



# NAVAL POSTGRADUATE SCHOOL

## Monterey, California



# THESIS

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

by

Jack Paschall III

December 1983

Thesis Advisor:

R. D. Zucker

Approved for Public Release, Distribution Unlimited

T215666

## REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS  
BEFORE COMPLETING FORM

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |  |                       |                                                                         |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|-----------------------|-------------------------------------------------------------------------|
| 1. REPORT NUMBER                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |  | 2. GOVT ACCESSION NO. | 3. RECIPIENT'S CATALOG NUMBER                                           |
| 4. TITLE (and Subtitle)<br>A Three-Dimensional Transonic, Potential Flow<br>Computer Program, Its Conversion to IBM<br>Fortran and Utilization                                                                                                                                                                                                                                                                                                                                                                   |  |                       | 5. TYPE OF REPORT & PERIOD COVERED<br>Master's Thesis;<br>December 1983 |
| 7. AUTHOR(s)<br>Jack Paschall III                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |  |                       | 6. PERFORMING ORG. REPORT NUMBER                                        |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS<br>Naval Postgraduate School<br>Monterey, California 93943                                                                                                                                                                                                                                                                                                                                                                                                           |  |                       | 8. CONTRACT OR GRANT NUMBER(s)                                          |
| 11. CONTROLLING OFFICE NAME AND ADDRESS<br>Naval Postgraduate School<br>Monterey, California 93943                                                                                                                                                                                                                                                                                                                                                                                                               |  |                       | 10. PROGRAM ELEMENT, PROJECT, TASK<br>AREA & WORK UNIT NUMBERS          |
| 14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)                                                                                                                                                                                                                                                                                                                                                                                                                                      |  |                       | 12. REPORT DATE<br>December 1983                                        |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |  |                       | 13. NUMBER OF PAGES<br>152                                              |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |  |                       | 15. SECURITY CLASS. (of this report)<br>UNCLASSIFIED                    |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |  |                       | 15a. DECLASSIFICATION/DOWNGRADING<br>SCHEDULE                           |
| 16. DISTRIBUTION STATEMENT (of this Report)<br><br>Approved for Public Release, Distribution Unlimited                                                                                                                                                                                                                                                                                                                                                                                                           |  |                       |                                                                         |
| 17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)                                                                                                                                                                                                                                                                                                                                                                                                                       |  |                       |                                                                         |
| 18. SUPPLEMENTARY NOTES                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |  |                       |                                                                         |
| 19. KEY WORDS (Continue on reverse side if necessary and identify by block number)<br><br>Computer Program; Inviscid, Three-Dimensional Transonic Potential Flow<br>Over Wings or Wing-Body Combinations                                                                                                                                                                                                                                                                                                         |  |                       |                                                                         |
| 20. ABSTRACT (Continue on reverse side if necessary and identify by block number)<br><br>This thesis describes the conversion of a computer program from<br>Fortran IV for the NOS 1.2 operating system of the CYBER 175 or CDC<br>6600 computer to Fortran IV compatible with the Naval Postgraduate<br>School IBM 3033 system. The converted program, called FLO27,<br>calculates the inviscid, three-dimensional transonic potential flow<br>over wings or wing-body combinations. The data input to FLO27 is |  |                       |                                                                         |

extensive; therefore, an interactive program was developed to aid the user in building the required input data file.

Approved for Public Release, Distribution Unlimited

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

by

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

Submitted in partial fulfillment of the  
requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL  
December 1983

---

## ABSTRACT

DTIC 12-10-1975  
NAVAL POSTGRADUATE SCHOOL  
MONTEREY, CALIF. 93943

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

## TABLE OF CONTENTS

|      |                                                  |     |
|------|--------------------------------------------------|-----|
| I.   | INTRODUCTION . . . . .                           | 9   |
|      | A. BACKGROUND . . . . .                          | 9   |
|      | B. VISCOUS/INVISCID SYSTEM OF PROGRAMS . . . . . | 10  |
| II.  | POTENTIAL FLOW PROGRAM FLO27 . . . . .           | 14  |
|      | A. RE-PROGRAMING . . . . .                       | 14  |
|      | B. PROGRAM DESCRIPTION . . . . .                 | 19  |
|      | 1. Program Input . . . . .                       | 22  |
|      | 2. Program Output . . . . .                      | 32  |
| III. | INTERACTIVE INPUT PROGRAM FLO27IN . . . . .      | 35  |
| IV.  | FLO27 BATCH SYSTEM EXECUTION . . . . .           | 38  |
| V.   | PROGRAM TEST RESULTS . . . . .                   | 40  |
|      | A. ACCEPTANCE TEST DATA . . . . .                | 40  |
|      | B. COMPARISON WITH OTHER PROGRAMS . . . . .      | 41  |
|      | C. AE-4501 CLASS PROJECT . . . . .               | 41  |
|      | APPENDIX A . . . . .                             | 44  |
|      | APPENDIX B . . . . .                             | 46  |
|      | APPENDIX C . . . . .                             | 47  |
|      | APPENDIX D . . . . .                             | 48  |
|      | APPENDIX E . . . . .                             | 70  |
|      | APPENDIX F . . . . .                             | 132 |
|      | LIST OF REFERENCES . . . . .                     | 151 |
|      | INITIAL DISTRIBUTION LIST . . . . .              | 152 |

## LIST OF TABLES

|      |                                                  |    |
|------|--------------------------------------------------|----|
| I.   | Viscous/Inviscid Wing System of Programs . . . . | 11 |
| II.  | CDC Magnetic Tape Files . . . . .                | 15 |
| III. | FLO27 Re-Programing Changes . . . . .            | 16 |
| IV.  | Initialized Input Variables . . . . .            | 18 |

## LIST OF FIGURES

|      |                                                   |    |
|------|---------------------------------------------------|----|
| 1.1. | Viscous/Inviscid Interaction Procedure . . . . .  | 12 |
| 2.1. | Dihedral Angle . . . . .                          | 27 |
| 2.2. | Wing Defining Geometry . . . . .                  | 28 |
| 2.3. | Section Defining Geometry . . . . .               | 31 |
| 2.4. | Versatec Plot of $C_p$ vs. $X/C$ . . . . .        | 34 |
| 5.1. | Program Calculated and Wind Tunnel Data . . . . . | 42 |

## ACKNOWLEDGEMENT

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

## I. INTRODUCTION

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

### A. BACKGROUND

In 1980 the Department of Aeronautics at the Naval Postgraduate School acquired an intricate computer program recently developed by the Boeing Commercial Airplane Company. This state-of-the-art program calculates three-dimensional transonic flow over wings and bodies in

both the outer-inviscid flow region governed by the transonic potential equation and the thin layer in which the first order, compressible boundary layer equations are assumed to be valid.

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

## B. VISCOUS/INVISCID SYSTEM OF PROGRAMS

The Viscous/Inviscid Wing System (VIWS) of programs calculates three-dimensional transonic flow over wings and wing body combinations including details of the laminar or turbulent flow in the three-dimensional viscous boundary layer. The flow field is calculated in two overlapping regions: an outer inviscid flow region governed by the transonic potential equation, and a thin boundary layer in which the first order, three-dimensional, compressible

boundary layer equations are assumed to hold and in which the effects of surface heat and mass transfer can be computed. A list of the VIWS of programs is presented in Table I.

TABLE I  
Viscous/Inviscid Wing System of Programs

| Program Name               | Description                                                  |
|----------------------------|--------------------------------------------------------------|
| F1027                      | Jameson-Caughey inviscid, transonic wing code                |
| A411IN                     | Reads geometry & velocity data, constructs coordinate system |
| VWIN                       | Potential flow boundary layer interface                      |
| A411AC1                    | Three-dimensional boundary layer program                     |
| INTERP                     | Boundary layer potential flow interface                      |
| A411F1<br>A411F2<br>A411FS | Graphics display programs                                    |

The basic sequence of calculations used by the VIWS to obtain matched viscous and inviscid solutions consists of an iterative loop in which the inviscid outer flow analysis and the boundary layer analysis are performed sequentially. The iterative sequence is continued until either convergence (satisfactory matching) is achieved, or the maximum number of iterations specified by the user has been performed. The VIWS programming sequence is shown schematically in Fig. 1.1.

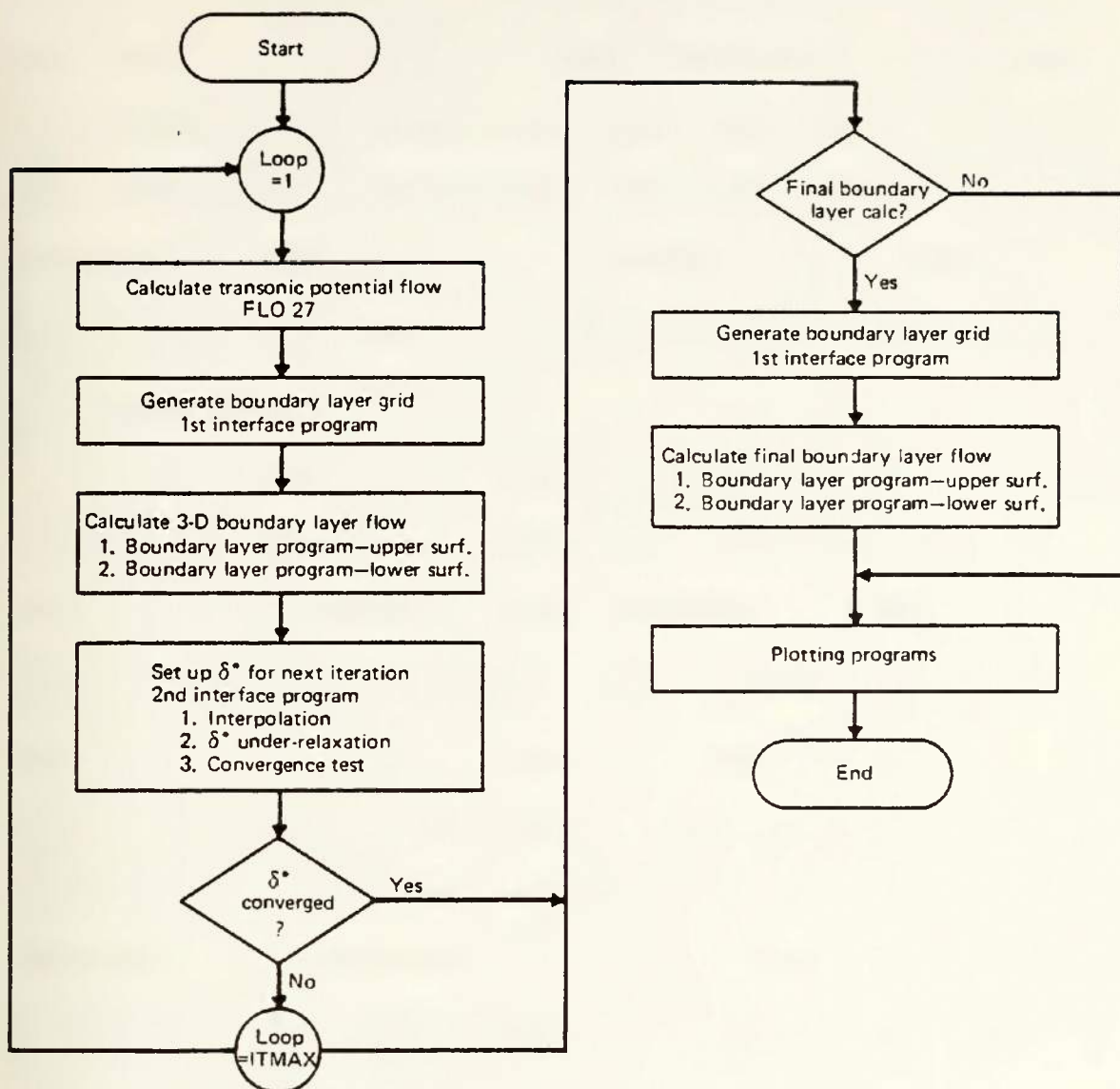


**Figure 1.1. Viscous/Inviscid Interaction Procedure**

The potential flow is calculated for the bare wing during the first iteration. In subsequent iterations, the effect of the boundary layer flow on the outer inviscid flow is

felt as a modification to the wing shape through the addition of the boundary layer displacement thickness. Convergence is recognized, and the iterations are stopped, when the maximum change between the new and old displacement thickness, expressed as a fraction of the maximum displacement thickness, is less than the convergence tolerance chosen by the user.

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

## II. POTENTIAL FLOW PROGRAM FLO27

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

### A. RE-PROGRAMING

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

The FLO27 program was converted to FORTRAN IV extended suitable for execution on the IBM computer using the NPS CDC

TABLE II  
CDC Magnetic Tape Files

| File/Records | Name    | Description                                                    |
|--------------|---------|----------------------------------------------------------------|
| 1 /2356      | FL027   | Potential Flow Program                                         |
| 2 /3194      | A411IN  | Reads geometry & velocity data<br>constructs coordinate system |
| 3 / 378      | VWIN    | Potential flow boundary-layer<br>interface                     |
| 4 /6611      | A411AO1 | Three-dimensional boundary<br>layer program                    |
| 5 /1977      | INTERP  | Boundary-layer potential flow<br>interface                     |
| 6 / 688      | A411PS  | Streamline plots                                               |
| 7 / 211      | A411P1  | One-dimensional plots                                          |
| 8 / 586      | A411P2  | Contour plots                                                  |
| 9 / 70       | COUPLE  | Procedure files                                                |
| 10 / 158     | ITER    |                                                                |
| 11 / 7       | DATAIN  |                                                                |
| 12 / 78      | FINAL   |                                                                |
| 13 / 434     | BOEB1   | Boeing McLean computer program                                 |
| 14 / 36      | CONTPLT | Contour plots                                                  |
| 15 / 17      | CORDPLT | One-dimensional plots                                          |
| 16 / 40      | STREPLT | Streamwise plots                                               |
| 17           | OUTF27  | Output from FL027                                              |
| 18           | OUTIFC  | Output from VWIN                                               |
| 19           | OUT411L | Output from boundary-layer,<br>lower surface                   |
| 20           | OUT411U | Output from boundary-layer,<br>upper surface                   |

to IFM conversion guide [Ref. 6]. The first step taken consisted of program compilation using the WATFIV compiler with its extended error messages. The listing which was

produced flagged all areas of the program which required revision. Program changes were accomplished utilizing this WATFIV listing. Some of the more general and repetitive changes are listed in Table III.

TABLE III

FLO27 Re-Programming Changes

| CDC Code                                                          | IBM Code Change                           |
|-------------------------------------------------------------------|-------------------------------------------|
| Variables: FREAD, FREAF, FWRIT, FWRIF, IREAD, IREAF, IWRIT, IWRIF | Eliminated from program                   |
| WRITE(IWRIT,600)                                                  | WRITE(6,600)                              |
| READ(IREAF,500)                                                   | READ(5,500)                               |
| READ 7, WRITE 7 or REWIND 7                                       | Changed to READ 14, WRITE 14 or REWIND 14 |
| Call SECCND(T)                                                    | Step eliminated                           |
| Call SSWITCH(1,ISTOP)                                             | Call SLITET(1,ISTOP)                      |
| Delimiter of form *                                               | Replaced by '                             |
| Comment cards with *                                              | Replaced by C                             |
| LEVEL statement                                                   | Step eliminated                           |
| If(UNIT(N).GT.0.) GO TO                                           | All of this type eliminated               |

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

The change required to translate this statement is presented below with the CDC code preceeding the IBM FORTRAN.

```

EUFFER OUT (N3,1) (G(1,1,1),G(MX,MY,1)) changed to
WRITE(N3) ((G(I,J,1),I=1,MX),J=1,MY)      and
EUFFER IN (N1,1) (G(1,1,M),G(MX,MY,M)) changed to
READ(N1,ERR= ) ((G(I,J,M),I=1,MX),J=1,MY)

```

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

In addition to the program changes required to run FLO27 on the IBM computer, several lines of code were added to modify the output format to a more usable form. A subroutine, VERTEC, which calls the Versatec plotter was also added to enhance program usefulness. This plotting routine is user controlled through an input variable and is explained in the next section. The modified FLO27 program source code is presented in Appendix E.

To facilitate program data entry several input variables which had recommended values were initialized to these values within the Main program and the subroutine GEOM. The initialized input variables and their values are presented in Table IV.

TABLE IV  
Initialized Input Variables

| AREA           | VARIABLE NAME | INITIALIZED VALUE |
|----------------|---------------|-------------------|
| MAIN<br>Prgrm. | XSCAL         | 0.0               |
|                | PSCAL         | 0.0               |
|                | FCONT         | 0.0               |
|                | P20           | 0.7               |
|                | P30           | 1.0               |
|                | FSMCO         | 0.0               |
|                | PTMAP         | 0.0               |
|                | BLCP          | 0.0               |
|                | WEIG          | 1.0               |
|                | PTCK          | 0.0               |
|                | FIX           | 0.0               |
|                | YSYM          | 0.0               |
| Subrt.<br>GEOM | FNB           | 2.0               |
|                | PX            | 0.0               |
|                | PZ            | 0.0               |
|                | TRL           | 0.0               |
|                | SLT           | 0.0               |
|                | XSING         | 0.0               |
|                | YSING         | 0.0               |

A complete description of each input variable in Table IV can be found on pages 19 through 23 of [Ref. 5].

## B. PROGRAM DESCRIPTION

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

$$\vec{V} = u\hat{i} + v\hat{j} + w\hat{k} \quad (2.1)$$

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

$$\frac{\partial}{\partial x}(\rho u) + \frac{\partial}{\partial y}(\rho v) + \frac{\partial}{\partial z}(\rho w) = 0 \quad (2.2)$$

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

$$u = \phi_x, \quad v = \phi_y, \quad w = \phi_z \quad (2.3)$$

With the introduction of the velocity potential, the continuity equation 2.2 becomes

$$\frac{\partial}{\partial x}(\rho\phi_x) + \frac{\partial}{\partial y}(\rho\phi_y) + \frac{\partial}{\partial z}(\rho\phi_z) = 0 \quad (2.4)$$

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

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

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

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

With equations 2.5 and 2.6 there are only two unknowns,  $\phi$  and  $\rho$ . They can be solved, subject to the boundary condition of flow tangency, using a finite volume technique. The basic numerical scheme for the solution is the

construction of a mesh from small volume elements (cubes) which are packed around the wing or wing body configuration. The cubes in the computational domain are separately mapped to distorted cubes in the physical domain by independent transformations from local coordinates  $X$ ,  $Y$  and  $Z$  to Cartesian coordinates  $x$ ,  $y$  and  $z$ . The mesh points are the vertices (corners) of the mapped cubes. The velocity potential and density are calculated at each vertex in the mesh. The pressure distribution can then be calculated from

$$P = \frac{\rho^\sigma}{\gamma M_\infty^2} \quad (2.7)$$

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

- 1) The tangential velocity components are equal on each side of the shock.
- 2) Continuity is maintained by keeping the product of  $\rho U_n$  constant across the shock (where  $U_n$  is the normal velocity component).
- 3) Discontinuous expansions (corresponding to an "expansion shock") are excluded from the flow field.

The assumption of isentropic flow along with the existence of shocks presents a contradiction which can only be resolved by limiting the flow to very weak shocks for which entropy and vorticity generation may be ignored. Thus, solutions will be valid only for subsonic free stream velocities.

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

#### 1. Program Input

The input to FLO27 consists of variables which are read with an 8F10.6 FORMAT. Each input card has a title card which precedes it. This title card contains the input variable name and effectively labels the input data for easy

reference. The title for each input data is placed in the same column as the input data it labels. The title cards are read with a 20A4 FORMAT. All numerical input values are real numbers. The following data deck, listed card by card, is the minimum input data required for a simple wing analysis. Each "card" can be interpreted as one line of data on your terminal. A complete sample data set is presented in Appendix C.

CARD 1 The Run Title (64 characters maximum)

CARD 2 Title card for the input variables  
FNX, FNY, FNZ, FMESH and FPLOT

CARD 3

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

$MAX = 160/2^{**}n$ , where  $n = FMESH - 1$ . (See Cols. 31-40 for FMESH)

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

$MAX = 16/2^{**}n$ , where  $n = FMESH - 1$ .

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

$MAX = 32/2^{**}n$ , where  $n = FMESH - 1$ .

Cols. 31-40 FMESH - Determines the number of times a program generated computational mesh is refined. Enter only 1.0, 2.0 or 3.0 for coarse, medium or fine mesh. If 3.0 is selected the program will calculate flow over the wing for the coarse mesh then half the mesh size (medium), recalculate, then half the mesh again (fine) and do a final potential flow calculation. Output parameters are printed for each mesh size for which calculations were performed.

Cols. 41-50 FELOT - Output flag

0.0 = Normal output without printer-plot of  $C_p$

1.0 = Normal output with printer-plot of  $C_p$   
at each computational mesh point for  
each wing section.

2.0 = Normal output with Versatec plots of  
 $C_p$  versus  $X/C$  for each wing section of  
the final mesh.

CARD 4 Title card for the input variables

FIT, COVO and P10

CARD 5-M One card for each computational mesh. Total  
number of cards equal to  $M = FMESH$ .

Cols. 1-10 FIT - A parameter which fixes the maximum number of iterations the program will use to converge the velocity potential to a specified tolerance (COVO). This parameter must be repeated for each mesh refinement.

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

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

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

CARD 7

Cols. 1-10 FMACH - Free stream Mach number

Cols. 11-20 YA - Yaw angle in degrees

Cols. 21-30 AL - Angle of attack in degrees

Cols. 31-40 CD0 - Drag coefficient due to skin friction. Unless known, an estimated value of 0.01 is recommended.

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

CARD 9

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

0.0 = Yawed wing, has no spanwise symmetry

1.0 = Swept wing, has spanwise symmetry

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

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

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

Cols. 41-50 FUS - Input the fuselage radius. Enter 0.0 for a wing-alone case.

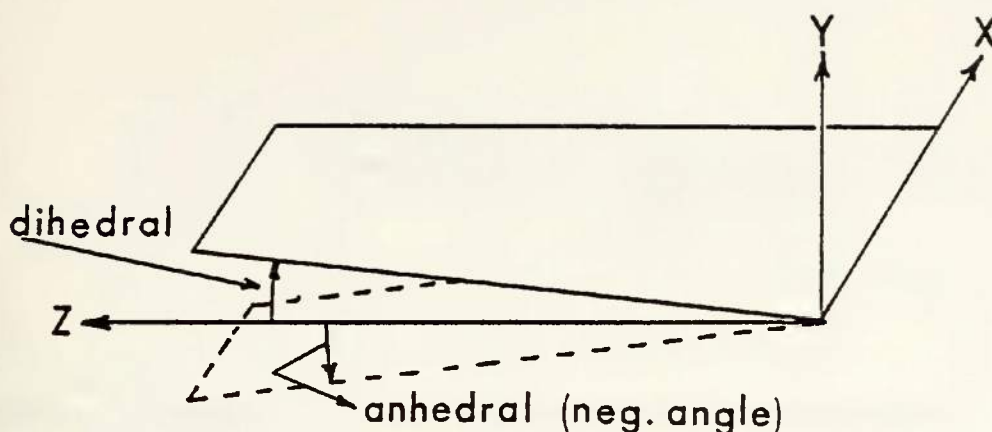


Figure 2.1. Dihedral Angle

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

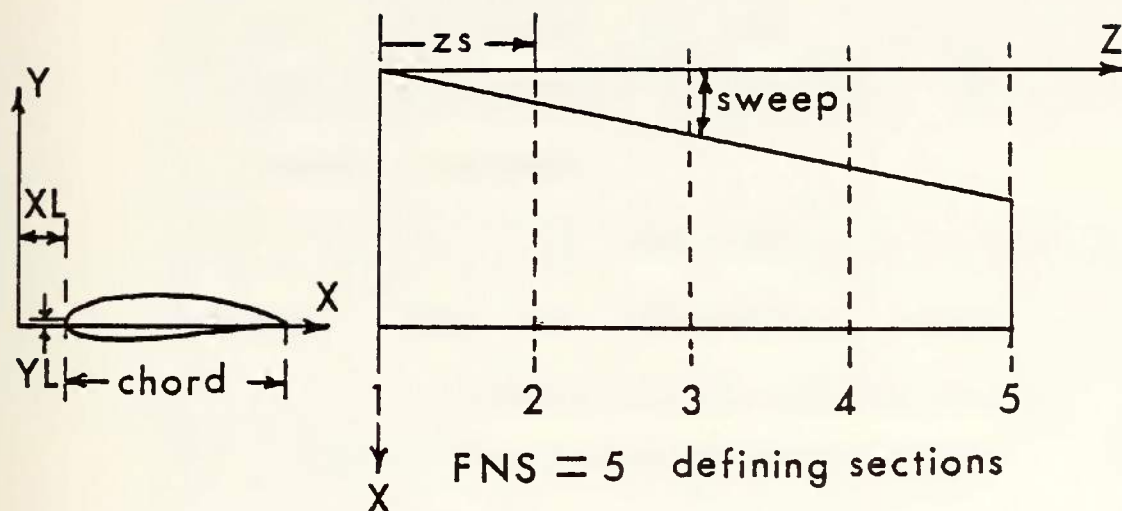


Figure 2.2. Wing Defining Geometry

CARD 10 Title card for the input variables

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

CARD 11

Cols. 1-10 ZS - The section spanwise coordinate

(Start at the centerline and work  
outboard)

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

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

Cols. 31-40 CHORD - Section chord length

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

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

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

CARD 12 Title card for the input variable

FN

CARD 13

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

CARD 14 Title card for the input variables

XP(I) and YP(I)

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

where  $N = \text{integer part of } (FN+2)/3$ .

