N70-28418

FORTRAN IV PROGRAM NONCON

NONCONICAL FLOW PAST SLENDER WING BODIES WITH LEADING-EDGE SEPARATION

By F. Y Su

Distribution of this report is provided in the interest of information exchange. Responsibility for the contents resides in the author or organization that prepared it

Prepared under Contract No NAS 2-3523 by
AIR VEHICLE CORPORATION
San Diego, California

for Ames Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

56

NOTICE

FORTRAN IV PROGRAM NONCON

NONCONICAL FLOW PAST SLENDER WING BODIES WITH LEADING-EDGE SEPARATION

By F Y Su

Report No. 358 October 1969

AIR VEHICLE CORPORATION
8873 Balboa Avenue
San Diego, California 92123

Telephone (714) 279-3325

SUMMARY

This report contains a listing and instructions for Program NONCON for which the mathematical development is presented in reference 1. The computer program determines the distribution of lift, pitching moment, and pressure on nonconical slender bodies with leading-edge separation. Input data is obtained from Program SMITH (ref. 2), which is valid for conical wing-bodies only.

The program is written in FORTRAN IV language for the CDC 3600 digital computer at the University of California, San Diego. With only minor modifications the program has been adapted for the IBM series 7040 and 7090 computers at the NASA Ames Research Center.

PROGRAM DESCRIPTION

Program NONCON consists of one main program containing twenty-three subroutines The function of each subroutine has been specified in the Listing

Input Description

Program NONCON is designed to read two kinds of input data 1) regular input data at the initial station from Program SMITH, and 2) restart input data for continuing the program from any station at which it may have stopped prematurely due to the time limitation

The initial input data are read in by using the format-free subroutine INLIST provided by the UCSD Computer Center At most other computer centers and with only slight changes, the data can be read by using the format-free NAMELIST feature of FORTRAN IV version 13

Regular input data specifications. - Definitions of the various parameters used in the regular input are given below

Configuration and sheet parameters: The following data are read in from the Program SMITH output. Units and normalization are the same as for Program SMITH, viz., velocities by the axial free stream component U, and vortex strengths by U tan δ.

NSHAPE	for circular, NSHAPE = 0, for ellipse, NSHAPE = 1
RADIUS	radius of circular body
AZAXIS	semiminor axis of elliptical body
BYAXIS	semimajor axis of elliptical body
N	number of sheet segments, number of pivotal points $(3 < N \leq 25)$
BETA	dihedral angle
DELTA(I)	difference of polar angle between two pivotal points
AE	desired value of angle of attack parameter, $AE = \sin\alpha/\tan\delta$

YSTARV real coordinate of isolated vortex in transformed plane

ZSTARV imaginary coordinate of isolated vortex in transformed plane

GVORT isolated vortex strength, normalized by U tan δ

D(I) distance from the isolated vortex to a sheet point I = even for a pivotal point and I = odd for a middle poi only the pivotal point polar distance must be read in

GS(I) sheet strength at a pivotal point (I = even) and middle point (I = odd), normalized by U tan δ, only the pivotal point strength must be read in

Nonconical shape parameters The following parameters are defined in figure 1 for two configurations, either with or without incidence The input of Program NONCON must be modified for different configurations. In order to read in tabulated data for nonconical shapes, the main program and appropriate subroutines must be changed. Any set of consistent units can be used for the following input variables, except that angular input is to be in degrees. The axial component of the free stream velocity U is taken as unity in Program NONCON, and all output vortex strengths are therefore in effect normalized by U.

ICASE ICASE = 1, double-delta wing configuration
ICASE = 2, ogive nose configuration

LINC LINC = 0 without incidence effect
LINC = 1 with incidence effect

XO initial station and starting point of nonconical section

XF final station of the configuration

DX increment in x

XCF final station of nonconical section

XCFS station shown in figure 1

SPANF semispan at the final station of nonconical section, it is sufficient to know either XCFS or SPANF

DELTA semi-apex angle of initial conical section (deg)

DELTAF semi-apex angle of final conical section (deg)

EDETAF final incidence angle (deg)

Tolerance limits

ACL — upper tolerance limit for all three loops, if the error exceeds this limit, the program will stop automatically

ACC1 — limit of percent change in vortex strength (occurs in LOOP1)

ACC2 — limit on adjustment of zero force condition (occurs in LOOP2)

ACC3 — limit on percent adjustment of sheet shape (occurs in LOOP3)

Iteration limits

NLOOP1 — limit for LOOP1, generally set to 25

NLOOP2 — limit for LOOP2, generally set to 10

NLOOP3 — limit for LOOP3, generally set to 15

Parameters involving pressure·

LPRES — LPRES = 0, no pressure will be calculated, LPRES = 1, pressure will be calculated at prescribed station

NP — total number of pressure stations

NPRE — number of locations on the surface per station at which pressure coefficients are to be calculated (NPRE ≤ 100)

PDX — increment in x when pressure coefficients are to be calculated. DX/PDX must be an integer greater than, or equal to, one

PRPRIT(I) — x station at which pressure coefficients are to be calculated (I ≤ 10)

Restart parameters

IRESTA — IRESTA = 0, program starts at initial station.
IRESTA = 1, program starts at the station where it stopped previously

LPUNCH — LPUNCH = 0, no punched output, LPUNCH = 1, punched output if the program stops due to the time limitation

TIME — maximum running time

END, STOP cards An END card should be inserted after the input data for each separate case After all cases have been specified, a STOP card is needed to end the reading process.

Typical regular input data· Typical regular input data are shown in figure 2 The order of the data can be alternated. Some of the unnecessary data can be neglected as shown in the second case of figure 2

Re-start input data specification. - The restart input data specification is divided into two parts. The data for the first part consists of the same input parameters required for the regular input data specification. The second part consists of a set of prepunched data cards (with a definite format and order) obtained from the output of the previous interrupted calculation. This part must be added after the END card of the regular data input as shown in figure 3.

Output Description

The calculated results will be printed out by the program at the initial station as shown in figure 4. Print out at intermediate stations is similar, except for omission of the title and some parameters A description of the important output parameters is given below.

General output specification. - General output parameters are defined below. Quantities referring to geometric properties of the model or to properties of the vortex sheet in either the physical cross-flow plane ($Z = y + i z$) or transformed cross-flow plane ($Z^* = y^* + i z^*$) are defined in figures 1 through 3 of references 1 and 2.

X	x station of the configuration
S	semispan
AZAXIS	semiminor axis of elliptical body at initial station
BYAXIS	semimajor axis of elliptical body or radius of circular body at initial station
YO	y coordinate of the tip of the wing in physical plane
ZO	z coordinate of the tip of the wing in physical plane
GVORT	Γ_v, vortex strength of the isolated vortex (based on $U = 1$)
YSTARV	y_v^*, real coordinate of the isolated vortex in transformed plane

ZSTARV	z_v^*, imaginary coordinate of the isolated vortex in transformed plane
RSV	$r_v^* = \left(y_v^{*2} + z_v^{*2}\right)^{1/2}$
THETASV	$\theta_v^* = \tan^{-1}(z_v^*/y_v^*)$ (deg)
YV	y_v, real coordinate of the isolated vortex in physical plane
ZV	z_v, imaginary coordinate of the isolated vortex in physical plane
RV	$r_v = \left(y_v^2 + z_v^2\right)^{1/2}$
THETAV	$\theta_v = \tan^{-1}(z_v/y_v)$ (deg)
H	polar angle of each middle and pivotal point (deg)
I	index for vortex sheet points (I = even for a pivotal point and I = odd for a middle point)
GS(I)	strength of the sheet segments (based on U = 1)
YS(I)	y_i^*, real coordinates of the sheet segments in the transformed plane
ZS(I)	z_i^*, imaginary coordinates of the sheet segments in the transformed plane
RS(I)	$r_i^* = \left(y_i^{*2} + z_i^{*2}\right)^{1/2}$
THETAS(I)	$\theta_i^* = \tan^{-1}(z_i^*/y_i^*)$ (deg)
D(I)	polar distances of the sheet segments
Y(I)	y_i, real coordinates of the sheet segments in the physical plane

Z(I) z_i, imaginary coordinates of the sheet segments in the physical plane

R(I) $r_i = \left(y_i^{\,2} + z_i^{\,2}\right)^{1/2}$

THETA(I) $\theta_i = \tan^{-1}(z_i/y_i)$ (deg)

Additional output are printed in self-explanatory titles, such as ANGLE OF ATTACK, LINEAR LIFT COEFF, etc.

Pressure output specification. - At prescribed axial stations PRPRIT, upper and lower surface pressure coeffients are printed out Pressure output parameters are described below.

Y real coordinate of a point on the body or wing surface

Z imaginary coordinate of a point on the body or wing surface

CP pressure coefficient

* * = 0 reliable data, * = 1 unreliable data due to the undefined log 0

Additional output - If the tolerance limit has been relaxed inside a loop, the maximum tolerance will be printed out before the general output as shown in figure 5.

When the indicator LPUNCH has been set equal to 1, a set of data cards with prescribed format will be punched out if the program has stopped due to time limits. This set of data must be kept in order, and will be used as a part of the restart input data

PROGRAM LISTING

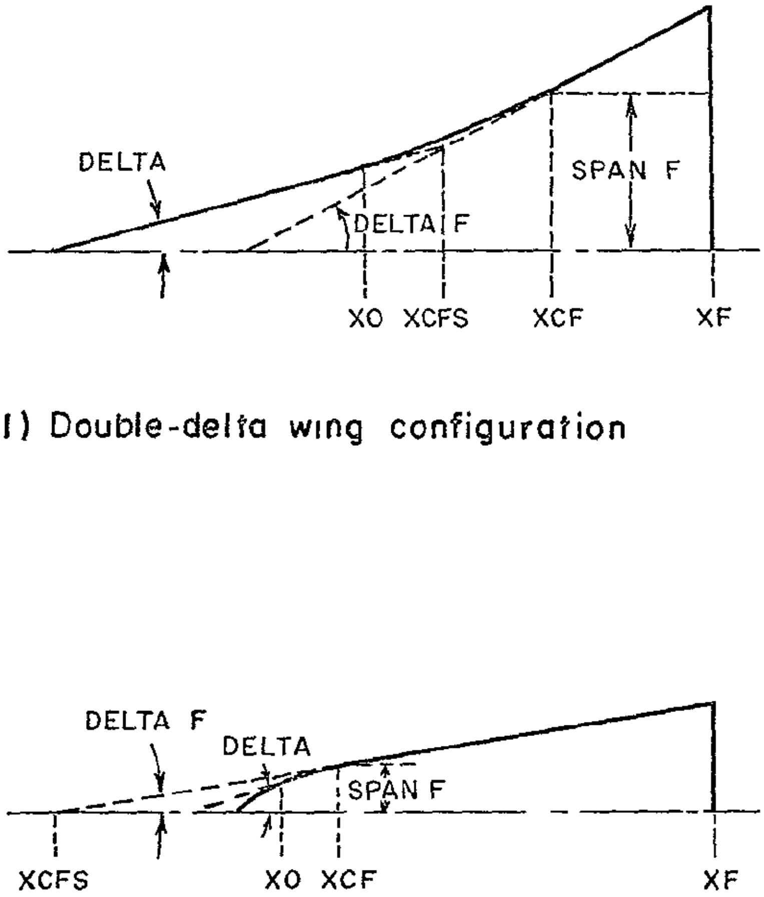
The FORTRAN IV listing of Program NONCON is given on pages 15 to 53

Air Vehicle Corporation
San Diego, California
October 1969

REFERENCES

1 Wei, M H Y., Levinsky, E. S., and Su, F. Y. Nonconical Theory of Flow Past Slender Wing-Bodies With Leading-Edge Separation, NASA CR

2. Levinsky, E. S. and Wei, M. H. Y. Nonlinear Lift and Pressure Distribution of Slender Conical Bodies With Strakes at Low Speeds, NASA CR-1202, Oct , 1968.



