGE 7900
     2TION',/' ',T2,'I',T8,'ALPHA(I)',T23,'CL(I)',T37,'GAMMA(I)',T51,   WGE 7950
     3'XTRALL(I)',T65,'YTRALL(I)',T79,'ZTRALL(I)',T94,'XTRALR(I)',T108, WGE 8000
     4'YTRALR(I)',T122,'ZTRALR(I)')                                     WGE 8050
  502 FORMAT(' ',T1,I2,T5,F13.6,T19,F13.6,T34,F13.6,T49,E13.6,T63,F13.6,WGE 8100
     1T77,F13.6,T92,E13.6,T106,F13.6,T120,F13.6)                        WGE 8150
  600 FORMAT('0',33A4)                                                  WGE 8200
C                                                                       WGE 8250
      END                                                               WGE 8300
C                                                                       WGE 8350
C                                                                       WGE 8400
C     *****************************************************************WGE 8450
C                                                                       WGE 8500
C                                                                       WGE 8550
C                                                                       WGE 8600
```

```
      SUBROUTINE PANEL(NIPMAX,NSECMA,NSECTO,NFLAG,NP,NCOUNT,LLL,PX,PY,
     1PZ,STOX1,STOY1,STOZ1,STOX2,STOY2,STOZ2,S,ANVX,ANVY,ANVZ,XC,YC,ZC)
C
C     ******************************************************************
C     ******************************************************************
C     SUBROUTINE TAKES INPUT BODY SURFACE POINTS AND GENERATES SURFACE
C     PANELS FROM IT.  FOR SYMMETRIC BODY INPUT OPTION, THE LEFT HALF
C     OF BODY IS INPUT. SUBROUTINE GENERATES LEFT SIDE PANELS AND
C     AUTOMATICALLY GENERATES THE CORRESPONDING MIRROR IMAGE RIGHT SIDE
C     PANELS.  FOR NONSYMMETRIC INPUT OPTION, ENTIRE SURFACE POINTS ARE
C     INPUT. EACH PANEL MUST BE GENERATED ONE BY ONE.
C     STOPER-- IS ERROR SIGNAL VARIABLE INITIALLY = 0.0. IT STAYS ZERO
C     IF NO FATAL PANELING ERRORS ARE DETECTED IN PANELING THE INPUT.
C     STOPER BECOMES = 1.0 WHEN THE FIRST AND ANY SUBSEQUENT FATAL
C     PANELING ERRORS ARE DETECTED. SUCH ERRORS ARE PRINTED AS FOUND,
C     ROUTINE CONTINUES TO GENERATE PANELS UNTIL INPUT IS FINISHED.
C     THEN IF STOPER = 1.0 ENTIRE PROGRAM EXECUTION IS STOPPED.
C     IF STOPER IS 0.0 AT END OF SUBROUTINE, PROGRAM CONTINUES.
C     ******************************************************************
C     ******************************************************************
C
      DIMENSION NCOUNT(NSECMA),NFLAG(NSECMA),PX(NSECMA,NIPMAX),
     1PY(NSECMA,NIPMAX),PZ(NSECMA,NIPMAX),STOX1(NIPMAX),STOY1(NIPMAX),
     2STOZ1(NIPMAX),STOX2(NIPMAX),STOY2(NIPMAX),STOZ2(NIPMAX),
     3ANVX(LLL),ANVY(LLL),ANVZ(LLL),S(LLL),XC(LLL),YC(LLL),ZC(LLL)
      COMMON /SYMTRE/ NPNSYM(2596),INSOLV(1298),NSYMET
      COMMON /OPTION/ NCALC,NLIST,NPUNCH
      INTEGER STARS(11),DASHES(11),STAR/'****'/,DASH/'----'/
C
      DO 1 KK=1,11
      STARS(KK)=STAR
      DASHES(KK)=DASH
    1 CONTINUE
C
      STOPER=0.00
C
```

```
C
C     PRINT HEADINGS FOR BODY PANEL GEOMETRY TABLE IF INPUT REQUESTS IT.
      IF((NCALC.EQ.0).AND.(NLIST.EQ.1)) GO TO 2
      WRITE(6,900)(STARS(KJL),KJL=1,11)
      WRITE(6,901)
      WRITE(6,902)(STARS(KJL),KJL=1,11)
      IF(NSYMET.EQ.0) WRITE(6,903)
      WRITE(6,904)
      WRITE(6,927)(DASHES(KJL),DASHES(KJL),DASHES(KJL),KJL=1,11)
      GO TO 3
    2 WRITE(6,905)
    3 CONTINUE
C
C     STORAGE SPACES STOX1(K),STOY1(K),STOZ1(K) CONTAIN THE COORDINATES
C     OF THE ORIGINAL K TH INPUT POINT ON THE J TH CROSS SECTION UNDER
C     CONSIDERATION DURING SUBROUTINE EXECUTION. STORAGE SPACES
C     STOX2(K),STOY2(K),STOZ2(K) CONTAIN THE COORDINATES OF THE ORIGINAL
C     K TH INPUT POINT OF THE J+1 TH CROSS SECTION UNDER CONSIDERATION.
C     AT ANY INSTANT IN THE ROUTINE, PANELS ARE BEING DEFINED BETWEEN
C     THE CROSS SECTIONS J AND J+1. AND THE INPUT FOUR CORNER POINTS
C     OF ONE PANEL ARE THE K TH AND K+1 TH POINTS OF THE CROSS SECTIONS.
C
      N=NSECTO-1
      I=0
      M=0
      MTOT=0
      J=0
    4 J=J+1
C
C     IF NFLAG(J) IS 1, THIS SECTION IS THE LAST SECTION OF A DISCRETE
C     BODY SEGMENT OR PANEL NETWORK. OR SECTION J+1 IS A REPEAT
C     DESCRIPTION OF SECTION J. SO PANELS ARE NOT TO BE GENERATED
C     BETWEEN THESE CURRENT J TH AND J+1 SECTIONS. THUS SKIP TO NEXT
C     SECTION PAIR BY INCREMENTING THE J+1 TH SECTION TO THE J TH ONE.
C     -SYMMETRIC INPUT ONLY- NCOUNT(J) IS NUMBER OF POINTS ON LEFT HALF
C     SIDE OF BODY INCLUDING TOP AND BOTTOM CENTERLINE POINTS.
```

```
C
      IF(NFLAG(J).EQ.1) GO TO 4
      LNIP=NCOUNT(J)-1
      NCO=NCOUNT(J)
      DO 5 K=1,NCO
      STOX1(K)=PX(J,K)
      STOY1(K)=PY(J,K)
      STOZ1(K)=PZ(J,K)
      STOX2(K)=PX(J+1,K)
      STOY2(K)=PY(J+1,K)
      STOZ2(K)=PZ(J+1,K)
    5 CONTINUE
    6 I=M
      DO 700 K=1,LNIP
      I=I+1
C
C     FOR SYMMETRIC BODY INPUT OPTION ONLY :
C     SET UP BOOK KEEPING ARRAY NPNSYM. IF NPNSYM(I) IS ZERO THEN THAT
C     PANEL WHOSE INDEX IS I IS ONE ON THE UN-INPUT RIGHT IMAGE SIDE OF
C     THE SYMMETRIC CONFIGURATION. IF NPNSYM(I) = NUM, NOT ZERO, THEN
C     PANEL I WAS ONE INPUT ON THE LEFT SIDE OF CONFIGURATION AND ITS
C     MIRROR IMAGE PANEL ON THE RIGHT SIDE IS PANEL NUMBER NUM.
C
      IF(NSYMET.NE.0) GO TO 7
      IOPOSE=MTOT+(2*LNIP)-(K-1)
      NPNSYM(I)=IOPOSE
      NPNSYM(IOPOSE)=0
    7 CONTINUE
C
C     COMPUTE COMPONENTS OF THE OUTWARD UNIT VECTOR TO PANEL I USING
C     THE ORIGINAL INPUT POINTS AS STORED IN STOX1,Y1,Z1 AND STOX2,Y2,
C     Z2. COMPONENTS ARE ANVX, ANVY,AND ANVZ.
      BX1=STOX2(K+1)-STOX1(K)
      BY1=STOY2(K+1)-STOY1(K)
      BZ1=STOZ2(K+1)-STOZ1(K)
      BX2=STOX2(K)-STOX1(K+1)
```

```
      BY2=STOY2(K)-STOY1(K+1)
      BZ2=STOZ2(K)-STOZ1(K+1)
      Q=SQRT(((BY1*BZ2)-(BY2*BZ1))**2+((BX2*BZ1)-(BX1*BZ2))**2+((BX1*BY2
     1)-(BX2*BY1))**2)
      ANVX(I)=((BY1*BZ2)-(BY2*BZ1))/Q
      ANVY(I)=((BX2*BZ1)-(BX1*BZ2))/Q
      ANVZ(I)=((BX1*BY2)-(BX2*BY1))/Q
C
C     FOR SYMMETRIC INPUT OPTION - ASSIGN UNIT VECTOR COMPONENTS TO
C     THE CORRESPONDING RIGHT SIDE MIRROR IMAGE PANELS.
      IF(NSYMET.NE.0) GO TO 8
      ANVX(NPNSYM(I))=ANVX(I)
      ANVY(NPNSYM(I))=-ANVY(I)
      ANVZ(NPNSYM(I))=ANVZ(I)
    8 CONTINUE
C
C
C
C     FIVE POSSIBILITIES IN THE PANELING EXIST:
C     CASE A   FOUR INPUT POINTS GENERATE A TRIANGULAR PANEL BETWEEN
C              SECTIONS J AND J+1 WITH TWO POINTS ON SECTION J
C              COINCIDING AND THE TWO POINTS ON SECTION J+1 NOT THE SAME
C     CASE B:  FOUR INPUT POINTS GENERATE A TRIANGULAR PANEL BETWEEN
C              SECTIONS J AND J+1 WITH THE TWO POINTS OF SECTION J+1
C              COINCIDING AND THE TWO POINTS ON SECTION J NOT THE SAME.
C     CASE C:  GENERAL CASE IN WHICH FOUR INPUT POINTS GENERATE A FOUR
C              SIDED PANEL BETWEEN SECTIONS J AND J+1. SINCE THIS PANEL
C              DOES NOT NECESSARILY LIE IN A PLANE, A PROJECTION OF THE
C              CORNER POINTS ONTO A PLANE IS NECESSARY IN ORDER TO
C              CONVERT THIS INPUT PANEL INTO A PLANE QUADRILATERAL.
C     CASE D:  UNLIKELY THOUGH POSSIBLE CASE IN WHICH THE FOUR SIDED
C              PANEL OF CASE C IS CONVERTED INTO THE PLANE QUADRILATERAL
C              BUT THE QUADRILATERAL IS TRIANGULAR WITH ONE CORNER ON
C              SECTION J (IE. PT(J,K)=PT(J,K+1)) AND THE OTHER TWO
C              CORNERS ARE ON SECTION J+1.
C     CASE E:  SIMILAR TO CASE D BUT WITH TWO CORNERS OF THE TRIANGLE ON
```

```
C             SECTION J AND THE THIRD CORNER ON SECTION J+1 :
C             (IE. PT(J+1,K) = PT(J+1,K+1) ).
C
C     NOW CHECK TO SEE IF TWO ADJACENT POINTS ON EITHER SECTION J OR
C     J+1 ARE THE SAME. IF THE SAME, THIS PANEL WAS INPUT AS A TRIANGLE
C     AND THUS LIES IN A PLANE AS INPUT. THE AREA S(I) AND CONTROL
C     POINTS ARE FOUND FROM THE INPUT GEOMETRY.
C
C
      IF(STOX1(K).NE.STOX1(K+1)) GO TO 9
      IF(STOY1(K) NE.STOY1(K+1)) GO TO 9
      IF(STOZ1(K).NE.STOZ1(K+1)) GO TO 9
C     THEN TWO POINTS ON SECTION J COINCIDE. PANEL I WAS INPUT AS A
C     TRIANGLE. - CASE A - FIND AREA OF IT :
      GO TO 100
    9 IF(STOX2(K).NE.STOX2(K+1)) GO TO 300
      IF(STOY2(K).NE.STOY2(K+1)) GO TO 300
      IF(STOZ2(K).NE.STOZ2(K+1)) GO TO 300
C     THEN TWO POINTS ON SECTION J+1 COINCIDE. PANEL I WAS INPUT AS A
C     TRIANGLE - CASE B-  FIND AREA OF IT :
      GO TO 200
C
C-----------------------------------------------------------------------
C
C
  100 CONTINUE
C
C     THIS IS CASE A.  TRIANGULAR PANEL.  COMPUTE AREA, S(I), AND
C     CONTROL POINT COORDINATES FOR IT. THEN GO TO STATEMENT 600 AND
C     INCREMENT TO CONSIDER THE NEXT PANEL.
C
C     GET LENGTHS OF THE THREE SIDES OF TRIANGLE, RS,SU, AND UR:
      RS=SQRT(((STOX2(K)-STOX1(K))**2)+((STOY2(K)-STOY1(K))**2)+
     1((STOZ2(K)-STOZ1(K))**2))
      SU=SQRT(((STOX2(K+1)-STOX1(K))**2)+((STOY2(K+1)-STOY1(K))**2)+
     1((STOZ2(K+1)-STOZ1(K))**2))
```

```
      UR=SQRT(((STOX2(K+1)-STOX2(K))**2)+((STOY2(K+1)-STOY2(K))**2)+
     1((STOZ2(K+1)-STOZ2(K))**2))
      STANG=((RS**2)+(SU**2)-(UR**2))/(2.*RS*SU)
C
C     CHECK IF ABS(STANG) LESS THAN OR EQUAL TO 1. IF NOT, NUMERICAL
C     ROUNDOFF ERROR HAS OCCURED POSSIBLY DUE TO TWO PANEL CORNER POINTS
C     BEING INPUT TOO CLOSE TOGETHER BUT NOT ACTUALLY COINCIDING.  WILL
C     MAKE STOPER =1.0. IF ERROR OCCURED.
      IF(ABS(STANG).GT.1.0) GO TO 101
      ANGST=ARCOS(STANG)
      GO TO 102
  101 JJJJ=J+1
      KKKK=K+1
      WRITE(6,916)K,J.K,KKKK,JJJJ
      STOPER=1.0
      GO TO 600
  102 CONTINUE
C
C     COMPUTE SURFACE AREA, S(I), OF TRIANGULAR PANEL.
      SINE=SQRT(1. - ((COS(ANGST))**2))
      S(I)=(RS * SU * SINE)/2.
C     MIRROR IMAGE PANEL SAME AREA ASSIGNED. SYMMETRIC INPUT ONLY :
      IF(NSYMET.EQ.0) S(NPNSYM(I))=S(I)
C
C     COMPUTE CONTROL POINT COORDINATES XC, YC, AND ZC, WHICH ARE
C     LOCATED AT THE CENTROID OF AREA OF THE TRIANGULAR PANEL I --CASE A
      DIFFX=ABS((STOX1(K)-STOX2(K))/3.)
      DIFFY=ABS((STOY1(K)-STOY2(K))/3.)
      DIFFZ=ABS((STOZ1(K)-STOZ2(K))/3.)
      IF(STOX2(K).LT.STOX1(K)) GO TO 103
      DX=STOX2(K)-DIFFX
      GO TO 104
  103 DX=STOX2(K)+DIFFX
  104 IF(STOY2(K).LT.STOY1(K)) GO TO 105
      DY=STOY2(K)-DIFFY
      GO TO 106
```

```
  105 DY=STOY2(K)+DIFFY                                                PAN10850
  106 IF(STOZ2(K).LT.STOZ1(K)) GO TO 107                               PAN10900
      DZ=STOZ2(K)-DIFFZ                                                PAN10950
      GO TO 108                                                        PAN11000
  107 DZ=STOZ2(K)+DIFFZ                                                PAN11050
  108 DIFFX=ABS((STOX1(K)-STOX2(K+1))/3.)                              PAN11100
      DIFFY=ABS((STOY1(K)-STOY2(K+1))/3.)                              PAN11150
      DIFFZ=ABS((STOZ1(K)-STOZ2(K+1))/3.)                              PAN11200
      IF(STOX2(K+1).LT.STOX1(K)) GO TO 109                             PAN11250
      EX=STOX2(K+1)-DIFFX                                              PAN11300
      GO TO 110                                                        PAN11350
  109 EX=STOX2(K+1)+DIFFX                                              PAN11400
  110 IF(STOY2(K+1).LT.STOY1(K)) GO TO 111                             PAN11450
      EY=STOY2(K+1)-DIFFY                                              PAN11500
      GO TO 112                                                        PAN11550
  111 EY=STOY2(K+1)+DIFFY                                              PAN11600
  112 IF(STOZ2(K+1).LT.STOZ1(K)) GO TO 113                             PAN11650
      EZ=STOZ2(K+1)-DIFFZ                                              PAN11700
      GO TO 114                                                        PAN11750
  113 EZ=STOZ2(K+1)+DIFFZ                                              PAN11800
  114 XC(I)=(DX+EX)/2.                                                 PAN11850
      YC(I)=(DY+EY)/2.                                                 PAN11900
      ZC(I)=(DZ+EZ)/2.                                                 PAN11950
C                                                                      PAN12000
C     FOR SYMMETRIC INPUT ONLY- ASSIGN CONTROL POINT COORDINATES TO    PAN12050
C     MIRROR IMAGE PANELS :                                            PAN12100
      IF(NSYMET.NE.0) GO TO 115                                        PAN12150
      XC(NPNSYM(I))=XC(I)                                              PAN12200
      YC(NPNSYM(I))=-YC(I)                                             PAN12250
      ZC(NPNSYM(I))=ZC(I)                                              PAN12300
  115 CONTINUE                                                         PAN12350
C                                                                      PAN12400
C     CALCULATIONS COMPLETE ON I TH PANEL ( AND ITS IMAGE PANEL IF     PAN12450
C     SYMMETRIC INPUT USED). PRINT TABLE ENTRY IF REQUIRED AND GO TO   PAN12500
C     I+1 TH PANEL.                                                    PAN12550
C                                                                      PAN12600
```

```
      IF((NCALC.EQ.0).AND.(NLIST.EQ.1)) GO TO 600                       PAN12650
      WRITE(6,906) STOX1(K),STOY1(K),STOZ1(K)                           PAN12700
      WRITE(6,906) STOX1(K+1),STOY1(K+1),STOZ1(K+1)                     PAN12750
      WRITE(6,907) I,STOX2(K),STOY2(K),STOZ2(K),XC(I),YC(I),ZC(I),S(I)  PAN12800
      WRITE(6,906) STOX2(K+1),STOY2(K+1),STOZ2(K+1)                     PAN12850
      WRITE(6,908)(DASHES(KJL),DASHES(KJL),DASHES(KJL),KJL=1,11)        PAN12900
C                                                                       PAN12950
C     NEXT IF SYMMETRIC OPTION USED, PRINT THE IMAGE PANEL (OPPOSITE)   PAN13000
C     TO PANEL I:                                                       PAN13050
C                                                                       PAN13100
      IF(NSYMET.EQ.1) GO TO 600                                         PAN13150
      PRX1=STOX1(K+1)                                                   PAN13200
      PRY1=-STOY1(K+1)                                                  PAN13250
      PRZ1=STOZ1(K+1)                                                   PAN13300
      PRX2=STOX1(K)                                                     PAN13350
      PRY2=-STOY1(K)                                                    PAN13400
      PRZ2=STOZ1(K)                                                     PAN13450
      PRX3=STOX2(K+1)                                                   PAN13500
      PRY3=-STOY2(K+1)                                                  PAN13550
      PRZ3=STOZ2(K+1)                                                   PAN13600
      PRX4=STOX2(K)                                                     PAN13650
      PRY4=-STOY2(K)                                                    PAN13700
      PRZ4=STOZ2(K)                                                     PAN13750
      WRITE(6,906) PRX1,PRY1,PRZ1                                       PAN13800
      WRITE(6,909) PRX2,PRY2,PRZ2                                       PAN13850
      WRITE(6,910) NPNSYM(I),PRX3,PRY3,PRZ3,XC(NPNSYM(I)),YC(NPNSYM(I)),PAN13900
     +ZC(NPNSYM(I)),S(NPNSYM(I))                                        PAN13950
      WRITE(6,911) I,PRX4,PRY4,PRZ4                                     PAN14000
      WRITE(6,908)(DASHES(KJL),DASHES(KJL),DASHES(KJL),KJL=1,11)        PAN14050
C                                                                       PAN14100
      GO TO 600                                                         PAN14150
C                                                                       PAN14200
C-----------------------------------------------------------------------PAN14250
C                                                                       PAN14300
C                                                                       PAN14350
  200 CONTINUE                                                          PAN14400
```

```
C                                                                       PAN14450
C     THIS IS CASE B.  TRIANGULAR PANEL.  COMPUTE AREA, S(I), AND       PAN14500
C     CONTROL POINT COORDINATES. THEN GO TO STATEMENT 600 AND           PAN14550
C     INCREMENT TO CONSIDER NEXT PANEL.                                 PAN14600
C                                                                       PAN14650
C     GET LENGTHS OF THE THREE SIDES OF TRIANGLE, RS, ST, AND TR :      PAN14700
      RS=SQRT(((STOX2(K)-STOX1(K))**2)+((STOY2(K)-STOY1(K))**2)+        PAN14750
     1((STOZ2(K)-STOZ1(K))**2))                                         PAN14800
      ST=SQRT(((STOX1(K+1)-STOX1(K))**2)+((STOY1(K+1)-STOY1(K))**2)+    PAN14850
     1((STOZ1(K+1)-STOZ1(K))**2))                                       PAN14900
      TR=SQRT(((STOX2(K+1)-STOX1(K+1))**2)+((STOY2(K+1)-STOY1(K+1))**2)+PAN14950
     1((STOZ2(K+1)-STOZ1(K+1))**2))                                     PAN15000
      URANG=((RS**2)+(TR**2)-(ST**2))/(2.*RS*TR)                        PAN15050
C     CHECK IF ABS(URANG) LESS THAN OR EQUAL TO 1. IF NOT, NUMERICAL    PAN15100
C     ROUNDOFF ERROR HAS OCCURED POSSIBLY DUE TO TWO PANEL CORNER POINTSPAN15150
C     BEING INPUT TOO CLOSE TOGETHER BUT NOT ACTUALLY COINCIDING.  WILL PAN15200
C     MAKE STOPER =1.0. IF ERROR OCCURED.                               PAN15250
      IF(ABS(URANG).GT.1.0) GO TO 201                                   PAN15300
      ANGUR=ARCOS(URANG)                                                PAN15350
      GO TO 202                                                         PAN15400
  201 JJJJ=J+1                                                          PAN15450
      KKKK=K+1                                                          PAN15500
      WRITE(6,917)K,KKKK,J,K,JJJJ                                       PAN15550
      STOPER=1.0                                                        PAN15600
      GO TO 600                                                         PAN15650
  202 CONTINUE                                                          PAN15700
C                                                                       PAN15750
C     COMPUTE SURFACE AREA, S(I), OF THE TRIANGULAR PANEL:              PAN15800
      SINE=SQRT(1.-((COS(ANGUR))**2))                                   PAN15850
      S(I)=(RS * TR * SINE)/2.                                          PAN15900
C     MIRROR IMAGE PANEL SAME AREA ASSIGNED. SYMMETRIC INPUT ONLY :     PAN15950
      IF(NSYMET.EQ.0) S(NPNSYM(I))=S(I)                                 PAN16000
C                                                                       PAN16050
C     COMPUTE CONTROL POINT COORDINATES XC, YC, AND ZC, WHICH ARE       PAN16100
C     LOCATED AT THE CENTROID OF AREA OF THE TRIANGULAR PANEL I --CASE BPAN16150
      DIFFX=ABS((STOX2(K)-STOX1(K))/3.)                                 PAN16200
```

```
      DIFFY=ABS((STOY2(K)-STOY1(K))/3.)                                 PAN16250
      DIFFZ=ABS((STOZ2(K)-STOZ1(K))/3.)                                 PAN16300
      IF(STOX1(K).LT.STOX2(K)) GO TO 203                                PAN16350
      DX=STOX1(K)-DIFFX                                                 PAN16400
      GO TO 204                                                         PAN16450
  203 DX=STOX1(K)+DIFFX                                                 PAN16500
  204 IF(STOY1(K).LT.STOY2(K)) GO TO 205                                PAN16550
      DY=STOY1(K)-DIFFY                                                 PAN16600
      GO TO 206                                                         PAN16650
  205 DY=STOY1(K)+DIFFY                                                 PAN16700
  206 IF(STOZ1(K).LT.STOZ2(K)) GO TO 207                                PAN16750
      DZ=STOZ1(K)-DIFFZ                                                 PAN16800
      GO TO 208                                                         PAN16850
  207 DZ=STOZ1(K)+DIFFZ                                                 PAN16900
  208 DIFFX=ABS((STOX2(K)-STOX1(K+1))/3.)                               PAN16950
      DIFFY=ABS((STOY2(K)-STOY1(K+1))/3.)                               PAN17000
      DIFFZ=ABS((STOZ2(K)-STOZ1(K+1))/3.)                               PAN17050
      IF(STOX1(K+1).LT.STOX2(K)) GO TO 209                              PAN17100
      EX=STOX1(K+1)-DIFFX                                               PAN17150
      GO TO 210                                                         PAN17200
  209 EX=STOX1(K+1)+DIFFX                                               PAN17250
  210 IF(STOY1(K+1).LT.STOY2(K)) GO TO 211                              PAN17300
      EY=STOY1(K+1)-DIFFY                                               PAN17350
      GO TO 212                                                         PAN17400
  211 EY=STOY1(K+1)+DIFFY                                               PAN17450
  212 IF(STOZ1(K+1).LT.STOZ2(K)) GO TO 213                              PAN17500
      EZ=STOZ1(K+1)-DIFFZ                                               PAN17550
      GO TO 214                                                         PAN17600
  213 EZ=STOZ1(K+1)+DIFFZ                                               PAN17650
  214 XC(I)=(DX+EX)/2.                                                  PAN17700
      YC(I)=(DY+EY)/2.                                                  PAN17750
      ZC(I)=(DZ+EZ)/2.                                                  PAN17800
C                                                                       PAN17850
C     FOR SYMMETRIC INPUT ONLY- ASSIGN CONTROL POINT COORDINATES TO     PAN17900
C     MIRROR IMAGE PANELS :                                             PAN17950
      IF(NSYMET.NE.0) GO TO 215                                         PAN18000
```

```
      XC(NPNSYM(I))=XC(I)
      YC(NPNSYM(I))=-YC(I)
      ZC(NPNSYM(I))=ZC(I)
  215 CONTINUE
C
C     CALCULATIONS COMPLETE ON I TH PANEL ( AND ITS IMAGE PANEL IF
C     SYMMETRIC INPUT USED). PRINT TABLE ENTRY IF REQUIRED AND GO TO
C     I+1 TH PANEL.
C
      IF((NCALC.EQ.0).AND.(NLIST.EQ.1)) GO TO 600
      WRITE(6,906) STOX1(K),STOY1(K),STOZ1(K)
      WRITE(6,906) STOX1(K+1),STOY1(K+1),STOZ1(K+1)
      WRITE(6,907) I,STOX2(K),STOY2(K),STOZ2(K),XC(I),YC(I),ZC(I),S(I)
      WRITE(6,906) STOX2(K+1),STOY2(K+1) STOZ2(K+1)
      WRITE(6,908)(DASHES(KJL),DASHES(KJL),DASHES(KJL),KJL=1,11)
C
C     NEXT IF SYMMETRIC OPTION USED, PRINT THE IMAGE PANEL (OPPOSITE)
C     TO PANEL I:
C
      IF(NSYMET.EQ.1) GO TO 600
      PRX1=STOX1(K+1)
      PRY1=-STOY1(K+1)
      PRZ1=STOZ1(K+1)
      PRX2=STOX1(K)
      PRY2=-STOY1(K)
      PRZ2=STOZ1(K)
      PRX3=STOX2(K+1)
      PRY3=-STOY2(K+1)
      PRZ3=STOZ2(K+1)
      PRX4=STOX2(K)
      PRY4=-STOY2(K)
      PRZ4=STOZ2(K)
      WRITE(6,906) PRX1,PRY1,PRZ1
      WRITE(6,909) PRX2,PRY2,PRZ2
      WRITE(6,910) NPNSYM(I),PRX3,PRY3,PRZ3,XC(NPNSYM(I)),YC(NPNSYM(I)),
     +ZC(NPNSYM(I)),S(NPNSYM(I))
```