The X and Y coordinates at each point are entered in pairs, three points to a card. (See Appendix C for sample input)

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

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

21-30 defining X coordinate for next  
point

31-40 defining Y coordinate for next  
point

41-50 defining X coordinate for  
following point

defining Y coordinate for  
following point

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

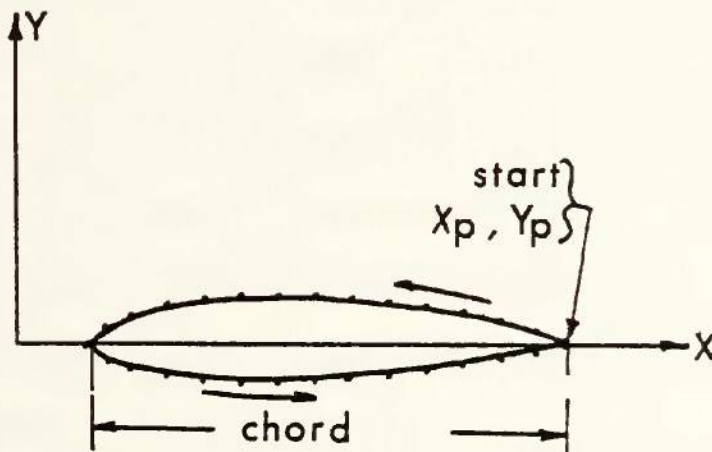


Figure 2.3. Section Defining Geometry

CARD 16 Title card containing the words in Cols. 1-80

END OF CALCULATIONS

CARD 17 Title card for the input variable

FNX

CARD 18

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

## 2. Program Output

Output from the FLO27 program varies with the value of the input variable FPLCT. When FPLCT is set equal to 0.0 a normal output is produced. This normal output contains (in order of occurrence): refined input geometry data including trailing edge slope and angle calculations; iterative solution of the potential flow mesh; section characteristics and wing characteristics. The iterative solution, section and wing characteristic data are repeated for each mesh refinement requested. Thus, if the input variable FMESH is set equal to 3, these data are calculated and output three times. The last data in the normal output consists of the non-dimensionalized chord ( $X/C$ ) and pressure

coefficient ( $C_p$ ) data at each computational mesh point for each wing section calculated during the final mesh. A sample of the normal output data is presented in Appendix D and represents the output data from Appendix C input data.

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

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

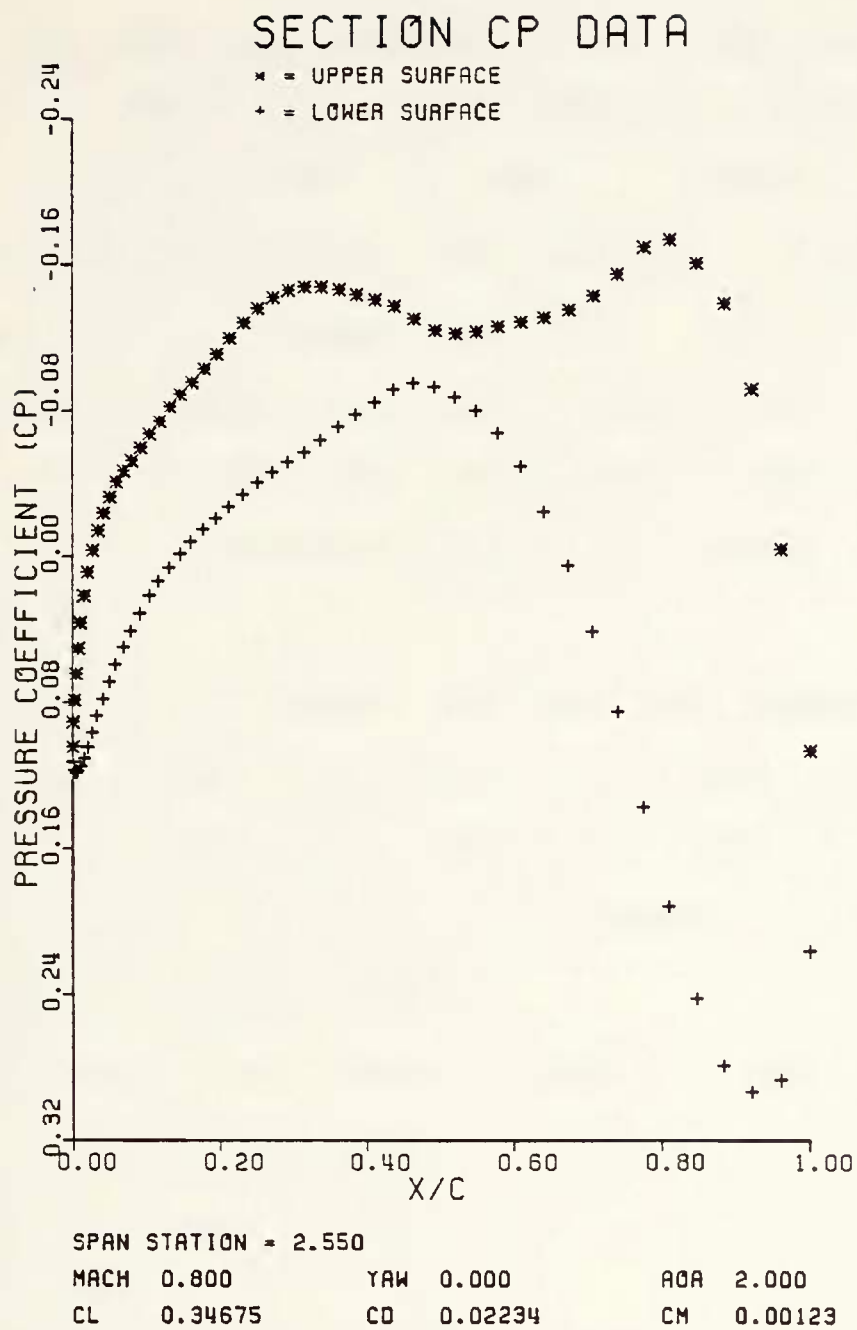


Figure 2.4. Versatec Plot of  $C_p$  vs.  $X/C$

### III. INTERACTIVE INPUT PROGRAM FLO27IN

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

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

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

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

CP LINK 0247P 191 120 RR then hit ENTER

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

AERO

STEP #4---Type and ENTER

ACC 120 D

STEP #5---Type and ENTER

LOAD FLO27IN (START

The screen will display the header for the interactive program. Answer each question presented. At the end of each question in parenthesis is the input data variable associated with that question and whether the input parameter is a real number (R) or an integer (I). Example:  
==> Enter the free stream Mach number (FMACH): (R). FMACH is the input data variable for the question. As you proceed through the FLO27IN program, opportunities to review and change input data will be presented. Should it become necessary to change your input data after completing the FLO27IN program, you can simply XEDIT the created data file.

The FLO27IN program also incorporates a library which contains the wing-section defining data for a number of current wing shapes. A copy of this library is presented in Appendix B. This feature will be displayed during program execution by the use of a menu from which the user can select a pre-defined wing section or define his own.

Upon completion of user inputs to the interactive program three additional data lines are automatically written to the bottom of the input file. They are:

END OF CALCULATION

FNX

0.0

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

#### IV. FLO27 BAICH SYSTEM EXECUTION

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

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

jobname = may contain up to 8 alphanumeric characters,

the first of which must be alphabetic.

nnnn = your user number

pppp = project number, assigned by professor

'ident' = contains the user's own identification

information. A maximum of 20 characters may be contained within the single quotation marks.

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

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

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

where nnnn = your user number

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

## V. PROGFAM TEST RESULTS

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

### A. ACCEPTANCE TEST DATA

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

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

decided that the IBM single precision accuracy was satisfactory.

#### B. COMPARISON WITH OTHER PROGRAMS

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

#### C. AE-4501 CLASS PROJECT

The final test phase was conducted by introducing the FLO27 program into the AE-4501 course as a class project. This was accomplished to determine student problems/comments concerning the data input program FLO27IN and to test an additional wing shape. The wing chosen for study was that of the A-7 airplane. The A-7 wing has a distinct leading edge notch at the approximate mid-span. When the planform geometry was run with the notch included the FLO27 program

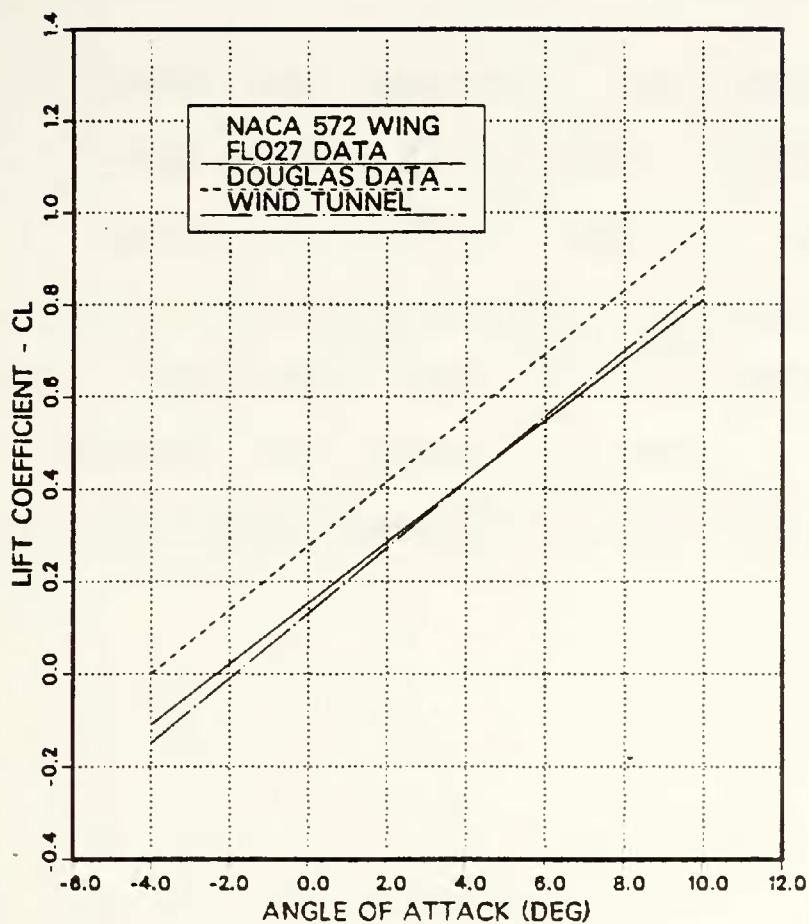


Figure 5.1. Program Calculated and Wind Tunnel Data

ran to completion but gave negative values for section and total induced drag coefficient. The value for the lift coefficient was low for the freestream Mach and angle-of-attack used. It was found that if the notch was

excluded from the wing geometry input data the program results were satisfactory both for induced drag and lift coefficient.

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

## APPENDIX A

```
C This JCL routine allocates sufficient space on the mass
C storage system to store the entire tape contents
//JACK JOB (3266,0178), 'PASCHALL-2759', CLASS=A
//*MAIN ORG=NEGV M1.3266P
// EXEC PGM=IEFBR14
//DD1 DD UNIT=3330V,MSVGP=PUB4C,DISP=(NEW,CATLG),
// DSN=MSS.S3266.WFLOW.DATA,SPACE=(CYL,(16,4,2))
/*
//
```

```
C This JCL routine is used to transfer all tape files to
C a partitioned data set in the mass storage system
//JACK JOB (3266,0178), 'PASCHALL-2759', CLASS=J
//*MAIN ORG=NPGVM1.3266P
//COPY PROC FILE=,MEM=
// EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD UNIT=3400-6,VCL=SER=WFLOW,DISP=(OLD,PASS),
// LABEL=(&FILE,BLP,,IN)
// DCB=(RECFM=F,BLKSIZE=80,DEN=3,OPTCD=Q)
//SYSUT2 DD DISP=(OLD,KEEP),
// DSN=MSS.S3266.WFLOW.SOURCE(&MEM),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=6400)
// PEND
// EXEC COPY,FILE=1,MEM=FLO27
// EXEC COPY,FILE=2,MEM=A411IN
// EXEC COPY,FILE=3,MEM=VWIN
// EXEC COPY,FILE=4,MEM=A411A01
// EXEC COPY,FILE=5,MEM=INTERF
// EXEC COPY,FILE=6,MEM=A411PS
// EXEC COPY,FILE=7,MEM=A411P1
// EXEC COPY,FILE=8,MEM=A411P2
// EXEC COPY,FILE=9,MEM=COUPLE
// EXEC COPY,FILE=10,MEM=ITER
// EXEC COPY,FILE=11,MEM=DATAIN
// EXEC COPY,FILE=12,MEM=FINAL
// EXEC COPY,FILE=13,MEM=BOEB1
// EXEC COPY,FILE=14,MEM=CONTPLT
// EXEC COPY,FILE=15,MEM=CORDPLT
// EXEC COPY,FILE=16,MEM=STREPLT
//COPY2 PROC FILE=,MEM=,LRECL=80,BLK=6400
// EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD UNIT=3400-6,VOL=SER=WFLOW,DISP=(OLD,PASS),
// LABEL=(&FILE,BLP,,IN)
// DCB=(RECFM=F,BLKSIZE=&LRECL,DEN=3,OPTCD=Q)
//SYSUT2 DD DISP=(OLD,KEEP),DSN=MSS.S3266.WFLOW.DATA(&MEM),
// DCB=(RECFM=FB,LRECL=&LRECL,BLKSIZE=&BLK)
// PEND
// EXEC COPY2,FILE=17,LRECL=150,BLK=6000,MEM=OUTF27
// EXEC COPY2,FILE=18,LRECL=150,BLK=6000,MEM=OUTIFC
// EXEC COPY2,FILE=19,LRECL=150,BLK=6000,MEM=OUT411L
// EXEC COPY2,FILE=20,LRECL=150,BLK=6000,MEM=OUT411U
/*
//
```

```

C This JCL routine moves all source code files from mass
C storage to the MVS 004 disk which can be accessed by
C entering GET MVS then following the screen instructions
C to move source files to your disk. If you want to move
C the data files to MVS 004 then change the word SOURCE
C to DATA in the JCL program below.
//JACK JOB (3266,0178), 'FASCHALL-2759', CLASS=A
//*MAIN ORG=NPGVM1.3266P
// EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//FROM DD DISP=SHR, DSN=MSS.S3266.WFLOW.SOURCE
//INTO DD UNIT=3350, VOL=SER=MVS004, DISP=(NEW,KEEP),
// SPACE=(CYL,(16,4,10),RLSE), DSN=S3266.SOURCE
//SYSUT3 DD UNIT=SYSDA, SPACE=(CYL,(2,2))
//SYSUT4 DD UNIT=SYSDA, SPACE=(CYL,(2,2))
//SYSIN DD *
COPY OUTDD=INTC, INDD=FFCM
/*
//

```

## APPENDIX B

### LIBRARY OF AIRFOIL SECTION GEOMETRIES

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

# APPENDIX C

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

// (STANDARD JOB CARD - SEE MVS USER'S GUIDE NO. MVS-01)

// EXEC FLC27

// GO.SYSIN CC \*

SAMPLE DATA (NACA 572 WING SECTION)

FNX 40.0 FNY 4.0 C FZ 8.0 P10 1.6 P11 1.3 P12 1.2 AT 0.0

FIT 100.0 C

20.0 C

5.0 C

FMACH 0.10

ZSYM 1.0

ZS 0.0 C

XL 0.0

XP(1) 1.000000

0.800000

0.500000

0.250000

0.100000

0.050000

0.025000

0.012500

0.006250

0.003125

0.001562

0.000781

0.000391

0.000195

0.000098

0.000049

0.000024

0.000012

0.000006

0.000003

0.000001

0.000000

FPLOT 0.0

FUS 0.0

THICK 1.0000

AT 0.0

FSEC 1.0

CHORD 8.6000

CD0 0.010000

DIHED 0.0

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

CHORD 8.6000

# APPENDIX D

THIS APPENDIX PRESENTS THE FLQ27 OUTPUT DATA PRODUCED FROM THE INPUT DATA OF THE PREVIOUS APPENDIX.

A46C MODIFIED FROM FLQ27 OF ANTCNY JAMESCN; CURANT INST ILUTE  
THREE DIMENSIONAL WING ANALYSIS IN TRANSONIC FLOW USING FINITE  
NACA 572 WING SECTION VOLUME SCHEME  
FUSELAGE RAL

0.0  
SWEEP  
32.3275  
CIHED  
0.0

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

NL = 21, XF(NL) = 0.0

(XP,YP)

|         |          |         |          |         |          |
|---------|----------|---------|----------|---------|----------|
| 1.00000 | -0.00130 | 0.95000 | -0.00480 | 0.90000 | -0.00820 |
| 0.60000 | -0.02600 | 0.50000 | -0.03200 | 0.40000 | -0.03580 |
| 0.20000 | -0.04220 | 0.15000 | -0.04200 | 0.10000 | -0.03750 |
| 0.03000 | -0.02270 | 0.02000 | -0.01680 | 0.01000 | -0.01200 |
| 0.03000 | 0.02900  | 0.00250 | 0.00500  | 0.00500 | 0.00700  |
| 0.20000 | 0.07260  | 0.04000 | 0.03500  | 0.05000 | 0.04140  |
| 0.60000 | 0.06420  | 0.25000 | 0.07680  | 0.30000 | 0.07880  |
| 1.00000 | 0.00130  | 0.70000 | 0.05150  | 0.80000 | 0.03760  |

SECTION DEFINITION AT Z = 0.0  
XLE  
0.0  
YMIN  
0.0  
YMAX  
1.0000  
JMAX  
33  
CHORD  
8.6000  
YMAX  
7880E-01  
THICKNESS RATIO  
1.0000  
JMAX  
33  
TWIST  
0.0  
YDIF  
0.1210

SECTION DEFINITION AT Z = 9.68750  
XLE  
0.0  
YMIN  
0.0  
YMAX  
1.0000  
JMAX  
33  
CHORD  
6.4500  
YMAX  
7880E-01  
THICKNESS RATIO  
1.0000  
JMAX  
33  
TWIST  
0.0  
YDIF  
0.1210

SECTION DEFINITION AT Z = 19.37500  
XLE  
0.0  
YMIN  
0.0  
YMAX  
1.0000  
JMAX  
33  
CHORD  
4.3000  
YMAX  
7880E-01  
THICKNESS RATIO  
1.0000  
JMAX  
33  
TWIST  
0.0  
YDIF  
0.1210

-0.4220E-01





# SECTION CHARACTERISTICS

MACH NO  
 0.10000  
 CL  
 20238  
 SPAN STATION  
 0.0 7499  
 3.8 7499  
 7.7 4999  
 11.6 2498  
 15.4 9998  
 19.3 7497  
 CL  
 24033  
 26515  
 28127  
 28695  
 24168

# WING CHARACTERISTICS

MACH NO  
 0.10000  
 CL  
 0.25575  
 CM YAW  
 -0.00467  
 CD FCRM  
 -0.32534E-02  
 CM ROLL  
 .23919

# ANG OF ATTACK

2.0000  
 CD  
 71088E-02  
 -0.24623E-02  
 -0.48115E-02  
 -0.57074E-02  
 -0.65451E-02  
 -0.70588E-02

# ANG CF ATTACK

2.0000  
 CD FRICTION  
 .10000E-01  
 CM PITCH  
 -0.33140

L/D FORM  
 -78.612  
 L/D  
 37.

52

# SECTION CHARACTERISTICS

|              |         |    |       |               |             |
|--------------|---------|----|-------|---------------|-------------|
| MACH NO      | 0.10000 | CL | 22093 | ANG GF ATTACK | 2.0000      |
| SPAN STATION | 0.0     | CL | 24319 | CD            | 1.0566      |
| 1.93750      |         |    | 26058 | CD            | 1.0667      |
| 3.81249      |         |    | 27508 | CD            | 1.0952      |
| 5.74595      |         |    | 28683 | CD            | 1.1248      |
| 7.68749      |         |    | 29643 | CD            | 1.1524      |
| 9.62499      |         |    | 30359 | CD            | 1.1750      |
| 11.56248     |         |    | 30863 | CD            | 1.1920      |
| 13.45998     |         |    | 30746 | CD            | 1.1986      |
| 15.43747     |         |    | 29141 | CD            | 1.1812      |
| 17.37497     |         |    | 21866 | CD            | 1.0990      |
|              |         |    |       | CD            | 1.78297E-01 |

# WING CHARACTERISTICS

|              |         |    |       |               |             |          |        |
|--------------|---------|----|-------|---------------|-------------|----------|--------|
| MACH NO      | 0.10000 | CL | 22093 | ANG GF ATTACK | 2.0000      | L/D      | 25.    |
| SPAN STATION | 0.0     | CL | 24319 | CD            | 1.0566      | L/D FORM | 317.42 |
| 1.93750      |         |    | 26058 | CD            | 1.0667      |          |        |
| 3.81249      |         |    | 27508 | CD            | 1.0952      |          |        |
| 5.74595      |         |    | 28683 | CD            | 1.1248      |          |        |
| 7.68749      |         |    | 29643 | CD            | 1.1524      |          |        |
| 9.62499      |         |    | 30359 | CD            | 1.1750      |          |        |
| 11.56248     |         |    | 30863 | CD            | 1.1920      |          |        |
| 13.45998     |         |    | 30746 | CD            | 1.1986      |          |        |
| 15.43747     |         |    | 29141 | CD            | 1.1812      |          |        |
| 17.37497     |         |    | 21866 | CD            | 1.0990      |          |        |
|              |         |    |       | CD            | 1.78297E-01 |          |        |

|        |              |           |      |        |
|--------|--------------|-----------|------|--------|
| MAX    | R            | ESIDAL    | J    | K      |
| -      | 0.198673E-03 | 120       | 16   | 433333 |
| 0.     | 0.38673E-03  | 160       | 15   |        |
| 0.     | 0.87373E-03  | 160       | 15   |        |
| 0.     | 0.73407E-03  | 160       | 15   |        |
| 0.     | 0.62240E-03  | 160       | 15   |        |
| CYCLE  | CONV         | TOLERANCE | E-05 |        |
| 0.8187 |              | 0.1000    |      |        |

# SECTION CHARACTERISTICS

MACH NO  
0.1000  
CL  
0.0

SPAN STATION

0.0 6875  
1.0 93750  
2.0 90625  
3.0 87429  
4.0 84374  
5.0 81249  
6.0 78124  
7.0 74999  
8.0 71874  
9.0 68749  
10.0 65624  
11.0 62499  
12.0 59373  
13.0 56248  
14.0 53123  
15.0 49998  
16.0 46872  
17.0 43747  
18.0 40622  
19.0 37497

## ANG OF ATTACK

2.0000

CD  
25222E-01  
13934E-01  
81939E-02  
57553E-02  
42226E-02  
34296E-02  
28453E-02  
24550E-02  
21906E-02  
16846E-02  
14920E-02  
12848E-02  
10850E-03  
83827E-03  
58364E-03  
32256E-03  
30467E-03  
14655E-03  
49591E-03  
68273E-04

CM  
11051  
10917  
10974  
11099  
11244  
11393  
11546  
11686  
11821  
11944  
12056  
12147  
12229  
12276  
12287  
12235  
12093  
12076  
11177  
95299E-01  
76890E-01

# WING CHARACTERISTICS

MACH NO  
0.1000  
CL  
0.0

CD FCRM-02  
38318E-02  
CM YAW  
0.00116  
CM RCLL  
25975

## ANG OF ATTACK

2.0000

CD FRICTICN  
10000E-01  
CM PITCH  
36670

L/D FORM  
73.379

L/D  
20.

A46C MODIFIED FROM FLC27 OF ANTICNY JAMESON, CCURANT INSTILUTE  
THREE DIMENSIONAL WING ANALYSIS IN TRANSONIC FLOW USING FINITE VOLUME SCHEME  
BOEING VERSION, PREPARED BY DR. HAI-CHOW CHEN STANDARD BOEING INPUT FORMAT  
END OF CALCULATION



|          |           |          |           |          |           |          |           |
|----------|-----------|----------|-----------|----------|-----------|----------|-----------|
| 0.035734 | -C.051941 | 0.043503 | -0.177332 | 0.052008 | -0.334910 | 0.061654 | -0.435076 |
| 0.072512 | -C.450556 | 0.084319 | -0.411025 | 0.096972 | -0.336700 | 0.110347 | -0.435076 |
| 0.124445 | -C.364866 | 0.139315 | -0.433279 | 0.155014 | -0.443650 | 0.171564 | -0.444444 |
| 0.188925 | -C.430921 | 0.207137 | -0.443005 | 0.226132 | -0.443181 | 0.275938 | -0.442164 |
| 0.266567 | -C.444812 | 0.288014 | -0.443053 | 0.310732 | -0.443822 | 0.333333 | -0.442164 |
| 0.357204 | -C.410345 | 0.381868 | -0.443557 | 0.407327 | -0.443501 | 0.433376 | -0.442164 |
| 0.460617 | -C.366049 | 0.488449 | -0.443557 | 0.517071 | -0.443501 | 0.546485 | -0.442164 |
| 0.576392 | -C.314431 | 0.607694 | -0.443557 | 0.639486 | -0.443501 | 0.672059 | -0.442164 |
| 0.705355 | -C.319805 | 0.739502 | -0.443557 | 0.774388 | -0.443501 | 0.810070 | -0.442164 |
| 0.846511 | -C.058370 | 0.883730 | -0.443557 | 0.921712 | -0.443501 | 0.960470 | -0.442164 |
| 1.000000 | -C.214168 |          |           |          |           |          |           |

SPAN STATION = 1.93750  
 NO. CF DATA POINTS = 101

|          |          |          |          |          |          |          |          |
|----------|----------|----------|----------|----------|----------|----------|----------|
| 1.000000 | C.235694 | 0.960452 | 0.206138 | 0.521777 | 0.159265 | 0.883857 | 0.100000 |
| 0.846730 | C.100136 | 0.810396 | 0.207724 | 0.664541 | 0.155583 | 0.740105 | 0.000000 |
| 0.706152 | C.024387 | 0.672858 | 0.202152 | 0.649055 | 0.002184 | 0.690654 | 0.000000 |
| 0.578298 | C.027180 | 0.543628 | 0.202249 | 0.410261 | 0.002184 | 0.385015 | 0.000000 |
| 0.463050 | C.023109 | 0.433684 | 0.202249 | 0.313940 | 0.002184 | 0.291351 | 0.000000 |
| 0.360527 | C.014254 | 0.325026 | 0.202249 | 0.230322 | 0.002184 | 0.211358 | 0.000000 |
| 0.270523 | C.019272 | 0.175832 | 0.202249 | 0.159106 | 0.002184 | 0.143458 | 0.000000 |
| 0.128543 | C.021484 | 0.114444 | 0.202249 | 0.101106 | 0.002184 | 0.088501 | 0.000000 |
| 0.076658 | C.021056 | 0.065638 | 0.202249 | 0.055508 | 0.002184 | 0.046334 | 0.000000 |
| 0.038057 | C.023664 | 0.030935 | 0.202249 | 0.024313 | 0.002184 | 0.018926 | 0.000000 |
| 0.013378 | C.025577 | 0.009362 | 0.202249 | 0.006189 | 0.002184 | 0.003363 | 0.000000 |
| 0.001528 | C.056265 | 0.003595 | 0.202249 | 0.001877 | 0.002184 | 0.000177 | 0.000000 |
| 0.012560 | C.042914 | 0.017041 | 0.202249 | 0.006189 | 0.002184 | 0.008618 | 0.000000 |
| 0.072512 | C.052309 | 0.043319 | 0.202249 | 0.022008 | 0.002184 | 0.028694 | 0.000000 |
| 0.124445 | C.041679 | 0.139316 | 0.202249 | 0.155014 | 0.002184 | 0.110348 | 0.000000 |
| 0.188925 | C.047219 | 0.207137 | 0.202249 | 0.226132 | 0.002184 | 0.171564 | 0.000000 |
| 0.266567 | C.047258 | 0.288015 | 0.202249 | 0.310732 | 0.002184 | 0.245939 | 0.000000 |
| 0.357204 | C.042291 | 0.381869 | 0.202249 | 0.407327 | 0.002184 | 0.333333 | 0.000000 |
| 0.460617 | C.036813 | 0.488450 | 0.202249 | 0.517071 | 0.002184 | 0.433376 | 0.000000 |
| 0.576392 | C.030606 | 0.607694 | 0.202249 | 0.639486 | 0.002184 | 0.546485 | 0.000000 |
| 0.705355 | C.018397 | 0.739503 | 0.202249 | 0.774388 | 0.002184 | 0.672059 | 0.000000 |
| 0.846511 | C.014443 | 0.883730 | 0.202249 | 0.921712 | 0.002184 | 0.810070 | 0.000000 |
| 1.000000 | C.021512 |          |          |          |          | 0.960470 | 0.000000 |