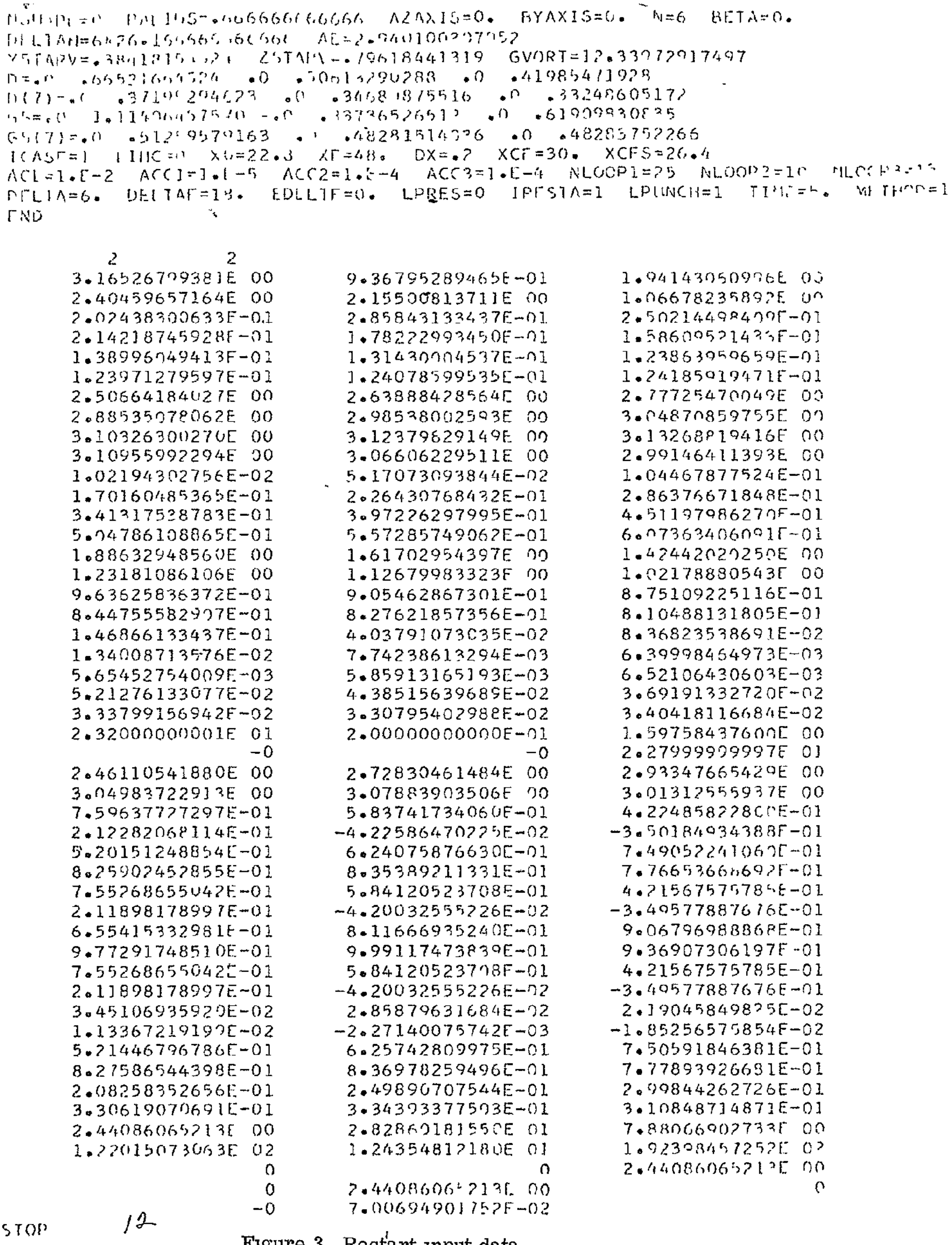
1) Double-delta wing configuration

2) Ogive nose configuration

Figure 1. - Nonconical shape parameters for double-delta wing and ogive nose configurations

```
NSHAPE=0  RADIUS=.66666666666  AZAXIS=0.  BYAXIS=0.  N=6  BETA=0.
DELTAH=6626.166666666666  AF=2.94010037652
YSTARV=.28418154324  ZSTARV=.70618441319  GVORT=12.33972917497
D=.0  .60521664'24  .0  .50612290288  .0  .41985471928
D(7)=.0  .37119296623  .0  .34628875516  .0  .33248601172
GS=.0  1.11405457571  .0  .82736516612  .0  .61909830836
GS(7)=.0  .51299577153  .0  .48281514336  .0  .48283712266
ICASE=1  LINC=0  X0=22.8  XF=48.  DX=.2  XCF=30.  XCFS=26.4  SPACE=0.
DELTA=6.  DELTAF=18.  EDELTF=0.
ACC=1.E-2  ACC1=1.E-5  ACC2=1.E-4  ACC3=1.E-4  NLOOP1=25  NLOOP2=10  NLOOP3=1
LPRES=1  NPRE=50  PRPRIT=19.2 24. 28.8 33.6 38.4  NP=5  PDX=.2
IRESTA=0  LPUNCH=1  TIME=5.  METHOD=1
FORCE=0. 0. 0.  . 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
END
LINC=0  ICASE=2  X0=3.4  XCFS=-14.4  XCF=12.  XF=33.6  DX=1.
DELTA=18.  DELTAF=6.
AF=.5344341375331
YSTARV=.15792270424  ZSTARV=.51714011261  GVORT=1.47501072939
D=.0  .39605552792  .0  .26797423290  .0  .20531561114
D(7)=.0  .17188530743  .0  .15265317118  .0  .13850551681
GS=.0  .21951842704  .0  .17973149339  .0  .11870438857
GS(7)=.0  .08873588095  .0  .07641131677  .0  .06985709291
LPRES=0  IRESTA=0  TIME=3.  METHOD=2
FORCE=.265  .6199  5.836 .5183  .402  .33367  .294  .272  .258
FORCE(10)=.451  .398  .319  .270  .254  .251
END
STOP
```

Figure 2. - Regular input data

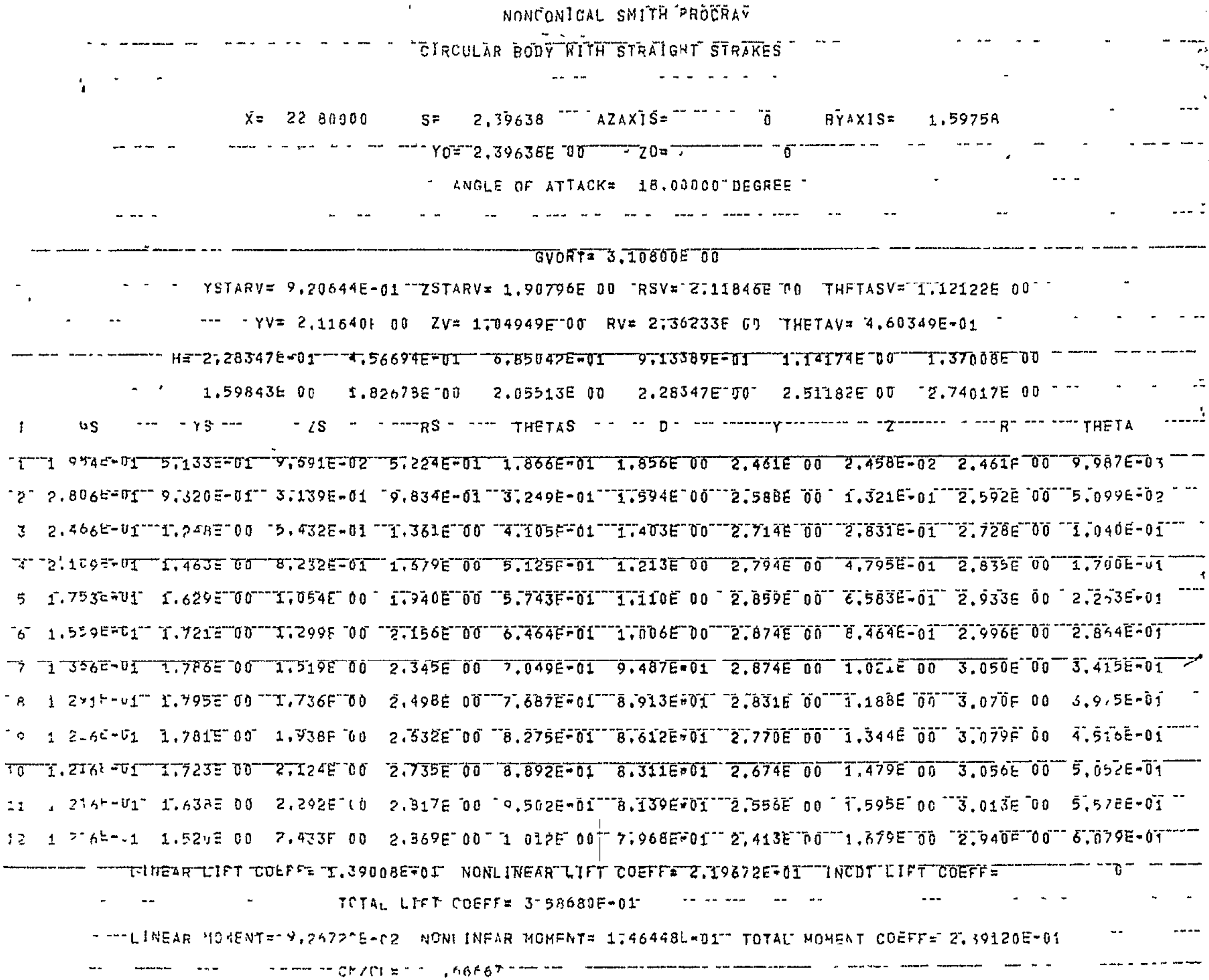


```
NGRDPT=0  RADIUS=.66666666666  AZAXIS=0.  BYAXIS=0.  N=6  BETA=0.
DELTAN=6476.16666666666   AE=2.94010022797952
YSTARV=.38418151521   ZSTAR=-.79618441319   GVORT=12.33972917497
D=.0  .66521664524  .0  .50613290288  .0  .41985471928
D(7)=.0  .37190294623  .0  .34688875516  .0  .33248605172
GS=.0  1.11490457540 -.0  .33736526512  .0  .61909830835
GS(7)=.0  .51259579163  .0  .48281514026  .0  .48285752266
ICASE=1  IINC=0  X0=22.3  XF=48.  DX=.2  XCF=30.  XCFS=26.4
ACL=1.E-2  ACC1=1.E-5  ACC2=1.E-4  ACC3=1.E-4  NLOOP1=25  NLOOP2=10  NLOOP3=15
DELTA=6.  DELTAF=18.  EDELTF=0.  LPRES=0  IPESTA=1  LPUNCH=1  TIME=5.  METHOD=1
END

       2              2
     3.16526799381E 00        9.36795289465E-01        1.94143050996E 00
     2.40459657164E 00        2.15500813711E 00        1.06678235892E 00
     2.02438300633E-01        2.85843133437E-01        2.50214498409E-01
     2.14218745928E-01        1.78222993450E-01        1.58609521435E-01
     1.38996049413E-01        1.31430004537E-01        1.23863959659E-01
     1.23971279597E-01        1.24078599535E-01        1.24185919471E-01
     2.50664184027E 00        2.63888428564E 00        2.77725470049E 00
     2.88535078062E 00        2.98538002593E 00        3.04870859755E 00
     3.10326300270E 00        3.12379629149E 00        3.13268819416E 00
     3.10955992294E 00        3.06606229511E 00        2.99146411393E 00
     1.02194302756E-02        5.17073093844E-02        1.04467877524E-01
     1.70160485365E-01        2.26430768432E-01        2.86376671848E-01
     3.41317528783E-01        3.97226297995E-01        4.51197986270E-01
     5.04786108865E-01        5.57285749062E-01        6.07363406091E-01
     1.88632948560E 00        1.61702954397E 00        1.42442020250E 00
     1.23181086106E 00        1.12679983323E 00        1.02178880543E 00
     9.63625836372E-01        9.05462867301E-01        8.75109225116E-01
     8.44755582907E-01        8.27621857356E-01        8.10488131805E-01
     1.46866133437E-01        4.03791073035E-02        8.36823538691E-02
     1.34008713576E-02        7.74238613294E-03        6.39998464973E-03
     5.65452754009E-03        5.85913165193E-03        6.52106430603E-03
     5.21276133077E-02        4.38515639689E-02        3.69191332720E-02
     3.33799156942E-02        3.30795402988E-02        3.40418116684E-02
     2.32000000001E 01        2.00000000000E-01        1.59758437600E 00
                    -0                       -0        2.27999999997E 01
     2.46110541880E 00        2.72830461484E 00        2.93347665429E 00
     3.04983722913E 00        3.07883903506E 00        3.01312555937E 00
     7.59637727297E-01        5.83741734060E-01        4.22485822800E-01
     2.12282068114E-01       -4.22586470225E-02       -3.50184934388E-01
     5.20151248854E-01        6.24075876630E-01        7.49052241060E-01
     8.25902452855E-01        8.35389211331E-01        7.76653666692E-01
     7.55268655042E-01        5.84120523708E-01        4.21567575785E-01
     2.11898178997E-01       -4.20032555226E-02       -3.49577887676E-01
     6.55415332981E-01        8.11666935240E-01        9.06796988868E-01
     9.77291748510E-01        9.99117473839E-01        9.36907306197E-01
     7.55268655042E-01        5.84120523708E-01        4.21567575785E-01
     2.11898178997E-01       -4.20032555226E-02       -3.49577887676E-01
     3.45106935920E-02        2.85879631684E-02        2.19045849825E-02
     1.13367219199E-02       -2.27140075742E-03       -1.85256575854E-02
     5.21446796786E-01        6.25742809975E-01        7.50591846381E-01
     8.27586544398E-01        8.36978259496E-01        7.77893926681E-01
     2.08258352656E-01        2.49890707544E-01        2.99844262726E-01
     3.30619070691E-01        3.34393377593E-01        3.10848714871E-01
     2.44086065213E 00        2.82869181550E 01        7.88066902733E 00
     1.22015073063E 02        1.24354812180E 01        1.92398457252E 02
                     0                        0        2.44086065213E 00
                     0        2.44086065213E 00                        0
                    -0        7.00694901752E-02
STOP
```

Figure 3. Restart input data

Figure 3. Restart input data



```
                              NONCONICAL SMITH PROGRAM

                          CIRCULAR BODY WITH STRAIGHT STRAKES

               X=  22 80000      S=   2,39638      AZAXIS=           0      BYAXIS=   1.59758
                             YO= 2,39638E 00      ZO=                0
                            ANGLE OF ATTACK=  18.00000 DEGREE

                                     GVORT= 3,10800E 00
           YSTARV= 9,20644E-01  ZSTARV= 1,90796E 00  RSV= 2,11846E 00  THETASV= 1,12122E 00
                 YV= 2,11640E 00  ZV= 1,04949E 00  RV= 2,36233E 00  THETAV= 4,60349E-01
        H= 2,28347E-01   4,56694E-01   6,85042E-01   9,13389E-01   1,14174E 00   1,37008E 00
           1,59843E 00   1,82678E 00   2,05513E 00   2,28347E 00   2,51182E 00   2,74017E 00
 I     GS        YS         ZS         RS       THETAS       D          Y          Z          R        THETA
 1  1 954E-01  5,133E-01  9,591E-02  5,224E-01  1,866E-01  1,856E 00  2,461E 00  2,458E-02  2,461E 00  9,987E-03
 2  2.806E-01  9,320E-01  3,139E-01  9,834E-01  3,249E-01  1,594E 00  2,588E 00  1,321E-01  2,592E 00  5,099E-02
 3  2,466E-01  1,248E 00  5,432E-01  1,361E 00  4,105E-01  1,403E 00  2,714E 00  2,831E-01  2,728E 00  1,040E-01
 4  2,109E-01  1,463E 00  8,232E-01  1,679E 00  5,125E-01  1,213E 00  2,794E 00  4,795E-01  2,835E 00  1,700E-01
 5  1,753E-01  1,629E 00  1,054E 00  1,940E 00  5,743E-01  1,110E 00  2,859E 00  6,583E-01  2,933E 00  2,253E-01
 6  1,559E-01  1,721E 00  1,299E 00  2,156E 00  6,464E-01  1,006E 00  2,874E 00  8,464E-01  2,996E 00  2,854E-01
 7  1 356E-01  1,786E 00  1,519E 00  2,345E 00  7,049E-01  9,487E-01  2,874E 00  1,021E 00  3,050E 00  3,415E-01
 8  1 291E-01  1,795E 00  1,736E 00  2,498E 00  7,687E-01  8,913E-01  2,831E 00  1,188E 00  3,070E 00  3,975E-01
 9  1 266E-01  1,781E 00  1,938E 00  2,632E 00  8,275E-01  8,612E-01  2,770E 00  1,344E 00  3,079E 00  4,516E-01
10  1,216E-01  1,723E 00  2,124E 00  2,735E 00  8,892E-01  8,311E-01  2,674E 00  1,479E 00  3,056E 00  5,052E-01
11  1 216E-01  1,638E 00  2,292E 00  2,817E 00  9,502E-01  8,139E-01  2,556E 00  1,595E 00  3,013E 00  5,578E-01
12  1 216E-01  1,520E 00  2,433E 00  2,869E 00  1 012E 00  7,968E-01  2,413E 00  1,679E 00  2,940E 00  6,079E-01
       LINEAR LIFT COEFF= 1,39008E-01  NONLINEAR LIFT COEFF= 2,19672E-01  INCDT LIFT COEFF=             0
                          TOTAL LIFT COEFF= 3 58680E-01
      LINEAR MOMENT= 9,26720E-02  NONLINEAR MOMENT= 1,46448E-01  TOTAL MOMENT COEFF= 2,39120E-01
                        CM/CL =    ,66667
```