```
      WRITE(6,911) I PRX4,PRY4,PR24                                     PAN19850
      WRITE(6,908)(DASHES(KJL),DASHES(      DASHES(KJL),KJL=1,11)       PAN19900
C                                                                       PAN19950
      GO TO 600                                                         PAN20000
C                                                                       PAN20050
C-----------------------------------------------------------------------PAN20100
C                                                                       PAN20150
C                                                                       PAN20200
  300 CONTINUE                                                          PAN20250
C                                                                       PAN20300
C                                                                       PAN20350
C                                                                       PAN20400
C                                                                       PAN20450
C     THIS IS EITHER CASE C , D, OR E. PROBABLY CASE C APPLIES.         PAN20500
C     THIS PANEL HAS BEEN INPUT WITH FOUR SIDES. FIRST, PROJECT THIS    PAN20550
C     NON PLANAR PANEL ONTO A PLANE. THIS GIVES RISE TO A PLANE         PAN20600
C     QUADRILATERAL HAVING CORNER POINTS NOW DIFFERENT FROM THE ORIGINALPAN20650
C     INPUT POINTS.  STORE THESE NEW CORNER POINTS IN PX, PY, AND PZ,   PAN20700
C     THUS REPLACING ORIGINAL INPUT POINT VALUES WHICH REMAIN STORED IN PAN20750
C     STOX1,Y1,AND Z1.  XBAR, YBAR, AND ZBAR AND DIS ARE FUNCTIONS OF   PAN20800
C     THE ORIGINAL INPUT VALUES AND ARE INVOLVED IN PANEL PROJECTION.   PAN20850
      XBAR=(STOX1(K+1)+STOX1(K)+STOX2(K)+STOX2(K+1))/4.                 PAN20900
      YBAR=(STOY1(K+1)+STOY1(K)+STOY2(K)+STOY2(K+1))/4.                 PAN20950
      ZBAR=(STOZ1(K+1)+STOZ1(K)+STOZ2(K)+STOZ2(K+1))/4.                 PAN21000
      DIS=ABS(ANVX(I)*(XBAR-STOX1(K+1))+ANVY(I)*(YBAR-STOY1(K+1))+ANVZ(IPAN21050
     1)*(ZBAR-STOZ1(K+1)))                                              PAN21100
      PX(J,K)=(STOX1(K)-(ANVX(I)*DIS))                                  PAN21150
      PY(J,K)=(STOY1(K)-(ANVY(I)*DIS))                                  PAN21200
      PZ(J,K)=(STOZ1(K)-(ANVZ(I)*DIS))                                  PAN21250
      PX(J,K+1)=(STOX1(K+1)+(ANVX(I)*DIS))                              PAN21300
      PY(J,K+1)=(STOY1(K+1)+(ANVY(I)*DIS))                              PAN21350
      PZ(J,K+1)=(STOZ1(K+1)+(ANVZ(I)*DIS))                              PAN21400
      PX(J+1,K)=(STOX2(K)+(ANVX(I)*DIS))                                PAN21450
      PY(J+1,K)=(STOY2(K)+(ANVY(I)*DIS))                                PAN21500
      PZ(J+1,K)=(STOZ2(K)+(ANVZ(I)*DIS))                                PAN21550
      PX(J+1,K+1)=(STOX2(K+1)-(ANVX(I)*DIS))                            PAN21600
```

```
      PY(J+1,K+1)=(STOY2(K+1)-(ANVY(I)*DIS))                            PAN21650
      PZ(J+1,K+1)=(STOZ2(K+1)-(ANVZ(I)*DIS))                            PAN21700
C                                                                       PAN21750
C     COMPUTE LENGTHS OF THE SIDES OF THE QUADRILATERAL RS, ST, TU, UR, PAN21800
C     AND THE DIAGONALS RT, AND SU. NOTE THE QUADRILATERAL MAY BE       PAN21850
C     A TRIANGLE OF CASE D OR CASE E TYPE . THIS IS CHECKED LATER.      PAN21900
      RS=SQRT((PX(J,K)-PX(J+1,K))**2+(PY(J,K)-PY(J+1,K))**2+            PAN21950
     1(PZ(J,K)-PZ(J+1,K))**2)                                           PAN22000
      ST=SQRT((PX(J,K+1)-PX(J,K))**2+(PY(J,K+1)-PY(J,K))**2+            PAN22050
     1(PZ(J,K+1)-PZ(J,K))**2)                                           PAN22100
      TU=SQRT((PX(J+1,K+1)-PX(J,K+1))**2+(PY(J+1,K+1)-PY(J,K+1))**2+    PAN22150
     1(PZ(J+1,K+1)-PZ(J,K+1))**2)                                       PAN22200
      UR=SQRT((PX(J+1,K)-PX(J+1,K+1))**2+(PY(J+1,K)-PY(J+1,K+1))**2+    PAN22250
     1(PZ(J+1,K)-PZ(J+1,K+1))**2)                                       PAN22300
      RT=SQRT((PX(J,K+1)-PX(J+1,K))**2+(PY(J,K+1)-PY(J+1,K))**2+        PAN22350
     1(PZ(J,K+1)-PZ(J+1,K))**2)                                         PAN22400
      SU=SQRT((PX(J+1,K+1)-PX(J,K))**2+(PY(J+1,K+1)-PY(J,K))**2+        PAN22450
     1(PZ(J+1,K+1)-PZ(J,K))**2)                                         PAN22500
C                                                                       PAN22550
C                                                                       PAN22600
C                                                                       PAN22650
C     CHECK IF TU AND SU ARE ZERO. IF THEY BOTH ARE THEN THE INPUT PANELPAN22700
C     WAS PROJECTED ONTO A LINE SUCH THAT THE NEW QUADRILATERAL IS      PAN22750
C     ATTEMPTING TO BE PLACED BETWEEN THE SAME X CROSS SECTION. THIS    PAN22800
C     PANEL WOULD THEN HAVE ZERO LENGTH ,FINITE WIDTH, BUT ZERO AREA.   PAN22850
C     IT IS VERY UNLIKELY , BUT IF IT OCCURS MAKE STOPER=1.0 . RUN      PAN22900
C     WILL FAIL.                                                        PAN22950
      IF((TU.NE 0.0).OR.(SU.NE.0.0)) GO TO 301                          PAN23000
      JJJJ=J+1                                                          PAN23050
      KKKK=K+1                                                          PAN23100
      WRITE(6,913)I,J,JJJJ,K,KKKK                                       PAN23150
      WRITE(6,922) J,K,STOX1(K),STOY1(K),STOZ1(K),KKKK,STOX1(KKKK),     PAN23200
     1STOY1(KKKK),STOZ1(KKKK)                                           PAN23250
      WRITE(6,923) JJJJ,K,STOX2(K),STOY2(K),STOZ2(K),KKKK,STOX2(KKKK),  PAN23300
     1STOY2(KKKK),STOZ2(KKKK)                                           PAN23350
      STOPER=1.0                                                        PAN23400
```

```
      GO TO 600                                                         PAN23450
C                                                                       PAN23500
C                                                                       PAN23550
  301 CONTINUE                                                          PAN23600
C                                                                       PAN23650
C     CHECK NOW TO SEE IF THE PLANAR QUADRILATERAL IS A TRIANGLE OF     PAN23700
C     THE CASE D OR CASE E TYPE. IF IT IS BRANCH TO SPECIAL EQUATIONS   PAN23750
C     TO GET THE AREA AND CONTROL POINT LOCATION OF IT.                 PAN23800
C     IF ST= 0.0  -- CASE D APPLIES TO THE PANEL.                       PAN23850
C     IF UR= 0.0  -- CASE E APPLIES TO THE PANEL.                       PAN23900
      IF(ST.EQ.0.0) GO TO 400                                           PAN23950
      IF(UR.EQ.0.0) GO TO 500                                           PAN24000
C                                                                       PAN24050
C     OTHERWISE --THIS IS CASE C --- A TRUE PLANAR QUADRILATERAL.       PAN24100
C     COMPUTE AREA USING FORMULA FOR GENERAL PLANE QUADRILATERALS. ALSO PAN24150
C     GET THE CONTROL POINT COORDINATES XC, YC, AND ZC, AT THE CENTROID PAN24200
C     OF AREA OF THE QUADRILATERAL.                                     PAN24250
C                                                                       PAN24300
      SANG=((RS**2)+(ST**2)-(RT**2))/(2.*RS*ST)                         PAN24350
      RANG=((UR**2)+(RS**2)-(SU**2))/(2.*UR*RS)                         PAN24400
C     CHECK : IF EITHER OR BOTH ABS(RANG) OR ABS(SANG) IS MORE THAN 1,  PAN24450
C     A FATAL ROUNDOFF ERROR HAS OCCURED. THE CAUSE IS PROBABLY DUE TO  PAN24500
C     CORNER POINTS BEING TO CLOSE TOGETHER FOR NUMERICAL STABILITY.    PAN24550
C     SET STOPER = 1.0.                                                 PAN24600
C                                                                       PAN24650
      IF(ABS(SANG).GT.1.0) GO TO 302                                    PAN24700
      ANGS=ARCOS(SANG)                                                  PAN24750
      GO TO 303                                                         PAN24800
  302 JJJJ=J+1                                                          PAN24850
      KKKK=K+1                                                          PAN24900
      WRITE(6,919)K,KKKK,J,K,KKKK,JJJJ                                  PAN24950
      STOPER=1.0                                                        PAN25000
      GO TO 600                                                         PAN25050
  303 IF(ABS(RANG).GT.1.0) GO TO 304                                    PAN25100
      ANGR=ARCOS(RANG)                                                  PAN25150
      GO TO 305                                                         PAN25200
```

```
  304 JJJJ=J+1                                                      PAN25250
      KKKK=K+1                                                      PAN25300
      WRITE(6,919)K,KKKK,J,K,KKKK,JJJJ                              PAN25350
      STOPER=1.0                                                    PAN25400
      GO TO 600                                                     PAN25450
  305 CONTINUE                                                      PAN25500
C                                                                   PAN25550
C     GET SURFACE AREA , S(I), FOR QUADRILATERAL. CHECK VALIDITY OF PAN25600
C     AREA VALUE BEFORE ACCEPTING IT.                               PAN25650
      F=(RS+ST+TU+UR)/2.                                            PAN25700
      AREASQ=((F-RS)*(F-ST)*(F-TU)*(F-UR))-(RS*ST*TU*UR*(COS        PAN25750
     1((ANGS+ANGR)/2  ))**2)                                        PAN25800
      IF(AREASQ.GE.0.0) GO TO 306                                   PAN25850
C     AREA INVALID - PRINT ERROR.                                   PAN25900
      JJJJ=J+1                                                      PAN25950
      KKKK=K+1                                                      PAN26000
      WRITE(6,920)I J,JJJJ,K,KKKK                                   PAN26050
      WRITE(6,921)                                                  PAN26100
      WRITE(6,922)J,K,STOX1(K),STOY1(K),STOZ1(K),KKKK,STOX1(KKKK),  PAN26150
     1STOY1(KKKK),STOZ1(KKKK)                                       PAN26200
      WRITE(6,923)JJJJ,K,STOX2(K),STOY2(K),STOZ2(K),KKKK,STOX2(KKKK),
                                                                    PAN26250
     1STOY2(KKKK),STOZ2(KKKK)                                       PAN26300
      STOPER=1.0                                                    PAN26350
      GO TO 600                                                     PAN26400
C                                                                   PAN26450
  306 S(I)=SQRT(AREASQ)                                             PAN26500
C                                                                   PAN26550
C     AREA VALID AND ACCEPTED.                                      PAN26600
C     MIRROR IMAGE PANEL SAME AREA ASSIGNED. SYMMETRIC INPUT ONLY : PAN26650
      IF(NSYMET.EQ.0) S(NPNSYM(I))=S(I)                             PAN26700
C                                                                   PAN26750
C                                                                   PAN26800
C     COMPUTE COORDINATES OF CONTROL POINT XC, YC, AND ZC ON THE I TH
                                                                    PAN26850
C     QUADRILATERAL . LOCATION AT CENTROID OF AREA. -- CASE C--     PAN26900
      GX=(PX(J,K)+PX(J+1,K))/2.                                     PAN26950
      GY=(PY(J,K)+PY(J+1,K))/2.                                     PAN27000
```

```
      GZ=(PZ(J,K)+PZ(J+1,K))/2.                                          PAN27050
      HX=(PX(J,K+1)+PX(J+1,K+1))/2.                                      PAN27100
      HY=(PY(J,K+1)+PY(J+1,K+1))/2.                                      PAN27150
      HZ=(PZ(J,K+1)+PZ(J+1,K+1))/2.                                      PAN27200
      XC(I)=(GX+HX)/2.                                                   PAN27250
      YC(I)=(GY+HY)/2.                                                   PAN27300
      ZC(I)=(GZ+HZ)/2.                                                   PAN27350
C                                                                        PAN27400
C     FOR SYMMETRIC INPUT ONLY- ASSIGN CONTROL POINT COORDINATES TO      PAN27450
C     MIRROR IMAGE PANELS :                                              PAN27500
      IF(NSYMET.NE.0) GO TO 307                                          PAN27550
      XC(NPNSYM(I))=XC(I)                                                PAN27600
      YC(NPNSYM(I))=-YC(I)                                               PAN27650
      ZC(NPNSYM(I))=ZC(I)                                                PAN27700
  307 CONTINUE                                                           PAN27750
C                                                                        PAN27800
C     CALCULATIONS COMPLETE ON I TH PANEL ( AND ITS IMAGE PANEL IF       PAN27850
C     SYMMETRIC INPUT USED). PRINT TABLE ENTRY IF REQUIRED AND GO TO     PAN27900
C     I+1 TH PANEL.                                                      PAN27950
C                                                                        PAN28000
      IF((NCALC.EQ.0).AND.(NLIST.EQ.1)) GO TO 600                        PAN28050
      WRITE(6,906) STOX1(K),STOY1(K),STOZ1(K)                            PAN28100
      WRITE(6,906) STOX1(K+1),STOY1(K+1),STOZ1(K+1)                      PAN28150
      WRITE(6,907) I,STOX2(K),STOY2(K),STOZ2(K),XC(I),YC(I),ZC(I),S(I)   PAN28200
      WRITE(6,906) STOX2(K+1),STOY2(K+1),STOZ2(K+1)                      PAN28250
      WRITE(6,908)(DASHES(KJL),DASHES(KJL),DASHES(KJL),KJL=1,11)         PAN28300
C                                                                        PAN28350
C     NEXT IF SYMMETRIC OPTION USED, PRINT THE IMAGE PANEL (OPPOSITE)    PAN28400
C     TO PANEL I:                                                        PAN28450
C                                                                        PAN28500
      IF(NSYMET.EQ.1) GO TO 600                                          PAN28550
      PRX1=STOX1(K+1)                                                    PAN28600
      PRY1=-STOY1(K+1)                                                   PAN28650
      PRZ1=STOZ1(K+1)                                                    PAN28700
      PRX2=STOX1(K)                                                      PAN28750
      PRY2=-STOY1(K)                                                     PAN28800
```

```
      PRZ2=STOZ1(K)                                                     PAN28850
      PRX3=STOX2(K+1)                                                   PAN28900
      PRY3=-STOY2(K+1)                                                  PAN28950
      PRZ3=STOZ2(K+1)                                                   PAN29000
      PRX4=STOX2(K)                                                     PAN29050
      PRY4=-STOY2(K)                                                    PAN29100
      PRZ4=STOZ2(K)                                                     PAN29150
      WRITE(6,906) PRX1,PRY1,PRZ1                                       PAN29200
      WRITE(6,909) PRX2,PRY2,PRZ2                                       PAN29250
      WRITE(6,910) NPNSYM(I),PRX3,PRY3,PRZ3,XC(NPNSYM(I)),YC(NPNSYM(I)),PAN29300
     +ZC(NPNSYM(I)),S(NPNSYM(I))                                        PAN29350
      WRITE(6,911) I,PRX4,PRY4,PRZ4                                     PAN29400
      WRITE(6,908)(DASHES(KJL),DASHES(KJL),DASHES(KJL),KJL=1,11)        PAN29450
C                                                                       PAN29500
      GO TO 600                                                         PAN29550
C                                                                       PAN29600
C-----------------------------------------------------------------------PAN29650
C                                                                       PAN29700
C                                                                       PAN29750
C                                                                       PAN29800
  400 CONTINUE                                                          PAN29850
C                                                                       PAN29900
C     -- THIS IS CASE D --.  PROJECTED TRIANGULAR PANEL.  GET SURFACE   PAN29950
C     AREA AND CONTROL POINT LOCATION:                                  PAN30000
C                                                                       PAN30050
      STANG=((RS**2)+(TU**2)-(UR**2))/(2.*RS*TU)                        PAN30100
C                                                                       PAN30150
C     CHECK ABS(STANG). IF IT IS MORE THAN 1.0 , ROUNDOFF ERROR HAS     PAN30200
C     OCCURED.  THIS IS DUE POSSIBLY TO HAVING CORNER POINTS TOO CLOSE  PAN30250
C     TOGETHER FOR NUMERICAL STABILITY. SET STOPER= 1.0  .              PAN30300
      IF(ABS(STANG).GT.1.0) GO TO 401                                   PAN30350
      ANGST=ARCOS(STANG)                                                PAN30400
      GO TO 402                                                         PAN30450
  401 JJJJ=J+1                                                          PAN30500
      KKKK=K+1                                                          PAN30550
      WRITE(6,924)K,KKKK,J,K,JJJJ                                       PAN30600
```

```
      STOPER=1.0
      GO TO 600
  402 CONTINUE
C
C     COMPUTE SURFACE AREA, S(I), OF THE TRIANGULAR PANEL -- CASE D:
      SINE=SQRT(1. - ((COS(ANGST))**2))
      S(I)=(RS * TU * SINE)/2.
C     MIRROR IMAGE PANEL SAME AREA ASSIGNED. SYMMETRIC INPUT ONLY :
      IF(NSYMET.EQ.0) S(NPNSYM(I))=S(I)
C
C     COMPUTE CONTROL POINT COORDINATES OF TRIANGULAR PANEL-- CASE D--:
      DIFFX=ABS((PX(J+1,K)-PX(J,K))/3.)
      DIFFY=ABS((PY(J+1,K)-PY(J,K))/3.)
      DIFFZ=ABS((PZ(J+1,K)-PZ(J K))/3.)
      IF(PX(J,K).LT.PX(J+1,K)) GO TO 403
      DX=PX(J,K)-DIFFX
      GO TO 404
  403 DX=PX(J,K)+DIFFX
  404 IF(PY(J,K).LT.PY(J+1,K)) GO TO 405
      DY=PY(J,K)-DIFFY
      GO TO 406
  405 DY=PY(J,K)+DIFFY
  406 IF(PZ(J,K).LT.PZ(J+1,K)) GO TO 407
      DZ=PZ(J,K)-DIFFZ
      GO TO 408
  407 DZ=PZ(J,K)+DIFFZ
  408 DIFFX=ABS((PX(J+1,K)-PX(J+1,K+1))/3.)
      DIFFY=ABS((PY(J+1,K)-PY(J+1,K+1))/3.)
      DIFFZ=ABS((PZ(J+1,K)-PZ(J+1,K+1))/3.)
      IF(PX(J+1,K+1).LT.PX(J+1,K)) GO TO 409
      EX=PX(J+1,K+1)-DIFFX
      GO TO 410
  409 EX=PX(J+1,K+1)+DIFFX
  410 IF(PY(J+1,K+1).LT.PY(J+1,K)) GO TO 411
      EY=PY(J+1,K+1)-DIFFY
      GO TO 412
```

```
  411 EY=PY(J+1,K+1)+DIFFY                                          PAN32450
  412 IF(PZ(J+1,K+1).LT.PZ(J+1,K)) GO TO 413                        PAN32500
      EZ=PZ(J+1,K+1)-DIFFZ                                          PAN32550
      GO TO 414                                                     PAN32600
  413 EZ=PZ(J+1,K+1)+DIFFZ                                          PAN32650
  414 XC(I)=(DX+EX)/2.                                              PAN32700
      YC(I)=(DY+EY)/2.                                              PAN32750
      ZC(I)=(DZ+EZ)/2.                                              PAN32800
C                                                                   PAN32850
C     FOR SYMMETRIC INPUT ONLY- ASSIGN CONTROL POINT COORDINATES TO PAN32900
C     MIRROR IMAGE PANELS :                                         PAN32950
      IF(NSYMET.NE.0) GO TO 415                                     PAN33000
      XC(NPNSYM(I))=XC(I)                                           PAN33050
      YC(NPNSYM(I))=-YC(I)                                          PAN33100
      ZC(NPNSYM(I))=ZC(I)                                           PAN33150
  415 CONTINUE                                                      PAN33200
C                                                                   PAN33250
C     CALCULATIONS COMPLETE ON I TH PANEL ( AND ITS IMAGE PANEL IF  PAN33300
C     SYMMETRIC INPUT USED). PRINT TABLE ENTRY IF REQUIRED AND GO TO PAN33350
C     I+1 TH PANEL.                                                 PAN33400
C                                                                   PAN33450
      IF((NCALC.EQ.0).AND.(NLIST.EQ.1)) GO TO 600                   PAN33500
      WRITE(6,906) STOX1(K),STOY1(K),STOZ1(K)                       PAN33550
      WRITE(6,906) STOX1(K+1),STOY1(K+1),STOZ1(K+1)                 PAN33600
      WRITE(6,907) I,STOX2(K),STOY2(K),STOZ2(K),XC(I),YC(I),ZC(I),S(I) PAN33650
      WRITE(6,906) STOX2(K+1),STOY2(K+1),STOZ2(K+1)                 PAN33700
      WRITE(6,908)(DASHES(KJL),DASHES(KJL),DASHES(KJL),KJL=1,11)    PAN33750
C                                                                   PAN33800
C     NEXT IF SYMMETRIC OPTION USED, PRINT THE IMAGE PANEL (OPPOSITE) PAN33850
C     TO PANEL I:                                                   PAN33900
C                                                                   PAN33950
      IF(NSYMET.EQ.1) GO TO 600                                     PAN34000
      PRX1=STOX1(K+1)                                               PAN34050
      PRY1=-STOY1(K+1)                                              PAN34100
      PRZ1=STOZ1(K+1)                                               PAN34150
      PRX2=STOX1(K)                                                 PAN34200
```

```
      PRY2=-STOY1(K)                                                    PAN34250
      PRZ2=STOZ1(K)                                                     PAN34300
      PRX3=STOX2(K+1)                                                   PAN34350
      PRY3=-STOY2(K+1)                                                  PAN34400
      PRZ3=STOZ2(K+1)                                                   PAN34450
      PRX4=STOX2(K)                                                     PAN34500
      PRY4=-STOY2(K)                                                    PAN34550
      PRZ4=STOZ2(K)                                                     PAN34600
      WRITE(6,906) PRX1,PRY1,PRZ1                                       PAN34650
      WRITE(6,909) PRX2,PRY2,PRZ2                                       PAN34700
      WRITE(6,910) NPNSYM(I),PRX3,PRY3,PRZ3,XC(NPNSYM(I)),YC(NPNSYM(I)),PAN34750
     +ZC(NPNSYM(I)),S(NPNSYM(I))                                        PAN34800
      WRITE(6,911) I,PRX4,PRY4,PRZ4                                     PAN34850
      WRITE(6,908)(DASHES(KJL) DASHES(KJL),DASHES(KJL),KJL=1,11)        PAN34900
C                                                                       PAN34950
      GO TO 600                                                         PAN35000
C                                                                       PAN35050
C-----------------------------------------------------------------------PAN35100
C                                                                       PAN35150
C                                                                       PAN35200
C                                                                       PAN35250
  500 CONTINUE                                                          PAN35300
C                                                                       PAN35350
C     -- THIS IS CASE E --  . PROJECTED TRIANGULAR PANEL.  COMPUTE AREA PAN35400
C     AND CONTROL POINT LOCATION.                                       PAN35450
C                                                                       PAN35500
      URANG=((RS**2)+(TU**2)-(ST**2))/(2.*RS*TU)                        PAN35550
C                                                                       PAN35600
C     CHECK ABS(URANG). IF IT IS MORE THAN 1.0 , A ROUNDOFF ERROR HAS   PAN35650
C     OCCURED.  THIS IS POSSIBLY DUE TO HAVING CORNER POINTS TOO CLOES  PAN35700
C     FOR NUMERICAL STABILITY.  SET STOPER=1.0  .                       PAN35750
      IF(ABS(URANG).GT.1.0) GO TO 501                                   PAN35800
      ANGUR=ARCOS(URANG)                                                PAN35850
      GO TO 502                                                         PAN35900
  501 JJJJ=J+1                                                          PAN35950
      KKKK=K+1                                                          PAN36000
```

```
      WRITE(6,925)K,J,K,KKKK,JJJJ                                       PAN36050
      STOPER=1.0                                                        PAN36100
      GO TO 600                                                         PAN36150
  502 CONTINUE                                                          PAN36200
C                                                                       PAN36250
C     COMPUTE SURFACE AREA, S(I), OF THIS TRIANGULAR PANEL -- CASE E --.PAN36300
      SINE=SQRT(1.-((COS(ANGUR))**2))                                   PAN36350
      S(I)=(RS * TU * SINE)/2.                                          PAN36400
C     MIRROR IMAGE PANEL SAME AREA ASSIGNED. SYMMETRIC INPUT ONLY :     PAN36450
      IF(NSYMET.EQ.0) S(NPNSYM(I))=S(I)                                 PAN36500
C                                                                       PAN36550
C     COMPUTE CONTROL POINT LOCATION FOR TRIANGULAR PANEL. CASE E.      PAN36600
      DIFFX=ABS((PX(J,K)-PX(J,K+1))/3.)                                 PAN36650
      DIFFY=ABS((PY(J,K)-PY(J,K+1))/3.)                                 PAN36700
      DIFFZ=ABS((PZ(J,K)-PZ(J,K+1))/3.)                                 PAN36750
      IF(PX(J,K+1).LT.PX(J,K)) GO TO 503                                PAN36800
      DX=PX(J,K+1)-DIFFX                                                PAN36850
      GO TO 504                                                         PAN36900
  503 DX=PX(J,K+1)+DIFFX                                                PAN36950
  504 IF(PY(J,K+1).LT.PY(J,K)) GO TO 505                                PAN37000
      DY=PY(J,K+1)-DIFFY                                                PAN37050
      GO TO 506                                                         PAN37100
  505 DY=PY(J,K+1)+DIFFY                                                PAN37150
  506 IF(PZ(J,K+1).LT.PZ(J,K)) GO TO 507                                PAN37200
      DZ=PZ(J,K+1)-DIFFZ                                                PAN37250
      GO TO 508                                                         PAN37300
  507 DZ=PZ(J,K+1)+DIFFZ                                                PAN37350
  508 DIFFX=ABS((PX(J,K)-PX(J+1,K))/3.)                                 PAN37400
      DIFFY=ABS((PY(J,K)-PY(J+1,K))/3.)                                 PAN37450
      DIFFZ=ABS((PZ(J,K)-PZ(J+1,K))/3.)                                 PAN37500
      IF(PX(J+1,K).LT.PX(J,K)) GO TO 509                                PAN37550
      EX=PX(J+1,K)-DIFFX                                                PAN37600
      GO TO 510                                                         PAN37650
  509 EX=PX(J+1,K)+DIFFX                                                PAN37700
  510 IF(PY(J+1,K).LT.PY(J,K)) GO TO 511                                PAN37750
      EY=PY(J+1,K)-DIFFY                                                PAN37800
```

```
      GO TO 512                                                         PAN37850
  511 EY=PY(J+1,K)+DIFFY                                                PAN37900
  512 IF(PZ(J+1,K).LT.PZ(J,K)) GO TO 513                                PAN37950
      EZ=PZ(J+1,K)-DIFFZ                                                PAN38000
      GO TO 514                                                         PAN38050
  513 EZ=PZ(J+1,K)+DIFFZ                                                PAN38100
  514 XC(I)=(DX+EX)/2.                                                  PAN38150
      YC(I)=(DY+EY)/2.                                                  PAN38200
      ZC(I)=(DZ+EZ)/2.                                                  PAN38250
C                                                                       PAN38300
C     FOR SYMMETRIC INPUT ONLY- ASSIGN CONTROL POINT COORDINATES TO     PAN38350
C     MIRROR IMAGE PANELS :                                             PAN38400
      IF(NSYMET.NE.0) GO TO 515                                         PAN38450
      XC(NPNSYM(I))=XC(I)                                               PAN38500
      YC(NPNSYM(I))=-YC(I)                                              PAN38550
      ZC(NPNSYM(I))=ZC(I)                                               PAN38600
  515 CONTINUE                                                          PAN38650
C                                                                       PAN38700
C     CALCULATIONS COMPLETE ON I TH PANEL ( AND ITS IMAGE PANEL IF      PAN38750
C     SYMMETRIC INPUT USED). PRINT TABLE ENTRY IF REQUIRED AND GO TO    PAN38800
C     I+1 TH PANEL.                                                     PAN38850
C                                                                       PAN38900
      IF((NCALC.EQ.0).AND.(NLIST.EQ.1)) GO TO 600                       PAN38950
      WRITE(6,906) STOX1(K),STOY1(K),STOZ1(K)                           PAN39000
      WRITE(6,906) STOX1(K+1) STOY1(K+1),STOZ1(K+1)                     PAN39050
      WRITE(6,907) I,STOX2(K),STOY2(K),STOZ2(K),XC(I),YC(I),ZC(I),S(I)  PAN39100
      WRITE(6,906) STOX2(K+1),STOY2(K+1),STOZ2(K+1)                     PAN39150
      WRITE(6,908)(DASHES(KJL),DASHES(KJL),DASHES(KJL),KJL=1,11)        PAN39200
C                                                                       PAN39250
C     NEXT IF SYMMETRIC OPTION USED, PRINT THE IMAGE PANEL (OPPOSITE)   PAN39300
C     TO PANEL I:                                                       PAN39350
C                                                                       PAN39400
      IF(NSYMET.EQ.1) GO TO 600                                         PAN39450
      PRX1=STOX1(K+1)                                                   PAN39500
      PRY1=-STOY1(K+1)                                                  PAN39550
      PRZ1=STOZ1(K+1)                                                   PAN39600
```