SPAN STATION = 2.90625  
 NO. CF DATA POINTS = 101

|          |          |          |          |          |          |          |          |
|----------|----------|----------|----------|----------|----------|----------|----------|
| 1.000000 | C.236239 | 0.960492 | 0.207084 | 0.521777 | 0.161171 | 0.883857 | 0.100000 |
| 0.846730 | C.102997 | 0.810356 | 0.208065 | 0.664541 | 0.005967 | 0.740105 | 0.000000 |
| 0.706152 | C.025155 | 0.672858 | 0.202574 | 0.410261 | 0.002184 | 0.609077 | 0.000000 |
|          |          |          |          |          |          |          |          |

[illegible]

SPAN STATION = 3.87499  
NO. OF DATA POINTS = 101

[illegible]

SPAN STATION = 4.84374  
 NO. OF DATA PCINTS = 101  
 X/C CP  
 1.000000 0.460617 -0.372240 0.488449 -0.356717 0.517071 -0.341032 0.546485 -0.323554  
 0.576692 0.301548 -0.301548 0.607694 0.272598 -0.272598 0.639466 0.672056 -0.202293  
 0.705355 -0.173943 -0.173943 0.739502 0.150774 -0.150774 0.774388 0.810059 -0.006466  
 0.846511 -0.036027 -0.036027 0.883730 0.014850 -0.014850 0.921712 0.960470 0.014686  
 1.000000 0.217029 0.217029

SPAN STATION = 4.84374  
 NO. OF DATA PCINTS = 101  
 X/C CP  
 1.000000 0.237192 0.237192 0.260000 0.0492 0.08446 0.08446 0.08446  
 0.846730 0.105858 0.105858 0.810255 0.083922 0.083922 0.774854 0.774854 0.774854  
 0.706152 0.033375 0.033375 0.672253 0.030518 0.030518 0.640641 0.640641 0.640641  
 0.578298 0.022889 0.022889 0.543628 0.006676 0.006676 0.519029 0.519029 0.519029  
 0.360050 0.018228 0.018228 0.336848 0.001394 0.001394 0.313941 0.313941 0.313941  
 0.270527 0.011685 0.011685 0.250026 0.006732 0.006732 0.230321 0.230321 0.230321  
 0.193235 0.013793 0.013793 0.175841 0.011732 0.011732 0.159224 0.159224 0.159224  
 0.128543 0.018605 0.018605 0.114441 0.016605 0.016605 0.101106 0.101106 0.101106  
 0.076658 0.020930 0.020930 0.065637 0.022967 0.022967 0.055507 0.055507 0.055507  
 0.038050 0.026198 0.026198 0.030635 0.032293 0.032293 0.024319 0.024319 0.024319  
 0.013378 0.049501 0.049501 0.009184 0.045640 0.045640 0.006159 0.006159 0.006159  
 0.001528 0.074111 0.074111 0.003595 0.069201 0.069201 0.00169 0.00169 0.00169  
 0.003528 0.174114 0.174114 0.007041 0.155512 0.155512 0.002208 0.002208 0.002208  
 0.007332 0.195525 0.195525 0.043503 0.186616 0.186616 0.035574 0.035574 0.035574  
 0.024444 0.613284 0.613284 0.084316 0.586730 0.586730 0.055012 0.055012 0.055012  
 0.188496 0.481388 0.481388 0.130713 0.429756 0.429756 0.102271 0.102271 0.102271  
 0.266567 0.510275 0.510275 0.288015 0.507422 0.507422 0.240732 0.240732 0.240732  
 0.357205 0.445434 0.445434 0.381869 0.426587 0.426587 0.317027 0.317027 0.317027  
 0.460618 0.330166 0.330166 0.487694 0.272257 0.272257 0.407321 0.407321 0.407321  
 0.576692 0.037450 0.037450 0.607694 0.037450 0.037450 0.639466 0.639466 0.639466  
 0.705355 0.037450 0.037450 0.739502 0.037450 0.037450 0.774388 0.774388 0.774388  
 0.846511 0.037450 0.037450 0.883730 0.037450 0.037450 0.921712 0.921712 0.921712  
 1.000000 0.217029 0.217029

SPAN STATION = 5.81249  
 NO. OF DATA PCINTS = 101  
 X/C CP  
 1.000000 0.237192 0.237192 0.260000 0.0492 0.08446 0.08446 0.08446  
 0.846730 0.105858 0.105858 0.810255 0.083922 0.083922 0.774854 0.774854 0.774854  
 0.706152 0.033375 0.033375 0.672253 0.030518 0.030518 0.640641 0.640641 0.640641  
 0.578298 0.022889 0.022889 0.543628 0.006676 0.006676 0.519029 0.519029 0.519029  
 0.360050 0.018228 0.018228 0.336848 0.001394 0.001394 0.313941 0.313941 0.313941  
 0.270527 0.011685 0.011685 0.250026 0.006732 0.006732 0.230321 0.230321 0.230321  
 0.193235 0.013793 0.013793 0.175841 0.011732 0.011732 0.159224 0.159224 0.159224  
 0.128543 0.018605 0.018605 0.114441 0.016605 0.016605 0.101106 0.101106 0.101106  
 0.076658 0.020930 0.020930 0.065637 0.022967 0.022967 0.055507 0.055507 0.055507  
 0.038050 0.026198 0.026198 0.030635 0.032293 0.032293 0.024319 0.024319 0.024319  
 0.013378 0.049501 0.049501 0.009184 0.045640 0.045640 0.006159 0.006159 0.006159  
 0.001528 0.074111 0.074111 0.003595 0.069201 0.069201 0.00169 0.00169 0.00169  
 0.003528 0.174114 0.174114 0.007041 0.155512 0.155512 0.002208 0.002208 0.002208  
 0.007332 0.195525 0.195525 0.043503 0.186616 0.186616 0.035574 0.035574 0.035574  
 0.024444 0.613284 0.613284 0.084316 0.586730 0.586730 0.055012 0.055012 0.055012  
 0.188496 0.481388 0.481388 0.130713 0.429756 0.429756 0.102271 0.102271 0.102271  
 0.266567 0.510275 0.510275 0.288015 0.507422 0.507422 0.240732 0.240732 0.240732  
 0.357205 0.445434 0.445434 0.381869 0.426587 0.426587 0.317027 0.317027 0.317027  
 0.460618 0.330166 0.330166 0.487694 0.272257 0.272257 0.407321 0.407321 0.407321  
 0.576692 0.037450 0.037450 0.607694 0.037450 0.037450 0.639466 0.639466 0.639466  
 0.705355 0.037450 0.037450 0.739502 0.037450 0.037450 0.774388 0.774388 0.774388  
 0.846511 0.037450 0.037450 0.883730 0.037450 0.037450 0.921712 0.921712 0.921712  
 1.000000 0.217029 0.217029







[illegible]

| SPAN NO. | STATION OF DATA | PGINTS = 101 | X/C | CP | X/C | CP | X/C | CP |
|----------|-----------------|--------------|-----|----|-----|----|-----|----|
| 1        | 000             | 23760        | 1   | 0  | 0   | 0  | 0   | 0  |
| 1        | 846             | 10926        | 4   | 0  | 0   | 0  | 0   | 0  |
| 0        | 706             | 03910        | 1   | 0  | 0   | 0  | 0   | 0  |
| 0        | 578             | 03106        | 3   | 0  | 0   | 0  | 0   | 0  |
| 0        | 298             | 00295        | 8   | 0  | 0   | 0  | 0   | 0  |
| 0        | 050             | 00758        | 3   | 0  | 0   | 0  | 0   | 0  |
| 0        | 360             | 02588        | 3   | 0  | 0   | 0  | 0   | 0  |
| 0        | 270             | 00956        | 3   | 0  | 0   | 0  | 0   | 0  |
| 0        | 193             | 11622        | 1   | 0  | 0   | 0  | 0   | 0  |
| 0        | 128             | 05441        | 1   | 0  | 0   | 0  | 0   | 0  |
| 0        | 766             | 05438        | 5   | 0  | 0   | 0  | 0   | 0  |
| 0        | 038             | 03668        | 2   | 0  | 0   | 0  | 0   | 0  |
| 0        | 133             | 21661        | 7   | 0  | 0   | 0  | 0   | 0  |
| 0        | 001             | 64672        | 7   | 0  | 0   | 0  | 0   | 0  |
| 0        | 015             | 41198        | 7   | 0  | 0   | 0  | 0   | 0  |
| 0        | 012             | 09877        | 3   | 0  | 0   | 0  | 0   | 0  |
| 0        | 035             | 27161        | 0   | 0  | 0   | 0  | 0   | 0  |
| 0        | 072             | 69170        | 3   | 0  | 0   | 0  | 0   | 0  |
| 0        | 184             | 53405        | 8   | 0  | 0   | 0  | 0   | 0  |
| 0        | 244             | 05676        | 1   | 0  | 0   | 0  | 0   | 0  |
| 0        | 188             | 54271        | 5   | 0  | 0   | 0  | 0   | 0  |
| 0        | 266             | 54271        | 5   | 0  | 0   | 0  | 0   | 0  |
| 0        | 357             | 46817        | 7   | 0  | 0   | 0  | 0   | 0  |
| 0        | 000             | 23760        | 1   | 0  | 0   | 0  | 0   | 0  |
| 0        | 846             | 10926        | 4   | 0  | 0   | 0  | 0   | 0  |
| 0        | 706             | 03910        | 1   | 0  | 0   | 0  | 0   | 0  |
| 0        | 578             | 03106        | 3   | 0  | 0   | 0  | 0   | 0  |
| 0        | 298             | 00295        | 8   | 0  | 0   | 0  | 0   | 0  |
| 0        | 050             | 00758        | 3   | 0  | 0   | 0  | 0   | 0  |
| 0        | 360             | 02588        | 3   | 0  | 0   | 0  | 0   | 0  |
| 0        | 270             | 00956        | 3   | 0  | 0   | 0  | 0   | 0  |
| 0        | 193             | 11622        | 1   | 0  | 0   | 0  | 0   | 0  |
| 0        | 128             | 05441        | 1   | 0  | 0   | 0  | 0   | 0  |
| 0        | 766             | 05438        | 5   | 0  | 0   | 0  | 0   | 0  |
| 0        | 038             | 03668        | 2   | 0  | 0   | 0  | 0   | 0  |
| 0        | 133             | 21661        | 7   | 0  | 0   | 0  | 0   | 0  |
| 0        | 001             | 64672        | 7   | 0  | 0   | 0  | 0   | 0  |
| 0        | 015             | 41198        | 7   | 0  | 0   | 0  | 0   | 0  |
| 0        | 012             | 09877        | 3   | 0  | 0   | 0  | 0   | 0  |
| 0        | 035             | 27161        | 0   | 0  | 0   | 0  | 0   | 0  |
| 0        | 072             | 69170        | 3   | 0  | 0   | 0  | 0   | 0  |
| 0        | 184             | 53405        | 8   | 0  | 0   | 0  | 0   | 0  |
| 0        | 244             | 05676        | 1   | 0  | 0   | 0  | 0   | 0  |
| 0        | 188             | 54271        | 5   | 0  | 0   | 0  | 0   | 0  |
| 0        | 266             | 54271        | 5   | 0  | 0   | 0  | 0   | 0  |
| 0        | 357             | 46817        | 7   | 0  | 0   | 0  | 0   | 0  |

|          |           |          |           |          |           |          |           |
|----------|-----------|----------|-----------|----------|-----------|----------|-----------|
| 0.460617 | -C.385346 | 0.488450 | -0.371584 | 0.517071 | -0.353830 | 0.546485 | -0.33443  |
| 0.576692 | -C.310566 | 0.607694 | -0.279852 | 0.639486 | -0.243425 | 0.672056 | -0.220707 |
| 0.705355 | -C.176945 | 0.739502 | -0.152886 | 0.774368 | -0.126234 | 0.810059 | -0.08703  |
| 0.846511 | -C.035729 | 0.883730 | 0.016212  | 0.921712 | 0.069618  | 0.960470 | 0.14572   |
| 1.000000 | C.220435  |          |           |          |           |          |           |

SPAN STATION = 12.59373  
NO. OF DATA PCINTS = 101

|          |          |          |          |          |           |          |          |
|----------|----------|----------|----------|----------|-----------|----------|----------|
| 1.000000 | C.237601 | 0.560491 | 0.209400 | 0.521777 | 0.165122  | 0.883857 | 0.13351  |
| 0.846730 | 0.105264 | 0.810396 | 0.088283 | 0.774854 | 0.068256  | 0.740105 | 0.03554  |
| 0.706151 | 0.035646 | 0.672957 | 0.037193 | 0.640641 | 0.0039646 | 0.609077 | 0.003564 |
| 0.578298 | 0.003106 | 0.548303 | 0.015259 | 0.519059 | 0.0001848 | 0.490653 | 0.000155 |
| 0.463050 | 0.002899 | 0.436282 | 0.002622 | 0.410261 | 0.0007520 | 0.385015 | 0.000512 |
| 0.360542 | 0.009820 | 0.336848 | 0.005760 | 0.330321 | 0.0094720 | 0.291835 | 0.001232 |
| 0.270527 | 0.014264 | 0.250026 | 0.014300 | 0.230321 | 0.0168434 | 0.211458 | 0.001713 |
| 0.193233 | 0.015594 | 0.175832 | 0.013206 | 0.159222 | 0.0122341 | 0.143458 | 0.001350 |
| 0.128543 | 0.016083 | 0.114441 | 0.017784 | 0.101106 | 0.0122341 | 0.088501 | 0.001699 |
| 0.076658 | 0.031258 | 0.065638 | 0.016730 | 0.055507 | 0.0191825 | 0.046334 | 0.028147 |
| 0.038050 | 0.032683 | 0.030635 | 0.035087 | 0.024313 | 0.0501360 | 0.018626 | 0.055355 |
| 0.013378 | 0.064504 | 0.009184 | 0.067016 | 0.006119 | 0.610760  | 0.003362 | 0.048185 |
| 0.001527 | 0.040381 | 0.003595 | 0.037011 | 0.006116 | 0.315208  | 0.009017 | 0.021185 |
| 0.012560 | 0.092098 | 0.017041 | 0.015580 | 0.022308 | 0.310592  | 0.028618 | 0.070002 |
| 0.035734 | 0.278098 | 0.043503 | 0.045070 | 0.052097 | 0.591857  | 0.061654 | 0.053385 |
| 0.072512 | 0.658209 | 0.084320 | 0.059642 | 0.095972 | 0.566483  | 0.110337 | 0.057525 |
| 0.124445 | 0.538341 | 0.139315 | 0.055377 | 0.155014 | 0.576198  | 0.171564 | 0.057525 |
| 0.188944 | 0.571030 | 0.207137 | 0.056375 | 0.226131 | 0.555679  | 0.245938 | 0.054511 |
| 0.266567 | 0.545323 | 0.288014 | 0.053019 | 0.302273 | 0.511553  | 0.333338 | 0.043891 |
| 0.357204 | 0.465932 | 0.381868 | 0.047251 | 0.407326 | 0.428276  | 0.433576 | 0.033508 |
| 0.460617 | 0.390444 | 0.488449 | 0.037254 | 0.517071 | 0.354631  | 0.546485 | 0.032072 |
| 0.576692 | 0.311102 | 0.607694 | 0.028022 | 0.639486 | 0.243723  | 0.672056 | 0.020891 |
| 0.705355 | 0.177034 | 0.739502 | 0.015292 | 0.774368 | 0.0226175 | 0.810059 | 0.001502 |
| 0.846511 | 0.035524 | 0.883730 | 0.001621 | 0.921712 | 0.0070163 | 0.960470 | 0.015027 |
| 1.000000 | C.220844 |          |          |          |           |          |          |

SPAN STATION = 13.56248  
NO. OF DATA PCINTS = 101

|          |          |          |          |          |           |          |          |
|----------|----------|----------|----------|----------|-----------|----------|----------|
| 1.000000 | C.237601 | 0.960492 | 0.209400 | 0.921777 | 0.165122  | 0.883857 | 0.13351  |
| 0.846730 | 0.105264 | 0.810396 | 0.088283 | 0.774854 | 0.068256  | 0.740105 | 0.03554  |
| 0.706152 | 0.035646 | 0.672958 | 0.037193 | 0.640642 | 0.0039646 | 0.609077 | 0.003564 |
| 0.578298 | 0.003106 | 0.548303 | 0.015804 | 0.519059 | 0.0001550 | 0.490654 | 0.000155 |
| 0.463050 | 0.002899 | 0.436283 | 0.002418 | 0.410261 | 0.0007458 | 0.385015 | 0.000512 |
| 0.360542 | 0.009855 | 0.336847 | 0.002060 | 0.330321 | 0.0094673 | 0.291835 | 0.001232 |
| 0.270527 | 0.014295 | 0.250026 | 0.014148 | 0.230321 | 0.0166706 | 0.211358 | 0.001712 |
| 0.193233 | 0.015354 | 0.175832 | 0.012989 | 0.159224 | 0.0129856 | 0.143458 | 0.001323 |
| 0.128542 | 0.016083 | 0.114441 | 0.014884 | 0.101106 | 0.0166706 | 0.088501 | 0.001699 |
| 0.076658 | 0.031258 | 0.065638 | 0.035087 | 0.024313 | 0.0501360 | 0.018626 | 0.055355 |
| 0.038050 | 0.032683 | 0.030635 | 0.037011 | 0.006116 | 0.315208  | 0.009017 | 0.021185 |
| 0.001527 | 0.040381 | 0.003595 | 0.037011 | 0.006116 | 0.315208  | 0.009017 | 0.021185 |
| 0.012560 | 0.092098 | 0.017041 | 0.015580 | 0.022308 | 0.310592  | 0.028618 | 0.070002 |
| 0.035734 | 0.278098 | 0.043503 | 0.045070 | 0.052097 | 0.591857  | 0.061654 | 0.053385 |
| 0.072512 | 0.658209 | 0.084320 | 0.059642 | 0.095972 | 0.566483  | 0.110337 | 0.057525 |
| 0.124445 | 0.538341 | 0.139315 | 0.055377 | 0.155014 | 0.576198  | 0.171564 | 0.057525 |
| 0.188944 | 0.571030 | 0.207137 | 0.056375 | 0.226131 | 0.555679  | 0.245938 | 0.054511 |
| 0.266567 | 0.545323 | 0.288014 | 0.053019 | 0.302273 | 0.511553  | 0.333338 | 0.043891 |
| 0.357204 | 0.465932 | 0.381868 | 0.047251 | 0.407326 | 0.428276  | 0.433576 | 0.033508 |
| 0.460617 | 0.390444 | 0.488449 | 0.037254 | 0.517071 | 0.354631  | 0.546485 | 0.032072 |
| 0.576692 | 0.311102 | 0.607694 | 0.028022 | 0.639486 | 0.243723  | 0.672056 | 0.020891 |
| 0.705355 | 0.177034 | 0.739502 | 0.015292 | 0.774368 | 0.0226175 | 0.810059 | 0.001502 |
| 0.846510 | 0.035524 | 0.883730 | 0.001621 | 0.921712 | 0.0070163 | 0.960470 | 0.015027 |
| 1.000000 | C.220844 |          |          |          |           |          |          |

[illegible]

SPAN STATION = 14.53123  
NO. OF DATA POINTS = 101

[illegible]

SPAN NO. OF DATA PCINTS = 15.499998

| SPAN NO. | OF DATA | PCINTS | X/C   | CP    | X/C   | CP    | X/C   | CP    | X/C   | CP    | X/C   | CP    |
|----------|---------|--------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1        | 0.000   | 0.000  | 0.560 | 0.492 | 0.094 | 0.009 | 0.521 | 0.777 | 0.163 | 0.577 | 0.883 | 0.857 |
| 0        | 0.846   | 0.730  | 0.810 | 0.396 | 0.835 | 0.310 | 0.774 | 0.854 | 0.673 | 0.302 | 0.740 | 0.105 |
| 0        | 0.706   | 0.152  | 0.672 | 0.358 | 0.358 | 0.550 | 0.640 | 0.641 | 0.386 | 0.992 | 0.609 | 0.077 |
| 0        | 0.578   | 0.297  | 0.030 | 0.303 | 0.148 | 0.550 | 0.519 | 0.059 | 0.007 | 0.150 | 0.490 | 0.653 |
| 0        | 0.463   | 0.090  | 0.000 | 0.282 | 0.052 | 0.597 | 0.413 | 0.410 | 0.000 | 0.400 | 0.385 | 0.015 |
| 0        | 0.360   | 0.527  | 0.336 | 0.847 | 0.568 | 0.325 | 0.330 | 0.224 | 0.000 | 0.936 | 0.291 | 0.838 |
| 0        | 0.270   | 0.235  | 0.250 | 0.026 | 0.076 | 0.257 | 0.159 | 0.322 | 0.000 | 0.656 | 0.211 | 0.358 |
| 0        | 0.193   | 0.543  | 0.125 | 0.583 | 0.128 | 0.227 | 0.101 | 0.105 | 0.000 | 0.179 | 0.088 | 0.501 |
| 0        | 0.128   | 0.658  | 0.152 | 0.440 | 0.176 | 0.246 | 0.055 | 0.017 | 0.000 | 0.410 | 0.046 | 0.334 |
| 0        | 0.038   | 0.378  | 0.065 | 0.637 | 0.170 | 0.294 | 0.024 | 0.313 | 0.000 | 0.986 | 0.003 | 0.556 |
| 0        | 0.013   | 0.378  | 0.000 | 0.518 | 0.404 | 0.047 | 0.006 | 0.188 | 0.000 | 0.362 | 0.000 | 0.362 |
| 0        | 0.001   | 0.527  | 0.000 | 0.362 | 0.673 | 0.300 | 0.061 | 0.187 | 0.000 | 0.449 | 0.009 | 0.017 |
| 0        | 0.001   | 0.528  | 0.000 | 0.359 | 0.309 | 0.647 | 0.022 | 0.399 | 0.000 | 0.557 | 0.028 | 0.618 |
| 0        | 0.012   | 0.734  | 0.043 | 0.501 | 0.428 | 0.718 | 0.052 | 0.008 | 0.000 | 0.257 | 0.061 | 0.654 |
| 0        | 0.072   | 0.512  | 0.084 | 0.319 | 0.647 | 0.189 | 0.096 | 0.972 | 0.000 | 0.582 | 0.110 | 0.348 |
| 0        | 0.124   | 0.444  | 0.139 | 0.315 | 0.562 | 0.192 | 0.155 | 0.014 | 0.000 | 0.520 | 0.171 | 0.564 |
| 0        | 0.188   | 0.545  | 0.207 | 0.137 | 0.567 | 0.461 | 0.261 | 0.131 | 0.000 | 0.626 | 0.333 | 0.338 |
| 0        | 0.266   | 0.567  | 0.288 | 0.015 | 0.531 | 0.114 | 0.407 | 0.277 | 0.000 | 0.517 | 0.433 | 0.576 |
| 0        | 0.357   | 0.617  | 0.381 | 0.869 | 0.447 | 0.686 | 0.517 | 0.071 | 0.000 | 0.260 | 0.546 | 0.485 |
| 0        | 0.460   | 0.617  | 0.468 | 0.449 | 0.368 | 0.765 | 0.407 | 0.337 | 0.000 | 0.338 | 0.672 | 0.059 |
| 0        | 0.576   | 0.692  | 0.607 | 0.654 | 0.275 | 0.155 | 0.639 | 0.465 | 0.000 | 0.220 | 0.810 | 0.059 |
| 0        | 0.705   | 0.395  | 0.673 | 0.502 | 0.147 | 0.572 | 0.774 | 0.388 | 0.000 | 0.309 | 0.960 | 0.470 |
| 0        | 0.846   | 0.511  | 0.883 | 0.730 | 0.020 | 0.572 | 0.521 | 0.712 | 0.000 | 0.107 | 0.000 | 0.470 |
| 1        | 0.000   | 0.000  | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |

