NASA CR 137579

THEORETICAL PREDICTION OF THICK WING AND PYLON-FUSELAGE-FANPOD-NACELLE AERODYNAMIC CHARACTERISTICS AT SUBCRITICAL SPEEDS

PART II - COMPUTER PROGRAM DESCRIPTION

By J. Kojima and J.R. Tulinius

July 24, 1974

Prepared under Contract No. NAS2-7904 by

LOS ANGELES AIRCRAFT DIVISION
ROCKWELL INTERNATIONAL
International Airport
Los Angeles, California 90009

for

AMES RESEARCH CENTER
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

NASA CR 137579

THEORETICAL PREDICTION OF THICK WING AND PYLON-FUSELAGE-FANPOD-NACELLE AERODYNAMIC CHARACTERISTICS AT SUBCRITICAL SPEEDS

PART II - COMPUTER PROGRAM DESCRIPTION

By J. Kojima and J.R. Tulinius

July 24, 1974

Prepared under Contract No. NAS2-7904 by

LOS ANGELES AIRCRAFT DIVISION
ROCKWELL INTERNATIONAL
International Airport
Los Angeles, California 90009

for

AMES RESEARCH CENTER
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

THEORETICAL PREDICTION OF THICK WING AND

PYLON-FUSELAGE-FANPOD-NACELLE

AERODYNAMIC CHARACTERISTICS

AT SUBCRITICAL SPEEDS

PART II - COMPUTER PROGRAM DESCRIPTION

By J. Kojima and J.R. Tulinius
Los Angeles Aircraft Division
Rockwell International

SUMMARY

This report describes the procedures required to operate the thick wing and pylon-fuselage-fanpod-nacelle computer program. The program computes surface velocities and pressures, section loads, and total configuration loads and pitching moment. Potential flow theory is used to compute the surface pressures and the associated lift, moment, and vortex drag. The skin friction drag is also computed.

Included in this report is a description of the program set up, input data, and output.

THIS PAGE
INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE
INTENTIONALLY LEFT BLANK

INTRODUCTION

The program was programmed for the Lawrence Berkley Laboratory CDC 7600 computer. The source deck was keypunched using the 026 Hollerith punch (BCD). The program was compiled with the run 76 compiler. Both the source and object decks are stored on a data cell at the Lawrence Berkley Laboratory using the CDC Update system.

The simplest and fastest way of estimating computing time and lines of output is by comparing known cases with cases to be run. The approximate length of computation times for the cases in this report were as follows:

CASE I:

a. Case 1. Fuselage-wing with 195 by 399 influence matrix A and $\alpha = 0$ degrees.

b. Case 2. Use case 1 influence matrix A and set $\alpha = 3$ degrees.

The computing units (CUS) were 1400 seconds and the central processor (CPU) time was 270 seconds.

CASE II:

a. Case 1. Fuselage-wing with 192 by 360 influence matrix A and $\alpha = 0$ degrees.

b. Case 2. Use case 1. influence matrix A and set $\alpha = 3$ degrees.

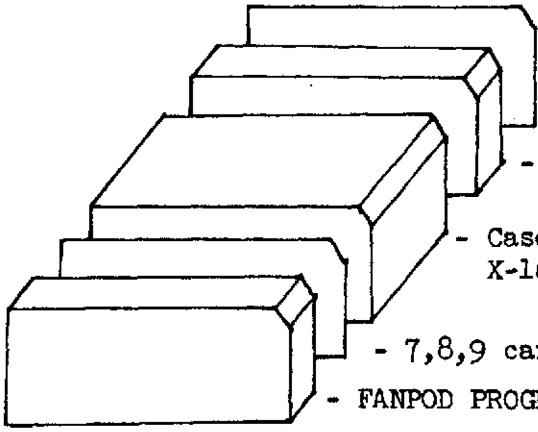
c. Case 3. Fuselage-fanpod-wing with 312 by 588 influence matrix A and $\alpha = 0$ degrees. Reused fuselage matrix A of case 1.

d. Case 4. Used the influence matrix A of case 3 and set $\alpha = 3$ degrees.

The computing units (CUS) were 5000 seconds and the central processor (CPU) time was 1000 seconds.

The lines of output were set by the job control card LGØ, LC = 10000., which should be sufficient for most jobs.

PROGRAM SETUP



REMARKS:

1. All data are read in by subroutine DECRD.
2. Data used by one case are always available for use in succeeding cases unless read over by new data in these cases.
3. The card containing the last piece of decimal data for a case contains a minus in column one.

***** SETUP TO COMPILE SOURCE DECK THEN STORE SOURCE AND OBJECT DECKS ON THE DATA CELL USING THE UPDATE SYSTEM.

```
A04NZXX,5,250,70000.482511,RCST5786P FANPOD
*NOTAPES,PSS
UPDATE(N)
RUN76(SA,I=COMPILE,NL50000)
REWIND(NEWPL,LGO)
COPYPSS.
CXIT.
REWIND(NEWPL,LGO)
COPYPSS.
(7-8-9) CARD
      .
      .
      .
      .
      .
(FANPOD SOURCE DECK)
      .
      .
      .
      .
      .
(7-8-9) CARD
GROUP=121,AERO
OWNER=R.C.SMITH
ACCOUNT=482511
WRITE LIBRARY=A04RCS
REPLACE SUBSET=UPDATPL
COPY 1 F FROM FILE=NEWPL
REPLACE SUBSET=OBJECT
COPY 1 F FROM FILE=LGO
END
(6-7-8-9) CARD
      .
      .
```

***** TO RUN PROGRAM WITH TEMPORAY MODIFICATIOMS

```
A04NZXX,5,4000,60000.482511,RCST5786P FANPOD
*NOTAPES,PSS
LIBCOPY,A04RCS,OLDPL,UPDATPL.
LIBCOPY,A04RCS,LIB,OBJECT.
UPDATE.
RUN76,I=COMPILE.
COPY,LIB/RX,LGO.
RFL(135000,[illegible]0000)
LGO,LC=10000.
(7-8-9) CARD
      .
      .
      .
( UPDATE CORRECTIONS)
      .
      .
      .
(7-8-9) CARD
      .
      .
      .
( INPUT DATA )
      .
      .
      .
(6-7-8-9) CARD
```