```
      PRX2=STOX1(K)                                                     PAN39650
      PRY2=-STOY1(K)                                                    PAN39700
      PRZ2=STOZ1(K)                                                     PAN39750
      PRX3=STOX2(K+1)                                                   PAN39800
      PRY3=-STOY2(K+1)                                                  PAN39850
      PRZ3=STOZ2(K+1)                                                   PAN39900
      PRX4=STOX2(K)                                                     PAN39950
      PRY4=-STOY2(K)                                                    PAN40000
      PRZ4=STOZ2(K)                                                     PAN40050
      WRITE(6,906) PRX1,PRY1,PRZ1                                       PAN40100
      WRITE(6,909) PRX2 PRY2,PRZ2                                       PAN40150
      WRITE(6,910) NPNSYM(I),PRX3,PRY3,PRZ3,XC(NPNSYM(I)),YC(NPNSYM(I)),PAN40200
     +ZC(NPNSYM(I)),S(NPNSYM(I))                                        PAN40250
      WRITE(6,911) I,PRX4,PRY4,PRZ4                                     PAN40300
      WRITE(6,908)(DASHES(KJL),DASHES(KJL),DASHES(KJL),KJL=1,11)        PAN40350
C                                                                       PAN40400
C-----------------------------------------------------------------------PAN40450
C                                                                       PAN40500
C                                                                       PAN40550
C                                                                       PAN40600
C                                                                       PAN40650
  600 IF(NSYMET.NE.0) GO TO 601                                         PAN40700
C                                                                       PAN40750
C     M ACCUMULATES CURRENT PANEL COUNT AS THIS STATION OF PANELS IS    PAN40800
C     GENERATED.                                                        PAN40850
C                                                                       PAN40900
      M=I+K                                                             PAN40950
      NP=M                                                              PAN41000
      GO TO 700                                                         PAN41050
  601 M=I                                                               PAN41100
      NP=M                                                              PAN41150
C                                                                       PAN41200
  700 CONTINUE                                                          PAN41250
C                                                                       PAN41300
C     HAVE GENERATED ANOTHER COMPLETE STATION OF PANELS. STORE CURRENT  PAN41350
C     PANEL COUNT IN MTOT BEFORE MOVING TO THE NEXT STATION OF PANELS.  PAN41400
```

```
C     IF THIS WAS THE LAST STATION OF PANELS TO BE DONE, THEN NP       PAN41450
C     CONTAINS THE TOTAL NUMBER OF PANELS GENERATED FOR ENTIRE CONFIG. PAN41500
C     THEN RETURN TO MAIN PROGRAM                                      PAN41550
C                                                                      PAN41600
      MTOT=M                                                           PAN41650
      IF(J.EQ.N) GO TO 800                                             PAN41700
      IF(NFLAG(J+1).EQ.1) GO TO 4                                      PAN41750
      J=J+1                                                            PAN41800
      LNIP=NCOUNT(J)-1                                                 PAN41850
      NCO=NCOUNT(J)                                                    PAN41900
      DO 701 K=1,NCO                                                   PAN41950
      STOX1(K)=STOX2(K)                                                PAN42000
      STOY1(K)=STOY2(K)                                                PAN42050
      STOZ1(K)=STOZ2(K)                                                PAN42100
      STOX2(K)=PX(J+1,K)                                               PAN42150
      STOY2(K)=PY(J+1,K)                                               PAN42200
      STOZ2(K)=PZ(J+1,K)                                               PAN42250
  701 CONTINUE                                                         PAN42300
      GO TO 6                                                          PAN42350
  800 IF((NCALC.EQ.1).AND.(STOPER.EQ.0.0)) GO TO 801                   PAN42400
      IF(STOPER.EQ.1.0) GO TO 802                                      PAN42450
      IF((NCALC.EQ.0).AND.(NLIST.EQ.1)) GO TO 803                      PAN42500
      IF((NCALC.EQ.0).AND.(NLIST.EQ.0)) GO TO 804                      PAN42550
  801 WRITE(6,912) NP                                                  PAN42600
      WRITE(6,913)                                                     PAN42650
      WRITE(6,914)(STARS(KJL),STARS(KJL),STARS(KJL),KJL=1,11)          PAN42700
      STOP                                                             PAN42750
  802 WRITE(6,926) NP                                                  PAN42800
C     RUN STOPS BECAUSE ONE OR MORE FATAL PANELING PROBLEMS EXIST.     PAN42850
      STOP                                                             PAN42900
  803 WRITE(6,912) NP                                                  PAN42950
      WRITE(6,914)(STARS(KJL),STARS(KJL),STARS(KJL),KJL=1,11)          PAN43000
      RETURN                                                           PAN43050
  804 WRITE(6,912) NP                                                  PAN43100
      WRITE(6,915)                                                     PAN43150
      WRITE(6,914)(STARS(KJL).STARS(KJL),STARS(KJL),KJL=1,11)          PAN43200
```

```
      RETURN                                                            PAN43250
C-----------------------------------------------------------------------PAN43300
C                                                                       PAN43350
C     FORMATS                                                           PAN43400
C                                                                       PAN43450
  900 FORMAT('1',T45,11A4)                                              PAN43500
  901 FORMAT(' ',T45,'* LISTING OF GENERATED BODY PANEL GEOMETRY *')    PAN43550
  902 FORMAT(' ',T45,11A4//)                                            PAN43600
  903 FORMAT(' ',T19,'//// SYMMETRIC BODY OPTION USED. MIRROR IMAGE (RIGPAN43650
     1HT SIDE) PANELS AUTOMATICALLY GENERATED. ////'//)                 PAN43700
  904 FORMAT('0',T29,'CORNER POINTS (1 AND 2 ON LEADING'/,' ',T29,      PAN43750
     1'SECTION, 3 AND 4 ON TRAILING SECTION)'/,' ',T2,'PANEL',T121,     PAN43800
     2'PANEL'/,' ',T2,'INDEX',T31,'X1',T46,'Y1',T61,'Z1',T121,'SURFACE'/PAN43850
     3,' ',T2,'NUMBER',T31,'X2',T46,'Y2',T61,'Z2',T77,'PANEL CONTROL POIPAN43900
     4NT COORDINATES',T122,'AREA'/,' ',T31,'X3',T46,'Y3',T61,'Z3'/,' ', PAN43950
     5T4,'I',T31,'X4',T46,'Y4',T61,'Z4',T76,'XC(I)',T91,'YC(I)',T106,   PAN44000
     6'ZC(I)',T122,'S(I)')                                              PAN44050
  905 FORMAT('0',T25,'//// NO PANEL GEOMETRY LIST REQUESTED ON THIS RUN PAN44100
     1- CONTINUE ON TO FLOW SOLUTIONS ////'/)                           PAN44150
  906 FORMAT(' ',T26,F13.6,T41,F13.6,T56,F13.6)                         PAN44200
  907 FORMAT(' ',T3,I4,T26,F13.6,T41,F13.6,T56,F13.6,T72,F13.6,T87,F13.6PAN44250
     1,T102,F13.6,T118,F13.6)                                           PAN44300
  908 FORMAT(' ',33A4)                                                  PAN44350
  909 FORMAT(' ',T10,'(THIS IS RIGHT',T26,F13.6,T41,F13.6,T56,F13.6)    PAN44400
  910 FORMAT(' ',T3,I4,T10,'SIDE IMAGE OF',T26,F13.6,T41,F13.6,T56,     PAN44450
     1F13.6,T72,F13.6,T87,F13.6,T102,F13.6,T118,F13.6)                  PAN44500
  911 FORMAT(' ',T10,'PANEL ',I4,')',T26,F13.6,T41,F13.6,T56,F13.6)     PAN44550
  912 FORMAT('0',T37,'//// ',I4,' TOTAL PANELS IN BODY GENERATED SUCCESSPAN44600
     1FULLY. ////')                                                     PAN44650
  913 FORMAT(' ',T21,'TEST RUN FOR LISTING OF GEOMETRY ONLY IS COMPLETE PAN44700
     1- NO FLOWS PREDICTED. END OF JOB. ////')                          PAN44750
  914 FORMAT('0' 33A4)                                                  PAN44800
  915 FORMAT(' ',T30,'//// END OF BODY PANEL GEOMETRY LISTING. MOVE ON TPAN44850
     1O FLOW CALCULATIONS. ////')                                       PAN44900
  916 FORMAT('0','DURING PANELING OF TRIANGULAR PANEL BETWEEN POINT K=',PAN44950
     1I3,' ON SECTION NSEC=',I3,' AND POINTS K=' I3,' AND K+1=',I3,' ON PAN45000
```

```
     2SECTION'/,' ','NSEC=',I3,' , NUMERICAL ROUNDOFF HAS OCCURRED. POSSPAN45050
     3IBLE CAUSE IS THAT PANEL IS TOO SMALL DUE TO THE TWO SECTIONS'/,   PAN45100
     4' ','LYING TOO CLOSE OR DUE TO THE TWO POINTS ON THE SECOND SECTIOPAN45150
     5N LYING TOO CLOSE TOGETHER FOR NUMERICAL STABILITY.'/,' ',         PAN45200
     6'PANELING INPUT IN THIS REGION MUST BE CORRECTED BEFORE PROGRAM WIPAN45250
     7LL WORK. PANEL GENERATION CONTINUES FOR'/,' ','ERROR CHECK. THEN PPAN45300
     8ROGRAM ABORTS.')                                                   PAN45350
 917 FORMAT('0','DURING PANELING OF TRIANGULAR PANEL BETWEEN POINTS K='PAN45400
     1,I3,' AND K+1=',I3,' ON SECTION NSEC=',I3,' AND POINT K=',I3,      PAN45450
     2'ON SECTION NSEC=',I3/,' ',' , NUMERICAL ROUNDOFF ERROR HAS OCCURRPAN45500
     3ED. POSSIBLE CAUSE IS THAT PANEL IS TOO SMALL DUE TO THE TWO SECTIPAN45550
     4ONS'/,' ','LYING TOO CLOSE OR DUE TO THE TWO POINTS ON THE FIRST SPAN45600
     5ECTION LYING TOO CLOSE TOGETHER FOR NUMERICAL STABILITY.'/,' ',    PAN45650
     6'PANELING INPUT IN THIS REGION MUST BE CORRECTED BEFORE PROGRAM WIPAN45700
     7LL WORK. PANEL GENERATION CONTINUES FOR'/,' ','ERROR CHECK. THEN PPAN45750
     8ROGRAM ABORTS.')                                                   PAN45800
 918 FORMAT('0','AFTER PROJECTION OF FOUR CORNER POINTS ONTO A PLANE, IPAN45850
     1NVALID POINTS RESULT FOR PANEL NUMBER I=',I5,' - ATTEMPT TO PANEL'PAN45900
     2/,' ','BETWEEN TWO STATIONS OF SAME X VALUE. IT INVOLVES SECTION NPAN45950
     3SEC=',I3,' AND SECTION NSEC=',I3,' WITH INPUT POINTS',I3,' AND'    PAN46000
     4I3/,' ','ON EACH SECTION. PANEL INVALID. NO AREA FOUND. MUST MODIFPAN46050
     5Y PANEL INPUT TO THIS REGION. CONTINUE GENERATION OF PANELS'/,' ',PAN46100
     6'FOR ERROR CHECKS. THEN RUN ABORTS.')                              PAN46150
 919 FORMAT('0','DURING PANELING OF FOUR SIDED PANEL WITH CORNER POINTSPAN46200
     1, ',I3,', AND',I3,' OF SECTION NSEC=',I5,' AND POINTS',I3,', AND',PAN46250
     2I3/,' ','OF SECTION NSEC=',I5,' NUMERICAL ROUNDOFF ERROR OCCURRED.PAN46300
     3 POSSIBLE CAUSE IS ADJACENT SECTIONS OR POINTS TOO CLOSE'/,' ',    PAN46350
     4'TOGETHER FOR NUMERICAL STABILITY. INPUT MUST BE MODIFIED IN THIS PAN46400
     5REGION. CONTINUE GENERATION OF PANELS FOR'/,' ',                   PAN46450
     6'ERROR CHECKS. THEN RUN ABORTS.')                                  PAN46500
 920 FORMAT('0','**** TROUBLE IN PANELING INPUT FOR PANEL NUMBER',I4,' PAN46550
     1PANEL HAS ZERO OR IMAGINARY AREA.'/,' ','INVOLVES INPUT SECTIONS',PAN46600
     2I4,' AND',I4,' WITH POINTS NUMBER',I4,' AND',I4,' OF THESE SECTIONPAN46650
     3S.')                                                               PAN46700
 921 FORMAT(' ','THIS WAS TO BE A QUADRILATERAL ELEMENT.'/)              PAN46750
 922 FORMAT('0',T20,'THE FOUR INPUT CORNER POINTS INVOLVED ARE:'/,' ','PAN46800
```

C-3

```
     1SECTION',I4,' POINT',I4,' -(',E14.7,',',E14.7,',',E14.7,') AND POIPAN46850
     2NT',I4,' -(',E14 7,',',E14.7,',',E14.7,')')                        PAN46900
  923 FORMAT(' ','SECTION',I4,' POINT',I4,' -(',E14.7,',',E14.7,',',     PAN46950
     1E14.7,') AND POINT',I4,' -(',E14.7,',',E14.7,',',E14.7,')')        PAN47000
  924 FORMAT('0','DURING PANELING INVOLVING POINTS,',I3,', AND',I3,'ON SPAN47050
     1ECTION NSEC=',I3,' AND AT POINT ',I3,' ON SECTION'/,' ',           PAN47100
     2'NSEC=',I3,', A PLANE TRIANGULAR PANEL WAS GENERATED. ROUNDOFF ERRPAN47150
     3OR OCCURRED. POSSIBLE CAUSE IS ADJACENT SECTIONS OR'/,' ',         PAN47200
     4'ADJACENT POINTS OF FIRST SECTION BEING TOO CLOSE TOGETHER FOR NUMPAN47250
     5ERICAL STABILITY. MUST MODIFY INPUT IN THIS REGION.'/,' ',         PAN47300
     6'CONTINUE GENERATION OF PANELS FOR ERROR CHECKS. THEN RUN ABORTS.'PAN47350
     7)                                                                  PAN47400
  925 FORMAT('0','DURING PANELING INVOLVING POINT,',I3,' ON SECTION NSECPAN47450
     1= ',I3,', AND POINTS,',I3,', AND',I3,' ON SECTION'/,' ',           PAN47500
     2'NSEC=',I3 ', A PLANE TRIANGULAR PANEL WAS GENERATED. ROUNDOFF ERRPAN47550
     3OR OCCURRED. POSSIBLE CAUSE IS ADJACENT SECTIONS OR'/,' ',         PAN47600
     4'ADJACENT POINTS OF THE SECOND SECTION BEING TOO CLOSE TOGETHER FOPAN47650
     5R NUMERICAL STABILITY. MUST MODIFY INPUT IN THIS REGION.'/,' ',    PAN47700
     6'CONTINUE GENERATION OF PANELS FOR ERROR CHECKS. THEN RUN ABORTS.'PAN47750
     7)                                                                  PAN47800
  926 FORMAT('0','FATAL PANELING ERRORS FORCE PROGRAM STOPPAGE--WOULD HAPAN47850
     1VE BEEN',I5,' BODY PANELS ')                                       PAN47900
  927 FORMAT(' ',33A4/)                                                  PAN47950
C-----------------------------------------------------------------------PAN48000
C                                                                        PAN48050
      END                                                                PAN48100
C                                                                        PAN48150
C     *****************************************************************PAN48200
C                                                                        PAN48250
C                                                                        PAN48300
```

```
      SUBROUTINE COEFIC(LLL,MD,NP,XC,YC,ZC,S,ANVX,ANVY,ANVZ,A)          COF   50
C                                                                       COF  100
C     ******************************************************************COF  150
C     ******************************************************************COF  200
C     SETS UP SYSTEM OF NP LINEAR EQUATIONS. USED WHEN NONSYMMETRIC     COF  250
C     BODY INPUT HAS BEEN SPECIFIED. UNKNOWNS OF SYSTEM ARE NALPHA SETS COF  300
C     OF BODY SOURCE STRENGTHS (SIGMA)-(ONE SET PER BODY ANGLE OF       COF  350
C     ATTACK CASE). CALCULATE NP COEFFICIENTS OF AERODYNAMIC MATRIX     COF  400
C     (LEFT SIDE OF LINEAR EQUATION SYSTEM) IN I TH ROW. THEN FOR SAME  COF  450
C     I TH ROW, THE NALPHA CONSTANTS ON THE RIGHT SIDE OF THE SYSTEM ARECOF  500
C     FOUND. ON THE I TH ROW, THERE ARE AS MANY SUCH CONSTANTS AS THERE COF  550
C     ARE INPUT BODY ANGLES OF ATTACK-ONE CONSTANT PER CASE ON EACH ROW.COF  600
C     THIS I TH ROW WHEN COMPLETE IS CONTAINED IN ROW MATRIX, A,.  I TH COF  650
C     ROW COEFFICIENTS ARE TRANSFERRED TO THE I TH LOGICAL RECORD OF A  COF  700
C     SEQUENTIAL SCRATCH FILE 9 ON MAGNETIC TAPE OR DISK. PROCESS       COF  750
C     REPEATS CONSECUTIVELY FROM FIRST ROW OF SYSTEM TO LAST NP TH ROW  COF  800
C     OF SYSTEM. SUBROUTINE RETURNS TO MAIN PROGRAM WITH EACH ROW       COF  850
C     (EQUATION) OF THE SYSTEM STORED IN SEQUENTIAL ORDER ON THE        COF  900
C     SCRATCH FILE.                                                     COF  950
C     ******************************************************************COF 1000
C     ******************************************************************COF 1050
C                                                                       COF 1100
      DIMENSION A(MD),S(LLL),ANVX(LLL),ANVY(LLL),ANVZ(LLL),XC(LLL),     COF 1150
     1YC(LLL),ZC(LLL)                                                   COF 1200
      COMMON /INPUTS/ ALPHA(6),BETA(6),VX(6),VY(6),VZ(6),V,NALPHA       COF 1250
      COMMON /INLET/ FLRATO(500),INDEX(500),NINFLO                      COF 1300
      COMMON /WING1/ CL(6),CHORD,DIHED,SPAN,SWEEP,XQR,YQR,ZQR,NWING     COF 1350
      REWIND 9                                                          COF 1400
      DO 9 I=1,NP                                                       COF 1450
      DO 2 J=1,NP                                                       COF 1500
      RX=XC(I)-XC(J)                                                    COF 1550
      RY=YC(I)-YC(J)                                                    COF 1600
      RZ=ZC(I)-ZC(J)                                                    COF 1650
      R=SQRT(RX**2 + RY**2 + RZ**2)                                     COF 1700
C                                                                       COF 1750
C     COMPUTE LEFT SIDE J TH COEFFICIENT ON I TH ROW.                   COF 1800
```

```
C                                                                       COF 1850
      IF(I.EQ.J) GO TO 1                                                COF 1900
      A(J)=(S(J)/(2.*3.141593))*(1./(R**3))*((RX*ANVX(I))+(RY*ANVY(I))+ COF 1950
     1(RZ*ANVZ(I)))                                                     COF 2000
      GO TO 2                                                           COF 2050
    1 A(J)=1.0                                                          COF 2100
    2 CONTINUE                                                          COF 2150
C                                                                       COF 2200
C     CALCULATE THE NALPHA NUMBER OF CONSTANTS ON RIGHT SIDE OF EQUATIONCOF 2250
C     I. THIS IS FOR I TH PANEL. CHECK IF PANEL I IS A THROUGH FLOW     COF 2300
C     PANEL. IF IT IS, ADD THE THROUGH FLOW VELOCITY TO THE CONSTANT    COF 2350
C     ON THIS EQUATION TO RELAX BOUNDARY CONDITION. IF WING IS PRESENT, COF 2400
C     ADD TO THE FREE STREAM VELOCITY, THE COMPONENTS OF VELOCITY AT    COF 2450
C     CONTROL POINT I DUE TO WING.  INFADD IS 0 IF PANEL I IS NOT       COF 2500
C     A THROUGH FLOW PANEL. INFADD= 1 IF PANEL IS THROUGH FLOW PANEL.   COF 2550
C                                                                       COF 2600
      IF(NINFLO.EQ.0) GO TO 4                                           COF 2650
      DO 3 L=1,NINFLO                                                   COF 2700
      IF(INDEX(L).NE.I) GO TO 3                                         COF 2750
      LIN=L                                                             COF 2800
      INFADD=1                                                          COF 2850
      GO TO 5                                                           COF 2900
    3 CONTINUE                                                          COF 2950
    4 INFADD=0                                                          COF 3000
    5 CONTINUE                                                          COF 3050
      NUP1=NP+1                                                         COF 3100
      NUP=NP+NALPHA                                                     COF 3150
      DO 8 J=NUP1,NUP                                                   COF 3200
      K=J-NP                                                            COF 3250
      IF(NWING.EQ.0) GO TO 6                                            COF 3300
      CALL WINGV(K,XC(I),YC(I),ZC(I),VELX,VELY,VELZ)                    COF 3350
      A(J)=-2.0*((ANVX(I)*(VX(K)+VELX))+(ANVY(I)*(VY(K)+VELY))+         COF 3400
     1(ANVZ(I)*(VZ(K)+VELZ)))                                           COF 3450
      GO TO 7                                                           COF 3500
    6 A(J)=-2.0*((ANVX(I)*VX(K))+(ANVY(I)*VY(K))+(ANVZ(I)*VZ(K)))       COF 3550
    7 CONTINUE                                                          COF 3600
```

```
C                                                                       COF 3650
C     IF INFADD= 1, RELAX BOUNDARY CONDITION ON PANEL I                 COF 3700
      IF(INFADD.EQ.0) GO TO 8                                           COF 3750
      A(J)=A(J) + (-2.0*FLRATO(LIN)*V)                                  COF 3800
    8 CONTINUE                                                          COF 3850
C                                                                       COF 3900
C     PUT I TH ROW ON I TH RECORD OF DISK FILE.                         COF 3950
C                                                                       COF 4000
      WRITE(9)(A(M),M=1,NUP)                                            COF 4050
    9 CONTINUE                                                          COF 4100
      END FILE 9                                                        COF 4150
      REWIND 9                                                          COF 4200
      RETURN                                                            COF 4250
      END                                                               COF 4300
C                                                                       COF 4350
C     ******************************************************************COF 4400
C                                                                       COF 4450
C                                                                       COF 4500
C                                                                       COF 4550
```

```
      SUBROUTINE SOLVE(LLL,MD,MAXALP,NP,A,BIG,VARIAB,SUM,ERR,
     1ITMAX,SIGMA)
C
C     ******************************************************************
C     ******************************************************************
C     USES GAUSS-SIEDEL ITERATION TO SOLVE TOTAL SYSTEM OF NP EQUATIONS
C     FOR NALPHA SETS OF NP UNKNOWNS, SIGMA (BODY PANEL SOURCE STRENGTHS
C     -- THIS SUBROUTINE IS EXECUTED WHEN NONSYMMETRIC BODY INPUT OPTION
C     HAS BEEN SPECIFIED--. A SET OF SIGMAS IS FOUND FOR EACH DIFFERENT
C     BODY (ALPHA-BETA) ORIENTATION CASE. ALL INPUT ALPHA-BETA CASES
C     ARE SOLVED AT ONCE USING A SINGLE EXECUTION OF THIS SUBROUTINE.
C     ALL CASES MUST CONVERGE OR ENTIRE JOB ABORTS. EQUATIONS OF SYSTEM
C     ARE STORED SEQUENTIALLY IN A DISK OR TAPE SCRATCH FILE 9. ON ONE
C     ITERATION EACH EQUATION IS SOLVED ONE AT A TIME IN ORDER FROM
C     FIRST TO LAST ONE. DURING AN ITERATION THE CURRENT EQUATION
C     EXAMINED IS READ FROM ITS RECORD OF THE SCRATCH FILE AND STORED
C     TEMPORARILY IN ROW MATRIX, A, FOR MANIPULATIONS. NP IS NUMBER OF
C     EQUATIONS AND UNKNOWNS PER ALPHA CASE = NUMBER OF BODY PANELS.
C     ******************************************************************
C     ******************************************************************
C
      DIMENSION A(MD),SIGMA(LLL MAXALP)
      DIMENSION BIG(MAXALP),VARIAB(MAXALP),SUM(MAXALP)
      COMMON /INPUTS/ ALPHA(6),BETA(6),VX(6),VY(6),VZ(6),V,NALPHA
      NUP=NP+NALPHA
      ERROR =10.0**ERR
      WRITE(6,100)
      WRITE(6,200)ITMAX
      WRITE(6,300) ERROR
C
C     ESTABLISH INITIAL GUESS FOR UNKNOWNS:
C
      DO 1 I=1,NP
      DO 1 J=1,NALPHA
      SIGMA(I,J)=0.0
    1 CONTINUE
```

```
C
C     PERFORM ITERATIONS
C
      DO 10 ITER=1,ITMAX
      REWIND 9
      DO 2 K=1,NALPHA
      BIG(K)=0.0
    2 CONTINUE
      WRITE(6,400)ITER
      WRITE(6,500)
C
C     WORK ON I TH EQUATION OF SYSTEM
C
      DO 7 I=1,NP
      DO 3 K=1,NALPHA
      SUM(K)=0.0
    3 CONTINUE
      READ(9)(A(K),K=1,NUP)
C
C     GET SUM OF ROW I COEFFICIENT-UNKNOWN PRODUCTS FOR EACH ALPHA SET.
C
      DO 5 J=1,NP
      IF(J.EQ.I) GO TO 5
      DO 4 K=1,NALPHA
      SUM(K)=SUM(K)+(A(J)*SIGMA(J,K))
    4 CONTINUE
    5 CONTINUE
C
C     SOLVE I TH EQUATION FOR I TH UNKNOWN (OF EACH K TH ALPHA CASE) -
C     THIS ITERATION. CALL IT VARIAB(K)
C
      DO 6 K=1,NALPHA
      M=NP+K
      VARIAB(K)=(A(M)-SUM(K))/A(I)
      IF(ABS(VARIAB(K)-SIGMA(I,K)).GT.BIG(K)) BIG(K)=ABS(VARIAB(K)-
     1SIGMA(I,K))
```

```
      SIGMA(I,K)=VARIAB(K)
    6 CONTINUE
    7 CONTINUE
C
C     ALL EQUATIONS DONE THIS ITERATION. CHECK FOR CONVERGENCE.
C
      NSIGNL=0
      DO 9 K=1,NALPHA
      IF(BIG(K).LT.ERROR) GO TO 8
      NSIGNL=1
      WRITE(6,600)ALPHA(K),BETA(K),K,BIG(K)
      GO TO 9
    8 WRITE(6,601)ALPHA(K),BETA(K),K,BIG(K)
    9 CONTINUE
C
C     CHECK NSIGNL: IF NSIGNL=0 ALL ALPHA CASES HAVE CONVERGED TO GOOD
C     SOLUTIONS. IF NSIGNL=1(NOT =0) THEN ONE OR MORE CASES HAVE NOT YET
C     CONVERGED FULLY - GO BACK AND ITERATE ALL ALPHA CASES AGAIN.
C
      IF(NSIGNL.NE.0) GO TO 10
      WRITE(6,700) ITER
      REWIND 9
      RETURN
   10 CONTINUE
C
C     IF 10 DO LOOP IS COMPLETELY EXECUTED, THEN THE MAXIMUM ALLOWED
C     NUMBER OF ITERATIONS WAS PERFORMED AND ONE OR MORE CASES FAILED TO
C     CONVERGE TO ACCURATE SOLUTIONS.--ABORT RUN.
C
      WRITE(6,800)
      REWIND 9
      STOP
C
C     FORMATS
C
  100 FORMAT('1',T42,'**** SOLUTION OF SOURCE STRENGTHS, SIGMA ****'/)
```