SPAN NO. OF DATA PCINTS = 16.46472

| SPAN NO. | OF DATA | PCINTS | X/C   | CP    | X/C   | CP    | X/C   | CP    | X/C   | CP    | X/C   | CP    |
|----------|---------|--------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1        | 0.000   | 0.000  | 0.560 | 0.491 | 0.094 | 0.009 | 0.521 | 0.777 | 0.163 | 0.577 | 0.883 | 0.857 |
| 0        | 0.846   | 0.730  | 0.810 | 0.396 | 0.835 | 0.310 | 0.774 | 0.854 | 0.673 | 0.302 | 0.740 | 0.105 |
| 0        | 0.706   | 0.152  | 0.672 | 0.358 | 0.358 | 0.550 | 0.640 | 0.641 | 0.386 | 0.992 | 0.609 | 0.077 |
| 0        | 0.578   | 0.298  | 0.030 | 0.303 | 0.148 | 0.550 | 0.519 | 0.059 | 0.007 | 0.150 | 0.490 | 0.654 |
| 0        | 0.463   | 0.090  | 0.000 | 0.282 | 0.052 | 0.597 | 0.413 | 0.410 | 0.000 | 0.400 | 0.385 | 0.015 |
| 0        | 0.360   | 0.542  | 0.336 | 0.847 | 0.568 | 0.325 | 0.330 | 0.224 | 0.000 | 0.936 | 0.291 | 0.831 |
| 0        | 0.270   | 0.235  | 0.250 | 0.026 | 0.076 | 0.257 | 0.159 | 0.322 | 0.000 | 0.656 | 0.211 | 0.358 |
| 0        | 0.193   | 0.543  | 0.125 | 0.583 | 0.128 | 0.227 | 0.101 | 0.105 | 0.000 | 0.179 | 0.088 | 0.501 |
| 0        | 0.128   | 0.658  | 0.152 | 0.440 | 0.176 | 0.246 | 0.055 | 0.017 | 0.000 | 0.410 | 0.046 | 0.334 |
| 0        | 0.038   | 0.378  | 0.065 | 0.637 | 0.170 | 0.294 | 0.024 | 0.313 | 0.000 | 0.986 | 0.003 | 0.556 |
| 0        | 0.013   | 0.378  | 0.000 | 0.518 | 0.404 | 0.047 | 0.006 | 0.188 | 0.000 | 0.362 | 0.000 | 0.362 |
| 0        | 0.001   | 0.527  | 0.000 | 0.362 | 0.673 | 0.300 | 0.061 | 0.187 | 0.000 | 0.449 | 0.009 | 0.017 |
| 0        | 0.001   | 0.528  | 0.000 | 0.359 | 0.309 | 0.647 | 0.022 | 0.399 | 0.000 | 0.557 | 0.028 | 0.618 |
| 0        | 0.012   | 0.734  | 0.043 | 0.501 | 0.428 | 0.718 | 0.052 | 0.008 | 0.000 | 0.257 | 0.061 | 0.654 |
| 0        | 0.072   | 0.512  | 0.084 | 0.319 | 0.647 | 0.189 | 0.096 | 0.972 | 0.000 | 0.582 | 0.110 | 0.348 |
| 0        | 0.124   | 0.444  | 0.139 | 0.315 | 0.562 | 0.192 | 0.155 | 0.014 | 0.000 | 0.520 | 0.171 | 0.564 |
| 0        | 0.188   | 0.545  | 0.207 | 0.137 | 0.567 | 0.461 | 0.261 | 0.131 | 0.000 | 0.626 | 0.333 | 0.338 |
| 0        | 0.266   | 0.567  | 0.288 | 0.015 | 0.531 | 0.114 | 0.407 | 0.277 | 0.000 | 0.517 | 0.433 | 0.576 |
| 0        | 0.357   | 0.617  | 0.381 | 0.869 | 0.447 | 0.686 | 0.517 | 0.071 | 0.000 | 0.260 | 0.546 | 0.485 |
| 0        | 0.460   | 0.617  | 0.468 | 0.449 | 0.368 | 0.765 | 0.407 | 0.337 | 0.000 | 0.338 | 0.672 | 0.059 |
| 0        | 0.576   | 0.692  | 0.607 | 0.654 | 0.275 | 0.155 | 0.639 | 0.465 | 0.000 | 0.220 | 0.810 | 0.059 |
| 0        | 0.705   | 0.395  | 0.673 | 0.502 | 0.147 | 0.572 | 0.774 | 0.388 | 0.000 | 0.309 | 0.960 | 0.470 |
| 0        | 0.846   | 0.511  | 0.883 | 0.730 | 0.020 | 0.572 | 0.521 | 0.712 | 0.000 | 0.107 | 0.000 | 0.470 |
| 1        | 0.000   | 0.000  | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |





|          |           |          |           |          |           |          |          |
|----------|-----------|----------|-----------|----------|-----------|----------|----------|
| 0.460619 | -C.245829 | 0.488450 | -0.239440 | 0.517072 | -0.229495 | 0.546486 | -0.21668 |
| 0.576693 | -C.204904 | 0.607655 | -0.186239 | 0.639487 | -0.163666 | 0.672057 | -0.14168 |
| 0.705356 | -C.124829 | 0.739503 | -0.112534 | 0.774389 | -0.097547 | 0.810060 | -0.07234 |
| 0.846512 | -C.038530 | 0.883731 | -0.006437 | 0.921712 | 0.025749  | 0.960469 | 0.007570 |
| 1.000000 | C.142642  |          |           |          |           |          |          |

## APPENDIX E

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

```

*****FLO27*****9/12/83*****
THREE DIMENSIONAL WING ANALYSIS IN TRANSONIC FLOW USING
FINITE VOLUME SCHEME WITH SHEARED PARABOLIC COORDINATES
PROGRAMMED BY ANTONY JAMESON, JANUARY-APRIL 1977

*****
THE BEING VERSION OF FLO27 WAS PREPARED BY DR. HAI-CHOW
CHEN WITH THE FOLLOWING MODIFICATIONS
1) TEMPORARY STORAGE OF THE LARGE CORE MEMORY REQUIREMENTS
   HAS BEEN IMPLEMENTED TO REDUCE THE COMPUTING COSTS
   BY BUFFERING DATA IN AND OUT OF CORE.

2) STANDARD BOEING INPUT FORMAT FOR THE WING SECTION
   HAS BEEN USED.

3) SUBPROGRAM BLIN HAS BEEN IMPLEMENTED TO ADD THE
   DISPLACEMENT THICKNESS TO THE ORIGINAL WING SECTIONS

4) WING SECTION LEADING EDGE SINGULAR POINT IS FOUND BY
   COMPUTING THE FOCUS OF A PARABOLA BY 2N+1 POINTS
   LEAST-SQUARE FIT CENTERED AT THE LEADING EDGE POINT.
   N IS SUPPLIED BY THE USER THROUGH INPUT CARD.

5) TRAILING EDGE CLOSURE ANGLE AND BISECTOR SLOPE ARE
   COMPUTED BASED ON BACKWARD DIFFERENCE.

6) OPTION FOR PRINTER-PLOTTING OF THE UNWRAPPED
   WING SECTIONS IS AVAILABLE

*****
THE FOLLOWING FILES ARE USED TO EXECUTE FLO27. SOME OF THESE
FILES ARE USED SUBSEQUENTLY IN OTHER MODULES OF THE VISCOUS/
INVISIDIC INTERACTIVE WING SYSTEM

FILE1 IS USED TO BUFFER DATA IN AND OUT OF CORE
FILE2 IS USED TO BUFFER DATA IN AND OUT OF CORE
FILE3 IS USED TO BUFFER DATA IN AND OUT OF CORE

```

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
FILE4 IS USED TO READ IN THE VELOCITY PCTENTIAL
GENERATED PREVIOUSLY

FILE8 IS WRITTEN FOR DATA TRANSFER TO BCEING
TUREULENT BCUNCARY LAYER PROGRAM A411.

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

FILE10 IS USED TO SAVE THE SECTION X, Y, Z
LOCATION FOR A411 IN WHICH CALCULATES THE CORRESPONDING
DISPLACEMENT THICKNESS
WHERE X, Y, Z SHOULD BE THE WING SURFACE LOCATION
FOR THE CURRENT RUN

FILE11 IS USED TO READ IN THE DISPLACEMENT
THICKNESS FROM A411 IN

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

TAPE13 IS USED TO SAVE PART OF THE OUTPUT
SKIPPED FROM THE LINE PRINTER

TAPE14 IS USED TO SAVE THE VELOCITY POTENTIAL
FOR FUTURE USE
*****
COMMON
1      G(161,18,3),SO(161,35),VORT(115),ZV(115),
2      IV(161,35),ITE1(35),ITE2(35),
3      AO(161),BO(18),XO(35),YC(35),ZO(35),SCAL(35),
4      IX,AY,NZ,KTE1,KTE2,ISYM,KSYM,FUS,
      YAW,CYAW,SVAW,ALPHA,CA,SA,FMACH,N1,N2,N3,IO
COMMON /CPF/NL
COMMON /PCKR/ PTCK
COMMON /FLO/ P1, P2, P3, FRES, IRES, JRES, KRES,ARES,CG,IG,JG,KG,AG,NSUP
COMMON /PARMT3/ XT3(161),VT3(161),ZT3(161),
      UT3(161),VT3(161),WT3(161),NO1,Z
COMMON /PRS/ XOCO(161),
DIMENSION
1      XS(161,11),YS(161,11),ZS(11),XLE(11),YLE(11),
      SLOFT(11),TRAIL(11),NP(11),EI(11),E2(11),E3(11),
      DI(161),D2(161),D3(161),SN(161),
      SV(161),SM(161),CP(161),XP(161),YP(161),
      YPO(35),ZPO(35),XMAX(35),XMIN(35),YMAX(35),YMIN(35),
      CHORD(35),SCL(35),SCD(35),SCM(35),TITLE(20),
      FIT(3),COVO(3),PI0(3),P20(3),P30(3),FSMOD(3),
      RES(20),CCOUNT(20),PTMAP(3),
1
DIMENSION

```

```

2 DUMX(161),CUMY(161),DELR(161)
C** ***** WHICH HAVE RECOMMENDED PROGRAM VALUES **
C** INITIALIZE INPUT PARAMETERS *****
XSCAL = 0.0
PSCAL = 0.0
FCCNT = 0.0
BLCP = 0.0
WEIG = 1.0
PTCK = 0.0
C** *****
ND = 161
NE = 161
KP LCT = 0
IP LCT = 1
IS TCP = 2
N1 = 1
N2 = 1
N3 = 2
REKIND 1 = 1
REKIND 2 = 1
REKIND 3 = 2
REKIND 4 = 3
REKIND 10
REKIND 13
REKIND 14
JO
RAD = 0
WRITE (6,600) = 57.295779513082
1 WRITE (6,600)
2 FORMAT (1X,A460) MCDIFIED FROM FLC27 OF ANTONY JAMESON,,
1 COURANT 50.0 THREE DIMENSIONAL WING ANALYSIS IN TRANSONIC FLOW,
2 OF OTHER FINITE VOLUME SCHEME/
3 27H USING VERSION, PREPARED BY DR. HAI-CHEN CHEN,
4 08OEINGARD BOEING INPUT FORMAT FOR WING SECTION DATA IS USED.)
READ (5,530) TITLE
WRITE (6,630) TITLE
READ (5,500) FNX,FNY,FNZ,FMESH,FPLGT
NX = FNX
NY = FNY
NZ = FNZ
MMESH = FMESH
IF (NX.LT.1) GO TO 301
KPLOT = ABS(FPLGT)
READ (5,500)

```



```

WRITE (6,52) K,NPCK,NPCK1,ZS(K)
WRITE (6,54) (OLMX(I),ZS(K),DUMY(I),I=1,NPCK)
45 CONTINUE
50 ENDFILE 10
52 FORMAT (1H0,5X,3I5,F11.4/)
54 FORMAT (12F11.4)
56 CONTINUE
IF (KSYM.NE.0) YA = 0.
ISYM = ISYMO
IF (AL.NE.0) ISYM = 0
YAW = YA/RAD
SYAW = COS(YAW)
CA = SIN(YAW)
SA = CYAW*COS(ALPHA)
IFAC (4) = CYAW*SIN(ALPHA)
IFAC (4) = GCTC 91
MX = NX,NY,NZ,NM,K1,K2,NIT
MY = NX +1
MZ = NY +2
DO 62 K=1,MZ
READ (4) ((G(I,J,1),I=1,MX),J=1,MY)
BUFFER (CLT(N3,1)) (G(I,1,1),G(MX,MY,1))
WRITE (N3) ((G(I,J,1),I=1,MX),J=1,MY)
IF (UNIT(N3).GT.0.) GO TO 1
BUFFER (CLT(N1,1)) (G(I,1,1),G(MX,MY,1))
WRITE (N1) ((G(I,J,1),I=1,MX),J=1,MY)
IF (UNIT(N1).GT.0.) GO TO 1
CONTINUE
(VORT(K),K=K1,K2)
62 READ (4)
REWIND N3
REWIND N1
REWIND 4
91 CALL CCCRD
1 CALL SINGL
1 CALL SUFF
1
2
3
IF (INC.EC.C) GC 291
IF (FCCNT.GE.1.) GO TO 101
NM = 1
NIT = 0
CALL ESTIM
IF (IG.EC.0) GO TO 1

```

C C C C

```

REWIND A3
REWIND N1
101 IF (PTCK .GE. 1.) WRITE (6,600)
FCCNT = 0
COV = COVG(NM)
P1 = P10(NM)
P2 = P20(NM)
P3 = P30(NM)
MIT = FIT(NM) +NIT
KIT = MIT
IF (KIT.LT.2) KIT=2
JIT = NIT
LRES = 0
MRES = (MIT -NIT -2)/200 +2
NX = 0
MY = 0
MZ = 0
KY = 0
K1 = 1
K2 = 1
IF (KSYN.EC.0) GO TC 103
K1 = 3
K2 = 3
LZ = NZ/2 +1
IF (KSYN.NE.0) LZ = 3
IF (PTCK.LE. 0.) GC TC 108
WRITE (6,104)
104 FORMAT(48H FOUND LOCATION OF LOCATION OF WING AND VORTEX SHEET,
27H IN CCGRDINATE PLANE Y = 0.,
27H ((I,K),K=K1,K2),I=1,NX))
DO 106 I=1,NX
WRITE (6,650) (IV(I,K),K=K1,K2)
106 CONTINUE
IMAP = FTMAP(NM)
IF (IMAP.EQ. 0) GC TC 830
WRITE (6,600)
WRITE (6,112)
112 FORMAT(49H CHORDWISE CELL DISTRIBUTION IN SQUARE RCOT PLANE,
24H AND MAPPED SURFACE COORDINATES AT CENTER LINE AND TIP)
DO 820 ISEC = LZ,KITE2,IMAP
WRITE (6,812) ISEC,ZP0(ISEC)
112 FORMAT(15H I = ,G1 = .4)
DO 820 I=1,3
WRITE (6,812) ISEC,ZP0(I,ISEC)
812 CALL PFX Y (2,NX,AO,SO(1,ISEC))
820 CONTINUE
830 IF (PTCK.LE. 0.) GC TC 130
WRITE (6,116)

```

```

116 FORMAT(15H0 TE LOCATION ,15H PCWER LAW )
WRITE (6,610) XLIM,AX
WRITE (6,600)
WRITE (6,118)
118 FORMAT(46H0NORMAL CELL DISTRIBUTION IN SQUARE ROOT PLANE/
15H0
DO 120 J=1,KY
120 WRITE (6,610) BC(J)
122 FORMAT(15H0 SCALE FACTOR,15H PCWER LAW )
WRITE (6,610) SY,AY
WRITE (6,600)
WRITE (6,124)
124 FORMAT(45H0SPANWISE CELL DISTRIBUTION AND SINGULAR LINE/
15H0 X SING ,15H Y SING )
DO 126 K=K1,K2
126 WRITE (6,610) ZO(K),XO(K),VO(K)
128 FORMAT(15H0 TIP LCCATIGN,15H PCWER LAW )
WRITE (6,610) ZLIM,AZ
130 CONTINUE
WRITE (6,600)
WRITE (6,132)
132 FORMAT(15H0ITERATIVE SOLUTION)
134 WRITE (6,134) MACH NO ,15H YAW ,15H ANG OF ATTACK)
FORMAT(15H0 FMACH,YA,AL ,15H NZ ,15H )
WRITE (6,610) NX ,15H NY ,15H
FORMAT(15H0 NX,NY,NZ ,15H RELAX FCT 2 ,15H RELAX FCT 3 )
WRITE (6,610) RELAX FCT 1 ,15H RELAX FCT 2 ,15H RELAX FCT 3 )
WRITE (6,140)
140 FORMAT(10H0ITERATION,
15H MAX CORREC N ,4H I ,4H J ,4H K ,15H AVG CORREC N ,
15H MAX RESIDUAL ,4H I ,4H J ,4H K ,15H AVG RESIDAL ,
12H CIRCULATION,15H SONIC PIS)
141 NIT
NIT = NIT +1
JIT = JIT +1
CALL MIXFLC
IF (10-FC=0) GO TO 151
JO
REWIND N1
REWIND N2
N1 = N1
N2 = N2
N3 = N3

```

```

N3
WRITE (6,66C) NIT,DG,IG,JG,KG,AG,FRES,IFES,JRES,KRES,ARES,
1 VGRIT(LZ),NSUP
  LRES = LRES +1
  IF (LRES.EQ.MRES) LRES = 1
  IF (LRES.NE.1) GO TO 143
  NRES = NRES +1
  COUNT(ARES) = NRES -1
  RES(NRES) = FRES
  IF (JIT.GE.KIT) GO TO 251
143 IF (NM.LE.1) .OR. NM.LT.MMESH) GO TO 148
  IF (ABS(DG).LE.COV) GO TC 251
148 CONTINUE
  IF (NIT.LT.MIT.AND.ABS(DG).GT.COV.AND.ABS(DG).LT.10.) GO TO 141
  GO TO 161
151 IF (JO.EC.1) GO TO 1
  REWIND A1
  REWIND A2
  JO
  N3
  N2
  N1
  GO TO 141
161 RATE = 0.
  IF (NRES.GT.1) RATE = (ABS(RES(NRES)/RES(1))) -COUNT(1)))
1
  WRITE (6,162) MAX RESIDUAL 1,15H MAX RESIDUAL 2,15H
  FORMAT(15H0 MAX RESIDUAL 1,15H MAX RESIDUAL 2,15H
162 15H REDUCTN/CYCLE: 15H CONV TOLERANCE)
  WRITE (6,670) RES(1),RES(NRES),COUNT(NRES),RATE
1,CCV
  WRITE (6,600)
  DO 164 N=1,3
  BUFFER IN (N1,1) (G(1,1,M),G(MX,MY,M))
  REAC (N1,ERR=151) ((G(I,J,M),I=1,MX),J=1,MY)
  IF (UNIT(N1).GT.0.) GO TO 151
164 CONTINUE
  LX = NX/2 +1
  K
  KKK
  IF (NM.LT.MMESH) GO TO 17C
  REWIND C
  NRC = KIE2 - KTE1 + 1
  WRITE (6) (TITLE(I),I=1,8),FMACH,ALPHA,NRC
17C CONTINUE

```

WORK

```

171 K IF (K.EC.MZ) GO TO 191
DO 172 J=1,MY
DO 172 I=1,MX
G(I,J,1) = G(I,J,2)
G(I,J,2) = G(I,J,3)
BUFFER IN (N1,1) (G(I,1,3),G(MX,MY,3))
READ (N1,ERR=151) ((G(I,J,3),I=1,MX),J=1,MY)
C
C IF (UNIT(N1).GT.0.) GO TO 151
IF (K.LT.KTEL.OR.K.GT.KTE2) GO TO 171
CALL VELD (K,SV,SM,CP,XP,YP,XMAX(K),XMIN(K),YMAX(K),YMIN(K))
I1 = ITEL(K)
I2 = ITE2(K)
CHCRD(K) = XP(I1) -XP(LX)
CALL FCRCF (I1,I2,XP,YP,CP,AL,CHORD(K),XO(K),YPO(K),
1 SCL(K),SCD(K),SCM(K))
KKK = KKK +1
IF (KPLCT.GT.1.AND.K.GT.KTEL) GC TC 185
IF (KPLCT.EQ.0.AND.KKK.GT.1) GO TO 185
WRITE (6,600)
WRITE (6,182)
182 FORMAT(24HSECTION CHARACTERISTICS/
15H0 MACH NO ,15H YAW
15H FMACH,YA,AL
184 WRITE (6,610)
WRITE (6,184)
184 FORMAT(15H HOSPAN STATION ,15H CL
15H CM
185 WRITE (6,61C) ZPO(K),SCL(K),SCD(K),SCM(K)
Z = ZPC(K)
IF (MIT.LE.0) GC TO 850
IF (KPLCT.LE.2) CALL CPLOT (2,NX,FMACH,XP,YF,CP,SM,I1,I2,KPLOT)
850 CONTINUE
WRITE CNE FILE ON TAPE 8
IF (NM.LT.NMESH) GO TO 186
WRITE (8) NCI
WRITE (8) (XT3(I),YT3(I),UT3(I),VT3(I),I=1,NCI)
NRC=I2-I1+1
WRITE CP VS X/C SECTION DATA FOR FINAL MESH ON TAPE 9
WRITE (5,900) ZPO(K)
WRITE (5,910) NRD
WRITE (5,920)
WRITE (5,95C) (XQCD(J),CP(J),J=1,I2)
FORMAT (15H HOSPAN STATION =,F12.5)
900 FORMAT (1X,20HNC. CF DATA POINTS =,I5)
910 FORMAT (1X,6H X/C,8X,2HCP,7X,3HX/C,8X,2HCF,7X,3HX/C,8X,2HCP,
920 17X,3HX/C,8X,2HCF,3X)
95C FORMAT (8F10.6)
186 CONTINUE

```

```

C WHEN KPLOT = 2 CALL SUBROUTINE VERTEC WHICH PLOTS CP VS X/C
C FOR EACH SECTION OF THE FINAL MESH
      IF (KPLOT.EQ.2.AND.NM.EQ.MMESH) CALL VERTEC(I1,I2,XOCD,CP,NRD,
        ZPO,FMACH,YA,AL,SCL,SCD,SCN,K)
    1 GO TO 171
  191 CONTINUE
    IF(NM.LT.MMESH)GO TC 200
    ENCFILE 8
    REWIND 8
    ENDFILE 5
    REWIND 5
  200 CONTINUE
    CALL TCTFOR (KTE1,KTE2,CHORD,SCL,SCD,SCN,XG,YPO,ZPO,
      CL,CD1,CMP,CMR,CMY,)
      CD1
      = CYAW*CD1
      = CDO +CD1
      VLD1
      = 0
      IF (ABS(CD1).GT.1.E-6) VLD1 = CL/CD1
      VLD
      = 0
      IF (ABS(CD).GT.1.E-6) VLD = CL/CD
      WRITE (6,60C)
      WRITE (6,192)
      FORMAT(21HOWING CHARACTERISTICS/
        MACH NO ,15H YAW
        FMACH,YA,AL
        CL
        ,15H CD FORM
        CD,CD1,CDO,CD,VLD1,VLD
        CM YAW ,15H CM RCLL
        CMY,CMR,CMP
        REWIND 11
      IF (KPLOT.LT.1) GO TO 201
      CALL RPLOT (IPLCT,NRES,RES,COUNT,TITLE,FMACH,YA,AL,NX,NY,NZ)
      CALL DRAW (IPLCT,XMAX,XMIN,YMAX,YMIN,ZPO,FUS,TITLE,NZ,KTE1,KTE2)
      CALL THREEED (IPLCT,SV,SM,CP,XP,YP,ZPO,TITLE,YA,AL,
        VLD,CL,CD,CHORD,XSCAL,PSCALE)
    1 IF (IO.EQ.0) GO TO 151
    IF (ISTCP.EQ.1) GO TO 301
    IF (NM.LT.MMESH) GC TC 203
    GO TO 1
  203 CONTINUE
      NX
      NY
      NZ
      = NX
      = NY
      = NZ
      +NX
      +NY
      +NZ

```

```

CALL CCCRD (NX,NY,NZ, KSYM,ZIIP,XLIM,ZLIM,
1 CALL SINGL (SY,AX,AZ,PX,PZ,AO,BC,ZO)
1 CALL SUFF (NS,NZ, KSYM,KTE1,KTE2,FUS,ZPC,E1,E2,E3,IND)
1 (ND,NE, EQ,XLIM,XO,YO,ZO,YFC,ZPC,E1,E2,E3,IND)
2 (YAW,XTEQ,XLIM, FIX,NP,XS,YS,ZS,SLOP,TRAIL,
3 AO,XO,ZO,SO,SCAL,ZV,IV,ITE1,ITE2,
XP,YP,SN,C1,D2,D3,IAD)
IF (INC.EQ.0) GC IC 291
CALL REFIN
IF (IO.EQ.0) GO TO 221
REWIND N1
NSMOO = FSMCC(NM)
IF (NSMCC.LT.1) GO TO 211
DO 202 N=1,NSMCC
CALL SMCC
IF (IO.EQ.0) GO TO 221
REWIND N1
REWIND N2
202 N1 = N1
211 N1 = N2
N2 = N3
N3 = N
NM = NM +1
NIT = 0
GO TO 101
221 NX = NX/2
NY = NY/2
NZ = NZ/2
CALL CCCRD (NX,NY,NZ, KSYM,ZIIP,XLIM,ZLIM,
1 CALL SINGL (SY,AX,AZ,PX,PZ,AO,BC,ZO)
1 CALL SUFF (NS,NZ, KSYM,KTE1,KTE2,FUS,ZPC,E1,E2,E3,IND)
2 (YAW,XTEQ,XLIM, FIX,NP,XS,YS,ZS,SLOP,TRAIL,
3 AO,XO,ZO,SO,SCAL,ZV,IV,ITE1,ITE2,
XP,YP,SN,C1,D2,D3,IAD)
IF (INC.EQ.0) GC IC 291
GO TO 151
251 K1 = KTE1 - ITE2(KTE2) -NX/2
K2 = KTE2 +ITE2(KTE2) -NX/2
WRITE (14) NX,NY,NZ,NM,K1,K2,NIT
DO 262 K=1,NZ
BUFFER IN ERR=281) ((G(I,J,I),I=1,MX),J=1,MY)
READ (UNIT,N1),GT,0) GO TO 281
IF (UNIT,N1),GT,0) ((G(I,J,I),I=1,MX),J=1,MY)
262 WRITE (14) ((G(I,J,I),I=1,MX),J=1,MY)

```

```

C
REWIND A1
WRITE (14) (VORT(K),K=K1,K2)
ENDFILE 14
REWIND 14
CALL SWITCH(1,ISTOP)
CALL SLIPET(1,ISTOP)
IF (ISTCF.EC.1) GO TO 161
J1 T
IF (INIT.LT.MIT.AND.ABS(DG).GT.COV.AND.ABS(DG).LT.10.) GC TO 141
GO TO 161
REWIND 4
281 GO TO 151
291 WRITE (6,600)
292 FORMAT(24HOBAD DATA,SPLINE FAILURE)
301 GO TO 1
CONTINUE
REWIND 10
IF (KPLCT.GT.0) CALL PLOT(0.,0.,955)
STOP
FORMAT(1X)
50C FORMAT(8F10.6)
51C FORMAT(2CA4)
60C FORMAT(1H1)
61C FORMAT(F12.5,7G15.5)
63C FORMAT(1H0,20A4)
64C FORMAT(1H8,7I15)
65C FORMAT(1X,3I13)
66C FORMAT(1I10,E15.5,3I4,E15.5,F10.5,I10,F10.3)
67C FORMAT(2E15.4,2F15.4,E15.4)
END

```



```

      IF (I.LT.NL) SI = -1.
      GO TO 850
820  CONTINUE
      S = 0.
      DL = DELR(I)
      DX = 0.
      DY = DL*SI
      GO TO 850
850  CONTINUE + DYXN**2
      S = 1.
      S = SQR1(S)
      S = 1./S
      F = 1.
      IF (DYXN *SI.LT.0.) F = -1.
      DL = DELR(I)
      DX = S*F
      DY = ABS(DYXN)*S*SI
      DX = DX + DL
      DY = DY + DL
      XT(I) = X2 + DX*WEIG
      YT(I) = Y2 + DY*WEIG
880  CONTINUE
      IF (PTCK .LE. 1.) GO TO 890
      WRITE (6,1000) I,F,S,DL,DX,DY,CYXN,XT(I),YT(I),CYX,WEIG
890  CONTINUE
      IF (I.EC.N) GO TO 900
      X1 = X2
      Y1 = Y2
      X2 = X3
      Y2 = Y3
      I = I+1
      GO TO 200
900  CONTINUE
      RETURN
1000  FORMAT (1H ,15,F7.2,9G13.5)
      END

```