****** SETUP FOR PERMANENT MODIFICATIONS

```
A04NZXX,5,100,60000.482511,RCST5786P FANPOD
*NOTAPES,PSS
LIBCOPY,A04RCS,OLDPL,UPDATPL.
UPDATE(F,N)
RUN76(S,I=COMPILE,NL20000)
COPYPSS.
(7-8-9) CARD
      .
      .
      .
      .
      .
( UPDATE CORRECTIONS)
      .
      .
      .
      .
      .
(7-8-9) CARD
GROUP=121,AERO
OWNER=R.C.SMITH
WRITE LIBRARY=A04RCS
REPLACE SUBSET=UPDATPL
COPY 1 F FROM FILE=NEWPL
REPLACE SUBSET=OBJECT
REWIND INFILE=LGO
COPY 1 F FROM FILE=LGO
REWRITE LIBRARY=A04RCS
END
(6-7-8-9) CARD
```

****** PRODUCTION RUN WITH NO MODIFICATIONS

```
A04NZXX,5,1400,60000.482511,RCST5786P FANPOD
*NOTAPES,PSS
LIBCOPY,A04RCS,LIB,OBJECT.
COPY,LIB/RX,LGO.
RFL(135000,500000)
LGO,LC=10000.
(7-8-9) CARD
      .
      .
      .
      .
      .
      .
( INPUT DATA )
      .
      .
      .
      .
      .
      .
(6-7-8-9) CARD
```

DESCRIPTION OF SUBROUTINES

NUMBER	SYMBOL	DESCRIPTION
1	MAIN	Main program controls general flow of program.
2	CØDIM	A control derivation interpolation method for a single curve.
3	WINGD	Table lookup of wing planform for X leading edge and Chord.
4	PFUNC	Calculate matrix S of P function (special spanwise load shape).
5	MSØLX	Special Householder method for solving a set of linear simultaneous equations (variable columns of B).
6	RDATA	Read, print and test input data.
7	DECRD	Read input data from cards.
8	GEØM	Geometry control program.
9	BXYZ	Calculate coordinate of fuselage vortices.
10	BQPTS	Calculate fuselage direction matrices and coordinates of control points.
11	BCLS	Calculate chordwise load shapes for the fuselage.
12	FXYZ	Calculate coordinate of fanpod vortices.
13	FQPTS	Calculate fanpod direction matrices and coordinates of control points.
14	FCLS	Calculate chordwise load shapes for the fanpod.
15	WXYZ	Calculate coordinate of wing vortices and control points.
16	BRXYZ	Calculate coordinate of fuselage-wing root section vortices.
17	FRXYZ	Calculate coordinate of fanpod-wing root sections vortices.

18	RTWI	Rotate wing root coordinates to the actual chord lines.
19	TWIST	Lookup twist angle and calculate CØS and SIN of twist.
20	MATCS	Calculate wing chordwise and spanwise matrices (C and S).
21	WTZCS	Calculate wing thickness slopes from deflections (z/c).
22	PXYZ	Calculate coordinates of pylon vortices and control points.
23	PTZCS	Calculate pylon thickness slopes from deflections (z/c).
24	NXYZ	Setup nacelle coordinates.
25	CØDMT	A controlled deviation interpolation method for multiple curves.
26	MATA	Control program for the influence matrix A computation routines.
27	AQXVB	Calculate matrix A with Q on X component and vortices on fuselage.
28	AQXVF	Calculate matrix A with Q on X component and vortices on fanpod.
29	AQXVW	Calculate matrix A with Q on X component and vortices on wing.
30	AQXVWR	Calculate matrix A with Q on X component and vortices on wing root section.
31	AQWVW	Calculate matrix A with Q on wing and vortices on wing.
32	AQWVWT	Calculate influence of wing thickness due to cranks, etc. on the velocity.
33	AQXVP	Calculate matrix A with Q on X component and vortices on pylon.
34	AQXVPR	Calculate matrix A with Q on X component and vortices on pylon root section

35	AQPVP	Calculate matrix A with Q on pylon and vortices on pylon.
36	AQXVPT	Calculate influence of pylon thickness due to cranks, etc. on the velocity.
37	NACELL	Control program for nacelle influence matrix A routines.
38	AQXVN	Compute matrices X,Y,Z components of nacelle matrix A and influence matrices of other components on the nacelle.
39	XYZ	Control for X,Y,Z matrices computation.
40	XYZ1	Compute X,Y,Z matrices for SJ less than 0.08.
41	XYZ2	Compute X,Y,Z matrices using Simpson rule integration.
42	ELIP	Calculate Hasting's approximation for elliptic integrals.
43	MATAPE	Combine influence matrix A and save on I/Ø unit 12.
44	MATB	Control program for boundary condition matrix B routines.
45	WCAS	Table lookup for the wing local angle of attack.
46	WCZCS	Calculate wing local angle of attack from deflection (z/c) input.
47	PCAS	Table lookup for the pylon local angle of attack.
48	PCZCS	Calculate pylon local angle of attack from deflection (z/c) input.
49	SØLU	Control program for matrix solution routines.
50	MSØL	Special Householder method for solving a set of linear simultaneous equations with matrix A on various I/Ø units.
51	MSØLP	Special Householder method for solving a set of linear simultaneous equations with the results save on various I/Ø units.
52	PARTM	Partision-Householder method for solving a set of linear simultaneous equations.