```
  200 FORMAT(' ',T35,'SPECIFIED MAXIMUM PERMITTED SOLUTION ITERATIONS=',SOL 5450
     1I10)                                                              SOL 5500
  300 FORMAT(' ',T35,'SPECIFIED MAXIMUM PERMITTED SOLUTION ERROR     =',SOL 5550
     11PE10.0/)                                                         SOL 5600
  400 FORMAT('0',T55,'/// ITERATION ',I3,' ///'/)                       SOL 5650
  500 FORMAT(' ',T38,'ALPHA, BETA,-CASE',T75,'SOLUTION ERROR'/)         SOL 5700
  600 FORMAT(' ',T31,F13.7,', ',F13.7,', ',I2,T76,1PE12.5)              SOL 5750
  601 FORMAT(' ',T31,F13.7,', ',F13.7,', ',I2,T76,1PE12.5,T90,'(CASE HASSOL 5800
     1 CONVERGED)')                                                     SOL 5850
  700 FORMAT('0',T26,'--- SOURCE STRENGTH SOLUTIONS CONVERGED FOR ALL CASOL 5900
     1SES WITHIN',I5,' ITERATIONS ---'/)                                SOL 5950
  800 FORMAT('0',T27,'**** FAILURE OF SOLUTION CONVERGENCE FOR ONE OR MOSOL 6000
     1RE CASES - RUN ABORTED ****'/)                                    SOL 6050
C                                                                       SOL 6100
      END                                                               SOL 6150
C                                                                       SOL 6200
C     ******************************************************************SOL 6250
C                                                                       SOL 6300
C                                                                       SOL 6350
C                                                                       SOL 6400
C                                                                       SOL 6450
```

```
      SUBROUTINE COFSYM(LLL,MD,NP,XC,YC,ZC,S,ANVX,ANVY,ANVZ,A)          CSY   50
C     ******************************************************************CSY  100
C     ******************************************************************CSY  150
C     SETS UP (NP/2) LINEAR EQUATIONS FOR SOLVING JUST THE LEFT SIDE    CSY  200
C     BODY PANEL SOURCE STRENGTHS. -- USED ONLY WHEN SYMMETRIC BODY     CSY  250
C     INPUT OPTION WAS SPECIFIED--. THE LEFT COEFFICIENTS AND RIGHT     CSY  300
C     CONSTANTS OF EACH EQUATION ARE AS DESCRIBED IN SUBROUTINE COEFIC. CSY  350
C     THIS SUBROUTINE RETURNS TO MAIN PROGRAM WITH THE (NP/2) EQUATIONS CSY  400
C     STORED SEQUENTIALLY ONE PER LOGICAL RECORD IN THE MAGNETIC TAPE   CSY  450
C     OR DISK SCRATCH FILE 9.                                           CSY  500
C     ******************************************************************CSY  550
C     ******************************************************************CSY  600
C                                                                       CSY  650
C                                                                       CSY  700
      DIMENSION A(MD),S(LLL),ANVX(LLL),ANVY(LLL),ANVZ(LLL),XC(LLL),     CSY  750
     1YC(LLL),ZC(LLL)                                                   CSY  800
      COMMON /INPUTS/ ALPHA(6),BETA(6),VX(6),VY(6),VZ(6),V,NALPHA       CSY  850
      COMMON /INLET/ FLRATO(500),INDEX(500),NINFLO                      CSY  900
      COMMON /SYMTRE/ NPNSYM(2596),INSOLV(1298),NSYMET                  CSY  950
      COMMON /WING1/ CL(6),CHORD,DIHED,SPAN,SWEEP,XQR,YQR,ZQR,NWING     CSY 1000
      REWIND 9                                                          CSY 1050
C                                                                       CSY 1100
C     ASSIGN PANEL INDEX-EQUATION MEMORY SYSTEM. SEQUENTIALLY SEARCH ALLCSY 1150
C     PANEL INDEXES USE ONLY THE PANELS ON LEFT SIDE OF CONFIGURATION   CSY 1200
C     ( THOSE FOR WHICH NPNSYM(I) IS NOT ZERO). THE MEMORY ARRAY IS     CSY 1250
C     INSOLV(L) WHERE THE L TH NUMBER IN ARRAY INSOLV HAS THE VALUE OF  CSY 1300
C     THE LEFT PANEL INDEX NUMBER K. PANEL K IS THE ONE FOR WHICH THE   CSY 1350
C     L TH EQUATION OF THE CONDENSED SYSTEM OF EQUATIONS IS WRITTEN.    CSY 1400
C     NOTE THIS CONDENSED SYSTEM NEEDS ONLY NP/2 EQUATIONS FOR THE NP/2 CSY 1450
C     UNKNOWNS ON THE LEFT SIDE OF THE BODY SINCE THE RIGHT SIDE IS     CSY 1500
C     SYMMETRIC WITH THE LEFT SIDE.                                     CSY 1550
C                                                                       CSY 1600
      KOUNT=0                                                           CSY 1650
      DO 1 K=1,NP                                                       CSY 1700
      IF(NPNSYM(K).EQ.0) GO TO 1                                        CSY 1750
      KOUNT=KOUNT+1                                                     CSY 1800
```

```
      INSOLV(KOUNT)=K
    1 CONTINUE
C
C     START GETTING COEFFICIENTS OF I TH EQUATION FOR PANEL INSOLV(I).
C
      NPHALF=NP/2
      DO 10 I=1,NPHALF
      DO 3 J=1,NPHALF
      RX=XC(INSOLV(I))-XC(INSOLV(J))
      RY=YC(INSOLV(I))-YC(INSOLV(J))
      RZ=ZC(INSOLV(I))-ZC(INSOLV(J))
      RYIMAG=YC(INSOLV(I))+YC(INSOLV(J))
      R=SQRT(RX **2 +RY**2 + RZ**2)
      RIMAG=SQRT(RX**2 + RYIMAG**2 + RZ**2)
C
C     COMPUTE LEFT SIDE J TH COEFFICIENT ON I TH ROW.
C
      IF(I.EQ.J) GO TO 2
      A(J)=(S(INSOLV(J))/(2.*3.141593))*(((1./(R**3))*((RX*ANVX(INSOLV(I
     1)))+(RY*ANVY(INSOLV(I)))+(RZ*ANVZ(INSOLV(I))))) + ((1./(RIMAG**3))
     2*((RX*ANVX(INSOLV(I)))+(RYIMAG*ANVY(INSOLV(I)))+(RZ*ANVZ(INSOLV(I)
     3)))))
      GO TO 3
    2 A(J)=1.0 +(((S(INSOLV(I))/(8.*3.141593))*(YC(INSOLV(I))/
     +((ABS(YC(INSOLV(I))))**3))) *ANVY(INSOLV(I)) )
    3 CONTINUE
C
C     CALCULATE THE NALPHA NUMBER OF CONSTANTS ON THE RIGHT SIDE OF
C     EQUATION I. THIS IS FOR PANEL INSOLV(I) ON THE LEFT SIDE OF BODY.
C     CHECK IF PANEL INSOLV(I) IS A THROUGH FLOW PANEL. IF IT IS, ADD
C     THE PRESCRIBED THROUGH FLOW VELOCITY TO THE CONSTANTS OF THIS
C     EQUATION IN ORDER TO RELAX THE BOUNDARY CONDITION. SET INFADD = 0
C     IF PANEL INSOLV(I) IS NOT A THROUGH FLOW PANEL. SET INFADD = 1 IF
C     PANEL IS A THROUGH FLOW PANEL.  IF WING IS PRESENT, ADD TO THE
C     FREE STREAM VELOCITY, THE COMPONENTS OF VELOCITY AT THE CONTROL
C     POINT OF PANEL INSOLV(I) DUE TO THE WING.
```

```
C
      IF(NINFLO.EQ.0) GO TO 5
      DO 4 L=1,NINFLO
      IF(INDEX(L).NE.INSOLV(I)) GO TO 4
      LIN=L
      INFADD=1
      GO TO 6
    4 CONTINUE
    5 INFADD=0
    6 CONTINUE
      NUP1=NPHALF+1
      NUP=NPHALF+NALPHA
      DO 9 J=NUP1,NUP
      K=J-NPHALF
      IF(NWING.EQ.0) GO TO 7
      CALL WINGV(K,XC(INSOLV(I)),YC(INSOLV(I)),ZC(INSOLV(I)),VELX,VELY,
     +VELZ)
      A(J)=-2 0*((ANVX(INSOLV(I))*(VX(K)+VELX))+(ANVY(INSOLV(I))*(VY(K)+
     1VELY))+(ANVZ(INSOLV(I))*(VZ(K)+VELZ)))
      GO TO 8
    7 A(J)=-2.0*((ANVX(INSOLV(I))*VX(K))+(ANVY(INSOLV(I))*VY(K))+
     1(ANVZ(INSOLV(I))*VZ(K)))
    8 CONTINUE
C
C     IF INFADD=1, RELAX BOUNDARY CONDITION ON PANEL INSOLV(I):
      IF(INFADD.EQ.0) GO TO 9
      A(J)=A(J)+(-2.0*FLRATO(LIN)*V)
    9 CONTINUE
C
C     PUT I TH ROW OF SYSTEM CN I TH RECORD OF SEQUENTIAL FILE 9.
C
      WRITE(9)(A(M),M=1,NUP)
   10 CONTINUE
      ENDFILE 9
      REWIND 9
      RETURN
```

```
      END                                                               CSY 5450
C                                                                       CSY 5500
C                                                                       CSY 5550
C     ******************************************************************CSY 5600
C                                                                       CSY 5650
C                                                                       CSY 5700
```

```
      SUBROUTINE SOLSYM(LLL,LLLHAF,MD,MAXALP,NP,A,BIG,VARIAB,SUM,ERR,   SOS   50
     1ITMAX,SIGMA,SIGSAV)                                               SOS  100
C                                                                       SOS  150
C     ******************************************************************SOS  200
C     ******************************************************************SOS  250
C     USES GAUSS-SIEDEL ITERATION TO SOLVE THE LEFT HALF BODY CONDENSED SOS  300
C     SYSTEM OF EQUATIONS FOR NALPHA SETS OF (NP/2) UNKNOWNS, SIGMA     SOS  350
C     (BODY PANEL SOURCE STRENGTHS). -- THIS SUBROUTINE IS EXECUTED     SOS  400
C     ONLY WHEN SYMMETRIC BODY INPUT OPTION HAS BEEN SPECIFIED--.A SET  SOS  450
C     OF SIGMAS IS FOUND FOR EACH DIFFERENT BODY (ALPHA-BETA)           SOS  500
C     ORIENTATION CASE. ALL INPUT ALPHA-BETA CASES ARE SOLVED AT ONCE   SOS  550
C     USING A SINGLE EXECUTION OF THIS SUBROUTINE. ALL CASES MUST       SOS  600
C     CONVERGE OR ENTIRE JOB ABORTS. EQUATIONS OF SYSTEM ARE STORED     SOS  650
C     SEQUENTIALLY IN A DISK OR TAPE SCRATCH FILE 9. ON ONE ITERATION   SOS  700
C     EACH EQUATION IS SOLVED ONE AT A TIME IN ORDER FROM FIRST TO LAST SOS  750
C     ONE. DURING AN ITERATION THE CURRENT EQUATION EXAMINED IS READ    SOS  800
C     FROM ITS RECORD OF THE SCRATCH FILE AND STORED TEMPORARILY IN ROW SOS  850
C     MATRIX,A, FOR MANIPULATIONS. ONLY SIGMAS FOR THE NP/2 PANELS ON   SOS  900
C     THE LEFT SIDE OF BODY ARE SOLVED. THE SIGMAS FOR THE RIGHT SIDE   SOS  950
C     MIRROR IMAGE PANELS ARE THEN KNOWN BY SYMMETRY.                   SOS 1000
C     ******************************************************************SOS 1050
C     ******************************************************************SOS 1100
C                                                                       SOS 1150
      DIMENSION A(MD) SIGMA(LLL,MAXALP)                                 SOS 1200
      DIMENSION BIG(MAXALP),VARIAB(MAXALP),SUM(MAXALP)                  SOS 1250
      DIMENSION SIGSAV(LLLHAF)                                          SOS 1300
      COMMON /INPUTS/ ALPHA(6),BETA(6),VX(6),VY(6),VZ(6),V,NALPHA       SOS 1350
      COMMON /SYMTRE/ NPNSYM(2596) INSOLV(1298),NSYMET                  SOS 1400
      NPHALF=NP/2                                                       SOS 1450
      NUP=NPHALF+NALPHA                                                 SOS 1500
      ERROR=10.0**ERR                                                   SOS 1550
      WRITE(6,100)                                                      SOS 1600
      WRITE(6,200)ITMAX                                                 SOS 1650
      WRITE(6,300) ERROR                                                SOS 1700
C                                                                       SOS 1750
C     ESTABLISH INITIAL GUESS FOR UNKNOWNS:                             SOS 1800
```

```
C                                                                   SOS 1850
      DO 1 I=1,NPHALF                                               SOS 1900
      DO 1 J=1,NALPHA                                               SOS 1950
      SIGMA(I,J)=0.0                                                SOS 2000
    1 CONTINUE                                                      SOS 2050
C                                                                   SOS 2100
C     PERFORM ITERATIONS                                            SOS 2150
C                                                                   SOS 2200
      DO 10 ITER=1,ITMAX                                            SOS 2250
      REWIND 9                                                      SOS 2300
      DO 2 K=1,NALPHA                                               SOS 2350
      BIG(K)=0.0                                                    SOS 2400
    2 CONTINUE                                                      SOS 2450
      WRITE(6,400)ITER                                              SOS 2500
      WRITE(6,500)                                                  SOS 2550
C                                                                   SOS 2600
C     WORK ON I TH EQUATION OF SYSTEM                               SOS 2650
C                                                                   SOS 2700
      DO 7 I=1,NPHALF                                               SOS 2750
      DO 3 K=1,NALPHA                                               SOS 2800
      SUM(K)=0.0                                                    SOS 2850
    3 CONTINUE                                                      SOS 2900
      READ(9)(A(K),K=1,NUP)                                         SOS 2950
C                                                                   SOS 3000
C     GET SUM OF ROW I COEFFICIENT-UNKNOWN PRODUCTS FOR EACH ALPHA SET. SOS 3050
C                                                                   SOS 3100
      DO 5 J=1,NPHALF                                               SOS 3150
      IF(J.EQ.I) GO TO 5                                            SOS 3200
      DO 4 K=1,NALPHA                                               SOS 3250
      SUM(K)=SUM(K)+(A(J)*SIGMA(J,K))                               SOS 3300
    4 CONTINUE                                                      SOS 3350
    5 CONTINUE                                                      SOS 3400
C                                                                   SOS 3450
C     SOLVE I TH EQUATION FOR I TH UNKNOWN (OF EACH K TH ALPHA CASE) -  SOS 3500
C     THIS ITERATION. CALL IT VARIAB(K).                            SOS 3550
C                                                                   SOS 3600
```

```
      DO 6 K=1,NALPHA                                                   SOS 3650
      M=NPHALF+K                                                        SOS 3700
      VARIAB(K)=(A(M)-SUM(K))/A(I)                                      SOS 3750
      IF(ABS(VARIAB(K)-SIGMA(I,K)).GT.BIG(K)) BIG(K)=ABS(VARIAB(K)-     SOS 3800
     1SIGMA(I,K))                                                       SOS 3850
      SIGMA(I,K)=VARIAB(K)                                              SOS 3900
    6 CONTINUE                                                          SOS 3950
    7 CONTINUE                                                          SOS 4000
C                                                                       SOS 4050
C     ALL EQUATIONS DONE THIS ITERATION CHECK FOR CONVERGENCE.          SOS 4100
C                                                                       SOS 4150
      NSIGNL=0                                                          SOS 4200
      DO 9 K=1,NALPHA                                                   SOS 4250
      IF(BIG(K).LT.ERROR) GO TO 8                                       SOS 4300
      NSIGNL=1                                                          SOS 4350
      WRITE(6,600)ALPHA(K),BETA(K),K,BIG(K)                             SOS 4400
      GO TO 9                                                           SOS 4450
    8 WRITE(6,601)ALPHA(K),BETA(K),K,BIG(K)                             SOS 4500
    9 CONTINUE                                                          SOS 4550
C                                                                       SOS 4600
C     CHECK NSIGNL: IF NSIGNL=0 ALL ALPHA CASES HAVE CONVERGED TO GOOD  SOS 4650
C     SOLUTIONS. IF NSIGNL=1(NOT =0) THEN ONE OR MORE CASES HAVE NOT YETSOS 4700
C     CONVERGED FULLY - GO BACK AND ITERATE ALL ALPHA CASES AGAIN.      SOS 4750
C                                                                       SOS 4800
      IF(NSIGNL.NE.0) GO TO 10                                          SOS 4850
      WRITE(6,700) ITER                                                 SOS 4900
      REWIND 9                                                          SOS 4950
C                                                                       SOS 5000
C     ALL SOLUTIONS HAVE CONVERGED.                                     SOS 5050
C                                                                       SOS 5100
      GO TO 11                                                          SOS 5150
   10 CONTINUE                                                          SOS 5200
C                                                                       SOS 5250
C     IF 10 DO LOOP IS COMPLETELY EXECUTED, THEN THE MAXIMUM ALLOWED    SOS 5300
C     NUMBER OF ITERATIONS WAS PERFORMED AND ONE OR MORE CASES FAILED TOSOS 5350
C     CONVERGE TO ACCURATE SOLUTIONS.--ABORT RUN.                       SOS 5400
```

```
C
      WRITE(6,800)
      REWIND 9
      STOP
C
   11 CONTINUE
C
C     CONDENSED EQUATION SYSTEM WAS SOLVED FOR SYMMETRIC CASE. FINALLY
C     MUST REARRANGE SOLUTION SIGMAS IN THE ARRAY SUCH THAT SIGMA(I,K)
C     IS THE VALUE FOR THE I TH PANEL IN THE K TH ALPHA CASE. ALSO MUST
C     ASSIGN PROPER SIGMA VALUE TO THE PROPER MIRROR IMAGE PANEL ON THE
C     RIGHT SIDE OF THE BODY.
C
      DO 14 K=1,NALPHA
C
C     STORE DUPLICATE RESULTS IN THE ARRAY SIGSAV- FOR THIS NALPHA CASE.
C
      DO 12 L=1,NPHALF
      SIGSAV(L)=SIGMA(L,K)
   12 CONTINUE
C
C     NOW IN SEQUENCE, PUT I TH SOLUTION FROM SAVED ARRAY INTO SIGMA
C     WHOSE PANEL INDEX WAS USED IN I TH SOLUTION. FIRST ASSIGN LEFT
C     SIDE PANEL'S VALUE FOLLOWED BY SAME VALUE FOR THE CORRESPONDING
C     RIGHT SIDE IMAGE PANEL.
C
      DO 13 I=1,NPHALF
      SIGMA(INSOLV(I),K)=SIGSAV(I)
      SIGMA(NPNSYM(INSOLV(I)),K)=SIGSAV(I)
   13 CONTINUE
   14 CONTINUE
C
C     NOW ALL NP SIGMAS HAVE BEEN REARRANGED SO THAT I TH SIGMA VALUE
C     IS THAT FOR THE I TH PANEL OF THE CONFIGURATION.
C
      RETURN
```

```
C                                                                         SOS 7250
C-------------------------------------------------------------------------SOS 7300
C     FORMATS                                                             SOS 7350
C                                                                         SOS 7400
  100 FORMAT('1' T42,'**** SOLUTION OF SOURCE STRENGTHS, SIGMA ****'/)    SOS 7450
  200 FORMAT(' ',T35,'SPECIFIED MAXIMUM PERMITTED SOLUTION ITERATIONS=',SOS 7500
     1I10)                                                                SOS 7550
  300 FORMAT(' ',T35,'SPECIFIED MAXIMUM PERMITTED SOLUTION ERROR     =',SOS 7600
     11PE10.0/)                                                           SOS 7650
  400 FORMAT('0',T55,'/// ITERATION ',I3,' ///'/)                         SOS 7700
  500 FORMAT(' ',T38,'ALPHA, BETA,-CASE',T75,'SOLUTION ERROR'/)           SOS 7750
  600 FORMAT(' ',T31,F13.7,', ',F13.7,', ',I2,T76,1PE12.5)                SOS 7800
  601 FORMAT(' ',T31,F13.7,', ',F13.7,', ',I2,T76,1PE12.5,T90,'(CASE HASSOS 7850
     1 CONVERGED)')                                                       SOS 7900
  700 FORMAT('0',T26,'--- SOURCE STRENGTH SOLUTIONS CONVERGED FOR ALL CASOS 7950
     1SES WITHIN',I5,' ITERATIONS ---'/)                                  SOS 8000
  800 FORMAT('0',T27,'**** FAILURE OF SOLUTION CONVERGENCE FOR ONE OR MOSOS 8050
     1RE CASES - RUN ABORTED ****'/)                                      SOS 8100
C                                                                         SOS 8150
      END                                                                 SOS 8200
C                                                                         SOS 8250
C     ********************************************************************SOS 8300
C                                                                         SOS 8350
C                                                                         SOS 8400
C                                                                         SOS 8450
C                                                                         SOS 8500
```

```
      SUBROUTINE VELOCI(LLL,MD,MAXALP,ICASE,NP,ANVX,ANVY,ANVZ,S,SIGMA,   VEL
     1XC,YC,ZC)                                                          VEL
C                                                                        VEL
C     *******************************************************************VEL
C     *******************************************************************VEL
C     COMPUTES BODY PANEL SURFACE VELOCITIES AND PRESSURE COEFFICIENT,   VEL
C     CP, AT ALL NP PANEL CONTROL POINTS. DURING ONE CALL OF THIS        VEL
C     SUBROUTINE, THESE SURFACE VELOCITIES AND PRESSURES ARE BEING       VEL
C     FOUND FOR ONE BODY ANGLE ( ALPHA-BETA ) ORIENTATION CASE, ICASE.   VEL
C     FOR THIS ICASE, START WITH PANEL I=1. FIND VELOCITY COMPONENTS,    VEL
C     VRX, VRY, VRZ, AND MAGNITUDE VR. BASED ON VR, CALCULATE THE        VEL
C     PRESSURE COEFFICIENT, CP, AT THE CONTROL POINT. PRINT RESULTS.     VEL
C     GO BACK AND REPEAT FOR I= 2,3,...IN TERN TO ... LAST NP TH PANEL.  VEL
C     THE VELOCITY COMPONENTS AND RESULTANT ARE THEN NORMALIZED BY       VEL
C     DIVIDING BY THE FREESTREAM VELOCITY MAGNITUDE, V. FINALLY, VRX,    VEL
C     VRY,VRZ,AND VR EXPRESS NORMALIZED QUANTITIES.                      VEL
C     *******************************************************************VEL
C     *******************************************************************VEL
C                                                                        VEL
      DIMENSION ANVX(LLL),ANVY(LLL),ANVZ(LLL),S(LLL),SIGMA(LLL MAXALP)   VEL
      DIMENSION XC(LLL),YC(LLL),ZC(LLL)                                  VEL
      COMMON /INPUTS/ ALPHA(6),BETA(6),VX(6),VY(6),VZ(6),V,NALPHA        VEL
      COMMON /INLET/ FLRATO(500),INDEX(500),NINFLO                       VEL
      COMMON /WING1/ CL(6),CHORD,DIHED,SPAN,SWEEP,XQR,YQR,ZQR,NWING      VEL
      INTEGER STARS(11),STAR/'****'/                                     VEL
      DO 1 K=1,11                                                        VEL
      STARS(K)=STAR                                                      VEL
    1 CONTINUE                                                           VEL
      WRITE(6,100)(STARS(K),K=1,11)                                      VEL
      WRITE(6,200)                                                       VEL
      WRITE(6,300) ICASE,ALPHA(ICASE)                                    VEL
      WRITE(6,400) BETA(ICASE)                                           VEL
      WRITE(6,101)(STARS(K),K=1,11)                                      VEL
      WRITE(6,500) V                                                     VEL
      WRITE(6,600)VX(ICASE)                                              VEL
      WRITE(6,601)VY(ICASE)                                              VEL
```

```
      WRITE(6,602)VZ(ICASE)
      WRITE(6,700) NP
      WRITE(6,800)
      WRITE(6,801)
      WRITE(6,802)
      DO 6 I=1,NP
      CX=0.0
      CY=0.0
      CZ=0.0
      SUMX=0.0
      SUMY=0.0
      SUMZ=0.0
C
C     COMPUTE COMPONENTS OF VELOCITY CX,CY,AND CZ AT I TH CONTROL POINT
C     DUE TO A POINT SOURCE AT CONTROL POINT J. SUM CONTRIBUTIONS DUE
C     TO ALL POINTS J ON I USING SUMX,SUMY,AND SUMZ. SUMX,SUMY,AND SUMZ
C     FINALLY CONTAIN THE SUM OF VELOCITY COMPONENTS AT CONTROL POINT I
C     DUE TO I AND DUE TO ALL OTHER J CONTROL POINTS ON THE BODY.
C
      DO 4 J=1,NP
      RX=XC(I)-XC(J)
      RY=YC(I)-YC(J)
      RZ=ZC(I)-ZC(J)
      R=SQRT(RX**2 + RY**2 + RZ**2)
C
C     DISTRIBUTED SOURCE STRENGTH SIGMA ON THE J TH PANEL ( FOR THE
C     ALPHA-BETA CASE,ICASE ) IS IN ARRAY ELEMENT SIGMA(J,ICASE).
C
      IF(J.EQ.I) GO TO 2
      CX=((S(J)/(4.*3.141593))*(RX/(R**3.)))*SIGMA(J,ICASE)
      CY=((S(J)/(4.*3.141593))*(RY/(R**3.)))*SIGMA(J,ICASE)
      CZ=((S(J)/(4.*3.141593))*(RZ/(R**3.)))*SIGMA(J,ICASE)
      GO TO 3
    2 CX=(ANVX(I)/2.)*SIGMA(J,ICASE)
      CY=(ANVY(I)/2.)*SIGMA(J,ICASE)
      CZ=(ANVZ(I)/2.)*SIGMA(J,ICASE)
```

```
    3 SUMX=SUMX + CX
      SUMY=SUMY + CY
      SUMZ=SUMZ + CZ
    4 CONTINUE
C
C     FOR THIS ORIENTATION, ICASE, GET THE NET VELOCITY COMPONENTS, VRX,
C     VRY, AND VRZ, AT CONTROL POINT I BY ADDING THE FREE STREAM
C     COMPONENT VELOCITIES TO THE SOURCE CAUSED COMPONENTS, SUMX, SUMY,
C     AND SUMZ. ALSO IF WING IS PRESENT ADD IN THE VELOCITY COMPONENTS
C     INDUCED AT POINT I BY THE WING VORTEX.
C
      VRX=SUMX + VX(ICASE)
      VRY=SUMY + VY(ICASE)
      VRZ=SUMZ + VZ(ICASE)
      IF(NWING.EQ.0) GO TO 5
      CALL WINGV(ICASE,XC(I),YC(I),ZC(I),VELX,VELY,VELZ)
      VRX=VRX + VELX
      VRY=VRY + VELY
      VRZ=VRZ + VELZ
    5 CONTINUE
C
C     NORMALIZE NET VELOCITY COMPONENTS BY FREE STREAM MAGNITUDE, AND
C     GET NET RESULTANT AND PRESSURE COEFFICIENT.
C
      VRX=VRX/V
      VRY=VRY/V
      VRZ=VRZ/V
      VR=SQRT(VRX**2 + VRY**2 + VRZ**2)
      CP=1.0 - (VR**2)
      WRITE(6,900)I,XC(I),YC(I),ZC(I),SIGMA(I,ICASE),VRX,VRY,VRZ,VR,CP
    6 CONTINUE
      WRITE(6,1000)(STARS(K),STARS(K),STARS(K),K=1,11)
      RETURN
C
C-----------------------------------------------------------------
C
```