```

C**SUBROUTINE PPXY*****
C  SUBROUTINE PPXY (I1,I2,X,Y)
C  SUBPROGRAM FOR LINE PRINTER PLOTTING OF THE UNWRAPPED
C  WING SECTIONS
COMMON /PCKR/ PTCK
COMMON /SHARE/ LINE(100)
DIMENSION X(1),Y(1)
DATA IB /1H /, IP /1H+/, KMAX /100/, ACC /1.5/,
1 IZ /1F/, ICONST /0/
DO 10 I=1,100
LINE(I) = IB
10 CONTINUE
YMAX = -1.0E35
YMIN = -YMAX
WICTH = KMAX - 5
DO 20 I = 1, I2
YMAX = AMAX1(YMAX, Y(I))
YMIN = AMIN1(YMIN, Y(I))
20 CONTINUE
VAL = ABS(YMAX) + ABS(YMIN)
S = WICTH/VAL
KK = 0
IF (ICONST.LE. YMAX .AND. ICONST.GE. YMIN) KK = S*(YMAX-ICONST)+ADD
IF (KK.NE.0) LINE(KK)=I2
DO 30 I=1, I2
K = S*(YMAX-Y(I)) + ADD
IF (K.LT.1) K = 1
IF (K.GT.KMAX) K = KMAX
LINE(K) = IP
WRITE(6,100) I,X(I),Y(I),LINE
LINE(K) = IB
IF (K.EQ.KK) LINE(KK) = IZ
30 CONTINUE
RETURN
100 FORMAT (1X,I3,2F10.4,4X,100A1)
ENC

```

```

C**SUBROUTINE LSQR*****
SUBROUTINE LSQR (NL,NB,XP,YP,XSING,YSING)
C
C SUBPROGRAM FOR WING SECTION LEADING EDGE SINGULAR POINT
C CALCULATION BY MEANS OF COMPUTING THE FCCUS OF A
C PARABOLA BY NB*2+1 POINTS LEAST-SQUARE FIT CENTERED AT
C THE LEADING EDGE
C
C NB: SUPPLY BY THE CALLING PROGRAM GEOM
COMMON /FCKR/ FTCK
DIMENSION XP(1),YP(1)
N1 = NL - NB
N2 = NL + NB
N = N2 - N1 + 1
A1 = N
B1 = 0.
C1 = 0.
A2 = 0.
B2 = 0.
C2 = 0.
A3 = 0.
B3 = 0.
C3 = 0.
D1 = 0.
D2 = 0.
D3 = 0.
SCALE2 = 100.
SCALE2 = 500.
DO 300 I = N1,N2
  YY = (YP(I) - YP(NL))*SCALE
  B1 = B1 + YY
  C1 = C1 + YP2
  YP3 = C2 + YP3
  YP4 = C3 + YP4
  XX = XP(I)*SCALE2
  C1 = C1 + XX
  YX = Y1 + YX
  D2 = D2 + YF2
  Y2X = D3 + Y2X
300 D3 = D3 + B1
  A2 = A2 + C1
  B2 = B2 + C1
  A3 = A3 + C2
  B3 = B3

```



```

DXAM = CXA
DXAM = CXA
DX2M = DXAM**2
60C CONTINUE LE 0.) GC TC 65C
WRITE (6,70C) I,X,Y,DX,DX2,R2,DXAM,DX2M,XP(1),YP(1)
65C CONTINUE LE 0.) GC TC 75C
IF (PTCK LE 0.) GC TC 75C
RA = R2/A1
WRITE (6,70C) N,RA,DXAM
70C FORMAT (113.9G13.5)
CONTINUE
ENTRY LSC
IF (DXAM LE 1.E-4) RETURN
WRITE (6,80C) DXAM
80C FORMAT (1H0.5X,'WARNING ??? DEVIATION OF THE LEADING EDGE PCINIS',
1 FROM PARABOLA IS GREATER THAN 0.001',6X,'DXAM =',G13.4//)
RETURN
END

```



```

ISYMO      = 1
XTEC       = 0.
CHORDO     = 0.
11 READ    (5,500) ZS(K),XL,YL,CHORD,THICK,AL,FSEC
IF (K.EC.1) ZS1 = ZS(1)
ALPHA      = AL/RAD
IF (K.GT.1.AND.FSEC.EQ.0.) GO TO 31
READ (5,500)
READ (5,510) FN
      NB = FN
      N1 = N + 1
      N1 = N + 1
      READ (5,520) (XF(N1-I),YP(N1-I),I=1,N)
52C FORMAT (6F10.0)
      DO 26 I=1,N
      IF (XP(I+1).LT.XP(I)) GC TC 26
      NL = I
      GO TO 800
26 CONTINUE
800 IF (FNB.LE.0.) GC TO 21
      DY1 = YF(1) - YF(2)
      DX1 = XF(1) - XF(2)
      DYU = YF(N) - YF(N-1)
      DXU = XF(N) - XF(N-1)
      TSU = CYU/DXU
      TSL = CYL/DXL
      TRL = ATAN2(DYL,DXL) - ATAN2(DYU,DXU)
      SLT = (TSL+TSU)*.5
      NB = FNB
      CALL LSCR (NL,NL,XP,YP,XSING,YSING)
21 WRITE (6,600) ZS(K)
22 WRITE (6,620) NL,XP(NL),YP(NL),I=1,N)
22 FORMAT (16F10.5/ 15H TE SLOPE ,15H X SING ,
1 15H Y SING
2 15H TRL,SLT,XSING,YSING
27 WRITE (6,610) TRL,SLT,XSING,YSING
62C FORMAT (16F10.5/ 15H TE SLOPE ,15H X SING ,
1 15H Y SING
2 15H TRL,SLT,XSING,YSING
31 SCALE = CHORD/(XP(1) - XP(NL))*THICK*SCALE
      XLE(K) = XL + (XSING - YP(NL))*THICK*SCALE
      YLE(K) = YL + (YSING - XP(NL))*THICK*SCALE
      XX = XP(NL)
      YY = YP(NL)

```

```

CA      = COS(ALPHA)
SA      = SIN(ALPHA)
DO 32 I=1,N
XS(I,K) = SCALE*((XP(I) -XX)*CA +THICK*(YP(I) -YY)*SA)
YS(I,K) = SCALE*(THICK*(YP(I) -YY)*CA - (XP(I) -XX)*SA)
32 SLOPT(K) = THICK*SLT -TAN(ALPHA)
TRAIL(K) = THICK*TRL/RAD
NP(K) = N
CHORDO = AMAX1(CHORDO,CHORD)
IF (YS-LE.O..CR.ALPHA.NE.O.) ISYMC = C
WRITE (6,42) ZS(K)
42 FORMAT (2F10.5,15H XLE RATIO,15H YLE THICK,15H XL,YL,CHORD,THICK,AL
1 WRITE (6,610)
2 YMIN = YP(NL)
YMAX = YMIN
DO 44 I=1,N
IF (YP(I) .GE. YMIN) GC TO 43
JMIN = I
YMIN = YP(I)
43 IF (YP(I) .LE. YMAX) GC TO 44
JMAX = I
YMAX = YP(I)
44 CONTINUE
YDIF = YMAX - YMIN
NN = N - 1
SUM = C.
DO 46 I = NL,NN
SUM = SUM + .5*(YP(I)+YP(I+1))*(XP(I+1)-XP(I))
46 CONTINUE
NM = NL - 1
DO 48 I = 1,NM
SUM = SUM + .5*(YP(I)+YP(I+1))*(XP(I+1)-XP(I))
48 CONTINUE
30C WRITE (6,300)
15H JMAX YMIN ,15H YDIF JMIN ,15H AREA YMAX
30C FORMAT (15H)
320 WRITE (6,320) YMIN,JMIN,YMAX,JMAX,YDIF,SUM
15H CALL LSC
IF (FUS-LE.O.) GO TO 61
R = AMAX1(0.,(FUS**2 - YLE(1)**2))
Z = ZS(K) - ZSI + SQRTR
R = FUS**2/(YLE(K)**2 +Z**2)
ZS(K) = Z*(1. -R)
YLE(K) = YLE(K)*(1. +R)
S = R*XS(NL,K)

```

```

XLE(K) = XLE(K) -S
DO 52 I=1,N
  XS(I,K) = XS(I,K) +S
  YS(I,K) = YS(I,K) * (1. +R)
61 K IF (K.LE.NS) GO TO 11
  Z0 = ZS(1)
  IF (K.SY.NE.0) Z0 = ZS(1) +ZS(NS)
  DO 62 K=1,NS
    XTEO = AMAX1(XTEO,XS(1,K))
    ZS(K) = ZS(K) -Z0
62 ZTIP = ZS(NS)
  RETURN
500 FORMAT(1X)
510 FORMAT(EF10.6)
600 FORMAT(1H1)
610 FORMAT(F12.4,7F15.4)
  ENC

```

```

C**SUBROUTINE COORD*****
SUBROUTINE COORD (NX,NY,NZ,KSVM,ZIIP,XLIM,ZLIM,
1 SETS UP STRETCHED PARABOLIC AND SPANWISE COORDINATES
DIMENSION AO(1),BO(1),ZO(1)
PI BOUND = 3.1415926535898
AX = .95
AY = .5
AZ = .5
XLIM = .625*BCUND
ZLIM = .625*BCUND
SY SCALZ = ZIIP/(1.000001*ZLIM)
LX = NX/2 + 1
PX = 2.*BCUND/NX
Q = PI/XLIM
R = PX/C
E DO 12 I=1,MX
12 KY = (I - LX)*DX
D = D + R*SIN(C*D)
IF (ABS(D).LE.XLIM) GO TO 12
B IF (D.LT.0.) B = -1.
A = 1. - ((C - B*XLIM)*E)**2
C = A**AX
D = B*XLIM + (D - B*XLIM)/C
12 AO(I) = D
DY = NY + 1
DO 22 J=1,KY
22 BO(J) = (KY - J)*DY
A = 1. - D*D
C = A**AY
LZ = SY*D/C + 1
K1 = NZ/2
K2 = 1 + NZ
DZ = 2.*BCUND/NZ
Q = PI/ZLIM
R = LZ/C
IF (KSVM.EQ.0) GO TO 31
LZ = 3
K1 = 2
K2 = NZ + 3
DZ = BOUND/NZ

```

```

      Q = Q + Q
      R = -PZ/Q
      DO 32 K=K1,K2
      D = (K - LZ)*DZ
      IF (ABS(C).LE.ZLIM) GO TO 32
      B = 1.
      IF (D.LT.0.) B = -1.
      A = 1. - (C - B*ZLIM)*E)**2
      C = A**AZ
      D = B*ZLIM + (D - B*ZLIM)/C
      D = SCALZ*C
31  E DO 32 ZO(K)
      RETURN
      END

```

```

C**SUBROUTINE SINGL (NS,NZ,KS,SYM,KTE1,KTE2,FUS,CHORDO,ZS,XLE,YLE,
SUBROUTINE SINGL (NS,NZ,KS,SYM,KTE1,KTE2,FUS,CHORDO,ZS,XLE,YLE,
1 GENERATES SINGULAR LINE FOR SQUARE ROOT TRANSFORMATION
DIMENSION ZS(1),XLE(1),YLE(1),YC(1),ZC(1),YPO(1),ZPO(1),
1 K1 = 1
K2 = NZ + 1
IF (KS,SYM,EC,0) GO TO 11
K1 = 2
K2 = NZ + 3
KTE1 = 3
11 DO 12 K=K1,K2
IF (ZC(K).LT.ZS(1)) KTE1 = K + 1
IF (ZC(K).LE.ZS(NS)) KTE2 = K
12 CONTINUE
CALL SFLIF (1,NS,ZS,XLE,E1,E2,E3,ZC,0,0,IND)
CALL INTPL (KTE1,KTE2,ZC,XO,1,NS,ZS,XLE,E1,E2,E3,0)
S1 = CHORDO*E1(1)
S2 = CHORDO*E1(NS)
CALL SFLIF (1,NS,ZS,YLE,E1,E2,E3,ZC,0,0,IND)
CALL INTPL (KTE1,KTE2,ZC,YO,1,NS,ZS,YLE,E1,E2,E3,0)
T1 = CHORDO*E1(1)
T2 = CHORDO*E1(NS)
XO(KTE1-1) = XO(KTE1) + XO(KTE1+1)
YO(KTE1-1) = YO(KTE1) + YO(KTE1+1)
IF (KS,SYM,NE,0) GO TO 31
N = KTE1 - 1
DO 22 K=K1,N
ZZ = (ZO(K) - ZO(KTE1))/CHORDO
A = EXP(ZZ)
XO(K) = XO(KTE1) + S*ZZ - (S1 - S)*A
YO(K) = YO(KTE1) + T*ZZ - (T1 - T)*A
22 N DO 32 K=N,K2
ZZ = (ZO(K) - ZO(KTE2))/CHORDO
A = EXP(-ZZ)
XO(K) = XO(KTE2) + S*ZZ + (S2 - S)*A
YO(K) = YO(KTE2) + T*ZZ + (T2 - T)*A
32 DO 42 K=K1,K2
YP0(K) = YO(K)
ZPC(K) = ZO(K)
IF (FUS,LE,C,0) GO TO 42
A = .25*(ZO(K)**2 - YO(K)**2) + FUS**2
B = .5*ZO(K)*YO(K)
S = SQRT(A**2 + B**2)

```

```

T = 0.
IF (S.GT.0.) T = .5*ATAN2(B,A)
S = Sqrt(S)
YP0(K) = .5*Y0(K) + S*SIN(T)
ZP0(K) = .5*Z0(K) + S*COS(T)
42 CONTINUE
RETURN
END

```

```

**SUBROUTINE SURF**
SUBROUTINE SURF
1  (ND,NE,NS,NX,NZ,ISYM,KSYM,KTE1,KTE2,
2  YAW,XTEO,XLIM,FX,NP,XS,ZS,SLOPT,TRAIL,
3  AO,XO,ZO,SO,SCAL,ZV,IV,ITE1,ITE2,
  XP,VP,SN,D1,D2,I,NL,
  XPI,NG,SURFACE,AT,MESH,POINTS
  DIMENSION SO(NE,1),XS(ND,1),YS(ND,1),ZS(1),SLOPT(1),TRAIL(1),
1  AO(1),XO(1),ZO(1),SCAL(1),ZV(1),D1(1),D2(1),
2  IV(NE,1),NP(1),ITE1(1),ITE2(1)
3  = 3.141592653589E
  TAN(YAW)
  = XTEO/XLIM**2
  = 2./NX +1
  = NX +1
  = NX +3
  = NZ -ISYM -ISYM
  = 1 -ISYM -ISYM
  = -1 -ISYM
  DO 2 K=1,NZ
1  ITE1(K) = MX
2  ITE2(K) = MX
  DO 2 I=1,NX
1  IV(I,K) = -2
2  SO(I,K) = 0
  K KTE1
  K2 K2 +1
  K1 K2 -1
  R2 I
  IF (ZS(K2) -ZS(K1))/(ZS(K2) -ZS(K1))
2  R1 = 1
2  R2 = -R2
  IF (FIX.EQ.C.) GO TO 31
  IF (R1*XS(1,K1) +R2*XS(1,K2)
  C = SQR(C/SSO)
  DO 26 I=2,NX
1  IF ((AC(1) +.5*DX).LI.-CC) I1 = I +1
2  IF ((AC(1) -.5*DX).LI.CC) I2 = I
  CONTINUE
26 KK = K1
31 RR = R1
41 N ANGL = NP(KK) +PI
  U = 1.
  V = 0.

```

```

DO 42 I=1,N
R = SQRT(XS(I,KK)**2 +YS(I,KK)**2)
IF (R.EC.C.) GO TO 43
ANGL = ANGL +ATAN2((U*YS(I,KK) -V*XS(I,KK)),
1 U (U*XS(I,KK) +V*YS(I,KK)))
V = XS(I,KK)
= YS(I,KK)
R = SQRT(R**2 +R)
XP(I) = R*CCS(.5*ANGL)
YP(I) = R*SIN(.5*ANGL)
GO TO 42
43 ANGL = PI
U = -1.
V = 0.
XP(I) = 0.
YP(I) = 0.
42 CONTINUE
S = AO(I2)/AMIN1(ABS(XP(I)),ABS(XP(N)))
SS = .5/S**2
DO 44 I=1,N
XP(I) = S*XP(I)
YP(I) = S*YP(I)
ANGL1 = ATAN(SLOPT(KK))
ANGL2 = ATAN(YS(I,KK)/XS(I,KK))
ANGL1 = ATAN(YS(N,KK)/XS(N,KK))
ANGL2 = ATAN(-.5*(ANGL1 -TRAIL(KK)))
T1 = ANGL - .5*(ANGL2 +TRAIL(KK))
T2 = TAN(ANGL1)
CALL SFLIF (I,N,XP,YP,D1,D2,D3,I1,I2,I1,T2,O{0,IND)
IF (INC.EC.O) WRITE (6,500) KK,K1,K2,N,FR,R1,R2,ZS(KK)
500 FORMAT (I2,HOBAC,MAPPING,4I10,4G13.4/)
CALL INTPL (I1,I2,AO,SA,I,A,XP,YP,C1,D2,D3,C)
X1 = .25*XS(I,KK)
A = SLOPT(KK)*(XS(I,KK) -X1)
B = 1. / (XS(I,KK) +PI)
ANGL = PI
U = 1.
V = 0.
M = I1 -1
DO 52 I=2,M
XX = SS*AO(I)**2
YY = B*(XX -X1) +A*ALOG(D)/D
R = YS(I,KK) +YY**2
ANGL = SQRT(XX**2 +YY**2)
U = ANGL +ATAN2((U*YY -V*XX),(U*XX +V*YY))
V = XX

```

```

52 R SN(I) = S*SCRT(R +R)
A = R*SIN(.5*ANGL)
B = SLGFT(KK)*(XS(N,KK) -X1)
U = 1./(XS(N,KK) -X1)
V = 0.
M = 0.
DO 54 I=M,NX
XX = I2 +1
YY = SS*AO(I)**2
R ANGL = B*(XX -X1) +A*ALOG(D)/D
U = YS(N,KK)**2 +YY**2
V = SQRT(XX**2 +YY**2)
V = ANGL +ATAN2((U*YY -V*XX),(U*XX +V*YY))
V = XX
V = YY
R SN(I) = S*SCRT(R +R)
A = R*SIN(.5*ANGL)
54 DO 62 I=2,NX
62 SO(I,K) = SO(I,K) +RR*SN(I)
KK IF (KK.EC.K2) GC TC 71
RR = K2
RR = R2
GO TO 41
71 SS (FIX.EQ.0.) = SS0
1SS IF (R1*XS(1,K1) +R2*XS(1,K2))/(AO(I1)**2 -SO(I1,K)**2)
SCAL(K) = SS +SS
ITE1(K) = I1
ITE2(K) = I2
ZV(K) = ZO(K) -TYAW*(XO(K) +SS*AC(I1)*AO(I1))
DO 72 I=I1,I2
72 IV(I,K) = I1 -1
M DO 74 I=1,M
ZZ = ZO(K) -TYAW*(XO(K) +SS*AO(I)*AO(I))
IF (ZZ.GE.ZV(KTEL)) IV(I,K) = IV0
74 CONTINUE
M = I2 +1
DO 76 I=M,MX
ZZ = ZO(K) -TYAW*(XO(K) +SS*AO(I)*AO(I))
IF (ZZ.GE.ZV(KTEL)) IV(I,K) = IV0
76 CONTINUE
K2 = K2 -1
K = K +1
IF (K.LE.KTE2) GC TC 21
K1 = 2
K2 = NZ

```

```

      IF (KSYN.EQ.0) GC TC 81
      K1 = 3
      K2 = NZ +2
      81 SCAL(K) = SCAL(KTE2)
      DO 82 I=1,MX
      ZZ = ZO(K) -TYAW*(XO(K) +SS*A0(I)*A0(I))
      IF (ZZ.LE.ZS(NS).AND.ZZ.GE.ZV(KTE1)) IV(I,K) = IVO
      82 CONTINUE
      K = K +1
      IF (K.LE.K2) GO TO 81
      SCAL(K) = SCAL(KTE2)
      N = KTE2
      IF (YAH.LE.0.) GO TC 93
      IO = ITE1(KTE2) +1
      DO 92 I=IO,LX
      N = N +1
      92 ZV(N) = ZO(KTE2) -TYAW*(XO(KTE2) +SS*A0(I)*A0(I))
      93 ZV(KTE1-1) = ITE1(KTE1)
      ZV(N+1) = ZO(KTE1-1) -TYAW*(XC(KTE1) +SS*A0(I)*A0(I))
      DO 102 K=K1,K2
      DO 104 I=2,MX
      IF (IV(I,K).GT.0) GC TC 104
      IF (IV(I+1,K+1).GT.C.OR.IV(I-1,K+1).GT.C) IV(I,K) = IV1
      IF (IV(I+1,K-1).GT.0.OR.IV(I-1,K-1).GT.C) IV(I,K) = IV1
      104 CONTINUE
      102 IF (SO(LX,K).LT.1.E-05) IV(LX,K) = 0
      N = KTE1 -1
      IF (KSYN.NE.0) RETURN
      DO 112 K=1,N
      112 SCAL(K) = SCAL(KTE1)
      RETURN
      END

```

```

C**SLBROUTINE ESTIM*****
C  SUBROUTINE ESTIM
C
C  INITIAL ESTIMATE OF REDUCED POTENTIAL
COMMON
1  IV(161), 18, 3), SO(161, 35), VOR1(115), ZV(115),
2  IV(161), 35), ITE1(35), ITE2(35),
3  AO(161), 18), XO(35), XC(35), ZC(35), SCAL(35),
4  NX, NY, NZ, KTE1, KTE2, ISYM, KSYM, FUS,
   YAW, CYAW, SYAW, ALPHA, CA, SA, FMACH, N1, N2, N3, IC
   = NX +1
   = NY +2
   = NZ +3
DO 12 I=1, 161
DO 12 J=1, 18
DO 12 K=1, 3
DO 12 G(1, J, K) = 0.
12 DO 22 I=1, 22 ((G(I, J, I), I=1, MX), J=1, MY)
   WRITE (N3) ((G(I, J, I), I=1, MX), J=1, MY)
22 CONTINUE
   K1 = KTE1 -1
   K2 = KTE2 +ITE2(KTE2) -NX/2
DO 32 K=K1, K2
32 VORT(K) = 0.
DO 10 I=1
10 RETURN
END

```

```

C**SUBROUTINE MIXFLO*****
SUBROUTINE MIXFLO*****
SOLUTION OF EQUATION SCHEME FOR MIXED SUBSONIC AND SUPERSONIC FLOW
USING FINITE VOLUME G(161,18,3),ITE1(35),ITE2(35),ZV(115),
COMMON IV(161,18,3),BO(18),XO(35),YC(35),ZO(35),SCAL(35),
NX,NY,NZ,KTE1,KTE2,ISYM,KSYM,FUS,
YAW,CYAW,SYAW,ALPHA,CA,SA,FMACH,N1,N2,N3,IO
COMMON/SPA/ GL(161,18),QQL(161,18),FL(161,18),
UL(161,18),VL(161,18),WL(161,18),
AL(161,18),BL(161,18),CL(161,18),
RESL(161,18),
COMMON/FLC/ PI,P2,P3,FRES,IRES,JRES,KRES,ARES,DG,IG,JG,KG,AG,NSUP
COMMON/SLP/ K,NV,LX,MX,KY,MY,J1,K1,FMACH2,AAO,Q1,Q2,RV,TYAW,IO
LX = NX/2 +1
KY = NX +1
MY = NY +2
J1 = NZ
IF (FMACH.GE.1.) J1 = 3
TYAW = SYAW/CYAW
FMACH2 = FMACH*2
AAO = 1./FMACH*2 +.2
Q1 = 2./F1
Q2 = 1./P2 -1.
TGT = 0.
FRES = 0.
ARES = 0.
DG = 0.
AG = 0.
NSUP = 0
K1 = 3
K2 = NZ +2
IF (KSYM.EQ.1) GO TO 1
K1 = 2
IF (FMACH.GE.1.) K1 = 3
K2 = NZ
1 DD 2 M=2,3
2 READ (N1,ERR=101) ((G(I,J,M),I=1,MX),J=1,MY)
CONTINUE
K = 1
NV = 1
RV = KTE1 -1
DO 12 I=1,MY
DO 12 J=1,MX
G(I,J,1) = G(I,J,2)
GL(I,J) = G(I,J,2)

```

```

12  QQL(I,J) = 0.
    FL(I,J) = 0.
    UL(I,J) = 0.
    VL(I,J) = 0.
    WL(I,J) = 0.
    AL(I,J) = 0.
    BL(I,J) = 0.
    CL(I,J) = 0.
    RESL(I,J) = 0.
    IF (KSYM.NE.0) GO TO 21
    CALL YSKEEP = 1.
    RV
    GO TO 51
21  DO 22 J=1,MY
    DO 22 I=1,NX
    G(I,J,2) = G(I,J,3)
    GL(I,J) = G(I,J,3)
22  READ(N1,ERR=101) ((G(I,J,3),I=1,MX),J=1,MY)
    WRITE(N2) ((G(I,J,1),I=1,MX),J=1,MY)
    K = K + 1
    GO TO 51
31  CALL YSKEEP = 1.
    RV
    IF (K.NE.KTE2) OR (YAH.LE.0.) GO TO 51
    IO = IO + 1
    DO 42 I=10,LX
    M = NX + 2 - I
    V = G(M,KY,1) - G(I,KY,1)
    NV = NV + 1
    VORT(NV) = VORT(NV) + P3*(V - VORT(NV))
42  VORT(NV) = VORT(NV) GO TO 61
51  IF (K.EC.K2) GO TO 61
    DO 52 J=1,MY
    DO 52 I=1,MX
    G(I,J,1) = G(I,J,2)
    G(I,J,2) = G(I,J,3)
52  READ(N1,ERR=101) ((G(I,J,3),I=1,MX),J=1,MY)
    WRITE(N2) ((G(I,J,1),I=1,MX),J=1,MY)
    K = K + 1
    GO TO 21
61  DO 62 N=2,3
    WRITE(N2) ((G(I,J,M),I=1,MX),J=1,MY)
62  COATINLE = FRES/64.
    FRES = AGRES/(64.*TOT)
    AG = AG/TCT
    IO = IO + 1
    RETURN
101 IO = 0

```

RETURN  
END