53	PLCAL	Control program for pressure computation routines.
54	BVRCP	Calculate fuselage velocity ratio and pressure coefficients at control points.
55	FVRCP	Calculate fanpod velocity ratio and pressure coefficients at control points.
56	WDPQ	Calculate wing linear pressure coefficients.
57	WVAP	Calculate wing nonlinear velocities and pressure coefficients with thickness.
58	PDPQ	Calculate pylon linear pressure coefficients.
59	PVAP	Calculate pylon nonlinear velocities and pressure coefficients with thickness.
60	SFC	Calculate wing F matrix.
61	TPS	Calculate wing-pylon leading and trailing edges TAN Ø*.
62	SPF	Calculate the derivatives of the wing F matrix.
63	BLØAD	Calculate the load coefficients for the fuselage.
64	FLØAD	Calculate the load coefficients for the fanpod.
65	WLØAD	Calculate the load coefficients for the wing.
66	PLØAD	Calculate the load coefficients for the pylon.
67	IDRAG	Calculate induced drag.
68	EMLQP	Computes coefficients for Emlord equations.
69	SKINF	Calculate skin friction drag.
70	SETSF	Setup skin friction drag input.
71	CØD	Special CØDIM.

INPUT DATA DESCRIPTION

INPUT DESCRIPTION

LOCATION	SYMBOL	DESCRIPTION
1	AR	Wing aspect ratio
2	SPAN	Wing span
3	MAC	Mean aerodynamic chord
4	MACH	Mach number
5	FCD	Fanpod chord
6	FAA	Fanpod angle of attack in degrees
7	XCG	X center of gravity for moment computation
8	WAA	Wing angle of attack in degrees
9	WAAI	0.0 if wing is flat and 1.0 if wing has twist or camber
10	WADI	0.0 if wing camber is described by local angle of attack in radians. 1.0 if camber is described by Z/c. Wing camber is described in locations 750 to 1199.
12	PRII	1.0 if the boundary condition matrix B, and the least squared boundary condition matrix B L.S. are to be printed. 0.0 if only matrix B is to be printed.
14	WSJC	Value of first J in the chordwise direction on the wing to be used in the lifting solution. This value should be such that the first control point is located at about x/c=.05. This value of J should be included in the list at loc. 1660-1689. The thickness solution will use all of the control points listed at loc. 1660-1689.
15	FMFI	This input indicates if the multiplication factors listed in loc. 600-634 are the same as those listed in loc. 635-669. If they are, loc. 635-669 may be omitted.

16	FCPI	This input indicates if the fanpod cross-sections described in loc. 90-564 are to be placed ⊥ to the camber line described in loc. 670-704 or ⊥ to the fanpod X axis. Use 0.0 if sections are to be placed ⊥ to the camber line and 1.0 if they are to be placed ⊥ to the X axis.
17	FLØI	This input indicates the type of vrotex grid input to be used to describe the fan-pod vortex grid in the chordwise direction. If the grid is to be placed at equal increments of X, input -1.0. No input is needed in loc. 1460-1609 if 0.0 or -1.0 is input here. If the grid is to be placed at X stations other than at equal increments of Ø or X, input a 1.0 here and list the X stations in loc. 1460-1609.
18	FTHI	This input indicates the type of fanpod vortex grid to be used to describe the lateral vortex grid distribution. If the grid is to be placed at equal roll angles Θ, input 0.0 here and no input is required in loc. 1610-1659. If a lateral vortex grid other than equal roll angles is desired, input 1.0 and list the values in loc. 1610-1659.
19	PVPI	Input here the number of vortices in the spanwise direction on the pylon minus the number of span stations where the pressures are to be computed on the pylon. This input is used to limit the spanwise pressure calculation to those points on the actual pylon rather than those on the pylon extension, which is inside the nacelle.
20	FNXS	Input the number of chordwise stations where the fanpod cross sections are to be described. The maximum number is 29.
21-49	$XS_1, XS_2, \cdots$	List of fanpod chord stations where the fanpod cross sections are to be described.
50	FNTY	Input the number of lateral fanpod stations where the fanpod cross sections are to be described. The maximum number is 36.

51-86	$\theta_1, \theta_2, \cdots$	List of fanpod lateral stations where the fanpod cross sections are defined. The list of θ's are input from 0.0 to 360.0 degrees.
87	FØX	X location of the nose of the starboard fanpod.
88	FØY	Y location of the nose of the starboard fanpod.
89	FØZ	Z location of the nose of the starboard fanpod.
90-564	$R_1, R_2, \cdots$	List of radii for fanpod at XS and θ locations. Input radii for first XS station and all of the θ's from 0.0 to 360.0 degrees, then the second XS station and all the θ's again. Continue this process for all of the XS stations.
565	FNXM	Input the number of chordwise fanpod stations where the YM and ZM multiplication factors will be applied and where the fanpod camber is defined. The body described by the radii in loc. 90-564 will be multiplied by the YM and ZM multiplication factors in the Y_F and Z_F directions, respectively. The fanpod camber will then be added to obtain the final description of the fanpod. It is not necessary to use the multiplication factors as a means of describing the fanpod. However, if the fanpod cross sections can be represented by a series of ellipses, the YM and ZM distributions can be used to input the major and minor axses of the elliptical cross sections. For this case only four radii, equal to unity, need be input. The maximum number of chordwise stations is 34.
566-599	$XMC_1, XMC_2, \cdots$	List of chordwise stations where the fanpod multiplication factors and camber are defined.
600-634	$YMC_1, YMC_2, \cdots$	List of YM fanpod multiplication factors. The Y components of the radii defined in loc, 90-564 will be multiplied by these factors.

635-669	$ZMC_1, ZMC_2, \cdots$	List of ZM fanpod multiplication factors. The Z components of the radii defined in loc. 90-564 will be multiplied by these factors.
670-704	$FZC_1, FZC_2, \cdots$	Distribution of fanpod camber. This camber will be applied to the fanpod after the radii defined in loc. 90-564 are multiplied by the multiplication factors defined in loc. 600-634 and loc. 635-669. If the radii as given in loc. 90-564 are for the actual fanpod and the use of multiplication factors or camber is not necessary, set FNXM=1.0, YM_1=1.0, ZM_1=1.0, and FZC_1=0.0.
705	WNX	The number of X/C stations where the wing mean camber surface will be defined in loc. 750-1199. The maximum number is 23.
706-729	$X/C_1, X/C_2, \cdots$	The array of X/C values where the wing mean camber surface will be defined in loc. 750-1199. The array must be in ascending order and must include the leading and trailing edge points.
730	WNE	The number of η stations where the wing mean camber surface will be defined in loc. 750-1199. The maximum number is 15.
731-749	$\eta_1, \eta_2, \cdots$	The array of η values where the wing mean camber surface will be defined in loc. 750-1199. The array must be in ascending order and the range must be such as to enclose all the spanwise control points.
750-1199		This table defines the wing mean camber surface in terms of local angle of attack in radians or in terms of Z/C. The contour described here excludes twist and flap deflections. The twist is defined in loc. 1970-1984. The flap deflections are given in loc. 1960 and 1965. If Z/C is input the program uses CODIM, described in Appendix A, to determine the local angles of attack. If the contour is described at stations other than the control point locations, the local angles of attack are obtained by straight line interpolation on Z in the spanwise direction and then by using CODIM in the chordwise direction. The local angles of attack or Z/C's are input for the first η station defined at loc. 731 and the X/C's defined at loc. 706-729, then the second η station, and etc.