Figure 4. - Output at the initial station

```
         ERROR IS TOO LARGE  ERMAX= 1.15749E-04  ACC3= 3.00000E-04

                              THE RESULTS AT X=  23.20000 AND S=   2.44086 ARE

                                          GVORT= 3.16527E 00

                YSTARV= 9.36795E-01  ZSTARV= 1.94143E 00  RSV= 2.15563E 00  THETASV= 1.12122E 00

                    YV= 2.15501E 00  ZV= 1.06678E 00  RV= 2.40460E 00  THETAV= 4.59659E-01

I     S           YS          ZS          RS         THETAS        D           Y           Z           R         THETA

1  2.024E-01  5.929E-01  1.011E-01  5.326E-01  1.909E-01  1.886E 00  2.507E 00  2.562E-02  2.507E 00  1.022E-02
2  2.858E-01  9.463E-01  3.244E-01  1.002E 00  3.296E-01  1.617E 00  2.635E 00  1.364E-01  2.639E 00  5.171E-02
3  2.502E-01  1.269E 00  5.563E-01  1.386E 00  4.131E-01  1.424E 00  2.762E 00  2.896E-01  2.777E 00  1.045E-01
4  2.142E-01  1.468E 00  8.398E-01  1.708E 00  5.138E-01  1.232E 00  2.844E 00  4.886E-01  2.885E 00  1.702E-01
5  1.752E-01  1.656E 00  1.074E 00  1.974E 00  5.754E-01  1.127E 00  2.909E 00  6.702E-01  2.985E 00  2.264E-01
6  1.540E-01  1.750E 00  1.323E 00  2.194E 00  6.473E-01  1.022E 00  2.925E 00  8.612E-01  3.049E 00  2.854E-01
7  1.370E-01  1.916E 00  1.547E 00  2.385E 00  7.055E-01  9.636E-01  2.924E 00  1.039E 00  3.103E 00  3.413E-01
8  1.314E-01  1.825E 00  1.767E 00  2.541E 00  7.692E-01  9.055E-01  2.881E 00  1.208E 00  3.124E 00  3.972E-01
9  [illegible]  1.811E 00  1.972E 00  2.678E 00  8.278E-01  8.751E-01  2.819E 00  1.366E 00  3.133E 00  4.512E-01
10  [illegible]  1.752E 00  2.161E 00  2.782E 00  8.894E-01  8.448E-01  2.722E 00  1.504E 00  3.110E 00  5.048E-01
11  [illegible]  1.766E 00  2.332E 00  2.866E 00  9.503E-01  8.276E-01  2.602E 00  1.622E 00  3.066E 00  5.573E-01
12  1.242E-01  1.546E 00  2.476E 00  2.919E 00  1.013E 00  8.105E-01  2.456E 00  1.707E 00  2.991E 00  6.074E-01
         LINEAR LIFT COEFF= 1.39303E-01   NONLINEAR LIFT COEFF= 2.19817E-01   INCDT LIFT COEFF=               0
                              TOTAL LIFT COEFF= 3.59120E-01
         LINEAR MOMENT= 9.29659E-02   NONLINEAR MOMENT= 1.46593E-01   TOTAL MOMENT COEFF= 2.39558E-01
                              CM/CL=      .66707
```