```
C       FORMATS                                                         VEL 5450
C                                                                       VEL 5500
  100 FORMAT('1',T44,11A4)                                              VEL 5550
  101 FORMAT(' ',T44,11A4)                                              VEL 5600
  200 FORMAT(' ',T44,'*  BODY SURFACE VELOCITIES AND PRESSURES   *')    VEL 5650
  300 FORMAT(' ',T44,'*',T48,'CASE ',I2,' - ALPHA = ',F9.3,' DEGREES',  VEL 5700
     1T87,'*')                                                          VEL 5750
  400 FORMAT(' ',T44,'*',T58,'BETA  = ',F9.3,' DEGREES',T87,'*')        VEL 5800
  500 FORMAT(/,'0',T40,'MAGNITUDE OF FREESTREAM VELOCITY   V = ',F13.7) VEL 5850
  600 FORMAT(' ',T40,'COMPONENTS OF FREESTREAM VELOCITY  VX= ',F13.7)   VEL 5900
  601 FORMAT(' ',T75,'VY= ',F13.7)                                      VEL 5950
  602 FORMAT(' ',T75,'VZ= ',F13.7)                                      VEL 6000
  700 FORMAT('0',T42,'TOTAL NUMBER OF PANEL ELEMENTS = ',I4/)           VEL 6050
  800 FORMAT(/,'0',T73,'NORMALIZED COMPONENTS AND',T114,'PRESSURE')     VEL 6100
  801 FORMAT(' ',T13,'CONTROL POINT COORDINATES',T46,'SOURCE STRENGTH',TVEL 6150
     169,'RESULTANT VELOCITY AT CONTROL POINT',T114,'COEFFICIENT')      VEL 6200
  802 FORMAT(' ',T2,'PANEL',T13,'XC',T26,'YC',T37,'ZC',T52,'SIGMA',T67,'VEL 6250
     1VRX/V',T80,'VRY/V',T91,'VRZ/V',T104,'VR/V',T117,'CP'/)            VEL 6300
  900 FORMAT(' ',T2,I5,T9,F10.4,T21,F10.4,T32,F10.4,T46,E15.8,T64,F10.4,VEL 6350
     1T76,F10.4,T88,F10.4,T100,F10.4,T113,F8.4)                         VEL 6400
 1000 FORMAT('0',33A4///)                                               VEL 6450
C                                                                       VEL 6500
      END                                                               VEL 6550
C                                                                       VEL 6600
C     ******************************************************************VEL 6650
C                                                                       VEL 6700
C                                                                       VEL 6750
C                                                                       VEL 6800
```

```
C     SUBROUTINE VPROPS(LLL,MD,MAXALP,ICASE,NP,ANVX,ANVY,ANVZ,S,SIGMA,  VPR   50
C    1XC,YC,ZC)                                                          VPR  100
C                                                                        VPR  150
C     *******************************************************************VPR  200
C     *******************************************************************VPR  250
C     PROPELLER PLANE VELOCITY CALCULATIONS FOR THE BODY ORIENTATION     VPR  300
C     CASE, ICASE. IT LOCATES POINTS IN THE PROPELLER PLANE DEFINED BY   VPR  350
C     INPUT DATA. AN IMAGINARY REFERENCE PROPELLER BLADE OF RADIUS,      VPR  400
C     RADIUS, ATTACHED TO THE HUB AND LYING IN THE PROPELLER PLANE IS    VPR  450
C     USED FOR GENERATING POINTS. POINTS ARE FOUND AS FOLLOWS: START AT  VPR  500
C     BLADE ZERO AZIMUTH POSITION AND DEFINE POINTS RADIALLY OUTWARD AT  VPR  550
C     THE AZIMUTH. NEXT MOVE TO NEXT CONSECUTIVE AZIMUTH POSITION AND    VPR  600
C     REPEAT ETC....UNTIL LAST AZIMUTH POSITION IS DONE.                 VPR  650
C                                                                        VPR  700
C     AS EACH POINT IS LOCATED, THE VELOCITY COMPONENTS THERE DUE TO     VPR  750
C     BODY SOURCES ARE COMPUTED AND ADDED TO FREE STREAM COMPONENTS.     VPR  800
C     ALSO, IF A WING IS MODELED, THE WING INDUCED VELOCITY AT THE       VPR  850
C     PROPELLER POINT IS ADDED. AT THE POINT, THE X,Y,Z COMPONENTS AND   VPR  900
C     RESULTANT MAGNITUDES ARE NONDIMENSIONALIZED BY DIVIDING BY THE     VPR  950
C     FREE STREAM MAGNITUDE, V, AND THESE NORMALIZED VALUES ARE PRINTED  VPR 1000
C     ON PART 1 OF THE OUTPUT TABLE.                                     VPR 1050
C                                                                        VPR 1100
C     THEN THE VELOCITIES ARE COMPUTED IN TERMS OF AXIAL, RADIAL, AND    VPR 1150
C     TANGENTIAL NORMALIZED COMPONENTS, ANGLES OF ROTATIONAL FLOW,       VPR 1200
C     ANGLES OF OUTFLOW, UPWASH AND SIDEWASH ANGLES. THESE VALUES ARE    VPR 1250
C     PRINTED IN OUTPUT TABLE PART 2.                                    VPR 1300
C                                                                        VPR 1350
C     OUTPUT IS PUNCHED ON CARDS IF PRINT OPTION IS CHOSEN. FIRST, TWO   VPR 1400
C     HEADING CARDS ARE PUNCHED. THEN A CARD WITH NUMBER OF AZIMUTH      VPR 1450
C     LOCATIONS, NUMBER OF RADIAL POSITIONS AT AN AZIMUTH, REFERENCE     VPR 1500
C     PROP RADIUS, TOTAL THRUST AXIS ANGLE OF ATTACK AND SIDESLIP, AND   VPR 1550
C     BODY ANGLE OF ATTACK AND SIDESLIP IS NEXT. LAST ARE SEVERAL GROUPSVPR 1600
C     OF CARDS. EACH GROUP HAS ITS AZIMUTH PUNCHED. AN  AZIMUTH GROUP    VPR 1650
C     HAS SEVERAL CARDS EACH WITH ITS RADIAL POSITION PUNCHED, THEN ITS  VPR 1700
C     NORMALIZED AXIAL AND TANGENTIAL VELOCITY COMPONENTS.               VPR 1750
C     *******************************************************************VPR 1800
```

```
C     ******************************************************************VPR 1850
C                                                                        VPR 1900
      SUBROUTINE VPROPS(LLL,MD,MAXALP,ICASE,NP,ANVX,ANVY,ANVZ,S,SIGMA,   VPR 1950
     1XC,YC,ZC)                                                          VPR 2000
      DIMENSION S(LLL),SIGMA(LLL,MAXALP),XC(LLL),YC(LLL),ZC(LLL)         VPR 2050
      DIMENSION ANVX(LLL),ANVY(LLL),ANVZ(LLL)                            VPR 2100
      COMMON /TITLE/ SYMBOL(160)                                         VPR 2150
      COMMON /INPUTS/ ALPHA(6),BETA(6),VX(6),VY(6),VZ(6),V,NALPHA        VPR 2200
      COMMON /PROP/ ALPHP,BETAP,XHUB,YHUB,ZHUB,DPSI,DRADUS,RADIUS,NRAD,  VPR 2250
     +NPOINT                                                             VPR 2300
      COMMON /OPTION/ NCALC,NLIST,NPUNCH                                 VPR 2350
      COMMON /WING1/ CL(6),CHORD,DIHED SPAN,SWEEP,XQR,YQR,ZQR,NWING      VPR 2400
      COMMON /INLET/ FLRATO(500) INDEX(500),NINFLO                       VPR 2450
      INTEGER STARS(11),STAR/'****'/                                     VPR 2500
C                                                                        VPR 2550
      DO 1 K=1,11                                                        VPR 2600
      STARS(K)=STAR                                                      VPR 2650
    1 CONTINUE                                                           VPR 2700
      IF(NPOINT.NE.0) GO TO 2                                            VPR 2750
      WRITE(6,100)                                                       VPR 2800
      WRITE(6,904)(STARS(K),STARS(K) STARS(K),K=1,11)                    VPR 2850
      RETURN                                                             VPR 2900
    2 REWIND 50                                                          VPR 2950
      WRITE(6,200) STAR,STAR,STAR,(STARS(K),K=1,10)                      VPR 3000
      WRITE(6,250)                                                       VPR 3050
      WRITE(6,300)                                                       VPR 3100
      WRITE(6,400)ICASE,ALPHA(ICASE)                                     VPR 3150
      WRITE(6,500)BETA(ICASE)                                            VPR 3200
      IF(NWING.EQ.1) GO TO 3                                             VPR 3250
      WRITE(6,501)                                                       VPR 3300
      GO TO 4                                                            VPR 3350
    3 WRITE(6,502) CL(ICASE)                                             VPR 3400
    4 WRITE(6,201) STAR,STAR,STAR,(STARS(K),K=1,10)                      VPR 3450
      WRITE(6,600)V                                                      VPR 3500
      WRITE(6,700)VX(ICASE)                                              VPR 3550
      WRITE(6,701)VY(ICASE)                                              VPR 3600
```

```
      WRITE(6,702)VZ(ICASE)                                             VPR 3650
      IF(NINFLO.EQ.0) GO TO 5                                           VPR 3700
      WRITE(6,703) FLRATO(1)                                            VPR 3750
      GO TO 6                                                           VPR 3800
    5 WRITE(6,704)                                                      VPR 3850
    6 WRITE(6,800)                                                      VPR 3900
      WRITE(6,801)                                                      VPR 3950
      WRITE(6,802)XHUB,YHUB,ZHUB                                        VPR 4000
      WRITE(6,803)RADIUS                                                VPR 4050
      WRITE(6,804)ALPHP                                                 VPR 4100
      WRITE(6,805)BETAP                                                 VPR 4150
      WRITE(6,806)                                                      VPR 4200
C                                                                       VPR 4250
C     FOR GIVEN INPUT PROPELLER PLANE ORIENTATION (ALPHP - BETAP),      VPR 4300
C     RELATIVE TO FUSELAGE AXES  COMPUTE EULER ANGLE ANADA WHICH        VPR 4350
C     CORRESPONDS TO THE (ALPHP-BETAP) PAIR.                            VPR 4400
C                                                                       VPR 4450
      CALL EULER(ALPHP,BETAP,ANADA)                                     VPR 4500
C                                                                       VPR 4550
C     PUT ANADA,BETAP, AND ALPHP IN TERMS OF RADIANS.                   VPR 4600
C                                                                       VPR 4650
      ANADAR=ANADA*3.141593/180.                                        VPR 4700
      BETRDN=BETAP*3.141593/180.                                        VPR 4750
      ALPRDN=ALPHP*3.141593/180.                                        VPR 4800
C                                                                       VPR 4850
C     CHECK DPSI INPUT CHOICE TO SEE IF IT IS AN INTEGER MULTIPLE OF    VPR 4900
C     360. THIS IS REQUIRED FOR EVEN AZIMUTH SPACING. OTHERWISE PRINT A VPR 4950
C     WARNING STATEMENT THAT AZIMUTH SPACING IS UNEQUAL FOR THIS RUN.   VPR 5000
C                                                                       VPR 5050
      PSINUM=360./DPSI                                                  VPR 5100
      IF(FLOAT(IFIX(PSINUM)).EQ.PSINUM) GO TO 7                         VPR 5150
      WRITE(6,1000) DPSI                                                VPR 5200
    7 NPSIS=IFIX(PSINUM)                                                VPR 5250
C                                                                       VPR 5300
C     PUNCH THE ICASE (BODY ALPHA-BETA) IDENTIFIER CARD IF PUNCHED CARD VPR 5350
C     OUTPUT OPTION WAS CHOSEN. ATOT IS ANGLE OF ATTACK OF PROP THRUST  VPR 5400
```

```
C     AXIS FROM FREE STREAM DIRECTION IN DEGREES POSITIVE UPWARD.       VPR 5450
C     BTOT IS SIDESLIP ANGLE OF THRUST AXIS FROM FREE STREAM DIRECTION  VPR 5500
C     IN DEGREES POSITIVE TOWARD RIGHT.                                 VPR 5550
C                                                                       VPR 5600
      IF(NPUNCH.EQ.0) GO TO 8                                           VPR 5650
      ATOT=ALPHA(ICASE)+ALPHP                                           VPR 5700
      BTOT=BETA(ICASE)+BETAP                                            VPR 5750
      WRITE(7,925)(SYMBOL(K),K=1,80)                                    VPR 5800
      WRITE(7,925)(SYMBOL(K),K=81,160)                                  VPR 5850
      WRITE(7,905)NPSIS,NRAD,RADIUS,ATOT,BTOT,ALPHA(ICASE),BETA(ICASE)  VPR 5900
    8 CONTINUE                                                          VPR 5950
      WRITE(6,900)                                                      VPR 6000
      WRITE(6,901)                                                      VPR 6050
C                                                                       VPR 6100
C     MOVE TO APPROPRIATE I TH AZIMUTH LOCATION                         VPR 6150
C                                                                       VPR 6200
      RDELTA=DRADUS*RADIUS                                              VPR 6250
      PSI=-DPSI                                                         VPR 6300
      DO 13 I=1,NPSIS                                                   VPR 6350
      PSI=PSI + DPSI                                                    VPR 6400
C     GET AZIMUTH ANGLE IN RADIANS:                                     VPR 6450
      PSIRDN=PSI*3.141593/180.                                          VPR 6500
C                                                                       VPR 6550
C     FOR THIS AZIMUTH, PSI, LOCATE POINTS STARTING AT HUB AND GOING    VPR 6600
C     OUTWARD RADIALLY TO THE J TH RADIAL STATION AT POSITION, RLOCAL.  VPR 6650
C                                                                       VPR 6700
      RLOCAL=-RDELTA                                                    VPR 6750
      DO 12 J=1,NRAD                                                    VPR 6800
      RLOCAL=RLOCAL + RDELTA                                            VPR 6850
      RRATIO=RLOCAL/RADIUS                                              VPR 6900
C                                                                       VPR 6950
C     FOR THIS POINT AT POSITION (RLOCAL,PSIRDN) IN TERMS OF COORDINATESVPR 7000
C     ATTACHED TO THE PROPELLER PLANE'S ROTATED AXIS SYSTEM, CONVERT    VPR 7050
C     FROM THAT CYLINDRICAL AXIS SYSTEM TO THE BODY FIXED CARTESIAN     VPR 7100
C     COORDINATE SYSTEM SO ITS CONVERTED POSITION IS POINTX, POINTY,    VPR 7150
C     POINTZ.                                                           VPR 7200
```

```
C                                                                       VPR 7250
      XX=-RLOCAL*((SIN(PSIRDN)*SIN(ANADAR)*COS(ALPRDN)) + (COS(PSIRDN)* VPR 7300
     1SIN(ALPRDN)))                                                     VPR 7350
      YY=RLOCAL*SIN(PSIRDN)*COS(ANADAR)                                 VPR 7400
      ZZ=RLOCAL*((SIN(PSIRDN)*SIN(ANADAR)*SIN(ALPRDN)) - (COS(PSIRDN)*  VPR 7450
     1COS(ALPRDN)))                                                     VPR 7500
      POINTX=XHUB + XX                                                  VPR 7550
      POINTY= YHUB + YY                                                 VPR 7600
      POINTZ=ZHUB + ZZ                                                  VPR 7650
C                                                                       VPR 7700
C     AT POINT LOCATION (RLOCAL,PSIRDN), OR (POINTX,POINTY,POINTZ),     VPR 7750
C     GET COMPONENTS AND RESULTANT OF THE NET TOTAL VELOCITY. THESE     VPR 7800
C     VELOCITIES ARE THE SUM OF THE FREE STREAM COMPONENT PLUS THE NET  VPR 7850
C     COMPONENTS OF VELOCITY AT THE POINT DUE TO ALL BODY PANEL SOURCES,VPR 7900
C     AND PLUS (IF WING IS PRESENT), THE VELOCITY COMPONENTS INDUCED AT VPR 7950
C     THE POINT BY THE WING VORTEX. THE SUM TOTAL VELOCITY COMPONENTS   VPR 8000
C     AT THE POINT ARE VTOTX, VTOTY, VTOTZ AND RESULTANT VTOT.          VPR 8050
C     NORMALIZE THESE VELOCITIES BY THE FREE STREAM VELOCITY MAGNITUDE  VPR 8100
C     BEFORE PRINTING AND/OR PUNCHING.                                  VPR 8150
C                                                                       VPR 8200
      DVX=0.0                                                           VPR 8250
      DVY=0.0                                                           VPR 8300
      DVZ=0.0                                                           VPR 8350
C                                                                       VPR 8400
C     SUM THE VELOCITY CONTRIBUTIONS OF ALL BODY SOURCE STRENGTHS.      VPR 8450
C                                                                       VPR 8500
      DO 10 K=1,NP                                                      VPR 8550
      RX=POINTX-XC(K)                                                   VPR 8600
      RY=POINTY-YC(K)                                                   VPR 8650
      RZ=POINTZ-ZC(K)                                                   VPR 8700
      R=SQRT(RX**2 + RY**2 + RZ**2)                                     VPR 8750
      IF(R.FQ.0.0) GO TO 9                                              VPR 8800
      DVX=DVX+((S(K)/(4.*3.141593))*SIGMA(K,ICASE)*((RX)/(R**3.)))      VPR 3850
      DVY=DVY+((S(K)/(4.*3.141593))*SIGMA(K,ICASE)*((RY)/(R**3.)))      VPR 8900
      DVZ=DVZ+((S(K)/(4.*3.141593))*SIGMA(K,ICASE)*((RZ)/(R**3.)))      VPR 8950
      GO TO 10                                                          VPR 9000
```

```
    9 DVX=DVX+((ANVX(K)/2.)*SIGMA(K,ICASE))
      DVY=DVY+((ANVY(K)/2.)*SIGMA(K,ICASE))
      DVZ=DVZ+((ANVZ(K)/2.)*SIGMA(K,ICASE))
   10 CONTINUE
C
C     NEXT ADD THE FREE STREAM VELOCITY COMPONENTS TO THE SUM AT POINT.
      VTOTX=DVX + VX(ICASE)
      VTOTY=DVY + VY(ICASE)
      VTOTZ=DVZ + VZ(ICASE)
C
C     LASTLY IF WING IS PRESENT ADD VELOCITY INDUCED BY THE WING ON THE
C     POINT. IF NO WING, THE SUM IS NOW COMPLETE.
C
      IF(NWING.EQ.0) GO TO 11
      CALL WINGV(ICASE,POINTX,POINTY,POINTZ,VELX,VELY,VELZ)
      VTOTX=VTOTX+VELX
      VTOTY=VTOTY+VELY
      VTOTZ=VTOTZ+VELZ
   11 VTOT=SQRT(VTOTX**2 + VTOTY**2 + VTOTZ**2)
C
C     NORMALIZE (NONDIMENSIONALIZE) THE VELOCITIES BY THE FREE STREAM
C     VELOCITY MAGNITUDE. PRINT RESULTS IN TABLE PART 1.
      VTOTX=VTOTX/V
      VTOTY=VTOTY/V
      VTOTZ=VTOTZ/V
      VTOT=VTOT/V
      WRITE(6,902)RRATIO,PSI,POINTX,POINTY,POINTZ,VTOTX,VTOTY,VTOTZ,VTOT
C
C     CALCULATE AT THIS PROPELLER PLANE POINT THE NONDIMENSIONALIZED
C     AXIAL, RADIAL, AND TANGENTIAL VELOCITY COMPONENTS. THEN COMPUTE
C     ANGLE OF ROTATIONAL FLOW, ANGLE OF OUTFLOW, UPWASH ANGLE AND
C     SIDEWASH ANGLE.
C
      VAXIAL=(COS(ALPRDN)*COS(ANADAR)*VTOTX)+(SIN(ANADAR)*VTOTY)-(SIN
     +(ALPRDN)*COS(ANADAR)*VTOTZ)
      VRADAL=(-SIN(ALPRDN)*COS(PSIRDN)-COS(ALPRDN)*SIN(ANADAR)*SIN(
```

```
     1PSIRDN))*VTOTX+(COS(ANADAR)*SIN(PSIRDN))*VTOTY+(SIN(ALPRDN)*SIN(   VPR10850
     2ANADAR)*SIN(PSIRDN)-COS(ALPRDN)*COS(PSIRDN))*VTOTZ                VPR10900
      VTANG=(SIN(ALPRDN)*SIN(PSIRDN)-COS(ALPRDN)*SIN(ANADAR)*COS(PSIRDN)VPR10950
     1)*VTOTX+(COS(ANADAR)*COS(PSIRDN))*VTOTY+(COS(ALPRDN)*SIN(PSIRDN)+ VPR11000
     2SIN(ALPRDN)*SIN(ANADAR)*COS(PSIRDN))*VTOTZ                        VPR11050
      CALL ANGLES(VTOTX,VTOTY,VTOTZ,VAXIAL,VRADAL,VTANG,PSI,ROTANG,     VPR11100
     1OUTANG,UPWASH,SIDWSH)                                             VPR11150
C                                                                       VPR11200
C     FOR THIS RADIUS AND AZIMUTH LOCATION WRITE THE NEW FOUND          VPR11250
C     VELOCITIES AND ANGLES ON A TEMPORARY SEQUENTIAL SCRATCH TAPE OR   VPR11300
C     DISK FILE 50 SO THEY MAY ALL BE PRINTED TOGETHER LATER IN         VPR11350
C     OUTPUT TABLE PART 2.                                              VPR11400
C     PUNCH THE FLOW CALCULATION FOR THIS POINT (RLOCAL,PSIRDN) IF      VPR11450
C     PUNCH OPTION WAS SPECIFIED.                                       VPR11500
      WRITE(50)RRATIO,PSI,VAXIAL,VRADAL,VTANG,ROTANG,OUTANG,UPWASH,     VPR11550
     1SIDWSH                                                            VPR11600
      IF(NPUNCH.EQ.0) GO TO 12                                          VPR11650
      WRITE(7,906) PSI,RRATIO,VAXIAL,VTANG                              VPR11700
   12 CONTINUE                                                          VPR11750
      WRITE(6,903)                                                      VPR11800
   13 CONTINUE                                                          VPR11850
      REWIND 50                                                         VPR11900
C                                                                       VPR11950
C-----------------------------------------------------------------------VPR12000
C                                                                       VPR12050
C     PRINT VELOCITIES AND FLOW ANGLES IN PART 2 OF OUTPUT FOR THIS     VPR12100
C     PROPELLER PLANE FROM SEQUENTIAL SCRATCH FILE 50.                  VPR12150
C                                                                       VPR12200
      WRITE(6,907) STAR,STAR,STAR,(STARS(K),K=1,10)                     VPR12250
      WRITE(6,908)                                                      VPR12300
      WRITE(6,909)                                                      VPR12350
      WRITE(6,910)                                                      VPR12400
      WRITE(6,400)ICASE,ALPHA(ICASE)                                    VPR12450
      WRITE(6,500)BETA(ICASE)                                           VPR12500
      IF(NWING.EQ.1) GO TO 14                                           VPR12550
      WRITE(6,501)                                                      VPR12600
```

```
      GO TO 15                                                          VPR12650
   14 WRITE(6,502) CL(ICASE)                                            VPR12700
   15 WRITE(6,201) STAR,STAR,STAR,(STARS(K),K=1,10)                     VPR12750
      IF(NINFLO.EQ.0) GO TO 16                                          VPR12800
      WRITE(6,705) FLRATO(1)                                            VPR12850
      GO TO 17                                                          VPR12900
   16 WRITE(6,706)                                                      VPR12950
   17 WRITE(6,911)                                                      VPR13000
      WRITE(6,912)                                                      VPR13050
      WRITE(6,913)                                                      VPR13100
      WRITE(6,914)                                                      VPR13150
      WRITE(6,915)                                                      VPR13200
      WRITE(6,916)                                                      VPR13250
      WRITE(6,917)                                                      VPR13300
      WRITE(6,918)                                                      VPR13350
      WRITE(6,919)                                                      VPR13400
      WRITE(6,920)                                                      VPR13450
      WRITE(6,921)                                                      VPR13500
      WRITE(6,922)                                                      VPR13550
      WRITE(6,803) RADIUS                                               VPR13600
      WRITE(6,804) ALPHP                                                VPR13650
      WRITE(6,805) BETAP                                                VPR13700
      WRITE(6,806)                                                      VPR13750
      WRITE(6,923)                                                      VPR13800
      DO 19 I=1,NPSIS                                                   VPR13850
      DO 18 J=1,NRAD                                                    VPR13900
      READ(50)RRATIO,PSI,VAXIAL,VRADAL,VTANG,ROTANG,OUTANG,UPWASH,       VPR13950
     1SIDWSH                                                            VPR14000
      WRITE(6,924) RRATIO,PSI,VAXIAL,VRADAL,VTANG,ROTANG,OUTANG,UPWASH, VPR14050
     1SIDWSH                                                            VPR14100
   18 CONTINUE                                                          VPR14150
      WRITE(6,903)                                                      VPR14200
   19 CONTINUE                                                          VPR14250
      REWIND 50                                                         VPR14300
C                                                                       VPR14350
      WRITE(6,904)(STARS(K),STARS(K),STARS(K),K=1,11)                   VPR14400
```

```
      RETURN                                                            VPR14450
C                                                                       VPR14500
C-----------------------------------------------------------------------VPR14550
C                                                                       VPR14600
C     FORMATS                                                           VPR14650
C                                                                       VPR14700
  100 FORMAT('0' T33,'/// NO PROPELLER PLANE FLOWFIELD CALCULATIONS MADEVPR14750
     1 ON THIS RUN ///'//)                                              VPR14800
  200 FORMAT('1',T40,13A4)                                              VPR14850
  201 FORMAT(' ',T40,13A4)                                              VPR14900
  250 FORMAT(' ',T40,'*',T49,'PART 1 : PROPELLER PLANE OUTPUT',T91,'*') VPR14950
  300 FORMAT(' ',T40,'*',T44,'VELOCITIES AT POINTS IN THE "ROPELLER PLANVPR15000
     1E',T91,'*')                                                       VPR15050
  400 FORMAT(' ',T40,'*',T48,'CASE ',I2,' - ALPHA = ',F9.3,' DEGREES',  VPR15100
     1T91,'*')                                                          VPR15150
  500 FORMAT(' ',T40,'*',T58,'BETA  = ',F9.3,' DEGREES',T91,'*')        VPR15200
  501 FORMAT(' ',T40 '*',T58,'NO WING (CL) PRESENT',T91,'*')            VPR15250
  502 FORMAT(' ',T40,'*',T60,'CL  =' F10.3,T91,'*')                     VPR15300
  600 FORMAT(/,'0' T40,'MAGNITUDE OF FREESTREAM VELOCITY   V = ',F13.7) VPR15350
  700 FORMAT(' ',T40,'COMPONENTS OF FREESTREAM VELOCITY  VX= ',F13.7)   VPR15400
  701 FORMAT(' ',T75,'VY= ' F13.7)                                      VPR15450
  702 FORMAT(' ',T75,'VZ= ',F13.7)                                      VPR15500
  703 FORMAT(' ',T46,'INLET FLOW RATIO SPECIFIED',/,' ',T46,'ON FIRST THVPR15550
     1ROUGH-FLOW PANEL    = ',F13.7,' (POS. IS INFLOW, NEG. IS OUTFLOW)'VPR15600
     2)                                                                 VPR15650
  704 FORMAT(' ',T40,'NO INLET OR OUTLET FLOW MODELED ON THE BODY')     VPR15700
  705 FORMAT(/,'0',T46 'INLET FLOW RATIO SPECIFIED' /,' ',T46,'ON FIRST VPR15750
     1THROUGH-FLOW PANEL   = ',F13.7,' (POS. IS INFLOW, NEG. IS OUTFLOW)VPR15800
     2')                                                                VPR15850
  706 FORMAT(/,'0',T40,'NO INLET OR OUTLET FLOW MODELED ON THE BODY')   VPR15900
  800 FORMAT('0',T50,'PROPELLER PLANE DATA (PART 1) : '/)               VPR15950
  801 FORMAT(' ',T22,'ALL VELOCITY COMPONENTS AND PROP PLANE ANGLES - REVPR16000
     1LATIVE TO BODY (AIRCRAFT) AXIS SYSTEM.')                          VPR16050
  802 FORMAT('0' T33,'COORDINATES OF PROP HUB RELATIVE TO BODY AXES-  XHVPR16100
     1UB= ',F12.7/' ',T81,'YHUB= ',F12.7/' ',T81,'ZHUB= ',F12.7)        VPR16150
  803 FORMAT(' ',T11,'REFERENCE BLADE RADIUS (RADIUS)   = ',F10.5)      VPR16200
```