```

B S T IF (S.GT.0.) I = .5*ATAN2(B,A) +FS)) T = .5*PI
IF (B.EC.0.) AND YM(I).LI.-(FS +FS)) T = -.5*PI
IF (B.EC.0.) SQR7(S)
YM(I) = .5*YM(I) +S*SIN(T)
ZM(I) = .5*ZM(I) +S*COS(T)
A B T = .25*(ZRM(I)**2 -YRM(I)**2) +FS**2
I = .5*ZRM(I)*YRM(I)
SQR7(A**2 +B**2)
IF (S.GT.0.) T = .5*ATAN2(B,A)
SQR7(S)
YRM(I) = .5*YRM(I) +S*SIN(T)
ZRM(I) = .5*ZRM(I) +S*COS(T)
24 DO 32 I=1,NX
31 XX XR(I+1) +XR(I) +XR(I+1) -XR(I) -XR(I)
1 XY +X(I+1) -X(I) +X(I) -X(I) -X(I)
1 XZ +X(I+1) +X(I) +X(I) +X(I) +X(I)
1 YX -(X(I+1) -Y(I) +Y(I) +Y(I) -Y(I)
1 YY +Y(I+1) +Y(I) +Y(I) +Y(I) -Y(I)
1 YZ +Y(I+1) +Y(I) +Y(I) +Y(I) +Y(I)
1 ZX -(Y(I+1) -Z(I) +Z(I) +Z(I) -Z(I)
1 ZY +Z(I+1) +Z(I) +Z(I) +Z(I) -Z(I)
1 ZZ +Z(I+1) +Z(I) +Z(I) +Z(I) +Z(I)
FX XX -YZ#ZZ -YX#ZZ -YX#ZZ
FY XX +YZ#ZZ +YX#ZZ +YX#ZZ
FZ XX -YZ#ZZ -ZY#ZZ -ZY#ZZ
FY Y -ZX#XX -ZY#XX -ZY#XX
FZ Y +ZX#XX +ZY#XX +ZY#XX
FX Z -XZ#YY -XZ#YY -XZ#YY
FY Z +XZ#YY +XZ#YY +XZ#YY
FZ Z -FX#XX -FX#XX -FX#XX
FM(I) = 1./FM(I)
A GX G(I+1,J,2) -G(I,J,2) +G(I+1,J+1,2) -G(I,J+1,2)

```



```

U V W QQL(I,J)
FL(I,J)
UL(I,J)
VL(I,J)
WL(I,J)
AA(I)
QA(I)
OP(I)
P(I)
Q(I)
R(I)
F(I)
FU(I)
FV(I)
FW(I)
IF(QA(I).LE.1.) GC TC 42
NSUP
F(I)
FUL(I)
FVV(I)
FWW(I)
GXX
GYZ
GX Y
GYZ
GZX
FXY
FYZ
FZX
P(I)
Q(I)
R(I)
42 CONTINUE
PF DO 52 I=2,NX
AV RESC(I)*RV
PB PF
IF ((FU(I) +FU(I+1)).LI.0.) PF = -P(I+1)
A UP(I) -UP(I-1) +UM(I) -UM(I-1)
1 +VP(I) +VP(I-1) -VM(I) -VM(I-1)
BV*(UR +UL(I,J))
BV*(VR +VL(I,J))
CV*(WR +WL(I,J))
QQR
= UR
= VR
= WR
= AAQ - .2*QC
= QQ/AA
= 16.*RESO(I)/(F*AA)
= Q(I)
= 0.
= 0.
= 0.
= A*U MAX1(ABS(U),ABS(V),ABS(W))
= F*U ABS(V)
= F*ABS(W)
= 1. GC TC 42
= NSUP +1
= A*(1. -1./QA(I))
= F*U*U
= F*V*V
= F*W*W
= G(I+1,J,2) -G(I,J,2) -G(I-1,J,2) +G(I-1,J,2)
= G(I,J,3) -G(I,J,2) -G(I,J,2) +G(I,J,2)
= G(I+1,J-1,2) -G(I,J,2) -G(I,J,2) +G(I,J,2)
= -G(I+1,J+1,2) -G(I-1,J+1,2) -G(I,J,3) +G(I,J,3)
= G(I+1,J,3) -G(I+1,J,3) -G(I-1,J,3) -G(I-1,J,3)
= .25*F
= F*U*V*GXY
= F*V*W*GYZ
= F*W*U*GZX
= F*U(I)*GXX
= F*V(I)*GYY
= F*W(I)*GZZ
= -P(2)
DO 52 I=2,NX
AV RESC(I)*RV
PB PF
IF ((FU(I) +FU(I+1)).LI.0.) PF = -P(I+1)
A UP(I) -UP(I-1) +UM(I) -UM(I-1)
1 +VP(I) +VP(I-1) -VM(I) -VM(I-1)

```



```

JRES
KR ES
63 AG (ABS(CG(I)),LE.ABS(DG(I)) GO TO 62
IG = K
JG = AG +ABS(CG(I))
KG = CG(I)
62 I DO 72 I=1,NX
71 QP(I) = QM(I)
DP(I) = DM(I)
FP(I) = FM(I)
VP(I) = VM(I)
WP(I) = WM(I)
AP(I) = AM(I)
BP(I) = BM(I)
CP(I) = CM(I)
ABP(I) = ABM(I)
BCP(I) = BCM(I)
CAP(I) = CAM(I)
72 ABCP(I) = ABCM(I)
J (J -KY) 21,81,101
IF (ITE2(K).EQ.MX) I2 = LX
81 IF (ISYM.EQ.1) I2 = NX
DO 82 I=1,NX
IF (ISYM.EQ.1) GO TO 83
IF (IV(I,K).EQ.2.OR.IV(I+1,K).EQ.2) GO TO 83
M QM(I) = NX +1
DM(I) = QP(M)
FM(I) = DP(M)
LM(I) = FP(M)
VM(I) = VP(M)
WM(I) = WP(M)
AM(I) = AP(M)
BM(I) = BP(M)
CM(I) = CP(M)
ABM(I) = ABP(M)
BCM(I) = BCP(M)
CAM(I) = CAP(M)
82 ABCM(I) = ABCP(M)
GO TO 82
QP(I) = QP(I)
DM(I) = DM(I)

```

```

FM(I)      = FP(I)
VM(I)      = UP(I)
WM(I)      = -VP(I)
AM(I)      = WP(I)
BM(I)      = AP(I)
CM(I)      = BP(I)
ABM(I)     = CP(I)
BCM(I)     = -ABP(I)
CAM(I)     = -BCF(I)
ABCM(I)    = CAP(I)
            = -ABCP(I)

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

101 SI
I1
I1
NV
IF (I.NE.I1.OR. ISYM.EQ.1) GO TO 103
V = G(I2,KY,2) - G(I1,KY,2)
NV = NV + 1
VORT(NV) = VORT(NV) + P3*(V - VORT(NV))
N = NV
I = I - 1
103 I
V
IF (IV(I,K).NE.1) GC TC 109
ZZ = ZO(K) - IVAW*(XO(K) + S1*AO(I)*AO(I))
105 IF (ZZ.GE.ZV(N-1)) GO TO 107
N = N - 1
GO TO 105

107 A = (ZZ - ZV(N-1))/(ZV(N) - ZV(N-1))
V = A*VORT(N) + (1. - A)*VCFI(A-1)
105 M = NX + 2 - I
G(I,KY+1,2) = G(M,KY-1,2) - V
G(M,KY+1,2) = G(I,KY-1,2) + V
IF (I.GT.1) GO TO 103
G(I,KY,2) = -.5*V

```

```
G(M,KY,2) = .5*V  
G(LX,KY+1,2) = G(LX,KY-1,2)  
RETURN  
END
```

```

C**SUBROUTINE VELC**
C      SUBROUTINE SURFACE (K,SV,SM,CP,XP,YP,XMAX,XMIN,YMAX,YMIN)
C      CALCULATES SURFACE VELOCITY
C      COMMON
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100
COMMON/SPA/ XL(161,18),YL(161,18),ZL(161,18),XR(161,18),ZR(161,18),
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100
COMMON /UVW/ VU(161,18),VW(161,18),VUW(161,18)
DIMENSION SV(1),SM(1),CP(1),XP(1),YP(1)
PI = 3.1415926535898
Q1 = .2*FMACH**2
T1 = 1./(.7*FMACH**2)
MX = NX + 1
NMAX = 1
AV = 1.5
IF (KSYM.EQ.1.AND.K.EQ.KTEL) GO TO 1
NMAX = 2
AV = K
1 N IF (K.NE.KTEL) GO TO 11
DO 2 J=1,2
DO 2 I=1,NX
X(I,J) = 0.
Y(I,J) = 0.
Z(I,J) = 0.
XR(I,J) = 0.
ZR(I,J) = 0.
11 N X = X0(N)/SCAL(KTEL)
YR = Y0(N)/SCAL(KTEL)
ZR = Z0(N)/SCAL(KTEL)
S2 = SCAL(N)/SCAL(KTEL)
S1 = 5*S2
FS = FUS/SCAL(KTEL)
DO 12 J=1,2
DO 12 I=1,NX
XL(I,J) = X(I,J)
YL(I,J) = Y(I,J)
ZL(I,J) = Z(I,J)
XL(I,J) = XR(I,J)

```



```

U V W X Y Z G U V W X Y Z G U V W X Y Z G U V W X Y Z G
IF (M.EC.2) GO TO 25
M = X(1,1) -X(I-1,1)
XX = Y(I,1) -Y(I-1,1)
YX = Z(I,1) -Z(I-1,1)
ZX = G(I,J,2) -G(I-1,J,2)
GO TO 26
25 IF (N.EC.NMAX) GO TO 27
M = 1
N = X(I+1,1) -X(I,1)
XX = Y(I+1,1) -Y(I,1)
XZ = Y(I+1,1) -Y(I,1)
YZ = Y(I+1,1) -Y(I,1)
ZZ = Z(I+1,1) -Z(I,1)
GX = G(I+1,J,2) -G(I,J,2)
GZ = G(I+1,J,2) -G(I,J,2)
GO TO 27
27 U = AV*L
V = AV*V
W = AV*H
UU(I) = L
VV(I) = V
WW(I) = H
Q SV(I) U*U +V*V +W*H
SM(I) = SQR(T(Q),01(I,1) +Q1*(1. -Q))
CP(I) = FMACH*SV(I)/SQR(T(Q)
XP(I) = T1*(Q*.5 -1.)
YP(I) = SCAL(KTEL)*X(I,1)
I1 ITEL(K)
I2 ITEL2(K)
XMAX = SCAL(KTEL)*X(I1,1)
XMIN = SCAL(KTEL)*X(I1,1)
YMAX = SCAL(KTEL)*Y(I1,1)
YMIN = SCAL(KTEL)*Y(I1,1)
DO 22 I=I1,I2
XMAX1(XMAX,XP(I))
XMIN1(XMIN,XP(I))
YMAX1(YMAX,YP(I))
YMIN1(YMIN,YP(I))
32 RETURN
END

```

```

**SUBROUTINE CPLCT**
SUBROUTINE CPLCT (I3,I4,FMACH,XP,YP,CP,SM,I1,I2,KPLOT)
PLCTS CF AT COMPUTATIONAL MESH POINTS
COMMON /PCKR/ PTCK
COMMON /PARMT3/ XT3(161), YT3(161), ZT3(161), NO1,Z
1 COMMON /UVW/ UU(161), VV(161), WW(161)
COMMON /PRS/ XOCO(161)
COMMON /SFARE/ LINE(90), DUMY(10)
DIMENSION KCDE(2), XP(1), YP(1), CP(1), SM(1)
DATA IST/1H*/
DATA I1/1H*/
NO1 = C
IMIN = I1 + (I2 - I1)/2
CHE = XF(I1) - XP(IMIN)
2 FORPAT(40H)PLOT OF CP AT COMPUTATIONAL MESH POINTS/ 8H CP ,
1 ICHO X , 1CH Y , 7HMAC H NO,
2 7H XCC , 2X , 5+CP* = , F8.4, 2X , 7HCFGRD = , F1C.4)
CPC = ((1. + .2*FMACH**2)**3.5 - 1.)/( .7*FMACH**2)
DO 12 I=1,90
12 LINE(I) = KCDE(I)
CPS = ((15. + FMACH**2)/6.)*3.5 - 1.)/( .7*FMACH**2)
IF (KPLOT.EC.0.CR.KPLOT.GT.1) GO TO 15
WRITE (6,2) CPS,CHE
15 CONTINUE
KS = 3C*(CPO - CPS) + 4.5
IF (KS.GE.1.AND. KS.LE.90) LINE(KS)=IST
DO 22 I=13,14
K = 30*(CFO - CP(I)) + 4.5
= MINC(9C,K)
= KCDE(2)
LINE(K) = XF(I) - XP(IMIN)/CHD
XOCO(I) = XCC
XOCO(I) = XCC
IF (KPLOT.EC.0.CR.KPLOT.GT.1) GO TO 20
WRITE (6,61C) XP(I),YP(I),SM(I),CP(I),XCC,LINE
20 CONTINUE
LINE(K) = KCDE(1)
IF (1.1.I1-OR.I.GT.I2) GO TO 22
NO1 = AC1 + 1
XT3(NO1) = XP(I)
YT3(NO1) = YP(I)
ZT3(NO1) = Z
UT3(NO1) = LU(I)
VT3(NO1) = VV(I)
WT3(NO1) = WW(I)
22 IF (K.EC.KS) LINE(KS) = IST
IF (KPLOT.EC.0.CR.KPLOT.GT.1) GO TO 25
CALL INVRT6

```

25 CONTINUE  
RETURN  
61C FORMAT (2F10.4, F7.4, F8.4, F7.4, 90A1)  
END

C\*\*SLBROUTINE INVRT6\*\*\*\*\*

SUBROUTINE INE INVRT6  
COMMON /PCRR/ PTCK

COMMON /PARMT3/ XT3(161), VT3(161), ZT3(161),  
1 NO1,Z  
DI PEN S I C N P(161,6), VT3(161), VT3(161), NO1,Z  
EQUIVALENCE (XT3(1),P(1,1))

DO 30 J=1,6  
DO 10 I=1,NCI  
M=NO1-I+1  
TEMP(M)=F(1,J)

10 CONTINUE  
DO 20 I=1,NCI  
P(I,J)=TEMP(I)

20 CONTINUE  
30 CONTINUE

1 WRITE (6,100) (1,XT3(I),ZT3(I),VT3(I),UT3(I),  
1 VT3(I),I=1,NO1)

RETURN

100 FORMAT(4H0 I ,1X,6HXI3(I),4X,6HZI3(I),4X,6HVT3(I),  
1 4X,6HWT3(I),4X,6HVT3(I),  
2 6X,4H I ,1X,6HXI3(I),4X,6HVT3(I),4X,6HWT3(I),  
3 4X,6HWT3(I),4X,6HVT3(I),  
4 (14,1X,6G10.3,2X,14,1X,6G10.3))  
END

```

C**SUBROUTINE FORCF (I1,I2,XP,YP,CP,AL,CHCRD,XM,YM,CL,CD,CM)
C
CALCULATES SECTION FORCE COEFFICIENTS
DIMENSION XP(1),YP(1),CP(1)
RAD = 57.29513082
ALPHA = AL/RAD
CL = 0.
CD = 0.
CM = 0.
N = 12
DO 12 I=1,N
  DX = (XP(I+1) - XP(I))/CHORD
  DY = (YP(I+1) - YP(I))/CHORD
  XA = (.5*(XF(I+1) + XP(I)) - XM)/CHORD
  YA = (.5*(YP(I+1) + YP(I)) - YM)/CHORD
  CPA = .5*(CP(I+1) + CP(I))
  DCL = -CPA*DY
  DCC = CL + DCL
  DCD = CD + DCC
  DCM = CM + DCD*YA - DCL*XA
  DCL = CL * COS(ALPHA) - CD * SIN(ALPHA)
  DCD = CL * SIN(ALPHA) + CD * COS(ALPHA)
  DCL = DCL
  DCD = DCD
  DCM = DCM
12
END

```

```

C**SUBROUTINE TCTFOR*****
SUBROUTINE TOTFOR(KTE1,KTE2,CHORD,SCL,SCD,SCM,X0,YPO,ZPO,
1 CALCULATES TOTAL FORCE COEFFICIENTS
DIMENSION CHORD(1),SCL(1),SCD(1),SCM(1),XC(1),YPO(1),ZPO(1)
SPAN = ZPO(KTE2) - ZPO(KTE1)
CL = 0.
CD = 0.
CMP = 0.
CMR = 0.
CMY = 0.
S = 0.
K = KTE1
QL = SCL(K)*CHGRD(K)
QD = SCD(K)*CHORD(K)
QM = CHCRD(K)*(SCM(K)*CHORD(K) - SCL(K)*X0(K) + SCD(K)*YPO(K))
11 DZ = .5*(ZPO(K+1) - ZPO(K))
AZ = .5*(ZPO(K+1) + ZPO(K))
PL = SCL(K+1)*CHCRD(K+1)
PD = SCD(K+1)*CHORD(K+1)
PM = CHCRD(K+1)*(SCM(K+1)*CHCRD(K+1) - SCL(K+1)*X0(K+1) + SCD(K+1)*YPO(K+1))
1 CLA = DZ*(PL + CL)
CDA = DZ*(PD + QD)
CL = CL + CLA
CD = CD + CDA
CMP = CMP + DZ*(PM + QM)
CMR = CMP + AZ*CLA
CMY = CMY + AZ*CDA
S = S + PL
PL = PD
PD = PM
PM = K
IF (K.LT.KTE2) GO TC 11
CL = CL/S
CD = CD/S
CMP = SPAN/S**2
CMR = (CMR)/(S*SPAN)
CMY = (CMY)/(S*SPAN)
RETURN
END

```

```

C**SUBROUTINE REFIN*****
C SUBROUTINE REFIN
  HALVES PESH SIZE
  COMMON
  1 G(161,18,3),SO(161,35),VORT(115),ZV(115),
  2 IV(161,35),ITE1(35),ITE2(35),Z0(35),SCAL(35),
  3 AO(161,18),BO(18),XO(35),YC(35),Z0(35),KSYN,FUS,
  4 NX,NY,NZ,KTE1,KTE2,ISYN,KSYN,FUS,
  MX,NY,NZ,KTE1,KTE2,ISYN,KSYN,FUS,
  YAW,CYAW,SYAW,ALPHA,CA,SA,FMACH,N1,N2,N3,IO
  = NX +1
  = NY +1
  = NY +2
  = NZ +3
  = NX/2 +1
  = NY/2 +2
  = NZ/2 +1
  = 1
  IF (KSYN.EQ.0) GC TC 11
  MZ0 = NZ/2 +3
  REAC (N1,ERR=401) ((G(I,J,1),I=1,MX0),J=1,MX0)
  K = 2
  11 REAC (N1,ERR=401) ((G(I,J,1),I=1,MXC),J=1,MX0)
  J
  J
  21 I = MX0
  I
  31 G(I1,JJ,1) = I -1
  I
  IF (I.GT.0) GO TO 31
  J
  JJ
  IF (J.GT.0) GO TO 21
  DO 42 J=1,KY,2
  42 DO 42 I=2,NX,2
  G(I1,J,1) = .5*(G(I+1,J,1) + G(I-1,J,1))
  DO 54 I=2,NY,2
  54 DO 54 J=1,KY,2
  G(I1,J,1) = .5*(G(I,J+1,1) + G(I,J-1,1))
  DO 52 G(I1,J,1) = 0.
  52 G(I1,NY,1) = 0.
  G(MX,NY,1) = 0.
  WRITE (N2) ((G(I,J,1),I=1,MX),J=1,MX)
  K
  IF (K.LE.MZC) GC TC 11
  REWIND N1
  READ (N2,ERR=401) ((G(I,J,1),I=1,MX),J=1,MX)
  READ (N2,ERR=401) ((G(I,J,3),I=1,MX),J=1,MX)

```

```

WRITE (N1) ((G(I,J,1),I=1,MX),J=1,MY)
K = 1
IF (KSYM.NE.O) K = 2
111 K = K + 1
DO 112 I=1,MY
DO 112 J=1,MX
112 G(I,J,2) = .5*(G(I,J,1) + G(I,J,3))
DO 122 M=2,3
WRITE (N1) ((G(I,J,M),I=1,MX),J=1,MY)
122 CONTINUE
IF (K.EC.MZC) GC TC 201
DO 132 J=1,MY
DO 132 I=1,MX
132 G(I,J,1) = G(I,J,3)
REAC (N2,ERR=401) ((G(I,J,3),I=1,MX),J=1,MY)
GO TO 111
201 REWIND N1
DO 202 M=1,3
REAC (N1,ERR=401) ((G(I,J,M),I=1,MX),J=1,MY)
202 CONTINUE
WRITE (N2) ((G(I,J,1),I=1,MX),J=1,MY)
TYAW = SYAW/CYAW
NV = KTEL - 1
VORT(NV) = 0.
K = 2
IF (KSYM.NE.O) GO TC 251
211 S1 = .5*SCAL(K)
N = NV
I = MX 0 + 1
IF (K.LT.KTEL.OR.K.GT.KTE2) GO TO 231
11 I = ITEL(K)
12 I = ITE2(K)
DO 212 I=1,I2
212 G(I,KY+1,2) = NV + 1 G(I,KY,2) + G(I,KY,2) - G(I,KY-1,2)
NV = G(I2,KY,2) - G(I1,KY,2)
VORT(NV) = NV
N = I1
IF (K.NE.KTE2.OR.YAW.LE.O.) GO TO 231
221 I = I + 1
M = NV + 1 - I
NV = NV + 1
VORT(NV) = G(M,KY,2) - G(I,KY,2)
IF (I.LT.MX0) GC TO 221
231 I = I1 - 1
V = 0.

```

```

233 IF (IV(I,K).NE.1) GO TO 237
      ZZ = ZO(K) -TYAW*(XO(K) +S1*AO(I)*AO(I))
      IF (ZZ.CE.ZV(N-1)) GO TO 235
      N = N -1
      GO TO 233
235 A = (ZZ -ZV(N-1))/(ZV(N) -ZV(N-1))
      V = A*VCRT(N) +1. -A)*VCRT(N-1)
      M = NX +2 -1
      G(I,KY+1,2) = G(M,KY-1,2) -V
      G(M,KY+1,2) = G(I,KY-1,2) +V
      IF (IV(I,K).NE.-1) GO TO 241
      G(I,KY,2) = .5*G(I,KY,1) +.25*(G(I,KY,3) +G(M,KY,3))
      IF (IV(I,K+1).L1.1)
      1G(I,KY,2) = .5*G(I,KY,3) +.25*(G(I,KY,1) +G(M,KY,1))
      G(M,KY,2) = G(I,KY,2)
      G(I,KY-1,2) = .5*G(I,KY,2) +G(I,KY-2,2)
      G(M,KY-1,2) = .5*(G(M,KY,2) +G(M,KY-2,2))
      241 IF (I.GT.1) GO TO 231
      G(I,KY,2) = -.5*V
      G(M,KY,2) = .5*V
      K = K +1
      251 IF (K.EC.MZ) GO TO 261
      DO 252 I=1,MX
      DO 252 J=1,MX
      G(I,J,1) = G(I,J,2)
      G(I,J,2) = G(I,J,3)
      252 WRITE (N2) ((G(I,J,1),I=1,MX),J=1,MX)
      READ (N1,ERR=401) ((G(I,J,3),I=1,MX),J=1,MX)
      GO TO 211
      261 VORT(NV+1) = 0.
      DO 262 M=2,3
      WRITE (N2) ((G(I,J,M),I=1,MX),J=1,MX)
      262 CONTINUE
      REWIND N1
      REWIND N2
      DO 302 K=1,MZ
      READ (N2,ERR=401) ((G(I,J,1),I=1,MX),J=1,MX)
      WRITE (N1) ((G(I,J,1),I=1,MX),J=1,MX)
      302 CONTINUE
      IO = 1
      401 RETURN
      IO = 0
      RETURN
      ENC

```



```

REWIND N1
REWIND N2
DO 42 K=1,MZ
  READ (N2,ERR=51) ((G(I,J,1),I=1,MX),J=1,MV)
  WRITE (N1) ((G(I,J,1),I=1,MX),J=1,MV)
42 CONTINUE
   = 1
   = 0
51 IO RETURN
END

```

```

C**SUBROUTINE SPLIF*****
SUBROUTINE SPLIF(M,N,S,F,FP,FPP,FPPP,KM,VM,KN,VN,MODE,FQM,IND)
C
C Spline fit - JAMES C.
C INTEGAL PLACED IN FPPP IF MODE GREATER THAN 0
C INDSET TO ZERO IF DATA ILLEGAL
DIMENSION S(1),F(1),FP(1),FPP(1),FPPP(1)
INC = 0
K = IABS(N - M)
IF (K - 1) 81,81,1
1 K = (N - M)/K
I = M
J = M + K
DS = S(J) - S(I)
IF (DS) 11,81,11
11 DF = (F(J) - F(I))/DS
IF (KM - 2) 12,13,14
12 U = 5
13 V = 3. * (DF - VM)/DS
GO TO 25
14 U = 0.
GO TO 25
15 V = -1. * VM
GO TO 25
16 U = J + K
17 J = S(J) - S(I)
IF (D*CS) 81,81,23
23 DF = (F(J) - F(I))/DS
B = 1. / (DS + DS + U)
U = B * DS
V = B * (6. * DF - V)
FP(I) = U
FPP(I) = V
IF (J - N) 21,31,21
31 IF (KN - 2) 32,33,34
32 V = (6. * VN - V)/U
GO TO 35
33 V = VN
GO TO 35
34 V = (DS * VN + FPP(I)) / (1. + FPP(I))
35 B = V
36 D = DS
41 DS = S(J) - S(I)

```

```

      FPPP(I)  -FP(I)*V
      = (V -U)/DS
      FPF(I)  = U
      = (F(J) -F(I))/[S  -DS*(V  +L  +U)/6.
      V      = U
      J      = I
      I      = I -K
      IF (J  -N) 41,51,41
51  I      = N -K
      FPPP(N)  = FPPP(I)
      FPP(N)  = B
      FP(N)   = DF  +D*(FPP(I)  +B  +B)/6.
      INC (MCCE) 81,81,61
61  FPPP(J)  = FQM
      V      = FPP(J)
      I      = J
      J      = J  +K
      DS     = S(J)  -S(I)
      U      = FPP(J)
      FPPP(J)  = FPPP(I)  +.5*DS*(F(I)  +F(J)  -[S*DS*(U  +V)/12.)
      V      = U
      IF (J  -N) 71,81,71
      IF (INC.EC.1) GC 71 90
81  WRITET (6,85) INC,MCCE,I,J,K,M,S(I),S(J),DS,L
85  FORMAT (6HOCHECK,6I10,4G13.4/)
86  WRITET (6,86) (S(I),F(I),I=M,N)
9C  RETURN
      ENC

```