NOT REPRODUCIBLE

Figure 5. - General output

```
      PROGRAM NONCON

C     THIS PROGRAM IS DESIGNED TO CALCULATE VORTEX SHEETS BY SMITH
C     NONCONICAL THEORY

      COMMON/1/RADIUS, AZAXIS, BYAXIS, S, BETA
      COMMON/2/NSHAPE
      COMMON/3/DX, X0, XXX, XF
      COMMON/4/ALPHA, AL, DELTA, DELTAF, SPANF, XSPF, TANAF, SINA1, SINA
     12, COSA2, COSA3, COSBT
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/6/GVORT, GVORTO, GS(50), GSO(50), RS(51), THETAS(50), YS(50
     1), ZS(50), D(51), H(52), R(50), RO(50), THETA(50), Y(50), Z(50), Z
     2SV, THETSV, YSTARV, ZSTARV, YV, YVO, ZV, ZVO, RV, RVO, THETAV
      COMMON/7/RO, THETAO, YO, ZO
      COMMON/9/EPS
      COMMON/10/LINC
      COMMON/11/DWDSGR(25), DWDSGI(25), F(25), COSPHI(25), SINPHI(25),
     1BSD(25)
      COMMON/12/X, NPRE, IPRIT
      COMMON/18/ACL
      COMMON/20/XEPF, EDELTF
      COMMON/21/DELTAH(50)
      COMMON/22/DGVDX, DYSVDX, DZSVDX, DDDX(25), DGDX(25), K, XY, TIME,
     1PRPRIT(10)
      COMMON/23/ICASE
      COMMON/24/NLOOP1, ACC1, SUM1(25), SUM10(25), SUM11(25), SSUM1(25),
     1 SUM2(25), SUM20(25), SUM21(25), SSUM2(25), CSPIX0(25), DWDSX0(25)
     2, RX0(25)
      COMMON/25/NLOOP2, ACC2, STEP
      COMMON/26/NLOOP3, ACC3, COTPHI(25), THETAO(50)
      COMMON/27/DH(50), HD(25), DD(50), DRR(50), HH(25)
      COMMON/28/A, B, C, P

      DIMENSION FORCE(15), DGDX1(25), DGDX2(25), DDDX1(25), DDDX2(25), D
     10(25)

      NAMELIST/INPUT/NSHAPE, RADIUS, AZAXIS, BYAXIS, N, BETA, DELTAH, AF
     1, YSTARV, ZSTARV, GVORT, D, GS, ICASE, LINC, X0, XF, DX, XCF, XCFS
     2, SPANF, DELTA, DELTAF, EDELTF , ACL, ACC1, ACC2, ACC3, NLOOP1, NL
     3OOP2, NLOOP3, LPRES, NPRE, PRPRIT, NP, PDX, IRESTA, LPUNCH, TIME,
     4METHOD, FORCE

   20 NW=INLIST(INPUT, LL)
      IF (NW .EQ. 3HEND) 21, 22
   21 PRINT 19
      CALL RESTAT (LPUNCH, 1)
      CALL CONST
      MP=0
      IF (LPRES .EQ. 0) GO TO 69
      KCONT=0
      KDEL=DX/PDX
   69 STEP=.001
      ALPHAA=ALPHA/RAD
      XSPF=XCF
      XEPF=XCF
      KSTOP=(XF-X0)/DX+2.
      IF (LINC .EQ. 0) 58, 59
   58 IF (ICASE .EQ. 1) 13, 14
   13 SPANF=(XCF-XCFS)*DELTAF+XCFS*DELTA
      GO TO 23
   14 SPANF=(XCF-XCFS)*DELTAF
```

```
23   CALL FIT (X0, XCF, SPANE, DELTA, DELTAF, A, B, C, P)
     GO TO 60
59   SPANE=(XCF-XCF0)*FDELTF
     CALL FIT (X0, XCF, SPANE, 0., FDELTF , A, B, C, P)
60   IF (IRESTA .EQ. 0) 72, 73
72   X=X0
     XX0=X0
     XXX=X0
     CALL NCONSH (0)
     CALL SETUP
     IF (NSHAPE .EQ. 0) 33, 34
33   BYAXIS=RADIUS
     IF (BETA .EQ. 0) 35, 36
35   IF (RADIUS .EQ. 0) 38, 39
38   PRINT 51
     GO TO 50
39   PRINT 52
     GO TO 50
36   PRINT 53
     GO TO 50
34   IF (BETA .EQ. 0.) 45, 46
45   PRINT 54
     GO TO 50
46   PRINT 55
50   IF (ICASE .EQ. 1) 15, 16
15   XY=X
     IPRIT=2
     GO TO 17
16   XY=X+14.4
     IPRIT=1
17   PRINT 1, XY,S, AZAXIS, BYAXIS, YO, ZO, ALPHAA
     CALL COEIGR (1)
     PRINT 2, GVORT, YSTARV, ZSTARV, RSV, THETSV,  YV, ZV, RV, THETAV,
    1(H(I), I=1, N2)
     DO 25 I=1, N2
25   PRINT 3, I, GS(I), YS(I), ZS(I), RS(I), THETAS(I), D(I), Y(I), Z(I
    1), R(I), THETA(I)
     CALL LIFT (0)
     CALL USEFUL
     DO 26 I=1, N
     L=2*I-1
     CSPIX0(I)=COSPHI(I)
     DWDSX0(I)=DWDSGR(I)
26   PX0(I)=R(L)
     DO 27 I=1, N
     SUM1(I)=0.
     SUM2(I)=0.
     SUM10(I)=COSPHI(I)
27   SUM20(I)=DWDSGR(I)
     DGVDX1=GVORT/X0
     DGVDX2=FORCE(3)*DELTAF**2
     DGVDX=DGVDX1
     DYVDX1=YSTARV/X0
     DYVDX2=FORCE(1)*DELTAF
     DYSVDX=DYVDX1
     DZVDX1=ZSTARV/X0
     DZVDX2=FORCE(2)*DELTAF
     DZSVDX=DZVDX1
     DO 28 I=1, N
     DGDX1(I)=GS(2*I)/X0
     DGDX2(I)=FORCE(I+9)*DELTAF**2
     DDDX1(I)=D(2*I)/X0
```

```
      DDDX2(I)=FORCE(I+3)*DELTAF
      DGDX(I)=DGDX1(I)
   28 DDDX(I)=DDDX1(I)
      KK=1
      GO TO 74
   73 CALL RESTAT (LPUNCH, 2)
      KK=K+1
   74 DO 29 K=KK, KSTOP
      GVORT0=GVORT
      YSV0=YSTARV
      ZSV0=ZSTARV
      RV0=RV
      YV0=YV
      ZV0=ZV
      DO 11 I=1, N2
      GS0(I)=GS(I)
      RO(I)=R(I)
   11 THETA0(I)=THETA(I)
      DO 32 I=1, N
      DO(I)=D(2*I)
   32 COTPHI(I)=COSPHI(I)/SINPHI(I)
      X0=X
      X=X+DX
      XXX=X
      IF (ICASE .EQ. 1) 18, 24
   18 XY=X
      GO TO 37
   24 XY=X+14.4
   37 GVORT=GVORT0+DGVDX*DX
      YSTARV=YSV0+DYSVDX*DX
      ZSTARV=ZSV0+DZSVDX*DX
      DO 30 I=1, N
      I2=2*I
      D(I2)=DO(I)+DDDX(I)*DX
   30 GS(I2)=GS0(I2)+DGDX(I)*DX
      CALL NCONSH (1)
      CALL SETUP
      CALL LOOP3 (ISTOP3)
      IF (ISTOP3.EQ. 1) GO TO 8
      PRINT 7, XY, S
      PRINT 41,GVORT, YSTARV, ZSTARV, RSV, THETSV,  YV, ZV, RV, THETAV
      DO  5 I=1, N2
    5 PRINT 3, I, GS(I), YS(I), ZS(I), RS(I), THETAS(I), D(I), Y(I), Z(
     1), R(I), THETA(I)
      CALL LIFT (1)
      IF (LPRES .EQ. 0) 42, 63
   63 IF (MP .EQ. 0) 64, 43
   64 IF (XY-PRPRIT(IPRIT)+DX+.001) 42, 65, 65
   65 DELTAX=DX
      DX=PDX
      MP=1
   43 K=K-1
      KCONT=KCONT+1
      IF (KCONT/KDEL .LT. 1) GO TO 66
      K=K+1
      KCONT=0
   66 IF (ABS((XY+DX)-PRPRIT(IPRIT)) .LE. .001) 47, 48
   47 CALL PREPRS (1)
      GO TO 42
   48 IF (ABS(XY-PRPRIT(IPRIT)) .LE. .001) 49, 42
   49 CALL PREPRS (2)
      DX=DELTAX
      MP=0
   42 DO 12 I=1, N
      SUM1(I)=SSUM1(I)
      SUM2(I)=SSUM2(I)
      SUM10(I)=SUM11(I)
   12 SUM20(I)=SUM21(I)
      IF (METHOD .EQ. 2) GO TO 67
      DGVDX=(GVORT-GVORT0)/DX
      DYSVDX=(YSTARV-YSV0)/DX
      DZSVDX=(ZSTARV-ZSV0)/DX
      DO 66 I=1, N
      L=2*I
      DGDX(I)=(GS(L)-GS0(L))/DX
   68 DDDX(I)=(D(L)-D0(I))/DX
      GO TO 61
   67 IF (X .LE. XCF) 56, 57
   56 XD=(X-XX0)/(XCF-XX0)
      DGVDX=DGVDX1+XD*(DGVDX2-DGVDX1)
      DYSVDX=DYVDX1+XD*(DYVDX2-DYVDX1)
      DZSVDX=DZVDX1+XD*(DZVDX2-DZVDX1)
      DO 31 I=1, N
      DGDX(I)=DGDX1(I)+XD*(DGDX2(I)-DGDX1(I))
   31 DDDX(I)=DDDX1(I)+XD*(DDDX2(I)-DDDX1(I))
      GO TO 61
   57 DGVDX=DGVDX2
      DYSVDX=DYVDX2
      DZSVDX=DZVDX2
      DO 62  I=1, N
      DGDX(I)=DGDX2(I)
   62 DDDX(I)=DDDX2(I)
   61 CALL RESTAT (LPUNCH, 3)
   29 CONTINUE
      GO TO 20
   22 IF (NW .EQ. 4HSTOP) 97, 98
   97 STOP
   98 PRINT 6
   99 NW=INLIST(INPUT, LL)
      IF (NW .EQ. 3HEND) 20, 99
    8 PRINT 9
      GO TO 20
    1 FORMAT (1H0, 23X, 2HX=F10.5, 5X, 2HS=F10.5, 5X, 7HAZAXIS=F10.5, 5X
     1, 7HBYAXIS=F10.5//42X, 3HYO=E12.5, 5X, 3HZO=E12.5//44X, 16HANGLE O
     2F ATTACK=F10.5, 1X, 6HDEGREE//)
    2 FORMAT (1H0, 51X, 6HGVORT=E12.5//20X, 7HYSTARV=E12.5, 2X, 7HZSTARV
     1=E12.5, 2X, 4HRSV=E12.5, 2X, 8HTHETASV=E12.5//25X, 3HYV=E12.5, 2XT
     2 3HZV=E12.5, 2X, 3HRV=E12.5, 2X, 7HTHETAV=E12.5//17X, 2HH=6(E12.5,
     3 2X)//19X, 6(E12.5, 2X)//2X, 1HI, 5X, 2HGS, 9X, 2HYS, 9X, 2HZS, 9X
     4, 2HRS, 7X, 6HTHETAS, 8X, 1HD, 10X, 1HY, 10X, 1HZ, 10X, 1HR, 7X, 5
     5HTHETA)
    3 FORMAT (1H0, I2, 10(1X, E10.3))
    6 FORMAT (1H1, 20X, 26H THIS CASE WILL BE IGNORED)
    7 FORMAT (1H0//36X, 'THE RESULTS AT X='F10.5, 1X, 'AND S='F10.5, 1X,
     1, 3HARE)
    9 FORMAT (1H0, 73HTHE PROGRAM IS STOPED DUE TO THE NONCONVERGENCE IN
     1 ONE OF THE THREE LOOPS)
   19 FORMAT (1H1, 48X, 24HNONCONICAL SMITH PROGRAM)
   41 FORMAT (1H0, 51X, 6HGVORT=E12.5//20X, 7HYSTARV=E12.5, 2X, 7HZSTARV
     1=E12.5, 2X, 4HRSV=E12.5, 2X, 8HTHETASV=E12.5//25X, 3HYV=E12.5, 2X,
     2 3HZV=E12.5, 2X, 3HRV=E12.5, 2X, 7HTHETAV=E12.5//2X, 1HI, 5X, 2HGS
     3, 9X, 2HYS, 9X, 2HZS, 9X, 2HRS, 7X, 6HTHETAS, 8X, 1HD, 10X, 1HY, 1
     40X, 1HZ, 10X, 1HR, 7X, 5HTHETA)
   51 FORMAT (1H0, 52X, 16HFLATE PLATE CASE//)
```

```
      SUBROUTINE CONST
C       THIS SUBROUTINE CALCULATES SOME USEFULL CONSTATNTS.

      COMMON/1/RADIUS, AZAXIS, BYAXIS, S, BETA
      COMMON/4/ALPHA, AE, DELTA, DELTAF, SPANF, XSPF, TANAF, SINA1, SINA
     12, COSA2, COSA3, COSBT
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/6/GVOPT, GVORT0, GS(50), GS0(50), RS(51), THETAS(50), YS(50
     1), ZS(50), D(51), H(52), R(50), R0(50), THETA(50), Y(50), Z(50), R
     2SV, THETSV, YSTARV, ZSTARV, YV, YV0, ZV, ZV0, RV, RV0, THETAV
      COMMON/20/XEPF, EDELTF
      COMMON/21/DELTAH(50)
      COMMON/27/DH(50), HD(25), DD(50), DRR(50), HH(25)

      N2=2*N
      N2M1=N2-1
      N2P1=N2+1
      PI=3.141592653589793
      TWOPI=2.*PI
      PIO2=PI/2.
      THPIO2=PI+PIO2
      RAD=PI/180.
      EDELTF=TAN(EDELTF*RAD)
      DELTA=TAN(DELTA*RAD)
      DELTAF=TAN(DELTAF*RAD)
      TANAF=AE*DELTA
      ALPHA=ASIN(TANAF)
      SINA1=SIN(ALPHA)
      SINA2=SIN(ALPHA)**2
      COSA2=COS(ALPHA)**2
      COSA3=COS(ALPHA)**3
      COSBT=COS(BETA*RAD)
      H(2)=DELTAH(1)*RAD
      DO 23 I=2, N
      I2=2*I
   23 H(I2)=H(I2-2)+DELTAH(I)*RAD
      H(1)=H(2)/2.
      DO 33 I=3, N2M1, 2
   33 H(I)=(H(I+1)+H(I-1))/2.
      H(N2P1)=H(N2)
      H(N2+2)=H(N2)
      DH(1)=H(2)
      DO 34 I=2, N2
   34 DH(I)=H(I+1)-H(I-1)
      NM1=N-1
      HD(1)=0.
      HD(2)=(H(6)+H(4)-H(2))/4.
      HD(N)=(3.*H(N2)-2.*H(N2-2)-H(N2-4))/4.
      DO 35 I=3, NM1
      I2=2*I
   35 HD(I)=(H(I2+2)+H(I2)-H(I2-2)-H(I2-4))/4.
      HH(1)=H(4)
      DO 36 I=2, N
   36 HH(I)=H(2*I+2)-H(2*I-2)
      RETURN
      END
```

```
52   FORMAT (1H0, 40X, 35HCIRCULAR BODY WITH STRAIGHT STRAKES//)
53   FORMAT (1H0, 40X, 35HCIRCULAR BODY WITH DIHEDRAL STRAKES//)
54   FORMAT (1H0, 39X, 37HELLIPTICAL BODY WITH STRAIGHT STRAKES//)
55   FORMAT (1H0, 39X, 37HELLIPTICAL BODY WITH DIHEDRAL STRAKES//)
     END
```

```
      SUBROUTINE RESTAT (L, M)

C       THIS SUBROUTINE READS AND PUNCHES NECERSSARY DATA FOR RESTARTING
C        THE PROGRAM NONCONSM.
C          L=0. THE PROGRAM NONCONSM DOES NOT NEED TO RESTART.
C          L=1, THE PROGRAN NONCONSM WILL BE RESTARTED AT ANY TIME.
C          M=1. PREPARES FOR RESTARTING.
C          M=2, READS AND CHECKES THE RESTARTING DATA.
C          M=3, PUNCHES OUT THE NECERSSARY DATA FOR RESTARTING.

      COMMON/3/DX, X0, XXX, XX0
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/6/GVORT, GVORT0, GS(50), GS0(50), RS(51), THETAS(50), YS(50
     1), ZS(50), D(51), H(52), R(50), RO(50), THETA(50), Y(50), Z(50), R
     2SV, THETSV, YSTARV, ZSTARV, XV, YV0, ZV, ZV0, RV, RV0, THETAV
      COMMON/7/RO, THETAO, YO, ZO
      COMMON/11/DWDSGR(25), DWDSGI(25), E(25), COSPHI(25), SINPHI(25), A
     1BSD(25)
      COMMON/12/X, NPPE, IPRIT
      COMMON/17/RADISS, ZAXISS, YAXISS
      COMMON/19/DBDX, DRDDX
      COMMON/22/DGVDX, DYSVDX, DZSVDX, DDDX(25), DGDX(25), K, XY, TIME,
     1PRPRIT(10)
      COMMON/24/NLOOP1, ACC1, SUM1(25), SUM10(25), SUM11(25), SSUM1(25),
     1 SUM2(25), SUM20(25), SUM21(25), SSUM2(25), CSPIX0(25), DWDSX0(25)
     2, RX0(25)
      COMMON/29/SUM3, SUM30, SUM31, SSUM3, SUM4, SUM40, SUM41, SSUM4, SU
     1M5, SUM50, SUM51, SSUM5, SUM6, SUM60, SUM61, SSUM6

      NAMELIST/SOO/K, IPRIT, GVORT, YSTARV, ZSTARV, RV, YV, ZV, GS, R, T
     1HETA, D, DGVDX, DYSVDX, DZSVDX, DGDX, DDDX, X, DX, RADISS, ZAXISS,
     2 YAXISS, XX0, RX0, CSPIX0, DWDSX0, COSPHI, SINPHI, SUM10, SUM1, SU
     3M20, SUM2, SUM30, SUM3, SUM40, SUM4, SUM50, SUM5, SUM60, SUM6, RO,
     4 THETAO, YO, ZO, DBDX, DRDDX

      IF (L .EQ. 0) 1, 2
    1 RETURN
    2 GO TO (3, 4, 5) M
    3 CALL STARTIME
      LTIME0=0
      LTIME=TIME*60000
      RETURN
    4 READ   96, K, IPRIT
      READ 95,
     1GVORT, YSTARV, ZSTARV, RV, YV, ZV, (GS(I), I=1, N2), (R(I), I=1, N
     22), (THETA(I), I=1, N2), (D(I), I=1, N2), DGVDX, DYSVDX, DZSVDX, (
     3DGDX(I), I-1, N), (DDDX(I), I=1, N), X, DX, RADISS, ZAXISS, YAXISS
     4,  XX0,  (RX0(I), I=1, N),  (CSPIX0(I),  I=1, N),  (DWDSX0(I),  I=
     51, N), (COSPHI(I), I=1, N), (SINPHI(I), I=1, N), (SUM10(I), I=1, N
     6), (SUM1(I), I=1, N), (SUM20(I), I=1, N), (SUM2(I), I=1, N), SUM30
     7, SUM3, SUM40, SUM4, SUM50, SUM5, SUM60, SUM6, PO, THETAO, YO, ZO,
     8 DBDX, DRDDX
      CALL OUTLIST (SOO)
      RETURN
    5 LTIME1=LAPSTIME(DUMMY)
      IF ((XY+2.*DX)-PRPRIT(IPRIT) .LE. .01) 44, 45
   44 IF (LTIME -4.*LTIME1+3.*LTIME0-30000)70, 71, 71
   45 IF (LTIME-2*LTIME1+LTIME0-30000) 70, 71, 71
   71 LTIME0=LTIME1
      RETURN
   70 PUNCH 96, K, IPRIT
      PUNCH 95,
    1GVORT, YSTARV, ZSTARV, RV, YV, ZV, (GS(I), I=1, N2), (R(I), I=1, N
    22), (THETA(I), I=1, N2), (D(I), I=1, N2), DGVDX, DYSVDX, DZSVDX, (
    3DGDX(I), I=1, N), (DDDX(I), I=1, N), X, DX, RADISS, ZAXISS, YAXISS
    4, XX0, (RX0(I), I=1, N), (CSPIX0(I), I=1, N), (DWDX0(I), I=
    51, N), (COSPHI(I), I=1, N), (SINPHI(I), I=1, N), (SUM10(I), I=1, N
    6), (SUM1(I), I=1, N), (SUM20(I), I=1, N), (SUM2(I), I=1, N), SUM30
    7, SUM3, SUM40, SUM4, SUM50, SUM5, SUM60, SUM6, R0, THETA0, Y0, Z0,
    8 DRDX, DRDDX
 94  FORMAT (1X, 3(E23.15, 2X))
 95  FORMAT (3E23.15)
 96  FORMAT (2I10)
     STOP
     END
```

```
      SUBROUTINE FIT (X0, XCF, SF, DELTA, DELTAF, A, B, C, P)
C     THIS SUBROUTINE FITS THE CURVE OF NONCONICAL SECTION BY A
C     THIRD ORDER POLYNOMIAL.

      S0=X0*DELTA
      X02=X0**2
      X03=X0**3
      XF2=XCF**2
      XF3=XCF**3
      A3=XF3-X03
      B3=XF2-X02
      C3=XCF-X0
      F3=SF-S0
      A1=3.*X02*C3-A3
      B1=2.*X0*C3-B3
      F1=DELTA*C3-F3
      A2=3.*B3
      B2=2.*C3
      F2=DELTAF-DELTA
      DV=A1*B2-A2*B1
      A=(B2*F1-B1*F2)/DV
      B=(A1*F2-A2*F1)/DV
      C=(F3-A3*A-B3*B)/C3
      P=S0-X03*A-X02*B-X0*C
      RETURN
      END
      SUBROUTINE NCONSH (L)

C     THIS SUBROUTINE CALCULATES SOME INPUT DATA FOR EACH SECTION OF X.
C     L=0, FOR X=X0.  L=1, FOR ANY X.

      COMMON/1/RADIUS, AZAXIS, BYAXIS, S, BETA
      COMMON/2/NSHAPE
      COMMON/3/DX, X0, X, XX0
      COMMON/4/ALPHA, AE, DELTA, DELTAF, SPANF, XSPF, TANAF, SINA1, SINA
     12, COSA2, COSA3, COSBT
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/6/GVORT, GVORT0, GS(50), GS0(50), RS(51), THETAS(50), YS(50
     1), ZS(50), D(51), H(52), R(50), R0(50), THETA(50), Y(50), Z(50), R
     2SV, THETSV, YSTARV, ZSTARV, YV, YV0, ZV, ZV0, RV, RV0, THETAV
      COMMON/9/EPS
      COMMON/10/LINC
      COMMON/17/RADISS, ZAXISS, YAXISS
      COMMON/19/DBDX, DRDDX
      COMMON/20/XEPF, EDELTF
      COMMON/23/ICASE
      COMMON/28/A, B, C, P

      IF (L .EQ. 0) 1, 2
    1 X1=1./DELTA
      RATIO=XX0/X1
      EPS=0.
      IF (NSHAPE .EQ. 0) 4, 5
    4 RADIUS=RADIUS*RATIO
      DRDDX=RADIUS/XX0
      RADISS=RADIUS
      GO TO 6
    5 AZAXIS=AZAXIS*RATIO
      ZAXISS=AZAXIS
      BYAXIS=BYAXIS*RATIO
      DBDX=BYAXIS/XX0
      YAXISS=BYAXIS
    6 YSTARV=YSTARV*RATIO
      ZSTARV=ZSTARV*RATIO
      DO 3 I=2, N2, 2
      GS(I)=GS(I)*RATIO*(DELTA)
    3 D(I)=D(I)*RATIO
      GVORT=GVORT*RATIO*DELTA
      S=RATIO
      RETURN
    2  IF (LINC .EQ. 0) 19, 21
   19 IF (X.LE. XSPF) 10, 11
   10 S=A*X**3+B*X**2+C*X+P
    9 RATIO=X/XX0
      IF (NSHAPE .EQ. 0) 7, 8
    7 IF (ICASE .EQ. 1) 12, 13
   12 RADIUS=RADISS*RATIO
      GO TO 20
   13 RADIUS=S*2./3.
      DRDDX=2.*(3.*A*X**2+2.*B*X+C)/3.
   20 RETURN
   11 S=SPANF+(X-XSPF)*DELTAF
      RATIO=X/XX0
      IF (NSHAPE .EQ. 0) 14, 8
   14 IF (ICASE .EQ. 1) 15, 16
   15 RADIUS=RADISS*RATIO
      GO TO 20
   16 RADIUS=S*2./3.
```