```
804 FORMAT(' ',T18,'ALPHA OF PROP (ALPHP)       = ',F10.5,' DEGREES - (VPR16250
   1POSITIVE FOR A THRUST COMPONENT TOWARD NEG. Z)')                   VPR16300
805 FORMAT(' ',T18,'BETA  OF PROP (BETAP)       = ',F10.5,' DEGREES - (VPR16350
   1POSITIVE FOR A THRUST COMPONENT TOWARD POS. Y)')                   VPR16400
806 FORMAT('0',T29,'*NOTE : AZIMUTH DEFINITION FOR PROPELLER PLANE AT VPR16450
   1ALPHP=BETAP= 0.0 :'/,' ',T37,'BLADE AZIMUTH ANGLE IS DEFINED 0 DEGVPR16500
   2REES UPWARD (ALONG NEGATIVE Z) ;'/,' ',T37,'INCREASES CLOCKWISE ASVPR16550
   3VIEWED IN POSITIVE X (NOSE FOREWARD) DIRECTION.'/)                 VPR16600
900 FORMAT(/ '0',T2,'POINT POSITION IN TERMS OF PROPELLER AXES',T50,'PVPR16650
   1OINT COORDINATES IN BODY-',T91,'NORMALIZED COMPONENTS AND RESULTANVPR16700
   2T'/,' ',T52,'CARTESIAN AXIS SYSTEM',T91,'RESULTANT VELOCITY AT THEVPR16750
   3 POINT RELATIVE TO')                                               VPR16800
901 FORMAT(' ' T4,'RADIAL POSITION',T23,'AZIMUTH POSITION',T91,'BODY FVPR16850
   1IXED (AIRCRAFT) AXIS SYSTEM'/,' ',T4,'NON-DIMENSIONAL',T26,'(DEGREVPR16900
   2ES)'/,' ',T5,'RLOCAL/RADIUS',T29,'PSI',T47,'POINTX',T61,'POINTY', VPR16950
   3T75,'POINTZ',T89,'VTOTX/V',T101,'VTOTY/V',T113,'VTOTZ/V',T125,     VPR17000
   4'VTOT/V'/)                                                         VPR17050
902 FORMAT(' ',T8,F7.3,T26,F8.3,T44,F12.6,T58,F12.6,T72,F12.6,T87,     VPR17100
   1F10.4,T99,F10.4,T111,F10.4,T123,F10.4)                             VPR17150
903 FORMAT(' ')                                                        VPR17200
904 FORMAT('0' 33A4)                                                   VPR17250
905 FORMAT(I5,T11,I5,T21,5F10.4)                                       VPR17300
906 FORMAT(F10.4,T21,F10.4,T41,F10.4,T61,F10.4)                        VPR17350
907 FORMAT(///,'0',T40,13A4)                                           VPR17400
908 FORMAT(' ' T40,'*',T48,'PART 2 : PROPELLER PLANE OUTPUT',T91,'*') VPR17450
909 FORMAT(' ',T40,'*',T46,'AXIAL, RADIAL, AND TANGENTIAL VELOCITIES',VPR17500
   1T91,'*')                                                           VPR17550
910 FORMAT(' ',T40,'*',T57,'AND FLOW ANGLES',T91,'*')                  VPR17600
911 FORMAT(//,' ' T50,'PROPELLER PLANE DATA (PART 2) .'/)              VPR17650
912 FORMAT(' ',T40,'*NOTE : A RIGHT HAND ROTATING PROPELLER IS ASSUMEDVPR17700
   1 FOR'/,' ',T48,'SIGN CONVENTION ON TANGENTIAL VELOCITIES BELOW:'/)VPR17750
913 FORMAT(' ',T28,'ALL VELOCITIES LISTED BELOW',T55,' - RELATIVE TO PVPR17800
   1ROPELLER PLANE CYLINDRICAL AXES')                                  VPR17850
914 FORMAT(' ',T18,'ANGLES OF ROTATIONAL FLOW AND OUTFLOW',T55,' - RELVPR17900
   1ATIVE TO PROPELLER PLANE CYLINDRICAL AXES')                        VPR17950
915 FORMAT(' ' T29,'UPWASH AND SIDEWASH ANGLES',T55,' - RELATIVE TO BOVPR18000
```

```
     1DY (AIRCRAFT) AXIS SYSTEM')                                       VPR18050
  916 FORMAT('0',T24,'NORMALIZED AXIAL VELOCITY (VAXIAL/V)',T60,' : POSIVPR18100
     1TIVE - IN DIRECTION OF PROPELLER THRUST')                         VPR18150
  917 FORMAT(' ',T23,'NORMALIZED RADIAL VELOCITY (VRADAL/V)',T60,' : POSVPR18200
     1ITIVE - FROM HUB TO TIP')                                         VPR18250
  918 FORMAT(' ',T20,'NORMALIZED TANGENTIAL VELOCITY (VTANG/V)',T60,' : VPR18300
     1POSITIVE - WITH DIRECTION OF LOCAL BLADE ROTATION')               VPR18350
  919 FORMAT(' ',T27,'ANGLE OF ROTATIONAL FLOW (ROTANG)')               VPR18400
  920 FORMAT(' ',T35,'ANGLE OF OUTFLOW (OUTANG)')                       VPR18450
  921 FORMAT(' ',T39,'UPWASH ANGLE (UPWASH)',T60,' : ANGLE AS SEEN IN BOVPR18500
     1DY AXIS X-Z PLANE')                                               VPR18550
  922 FORMAT(' ',T37,'SIDEWASH ANGLE (SIDWSH)',T60,' : ANGLE AS SEEN IN VPR18600
     1BODY AXIS X-Y PLANE'/)                                            VPR18650
  923 FORMAT(/,'0',T2,'POINT POSITION IN TERMS OF PROPELLER AXES',T49,'NVPR18700
     1ORMALIZED VELOCITY COMPONENTS',T95,'FLOW DIRECTION ANGLES'/,' ',  VPR18750
     2T49,'IN PROPELLER CYLINDRICAL AXES'/,' ',T4,'RADIAL POSITION',T24,VPR18800
     3'AZIMUTH POSITION'/,' ' T4,'NON-DIMENSIONAL',T27,'(DEGREES)',T87, VPR18850
     4'ROTANG',T98,'OUTANG',T109,'UPWASH',T120,'SIDWSH'/,' ',T5,'RLOCAL/VPR18900
     5RADIUS',T30,'PSI',T48,'VAXIAL/V',T60,'VRADAL/V',T73,'VTANG/V',T85,VPR18950
     6'(DEGREES)',T96,'(DEGREES)',T107,'(DEGREES)',T118,'(DEGREES)'/)   VPR19000
  924 FORMAT(' ',T7,F7.3,T28,F8.3,T47,F10.4,T59,F10.4,T71,F10.4,T85,F9.4VPR19050
     1,T96,F9.4,T107,F9.4,T118,F9.4)                                    VPR19100
  925 FORMAT(80A1)                                                      VPR19150
 1000 FORMAT('0',T1,'///WARNING/// - INPUT VALUE OF DPSI= ',F8.3,' DOES VPR19200
     1NOT DIVIDE 360 BY INTEGER AMOUNT. CALCULATIONS'/,' ',T17,'ARE MADEVPR19250
     2 BUT AZIMUTHAL SPACING OF POINTS AROUND THE PROP DISK WILL BE UNEVVPR19300
     3EN.'/)                                                            VPR19350
C                                                                       VPR19400
      END                                                               VPR19450
C                                                                       VPR19500
C     *****************************************************************VPR19550
C                                                                       VPR19600
C                                                                       VPR19650
C                                                                       VPR19700
```

```
C     SUBROUTINE WINGV(ICASE,X,Y,Z,VELX,VELY,VELZ)                       WGV   50
C                                                                        WGV  100
C     *****************************************************************WGV  150
C     *****************************************************************WGV  200
C     CALCULATES THE COMPONENTS OF VELOCITY INDUCED AT POINT (X,Y,Z) DUEWGV  250
C     TO THE WING HORSESHOE VORTEX MODEL GENERATED IN SUBROUTINE WGEOM. WGV  300
C     INPUT PARAMETERS INCLUDE THE FOLLOWING:                            WGV  350
C      ICASE:          BODY ORIENTATION (ALPHA-BETA) AND LIFT (CL) CASE  WGV  400
C                       FOR WHICH CALCULATION IS BEING MADE.             WGV  450
C      X,Y, AND Z:     COORDINATES OF POINT AT WHICH INDUCED VELOCITY    WGV  500
C                      COMPONENTS ARE TO BE FOUND.                       WGV  550
C      XQR,YQR,ZQR:    COORDINATES OF HORSESHOE BOUND VORTEX MID POINT   WGV  600
C                       FROM COMMON BLOCK /WING1/.                       WGV  650
C     REMAINING INPUT PARAMETERS WERE CALCULATED IN SUBROUTINE WGEOM ANDWGV  700
C     ARE OBTAINED FROM COMMON BLOCK /WING2/:                            WGV  750
C      XTRALL(ICASE): COORDINATES OF THE AFT END POINT OF LEFT SIDE      WGV  800
C      YTRALL(ICASE): TRAILING VORTEX FILAMENT - IS ''STARTING'' POINT   WGV  850
C      ZTRALL(ICASE): OF THAT FILAMENT CORRESPONDING TO INPUT BODY       WGV  900
C                     ORIENTATION CASE, ICASE.                           WGV  950
C                                                                        WGV 1000
C      XBTIPL:        COORDINATES OF LEFT END OF BOUND VORTEX FILAMENT. WGV 1050
C      YBTIPL:        ALSO IT IS THE POINT FROM WHICH LEFT TRAILING      WGV 1100
C      ZBTIPL:        VORTEX IS SHED. SERVES AS ''END'' POINT OF LEFT    WGV 1150
C                     TRAILING VORTEX FILAMENT.                          WGV 1200
C                                                                        WGV 1250
C      XBTIPR:        COORDINATES OF RIGHT END OF BOUND VORTEX FILAMENT.WGV 1300
C      YBTIPR:        ALSO IT IS THE POINT FROM WHICH THE RIGHT SIDE     WGV 1350
C                     TRAILING VORTEX IS SHED. SERVES AS ''STARTING''    WGV 1400
C                     POINT OF THE RIGHT TRAILING VORTEX FILAMENT.       WGV 1450
C                                                                        WGV 1500
C      XTRALR(ICASE): COORDINATES OF THE AFT END POINT OF THE RIGHT SIDEWGV 1550
C      YTRALR(ICASE): TRAILING VORTEX FILAMENT - IS THE ''END'' POINT OFWGV 1600
C      ZTRALR(ICASE): THAT FILAMENT CORRESPONDING TO THE INPUT BODY      WGV 1650
C                     ORIENTATION CASE, ICASE.                           WGV 1700
C                                                                        WGV 1750
C      GAMMA(ICASE):  STRENGTH OF EACH FILAMENT OF THE HORSESHOE VORTEX WGV 1800
```

```
C                     CORRESPONDING TO THE INPUT BODY ORIENTATION       WGV 1850
C                     CASE, ICASE.                                      WGV 1900
C                                                                       WGV 1950
C     OUTPUT VALUES RETURNED:                                           WGV 2000
C      VELX:          X,Y, AND Z COMPONENTS OF THE VELOCITY INDUCED AT  WGV 2050
C      VELY:          POINT (X,Y,Z) BY THE ENTIRE WING HORSESHOE        WGV 2100
C      VELZ:          VORTEX SYSTEM.                                    WGV 2150
C                                                                       WGV 2200
C     ROUTINE BREAKS UP THE HORSESHOE VORTEX INTO FOUR STRAIGHT VORTEX  WGV 2250
C     FILAMENT PARTS; LEFT TRAILING FILAMENT, LEFT HALF OF BOUND        WGV 2300
C     FILAMENT, RIGHT HALF OF BOUND FILAMENT, AND RIGHT TRAILING        WGV 2350
C     FILAMENT. THE VELOCITY CONTRIBUTION DUE TO EACH OF THESE FOUR     WGV 2400
C     FILAMENTS IS FOUND IN TURN USING SUBROUTINE VORTEX. THE FOUR      WGV 2450
C     FILAMENT CONTRIBUTIONS ARE SUMMED TO GIVE THE NET CONTRIBUTION OF WGV 2500
C     THE ENTIRE HORSESHOE SYSTEM.                                      WGV 2550
C     ******************************************************************WGV 2600
C     ******************************************************************WGV 2650
C                                                                       WGV 2700
      SUBROUTINE WINGV(ICASE,X,Y,Z,VELX,VELY,VELZ)                      WGV 2750
      COMMON /WING1/ CL(6),CHORD,DIHED,SPAN,SWEEP,XQR,YQR,ZQR,NWING     WGV 2800
      COMMON /WING2/ GAMMA(6),XTRALL(6),YTRALL(6),ZTRALL(6),XTRALR(6),  WGV 2850
     1YTRALR(6),ZTRALR(6),XBTIPL,YBTIPL,ZBTIPL,XBTIPR,YBTIPR,ZBTIPR     WGV 2900
      VELX=0.0                                                          WGV 2950
      VELY=0.0                                                          WGV 3000
      VELZ=0.0                                                          WGV 3050
C                                                                       WGV 3100
C     GET CONTRIBUTION OF LEFT SIDE TRAILING VORTEX FILAMENT:           WGV 3150
C                                                                       WGV 3200
      CALL VORTEX(X,Y,Z,XTRALL(ICASE),YTRALL(ICASE),ZTRALL(ICASE),      WGV 3250
     1XBTIPL,YBTIPL,ZBTIPL,GAMMA(ICASE),VVX,VVY,VVZ)                    WGV 3300
      VELX=VELX+VVX                                                     WGV 3350
      VELY=VELY+VVY                                                     WGV 3400
      VELZ=VELZ+VVZ                                                     WGV 3450
C                                                                       WGV 3500
C     GET CONTRIBUTION OF LEFT HALF OF BOUND VORTEX FILAMENT:           WGV 3550
C                                                                       WGV 3600
```

```
      CALL VORTEX(X,Y,Z,XBTIPL,YBTIPL ZBTIPL,XQR,YQR,ZQR,GAMMA(ICASE),
     1VVX,VVY,VVZ)
      VELX=VELX+VVX
      VELY=VELY+VVY
      VELZ=VELZ+VVZ
C
C     GET CONTRIBUTION OF RIGHT HALF OF BOUND VORTEX FILAMENT:
C
      CALL VORTEX(X,Y,Z,XQR,YQR,ZQR,XBTIPR,YBTIPR,ZBTIPR,GAMMA(ICASE),
     1VVX,VVY,VVZ)
      VELX=VELX+VVX
      VELY=VELY+VVY
      VELZ=VELZ+VVZ
C
C     LAST GET CONTRIBUTION OF RIGHT SIDE TRAILING VORTEX FILAMENT. THE
C     SUM IS THE DESIRED FINAL INDUCED VELOCITY COMPONENTS.
C
      CALL VORTEX(X,Y,Z,XBTIPR,YBTIPR,ZBTIPR,XTRALR(ICASE),
     1YTRALR(ICASE),ZTRALR(ICASE),GAMMA(ICASE),VVX,VVY,VVZ)
      VELX=VELX+VVX
      VELY=VELY+VVY
      VELZ=VELZ+VVZ
      RETURN
      END
C
C
C     ***********************************************************************
C
C
C
```

```
C     SUBROUTINE VORTEX (X,Y,Z,X1,Y1,Z1,X2,Y2,Z2,GAMMA,VVX,VVY,VVZ)      VOR
C                                                                        VOR
C     *******************************************************************VOR
C     *******************************************************************VOR
C     CALCULATES AND RETURNS VELOCITY COMPONENTS VVX, VVY, AND VVZ       VOR
C     INDUCED BY A STRAIGHT LINE FINITE LENGTH VORTEX FILAMENT USING     VOR
C     THE BIOT-SAVART LAW. INPUT QUANTITIES TO THE SUBROUTINE:           VOR
C     X,Y,Z     : COORDINATES OF POINT AT WHICH INDUCED VELOCITIES ARE TOVOR
C                 BE FOUND.                                              VOR
C     X1,Y1,Z1 : COORDINATES OF STARTING POINT OF FILAMENT.              VOR
C     X2,Y2,Z2 : COORDINATES OF ENDPOINT OF FILAMENT.                    VOR
C     (FILAMENT POINTS ARE DEFINED IN THE (RIGHT HAND RULE) VECTOR       VOR
C     SENSE - CIRCULATION WOULD TURN ITSELF FROM THE STARTING POINT TO   VOR
C     THE END POINT).                                                    VOR
C     GAMMA     : CONSTANT STRENGTH OF THE FILAMENT CIRCULATION.         VOR
C     V         : FREESTREAM VELOCITY MAGNITUDE (USED TO SET CUTOFF LIMITVOR
C                 ON MAXIMUM INDUCED VELOCITY).                          VOR
C                                                                        VOR
C     OUTPUT QUANTITIES RETURNED - (NO ERROR RETURNS EXIST) :            VOR
C     VVX, VVY, VVZ : COMPONENTS OF VELOCITY INDUCED BY VORTEX FILAMENT.VOR
C       - ALL SET TO ZERO IF (X,Y,Z) LIES ON THE LINE OF THE FILAMENT.   VOR
C       - EACH CALCULATED USING BIOT-SAVART LAW IF POINT IS NOT ON THE   VOR
C         LINE OF THE FILAMENT.                                          VOR
C       - IF MAGNITUDE OF TOTAL INDUCED VELOCITY IS MORE THAN 20         VOR
C         PERCENT OF V, THE COMPUTED MAGNITUDE IS IGNORED AND REPLACED   VOR
C         BY A VALUE EQUAL TO 20 PERCENT OF V. EACH INDUCED COMPONENT    VOR
C         IS ADJUSTED ACCORDINGLY.                                       VOR
C                                                                        VOR
C     IMPOSING THE LIMIT OF 0.2*V ON A SINGLE FILAMENT PREVENTS THE      VOR
C     TOTAL VELOCITY INDUCED BY THE WING HORSESHOE VORTEX FROM GOING     VOR
C     MUCH ABOVE 0.4*V.                                                  VOR
C     *******************************************************************VOR
C     *******************************************************************VOR
C                                                                        VOR
      SUBROUTINE VORTEX (X,Y,Z,X1,Y1,Z1,X2,Y2,Z2,GAMMA,VVX,VVY,VVZ)      VOR
      COMMON /INPUTS/ ALPHA(6),BETA(6),VX(6),VY(6),VZ(6),V,NALPHA        VOR
```

```
C
C     R IS THE LENGTH OF THE VORTEX FILAMENT.
C     DEFINE (XC,YC,ZC) AS THE POINT LYING ON THE LINE CONTAINING THE
C     VORTEX FILAMENT SUCH THAT A LINE FROM (X,Y,Z) TO (XC,YC,ZC) IS
C     PERPENDICULAR TO THE FILAMENT. THIS PERPENDICULAR LINE GIVES THE
C     DISTANCE H FROM THE POINT (X,Y,Z) TO THE FILAMENT.
C
      A=X2-X1
      B=Y2-Y1
      C=Z2-Z1
      R=SQRT(A**2 + B**2 + C**2)
      XC=(A**2*X + (B**2+C**2)*X1 + A*(B*(Y-Y1)+C*(Z-Z1)))/R**2
      YC=(B**2*Y + (A**2+C**2)*Y1 + B*(A*(X-X1)+C*(Z-Z1)))/R**2
      ZC=(C**2*Z + (A**2+B**2)*Z1 + C*(A*(X-X1)+B*(Y-Y1)))/R**2
      H=SQRT((XC-X)**2 + (YC-Y)**2 + (ZC-Z)**2)
C
C     IF H IS ZERO, THE POINT LIES ON THE LINE CONTAINING THE FILAMENT.
C     IF SO, SET INDUCED VELOCITY COMPONENTS TO ZERO AND RETURN.
C
      IF(H.GT.0.0) GO TO 1
      VVX=0.0
      VVY=0.0
      VVZ=0.0
      RETURN
C
    1 CONTINUE
C
C     DISTANCE H IS NOT ZERO. COMPUTE VELOCITY BY BIOT-SAVART LAW:
C     D1 IS LENGTH OF LINE SEGMENT JOINING POINT (X,Y,Z) AND FILAMENT
C     STARTING POINT (X1,Y1,Z1). D2 IS LENGTH OF LINE SEGMENT JOINING
C     POINT (X,Y,Z) AND FILAMENT END POINT (X2,Y2,Z2). TOGETHER,
C     SEGMENTS R, D1, AND D2 FORM A TRIANGLE OF BASE R WITH HEIGHT H.
C     COSALP IS COSINE OF THE INSIDE ANGLE BETWEEN SIDE R AND SIDE D2
C     OF THIS TRIANGLE. COSBET IS COSINE OF THE INSIDE ANGLE BETWEEN
C     SIDE R AND SIDE D1 OF THE TRIANGLE. THESE COSINES ARE FOUND USING
C     THE COSINE LAW OF TRIANGLES. THEN THE SIMPLE BIOT-SAVART LAW FOR
```

```
C     STRAIGHT LINE FINITE VORTEX FILAMENTS IS USED TO COMPUTE THE
C     MAGNITUDE OF VORTEX INDUCED VELOCITY, VINDUC, AT POINT (X,Y,Z).
C
      D1=SQRT((X1-X)**2 + (Y1-Y)**2 + (Z1-Z)**2)
      D2=SQRT((X2-X)**2 + (Y2-Y)**2 + (Z2-Z)**2)
      COSALP=(R**2 + D2**2 - D1**2)/(2.0*R*D2)
      COSBET=(R**2 + D1**2 - D2**2)/(2.0*R*D1)
      VINDUC=(GAMMA*(COSALP+COSBET))/(4.0*3.141593*H)
C
C     CHECK IF CALCULATED VELOCITY IS MORE THAN 0.2*V. IF IT IS, IGNORE
C     THE VALUE AND REPLACE IT WITH THE VELOCITY LIMIT OF 0.2*V.
C
      IF(VINDUC.GT.(0.2*V)) VINDUC=0.2*V
C
C     DIRECTION OF VELOCITY, VINDUC, IS ALONG A UNIT VECTOR. THE UNIT
C     VECTOR IS FOUND AS THE VECTOR CROSS PRODUCT OF (SMALL R VECTOR
C     FROM X,Y,Z TO XC,YC,ZC) CROSS (VORTEX FILAMENT VECTOR R OF LENGTH
C     R). THIS PRODUCT, OF MAGNITUDE RXRMAG, WHEN NORMALIZED TO UNIT
C     LENGTH IS THE INDUCED VELOCITY DIRECTION VECTOR. THIS UNIT VECTOR
C     HAS SCALER COMPONENTS XNORM, YNORM, AND ZNORM GIVEN BELOW. THEN
C     THESE SCALER COMPONENTS COMBINED WITH MAGNITUDE OF TOTAL INDUCED
C     VELOCITY GIVE THE THREE SCALER COMPONENTS OF VORTEX INDUCED
C     VELOCITY AT POINT (X,Y,Z).
C
      RXRMAG=SQRT((C*(YC-Y)-B*(ZC-Z))**2 + (A*(ZC-Z)-C*(XC-X))**2 +
     + (B*(XC-X)-A*(YC-Y))**2)
      XNORM=(C*(YC-Y)-B*(ZC-Z))/RXRMAG
      YNORM=(A*(ZC-Z)-C*(XC-X))/RXRMAG
      ZNORM=(B*(XC-X)-A*(YC-Y))/RXRMAG
      VVX=VINDUC*XNORM
      VVY=VINDUC*YNORM
      VVZ=VINDUC*ZNORM
      RETURN
      END
C
C     ******************************************************************
```

VOR 5450 C
VOR 5500 C
VOR 5550 C

```
      SUBROUTINE ANGLES(VTOTX,VTOTY,VTOTZ,VAXIAL,VRADAL,VTANG,PSI,      ANG
     1ROTANG,OUTANG,UPWASH,SIDWSH)                                      ANG
C                                                                       ANG
C     *****************************************************************ANG
C     *****************************************************************ANG
C     COMPUTES FLOW ANGLES AT A SINGLE POINT IN PROPELLER PLANE (IN     ANG
C     DEGREES).  LOGIC CHECKS FOR PROPER QUADRANT OF ANGLE              ANG
C     PRODUCED BY INVERSE TAN FUNCTIONS.  THE OUTPUT IS THE SET OF      ANG
C     ANGLES : ANGLE OF ROTATIONAL FLOW, ANGLE OF OUTFLOW, UPWASH,      ANG
C     AND SIDEWASH.                                                     ANG
C     *****************************************************************ANG
C     *****************************************************************ANG
C                                                                       ANG
      PI=3.141593                                                       ANG
C                                                                       ANG
C     COMPUTE ANGLES OF ROTATIONAL FLOW AND OUTFLOW                     ANG
C     CHECK FOR SPECIAL CASE OF NO AXIAL FLOW VELOCITY--                ANG
C                                                                       ANG
      IF(VAXIAL.NE.0.0)GO TO 50                                         ANG
      IF(VRADAL.GT.0.0)GO TO 10                                         ANG
      OUTANG=-90.0                                                      ANG
      GO TO 20                                                          ANG
   10 OUTANG=90.0                                                       ANG
   20 IF(VTANG.GT.0.0) GO TO 30                                         ANG
      ROTANG=-90.0                                                      ANG
      GO TO 40                                                          ANG
   30 ROTANG=90.0                                                       ANG
   40 GO TO 60                                                          ANG
C                                                                       ANG
C     AXIAL VELOCITY IS NON-ZERO -- GET ANGLES AND PROPER QUADRANT      ANG
C                                                                       ANG
   50 ROTANG=ATAN(-VTANG/VAXIAL)*(180./PI)                              ANG
      OUTANG=ATAN(-VRADAL/VAXIAL)*(180./PI)                             ANG
      IF((VAXIAL.GT.0.0).AND.(VTANG.EQ.0.0)) ROTANG=180.0               ANG
      IF((VAXIAL.GT.0.0).AND.(VTANG.GT.0.0)) ROTANG=ROTANG+180.0        ANG
      IF((VAXIAL.GT.0.0).AND.(VTANG.LT.0.0)) ROTANG=ROTANG-180.0        ANG
```