1200	WØX	X location of wing apex.
1201	WØY	Y location of wing apex.
1202	WØZ	Z location of wing apex.
1203	WITØ	This location specifies the number of the fanpod meridian line to which the outboard panel of the wing attaches. The meridian lines are numbered in ascending order in the clockwise direction around the fanpod when looking at the starboard fanpod from rear to front. The meridian line on top of the fanpod is considered the last.
1204	WITI	This location specifies the number of the fanpod meridian line to which the inboard panel of the wing attaches. The meridian lines are numbered in ascending order in the clockwise direction around the fanpod when looking at the starboard fanpod from rear to front. The meridian line on top of the fanpod is considered the last.
1205	WNVSØ	Number of vortices in the spanwise direction on the outboard wing panels. This number includes the vortices in the outboard root sections as well as the outboard planar panels. Both starboard and port wing panels are included in this number. This input is an even integer for wing only case and an odd integer for all other combinations. The maximum number is 50.
1206	WNVSI	Number of vortices in the spanwise direction on the inboard wing panels. Both starboard and port wing inboard root and planar sections are included in this number. This input is an odd integer if the combination contains no fuselage and an even integer with a fuselage. The maximum number is 50.
1207	WITB	This location specifies the number of the fuselage meridian line to which the wing attaches. The meridian lines are numbered in ascending order in the clockwise direction around the fuselage when looking from the rear to front.
1208	XNEØ	Number of η in outer wing table in loc. 1210-1239.

1209 XNEI Number of η in inner wing table in loc. 1240-1269.

1210-1269 η_1, X_{LE}, X_{TE}, This input describes the wing planform. If a wing alone case is being analyzed, the X distances from the wing apex to the wing leading and trailing edges are input at a series of span stations. The leading and trailing edges are straight lined between the input stations. If a freestream edge results from a flap chord extension or nacelle planview, at any station other than the center or tip, input the freestream edge over an increment of $\Delta\eta$ of .0001. The η stations where the wing planform is defined are input for $0 \leq \eta \leq 1.0$. Input the η station, then the X distance of the leading edge, and then the X distance of the trailing edge. Repeat this for each η station.

If a fanpod is included in the analysis, the wing planform is described as an inboard and outboard panel. Each of these panels is divided into a root section and planar panel. For the outboard panel the leading edge of the root section is divided into three evenly spaced divisions. The width of these divisions is equal to $\Delta\eta_0 = 2(1-\eta_{01})/(\text{WNVSØ}-1)$, where η_{01} is the η station where the leading edge of the outboard panel attaches to the fanpod. η_{01} is input in loc. 1210, $\eta_{01} + \Delta\eta_0$ in loc. 1213, $\eta_{01} + 2\Delta\eta_0$ in loc. 1216, and $\eta_{01} + 3\Delta\eta_0$ in loc. 1019. The X distances of the wing leading edge from the wing apex at η_{01}, $\eta_{01}+\Delta\eta_0$, $\eta_{01}+2\Delta\eta_0$, and $\eta_{01}+3\Delta\eta_0$ are input at loc. 1212, 1214, 1217, and 1220, respectively. The X distances of the wing trailing edge from the wing apex at η_{01} and $\eta_{01}+3\Delta\eta_0$ are input at loc. 1212 and 1221, respectively.

S'_{01} and S'_{02} are input at loc. 1215 and 1218, respectively. Where S'_{01} and S'_{02} are perpendicular distances from the outer fanpod trailing edge fillet chord line to the wing trailing edge. The chord line is drawn between the trailing edge points at η_{01} and $\eta_{01} + 3\Delta\eta_0$. S'_{01} and S'_{02} are measured from the chord line to the wing trailing edge at points 1/3 and 2/3 of the distance along the chord line going from η_{O1} to $\eta_{O1}+3\Delta\eta_O$, respectively. A sketch showing these quantities is given in the input format description at loc. 1220. The planar wing panel is described in the same manner as the wing alone case.

The inboard root section leading edge is also divided into three evenly spaced divisions. The width of these divisions is equal to $\Delta\eta_i=2\eta_N/(\text{WNVSI}-1)$, where η_N is the η station where the leading edge of the inboard panel attaches to the fanpod. The inboard panel input starts in loc. 1240. The planar portion of the inboard panel is input in the same manner as the wing alone case. The inboard root section starts at $\eta_N-3\Delta\eta_i$. At $\eta=\eta_N-3\Delta\eta_i$ the X distances from the wing apex to the leading and trailing edges are input. At $\eta=\eta_N-2\Delta\eta_i$ and $\eta=\eta_N-\Delta\eta_i$ the X distances from the wing apex to the leading edge and the S_1' and S_2' distances are input, respectively. The distances S_1' and S'_2 are obtained for the inboard fanpod fillet in the same manner as S_{O1}' and S_{O2}' for the outboard fanpod fillet. If there are no fanpod fillets on the wing trailing edge the S' values are equal to zero. At η_N the X distances from the wing apex to the leading and trailing edges are input. A sketch of the inboard root section is shown in the input format description.

1270 WNVC Input the number of vortices and sources in the chordwise direction. Usually 20 is enough. The maximum number is 40.