24

```
      GO TO 20
  8   AZAXIS=ZAXISS*RATIO
      BYAXIS=YAXISS*RATIO
      RETURN
 21   RATIO=X/XX0
      RADIUS=RADISS*RATIO
      S=X*DELTA
      IF (X .LE. XEPE) 17, 18
 17   EPS=3.*A*X**2+2.*B*X+C
      RETURN
 18   EPS=EDELTF
      RETURN
      END
      SUBROUTINE SETUP

C     THIS SUBROUTINE SETS UP ALL CONSTANTS FOR THE TRANSFORMATION

      COMMON/1/RADIUS, AZAXIS, BYAXIS, S, BETA
      COMMON/2/NSHAPE
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/7/RO, THETAO, YO, ZO
      COMMON/8/SPRSQS, SMRSQS
      COMMON/13/   CA, CB, CBMA, CBMASQ, CIETAZ, CRSQ
      COMMON/14/AZ, BY, AZSQ, BYSQ, XIZ, ETAZ, BMA, BSMAS, RSQ
      COMMON/15/   A, B, ROOT4, SINANG, COSANG, BETAR
      COMMON/30/TLIFT, UPPER, XLOWER

      COMPLEX CA, CB, CBMA, CBMASQ, CIETAZ, CRSQ

      IF (NSHAPE .EQ. 1) 1, 2
    1 AZ=AZAXIS
      CA=AZ
      BY=BYAXIS
      CB=BY
      AZSQ=AZ**2
      BYSQ=BY**2
      BETAR=BETA*RAD
      T=2.*BETAR
      BMA=BY-AZ
      CBMA=BMA
      BMASQ=BMA**2
      CBMASQ=BMASQ
      BSMAS=BY**2-AZ**2
      A=COS(T)-BSMAS/S**2
      B=SIN(T)
      ROOT4=SQRT(SQRT(A**2+B**2))
      THET=ATAN2(B, A)
      IF (THET .LE. 0.) 3, 4
    3 THET=THET+TWOPI
    4 ANG=.5*THET
      SINANG=SIN(ANG)
      COSANG=COS(ANG)
      FACT=BY*ROOT4
      XIZ=S*(FACT*COSANG-AZ*COS(BETAR))/BMA
      XIZSQ=XIZ**2
      ETAZ=S*(FACT*SINANG-AZ*SIN(BETAR))/BMA
      CIETAZ=ETAZ*(0., 1.)
      TLIFT=XIZ**2+BYSQ
      UPPER=SQRT((BSMAS-BMA*ETAZ)**2/BMASQ+XIZSQ)
      XLOWER=-SQRT((BSMAS+BMA*ETAZ)**2/BMASQ+XIZSQ)
      GO TO 7
    2 RSQ=RADIUS**2
      CRSQ=RSQ
      IF (RSQ .EQ. 0.) 5, 6
    5 BETA=0.
    6 BETAR=BETA*RAD
      SPRSQS=S+RSQ/S
      ETAZ=(S+RSQ/S)*SIN(BETAR)
      CIETAZ=ETAZ*(0., 1.)
      SMRSQS=S-RSQ/S
      XIZ=(S-RSQ/S)*COS(BETAR)
      XIZSQ=XIZ**2
      TLIFT=XIZ**2+RSQ
      UPPER=SQRT((2.*RADIUS-ETAZ)**2+XIZSQ)
      XLOWER=-SQRT((2.*RADIUS+ETAZ)**2+XIZSQ)
```

```
 7   CALL YZCOMP (0., 0., YO, ZO)
     RO=SQRT(YO**2+ZO**2)
     THETAO=ATAN(ZO/YO)
     RETURN
     END
```

```
      SUBROUTINE YZSCOR (Y, Z, YS, ZS)

      THIS SUBROUTINE DOES THE TRANSFORMATION BETWEEN Z AND ZSTAR

      COMMON/2/NSHAPE
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/14/AZ, BY, AZSQ, BYSQ, XIZ, ETAZ, BMA, BSMAS, RSQ

      IF (NSHAPE .EQ. 1) 1, 2
 2    DENOM=Y**2+Z**2
      REL=Y-RSQ*Y/DENOM
      AIM=Z+RSQ*Z/DENOM-ETAZ
      TOP=2.*REL*AIM
      BOTTOM=REL**2-AIM**2-XIZ**2
      RS=SQRT(SQRT(TOP**2+BOTTOM**2))
      IF (TOP) 8, 12, 8
12    CALL ANGDET (REL, AIM, PHI, BOTTOM, COSTRM, SINTRM)
      GO TO 10
 1    TOP=2.*Y*Z
      BOTTOM=Y**2-Z**2-BYSQ+AZSQ
      R=SQRT(SQRT(TOP**2+BOTTOM**2))
      IF (TOP) 4, 5, 4
 5    CALL ANGDET (Y, Z, PHI, BOTTOM, COSTRM, SINTRM)
      GO TO 6
 4    THET=ATAN2(TOP, BOTTOM)
      IF (THET .LT. 0.)THET=THET+TWOPI
      PHI=.5*THET
      COSTRM=COS(PHI)
      SINTRM=SIN(PHI)
      IF (COSTRM .GE. 0.) 6, 7
 7    COSTRM=-COSTRM
      SINTRM=-SINTRM
 6    REL=BY*R*COSTRM-AZ*Y
      AIM=BY*R*SINTRM-AZ*Z+BMA*ETAZ
      TOP=(2.*REL*AIM)/BMA**2
      BOTTOM=(REL**2-AIM**2)/BMA**2-XIZ**2
      RS=SQRT(SQRT(TOP**2+BOTTOM**2))
      IF (TOP) 8, 9, 8
 9    CALL ANGDET (REL/BMA, AIM/BMA, PHI, BOTTOM, COSTRM, SINTRM)
      GO TO 10
 8    THET=ATAN2(TOP, BOTTOM)
      IF (THET .LT. 0.) THET=THET+TWOPI
      PHI=.5*THET
      COSTRM=COS(PHI)
      SINTRM=SIN(PHI)
      IF (COSTRM .GE. 0.) 10, 11
11    COSTRM=-COSTRM
      SINTRM=-SINTRM
10    YS=RS*COSTRM
      ZS=RS*SINTRM
      RETURN
      END
      SUBROUTINE YZCOMP (YS, ZS, Y, Z)
C     THIS SUBROUTINE DOES THE TRANSFORMATION BETWEEN ZSTAR AND Z
      COMMON/2/NSHAPE
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/14/AZ, BY, AZSQ, BYSQ, XIZ, ETAZ, BMA, BSMAS, RSQ
      TOP=2.*YS*ZS
      BOTTOM=YS**2-ZS**2+XIZ**2
      RHO=SQRT(SQRT(TOP**2+BOTTOM**2))
      IF (TOP) 1, 2, 1
    2 CALL ANGDET (YS, ZS, PHI, BOTTOM, COSTRM, SINTRM)
      GO TO 3
    1 THET=ATAN2(TOP, BOTTOM)
      IF (THET .LT. 0.) 4, 5
    4 THET=THET+TWOPI
    5 PHI=.5*THET
      COSTRM=COS(PHI)
      SINTRM=SIN(PHI)
      IF (COSTRM .GE. 0.) 3, 6
    6 COSTRM=-COSTRM
      SINTRM=-SINTRM
    3 IF (NSHAPE .EQ. 0) 7, 8
    8 AL=BMA*RHO*COSTRM
      BE=BMA*(RHO*SINTRM+ETAZ)
      TOP=2.*AL*BE
      BOTTOM=BSMAS**2+AL**2-BE**2
      RHO=SQRT(SQRT(TOP**2+BOTTOM**2))
      IF (TOP) 9,10, 9
   10 CALL ANGDET (AL, BE, PHI, BOTTOM, COSTRM, SINTRM)
      GO TO 11
    9 THET=ATAN2(TOP, BOTTOM)
      IF (THET .LT. 0.) 12, 13
   12 THET=THET+TWOPI
   13 PHI=.5*THET
      COSTRM=COS(PHI)
      SINTRM=SIN(PHI)
   11 YTERM=RHO*COSTRM
      ZTERM=RHO*SINTRM
      Y=(AZ*AL+BY*YTERM)/BSMAS
      Z=(AZ*BE+BY*ZTERM)/BSMAS
      IF (Y) 14, 15, 15
   15 IF (AZ*AL-BY*YTERM) 16, 17, 17
   17 IF (((Y/BY)**2+(Z/AZ)**2+.0001) .GE. 1.) 16, 14
   14 Y=(AZ*AL-BY*YTERM)/BSMAS
      Z=(AZ*BE-BY*ZTERM)/BSMAS
   16 CONTINUE
      RETURN
    7 A=RHO*COSTRM
      B=RHO*SINTRM+ETAZ
      IF (RSQ) 18, 19, 18
   19 Y=A
      Z=B
      GO TO 20
   18 TOP=2.*A*B
      BOTTOM=A**2-B**2+4.*RSQ
      RHO=SQRT(SQRT(TOP**2+BOTTOM**2))
      IF (TOP) 21, 22, 21
   22 CALL ANGDET (A, B, PHI, BOTTOM, COSTRM, SINTRM)
      GO TO 23
   21 THET=ATAN2(TOP, BOTTOM)
```

```
      IF (THET .LT. 0.) 24, 25
   24 THET=THET+TWOPI
   25 PHI=.5*THET
      COSTRM=COS(PHI)
      SINTRM=SIN(PHI)
   23 YTERM=RHO*COSTRM
      ZTERM=RHO*SINTRM
      Y=.5*(A+YTERM)
      Z=.5*(B+ZTERM)
      IF (A+YTERM) 26, 27, 27
   27 IF (A-YTERM) 20, 28, 28
   28 IF ((Y**2+Z**2+.0001) .GE. RSQ) 20, 26
   26 Y=.5*(A-YTERM)
      Z=.5*(B-ZTERM)
   20 CONTINUE
      RETURN
      END
      SUBROUTINE DERIVE (YS, ZS, Y, Z, DR, DI)

C     THIS SUBROUTINE CALCULATES DZ/DZS

      COMMON/2/NSHAPE
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/13/   CA, CB, CBMA, CBMASQ, CIETAZ, CRSQ
      COMMON/14/AZ, BY, AZSQ, BYSQ, XIZ, ETAZ, BMA, BSMAS, RSQ
      COMMON/16/ROOT

      COMPLEX ROOT, CA, CB, CBMA, CBMASQ, CIETAZ, CAPZS, CAPZ, DZDZS, CR
     1SQ, FACTOR

      CAPZS=CMPLX(YS, ZS)
      CAPZ=CMPLX(Y, Z)
      IF (NSHAPE .EQ. 0) 1, 2
    2 TOP=2.*Y*Z
      BOTTOM=Y**2-Z**2-BSMAS
      RHO=SQRT(SQRT(TOP**2+BOTTOM**2))
      IF (TOP) 3, 4, 3
    4 CALL ANGDET (Y, Z, PHI, BOTTOM, COSTRM, SINTRM)
      GO TO 5
    3 THET=ATAN2(TOP, BOTTOM)
      IF (THET .LT. 0.) 6, 7
    6 THET=THET+TWOPI
    7 PHI=.5*THET
      COSTRM=COS(PHI)
      SINTRM=SIN(PHI)
      IF (COSTRM .GE. 0) 5, 8
      COSTRM=-COSTRM
      SINTRM=-SINTRM
      ROOT=CMPLX(RHO*COSTRM, RHO*SINTRM)
      DZDZS=(CBMASQ*CAPZS*ROOT)/((CB*ROOT-CA*CAPZ-CBMA*CIETAZ)*(CB*CAPZ-
     1CA*ROOT))
      GO TO 9
      FACTOR=CRSQ/CAPZ
      DZDZS=CAPZS/((CAPZ-FACTOR-CIETAZ)*((1.,0.)+FACTOR/CAPZ))
      DR=REAL(DZDZS)
      DI=AIMAG(DZDZS)
      RETURN
      END
      SUBROUTINE ANGDET (Y, Z, PHI, R. COSTRM, SINTRM)
C     THIS SUBROUTINE DETERMINES ANGLE OF ((Y+IZ)**2+REAL)**.5 WHEN Y OR
C     Z EQUALS ZERO
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      IF (R) 1, 2, 2
    1 IF (Y) 5. 3, 5
    3 IF (Z) 4, 2, 5
    5 PHI=PIO2
      COSTRM=0.
      SINTRM=1.
      RETURN
    2 PHI=0.
      COSTRM=1.
      SINTRM=0.
      RETURN
    4 PHI=THPIO2
      COSTRM=0.
      SINTRM=-1.
      RETURN
      END
      SUBROUTINE COFIGR (M)
C     THIS SUBROUTINE CALCULATES ALL CONFIGURATIONS OF SHEETS AND
C     THE ISOLATED VORTEX
C     M=1,        GIVE YSV, ZSV, D(PIV), H(PIV), GV, GS(PIV)
C                   CALCUTE ALL THE OTHERS
C     M=0,        CALCULATE G(PIV) ONLY


      COMMON/3/DX, X0, X, XX0
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/6/GVORT, GVORT0, GS(50), GS0(50), RS(51), THETAS(50), YS(50
     1), ZS(50), D(51), H(52), R(50), RO(50), THETA(50), Y(50), Z(50), R
     2SV, THETSV, YSTARV, ZSTARV, YV, YV0, ZV, ZV0, RV, RV0, THETAV
      COMMON/7/RO, THETAO, YO, ZO
      COMMON/27/DH(50), HD(25), DD(50), DRR(50), HH(25)

      NM1=N-1
      IF (M .EQ. 1)1, 2
    1 RSV=SQRT(YSTARV**2+ZSTARV**2)
      THETSV=ATAN (ZSTARV/YSTARV)
      CALL YZCOMP (YSTARV, ZSTARV, YV, ZV)
      RV=SQRT(YV**2+ZV**2)
      THETAV=ATAN(ZV/YV)
      D(1)=(RSV+D(2))/2.
      ARG=THETSV+H(1)
      YS(1)=YSTARV-D(1)*COS(ARG)
      ZS(1)=ZSTARV-D(1)*SIN(ARG)
      RS(1)=SQRT(YS(1)**2+ZS(1)**2)
      THETAS(1)=ATAN(ZS(1)/YS(1))
      CALL YZCOMP (YS(1), ZS(1), Y(1), Z(1))
      R(1)=SQRT(Y(1)**2+Z(1)**2)
      THETA(1)=ATAN(Z(1)/Y(1))
      DO 12 I=3, N2M1, 2
   12 D(I)= (D(I+1)+D(I-1))/2.
      DO 13 I=2, N2
      ARG=THETSV+H(I)
      YS(I)=YSTARV-D(I)*COS(ARG)
      ZS(I)=ZSTARV-D(I)*SIN(ARG)
      RS(I)=SQRT(YS(I)**2+ZS(I)**2)
      THETAS(I)=ATAN(ZS(I)/YS(I))
      CALL YZCOMP (YS(I), ZS(I), Y(I), Z(I))
      R(I)=SQRT(Y(I)**2+Z(I)**2)
   13 THETA(I)=ATAN(Z(I)/Y(I))
   21 D(N2P1)=D(N2)
      R(N2P1)=R(N2)
      DD(1)=D(2)-RSV