```
      IF((VAXIAL.GT.0.0).AND.(VRADAL.EQ.0.0)) OUTANG=180.0          ANG 1850
      IF((VAXIAL.GT.0.0).AND.(VRADAL.GT.0.0)) OUTANG=OUTANG+180.0   ANG 1900
      IF((VAXIAL.GT.0.0).AND.(VRADAL.LT.0.0)) OUTANG=OUTANG-180.0   ANG 1950
C                                                                   ANG 2000
C     COMPUTE ANGLES OF UPWASH AND SIDE WASH                        ANG 2050
C     CHECK FOR SPECIAL CASE OF ZERO X COMPONENT OF VELOCITY        ANG 2100
C                                                                   ANG 2150
   60 IF(VTOTX.NE.0.0) GO TO 70                                     ANG 2200
      IF(VTOTY.EQ.0.0) SIDWSH=0.0                                   ANG 2250
      IF(VTOTY.GT.0.0) SIDWSH=90.0                                  ANG 2300
      IF(VTOTY.LT.0.0) SIDWSH=-90.0                                 ANG 2350
      IF(VTOTZ.EQ.0.0) UPWASH=0.0                                   ANG 2400
      IF(VTOTZ.GT.0.0) UPWASH=-90.0                                 ANG 2450
      IF(VTOTZ.LT.0.0) UPWASH=90.0                                  ANG 2500
      RETURN                                                        ANG 2550
C                                                                   ANG 2600
C     X VELOCITY COMPONENT NON-ZERO -- GET ANGLES AND PROPER QUADRANT ANG 2650
C                                                                   ANG 2700
   70 UPWASH=ATAN(VTOTZ/VTOTX)*(180./PI)                            ANG 2750
      SIDWSH=ATAN(-VTOTY/VTOTX)*(180./PI)                           ANG 2800
      IF((VTOTX.GT.0.0).AND.(VTOTZ.EQ.0.0)) UPWASH=180.0            ANG 2850
      IF((VTOTX.GT.0.0).AND.(VTOTZ.LT.0.0)) UPWASH=UPWASH+180.0     ANG 2900
      IF((VTOTX.GT.0.0).AND.(VTOTZ.GT.0.0)) UPWASH=UPWASH-180.0     ANG 2950
      IF((VTOTX.GT.0.0).AND.(VTOTY.EQ.0.0)) SIDWSH=180.0            ANG 3000
      IF((VTOTX.GT.0.0).AND.(VTOTY.LT.0.0)) SIDWSH=SIDWSH-180.0     ANG 3050
      IF((VTOTX.GT.0.0).AND.(VTOTY.GT.0.0)) SIDWSH=SIDWSH+180.0     ANG 3100
      RETURN                                                        ANG 3150
      END                                                           ANG 3200
```

APPENDIX C

Sample Case

C.1 Configuration

To illustrate the input and output for the computer program, an example has been run.

A simple configuration made up of 16 triangular body panels is used as shown in the three-view drawing of Figure C.1. The unit length scale is shown on the y axis in the top view of the configuration. The body consists of two octahedrons in tandem but not connected. This two-part body is used to illustrate the paneling of bodies having several body panel networks. The origin of the coordinate system is at the nose of the body, on the plane of symmetry. Panel numbers, as will be generated by the program, are shown on Figure C.1 where panels are visible.

Body Characteristics: (length unit shown on top view of Figure C.1)

length overall	4.25 units
length front part	2.00 units
length aft part	2.00 units
width	2.00 units
height	2.00 units

Four panels will be designated as inlet or ourlet panels. Panels 1 and 4, on the top front of the body, are inlet panels with an inlet velocity ratio of 0.1. Panels 14 and 15, on the lower rear of the body, are outlet panels with an outlet velocity ratio of 0.1.

A propeller plane is positioned as shown in Figure C.1. It is oriented perpendicular to the body longitudinal axis. Points on the propeller plane are shown on the front view in Figure C.1.

Propeller Plane Characteristics: (length unit shown on top view of Figure C.1)

hub coordinates	(0.25, 0.0, 0.0)
reference radius	1.5 units
azimuth position of points	0, 120, and 240 degrees
radial number of points	11
radial spacing of points	10 percent of reference radius

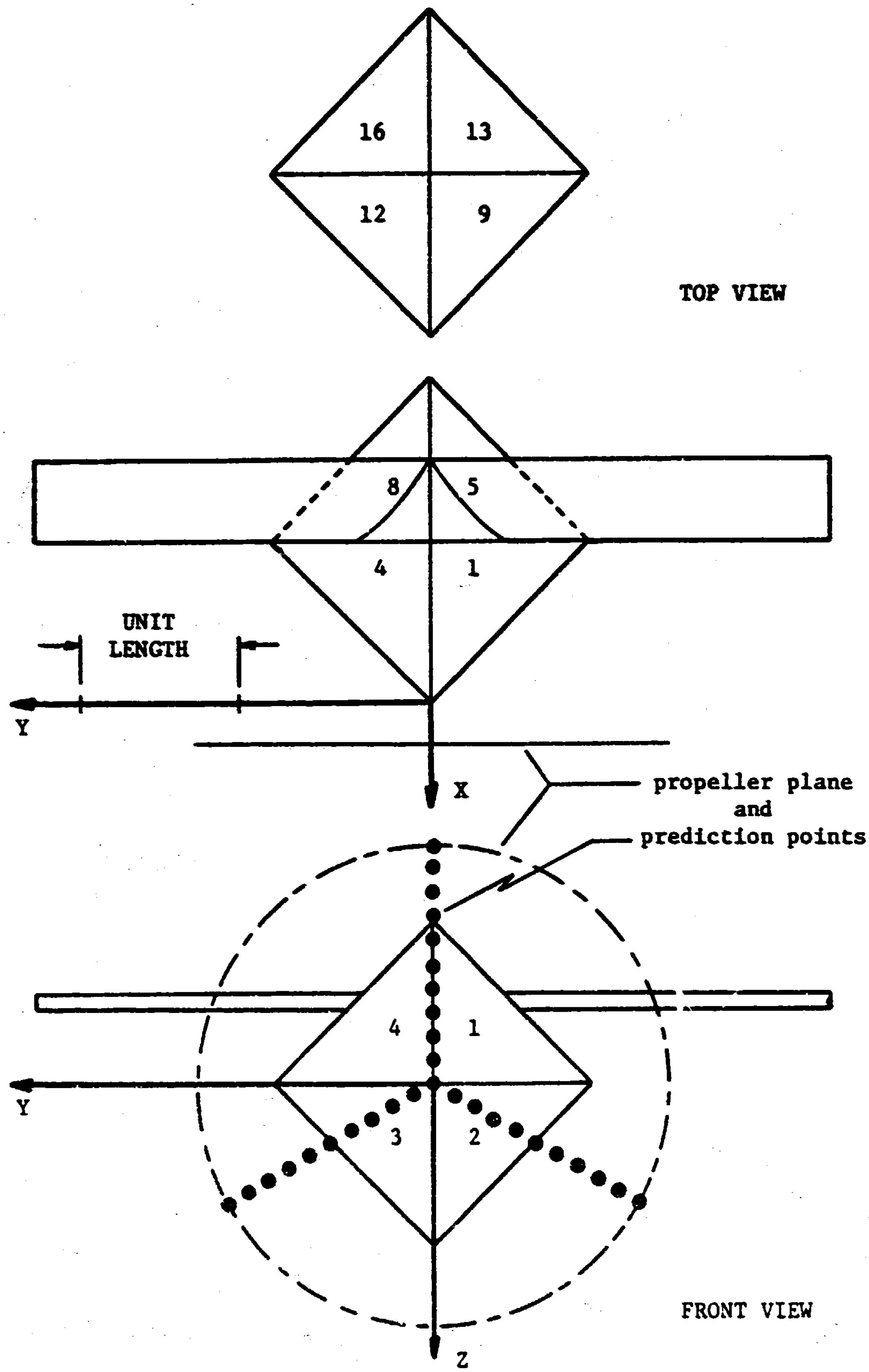
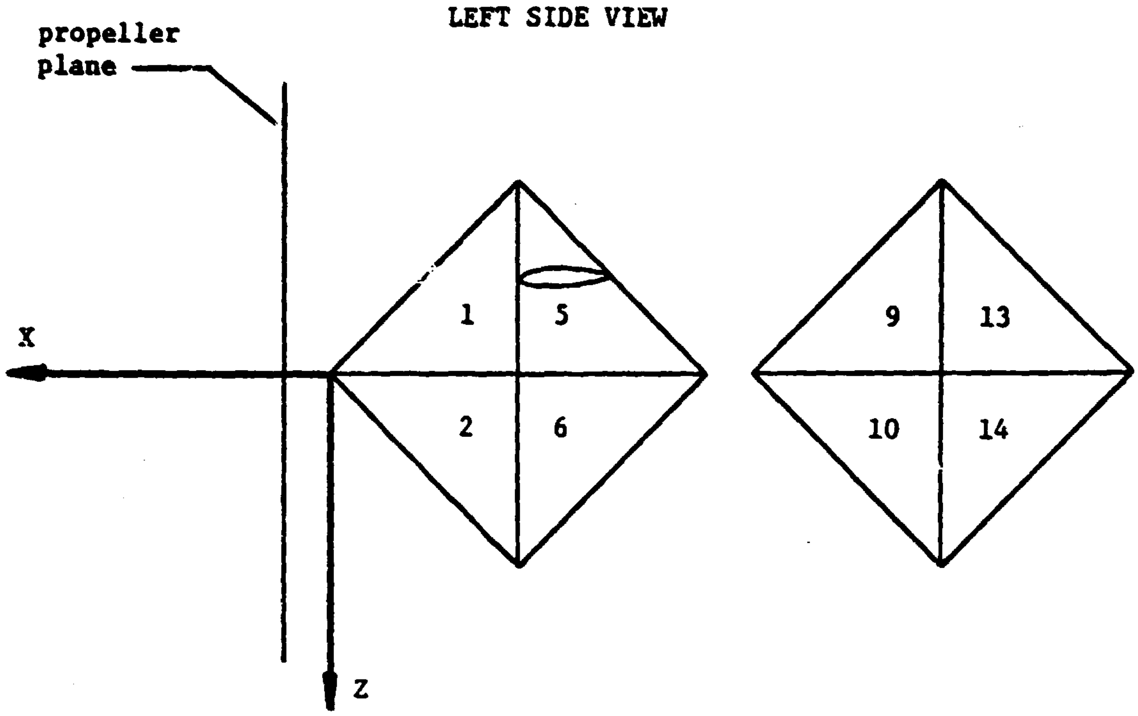


Figure C.1 4-View drawing of sample configuration (sheet 1 of 2)



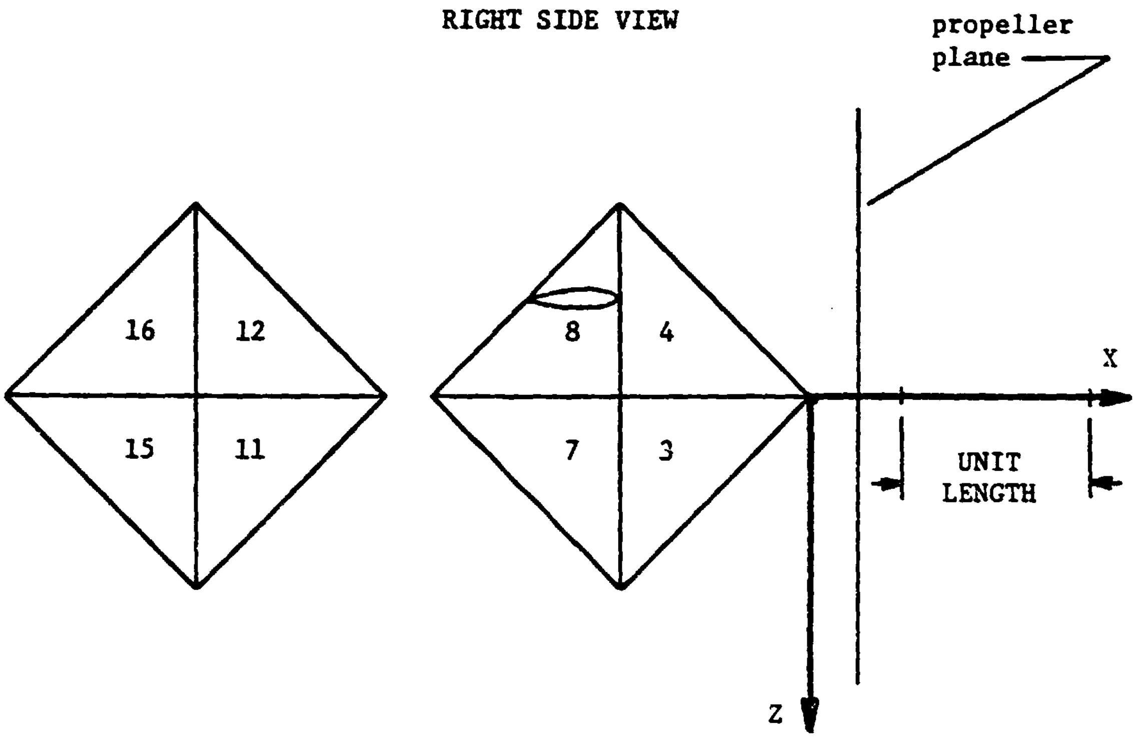


Figure C.1 Sample configuration (sheet 2 of 2)

A rectangular unswept wing having no dihedral or sweep is attached to the front part of the body as shown in Figure C.1.

Wing Characteristics: (length unit shown on top view of Figure C.1)

span	5.0 units
chord	0.5 units
location of root quarter chord	(-1.25, 0.0, -0.5)
wing C_L	0.1096 at 0 degrees body angle of attack
	1.0960 at 10 degrees body angle of attack

The configuration will be run at two orientations. First is 0.0 degrees body angle of attack and sideslip, second is 10.0 degrees body angle of attack and 0 degrees sideslip.

C.2 Input Data Card Organization

On the following two pages is a listing of the input card data for this sample case. The first and last lines of each page are not data but indicate card column numbers.

The body geometry is described using six cross sections. Since the symmetric input option can be used for this configuration, only the left side of the sections is given. Thus, only three points are specified on the cross section peripheries. Note, had the symmetric option not been used, then five points per cross section would have been needed.

This will be a normal flow prediction run since the geometry is simple and the panels may be identified without making a geometry test run. Also, the listing of body panel geometry is requested.

Of the four inlet/outlet panels, only the two panels on the left side, numbers 1 and 14, are specified. This is in accordance with symmetric input rules.

Refer to the following two pages for the actual input card data.

```
12345678901234567890123456789012345678901234567890123456789012345678901234567890
SAMPLE CASE - TANDEM OCTAHENDRONS WITH WING AND PROP., 2 INLET AND 2 OUTLET
PANELS. SYMMETRIC INPUT OPTION USED. 0 AND 10 DEGREES ANGLE OF ATTACK.
1.0
2
0.0                 0.0
10.0                0.0
0
06
01        03                  0
0.0                 0.0                 0.0
0.0                 0.0                 0.0
0.0                 0.0                 0.0
02        03                  0
-1.0                0.0                 -1.0
-1.0                -1.0                0.0
-1.0                0.0                 1.0
03        03                  1
-2.0                0.0                 0.0
-2.0                0.0                 0.0
-2.0                0.0                 0.0
04        03                  0
-2.25               0.0                 0.0
-2.25               0.0                 0.0
-2.25               0.0                 0.0
05        03                  0
-3.25               0.0                 -1.0
-3.25               -1.0                0.0
-3.25               0.0                 1.0
06        03                  0
-4.25               0.0                 0.0
-4.25               0.0                 0.0
-4.25               0.0                 0.0
0
0
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
12345678901234567890123456789012345678901234567890123456789012345678901234567890
0002
0001                0.1
0014                -0.1
1
-1.125    0.0       -0.5      5.0       0.5       0.0       0.0
0.1096
1.096
1
0.25      0.0       0.0       0.0       0.0       1.5
120.0     0.1       11
0
25        -4.0
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

C.3 Printed Output Listing

The printed output obtained for the sample case run is reproduced on the remaining 15 pages of this appendix.

```
***********************************************************************
* LISTING OF ALL INPUT DATA CARDS TO ALLOW USER CHECKING OF ERRORS *
***********************************************************************

SAMPLE CASE - TANDEM OCTAHEDRONS WITH WING AND PROP., 2 INLET AND 2 OUTLET
PANELS. SYMMETRIC INPUT OPTION USED. 0 AND 10 DEGREES ANGLE OF ATTACK.

V=      1.0000000
NALPHA=  2
ALPHA( 1)=      0.0            (ETA( 1)=     0.0
ALPHA( 2)=    10.0000000       (ETA( 2)=     0.0

NSYMET=  0

NSECTO=  6

NSEC=  1      NIP=   3        NEND= 0
X=      0.0                   Y=      0.0                  Z=     0.0
X=      0.0                   Y=      0.0                  Z=     0.0
X=      0.0                   Y=      0.0                  Z=     0.0

NSEC=  2      NIP=   3        NEND= 0
X=     -1.0000000             Y=      0.0                  Z=    -1.0000000
X=     -1.0000000             Y=     -1.0000000            Z=     0.0
X=     -1.0000000             Y=      0.0                  Z=     1.0000000

NSEC=  3      NIP=   3        NEND= 1
X=     -2.0000000             Y=      0.0                  Z=     0.0
X=     -2.0000000             Y=      0.0                  Z=     0.0
X=     -2.0000000             Y=      0.0                  Z=     0.0

NSEC=  4      NIP=   3        NEND= 0
X=     -2.2500000             Y=      0.0                  Z=     0.0
X=     -2.2500000             Y=      0.0                  Z=     0.0
X=     -2.2500000             Y=      0.0                  Z=     0.0

NSEC=  5      NIP=   3        NEND= 0
X=     -3.2500000             Y=      0.0                  Z=    -1.0000000
X=     -3.2500000             Y=     -1.0000000            Z=     0.0
X=     -3.2500000             Y=      0.0                  Z=     1.0000000

NSEC=  6      NIP=   3        NEND= 0
X=     -4.2500000             Y=      0.0                  Z=     0.0
X=     -4.2500000             Y=      0.0                  Z=     0.0
X=     -4.2500000             Y=      0.0                  Z=     0.0

NLIST= 0
NCALC= 0

NINFLO=      2
INDEX(   1)=     1     FLRATO(   1)=     0.1000000
INDEX(   2)=    14     FLRATO(   2)=    -0.1000000

NWING=  1
XOR=    -1.1250000 YOR=     0.0        ZOR=    -0.5000000 SPAN=    5.0000000 CHORD=    0.5000000 DIHED=    0.0      SWEEP=    0.0
CL( 1)=      0.1095999
CL( 2)=      1.0959997

NPOINT=  1
```

```
XHUB=      0.2500000  YHUB=       0.0           ZHUB=        0.0
DPSI=    120.0000000      DRADUS=       0.1000000     NRAD=    11       ALPHP=       0.0        BETAP=       0.0          RADIUS=      1.5000000
NPUNCH=   0

ITMAX=  25      ERR=     -4.0000000

////END OF INPUT DATA --- ALL DATA READ SUCCESSFULLY////

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

                                    *  OPTIONS REQUESTED AND CONTENTS OF THIS RUN  *

            1) NORMAL RUN OPTION USED - BODY PANELING GENERATED, EQUATIONS SOLVED, AND FLOW PREDICTIONS MADE :
            2) SURFACE FLOW CALCULATIONS MADE FOR ALL  2 INPUT BODY (ALPHA-BETA) ORIENTATIONS.
            3) SYMMETRIC BODY (LEFT SIDE) INPUT OPTION USED.
            4)    2 INLET AND/OR OUTLET PANELS ON LEFT HALF OF SYMMETRIC BODY ARE SPECIFIED.
            5) WING (WITH PROPER CL FOR EACH ALPHA) IS MODELED.
            6) PROPELLER PLANE FLOW CALCULATION MADE FOR ALL BODY (ALPHA-BETA) ORIENTATIONS.
            7) NO PUNCHED PROPELLER PLANE FLOW OUTPUT.

****************************************************************************************************************
```

****

* WING HORSESHOE VORTEX STRENGTHS AND GEOMETRY *

* (COORDINATES RELATIVE TO BODY AXIS SYSTEM) *

****

*** ASSUMPTIONS ***

- TRAILING VORTICES SEPARATED BY SPAN OF (PI/4) TIMES THE PHYSICAL WING SPAN.
- TRAILING VORTICES EXTEND AFT 100 CHORD LENGTHS AND TRAIL PARALLEL TO FREE STREAM VELOCITY.
- BOUND VORTEX FILAMENT COMPOSED OF TWO SEMISPANS - EACH WITH DIHEDRAL AND SWEEP.
- WING INPUT CL(I) ASSUMED EQUAL TO WING ROOT SECTION CL(I).

*** INPUT GEOMETRY AND FLIGHT CONDITION ***

			WING ROOT QUARTER CHORD LOCATION (COORDINATES OF BOUND VORTEX MID POINT)				
V	WING SPAN	ROOT CHORD	XQR	YQR	ZQR	DIHEDRAL (DEGREES)	SWEEP (DEGREES)
1.000000	5.000000	0.500000	-1.125000	0.0	-0.500000	0.0	0.0

*** BOUND VORTEX GEOMETRY - CONSTANT FOR ALL BODY ALPHA ORIENTATIONS ***

LEFT BOUND END POINT COORDINATES (LEFT TRAILING VORTEX END POINT)			RIGHT BOUND END POINT COORDINATES (RIGHT TRAILING VORTEX STARTING POINT)					
XBTIPL	YBTIPL	ZBTIPL	XBTIPR	YBTIPR	ZBTIPR	SPAN	DIHEDRAL (DEGREES)	SWEEP (DEGREES)
-1.125000	-1.963495	-0.500000	-1.125000	1.963495	-0.500000	3.926991	0.0	0.0

*** HORSESHOE VORTEX STRENGTH AND GEOMETRY - DEPENDENT ON BODY ORIENTATION ***

				LEFT TRAILING VORTEX AFT END LOCATION			RIGHT TRAILING VORTEX AFT END LOCATION		
I	INPUT ALPHA(I)	INPUT CL(I)	STRENGTH GAMMA(I)	XTRALL(I)	YTRALL(I)	ZTRALL(I)	XTRALR(I)	YTRALR(I)	ZTRALR(I)
1	0.0	0.109600	0.027400	-0.511250E 02	-1.963495	-0.500000	-0.511250E 02	1.963495	-0.500000
2	10.000000	1.396000	0.274000	-0.511250E 02	-1.963495	-9.053497	-0.511250E 02	1.963495	-9.050497

****

* LISTING OF GENERATED BODY PANEL GEOMETRY *

//// SYMMETRIC BODY OPTION USED. MIRROR IMAGE (RIGHT SIDE) PANELS AUTOMATICALLY GENERATED. ////

PANEL INDEX NUMBER I		CORNER POINTS (1 AND 2 ON LEADING SECTION, 3 AND 4 ON TRAILING SECTION) X1 X2 X3 X4	 Y1 Y2 Y3 Y4	 Z1 Z2 Z3 Z4	PANEL CONTROL POINT COORDINATES XC(I)	 YC(I)	 ZC(I)	PANEL SURFACE AREA S(I)
1		0.0 0.0 -1.000000 -1.000000	0.0 0.0 0.0 -1.000000	0.0 0.0 -1.000000 0.0	-0.666667	-0.333333	-0.333333	0.866024
4	(THIS IS RIGHT SIDE IMAGE OF PANEL 1)	0.0 0.0 -1.000000 -1.000000	0.0 0.0 1.000000 0.0	0.0 0.0 0.0 -1.000000	-0.666667	0.333333	-0.333333	0.866024
2		0.0 0.0 -1.000000 -1.000000	0.0 0.0 -1.000000 0.0	0.0 0.0 0.0 1.000000	-0.666667	-0.333333	0.333333	0.866024
3	(THIS IS RIGHT SIDE IMAGE OF PANEL 2)	0.0 0.0 -1.000000 -1.000000	0.0 0.0 0.0 1.000000	0.0 0.0 1.000000 0.0	-0.666667	0.333333	0.333333	0.866024
5		-1.000000 -1.000000 -2.000000 -2.000000	0.0 -1.000000 0.0 0.0	-1.000000 0.0 0.0 0.0	-1.333333	-0.333333	-0.333333	0.866024
8	(THIS IS RIGHT SIDE IMAGE OF PANEL 5)	-1.000000 -1.000000 -2.000000 -2.000000	1.000000 0.0 0.0 0.0	0.0 -1.000000 0.0 0.0	-1.333333	0.333333	-0.333333	0.866024
6		-1.000000 -1.000000 -2.000000 -2.000000	-1.000000 0.0 0.0 0.0	0.0 1.000000 0.0 0.0	-1.333333	-0.333333	0.333333	0.866024
7	(THIS IS RIGHT SIDE IMAGE OF PANEL 6)	-1.000000 -1.000000 -2.000000 -2.000000	0.0 1.000000 0.0 0.0	1.000000 0.0 0.0 0.0	-1.333333	0.333333	0.333333	0.866024
		-2.250000 -2.250000	0.0 0.0	0.0 0.0				

9		-3.250000 -3.250000	0.0 -1.000000	-1.000000 0.0	-2.916666	-0.333333	-0.333333	0.866024
12	(THIS IS RIGHT SIDE IMAGE OF PANEL 9)	-2.250000 -2.250000 -3.250000 -3.250000	0.0 0.0 1.000000 0.0	0.0 0.0 0.0 -1.000000	-2.916666	0.333333	-0.333333	0.866024
10		-2.250000 -2.250000 -3.250000 -3.250000	0.0 0.0 -1.000000 0.0	0.0 0.0 0.0 1.000000	-2.916666	-0.333333	0.333333	0.866024
11	(THIS IS RIGHT SIDE IMAGE OF PANEL 10)	-2.250000 -2.250000 -3.250000 -3.250000	0.0 0.0 0.0 1.000000	0.0 0.0 1.000000 0.0	-2.916666	0.333333	0.333333	0.866024
13		-3.250000 -3.250000 -4.250000 -4.250000	0.0 -1.000000 0.0 0.0	-1.000000 0.0 0.0 0.0	-3.583333	-0.333333	-0.333333	0.866024
16	(THIS IS RIGHT SIDE IMAGE OF PANEL 13)	-3.250000 -3.250000 -4.250000 -4.250000	1.000000 0.0 0.0 0.0	0.0 -1.000000 0.0 0.0	-3.583333	0.333333	-0.333333	0.866024
14		-3.250000 -3.250000 -4.250000 -4.250000	-1.000000 0.0 0.0 0.0	0.0 1.000000 0.0 0.0	-3.583333	-0.333333	0.333333	0.866024
15	(THIS IS RIGHT SIDE IMAGE OF PANEL 14)	-3.250000 -3.250000 -4.250000 -4.250000	0.0 1.000000 0.0 0.0	1.000000 0.0 0.0 0.0	-3.583333	0.333333	0.333333	0.866024

//// 16 TOTAL PANELS IN BODY GENERATED SUCCESSFULLY. ////
//// END OF BODY PANEL GEOMETRY LISTING. MOVE ON TO FLOW CALCULATIONS. ////

****

```
**** SOLUTION OF SOURCE STRENGTHS, SIGMA ****

SPECIFIED MAXIMUM PERMITTED SOLUTION ITERATIONS=        25
SPECIFIED MAXIMUM PERMITTED SOLUTION ERROR      =    1.E-04

                    /// ITERATION   1 ///

   ALPHA, BETA,-CASE                              SOLUTION ERROR

 0.0       ,      0.0       , 1                      1.31439E 00
10.0000000,       0.0       , 2                      1.44090E 00

                    /// ITERATION   2 ///

   ALPHA, BETA,-CASE                              SOLUTION ERROR

 0.0       ,      0.0       , 1                      4.18229E-01
10.0000000,       0.0       , 2                      3.84574E-01

                    /// ITERATION   3 ///

   ALPHA, BETA,-CASE                              SOLUTION ERROR

 0.0       ,      0.0       , 1                      1.16396E-01
10.0000000,       0.0       , 2                      1.10683E-01

                    /// ITERATION   4 ///

   ALPHA, BETA,-CASE                              SOLUTION ERROR

 0.0       ,      0.0       , 1                      3.50483E-02
10.0000000,       0.0       , 2                      3.05387E-02

                    /// ITERATION   5 ///

   ALPHA, BETA,-CASE                              SOLUTION ERROR

 0.0       ,      0.0       , 1                      3.71265E-03
10.0000000,       0.0       , 2                      3.57056E-03

                    /// ITERATION   6 ///

   ALPHA, BETA,-CASE                              SOLUTION ERROR

 0.0       ,      0.0       , 1                      1.24162E-03
10.0000000,       0.0       , 2                      1.17987E-03

                    /// ITERATION   7 ///

   ALPHA, BETA,-CASE                              SOLUTION ERROR

 0.0       ,      0.0       , 1                      2.70128E-04
10.0000000,       0.0       , 2                      2.44200E-04

                    /// ITERATION   8 ///

   ALPHA, BETA,-CASE                              SOLUTION ERROR

0.0        ,      0.0       , 1                      3.33786E-05  (CASE HAS CONVERGED)
```

```
      10.0000000,     0.0      , 2             3.14713E-05  (CASE HAS CONVERGED)
--- SOURCE STRENGTH SOLUTIONS CONVERGED FOR ALL CASES WITHIN    8 ITERATIONS ---
```

```
**********************************************
*  BODY SURFACE VELOCITIES AND PRESSURES     *
*    CASE  1 - ALPHA =      0.0   DEGREES    *
*              BETA  =      0.0   DEGREES    *
**********************************************
```

MAGNITUDE OF FREESTREAM VELOCITY   V =   1.0000000
COMPONENTS OF FREESTREAM VELOCITY VX=   -1.0000000
VY=   0.0
VZ=   0.0

TOTAL NUMBER OF PANEL ELEMENTS =   16

PANEL	CONTROL POINT COORDINATES XC	YC	ZC	SOURCE STRENGTH SIGMA	NORMALIZED COMPONENTS AND RESULTANT VELOCITY AT CONTROL POINT VRX/V	VRY/V	VRZ/V	VR/V	PRESSURE COEFFICIENT CP
1	-0.6667	-0.3333	-0.3333	0.97258073E 00	-1.0270	-0.4062	-0.4476	1.1917	-0.4201
2	-0.6667	-0.3333	0.3333	0.12145882E 01	-0.9628	-0.4992	0.4636	1.1794	-0.3910
3	-0.6667	0.3333	0.3333	0.12145882E 01	-0.9628	0.4992	0.4636	1.1794	-0.3910
4	-0.6667	0.3333	-0.3333	0.97258073E 00	-1.0270	0.4062	-0.4476	1.1917	-0.4201
5	-1.3333	-0.3333	-0.3333	-0.11167336E 01	-0.9262	0.4584	0.4678	1.1344	-0.2868
6	-1.3333	-0.3333	0.3333	-0.11680355E 01	-0.9470	0.4721	-0.4749	1.1598	-0.3452
7	-1.3333	0.3333	0.3333	-0.11680355E 01	-0.9470	-0.4721	-0.4749	1.1598	-0.3452
8	-1.3333	0.3333	-0.3333	-0.11167336E 01	-0.9262	-0.4584	0.4678	1.1344	-0.2868
9	-2.9167	-0.3333	-0.3333	0.11725006E 01	-0.9524	-0.4753	-0.4771	1.1665	-0.3607
10	-2.9167	-0.3333	0.3333	0.11403294E 01	-0.9339	-0.4686	0.4653	1.1438	-0.3083
11	-2.9167	0.3333	0.3333	0.11403294E 01	-0.9339	0.4686	0.4653	1.1438	-0.3083
12	-2.9167	0.3333	-0.3333	0.11725006E 01	-0.9524	0.4753	-0.4771	1.1665	-0.3607
13	-3.5833	-0.3333	-0.3333	-0.12135124E 01	-0.9699	0.4986	0.4713	1.1881	-0.4115
14	-3.5833	-0.3333	0.3333	-0.99388343E 00	-1.0292	0.4136	-0.4423	1.1941	-0.4259
15	-3.5833	0.3333	0.3333	-0.99388343E 00	-1.0292	-0.4136	-0.4423	1.1941	-0.4259
16	-3.5833	0.3333	-0.3333	-0.12135124E 01	-0.9699	-0.4986	0.4713	1.1881	-0.4115

```
*****************************************************************************************
```

```
**********************************************************
*        PART 1 : PROPELLER PLANE OUTPUT                 *
*    VELOCITIES AT POINTS IN THE PROPELLER PLANE         *
*        CASE  1 - ALPHA =     0.0   DEGREES             *
*                  BETA  =     0.0   DEGREES             *
*                  CL    =     0.110                     *
**********************************************************
```

MAGNITUDE OF FREESTREAM VELOCITY V = 1.0000000
COMPONENTS OF FREESTREAM VELOCITY VX= -1.0000000
VY= 0.0
VZ= 0.0
INLET FLOW RATIO SPECIFIED
ON FIRST THROUGH-FLOW PANEL = 0.1000000 (POS. IS INFLOW, NEG. IS OUTFLOW)

PROPELLER PLANE DATA (PART 1) :

ALL VELOCITY COMPONENTS AND PROP PLANE ANGLES - RELATIVE TO BODY (AIRCRAFT) AXIS SYSTEM.