1272 WNPC This is the number of net pressure coefficients to be computed in the chordwise direction on the wing. The list of X/C's where the coefficients are computed is given in loc. 1870-1899. The maximum number is 30. This input is usually unnecessary for the thick wing case since the net pressure coefficients are printed at the control point locations.

1273	WNPS	This is the number of net pressure coefficients to be computed in the spanwise direction on the wing. The list of η stations where the pressure coefficients are printed are given at loc. 1900-1929. The maximum number is 30. This input is unnecessary if a spanwise discrete analysis is done. If a thick wing analysis is done the pressure coefficients are automatically printed at the control point locations making this input unnecessary.
1274	FNVX	This is the number of vortex grid lines in the longitudinal direction on the fanpod. There is one more line than the number of vortices in the longitudinal direction. The nose and tail end of the fanpod must be included. The maximum number is 100. The list of X grid stations is given in loc. 1460-1609.
1275	FNVY	This is the number of lateral vortex grid lines around the fanpod. This number should be kept as small as possible, 8 to 12, in order to keep computing time down. The list of Θ angles for the lateral grid lines is given in loc. 1610-1659. The maximum number is 40.
1276	FNDY	This is the number of divisions a circumferential fanpod vortex line segment is divided into between adjacent lateral vortex grid lines. The number is an odd integer. Usually equal to 3 to 7. These divisions permit the fanpod vortex grid to better map the actual contour.
1277	WNJC	This is the number of control points per chord on the wing. The list of control points is given in loc. 1660-1689. The maximum number is 30. Usually WNJC $\approx$ 2 WNU.
1278	WNJS	This is the number of control points per semispan on the wing. The list of control point locations is given in loc. 1690-1719. Usually WNJS $\approx$ 2 WNW. This input is unnecessary if a discrete solution is done in the spanwise direction on the wing. The maximum number is 30.

1279	FNJX	This is the number of control points in the longitudinal direction along a meridian line on the fanpod. The list of control points is given in loc. 1720-1869. Usually FNJX $\approx$ 2 FNF. The maximum number is 75.
1280	WNU	This is the number of assumed vorticity functions in the chordwise direction on the wing. The functions are $\mathrm{COT}\theta/2$, $\mathrm{SIN}\theta$, $\mathrm{SIN}2\theta, \cdots$ plus two flap functions. The number of SINE terms used is equal to (WNU-1) minus the number of flap terms designated in loc. 1960 and 1965. Usually 6 to 8 functions in total are sufficient. The maximum number is 10.
1281	WNW	The number or assumed spanwise vorticity functions on the wing. If this input is zero, a discrete spanwise wing solution will be obtained. The standard functions used are $\eta^{W}\sqrt{1-\eta^2}$, where the power W is input in loc. 1285-1294. Also, special functions can be used to account for discontinuities in the leading or trailing edge sweep, flaps, or fanpod-wing juncture. These functions are designated in the loc. 1285-1294 list by the location number in the input where they are described. The special spanwise functions are described in loc. 1930, 1935, 1940, 1945, 1950, and 1955. The maximum number of spanwise functions is 10.
1282	FNF	The number of chordwise vorticity functions used along each meridian line. There are standard functions listed in the input format description at loc. 1295. Also, special functions can be input if the standard functions are insufficient. The special functions are described at loc. 1310, 1360, and 1410. The special functions are designated in the vorticity function list at loc. 1295-1319 by the input location where they are described. The maximum number of fanpod longitudinal functions is 25.

1285-1294	$W_1, W_2 \cdots$	This is the list of spanwise vorticity functions used on the wing. The functions desired are indicated by the power W of the series $\eta^W\sqrt{1-\eta^2}$ or by the input location number where a special function is described; such as 1930, 1935, 1940, 1945, 1950, or 1955. Since only symmetrical loading conditions can be treated W must always be an even integer. The elliptical loading is designated by W=0. The usual values of W are 0, 2, 4, 6, and 8.
1295-1319	$F_1, F_2 \cdots$	This is the list of longitudinal functions used on the fanpod. The functions available are designated as follows:

F for Standard Functions		F for Linear Functions
1.0	COTØ/2	$(x/c)_1$
2.0	COT(90-Ø/2)	$(x/c)_2$
3.0	SINØ	$(x/c)_3$
4.0	COSØ	$(x/c)_4$
5.0	SIN2Ø	$(x/c)_5$
6.0	COS2Ø	$(x/c)_6$
.	.	.
.	.	.
.	.	.

1460-1609	$FVX_1, FVX_2, \cdots$	This is the list of X stations for the fanpod vortex grid. The list must include $X_F=0$ and $X_F=C_F$. The maximum number is 100.
1610-1659	$FVT_1, FVT_2, \cdots$	This is the list of lateral vortex grid stations. This list of θ's refer to the fanpod body before the multiplication factors and camber are applied. The θ's are input from 0.0 to sum value less than 360 degrees. The first θ must be zero. The maximum number is 40.
1660-1689	$WJC_1, WJC_2, \cdots$	This is the list of control points in the chordwise direction on the wing. The control point is designated by the number of the vortex aft of the leading edge on which the control point is placed. The maximum number is 30. However, a particular case can never exceed WNVC. in loc. 1270.

1690-1719	$WJS_1, WJS_2, \cdots$	This is the list of control points in the spanwise direction on the wing. Since the configuration is symmetrical about the X-Z plane, control points are only placed on the starboard side. The control point is designated by the number of the vortex inboard of the starboard wing tip on which the control point is placed. The maximum number is 30. However, in a particular case the number cannot exceed (WNVS+1)/2 for the wing alone case or in the case of a fanpod on the configuration (WNVSØ+WNVSI+1)/2. This input is unnecessary if a discrete solution is done in the spanwise direction on the wing.
1720-1869	$FJX_1, FJX_2, \cdots$	This is the list of control points on the fanpod in the longitudinal direction. The control point is designated by the number of the vortex aft of the fanpod nose on which the control point is placed. The maximum number is 75. However, the number cannot exceed FNVX-1.
1870-1899	$WPC_1, WPC_2, \cdots$	This is the list of X/C stations on the wing where the net pressure coefficients are printed. The maximum number is 30. Unnecessary if wing has thickness.
1900-1929	$WPS_1, WPS_2, \cdots$	This is the list of η stations on the wing where the net pressure coefficients are printed. The maximum number is 30. This input is not necessary if spanwise solution is discrete or if wing has thickness.
1930		Indicator for special wing spanwise vorticity function. If "P" function is used input 0.0, if trailing edge flap function is used input-1.0, and if leading edge flap or Krueger function is used input 1.0.
1931	η^*	ETA station where break in leading and trailing edge of wing occurs. This input is associated with a "P" function. Leave blank for flap functions. η^* should be at a control point location.