      DRR(1)=R(2)-RO
      DO 15 I=2, N2
      IP1=I+1
      IM1=I-1
      DD(I)=D(IP1)-D(IM1)
   15 DPR(I)=R(IP1)-R(IM1)
      GS(N2-1)=(DH(N2-1)+DH(N2-2))*(GS(N2)+GS(N2-2))/(2.*(DH(N2-1)+DH(N2
     1-3)))
      DO 19 I=2, NM1
      I2=2*(N-I+1)
   19 GS(I2-1)=((DH(I2-1)+DH(I2+1))*GS(I2)-DH(I2-1)*GS(I2+1))/DH(I2+1)
      GS(1)=(H(4)*GS(2)-2.*H(2)**2*GS(3)/(H(2)+H(4)))/(2.*DH(3))
      RETURN
```

```
2   GS(2)=(2.*DH(3)*GS(1)+2.*H(2)**2*GS(3)/(H(2)+H(4)))/H(4)
    DO 18 I=2, NM1
    I2=2*I
18  GS(I2)=(DH(I2-1)*GS(I2+1)+DH(I2+1)*GS(I2-1))/(DH(I2-1)+DH(I2+1))
    GS(N2)=((2.*DH(N2-1)+DH(N2-3))*GS(N2-1)-DH(N2-1)*GS(N2-3))/(DH(N2-
   11)+DH(N2-3))
    RETURN
    END
```

```
      SUBROUTINE SOURCE (Y, Z, DR, DI, SR, SI)
C     THIS SUBROUTINE CALCULATES THE SOURCE TERM IN COMPLEX VELOCITY
      COMMON/2/NSHAPE
      COMMON/13/    CA, CB, CBMA, CBMASQ, CIETAZ, CRSQ
      COMMON/16/ROOT
      COMMON/19/DBDX, DRDDX
      COMPLEX ROOT, DZDZS, S, CA, CB, CBMA, CBMASQ, CIETAZ, CRSQ, CAPZ,
     1CDRDDX, CDBDX
      CAPZ=CMPLX(Y, Z)
      DZDZS=CMPLX(DR, DI)
      IF (NSHAPE .EQ. 0) 1, 2
    2 CDBDX=DBDX*(1.,0.)
      S=CA*CDBDX*DZDZS/ROOT
      GO TO 3
    1 CDRDDX=DRDDX*(1.,0.)
      S=CSQRT(CRSQ)*CDRDDX*DZDZS/CAPZ
    3 SR=REAL(S)
      SI=AIMAG(S)
      RETURN
      END
      SUBROUTINE EPSLN (YS, ZS, Y, Z, DR, DI, ER, EI)
C
C     THIS SUBROUTINE CALCULATES EPS=-.5*((D2Z/DZS2)/(DZ/DZS))
C
      COMMON/2/NSHAPE
      COMMON/13/    CA, CB, CBMA, CBMASQ, CIETAZ, CRSQ
      COMMON/16/ROOT
C
      COMPLEX CA, CB, CBMA, CBMASQ, CIETAZ, ROOT, V, W, CAPZS, CAPZ, DZD
     1ZS, EPS, CRSQ, T1, T2, D1, D2
C
      CAPZS=CMPLX(YS, ZS)
      CAPZ=CMPLX(Y, Z)
      DZDZS=CMPLX(DR, DI)
      IF (NSHAPE .EQ. 0) 1, 2
    2 V=CB*ROOT-CA*CAPZ-CIETAZ*CBMA
      W=CB*CAPZ-CA*ROOT
      EPS=(-.5,0.)*((1.,0.)/CAPZS+DZDZS*(CAPZ/ROOT-W/V-(CB*ROOT-CA*CAPZ)
     1/W)/ROOT)
      GO TO 3
    1 T1=CRSQ/CAPZ
      T2=T1/CAPZ
      D1=CAPZ-T1-CIETAZ
      D2=(1.,0.)+T2
      EPS=(.5,0.)*((D2/D1-(2.,0.)*(T2/CAPZ)/D2)*DZDZS-(1.,0.)/CAPZS)
    3 ER=REAL(EPS)
      EI=AIMAG(EPS)
      RETURN
      END
      SUBROUTINE DWINC (YS. ZS. DIDZSR. DIDZSI)

C     THIS SUBROUTINE CALCULATES THE TERM DW(INC)/DZS

      COMMON/1/RADIUS. AZAXIS. BYAXIS, S. BETA
      COMMON/5/PI, TWOPI. PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/8/SPRSQS, SMRSQS
      COMMON/9/EPS

      COMPLEX CZS, CA, CB, CD. CE, CF, CT, CDIDZS, CEPS, CTT, CPI2. CPI

      TT=ACOS(2.*RADIUS*S/(S**2+RADIUS**2))
      IF ((YS .EQ. 0.) .AND. (ZS .EQ. 0.)) 6, 7
    6 DIDZSR=0.
      DIDZSI=-EPS*(TT/PI+.5)
      RETURN
    7 CZS=CMPLX(YS, ZS)
      CD=CMPLX(0., SMRSQS)
      CEPS=CMPLX(EPS, 0.)
      CPI2=CMPLX(TWOPI, 0.)
      CTT=CMPLX(0., TT)
      CPI=CMPLX(PI, 0.)
      CE=(2.*RADIUS*(1., 0.))*CZS
      TOP=2.*ZS*YS
      BOTTOM=YS**2-ZS**2+SPRSQS**2
      RS=SQRT(SQRT(TOP**2+BOTTOM**2))
      IF (TOP) 1, 2, 1
    2 CALL ANGDET (YS, ZS, PHI, BOTTOM, COSTRM. SINTRM)
      GO TO 3
    1 THET=ATAN2(TOP, BOTTOM)
      IF (THET .LE. 0.) THET=THET+TWOPI
      PHI=.5*THET
      COSTRM=COS(PHI)
      SINTRM=SIN(PHI)
      IF (COSTRM .GE. 0.) 3, 5
    5 COSTRM=-COSTRM
      SINTRM=-SINTRM
    3 CT=CMPLX(RS*COSTRM, RS*SINTRM)
      CF=CD*CT
      CA=CLOG(CF+CE)
      CB=CLOG(CF-CE)
      AAR=REAL(CA)
      AAI=AIMAG(CA)
      IF (AAI .LE. 0.) AAI=AAI+TWOPI
      CA=CMPLX(AAR, AAI)
      BBR=REAL(CB)
      BBI=AIMAG(CB)
      IF (BBI .LE. 0.) BBI=BBI+TWOPI
      CB=CMPLX (BBR, BBI)
      CDIDZS= CEPS*(CZS*(CA-CB)/(CPI2*CT)-CTT/CPI-(0., 1.)*((1., 0.)-CZS
     1/CT)/(2., 0.))
      DIDZSR=REAL(CDIDZS)
      DIDZSI=AIMAG(CDIDZS)
      RETURN
      END
      SUBROUTINE USEFUL

      THIS SUBROUTINE CALCULATES SOME USEFUL EXPRESSIONS FOR
      THE LOOP1 AND THE LOOP3

      COMMON/3/DX, X0, X, XX0
      COMMON/4/ALPHA, AF, DELTA, DELTAF, SPANF, XSPF, TANAF, SINA1, SINA
     12, COSA2, COSA3, COSBT
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/6/GVORT, GVORT0, GS(50), GS0(50), RS(51), THETAS(50), YS(50
     1), ZS(50), D(51), H(52), R(50), R0(50), THETA(50), Y(50), Z(50), R
     2SV, THETSV, YSTARV, ZSTARV, YV, YV0, ZV, ZV0, RV, RV0, THETAV
      COMMON/7/R0, THETA0, Y0, Z0
      COMMON/10/LINC
      COMMON/11/DWDSGR(25), DWDSGI(25), E(25), COSPHI(25), SINPHI(25), A
     1BSD(25)
      COMMON/27/DH(50), HD(25), DD(50), DRR(50), HH(25)

      DIMENSION DR(25), DI(25), DWDZSR(25), DWDZSI(25)

      DO 3 I=1, N
      L=2*I-1
      CALL DERIVE (YS(L), ZS(L),Y(L), Z(L), DR(I), DI(I))
    3 ABSD(I)=1./SQRT(DR(I)**2+DI(I)**2)
      DO 1 I=1, N
      I2M1=2*I-1
      CALL SOURCE (Y(I2M1), Z(I2M1), DR(I), DI(I), SR, SI)
      IF (LINC .EQ. 0) 6, 7
    6 DIDZSR=0.
      DIDZSI=0.
      GO TO 4
    7 CALL DWINC (YS(I2M1), ZS(I2M1), DIDZSR, DIDZSI)
    4 YSMYSV=YS(I2M1)-YSTARV
      YSPYSV=YS(I2M1)+YSTARV
      ZSMZSV=ZS(I2M1)-ZSTARV
      YDENV=(YSMYSV**2+ZSMZSV**2)*(YSPYSV**2+ZSMZSV**2)
      SUMR=0.
      SUMI=0.
      DO 2 K=1, N
      K2=2*K
      YSMYS=YS(I2M1)-YS(K2)
      YSPYS=YS(I2M1)+YS(K2)
      ZSMZS=ZS(I2M1)-ZS(K2)
      YDENS=(YSMYS**2+ZSMZS**2)*(YSPYS**2+ZSMZS**2)
      SUMR=SUMR+GS(K2)*HH(K)*YS(K2)*ZSMZS*(YSMYS+YSPYS)/(TWOPI*YDENS)
    2 SUMI=SUMI+GS(K2)*HH(K)*YS(K2)*(YSMYS*YSPYS-ZSMZS**2)/(TWOPI*YDENS)
      DWDZSR(I)=-GVORT    *YSTARV*ZSMZSV*(YSMYSV+YSPYSV)/(PI*YDENV)-SUMR+
     1SR+DIDZSR
    1 DWDZSI(I)=-(GVORT    *YSTARV*(YSMYSV*YSPYSV-ZSMZSV**2)/(PI*YDENV)+S
     1UMI+TANAF-SI-DIDZSI)
      DO 5 I=1, N
      L=2*I-1
      E(I)=SQRT(D(L)**2+(DD(L)/DH(L))**2)
      COSPHI(I)=DRR(L)*ABSD(I)/(E(I)*DH(L))
      SINPHI(I)=SQRT(1.-COSPHI(I)**2)
      ANG=THETSV+H(L)
      ONEI=(COS(ANG)*D(L)+SIN(ANG)*DD(L)/DH(L))/E(I)*(-1.)
      TWOR=(COS(ANG)*DD(L)/DH(L)-SIN(ANG)*D(L))/E(I)*(-1.)
      DWDSGR(I)=ABSD(I)*(TWOR*DWDZSR(I)-ONEI*DWDZSI(I))
    5 DWDSGI(I)=ABSD(I)*(TWOR*DWDZSI(I)+ONEI*DWDZSR(I))
      RETURN
      END
      SUBROUTINE COJGDW (CDWZVR, CDWZVI)

      THIS SUBROUTINE CALCULATES THE COMPLEX CONJUGATE OF THE
       VELOCITY FIELD AT THE ISOLATED VORTEX

      COMMON/4/ALPHA, AE, DELTA, DELTAF, SPANF, XSPF, TANAF, SINA1, SINA
     12, COSA2, COSA3, COSBT
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/6/GVORT, GVORTO, GS(50), GSO(50), RS(51), THETAS(50), YS(50
     1), ZS(50), D(51), H(52), R(50), RO(50), THETA(50), Y(50), Z(50), R
     2SV, THETSV, YSTARV, ZSTARV, YV, YVO, ZV, ZVO, RV, RVO, THETAV
      COMMON/10/LINC
      COMMON/27/DH(50), HD(25), DD(50), DRR(50), HH(25)

      CALL DERIVE (YSTARV, ZSTARV, YV, ZV, DVR, DVI)
      CALL SOURCE (YV, ZV, DVR, DVI, SVR, SVI)
      CALL EPSLN (YSTARV, ZSTARV, YV, ZV, DVR, DVI, EVR, EVI)
      IF (LINC .EQ. 0) 1, 2
1     DIDZSR=0.
      DIDZSI=0.
      GO TO 3
2     CALL DWINC (YSTARV, ZSTARV, DIDZSR, DIDZSI)
3     SUMR=0.
      SUMI=0.
      DO 4 K=1, N
      K2=2*K
      YSVMYS=YSTARV-YS(K2)
      YSVPYS=YSTARV+YS(K2)
      ZSVMZS=ZSTARV-ZS(K2)
      YDENSV=(YSVMYS**2+ZSVMZS**2)*(YSVPYS**2+ZSVMZS**2)
      SUMR=SUMR-GS   (K2)*HH(K)*YS(K2)*ZSVMZS*(YSVMYS+YSVPYS)/(TWOPI*YDEN
     1SV)
4     SUMI=SUMI+GS   (K2)*HH(K)*YS(K2)*(YSVMYS*YSVPYS-ZSVMZS**2)/(TWOPI*Y
     1DENSV)
      SUMR=SUMR+SVR+GVORT*EVI/TWOPI+DIDZSR
      SUMI=-(TANAF-GVORT/(2.*TWOPI*YSTARV)+SUMI+GVORT*EVR/TWOPI-DIDZSI-S
     1VI)
      ABDV=DVR**2+DVI**2
      DZSZVR=DVR/ABDV
      DZSZVI=-DVI/ABDV
      CDWZVR=DZSZVR*SUMR-DZSZVI*SUMI
      CDWZVI=-(DZSZVI*SUMR+DZSZVR*SUMI)
      RETURN
      END
      SUBROUTINE GAMERV (GVORD)

      THIS SUBROUTINE CALCULATES THE NEW VORTEX STRENGTH OF THE ISOLATED
      VORTEX BY SATISFYING THE KUTTA CONDITION

      COMMON/4/ALPHA, AF, DELTA, DELTAF, SPANF, XSPF, TANAF, SINA1, SINA
     12, COSA2, COSA3, COSBT
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/6/GVORT, GVORTO, GS(50), GSO(50), RS(51), THETAS(50), YS(50
     1), ZS(50), D(51), H(52), R(50), RO(50), THETA(50), Y(50), Z(50), R
     2SV, THETSV, YSTARV, ZSTARV, YV, YVO, ZV, ZVO, RV, RVO, THETAV
      COMMON/10/LINC

      SUM=0.
      DO 1 I=3, N
      I2=2*I
    1 SUM=SUM+(H(I2+2)-H(I2-2))*GS(I2)*COS(THETAS(I2))/RS(I2)
      IF (LINC .EQ. 0) 2, 3
    2 DIDZSI=0.
      GO TO 4
    3 CALL DWINC (0., 0., DIDZSR, DIDZSI)
    4 DIV=COS(THETSV)/(PI*RSV)
      GVORD=TANAF-GS(2)*H(4)**2*COS(THETAS(2))/(TWOPI*(H(4)-H(2))*RS(2))
     1-GS(4)*(H(6)-H(2)*H(4)/(H(4)-H(2)))*COS(THETAS(4))/(TWOPI*RS(4))-S
     2UM/TWOPI+DIDZSI
      GVORD=GVORD/DIV
      RETURN
      END
      SUBROUTINE LOOP1 (ISTOP1)

      THIS SUBROUTINE IS DESIGNED TO SATISFY THE PRISSURE CONDITION

      COMMON/3/DX, XC, X, XX0
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/6/GVORT, GVORT0, GS(50), GS0(50), RS(51), THETAS(50), YS(50
     1), ZS(50), D(51), H(52), R(50), R0(50), THETA(50), Y(50), Z(50), R