```

C**SUBROUTINE VERTEC*****
1 SUBROUTINE VERTEC(I1,I2,XCCD,CP,NRD,ZPC,FMACH,YA,AL,
  SCL,SCD,SCM,K)
C
C SUBROUTINE FOR VERSATEC PLCTTING OF THE PRESSURE COEFFICIENT
C VS NON-CIMENSIONAL CHORD (X/C) FOR EACH SECTION OF THE FINAL
C MESH
C
  REAL PXC(165),PCP(165),XCLQ(85),XCLP(85),CPLO(85),CPUP(85),
1 XCCD(161),CP(161),ZPO(35),SCL(35),SCD(35),SCM(35),
2 FMACH,YA,AL
  INTEGER I,J,NUM,NUM1,I1,I2,NRD,K
C
C INITIALIZE ARRAYS AND DATA TO ZERO.
  NUM = 0.0
  NUM1 = 0.0
  DO 10 I=1,165
    PXC(I) = 0.0
    PCP(I) = 0.0
  10 CONTINUE
  DO 20 J=1,85
    XCLC(J) = 0.0
    CPLC(J) = 0.0
    XCUP(J) = 0.0
    CPUP(J) = 0.0
  20 CONTINUE
C
C READ IN X/C AND CP DATA INTO NEW ARRAY STARTING AT ARRAY
C ELEMENT NUMBER 1
  DO 30 I=1,I2
    PXC(I-1) + 1 = XCCD(I)
    PCP(I-1) + 1 = CP(I)
  30 CONTINUE
  PXC(NRD) = 1.0
C
C PUT THE DATA INTO TWO ARRAYS, ONE FOR THE LOWER SURFACE
C AND ONE FOR THE UPPER SURFACE
  NUM = (NRD-1)/2
  NUM1 = NUM + 1
  DO 40 I=1,NUM1
    XCLC(I) = PXC(I)
    CPLC(I) = PCP(I)
  40 CONTINUE
  DO 50 J=NUM1,NRD
    XCUP(J-NUM) = PXC(J)
    CPUP(J-NUM) = PCP(J)
  50 CONTINUE
C INITIALIZE THE VERSATEC PLCTTER SYSTEM

```

```

C      CALL PLCTS (0.0,0.0,0.0,0)
C
C      SCALE THE DATA TO AN 5.0 X 7.0 INCH SPACE
CALL SCALE (PXC,5.0,NRD,+1)
CALL SCALE (PCP,7.0,NRD,-1)
C
C      DRAW THE X AND Y AXIS
CALL AXIS (1.0,2.0,'X/C',-3,5,0.0,0,PXC(NRD+1),PXC(NRD+2))
CALL AXIS (1.0,2.0,'PRESSURE COEFFICIENT (CF)',25,
>7.0,90.0,PCP(NRD+1),PCP(NRD+2))
C      PUT SCALE FACTORS INTO TWO ARRAYS FOR UPPER AND LOWER SURFACE
XCLO(NUM1+1) = PXC(NRD+1)
XCLO(NUM1+2) = PXC(NRD+2)
CPLC(NUM1+1) = PCP(NRD+1)
CPLC(NUM1+2) = PCP(NRD+2)
C
XCUP(NUM1+1) = PXC(NRD+1)
XCUP(NUM1+2) = PXC(NRD+2)
CPLP(NUM1+1) = PCP(NRD+1)
CPLP(NUM1+2) = PCP(NRD+2)
C
C      PLCT THE DATA PCINTS
CALL NEWPEN (2)
CALL PLCT (1.0,2.0,-3)
CALL LINE (XCLO,CPLC,NUM1,1,-1,1)
CALL LINE (XCUP,CPLP,NUM1,1,-1,1)
C
C      PLACE TITLE AT TOP CF PAGE
CALL NEWPEN (3)
CALL SYMBCL (1,25,7.5,0.2,'SECTION CP DATA ',0.0,16)
CALL SYMBCL (1)
CALL SYMBCL (1,25,7.25,0.1,'* = UPPER SURFACE ',0.0,18)
CALL SYMBCL (1,25,7.0,0.1,'+ = LOWER SURFACE ',0.0,18)
C
C      PLACE THE FCTING INFORMATION ON PLOT
CALL NEWPEN (2)
CALL SYMBCL (0,0,-75.0,1,'SPAN STATION = ',0.0,15)
CALL SYMBCL (999.0,1,ZPO(K),C,0,+3)
CALL SYMBCL (0,0,-1.0,0.1,'MACF ',0.0,6)
CALL SYMBCL (999.0,1,FMAC,0.0,+3)
CALL SYMBCL (2.0,-1.0,0.1,'YAW ',0.0,5)
CALL SYMBCL (999.0,1,YAW,0.0,+3)
CALL SYMBCL (4.0,-1.0,0.1,'ADA ',0.0,5)
CALL SYMBCL (999.0,1,ADA,0.0,+3)
CALL SYMBCL (0,0,-1.25,0.1,'CL ',0.0,6)
CALL SYMBCL (999.0,1,SCL(K),C,0,+3)
CALL SYMBCL (2.0,-1.25,0.1,'CD ',0.0,5)

```

```

CALL NUMBER (995., 999., 0.1, SCD(K), 0.0, +5)
CALL SYMBCL (4.0, -1.25, 0.1, CM, 0.0, +5)
CALL NUMBER (995., 999., 0.1, SCM(K), 0.0, +5)

C END PLCTTING
CALL PLCT (0.0, 0.0, +955)
RETURN
END

```

# APPENDIX F

THIS APPENDIX PRESENTS THE SOURCE CODE FOR THE INTERACTIVE PROGRAM FLO27IN

```

C** FLO27IN FORTRAN-1C/O S/83 C**
C** A PROGRAM TO INTERACTIVELY DEFINE A PASCHALL AERO ENGINEERING C
C** THIS PROGRAM WRITES AN INPUT FILE "FLOC27 DATA.A" TO THE USERS C
C** A DISK WHICH CAN BE SUBSEQUENTLY EXEDITED FOR JOB CARD ENTRY AND C
C** BATCH SYSTEM EXECUTION. C**
C** REAL C**
1 TITLE(16), FMESH, FAX, FNY, FNZ, FPLOT, FIT(3), COVD(3), P10(3), C
2 FMACH, YA, AL, CCO, ZSYM, FNS, SWEEP, DIPTED, FUS, ZS, XL, YL, CHORG, C
3 THICK, AT, FSEC, FN, YSYM, XP(161), YP(161), ZERG, SCWFN, SCWXP(67), C
4 SCWXP(67), SYMFN, SYMXP(59), SYMYP(55), N572FN, N572XP(41), YP(47), C
5 N572VP(41), FPFN, FPXP(31), FLYP(31), FLXFP(49), LISYP(49), N10FN, C
6 A7FN, A7XP(33), A7YP(37), N10YP(37), N34FN, N34XP(45), N35FN, C
7 N1CXP(45), N35YP(45), N64FN, N64XP(45), N64YP(45), N66FN, C
8 N6EXP(45), N63YP(51), N63AFN, N63AXP(51), N63AYP(51), N640FN, C
9 N6EXP(51), N63YP(51), N63AXP(51), N64AXP(51), N64AYP(51), N650FN, C
10 N6EXP(51), N650YP(51), N65AFN, N65AXP(51), N65AYP(51), N660FN, C
11 N6EXP(51), N66CYP(51) C**
12 INTEGER I, IA, J, JL, M, N, NI, FLAG, NUM, SCWNUM, SYMNUM, N57NUM, FPNUM, C
13 F14NUM, A7NUM, LISNUM, N10NUM, N34NUM, N35NUM, N64NUM, N66NUM, C
14 N16NUM, N63NUM, N63AN, N64ON, N64AN, N65ON, N65AN, N66ON C**
C** DATA SCWXP/ C**
1 8043, 7654, 7335, 6962, 6575, 6174, 5764, 5322, 4835, 8382, C
2 4293, 3683, 2987, 2184, 1727, 1218, 0942, 0648, 0335, C
3 0204, 0150, 0096, 0055, 0028, 0006, 0000, 0006, 0028, C
4 0055, 0096, 0150, 0204, 0335, 0648, 0942, 1218, 1727, C
5 2184, 2987, 3683, 4293, 4835, 5322, 5764, 6174, 6575, C
6 6962, 7335, 7694, 8043, 8382, 8713, 9038, 9361, 9522, C
7 9683, 9853, 9937, 1000, 0374, 0381, 0414, 0438, 0461, 0506, 0544, 0579, C
8 SCWXP/ (363926, 0374, 0381, 0414, 0438, 0461, 0506, 0544, 0579, C
1 0611, 0641, 0669, 0695, 0725, 0751, 0780, 0805, 0827, C
2 0847, 0859, 0860, 0843, 0826, 0792, 0764, 0723, 0654, C
3 0607, 0581, 0545, 0507, 0471, 0420, 0378, 0336, 0284, C
4 0246, 0206, 0167, 0136, 0080, 0045, 0007, 0007, 0007, 0007, C
5 0130, 0140, 0126, 0105, 0077, 0044, 0007, 0007, 0007, C
6 0081, 0129, 0178, 0228, 0273, 0310, 0334, 0347, 0346, C
7 0339, 0329, 0308, 0302, 0293, C
8

```

```

C***SYMMETRICAL WING SECTION DATA***
C***DATA SYMFNA59.0/,SYMNUM/59/,
1 1250.0000.8500.7500.7000.6500.6000.
2 5500.5000.4500.3500.2500.2000.1500.
3 1250.1000.0750.0350.0250.0150.0100.
4 0050.0025.0000.0025.0100.0150.0200.0250.
5 0350.0500.0750.1000.1250.1500.2000.3000.
6 3500.4000.4500.5000.6000.6500.7000.7500.
7 8000.8500.9000.9500.1.0000.
SYMVF/ 0025.0075.0130.0180.0290.0340.0375.0415.
1 0450.0475.0505.0525.0550.0555.0540.0500.
2 0480.0447.0398.0330.0265.0230.0167.0140.
3 0090.0065.0000.0065.0090.0140.0167.0200.
4 0030.0026.0033.0039.0047.0080.0050.0040.
5 0550.0055.0055.0055.0055.0055.0045.0041.
6 0375.0034.0029.0024.0018.0013.0007.0002.
7
C***NACA 24-30-0 SECTION DATA***
C***DATA N572XF/1.0/,N57NUM/41/,
1 2500.0000.9000.8000.7000.6000.5000.4000.3000.
2 0500.0025.0000.0025.0050.0100.0200.0300.0100.
3 0500.1000.1500.2000.2500.3000.4000.5000.6000.
4 7000.8000.9000.9500.1.0000.
N572VF/ 0013.0120.0204.0376.0515.0642.0730.0780.0788.
1 0768.0726.0661.0563.0414.0350.0290.0220.0128.
2 0070.0050.0000.0050.0075.0090.0120.0168.0227.
3 0268.0030.0037.0042.0042.0042.0042.0042.0042.
4 0320.0026.0015.0015.0015.0082.0048.0013.
5
C***FFNF/31.0/,FPNUM/31/,
C***DATA FFPX/1.0000.5000.0300.0250.0200.0150.0100.0075.0050.
1 0025.0020.0015.0010.0005.0005.0005.0005.0005.
2 0010.0015.0020.0025.0030.0035.0040.0045.0050.
3 0250.0300.0350.0400.0450.0500.0550.0600.0650.
4 FPF/ 0000.0003.0003.0003.0003.0003.0003.0003.0003.
1 0000.155.0000.160.0000.165.0000.170.0000.175.
2 0000.045.0000.025.0000.025.0000.025.0000.025.
3 0000.085.0000.0100.0000.0100.0000.0100.0000.0100.
4 0000.221.0000.0256.0000.0282.0000.0294.0000.0300.
5 0000.3/
6
C***FFNF/47.0/,F14NUM/47/,
C***DATA F14XP/1.0000.8857.7714.6571.5429.4857.4286.3714.3143.
1 2571.2000.1429.0857.0571.0286.0229.0171.0114.
2

```

```

0057, .0029, .0016, .0005, .0000, .0010, .0020,
0029, .0057, .0114, .0229, .0286, .0571, .1429,
2000, .2571, .3143, .3714, .4286, .4857, .5429, .6571,
.8571, 1.0000,
F14VF/ .0000, .0166, .0320, .0451, .0549, .0571, .0574, .0574, .0573,
.0560, .0514, .0457, .0343, .0280, .0185, .0171, .0143, .0114,
.0076, .0046, .0036, .0023, .0015, .0000, .0013, .0017, .0017,
-.0028, -.0037, -.0059, -.0091, -.0115, -.0137, -.0160, -.0202,
-.0240, -.0289, -.0343, -.0366, -.0375, -.0375, -.0366, -.0343,
-.0301, -.0229, -.0142, -.0059, .0000/

```

```

C*****A-7 WING SECTION - TYPICAL*****
DATA A7FN/33.0, A7ANUM/33/,
A7XF/1.000, .8000, .6000, .4300, .3000, .2000, .1000, .0750, .0500,
.0250, .0125, .0075, .0050, .0025, .0015, .0010, .0005, .0010,
.0015, .0025, .0050, .0075, .0125, .0250, .0500, .0750, .1000,
.2000, .3000, .4300, .6000, .8000, 1.0000,
A7VF/ .0000, .0163, .0301, .0301, .0350, .0309, .0289, .0090, .0032,
-.0031, -.0100, -.0146, -.0170, -.0185, -.0203, -.0213,
-.0219, -.0245, -.0265, -.0270, -.0277, -.0293, -.0301,
-.0313, -.0329, -.0337, -.0339, -.0339, -.0289, -.0289, -.0330,
-.0350, -.0301, -.0163, .0000/

```

```

C*****LIS SAMAN 7769 AIRFOIL SECTION*****
DATA LISFN/49.0, LISNUM/49/,
LISXF/1.000, .9500, .9000, .8000, .7000, .6000, .5000, .4000, .3000,
.2000, .1500, .1000, .0750, .0500, .0350, .0250, .0175, .0125,
.0075, .0050, .0025, .0010, .0005, .0005, .0010, .0010,
.0020, .0025, .0050, .0075, .0125, .0175, .0250, .0350, .0500,
.0750, .1000, .1500, .2000, .3000, .4000, .5000, .6000, .7000,
.8000, .9000, .9500, 1.0000,
LISVF/ .0000, .0041, .0084, .0181, .0216, .0486, .0696, .0897, .0992,
.0526, .0840, .0706, .0615, .0456, .0406, .0334, .0272, .0225,
.0171, .0140, .0102, .0054, .0070, .0055, .0000, .0031, .0048,
-.0070, -.0080, -.0124, -.0137, -.0164, -.0182, -.0201, -.0216,
-.0230, -.0230, -.0216, -.0176, -.0138, -.0106, -.0091, -.0075,
-.0060, -.0045, -.0030, -.0016, -.0008, .0000/

```

```

C*****NACA 0010 AIRFOIL SECTION*****
DATA N1CFN/37.0, N1CNUM/37/,
N10XP/1.000, .5500, .9000, .8000, .7000, .6000, .5000, .4000, .3000,
.2500, .2000, .1500, .1000, .0750, .0500, .0250, .0125, .0050,
.0000, .0050, .0125, .0250, .0500, .0750, .1000, .1500, .2000,
.2500, .3000, .4000, .5000, .6000, .7000, .8000, .9000, .9500,
1.0000,
N10VF/ .0000, .0010, .0067, .0120, .0218, .0305, .0380, .0442, .0485,
.0500, .0495, .0478, .0445, .0390, .0350, .0296, .0217,
.0157, .0058, .0000, -.0098, -.0157, -.0217, -.0296,

```

```

C*****
4 5      -03500,-03902,-04455,-04782,-04952,-05002,-04837,
-04412,-03803,-03053,-02187,-01207,-00672,-00105,
C*****
DATA N34FN/45.0/N34NUM/45/,
1 2 3 4 5      0000,9500,8000,7000,6000,5000,4000,3000,
2000,1500,1000,0750,0500,0250,0125,0075,
0050,0025,0010,0005,0000,0000,0000,0000,
0075,0100,0125,0250,0500,0750,1000,1500,
3000,4000,5000,6000,7000,8000,9000,9500,
N34YP/00100,00856,01156,02767,03733,04433,04856,05000,
04833,04244,03744,03044,02611,02078,01400,00944,
00840,00670,00550,00375,00200,0000,0000,0000,
-00250,-00375,-00700,-00700,-00840,-00944,-01400,
-002078,-002611,-003044,-003744,-04244,-04833,-05000,
-04856,-04433,-03733,-02767,-01556,-00856,-00100/
C*****
DATA N35FN/45.0/N35NUM/45/,
1 2 3 4 5      0000,9500,8000,7000,6000,5000,4000,3000,
2000,1500,1000,0750,0500,0250,0125,0075,
0050,0025,0010,0005,0000,0000,0000,0000,
0075,0100,0125,0250,0500,0750,1000,1500,
3000,4000,5000,6000,7000,8000,9000,9500,
N35YP/00100,01178,02100,03500,04385,04867,05000,04878,
04478,03789,03289,02667,02289,01844,01267,00878,
00775,00650,00505,00370,00250,00180,0000,00180,
-00250,-00370,-00505,-00650,-00775,-00878,-01267,
-001844,-002285,-002667,-003285,-003789,-04478,-04878,
-05000,-04867,-04389,-03500,-02100,-01178,-00100/
C*****
DATA N64FN/45.0/N64NUM/45/,
1 2 3 4 5      0000,9500,8000,7000,6000,5000,4000,3000,
2000,1500,1000,0750,0500,0250,0125,0075,
0050,0025,0010,0005,0000,0000,0000,0000,
0075,0100,0125,0250,0500,0750,1000,1500,
3000,4000,5000,6000,7000,8000,9000,9500,
N64YP/00100,00856,01156,02767,03733,04433,04856,05000,
04856,04411,04056,03533,03178,02722,02044,01511,
01395,01210,01000,00730,00480,00350,00204,
-00480,-00730,-01000,-01210,-00480,-00350,-02044,
-002722,-003178,-003533,-004056,-04411,-04856,-05000,
-04856,-04433,-03733,-02767,-01556,-00856,-00100/
C*****
DATA N66FN/45.0/N66NUM/45/,
1 2 3 4 5      0000,9500,8000,7000,6000,5000,4000,3000,

```

```

2000, .1500, .1000, .0750, .0500, .0250, .0125, .0100, .0075,
.0050, .0025, .0010, .0005, .0001, .0001, .0001, .0001,
.0075, .0100, .0125, .0150, .0175, .0200, .0225, .0250,
.3000, .4000, .5000, .6000, .7000, .8000, .9000, .9500, 1.00,
N66 YF/ .00100, .01656, .03856, .04300, .04889, .05556, .06222,
.04578, .04178, .03856, .03500, .03170, .02833, .02500,
.01320, .01170, .00585, .00300, .00170, .00100, .0005,
- .00480, - .00730, - .00985, - .01240, - .01489, - .02011,
- .02656, - .03085, - .03400, - .03856, - .04178, - .04578,
- .04956, - .05000, - .04889, - .04300, - .02833, - .01656, - .00100/

```

```

C ***** NACA 16-005 SECTION DATA *****
DATA N16FN/45.0/N16NUM/45/,
N16 XF/1.0000, .9500, .9000, .8000, .7000, .6000, .5000, .4000, .3000,
.2000, .1500, .1000, .0750, .0500, .0250, .0125, .0100, .0075,
.0050, .0025, .0010, .0005, .0001, .0001, .0001, .0001,
.0075, .0100, .0125, .0150, .0175, .0200, .0225, .0250,
.3000, .4000, .5000, .6000, .7000, .8000, .9000, .9500, 1.00,
N16 YF/ .00090, .01061, .01888, .03145, .03952, .04376, .04500, .04391,
.04063, .03498, .03101, .02595, .02274, .01882, .01354, .00969,
.00840, .00718, .00595, .00420, .00310, .00215, .0015,
- .00300, - .00420, - .00595, - .00718, - .00840, - .00969, - .01354,
- .01882, - .02274, - .02595, - .02932, - .03145, - .03391,
- .04500, - .04376, - .03952, - .03145, - .01888, - .01061, - .00969/

```

```

C ***** NACA 63-010 SECTION DATA *****
DATA N63FN/51.0/N63NUM/51/,
N63 XF/1.0000, .9500, .9000, .8500, .8000, .7500, .7000, .6500, .6000,
.5500, .5000, .4500, .4000, .3500, .3000, .2500, .2000, .1500,
.1000, .0750, .0500, .0250, .0125, .0075, .0050, .0025, .0010,
.0075, .0125, .0250, .0500, .0750, .1000, .1500, .2000, .2500,
.3000, .3500, .4000, .4500, .5000, .5500, .6000, .6500, .7000,
.7500, .8000, .8500, .9000, .9500, 1.00,
N63 YF/ .00000, .00214, .00604, .01088, .01618, .02166, .02712, .03234,
.03715, .04140, .04496, .04766, .04938, .05000, .04938, .04753,
.04445, .03994, .03362, .02950, .02440, .01756, .01275, .01004,
.00829, .00400, .00162, .00089, .00044, .00022, .00011, .00005,
- .02950, - .03362, - .03766, - .04140, - .04496, - .04753, - .04938,
- .04938, - .04766, - .04140, - .03362, - .02950, - .02712,
- .02166, - .01618, - .01088, - .00604, - .00214, .0.00/

```

```

C ***** NACA 63A010 SECTION DATA *****
DATA N63AFN/51.0/N63AN/51/,
N63A XF/1.0000, .9500, .9000, .8500, .8000, .7500, .7000, .6500, .6000,
.5500, .5000, .4500, .4000, .3500, .3000, .2500, .2000, .1500,
.1000, .0750, .0500, .0250, .0125, .0075, .0050, .0025, .0010,
.0075, .0125, .0250, .0500, .0750, .1000, .1500, .2000, .2500,
.3000, .3500, .4000, .4500, .5000, .5500, .6000, .6500, .7000,

```

```

6      7500, 8000, 8500, 9000, 9500, 1.00/, 02545, 03044, 03517,
N63AYP/ 00021, 00525, 01030, 01535, 02040, 02545, 03044, 03517,
03943, 04311, 04613, 04837, 04968, 04995, 04913, 04714,
04400, 03950, 03324, 02917, 02412, 01737, 01250, 00983,
00815, 0000, 00815, 00583, 00125, 00173, 00412,
002917, 03324, 03950, 04400, 04714, 04913, 04995,
04968, 04837, 04613, 04311, 03943, 03517, 03044,
02545, 02040, 01535, 01030, 00525, 00021/

C*****
C*****NA CA 64-010 SECTION DATA*****
DATA N640FN/51.0/, N640N/51/,
1 N640XF/1.0000, 9500, 9000, 8500, 8000, 7500, 7000, 6500, 6000,
2 5500, 5000, 4500, 4000, 3500, 3000, 2500, 2000, 1500,
3 1000, 0750, 0500, 0250, 0125, 0075, 0050, 0050,
4 0075, 0125, 0250, 0500, 0750, 1000, 1500, 2000, 2500,
5 3000, 3500, 4000, 4500, 5000, 5500, 6000, 6500, 7000,
6 7500, 8000, 8500, 9000, 9500, 1.00/,
7 0000, 00248, 00671, 01176, 01722, 02281, 02827, 03345,
N640YP/ 03820, 04238, 04586, 04843, 04988, 04980, 04864, 04635,
04302, 03842, 03221, 02826, 02343, 01701, 01250, 00985,
00820, 0000, 00820, 00589, 00125, 00170, 00433,
02826, 03221, 03842, 04302, 04639, 04864, 04980,
04988, 04843, 04586, 04238, 03820, 03345, 02827,
02281, 01722, 01176, 00671, 00248, 0.00/

C*****
C*****NA CA 64A010 SECTION DATA*****
DATA N64AFN/51.0/, N64AN/51/,
1 N64AXF/1.0000, 9500, 9000, 8500, 8000, 7500, 7000, 6500, 6000,
2 5500, 5000, 4500, 4000, 3500, 3000, 2500, 2000, 1500,
3 1000, 0750, 0500, 0250, 0125, 0075, 0050, 0050,
4 0075, 0125, 0250, 0500, 0750, 1000, 1500, 2000, 2500,
5 3000, 3500, 4000, 4500, 5000, 5500, 6000, 6500, 7000,
6 7500, 8000, 8500, 9000, 9500, 1.00/,
7 00021, 00541, 01062, 01582, 02103, 02623, 03127, 03597,
N64AYP/ 04021, 04388, 04684, 04995, 04968, 04837, 04606,
04272, 03819, 03199, 02327, 01688, 01225, 00969,
00804, 0000, 00804, 00569, 00125, 00168, 00327,
02905, 03199, 03819, 04272, 04606, 04837, 04968,
04995, 04894, 04684, 04388, 04021, 03597, 03127,
02623, 02103, 01582, 01062, 00541, 00021/

C*****
C*****NA CA 65-010 SECTION DATA*****
DATA N650FN/51.0/, N650N/51/,
1 N650XF/1.0000, 9500, 9000, 8500, 8000, 7500, 7000, 6500, 6000,
2 5500, 5000, 4500, 4000, 3500, 3000, 2500, 2000, 1500,
3 1000, 0750, 0500, 0250, 0125, 0075, 0050, 0050,
4 0075, 0125, 0250, 0500, 0750, 1000, 1500, 2000, 2500,
5 3000, 3500, 4000, 4500, 5000, 5500, 6000, 6500, 7000,
6 7500, 8000, 8500, 9000, 9500, 1.00/,
7 0000, 00248, 00671, 01176, 01722, 02281, 02827, 03345,
N650YP/ 03820, 04238, 04586, 04843, 04988, 04980, 04864, 04635,
04302, 03842, 03221, 02826, 02343, 01701, 01250, 00985,
00820, 0000, 00820, 00589, 00125, 00170, 00433,
02826, 03221, 03842, 04302, 04639, 04864, 04980,
04988, 04843, 04586, 04238, 03820, 03345, 02827,
02281, 01722, 01176, 00671, 00248, 0.00/

```



# C TITLE PAGE AND INSTRUCTIONS

```

CALL FRICMS ('CLRSCRN ')
WRITE (6,410)
WRITE (6,410)
WRITE (6,420)
WRITE (6,430)
WRITE (6,410)
WRITE (6,410)
WRITE (6,440)
WRITE (6,410)

```

## C FIRST LINE INPUT DATA--DEFINE COMPUTATIONAL GRID

```

WRITE (6,450)
WRITE (6,460)
READ (5,1000) TITLE
CONTINUE
WRITE (6,470)
READ (5,*) FMESH
WRITE (6,480)
READ (5,*) FNX
WRITE (6,490)
READ (5,*) FNY
CALL FRICMS ('CLRSCRN ')
WRITE (6,500)
READ (5,*) FNZ
WRITE (6,510)
READ (5,*) FPLOT

```

## C SUMMARY OF FIRST LINE INPUT DATA