1932	RL, f_i, K_i	The left hand range of influence of the "P" function downwash is input if loc. 1930 is 0.0. For this case RL=.02 is a suggested value. If a leading or trailing edge flap function is indicated in loc. 1930, input the η station of the inboard discontinuity in local angle of attack due to the flap deflection. This ETA station should occur at the edge of a vortex strip.
1933	RR, f_O, Y_O	The right hand range of influence of the "P" function downwash is input if loc. 1930 is 0.0. For this case RR=.02 is a suggested value. If a leading or trailing edge flap function is indicated in loc. 1930, input the η station of the outboard discontinuity in local angle of attack due to the flap deflection. This ETA station should occur at the edge of a vortex strip.
1935-1938		For second special wing spanwise vorticity function. Input similar to locations 1930-1933.
1940-1943		For third special wing spanwise vorticity function. Input similar to locations 1930-1933.
1945-1948		For fourth special wing spanwise vorticity function. Input similar to locations 1930-1933.
1950-1953		For fifth special wing spanwise vorticity function. Input similar to locations 1930-1933.
1955-1958		For sixth special wing spanwise vorticity function. Input similar to locations 1930-1933.
1960	δ_f	Deflection of wing trailing edge flap in radians.
1961	$(X/C)_f$	(X/C) location of wing trailing edge flap hinge line. This location should occur at a vortex station.
1965	δ_K	Deflection of wing leading edge flap in radians.

1966	$(X/C)_K$	(X/C) location of wing leading edge flap hinge line. This location should occur at a vortex station.
1970-1984	1, 2, $\cdots$	Wing twist table. Wing twist in radians is input for the stations listed in loc. 731-745.
1985-1999	$\Delta ZLE_1, \Delta ZLE_2, \cdots$	Delta Z of wing leading edge table. ZLE is input for η stations listed in loc. loc 731-745.
2000	NXC	Number of X/C stations where wing thickness is described. The maximum number is 23. CODIM is used to interpolate between input data in the chordwise direction to obtain the thickness and slope of thickness at the source lines.
2001-2024	$(X/C)_1, (X/C)_2, \cdots$	List of (X/C) stations where wing thickness is described.
2025	NETA	Number of η stations where wing thickness is described. The maximum number is 23. Straight line interpolation is used in the spanwise direction to obtain the thickness and slope of the thickness at the source lines.
2026-2049	$\eta_1, \eta_2, \cdots$	List of η stations where wing thickness is described.
2050-2499	$(Z/C)_1, (Z/C)_2, \cdots$	Wing thickness table. The wing thickness is input as Z/C at the first station listed at loc. 2026 and all the X/C stations listed at loc. 2001-2024. The Z/C is then input for the second span station, etc.
2490	ATP	1.0 if pylon attaches to fuselage. 2.0 if pylon attaches to fanpod. 3.0 if pylon attaches to wing.
2491	PITH	This location specifies the number of the fuselage or fanpod meridian line to which the pylon attaches. The meridian lines are numbered in ascending order in the clockwise direction when looking from rear to front.

2492	PDA	Pylon dihedral angle in degrees. The angle is measured from the Z-axis in the clockwise direction when looking from rear to front.
2500	PHT	Pylon height measured from the fanpod X-Y plane to the top of the nacelle, including the pylon extension through the nacelle if used.
2501	PØX	X location of pylon apex.
2502	PØY	Y location of starboard pylon apex.
2503	PØZ	Z location of pylon apex. Set equal to zero.
2504	PAA	Pylon angle of attack in degrees. This angle is positive in the counter clockwise direction if the right hand rule is used about the positive Z axis.
2505	PAAI	Pylon local angle of attack indicator. Input 0.0 if pylon is flat. Input 1.0 if pylon has twist or camber.
2506	PADI	Pylon camber table indicator. Use 0.0 if camber is input as local angles of attack in radians. Use 1.0 if camber is input as Z/C.
2507	PNVC	Number of vortices and source lines in the chordwise direction on the pylon. The maximum number is 19.
2508	PNVS	Number of vortices and source line segments in the spanwise direction on the pylon. The maximum number is 20.
2509	PUN	Number of chordwise assumed vorticity functions on the pylon. Similar to loc. 1280 for the wing. Maximum number is 10.

2510	PSJC	The value of the first control point in the chordwise direction on the pylon to be used in the lifting solution. This input is similar to that for the wing at loc. 14. A more complete description of the input is given there.
2511	PNJC	Number of control points in the chordwise direction on the pylon. The maximum number is 18.
2512-2529	$PJC_1, PJC_2, \cdots$	List of pylon control points in the chordwise direction. This input is similar to loc. 1660-1689 for the wing.
2530	PNX	Number of S/C stations on the pylon at which the pylon camber is described. The maximum number is 19.
2531-2549	$PXC_1, PXC_2, \cdots$	List of X/C stations on the pylon at which the pylon camber is described. The pylon camber is described at loc. 2570-2939.
2550	PNE	Number of η stations on the pylon at which the pylon camber and twist are described. The maximum number is 19.
2551-2569	$\eta_1, \eta_2, \cdots$	The list of η stations on the pylon at which the pylon camber and twist are described. The twist is described at loc. 2940-2959.
2570-2939	$PAD_1, PAD_2, \cdots$	Pylon camber table. The camber is input in terms of local angle of attack in radians or in terms of Z/C at the X/C and η stations given at loc. 2531-2549 and loc. 2551-2569, respectively. The camber is input for the first η station and all of the (X/C)'s, then the second η station and all of the (X/C)'s, etc. This table is unnecessary if wing is flat.
2940-2959	$\varepsilon_1, \varepsilon_2, \cdots$	Pylon twist table. The pylon twist is given in radians at the η stations listed at loc. 2551-2569.