     2SV, THETSV, YSTARV, ZSTARV, YV, YV0, ZV, ZV0, RV, RV0, THETAV
      COMMON/11/DWDSGR(25), DWDSGI(25), E(25), COSPHI(25), SINPHI(25), A
     1BSD(25)
      COMMON/18/ACL
      COMMON/24/NLOOP1, ACC1, SUM1(25), SUM10(25), SUM11(25), SSUM1(25),
     1 SUM2(25), SUM20(25), SUM21(25), SSUM2(25), CSPIX0(25), DWDSX0(25)
     2, RX0(25)
      COMMON/27/DH(50), HD(25), DD(50), DRR(50), HH(25)

      DIMENSION AD(25), G(50), ER(25)

      ISTOP1=0
      ACC=ACC1
      NITER=0
      CALL COFIGR (1)
    6 IPASS=0
      DO 3 I=1, N2
    3 G(I)=GS(I)
      GVORD=GVORT
      CALL USEFUL
      DO 1 I=1, N
      L=2*I-1
      SUM11(I)=COSPHI(I)
      SSUM1(I)=SUM1(I)+(SUM10(I)+SUM11(I))*(R(L)-R0(L))/2.
      SUM21(I)=DWDSGR(I)
      SSUM2(I)=SUM2(I)+(SUM20(I)+SUM21(I))*(X-X0)/2.
    1 AD(I)=(CSPIX0(I)*RX0(I)+SSUM1-DWDSX0(I)*XX0-SSUM2(I))*ABSD(I)/E(I)
      SUM=SSUM=0.
      DO 2 I=1, N
      IN=N+1-I
      SUM=SUM+SSUM
      L=2*IN-1
      GS(L)=(-GVORT-SUM)/(.5*DH(L)+AD(IN))
    2 SSUM=HD(IN)*GS(L)
      CALL COFIGR (0)
      DO 4 I=1, N
      I2=2*I
      ER(I)=ABS((GS(I2)-G(I2))/G(I2))
      IF (ER(I) .GT. ACC1) IPASS=1
    4 CONTINUE
      CALL GAMERV (GVORT)
      ERROR=ABS((GVORT-GVORD)/GVORD)
      IF (ERROR .LE. ACC1 .AND. IPASS .EQ. 0) GO TO 5
      NITER=NITER+1
      IF (NITER .LE. NLOOP1) GO TO 9
      ACC1=ACC1+ACC
      IF (ACC1 .GT. ACL ) 22, 9
   22 ISTOP1=1
      PRINT 20, (ER(I), I=1, N), ERROR, ACC1
      RETURN
    9 GVORT=(GVORT+GVORD)/2.
      DO 10 I=1, N2
   10 GS(I)=(GS(I)+G(I))/2.
      GO TO 6
      CONTINUE
      IF (ACC1 .EQ. ACC) GO TO 25
      PRINT 20, (ER(I), I=1, N), ERROR, ACC1
      ACC1=ACC
   25 RETURN
   20 FORMAT (1H0, 20HERRORS ARE TOO LARGE, 2X, 3HER=6(E12.5, 2X)//1X, 7
     1HERROR= E12.5, 2X, 5HACC1=E12.5)
      END
      SUBROUTINE LOOP2 (M, ISTOP2)

      THIS SUBROUTINE IS DESIGNED TO SATISFY THE FORCE BALANCE EQUATION

      COMMON/3/DX, XO, X, XXO
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/6/GVORT, GVORTO, GS(50), GSO(50), RS(51), THETAS(50), YS(50
     1), ZS(50), D(51), H(52), R(50), RO(50), THETA(50), Y(50), Z(50), R
     2SV, THETSV, YSTARV, ZSTARV, YV, YVO, ZV, ZVO, RV, RVO, THETAV
      COMMON/18/ACL
      COMMON/25/NLOOP2, ACC2, STEP

      DIMENSION YSV(3), ZSV(3), VORLFT(4), ZETAR(4), ZETAI(4), ER(4)

      ISTOP2=0
      ACC=ACC2
      IF (M) 3, 2, 3
    2 CALL LOOP1 (ISTOP1)
      IF (ISTOP1 .EQ. 1) GO TO 28
      RETURN
   10 PRINT 31, ER(4), ACC2
   28 ISTOP2=1
      RETURN
    3 NITER=0
      I=1
      YSV(1)=YSTARV
      ZSV(1)=ZSTARV
   27 CALL LOOP1 (ISTOP1)
      IF (ISTOP1 .EQ. 1) GO TO 10
      NITER=NITER+1
      VORLFT(I)=GVORT*YSTARV*4.
      CALL COJGDW (CDWZVR, CDWZVI)
      ZETAR(I)=(GVORT-GVORTO)/DX*(YV-Y(N2))-GVORT*(CDWZVR-(YV-YVO)/DX)
      ZETAI(I)=(GVORT-GVORTO)/DX*(ZV-Z(N2))-GVORT*(CDWZVI-(ZV-ZVO)/DX)
      ER(I)=(ZETAR(I)**2+ZETAI(I)**2)/VORLFT(I)**2
      IF (ER(I) .LE. ACC2) 1, 4
      IF (ACC2 .EQ. ACC) GO TO 35
      PRINT 31, ER(4), ACC2
      ACC2=ACC
   35 RETURN
    4 GO TO (5, 6, 7, 8) I
    5 YSTARV=YSV(1)+STEP
      YSV(2)=YSTARV
      ZSV(2)=ZSTARV
      I=2
      GO TO 9
    6 YSTARV=YSV(1)
      YSV(3)=YSTARV
      ZSTARV=ZSV(1)+STEP
      ZSV(3)=ZSTARV
      I=3
      GO TO 9
    9 RSVOLD=RSV
      RSV=SQRT(YSTARV**2+ZSTARV**2)
      DELRSV=RSV-RSVOLD
      TERM=DELRSV/H(N2)
      DO 26 J=1, N
      I2=2*J
   26 D(I2)=D(I2)+DELRSV-TERM*H(I2)
      GO TO 27
    7 TERM1=ZETAR(2)*ZETAI(3)-ZETAR(3)*ZETAI(2)
      TERM2=ZETAR(1)*ZETAI(3)-ZETAR(3)*ZETAI(1)
      TERM3=ZETAR(1)*ZETAR(2)-ZETAR(2)*ZETAI(1)
      DETERM=TERM1-TERM2+TERM3
      YSVZ=(YSV(1)*TERM1-YSV(2)*TERM2+YSV(3)*TERM3)/DETERM
      ZSVZ=(ZSV(1)*TERM1-ZSV(2)*TERM2+ZSV(3)*TERM3)/DETERM
      IF (ABS((YSVZ-YSTARV)/YSTARV) .GE. .5) 12, 13
 13   IF (ABS((ZSVZ-ZSTARV)/ZSTARV) .GE. .5) 12, 14
  8   ERRMAX=AMAX1(ER(1), ER(2), ER(3))
  7   IF (ER(4) .GT. ERRMAX) 15, 16
 15   IF (NTRY .GT.  6) 30, 18
 30   ACC2=ACC2+ACC
      IF (ACC2 .GT. ACL  ) 10, 18
 18   NTRY=NTRY+1
      NITER=NITER-1
      ERRMIN=AMIN1(ER(1), ER(2), ER(3))
      DO 19 J=1, 3
      IF (ER(J) .NE. ERRMIN) 19, 20
 20   L=J
      GO TO 21
 19   CONTINUE
 21   YSVZ=YSV(L)+(YSVZ-YSV(L))/2.
      ZSVZ=ZSV(L)+(ZSVZ-ZSV(L))/2.
 14   YSTARV=YSVZ
      ZSTARV=ZSVZ
      I=4
      MM=1
      GO TO 9
 16   IF (NITER .EQ. NLOOP2) 30, 22
 22   NTRY=0
      DO 23 J=1, 3
      IF (ER(J) .EQ. ERRMAX) 24, 23
      CONTINUE
      YSV(J)=YSVZ
      ZSV(J)=ZSVZ
      ZETAR(J)=ZETAR(4)
      ZETAI(J)=ZETAI(4)
      ER(J)=ER(4)
      VORLFT(J)=VORLFT(4)
      GO TO 7
      TAU=ATAN((ZSVZ-ZSTARV)/(YSVZ-YSTARV))
      YSVZ=YSTARV+.01*COS(TAU)
      ZSVZ=ZSTARV+.01*SIN(TAU)
      GO TO 14
 31   FORMAT (1H0, 18HERROR IS TOO LARGE , 2X, 6HERROR=F12.5, 2X, 5HACC2=
     1F12.5)
      END
      SUBROUTINE LOOP3 (ISTOP3)
C     THIS SUBROUTINE IS DESIGNED TO SATISFY THE NORMAL VELOCITY
      COMMON/1/RADIUS, AZAXIS, BYAXIS, S, BETA
      COMMON/3/DX, X0, X, XX0
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/6/GVORT, GVORTO, GS(50), GSO(50), RS(51), THETAS(50), YS(50
     1), ZS(50), D(51), H(52), R(50), RO(50), THETA(50), Y(50), Z(50), R
     2SV, THETSV, YSTARV, ZSTARV, YV, YVO, ZV, ZVO, RV, RVO, THETAV
      COMMON/11/DWDSGR(25), DWDSGI(25), E(25), COSPHI(25), SINPHI(25), A
     1BSD(25)
      COMMON/18/ACL
      COMMON/26/NLOOP3, ACC3, COTPHI(25), THETAO(50)
      COMMON/27/DH(50), HD(25), DD(50), DRR(50), HH(25)
      DIMENSION ETA(25), DOLD(25), DEL(25), ER(25)
      ISTOP3=0
      ACC=ACC3
      DELOLD=0.
      APRIME=.05
      NITER=0
      M=0
   12 CALL LOOP2 (M, ISTOP2)
      IF (ISTOP2 .EQ. 1) GO TO 23
      DO 6 I=1, N
    6 DOLD(I)=D(2*I)
      DO 1 I=1, N
      L=2*I-1
      ETA(I)=((R(L)-RO(L)-RO(L)*COTPHI(I)*(THETA(L)-THETAO(L)))/DX-DWDSG
     1I(I)/SINPHI(I))/DWDSGR(I)*SINPHI(I)
    1 ETA(I)=ASIN(ETA(I))
      DEL(1)=-ETA(1)*(D(2)**2+RSV**2-2.*D(2)*RSV*COS(DH(1)))/(RSV*SIN(DH
     1(1)))
      DO 2 I=2, N
      L=2*I-1
      I2=2*I
      I2M2=I2-2
    2 DEL(I)=(D(I2)*DEL(I-1))/D(I2M2)-ETA(I)*(D(I2)**2+D(I2M2)**2-2.*D(I
     12)*D(I2M2)*COS(DH(L)))/(D(I2M2)*SIN(DH(L)))
      DO 3 I=1, N
    3 ER(I)=ABS(DEL(I)/D(2*I))
      NITER=NITER+1
      ERRMAX=ARRAYMAX(ER, N)
      IF (ERRMAX .LE. ACC3) 7,8
    8 IF (NITER .GE. NLOOP3) 9, 10
    9 ACC3=ACC3+ACC
      IF (ACC3 .GT. ACL  ) 13, 10
   13 PRINT 25, ERRMAX, ACC3
   23 ISTOP3=1
      RETURN
   10 M=1
      IF (DEL(N)*DELOLD) 4, 5, 5
    5 DELOLD=DEL(N)
      GO TO 14
    4 APRIME=APRIME/2.
      DELOLD=0.
   14 IF (ERRMAX .LE. APRIME) 15, 16
   15 FACTER=1.
      GO TO 17
   16 FACTER=APRIME/ERRMAX
   17 CONTINUE
      DO 18 I=1, N
   18 D(2*I)=DOLD(I)+DEL(I)*FACTER
      GO TO 12
    7 IF (ACC3 .EQ. ACC) GO TO 30
      PRINT 25, ERRMAX, ACC3
      ACC3=ACC
   30 RETURN
   25 FORMAT (1H0, 18HERROR IS TOO LARGE, 2X, 6HERMAX=E12.5, 2X, 5HACC3=
     1E12.5)
      END
      SUBROUTINE LIFT (K)

C       THIS SUBROUTINE CALCULATES LINEAR LIFT COEF. AND NONLINEAR LIFT
C     COEFF..    K=0, CALCULATES EVERYTHING AT XO STATION.
C                K=1, CALCULATES EVERYTHING AT ANY STATION.

      COMMON/1/RADIUS, AZAXIS, BYAXIS, S, BETA
      COMMON/3/DX, XO, X, XXO
      COMMON/4/ALPHA, AL, DELTA, DELTAF, SPANF, XSPF, TANAF, SINA1, SINA
     12, COSA2, COSA3, COSBT
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/6/GVORT, GVORTO, GS(50), GSO(50), RS(51), THETAS(50), YS(50
     1), ZS(50), D(51), H(52), R(50), RO(50), THETA(50), Y(50), Z(50), R
     2SV, THETSV, YSTARV, ZSTARV, YV, YVO, ZV, ZVO, RV, RVO, THETAV
      COMMON/8/SPRSQS, SMRSQS
      COMMON/9/EPS
      COMMON/10/LINC
      COMMON/27/DH(50), HD(25), DD(50), DRR(50), HH(25)
      COMMON/29/SUM3, SUM30, SUM31, SSUM3, SUM4, SUM40, SUM41, SSUM4, SU
     1M5, SUM50, SUM51, SSUM5, SUM6, SUM60, SUM61, SSUM6
      COMMON/30/TLIFT, UPPER, XLOWER

      IF (K .EQ. 0) 5, 6
    5 SUM30=S
      SUM3=.5*S*X
      CL1=PI*SINA1*COSA2*TLIFT/(COSBT*SUM3)
      SUM=0.
      DO 40 I=1, N
      L=2*I
   40 SUM=SUM+GS(L)*2.*YS(L)*HH(I)
      CL2=2.*(GVORT*2.*YSTARV+SUM/2.)*COSA3/(COSBT*SUM3*2.)
      CL=CL1+CL2
      SUM40=CL1*2.*SUM3
      SUM50=CL2*2.*SUM3
      SUM4=2.*CL1/3.*2.*SUM3*XO
      SUM5=2.*CL2/3.*2.*SUM3*XO
      AOM1=SUM4/(2.*SUM3*XO)
      AOM2=SUM5/(2.*SUM3*XO)
      AOM=AOM1+AOM2
      RATIOO=AOM/CL
      CLIN=0.
      SUM60=0.
      SUM6=0.
      SSUM6=0.
      GO TO 8
    6 SUM31=S
      SSUM3=SUM3+(SUM30+SUM31)*(X-XO)/2.
      CL1=PI*SINA1*COSA2*TLIFT/(COSBT*SSUM3)
      SUM=0.
      DO  1 I=1, N
      L=2*I
    1 SUM=SUM+GS(L)*2.*YS(L)*HH(I)
      CL2=2.*(GVORT*2.*YSTARV+SUM/2.)*COSA3/(COSBT*SSUM3*2.)
      CL=CL1+CL2
      SUM41=CL1*2.*SSUM3
      SUM51=CL2*2.*SSUM3
      SSUM4=SUM4+(X+XO)*(SUM41-SUM40)/2.
      AOM1=SSUM4/(2.*SSUM3*X)