COORDINATES OF PROP HUB RELATIVE TO BODY AXES- XHUB= 0.2500000
YHUB= 0.0
ZHUB= 0.0

REFERENCE BLADE RADIUS (RADIUS) = 1.50000
ALPHA OF PROP (ALPHP) = 0.0 DEGREES - (POSITIVE FOR A THRUST COMPONENT TOWARD NEG. Z)
BETA OF PROP (BETAP) = 0.0 DEGREES - (POSITIVE FOR A THRUST COMPONENT TOWARD POS. Y)

*NOTE : AZIMUTH DEFINITION FOR PROPELLER PLANE AT ALPHP=BETAP= 0.0 :
BLADE AZIMUTH ANGLE IS DEFINED 0 DEGREES UPWARD (ALONG NEGATIVE Z) ;
INCREASES CLOCKWISE ASVIEWED IN POSITIVE X (NOSE FOREWARD) DIRECTION.

POINT POSITION IN TERMS OF PROPELLER AXES		POINT COORDINATES IN BODY-CARTESIAN AXIS SYSTEM			NORMALIZED COMPONENTS AND RESULTANT RESULTANT VELOCITY AT THE POINT RELATIVE TO BODY FIXED (AIRCRAFT) AXIS SYSTEM			
RADIAL POSITION NON-DIMENSIONAL RLOCAL/RADIUS	AZIMUTH POSITION (DEGREES) PSI	POINTX	POINTY	POINTZ	VTOTX/V	VTOTY/V	VTOTZ/V	VTOT/V
0.0	0.0	0.250000	0.0	0.0	-0.8469	0.0	-0.0111	0.8470
0.100	0.0	0.250000	0.0	-0.150000	-0.8536	0.0	-0.0302	0.8542
0.200	0.0	0.250000	0.0	-0.300000	-0.8657	0.0	-0.0475	0.8670
0.300	0.0	0.250000	0.0	-0.450000	-0.8828	0.0	-0.0612	0.8850
0.400	0.0	0.250000	0.0	-0.600000	-0.9037	0.0	-0.0699	0.9064
0.500	0.0	0.250000	0.0	-0.750000	-0.9255	0.0000	-0.0725	0.9283
0.600	0.0	0.250000	0.0	-0.899999	-0.9455	0.0000	-0.0700	0.9481
0.700	0.0	0.250000	0.0	-1.049999	-0.9621	0.0000	-0.0640	0.9642
0.800	0.0	0.250000	0.0	-1.199999	-0.9744	0.0000	-0.0563	0.9765
0.900	0.0	0.250000	0.0	-1.349998	-0.9842	-0.0000	-0.0485	0.9854
1.000	0.0	0.250000	0.0	-1.499998	-0.9907	-0.0000	-0.0411	0.9915
0.0	120.000	0.250000	0.0	0.0	-0.8469	0.0	-0.0111	0.8470
0.100	120.000	0.250000	0.129904	0.075000	-0.8478	0.0172	-0.0010	0.8482
0.200	120.000	0.250000	0.259808	0.149999	-0.8547	0.0338	0.0090	0.8554
0.300	120.000	0.250000	0.389711	0.224999	-0.8680	0.0484	0.0178	0.8695
0.400	120.000	0.250000	0.519615	0.299999	-0.8868	0.0591	0.0245	0.8891
0.500	120.000	0.250000	0.649519	0.374999	-0.9086	0.0642	0.0285	0.9113
0.600	120.000	0.250000	0.779423	0.449999	-0.9303	0.0639	0.0297	0.9330
0.700	120.000	0.250000	0.909327	0.524998	-0.9493	0.0597	0.0285	0.9516
0.800	120.000	0.250000	1.039229	0.599998	-0.9644	0.0523	0.0260	0.9662

0.900	120.000	0.250000	1.169133	0.674998	-0.9757	0.0463	0.0229	0.9771
1.000	120.000	0.250000	1.299037	0.749997	-0.9834	0.0395	0.0197	0.9448
0.0	240.000	0.250000	0.0	0.0	-0.8469	0.0	-0.0111	0.8473
0.100	240.000	0.250000	-0.129904	0.075000	-0.8478	-0.0172	-0.0010	0.8430
0.200	240.000	0.250000	-0.259807	0.150000	-0.8547	-0.0738	0.0090	0.8554
0.300	240.000	0.250000	-0.389711	0.225000	-0.8680	-0.0484	0.0178	0.8695
0.400	240.000	0.250000	-0.519615	0.300000	-0.8868	-0.0551	0.0245	0.8891
0.500	240.000	0.250000	-0.649518	0.375000	-0.9086	-0.0642	0.0285	0.9113
0.600	240.000	0.250000	-0.779422	0.450000	-0.9303	-0.0636	0.0297	0.9333
0.700	240.000	0.250000	-0.909726	0.525000	-0.9493	-0.0597	0.0285	0.9516
0.800	240.000	0.250000	-1.039228	0.600000	-0.9644	-0.0533	0.0260	0.9662
0.900	240.000	0.250000	-1.169132	0.675000	-0.9757	-0.0463	0.0229	0.9771
1.000	240.000	0.250000	-1.299035	0.750000	-0.9838	-0.0395	0.0197	0.9849

```
*******************************************************
*      PART 2 : PROPELLER PLANE OUTPUT                *
*   AXIAL, RADIAL, AND TANGENTIAL VELOCITIES          *
*              AND FLOW ANGLES                        *
*      CASE  1 - ALPHA =      0.0   DEGREES           *
*                BETA  =      0.0   DEGREES           *
*                CL    =      0.110                   *
*******************************************************
```

INLET FLOW RATIO SPECIFIED
ON FIRST THROUGH-FLOW PANEL = 0.1000000 (POS. IS INFLOW, NEG. IS OUTFLOW)

PROPELLER PLANE DATA (PART 2) :

*NOTE : A RIGHT HAND ROTATING PROPELLER IS ASSUMED FOR
SIGN CONVENTION ON TANGENTIAL VELOCITIES BELOW:

ALL VELOCITIES LISTED BELOW - RELATIVE TO PROPELLER PLANE CYLINDRICAL AXES
ANGLES OF ROTATIONAL FLOW AND OUTFLOW - RELATIVE TO PROPELLER PLANE CYLINDRICAL AXES
UPWASH AND SIDEWASH ANGLES - RELATIVE TO BODY (AIRCRAFT) AXIS SYSTEM

NORMALIZED AXIAL VELOCITY (VAXIAL/V) : POSITIVE - IN DIRECTION OF PROPELLER THRUST
NORMALIZED RADIAL VELOCITY (VRADAL/V) : POSITIVE - FROM HUB TO TIP
NORMALIZED TANGENTIAL VELOCITY (VTANG/V) : POSITIVE - WITH DIRECTION OF LOCAL BLADE ROTATION
ANGLE OF ROTATIONAL FLOW (ROTANG)
ANGLE OF OUTFLOW (OUTANG)
UPWASH ANGLE (UPWASH) : ANGLE AS SEEN IN BODY AXIS X-Z PLANE
SIDEWASH ANGLE (SIDWSH) : ANGLE AS SEEN IN BODY AXIS X-Y PLANE

REFERENCE BLADE RADIUS (RADIUS) = 1.50000
ALPHA OF PROP (ALPHP) = 0.0 DEGREES - (POSITIVE FOR A THRUST COMPONENT TOWARD NEG. Z)
BETA OF PROP (BETAP) = 0.0 DEGREES - (POSITIVE FOR A THRUST COMPONENT TOWARD POS. Y)

*NOTE : AZIMUTH DEFINITION FOR PROPELLER PLANE AT ALPHP=BETAP= 0.0 :
BLADE AZIMUTH ANGLE IS DEFINED 0 DEGREES UPWARD (ALONG NEGATIVE Z) &
INCREASES CLOCKWISE AS VIEWED IN POSITIVE X (NOSE FORWARD) DIRECTION.

POINT POSITION IN TERMS OF PROPELLER AXES		NORMALIZED VELOCITY COMPONENTS IN PROPELLER CYLINDRICAL AXES			FLOW DIRECTION ANGLES			
RADIAL POSITION NON-DIMENSIONAL RLOCAL/RADIUS	AZIMUTH POSITION (DEGREES) PSI	VAXIAL/V	VRADAL/V	VTANG/V	ROTANG (DEGREES)	OUTANG (DEGREES)	UPWASH (DEGREES)	SIDWSH (DEGREES)
0.0	0.0	-0.8469	0.0111	0.0	0.0	0.7504	0.7504	0.0
0.100	0.0	-0.8536	0.0302	0.0	0.0	2.0268	2.0268	0.0
0.200	0.0	-0.8657	0.0473	0.0	0.0	3.1306	3.1306	0.0
0.300	0.0	-0.8828	0.0612	0.0	0.0	3.9639	3.9639	0.0
0.400	0.0	-0.9037	0.0699	0.0	0.0	4.4202	4.4202	0.0
0.500	0.0	-0.9255	0.0725	0.0000	0.0000	4.4813	4.4813	0.0000
0.600	0.0	-0.9455	0.0703	0.0000	0.0000	4.2331	4.2331	0.0000
0.700	0.0	-0.9621	0.0649	0.0000	0.0000	3.8044	3.8044	0.0000
0.800	0.0	-0.9749	0.0563	0.0000	0.0000	3.3082	3.3082	0.0000
0.900	0.0	-0.9842	0.0485	-0.0000	-0.0000	2.8184	2.8184	-0.0000
1.000	0.0	-0.9907	0.0411	-0.0000	-0.0000	2.3740	2.3740	-0.0000
0.0	120.000	-0.8469	-0.0055	-0.0096	-0.6499	-0.3752	0.7504	0.0000
0.100	120.000	-0.8478	0.0144	-0.0094	-0.6358	0.9725	0.0643	1.1600
0.200	120.000	-0.8547	0.0338	-0.0091	-0.6126	2.2619	-0.6009	2.2650
0.300	120.000	-0.8680	0.0508	-0.0088	-0.5827	3.3522	-1.1732	3.1944
0.400	120.000	-0.8868	0.0634	-0.0083	-0.5348	4.0901	-1.5850	3.8099
0.500	120.000	-0.9086	0.0699	-0.0074	-0.4661	4.3959	-1.7980	4.0408
0.600	120.000	-0.9303	0.0702	-0.0063	-0.3872	4.3147	-1.8255	3.9311
0.700	120.000	-0.9493	0.0660	-0.0052	-0.3114	3.9745	-1.7203	3.5985
0.800	120.000	-0.9644	0.0592	-0.0042	-0.2467	3.5104	-1.5434	3.1640
0.900	120.000	-0.9757	0.0515	-0.0033	-0.1953	3.0209	-1.3425	2.7142
1.000	120.000	-0.9838	0.0440	-0.0027	-0.1559	2.5621	-1.1467	2.2971
0.0	240.000	-0.8469	-0.0055	0.0096	0.6499	-0.3752	0.7504	-0.0000
0.100	240.000	-0.8478	0.0144	0.0094	0.6358	0.9725	0.0643	-1.1600
0.200	240.000	-0.8547	0.0338	0.0091	0.6126	2.2619	-0.6010	-2.2650
0.300	240.000	-0.8680	0.0508	0.0088	0.5827	3.3522	-1.1732	-3.1944
0.400	240.000	-0.8868	0.0634	0.0083	0.5348	4.0901	-1.5850	-3.8099
0.500	240.000	-0.9086	0.0699	0.0074	0.4661	4.3959	-1.7981	-4.0408
0.600	240.000	-0.9303	0.0702	0.0063	0.3872	4.3147	-1.8255	-3.9311
0.700	240.000	-0.9493	0.0660	0.0052	0.3114	3.9745	-1.7203	-3.5985
0.800	240.000	-0.9644	0.0592	0.0042	0.2467	3.5104	-1.5434	-3.1640
0.900	240.000	-0.9757	0.0515	0.0033	0.1953	3.0209	-1.3425	-2.7142
1.000	240.000	-0.9838	0.0440	0.0027	0.1559	2.5621	-1.1467	-2.2971

****

```
**************************************************
*  BODY SURFACE VELOCITIES AND PRESSURES          *
*    CASE  2 - ALPHA =     10.000 DEGREES         *
*              BETA  =      0.0   DEGREES         *
**************************************************
```

MAGNITUDE OF FREESTREAM VELOCITY V = 1.0000000
COMPONENTS OF FREESTREAM VELOCITY VX= -0.9848078
VY= 0.0
VZ= -0.1736481

TOTAL NUMBER OF PANEL ELEMENTS = 16

PANEL	CONTROL POINT COORDINATES XC	YC	ZC	SOURCE STRENGTH SIGMA	NORMALIZED COMPONENTS AND RESULTANT VELOCITY AT CONTROL POINT VRX/V	VRY/V	VRZ/V	VR/V	PRESSURE COEFFICIENT CP
1	-0.6667	-0.3333	-0.3333	0.60754889E 00	-1.0627	-0.2629	-0.6266	1.2614	-0.5911
2	-0.6667	-0.3333	0.3333	0.13727903E 01	-0.8216	-0.5641	0.2575	1.0294	-0.0597
3	-0.6667	0.3333	0.3333	0.13727903E 01	-0.8216	0.5641	0.2575	1.0294	-0.0597
4	-0.6667	0.3333	-0.3333	0.60754889E 00	-1.0627	0.2629	-0.6266	1.2614	-0.5911
5	-1.3333	-0.3333	-0.3333	-0.10514755E 01	-0.7823	0.4328	0.3495	0.9599	0.0786
6	-1.3333	-0.3333	0.3333	-0.95695859E 00	-0.9562	0.3746	-0.5816	1.1802	-0.3928
7	-1.3333	0.3333	0.3333	-0.95695859E 00	-0.9562	-0.3746	-0.5816	1.1802	-0.3928
8	-1.3333	0.3333	-0.3333	-0.10514755E 01	-0.7823	-0.4328	0.3495	0.9599	0.0786
9	-2.9167	-0.3333	-0.3333	0.10023565E 01	-1.0151	-0.4091	-0.6060	1.2510	-0.5649
10	-2.9167	-0.3333	0.3333	0.12923098E 01	-0.8581	-0.5376	0.3204	1.0621	-0.1280
11	-2.9167	0.3333	0.3333	0.12923098E 01	-0.8581	0.5376	0.3204	1.0621	-0.1280
12	-2.9167	0.3333	-0.3333	0.10023565E 01	-1.0151	0.4091	-0.6060	1.2510	-0.5649
13	-3.5833	-0.3333	-0.3333	-0.13740759E 01	-0.8941	0.5606	0.3335	1.1068	-0.2250
14	-3.5833	-0.3333	0.3333	-0.81431252E 00	-1.0890	0.3347	-0.5812	1.2790	-0.6358
15	-3.5833	0.3333	0.3333	-0.81431252E 00	-1.0890	-0.3347	-0.5812	1.2790	-0.6358
16	-3.5833	0.3333	-0.3333	-0.13740759E 01	-0.8941	-0.5606	0.3335	1.1068	-0.2250

```
******************************************************************************************
```

```
**********************************************************
*        PART 1 : PROPELLER PLANE OUTPUT                 *
*   VELOCITIES AT POINTS IN THE PROPELLER PLANE          *
*     CASE  2 - ALPHA =     10.000 DEGREES               *
*               BETA  =      0.0   DEGREES               *
*               CL    =      1.096                       *
**********************************************************
```

```
MAGNITUDE OF FREESTREAM VELOCITY    V =     1.0000000
COMPONENTS OF FREESTREAM VELOCITY   VX=    -0.9848078
                                    VY=     0.0
                                    VZ=    -0.1736481
       INLET FLOW RATIO SPECIFIED
       ON FIRST THROUGH-FLOW PANEL    =     0.1000000 (POS. IS INFLOW, NEG. IS OUTFLOW)
```

PROPELLER PLANE DATA (PART 1) :

ALL VELOCITY COMPONENTS AND PROP PLANE ANGLES - RELATIVE TO BODY (AIRCRAFT) AXIS SYSTEM.

```
COORDINATES OF PROP HUB RELATIVE TO BODY AXES-  XHUB=   0.2500300
                                                YHUB=   0.0
                                                ZHUB=   0.0
REFERENCE BLADE RADIUS (RADIUS)  =   1.50000
      ALPHA OF PROP (ALPHP)      =   0.0     DEGREES - (POSITIVE FOR A THRUST COMPONENT TOWARD NEG. Z)
      BETA  OF PROP (BETAP)      =   0.0     DEGREES - (POSITIVE FOR A THRUST COMPONENT TOWARD POS. Y)
```

*NOTE : AZIMUTH DEFINITION FOR PROPELLER PLANE AT ALPHP=BETAP= 0.0 :
BLADE AZIMUTH ANGLE IS DEFINED 0 DEGREES UPWARD (ALONG NEGATIVE Z) :
INCREASES CLOCKWISE ASVIEWED IN POSITIVE X (NOSE FOREWARD) DIRECTION.

POINT POSITION IN TERMS OF PROPELLER AXES		POINT COORDINATES IN BODY-CARTESIAN AXIS SYSTEM			NORMALIZED COMPONENTS AND RESULTANT RESULTANT VELOCITY AT THE POINT RELATIVE TO BODY FIXED (AIRCRAFT) AXIS SYSTEM			
RADIAL POSITION NON-DIMENSIONAL RLOCAL/RADIUS	AZIMUTH POSITION (DEGREES) PSI	POINTX	POINTY	POINTZ	VTOTX/V	VTOTY/V	VTOTZ/V	VTOT/V
0.0	0.0	0.250000	0.0	0.0	-0.8365	-0.0000	-0.2212	0.8652
0.100	0.0	0.250000	0.0	-0.150000	-0.8535	-0.0000	-0.2383	0.8861
0.200	0.0	0.250000	0.0	-0.300000	-0.8738	0.0000	-0.2502	0.9089
0.300	0.0	0.250000	0.0	-0.450000	-0.8959	0.0000	-0.2571	0.9321
0.400	0.0	0.250000	0.0	-0.600000	-0.9184	0.0000	-0.2589	0.9542
0.500	0.0	0.250000	0.0	-0.750000	-0.9393	0.0000	-0.2559	0.9736
0.600	0.0	0.250000	0.0	-0.899999	-0.9573	0.0000	-0.2493	0.9892
0.700	0.0	0.250000	0.0	-1.049999	-0.9714	0.0000	-0.2406	1.0008
0.800	0.0	0.250000	0.0	-1.199999	-0.9819	0.0000	-0.2313	1.0087
0.900	0.0	0.250000	0.0	-1.349998	-0.9891	0.0000	-0.2223	1.0138
1.000	0.0	0.250000	0.0	-1.499998	-0.9938	-0.0000	-0.2141	1.0166
0.0	120.000	0.250000	0.0	0.0	-0.8365	-0.0000	-0.2212	0.8652
0.100	120.000	0.250000	0.129904	0.075000	-0.8319	0.0162	-0.2103	0.8583
0.200	120.000	0.250000	0.259808	0.149999	-0.8337	0.0326	-0.1581	0.8575
0.300	120.000	0.250000	0.389711	0.224999	-0.8426	0.0477	-0.1862	0.8643
0.400	120.000	0.250000	0.519615	0.299999	-0.8584	0.0593	-0.1754	0.8782
0.500	120.000	0.250000	0.649519	0.374999	-0.8787	0.0656	-0.1673	0.8969
0.600	120.000	0.250000	0.779423	0.449999	-0.8999	0.0663	-0.1624	0.9160
0.700	120.000	0.250000	0.909327	0.524998	-0.9191	0.0629	-0.1603	0.9351
0.800	120.000	0.250000	1.039229	0.599998	-0.9349	0.0570	-0.1602	0.9532

0.900	120.000	0.250000	1.169133	0.674998	-0.9471	0.0503	-0.1611	0.9620
1.000	120.000	0.250000	1.299037	0.749997	-0.9561	0.0437	-0.1625	0.9708
0.0	240.000	0.250000	0.0	0.0	-0.8365	-0.0000	-0.2212	0.8652
0.100	240.000	0.250000	-0.129904	0.075000	-0.8319	-0.0162	-0.2103	0.8543
0.200	240.000	0.250000	-0.259807	0.150000	-0.8337	-0.0326	-0.1983	0.8575
0.300	240.000	0.250000	-0.389711	0.225000	-0.8426	-0.0477	-0.1862	0.8643
0.400	240.000	0.250000	-0.519615	0.300000	-0.8584	-0.0593	-0.1754	0.8792
0.500	240.000	0.250000	-0.649518	0.375000	-0.8787	-0.0656	-0.1673	0.8969
0.600	240.000	0.250000	-0.779422	0.450000	-0.8999	-0.0663	-0.1624	0.9160
0.700	240.000	0.250000	-0.909326	0.525000	-0.9191	-0.0629	-0.1601	0.9351
0.800	240.000	0.250000	-1.039228	0.600000	-0.9349	-0.0570	-0.1602	0.9502
0.900	240.000	0.250000	-1.169132	0.675000	-0.9471	-0.0503	-0.1611	0.9620
1.000	240.000	0.250000	-1.299035	0.750000	-0.9561	-0.0437	-0.1625	0.9708

```
****************************************************
*        PART 2 : PROPELLER PLANE OUTPUT           *
*   AXIAL, RADIAL, AND TANGENTIAL VELOCITIES       *
*              AND FLOW ANGLES                     *
*     CASE  2 - ALPHA =    10.000 DEGREES          *
*               BETA  =     0.0   DEGREES          *
*               CL    =     1.096                  *
****************************************************
```

INLET FLOW RATIO SPECIFIED
ON FIRST THROUGH-FLOW PANEL = 0.1000000 (POS. IS INFLOW, NEG. IS OUTFLOW)

PROPELLER PLANE DATA (PART 2) :

*NOTE : A RIGHT HAND ROTATING PROPELLER IS ASSUMED FOR
SIGN CONVENTION ON TANGENTIAL VELOCITIES BELOW:

ALL VELOCITIES LISTED BELOW - RELATIVE TO PROPELLER PLANE CYLINDRICAL AXES
ANGLES OF ROTATIONAL FLOW AND OUTFLOW - RELATIVE TO PROPELLER PLANE CYLINDRICAL AXES
UPWASH AND SIDEWASH ANGLES - RELATIVE TO BODY (AIRCRAFT) AXIS SYSTEM

NORMALIZED AXIAL VELOCITY (VAXIAL/V) : POSITIVE - IN DIRECTION OF PROPELLER THRUST
NORMALIZED RADIAL VELOCITY (VRADAL/V) : POSITIVE - FROM HUB TO TIP
NORMALIZED TANGENTIAL VELOCITY (VTANG/V) : POSITIVE - WITH DIRECTION OF LOCAL BLADE ROTATION
ANGLE OF ROTATIONAL FLOW (ROTANG)
ANGLE OF OUTFLOW (OUTANG)
UPWASH ANGLE (UPWASH) : ANGLE AS SEEN IN BODY AXIS X-Z PLANE
SIDEWASH ANGLE (SIDWSH) : ANGLE AS SEEN IN BODY AXIS X-Y PLANE

REFERENCE BLADE RADIUS (RADIUS) = 1.50000
ALPHA OF PROP (ALPHP) = 0.0 DEGREES - (POSITIVE FOR A THRUST COMPONENT TOWARD NEG. Z)
BETA OF PROP (BETAP) = 0.0 DEGREES - (POSITIVE FOR A THRUST COMPONENT TOWARD POS. Y)

*NOTE : AZIMUTH DEFINITION FOR PROPELLER PLANE AT ALPHP=BETAP= 0.0 :
BLADE AZIMUTH ANGLE IS DEFINED 0 DEGREES UPWARD (ALONG NEGATIVE Z) ;
INCREASES CLOCKWISE AS VIEWED IN POSITIVE X (NOSE FORWARD) DIRECTION.

POINT POSITION IN TERMS OF PROPELLER AXES		NORMALIZED VELOCITY COMPONENTS IN PROPELLER CYLINDRICAL AXES			FLOW DIRECTION ANGLES			
RADIAL POSITION NON-DIMENSIONAL RLOCAL/RADIUS	AZIMUTH POSITION (DEGREES) PSI	VAXIAL/V	VRADAL/V	VTANG/V	ROTANG (DEGREES)	OUTANG (DEGREES)	UPWASH (DEGREES)	SIDWSH (DEGREES)
0.0	0.0	-0.8365	0.2212	-0.0000	-0.0000	14.8112	14.8112	-0.0000
0.100	0.0	-0.8535	0.2383	-0.0000	-0.0000	15.6011	15.6011	-0.0000
0.200	0.0	-0.8738	0.2507	0.0000	0.0000	15.9811	15.9811	0.0000
0.300	0.0	-0.8959	0.2571	0.0000	0.0000	16.0135	16.0135	0.0000
0.400	0.0	-0.9184	0.2589	0.0000	0.0000	15.7442	15.7442	0.0000
0.500	0.0	-0.9393	0.2559	0.0000	0.0000	15.2405	15.2405	0.0000
0.600	0.0	-0.9573	0.2493	0.0000	0.0000	14.5982	14.5982	0.0000
0.700	0.0	-0.9714	0.2406	0.0000	0.0000	13.9127	13.9127	0.0000
0.800	0.0	-0.9819	0.2313	0.0000	0.0000	13.2546	13.2546	0.0000
0.900	0.0	-0.9891	0.2223	0.0000	0.0000	12.6642	12.6642	0.0000
1.000	0.0	-0.9938	0.2141	-0.0000	-0.0000	12.1572	12.1572	-0.0000
0.0	120.000	-0.8365	-0.1106	-0.1915	-12.8981	-7.5314	14.8112	-0.0000
0.100	120.000	-0.8319	-0.0911	-0.1903	-12.8824	-6.2519	14.1894	1.1159
0.200	120.000	-0.8337	-0.0709	-0.1880	-12.7097	-4.8622	13.3794	2.2391
0.300	120.000	-0.8426	-0.0518	-0.1851	-12.3882	-3.5156	12.4587	3.2405
0.400	120.000	-0.8584	-0.0363	-0.1816	-11.9429	-2.4245	11.5493	3.9522
0.500	120.000	-0.8787	-0.0269	-0.1777	-11.4338	-1.7515	10.7824	4.2687
0.600	120.000	-0.8999	-0.0238	-0.1739	-10.9344	-1.5128	10.2325	4.2165
0.700	120.000	-0.9191	-0.0257	-0.1703	-10.4970	-1.6026	9.8956	3.9137
0.800	120.000	-0.9349	-0.0307	-0.1672	-10.1413	-1.8814	9.7221	3.4900
0.900	120.000	-0.9471	-0.0370	-0.1647	-9.8641	-2.2369	9.6546	3.0403
1.000	120.000	-0.9561	-0.0434	-0.1626	-9.6522	-2.5993	9.6473	2.6179
0.0	240.000	-0.8365	-0.1106	0.1915	12.8981	-7.5314	14.8112	-0.0000
0.100	240.000	-0.8319	-0.0911	0.1903	12.8824	-6.2519	14.1894	-1.1159
0.200	240.000	-0.8337	-0.0709	0.1880	12.7097	-4.8622	13.3794	-2.2391
0.300	240.000	-0.8426	-0.0518	0.1851	12.3882	-3.5157	12.4587	-3.2405
0.400	240.000	-0.8584	-0.0363	0.1816	11.9429	-2.4246	11.5493	-3.9522
0.500	240.000	-0.8787	-0.0269	0.1777	11.4338	-1.7515	10.7824	-4.2687
0.600	240.000	-0.8999	-0.0238	0.1739	10.9343	-1.5128	10.2325	-4.2165
0.700	240.000	-0.9191	-0.0257	0.1703	10.4969	-1.6026	9.8956	-3.9137
0.800	240.000	-0.9349	-0.0307	0.1672	10.1413	-1.8814	9.7221	-3.4900
0.900	240.000	-0.9471	-0.0370	0.1647	9.8641	-2.2369	9.6545	-3.0403
1.000	240.000	-0.9561	-0.0434	0.1626	9.6522	-2.5993	9.6473	-2.6179

*** END OF JOB - ALL PARTS COMPLETED SUCCESSFULLY ***