2960-2965	η, X_{LE}, X_{TE}	Pylon planform table. The pylon planform is input in the same way as the outboard panel of the wing. Refer to input loc. 1210-1239. The X distances of the leading and trailing edges are relative to PØX. The pylon η's are relative to PØZ and equal to Y_P/P_{HT}.
3000	NXC	Number of X/C stations on the pylon where the pylon thickness is defined. The maximum number is 19.
3001-3019	$(X/C)_1, (X/C)_2, \cdots$	List of X/C stations on the pylon where the pylon thickness is defined.
3020	NETA	Number of η stations on the pylon where the pylon thickness is defined. The maximum number is 18.
3021-3039	$\eta_1, \eta_2, \cdots$	List of η stations on the pylon where the pylon thickness is defined.
3040-3369	$(Z/C)_1, (Z/C)_2, \cdots$	Pylon thickness table. The pylon thickness is input as Z/C at the X/C stations listed at loc. 3001-3019 and the stations listed at loc. 3021-3039. The Z/C's are input at the first η station and all of the X/C stations, then the second η, and etc. The Z/C's at the source lines are obtained from these inputs by straight line interpolation in the spanwise direction and CODIM in the chordwise direction.
3380	XLFX	Fanpod X direction area of influence. This input specifies the X range of influence of a vortex in terms of its grid length. The vortex influence is computed for those control points within this range. Those control points outside of this range will be assigned zero influence from the vortex.
3381	XLFY	Fanpod Y-Z direction area of influence. This input specifies the range of influence, in planes perpendicular to the X axis of a fanpod vortex in terms of its grid width. The vortex influence is computed for those control points within this range. Those control points outside of this range will be assigned zero influence from the vortex.

3381	XLW	Wing Y-Z direction area of influence. This input specifies the range of influence, in planes perpendicular to the X axis, of a wing vortex in terms of its grid width. The vortex influence is computed for those control points within this range. Those control points outside of this range will be assigned zero influence from the vortex.
3382	XLP	Pylon Y-Z direction area of influence. This input specifies the fange of influence, in planes perpendicular to the X axis, of a pylon vortex in terms of its grid width. The vortex influence is computed for those control points within this range. Those control points outside of this range will be assigned zero influence from the vortex.
3383	XLNC	Nacelle X direction area of influence. This input specifies the X range of influence of a source frustum in terms of the frustum's average diameter. The frustum's influence is computed for those control points within this range. Those control points outside of this range will be assigned zero influence from the frustum.
3385	XLNR	Nacelle Y_N direction area of influence. This input specifies the range of influence, in planes, perpendicular to the X axis, of a source frustum in terms of the frustum's average diameter. The frustum's influence is computed for those control points within this range. Those control points outside of this range will be assigned zero influence from the frustum.
3386	XLBX	Fuselage X direction area of influence. This input specifies the X range of influence of a vortex in terms of its grid length. The vortex influence is computed for those control points within this range. Those control points outside of this range will be assigned zero influence from the vortex.
3387	XLBY	Fuselage Y-Z direction area of influence. This input specifies the range of influence, in planes perpendicular to the X axis of a fuselage vortex in terms of its grid width. The vortex influence is computed for those control points within this range. Those control points outside of this range will be assigned zero influence from the vortex.

3390	NB	Number of bodies in the nacelle representation. This input is either one or two.
3391	XNØ	The X location of the nacelle lip.
3392	YNØ	The Y location of the starboard nacelle X_N-Y_N plane.
3393	ZNØ	The Z location of the axis of rotation of the nacelle.
3394	XNP1	Number of points used to define the nacelle contour. Maximum number is XNP1=140-XNP2.
3395	XNP_2	Number of points used to define second nacelle body contour. Maximum number is XNP2=140-XNP1.
3400-3549	$XB1_1, XB1_2, \cdots$	List of X_N components of the points describing the nacelle contour. These points are input from the trailing edge down around the internal surface and then over the external surface in a clockwise direction as shown in the sketch in the input format description.
3550-3699	$YB1_1, YB1_2, \cdots$	List of Y_N components of the points describing the nacelle contour. These points are input in the same sequence as the X_N points. All of these inputs are positive.
3700-3849	$XB2_1, XB2_2, \cdots$	List of X_N components of the points describing the second nacelle body contour. These points are input from the nose to the tail end of the body as shown in the sketch in the input format description.
3850-3999	$YB2_1, YB2_2, \cdots$	List of Y_N components of the points describing the second nacelle body contour. These points are input in the same sequence as the X_N points for this body. All of these inputs are positive.
4000	IDB	Fuselage indicator. If 0.0, component does not exist. If 1.0, calculate influence matrix A. If 2.0, use component influence matrix A from prior case.

4001	IDF	Fanpod indicator. Same specification as above.
4002	IDW	Wing indicator. Same specification as above.
4003	IDP	Pylon indicator. Same specification as above.
4004	IDN	Nacelle indicator. Same specification as above.
4005	BCD	Fuselage chord.
4006	BAA	Fuselage angle of attack in degrees
4007	BMFI	This input indicates if the multiplication factors listed in loc. 4600-4634 are the same as those listed in loc. 4635-4669. If they are, loc. 4635-4669 may be omitted.
4008	BCPI	This input indicates if the fuselage cross sections described in loc. 4090-4564 are to be placed ⊥ to the camber line described in loc. 4670-4704 or ⊥ to the fuselage X axis. Use 0.0 if sections are to be placed ⊥ to the camber line and 1.0 if they are to be placed to the X axis.
4009	BLØI	This input indicates the type of vortex grid input to be used to describe the fuselage vortex grid in the chordwise direction. If the grid is to be placed at equal increments of X, input -1.0. No input is needed in loc. 4735-4884 if 0.0 or -1.0 is input here. If the grid is to be placed at X stations other than at equal increments of Ø or X, input a 1.0 here and list the X stations in loc. 4735-4884.
4010	BTHI	This input indicates the type of fuselage vortex grid to be used to describe the lateral vortex grid distribution. If the grid is to be placed at equal roll angles Θ, input 0.0 here and no input is required in loc. 4885-4904. If a lateral vortex grid other than equal roll angles is desired, input 1.0 and list the values in loc. 4885-4904.