      SSUM5=SUM5+(X+XO)*(SUM51-SUM50)/2.
      AOM2=SSUM5/(2.*SSUM3*X)
      AOM=AOM1+AOM2
      RATIOO=AOM/CL
   IF (LINC .EQ. 0) 3, 2
 3  CLIN=0.
    GO TO 7
 2  R2=RADIUS**2
    THET=ATAN2(S**2-R2, 2.*RADIUS*S)
    IF (THET .LE. 0.) THET=THET+TWOPI
    CLIN=EPS*COSA3*((TLIFT+R2)*ACOS(2.*RADIUS*S/(S**2+R2))+THET*(SPRSQ
   1S**2-TLIFT-R2)+SMRSQS*(PI*SMRSQS/2.-4.*RADIUS))/ (COSBT*SSUM3)*2.
    CL=CL+CLIN
    SUM61=CLIN*2.*SSUM3
    SSUM6=SUM6+(X+X0)*(SUM61-SUM60)/2.
    AOM1=SSUM6/(2.*SSUM3*X)
    AOM=AOM+AOM1
 7  SUM30=SUM31
    SUM40=SUM41
    SUM50=SUM51
    SUM3=SSUM3
    SUM4=SSUM4
    SUM5=SSUM5
    SUM6=SSUM6
 8  PRINT 4, CL1, CL2, CLIN, CL, AOM1, AOM2, AOM, RATIOO
 4  FORMAT (1H0, 12X, 18HLINEAR LIFT COEFF=E12.5, 2X, 21HNONLINEAR LIF
   1T COEFF=E12.5, 2X, 17HINCDT LIFT COEFF=E12.5//33X, 17HTOTAL LIFT C
   2OEFF=E12.5//13X, 14HLINEAR MOMENT=E12.5, 2X, 17HNONLINEAR MOMENT=E
   312.5, 2X, 19HTOTAL MOMENT COEFF=E12.5//33X, 6HCM/CL=F10.5)
    RETURN
    END
```

```
      SUBROUTINE PREPRS (LM)

       THIS SUBROUTINE PREPARES AND CALCULATES THE PRESSURE.
       LM=1.  PREPARES ALL INFORMATIONS AT THE FIRST STATION.
       LM=2.  CALCULATES THE PRESSURE AT SECOND STATION.

      COMMON/1/RADIUS, AZAXIS, BYAXIS, S, BETA
      COMMON/2/NSHAPE
      COMMON/3/DX, X0, XXX, XX0
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/6/GVORT, GVORTO, CS(50), GS0(50), RS(51), THETAS(50), YS(5
     11), ZS(51), D(51), H(52), R(50), R0(50), THETA(50), Y(50), Z(50), P
     2SV, THETSV, YSTARV, ZSTARV, YV, YV0, ZV, ZV0, PV, PV0, THETAV
      COMMON/10/X, NPRE, IPRIT
      COMMON/30/TLIFT, UPPER, XLOWER
      COMMON/31/CZVS, CZS(25)

      DIMENSION YPS(202), ZPS(202), YP(202), ZP(202), PHIX0(202), PHIX(2
     102), PRL(202), NOPHI(202), PHIXY(202), PHIXZ(202)

      COMPLEX CZVS, CZS

      M1=NPRE+1
      M2=NPRE+2
      M3=2*NPRE+2
      DO 3 KJ=1, M3
    3 NOPHI(KJ)=0
      IF (LM .EQ. 1) 1, 2
    1 X=X+DX
      XXX=X
      CALL NCONSH (1)
      CALL SETUP
      YPS(1)=0.
      YPS(M1)=0.
      YPS(M2)=0.
      YPS(M3)=0.
      ZPS(1)=.0001*UPPER
      ZPS(M1)=.9999*UPPER
      ZPS(M2)=.0001*XLOWER
      ZPS(M3)=.9999*XLOWER
      CALL YZCOMP (YPS( 1), ZPS( 1), YP( 1), ZP( 1))
      CALL YZCOMP (YPS(M1), ZPS(M1), YP(M1), ZP(M1))
      CALL YZCOMP (YPS(M2), ZPS(M2), YP(M2), ZP(M2))
      CALL YZCOMP (YPS(M3), ZPS(M3), YP(M3), ZP(M3))
      UPPER=UPPER/NPRE
      XLOWER=XLOWER/NPRE
      DO 44 I=2, NPRE
      YPS(I)=0.
      ZPS(I)=UPPER*(I-1)
      CALL YZCOMP (YPS(I), ZPS(I), YP(I), ZP(I))
      J=I+M1
      YPS(J)=0.
      ZPS(J)=XLOWER*(I-1)
   44 CALL YZCOMP (YPS(J), ZPS(J), YP(J), ZP(J))
      X=X-DX
      XXX=X
      CALL NCONSH (1)
      CALL SETUP
      CZVS=CMPLX(YSTARV, ZSTARV)
      DO 51 I=1, N
      I2=2*I
   51 CZS(I)=CMPLX(YS(I2), ZS(I2))
```

```
      DO 46 I=1, M1
      CALL  YZSCOP (YP(I), ZP(I), YPS(I), ZPS(I))
      IF (ZPS(I) .EQ. 0.) ZPS(I)=ZPS(I)+1.E-8
      CALL PHII (YPS(I), ZPS(I), YP(I), ZP(I), PHIXO(I), MLP)
      IF (MLP .EQ. 1) 45, 58
45    NOPHI(I)=1
58    J=I+M1
      CALL  YZSCOP (YP(J), ZP(J), YPS(J), ZPS(J))
      IF (((AZAXIS .EQ. 0) .AND. (NSHAPE .EQ. 1)) .OR. ((RADIUS .EQ. 0)
     1.AND. (NSHAPE .EQ. 0))) 59, 60
59    ZPS(J)=-ZPS(J)
      GO TO 40
60    IF (ZP(J) .EQ. 0.) ZPS(J)=-ZPS(J)
      IF (ZPS(J) .EQ.0.) ZPS(J)=ZPS(J)-1.E-8
40    CALL PHII (YPS(J), ZPS(J), YP(J), ZP(J), PHIXO(J), MLP)
      IF (MLP .EQ. 1) 61, 46
61    NOPHI(J)=1
46    CONTINUE
      PETURN
 2    PRINT 61
      YPS(1)=0.
      YPS(M1)=0.
      YPS(M2)=0.
      YPS(M3)=0.
      ZPS(1)=.0001*UPPER
      ZPS(M1)=.9999*UPPER
      ZPS(M2)=.0001*XLOWER
      ZPS(M3)=.9999*XLOWER
      CALL YZCOMP (YPS( 1), ZPS( 1), YP( 1), ZP( 1))
      CALL YZCOMP (YPS(M1), ZPS(M1), YP(M1), ZP(M1))
      CALL YZCOMP (YPS(M2), ZPS(M2), YP(M2), ZP(M2))
      CALL YZCOMP (YPS(M3), ZPS(M3), YP(M3), ZP(M3))
      UPPER=UPPER/NPRE
      XLOWER=XLOWER/NPRE
      DO 50 I=2, NPRE
      YPS(I)=0.
      ZPS(I)=UPPER*(I-1)
      CALL YZCOMP (YPS(I), ZPS(I), YP(I), ZP(I))
      J=I+M1
      YPS(J)=0.
      ZPS(J)=XLOWER*(I-1)
50    CALL YZCOMP (YPS(J), ZPS(J), YP(J), ZP(J))
      CZVS=CMPLX(YSTARV, ZSTARV)
      DO 52 I=1, N
      I2=2*I
52    CZS(I)=CMPLX(YS(I2), ZS(I2))
      DO 53 I=2, M1
      J=I+M1
      IM1=I-1
      JM1=J-1
      CALL PHII (YPS(I), ZPS(I), YP(I), ZP(I), PHIX (I), MLP)
      IF (MLP .EQ. 1) 54, 55
54    NOPHI(I)=1
55    CALL PHII(YPS(IM1), ZPS(I), YP(IM1), ZP(I), PHIXY(I), MLP)
      IF (MLP .EQ. 1) 4, 5
 4    NOPHI(I)=1
 5    CALL PHII(YPS(I), ZPS(IM1), YP(I), ZP(IM1), PHIXZ(I), MLP)
      IF (MLP .EQ. 1) 6, 7
 6    NOPHI(I)=1
 7    CALL PRESS (YPS(I), ZPS(I), YP(I), ZP(I), PHIXO(IM1), PHIX(I), PHI
     1XY(I), PHIXZ(I), PRE(I))
      CALL PHII (YPS(J), ZPS(J), YP(J), ZP(J), PHIX (J), MLP)
```

```
    IF (MLP .EQ. 1) 56, 57
56  NOPHI(J)=1
57  CALL PHII(YPS(JM1), ZPS(J), YP(JM1), ZP(J), PHIXY(J), MLP)
    IF (MLP .EQ. 1) 8, 9
 8  NOPHI(I)=1
 9  CALL PHII(YPS(J), ZPS(JM1), YP(J), ZP(JM1), PHIXZ(J), MLP)
    IF (MLP .EQ. 1) 10, 11
10  NOPHI(I)=1
11  CALL PRESS (YPS(J), ZPS(J), YP(J), ZP(J), PHIXO(JM1), PHIX(J), PHI
   1XY(J), PHIXZ(J), PPE(J))
53  PRINT 60, I, YP(I), ZP(I), PRE(I), NOPHI(I), I, YP(J), ZP(J), PRE(
   1J), NOPHI(J)
    DO 12 KJ=1, M3
12  NOPHI(KJ)=0
    IPRIT=IPRIT+1
61  FORMAT (1H0, 1X, 36HPRESSURE COEFF. ON UPPER SURFACE ARE, 24X, 36H
   1PRESSURE COEFF. ON LOWER SURFACE ARE //1H0, 1X, 1HI, 8X, 1HY, 13X,
   2 1HZ, 12X, 2HCP, 7X, 1H*, 1X, 1H$, 1X, 1HI, 8X, 1HY, 13X, 1HZ, 12X
   3, 2HCP, 7X, 1H*/61X, 1H$)
60  FORMAT (61X, 1H$/1X, I2, 3(2X, E12.5), 2X, I1, 1X, 1H$, 1X, I2, 3(
   12X, E12.5), 2X, I1)
    RETURN
    END
```

```
      SUBROUTINE PHI(YPS, ZPS, YP, ZP, PHI, M)

      THIS SUBROUTINE CALCULATES THE IMAGINE PART OF THE POTENTIAL
      FUNTION AT ANY POSITION X

      COMMON/1/RADIUS, AZAXIS, BYAXIS, S, BETA
      COMMON/2/NSHAPE
      COMMON/4/ALPHA, AC, DELTA, DELTAF, SPANF, XSPF, TANAF, SINA1, SINA
     12, COSA2, COSA3, COSBT
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/6/GVORT, GVORTO, GS(50), GSO(50), PS(51), THETAS(50), YS(50
     1), ZS(50), D(51), H(52), R(50), RO(50), THETA(50), Y(50), Z(50), R
     2SV, THETSV, YSTARV, ZSTARV, YV, YVO, ZV, ZVO, RV, RVO, THETAV
      COMMON/16/ROOTR, ROOTI
      COMMON/19/DBDX, DRDDX
      COMMON/27/DH(50), HD(25), DD(50), DRR(50), HH(25)
      COMMON/31/CZVS, CZS(25)

      COMPLEX CZPS, CZVS, CZS, CL

      M=0
      CZPS=CMPLX(YPS, ZPS)
      SUM=0.
      DO 1 I=1, N
      I2=2*I
      CL=CLOG((CZPS-CZS(I))/(CZPS+CONJG(CZS(I))))
      IF (ZPS .GT. ZS(I2)) 2, 3
    3 IF (ZPS .EQ. ZS(I2)) 4, 5
    5 IF (ZPS .GT. 0.) 6, 7
    6 CL=CL+(TWOPI*(0., 1.))
      GO TO 2
    7 IF (ZPS .EQ. 0.) 4, 2
    2 FACT=AIMAG(CL)
    1 SUM=SUM+GS(I2)*HH(I)*FACT
      CL=CLOG((CZPS-CZVS)/(CZPS+CONJG(CZVS)))
      IF (ZPS .GT. ZSTARV) 8, 9
    9 IF (ZPS .EQ. ZSTARV) 4, 10
   10 IF (ZPS .GT. 0.) 11, 12
   11 CL=CL+(TWOPI*(0., 1.))
      GO TO 8
   12 IF (ZPS .EQ. 0.) 4, 8
    8 FACT=AIMAG(CL)
      IF (NSHAPE .EQ. 0) 14, 15
   14 WSR=RADIUS*DRDDX*.5*ALOG(YP**2+ZP**2)
      GO TO 16
   15 WSR=AZAXIS*DBDX   *.5*ALOG((YP+ROOTR)**2+(ZP+ROOTI)**2)
   16 PHI=TANAF*ZPS+GVORT*FACT/TWOPI+SUM/(2.*TWOPI)+WSR
      RETURN
    4 M=1
      PHI=0.
      RETURN
      END
      SUBROUTINE PRESS (YPS, ZPS, YP, ZP, PHIO, PHI, PHIXY, PHIXZ, CP)
C     THIS SUBROUTINE CALCULATES PRESSURE COEFFICIENT
      COMMON/3/DX, XO, X, XXO
      COMMON/4/ALPHA, AF, DELTA, DELTAF, SPANF, XSPF, TANAF, SINA1, SINA
     12, COSA2, COSA3, COSBT
      COMMON/5/PI, TWOPI, PIO2, THPIO2, RAD, N, N2, N2M1, N2P1
      COMMON/6/GVORT, GVORTO, GS(50), GSO(50), RS(51), THETAS(50), YS(50
     1), ZS(50), D(51), H(52), R(50), RO(50), THETA(50), Y(50), Z(50), R
     2SV, THETSV, YSTARV, ZSTARV, YV, YVO, ZV, ZVO, RV, RVO, THETAV
      COMMON/27/DH(50), HD(25), DD(50), DRR(50), HH(25)
      CALL DERIVE (YPS, ZPS, YP, ZP, DR, DI)
      ABSDSQ=1./(DR**2+DI**2)
      CALL SOURCE (YP, ZP, DR, DI, SR, SI)
      YSMYSV=YPS-YSTARV
      YSPYSV=YPS+YSTARV
      ZSMZSV=ZPS-ZSTARV
      YDENV=(YSMYSV**2+ZSMZSV**2)*(YSPYSV**2+ZSMZSV**2)
      SUMR=0.
      SUMI=0.
      DO 2 K=1, N
      K2=2*K
      YSMYS=YPS-YS(K2)
      YSPYS=YPS+YS(K2)
      ZSMZS=ZPS-ZS(K2)
      YDENS=(YSMYS**2+ZSMZS**2)*(YSPYS**2+ZSMZS**2)
      SUMR=SUMR+GS(K2)*HH(K)*YS(K2)*ZSMZS*(YSMYS+YSPYS)/(TWOPI*YDENS)
    2 SUMI=SUMI+GS(K2)*HH(K)*YS(K2)*(YSMYS*YSPYS-ZSMZS**2)/(TWOPI*YDENS)
      DWDZSR    =-GVORT    *YSTARV*ZSMZSV*(YSMYSV+YSPYSV)/(PI*YDENV)-SUMR+
     1SR
      DWDZSI    =-(GVORT    *YSTARV*(YSMYSV*YSPYSV-ZSMZSV**2)/(PI*YDENV)+S
     1UMI+TANAF-SI)
      DWSQ=DWDZSR**2+DWDZSI**2
      PHIX=(PHIXY+PHIXZ-PHI-PHIO)/(X-XO)
      CP=SINA2-(2.*PHIX+ABSDSQ*DWSQ)*COSA2
      RETURN
      END
```