```

CALL FRICMS ('CLRSCRN ')
WRITE (6,520)
READ (5,*) ANS
IF (ANS.EQ.2) GO TO 20
WRITE (6,521)
WRITE (6,522) FNX,FNY,FNZ,FMESH,FPLOT
WRITE (6,530)
READ (5,*) ANS
IF (ANS.EQ.1) GO TO 10
20 CONTINUE

```

## C SECOND, THIRD AND FORTH LINES INPUT DATA--ITERATION AND CONVERGENCE TOLERANCE FOR GRID. NUMBER OF LINES EQUAL TO M = FMESH

```

M = IFIX(FMESH)
CALL FRICMS ('CLRSCRN ')

```

```

WRITE (6,450)
DO 30 I=1,M
  IF (I.EC. 1) WRITE (6,541)
  IF (I.EC. 2) WRITE (6,542)
  IF (I.EC. 3) WRITE (6,543)
  WRITE (6,550)
  REAL (5,*) FIT(I)
  WRITE (6,560)
  REAL (5,*) CCVD(I)
  WRITE (6,570)
  REAL (5,*) P10(I)
  CALL FRICMS ('CLRSCRN ')
30 CONTINUE
C-----
C SUMMARY OF SECOND, THIRD AND FORTH LINES INPUT DATA
C-----
  CALL FRICMS ('CLRSCRN ')
  WRITE (6,580)
  READ (5,*) ANS
  IF (ANS.GE.2) GO TO 40
  WRITE (6,581)
  WRITE (6,582) (FIT(I),CCVD(I),P10(I)),I=1,M)
  WRITE (6,590)
  READ (5,*) ANS
  IF (ANS.EQ.1) GO TO 20
40 CONTINUE
C-----
C FIFTH LINE INPUT DATA--MACH NO., YAW ANGLE, AOA, SKIN FRICTION DRAG
C-----
  CALL FRICMS ('CLRSCRN ')
  WRITE (6,450)
  WRITE (6,600)
  READ (5,*) FMACH
  WRITE (6,610)
  READ (5,*) YA
  WRITE (6,620)
  READ (5,*) AL
  WRITE (6,630)
  READ (5,*) CDO
C-----
C SUMMARY OF FIFTH LINE INPUT DATA
C-----
  CALL FRICMS ('CLRSCRN ')
  WRITE (6,640)
  READ (5,*) ANS
  IF (ANS.GE.2) GO TO 50
  WRITE (6,641)
  WRITE (6,642) FMACH,YA,AL,CDO

```

```

WRITE (6,650)
READ (5,*) ANS
IF (ANS.EQ.1) GO TO 40
50 CONTINUE
C-----
C SIXTH LINE INPUT DATA--WING PLANFORM SYMMETRY, NUMBER OF SECTIONS,
C SWEEP, DIHEDRAL ANGLE AND FUSELAGE RADIUS
C-----
CALL FRICMS ('CLRSCRN ')
WRITE (6,450)
WRITE (6,660)
READ (5,*) ZSYM
WRITE (6,670)
READ (5,*) FNS
WRITE (6,680)
READ (5,*) SWEEP
WRITE (6,690)
READ (5,*) DIHED
WRITE (6,700)
READ (5,*) FUS
C-----
C SUMMARY OF SIXTH LINE INPUT DATA
C-----
CALL FRICMS ('CLRSCRN ')
WRITE (6,710)
READ (5,*) ANS
IF (ANS.GE.2) GO TO 60
WRITE (6,711) ZSYM,FNS,SWEEP,DIHED,FUS
WRITE (6,720)
READ (5,*) ANS
IF (ANS.EQ.1) GO TO 50
60 CONTINUE
C-----
C WRITE JCL CARDS TO TOP OF FILE ON USER'S "A" DISK
C-----
WRITE (8,1200)
WRITE (8,1210)
C-----
C WRITE FIRST SIX LINES OF DATA TO FILE ON USER'S "A" DISK
C-----
WRITE (8,1010) TITLE
WRITE (8,1020) FNX,FNY,FNZ,FMESH,FFLCT
WRITE (8,1030) ((FIT(I),COVO(I),P10(I)),I=1,M)
WRITE (8,1040) FMACH,YA,AL,CDO
WRITE (8,1050)
WRITE (8,1060)
WRITE (8,1070)

```

```

WRITE (8,1080)
WRITE (8,1090) ZSYM,FNS,SWEEP,CIHEC,FUS
C-----
C SECTION INFLT DATA
C-----
N = IFIX(FNS)
CALL FRTCMS ('CLRSCRN ')
WRITE (6,450)
WRITE (6,730)
WRITE (6,410)
DO 200 I=1,N
  WRITE (6,760) I
  WRITE (6,770)
  REAC (5,*) ZS
  WRITE (6,780)
  REAC (5,*) XL
  WRITE (6,790)
  REAC (5,*) YL
  WRITE (6,800)
  REAC (5,*) CFGRD
  WRITE (6,810)
  REAC (5,*) THICK
  WRITE (6,820)
  REAC (5,*) AT
  CALL FRTCMS ('CLRSCRN ')
  WRITE (6,830)
  REAC (5,*) FSEC
  WRITE (8,1100)
  WRITE (8,1110) ZS,XL,YL,CHORD,THICK,AT,FSEC
  IF (FSEC.EQ.0.0) GO TO 190
  CALL FRTCMS ('CLRSCRN ')
  WRITE (6,450)
  WRITE (6,740)
  WRITE (6,741)
  WRITE (6,750)
  REAC (5,*) ANS
  FLAG = ANS
  IF (FLAG.GT.0) GO TO 70
  WRITE (6,840)
  REAC (5,*) YSYM
  CALL FRTCMS ('CLRSCRN ')
  WRITE (6,850)
  REAC (5,*) FN
C-----
C USER INPUT X AND Y COORDINATES FOR WING SECTION DEFINING GEOMETRY
C-----
NUM = IFIX(FN)
CALL FRTCMS ('CLRSCRN ')

```

```

180 WRITE (6,450) I
    WRITE (6,860) I
    DO 180 J=1,NUM
        WRITE (6,870) J,XP(J),YP(J)
        READ (5,*) XP(J),YP(J)
        CONTINUE
    IF (YSYM .EQ. 0.0) GC TC 185
    NI = NUM
    J1 = NUM
    DO 183 IA=1,J1
        XP(IA)
        XF(NI-IA) = -XP(IA)
        YF(NI-IA) = -YP(IA)
        CONTINUE
    NUM = NI
    FN = 1 + (FN - 1.0)
    CONTINUE
    WRITE (8,1120) FN
    WRITE (8,1130) FN
    WRITE (8,1140) FN
    WRITE (8,1150) ((XP(J),YP(J)),J=1,NUM)
    GO TO 190
    CONTINUE
70

```

```

C-----
C MENU INPUT X AND Y COORDINATES FOR WING SECTION DEFINING GEOMETRY
C-----

```

```

IF (FLAG .EQ. 1) GO TO 81
IF (FLAG .EQ. 2) GO TO 82
IF (FLAG .EQ. 3) GO TO 83
IF (FLAG .EQ. 4) GO TO 84
IF (FLAG .EQ. 5) GO TO 85
IF (FLAG .EQ. 6) GO TO 86
IF (FLAG .EQ. 7) GO TO 87
IF (FLAG .EQ. 8) GO TO 88
IF (FLAG .EQ. 9) GO TO 89
IF (FLAG .EQ. 10) GO TO 90
IF (FLAG .EQ. 11) GC TO 91
IF (FLAG .EQ. 12) GC TO 92
IF (FLAG .EQ. 13) GC TO 93
IF (FLAG .EQ. 14) GC TO 94
IF (FLAG .EQ. 15) GC TO 95
IF (FLAG .EQ. 16) GC TO 96
IF (FLAG .EQ. 17) GC TO 97
IF (FLAG .EQ. 18) GC TO 98
IF (FLAG .EQ. 19) GC TO 99
IF (FLAG .EQ. 20) GO TO 100
CONTINUE
WRITE (8,1120) FPFN

```

81

```

WRITE (8,1140)
GC TO 190 ((FPXP(J),FPYP(J),J=1,FPNUM)
CCNTINUE
WRITE (8,1120) SYMFN
WRITE (8,1130)
WRITE (8,1140) ((SYMXP(J),SYMYP(J)),J=1,SYMNUM)
WRITE (8,1150)
GC TO 190
CCNTINUE
WRITE (8,1120) SCWFN
WRITE (8,1130)
WRITE (8,1140) ((SCWXP(J),SCWYP(J)),J=1,SCWNUM)
WRITE (8,1150)
GC TO 190
CCNTINUE
WRITE (8,1120) N572FN
WRITE (8,1130)
WRITE (8,1140) ((N572XP(J),N572YP(J)),J=1,N57NUM)
WRITE (8,1150)
GC TO 190
CCNTINUE
WRITE (8,1120) F14FN
WRITE (8,1130)
WRITE (8,1140) ((F14XP(J),F14YP(J)),J=1,F14NUM)
WRITE (8,1150)
GC TO 190
CCNTINUE
WRITE (8,1120) A7FN
WRITE (8,1130)
WRITE (8,1140) ((A7XP(J),A7YP(J)),J=1,A7NUM)
WRITE (8,1150)
GC TO 190
CCNTINUE
WRITE (8,1120) LISFN
WRITE (8,1130)
WRITE (8,1140) ((LISXP(J),LISYP(J)),J=1,LISNUM)
WRITE (8,1150)
GC TO 190
CCNTINUE
WRITE (8,1120) N10FN
WRITE (8,1130)
WRITE (8,1140) ((N10XP(J),N10YP(J)),J=1,N10NUM)
WRITE (8,1150)
GC TO 190
CCNTINUE
WRITE (8,1120) N34FN
WRITE (8,1130)

```

82

83

84

85

86

87

88

89



```

98      WRITE (8,1140) ((N64AXP(J),N64AYP(J)),J=1,N64AN)
        GC TO 190
        CCNTINUE
        WRITE (8,1120) N650FN
        WRITE (8,1130)
        WRITE (8,1140) ((N650XP(J),N650YP(J)),J=1,N650N)
        GC TO 190
        CCNTINUE
        WRITE (8,1120) N65AFN
        WRITE (8,1130)
        WRITE (8,1140) ((N65AXP(J),N65AYP(J)),J=1,N65AN)
        GC TO 190
        CCNTINUE
        WRITE (8,1120) N660FN
        WRITE (8,1130)
        WRITE (8,1140) ((N660XP(J),N660YP(J)),J=1,N660N)
        GC TO 190
        CCNTINUE
        CALL FRTCMS ('CLRSCRN ')
190
200 CONTINUE
C-----
C WRITE THREE LINES TO USER'S FILE INDICATING END OF CALCULATION
C-----
        WRITE (8,1160)
        WRITE (8,1170)
        WRITE (8,1180) ZERC
C-----
C WRITE JCL CARDS TO BOTTOM OF FILE CN USER'S "A" DISK
C-----
        WRITE (8,1220)
        WRITE (8,1230)
C-----
C INDICATE TO USER THAT INPUT IS COMPLETE
C-----
        CALL FRTCMS ('CLRSCRN ')
        WRITE (8,450)
        WRITE (8,880)
        WRITE (8,410)
        WRITE (8,890)
C-----
C FCRMAT STATEMENTS
C-----
        RETURN
410 FORMAT (1X,79H=====)
1=====)

```

```

420 FORMAT (/,25X,24HFLO27 DATA INPUT PROGRAM,/,33X,7HAE-4501//)
430 IF A DISK WHICH THIS PROGRAM INTERACTIVELY WRITES A DATA FILE TO YOU
2AL FLO27,/,)
440 1 FLO27,/,)
1 FLO27,/,)
2AL,/,8X,57HENTER DATA FOR THE POTENTIAL FLOW PROGRAM IN FREE
1 FLO27,/,8X,52HENTER EACH QUESTION THE FORMAT IS GIVEN: (R) -RE
2AL,/,8X,14H(1) - INTEGER,/,8X,45HEXAMPLE: (R,R) INPUT 2.5,6.789
3OR (1) INPUT 5,/,)
450 1 FLO27,/,8X,14H(1) INPUT 5,/,)
1 FLO27,/,8X,14H(1) INPUT 5,/,)
460 1 FLO27,/,8X,14H(1) INPUT 5,/,)
1 FLO27,/,8X,14H(1) INPUT 5,/,)
470 1 FLO27,/,8X,14H(1) INPUT 5,/,)
1 FLO27,/,8X,14H(1) INPUT 5,/,)
480 1 FLO27,/,8X,14H(1) INPUT 5,/,)
1 FLO27,/,8X,14H(1) INPUT 5,/,)
490 1 FLO27,/,8X,14H(1) INPUT 5,/,)
1 FLO27,/,8X,14H(1) INPUT 5,/,)
500 1 FLO27,/,8X,14H(1) INPUT 5,/,)
1 FLO27,/,8X,14H(1) INPUT 5,/,)
510 1 FLO27,/,8X,14H(1) INPUT 5,/,)
1 FLO27,/,8X,14H(1) INPUT 5,/,)
520 1 FLO27,/,8X,14H(1) INPUT 5,/,)
1 FLO27,/,8X,14H(1) INPUT 5,/,)
521 1 FLO27,/,8X,14H(1) INPUT 5,/,)
522 1 FLO27,/,8X,14H(1) INPUT 5,/,)
530 1 FLO27,/,8X,14H(1) INPUT 5,/,)
1 FLO27,/,8X,14H(1) INPUT 5,/,)
541 1 FLO27,/,8X,14H(1) INPUT 5,/,)
542 1 FLO27,/,8X,14H(1) INPUT 5,/,)
543 1 FLO27,/,8X,14H(1) INPUT 5,/,)
550 1 FLO27,/,8X,14H(1) INPUT 5,/,)
1 FLO27,/,8X,14H(1) INPUT 5,/,)
220 1 FLO27,/,8X,14H(1) INPUT 5,/,)
3)
560 1 FLO27,/,8X,14H(1) INPUT 5,/,)
570 1 FLO27,/,8X,14H(1) INPUT 5,/,)
26 FLO27,/,8X,14H(1) INPUT 5,/,)
3NE MESH

```

```

580 FORMAT (1X,55H SUMMARY CF SECOND, THIRD AND FOURTH LINES OF INPUT DATA?,
581 11A2,/,1X,25H=> ENTER 1 = YES; 2 = NO)
582 FORMAT (1X,3HFIT,7X,4HCOVO,6X,3HP10)
583 FORMAT (1X,F5.1,5X,F7.6,3X,F3.1)
584 FORMAT (/,1X,48HCFANGE SECOND, THIRD AND FOURTH LINES INPUT DATA?,
585 1/,1X,25H=> ENTER 1 = YES; 2 = NO)
586 FORMAT (1X,46H=> ENTER FREE STREAM MACH NUMBER (FMACH): (R))
587 FORMAT (1X,40H=> ENTER VAWG ANGLE IN DEGREES (VA): (R))
588 FORMAT (1X,46H=> ENTER ANGLE COEFFICIENT IN DEGREES (AL): (R))
589 FORMAT (1X,58H=> ENTER DRAG COEFFICIENT DUE TO SKIN FRICTION (CDO
590 1): (R),/,5X,48H (UNLESS OTHERWISE AVAILABLE C.O1 IS RECOMMENDED))
591 FORMAT (1X,33H SUMMARY CF FIFTH LINE INPUT DATA?,/,1X,25H=> ENTER
592 11 = YES; 2 = NO)
593 FORMAT (1X,5HF MACH,5X,2HYA,8X,2HAL,8X,3FCD01
594 1X,F4.2,6X,F3.1,7X,F8.6,/)
595 FORMAT (1X,29HCFANGE FIFTH LINE INPUT DATA?,/,1X,25H=> ENTER 1 =
596 1YES; 2 = NO)
597 FORMAT (1X,52H=> ENTER WING PLANFORM SYMMETRY TRIGGER (ZSYM): (R)
598 1/,1X,41H0.0 = YAWED WING HAS NC SPANWISE SYMMETRY,/,1X,38H1.0 = S
599 2WEPT WING HAS SPANWISE SYMMETRY)
600 FORMAT (1X,78H=> ENTER NUMBER OF SECTIONS WHERE WING SECTION GEOM
601 1ETRY IS DEFINED (FNS): (R),/,5X,53H (VALUE MUST BE > OR = 3.0, BUT
602 2NOT GREATER THAN 11.0))
603 FORMAT (1X,58H=> ENTER LEADING EDGE SWEEP ANGLE IN DEGREES (SWEEP
604 1): (R))
605 FORMAT (1X,48H=> ENTER DIHEDRAL ANGLE IN DEGREES (DIHED): (R))
606 FORMAT (1X,36H=> ENTER FUSELAGE RADIUS (FUS): (R),/,5X,33HNOTE: U
607 1SE 0.0 FOR WING-ALCNE CASE)
608 FORMAT (1X,33H SUMMARY OF SIXTH LINE INPUT DATA?,/,1X,25H=> ENTER
609 11 = YES; 2 = NO)
610 FORMAT (1X,4HZ SYM,6X,3HFNS,7X,5HSWEEP,5X,5HCHIED,5X,3HFUS)
611 1X,F3.1,7X,F4.1,6X,F6.3,4X,F6.6,/,/)
612 FORMAT (1X,29HCFANGE SIXTH LINE INPUT DATA?,/,1X,25H=> ENTER 1 =
613 1YES; 2 = NO)
614 FORMAT (5X,66H THE NEXT SET OF INPUT DATA WILL BE REPEATED FOR EACH
615 1 WING SECTION,/,5X,63HX AND Y COORDINATES DEFINING THE WING SECTI
616 2 ON SHAPE ARE ENTERED,/,5X,67HS STARTING WITH THE UPPER SURFACE TRAIL
617 3 ING EDGE AND PROCEEDING ARCUND,/,5X,35H WITH THE LOWER SURFACE TRAILI
618 4 NG EDGE,/,5X,57HX AND Y COORDINATES ARE NORMALIZED WITH THE CHORD
619 5 LENGTH,/,5X,46HWING SECTION DEFINING COOR.,)
620 6,5X,36H BY THE USER OR SECTION FROM A MENU,)
621 1ION COORD. DATA,/,6X,19H1 = FLAT PLATE DATA,/,6X,49H2 = SYMMETRICA
622 2L WING (11% THICKNESS AT 30% CHORD),/,6X,61H3 = SUPERCritical WING (CA
623 3 MBER, 12% THICKNESS AT 30% CHORD),/,6X,51H4 = NACA 24-30 (CA
624 4 MBER, 12% THICKNESS AT 30% CHORD),/,6X,52H5 = F14 WING (CAMBEREC,
625 59.5% CHORD, 12% THICKNESS AT 37% CHORD),/,6X,66H6 = A-7 WING (7 DEG DR COP AT
626 620% CHORD, 7% THICKNESS AT 43% CHORD),/,6X,64H7 = LISSAMAN 7769 AI

```

```

741 7RF CIL (CAMBERED, 1, 1, THICKNESS AT 30% CHORD), /, 6X, 55H8 = NACA 0010
8 (SYMMETRICAL, 10% THICKNESS AT 30% CHORD), /, 6X, 58H9 = NACA 0010-3
94 (SYMMETRICAL, 10% THICKNESS AT 40% CHORD), /, 5X, 59H10 = NACA 0010
>-35 (SYMMETRICAL, 10% THICKNESS AT 50% CHORD), /, 5X, 59H11 = NACA 00
110-64 (SYMMETRICAL, 10% THICKNESS AT 40% CHORD), /, 5X, 59H12 = NACA
20010-66 (SYMMETRICAL, 10% THICKNESS AT 50% CHORD), /, 5X, 58H13 = NACA
3A 16-00C (SYMMETRICAL, 9% THICKNESS AT 50% CHORD), /, 5X, 58H14 = NACA
4A 63-01C (SYMMETRICAL, 10% THICKNESS AT 35% CHORD), /, 5X, 58H15 = NA
5CA 63A010 (SYMMETRICAL, 10% THICKNESS AT 40% CHORD), /, 5X, 58H16 = N
6ACA 64A010 (SYMMETRICAL, 10% THICKNESS AT 40% CHORD), /, 5X, 58H17 =
7NACA 65E-010 (SYMMETRICAL, 10% THICKNESS AT 40% CHORD), /, 5X, 58H18 =
8 NACA 65E-010 (SYMMETRICAL, 10% THICKNESS AT 40% CHORD), /, 5X, 58H18 =
1CHORD), /, 5X, 58H20 = NACA 66-010 (SYMMETRICAL, 10% THICKNESS AT 45%
2 CHORD), /, 5X, 58H20 = NACA 66-010 (SYMMETRICAL, 10% THICKNESS AT 45%
750 FORMAT (1X, 35H==> ENTER DESIRED NUMBER FROM MENU.)
760 FORMAT (15X, 22H***WING SECTION NUMBER, 12, 1X, 13HPARAMETERS***, /, 5X,
142HNOTE: ALL DIMENSIONS MUST BE IN SAME UNITS, //)
770 FORMAT (1X, 60H==> ENTER THE SPANWISE COORDINATE FOR THIS SECTION (
1ZS): (R))
780 FORMAT (1X, 53H==> ENTER SECTION LEADING EDGE X COORDINATE (XL): (R
1))
790 FORMAT (1X, 53H==> ENTER SECTION LEADING EDGE Y COORDINATE (YL): (R
1))
800 FORMAT (1X, 43H==> ENTER SECTION CHORD LENGTH (CHORD): (R))
810 FORMAT (1X, 55H==> ENTER SECTION THICKNESS SCALING FACTOR (THICK):
1(R))
820 FORMAT (1X, 50H==> ENTER SECTION TWIST ANGLE IN DEGREES (AT): (R))
830 FORMAT (1X, 65H==> ENTER FLAG FOR DEFINING NEW WING SECTION GEOMETR
1Y (FSEC): (R), /, 1X, 40H1.0 = DEFINE A NEW WING SECTION GEOMETRY, /, 1
2X, 67H0.0 = COPY THE WING SECTION DEFINITION MUST ENTER 1.0)
840 FORMAT (1X, 38HNOTE: ENTER FLAG FOR INDICATING WHETHER OR NOT, /, 5X, 43H
1THE WING SECTION IS SYMMETRICAL (YSYM): (R), /, 1X, 20H0.0 = ACNSYMMET
2TRICAL, /, 1X, 25H1.0 = SYMMETRICAL SECTION, /, 1X, 53HNOTE: IF SYMMETRI
3CAL SECTION, YOU ONLY HAVE TO INPUT, /, 7X, 46HDEFINING POINTS FOR T
4HE SECTION UPPER SURFACE.)
850 FORMAT (1X, 58H==> ENTER NUMBER OF WING SECTION DEFINING POINTS (FN
1): (R), /, 1X, 38HNOTE: MAXIMUM NUMBER OF POINTS IS 161.)
860 FORMAT (17X, 22H***WING SECTION NUMBER, 12, 1X, 22HX AND Y COORDINATES
1#*//)
870 FORMAT (1X, 67H==> ENTER WING SECTION DEFINING POINT X AND Y COORD
1 (XP, YP): (R, R))
880 FORMAT (5X, 64HTHREE ADDITIONAL DATA LINES WILL BE AUTOMATICALLY WR
1ITTEN TO THE, /, 5X, 43HBOTTOM OF YOUR INPUT FILE. THESE LINES ARE: /
2, 5X, 18HEND OF CALCULATION, /, 5X, 3HFNX, /, 5X, 3HFO.0)
890 FORMAT (5X, 70HTHIS COMPLETES THE INPUT DATA. A FILE WITH <FILENAME
1> <FILETYPE> FLG27, /, 5X, 69HDATA IN HAS BEEN WRITTEN TO YOUR "A" DIS

```



## LIST OF REFERENCES

1. Douglas Aircraft Company Report MDC J5519 (May 1973), A Theoretical Method for Calculating the Aerodynamic Characteristics of Arbitrary Jet-Flapped Wings, Second Edition 1977.
2. Cebeci, T. and Bradshaw, P., Momentum Transfer in Boundary Layers, McGraw-Hill Book Company, 1977.
3. McLean, J. D. and Randall, J. L., Computer Program to Calculate Three-Dimensional Boundary Layer Flows Over Wings with Wall Mass Transfer, NASA Contractor Report 3123, 1979.
4. Randall, J. L. and McLean, J. D., Computer Programs to Calculate Three-Dimensional Boundary Layer Flow Over Wings with Wall Mass Transfer, Program Maintenance Document, Vol. 1, NASA CR-158968, 1978.
5. Randall, J. L. and McLean, J. D., Computer Programs to Calculate Three-Dimensional Boundary Layer Flow Over Wings with Wall Mass Transfer, Program User's, NASA CR-158967, 1978.
6. Bogart, J. R., CDC to IBM FORTRAN IV Conversion Guide, Naval Postgraduate School W. R. Church Computer Center Technical Note No. MVS-04, August 1981.
7. Jameson, A. and Caughey, D. A., A Finite Volume Method for Transonic Potential Flow Calculations, AIAA Paper 77-635, 1977.
8. Anderson, R. F., Determination of the Characteristics of Tapered Wings, pp. 11, NACA Report 572, 1936

INITIAL DISTRIBUTION LIST

|                                                                                                                                         | No. Copies |
|-----------------------------------------------------------------------------------------------------------------------------------------|------------|
| 1. Defense Technical Information Center<br>Cameron Station<br>Alexandria, Virginia 22314                                                | 2          |
| 2. Library, Code 0142<br>Naval Postgraduate School<br>Monterey, California 93943                                                        | 2          |
| 3. Department Chairman, Code 67<br>Department of Aeronautical Engineering<br>Naval Postgraduate School<br>Monterey, California 93943    | 1          |
| 4. Dr. Robert D. Zucker, Code 67Zu<br>Department of Aeronautical Engineering<br>Naval Postgraduate School<br>Monterey, California 93943 | 5          |
| 5. Dr. Oscar Bitlarz, Code 67Bi<br>Department of Aeronautical Engineering<br>Naval Postgraduate School<br>Monterey, California 93943    | 2          |
| 6. LCDR Lowell D. Boaz, Code 67Bo<br>Department of Aeronautical Engineering<br>Naval Postgraduate School<br>Monterey, California 93943  | 2          |
| 7. CDR Jack Paschall III, Code 3107<br>Naval Weapons Center<br>China Lake, California 93555                                             | 2          |

Thesis  
P252  
c.1

Thesis  
P252  
c.1

Paschall

2 0 7 4 2 4

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

9 APR 93

80517

207424

Thesis

P252

c.1

Paschall

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



thesP252

A three-dimensional transonic, potential



3 2768 001 98049 3

DUDLEY KNOX LIBRARY