4015	BQX	X location of the nose of the starboard fuselage.
4016	BQY	Y location of the nose of the starboard fuselage.
4017	BQZ	Z location of the nose of the starboard fuselage.
4020	BNXS	Input the number of chordwise stations where the fuselage cross sections are to be described. The maximum number is 29.
4021-4049	$XS_1, XS_2, \cdots$	List of fuselage chord stations where the fuselage cross sections are to be described.
4050	BNTY	Input the number of lateral fuselage stations where the fuselage cross sections are to be described. The maximum number is 36.
4051-4086	$\theta_1, \theta_2, \cdots$	List of fuselage lateral stations where the fuselage cross sections are defined. The list of θ's are input from 0.0 to 180.0 degrees.
4090-4564	$R_1, R_2, \cdots$	List of radii for fuselage at XS and locations. Input radii for first XS station and all of the θ's from 0.0 to 180.0 degrees, then the second XS station and all the θ's again. Continue this process for all of the XS stations.
4565	BNXM	Input the number of chordwise fuselage stations where the YM and ZM multiplication factors will be applied and where the fuselage camber is defined. The body described by the radii in loc. 4090-4564 will be multiplied by the YM and ZM multiplication factors in the Y_F and Z_F directions, respectively. The fuselage camber will then be added to obtain the final description of the fuselage. It is not necessary to use the multiplication factors as a means of describing the fuselage. However, if the fuselage cross sections can be represented by a series of ellipses, the YM and ZM distributions can be used to input the major and minor axses of the elliptical cross sections. For this case only four radii, equal to unity, need be input. The maximum number of chordwise station is 34.

4566-4599	$XMC_1, XMC_2, \cdots$	List of chordwise stations where the fuselage multiplication factors and camber are defined.
4600-4634	$YMC_1, YMC_2, \cdots$	List of fuselage multiplication factors. The Y components of the radii defined in loc. 4090-4564 will be multiplied by these factors
4670-4704	$BZC_1, BZC_2, \cdots$	Distribution of fuselage camber. This camber will be applied to the fuselage after the radii defined in loc. 4090-4564 are multiplied by the multiplication factors defined in loc. 4600-4634 and loc. 4635-4669. If the radii as given in loc. 4090-4564 are for the actual fuselage and the use of multiplication factors or camber is not necessary, set BNXM=1.0, YM_1=1.0, ZM_1=1.0, and BZC_1=0.0.
4705	BNVX	This is the number of vortex grid lines in the longitudinal direction on the fuselage. There is one more line than the number of vortices in the longitudinal direction. The nose and tail end of the fuselage must be included. The maximum number is 150. The list of X grid stations is given in loc. 4735-4884.
4706	BNVY	This is the number of lateral vortex grid lines around the fuselage. This number should be kept as small as possible, 8 to 12, in order to keep computing time down. The list of θ angles for the lateral grid lines is given in loc. 4885-4904. The maximum number is 20.
4707	BNDV	This is the number of divisions a circumferential fuselage vortex line segment is divided into between adjacent lateral vortex grid lines. The number is an odd integer. Usually equal to 3 to 7. These divisions permit the fuselage vortex grid to better map the actual contour.
4708	BNJX	This is the number of control points in the longitudinal direction along a meridian line on the fuselage. The list of control points is given in loc. 4095-4979. Usually BNJX 2 BNF. The maximum number is 75.

4709	BNF	The number of chordwise vorticity functions used along each meridian line. There are standard functions listed in the input format description at loc 4710. Also, linearly varying functions over segments of the fuselage can be used. These have been shown to improve the numerical stability in cases run in the LAAD wing-body program. The maximum number of fuselage longitudinal functions is 25.
4710-4734	$F_1, F_2, \cdots$	This is the list of longitudinal functions used on the fuselage. The functions available are designated as follows:

F for Standard Functions		F for Linear Functions
1.0	COTØ/	$(x/c)_1$
2.0	COT(90-Ø/2)	$(x/c)_2$
3.0	SINØ	$(x/c)_3$
4.0	COSØ	$(x/c)_4$
5.0	SIN2Ø	$(x/c)_5$
6.0	COS2Ø	$(x/c)_6$
.	.	.
.	.	.
.	.	.

4735-4884	$BVX_1, BVX_2, \cdots$	This is the list of X stations for the fuselage vortex grid. The list must include BVX=0.0 and BVX_N=BCD. The maximum number is 20.
4885-4904	$BVT_1, BVT_2, \cdots$	This is the list of lateral vortex grid stations. This list of θ's refer to the fuselage body before the multiplication factors and camber are applied. The θ's are input from 0.0 to sum value less than 180 degrees. The first must be zero. The maximum number is 20.
4905-4979	$BJX_1, BJX_2, \cdots$	This is the list of control points on the fuselage in the longitudinal direction. The control point is designated by the number of the vortex aft of the fuselage nose on which the control point is placed. The maximum number is 75. However, the number cannot exceed BNVX-1.

ADDITIONAL DATA FOR SKIN FRICTION DRAG

LOCATION	SYMBOL	DESCRIPTION
4980	TFS	Freestream static temperature in degrees RANKINE. If TFS is set to 0.0, the skin friction drag calculation is bypassed.
4981	PFS	Freestream static pressure in $(\#/FT^2)$
4982	CK	Airfoil thickness correction. Usually CK=2.0 for airfoils with maximum thickness at 30 percent chord and CK=1.2 for NACA 64 and 65 series airfoils.
4983	TRID	Transition from laminar to turbulent point indicator. If -1.0, transition points in loc. 4990-4994 are input. If 0.0, the natural flat plate transition points are calculated.
4984	KS	Equivalent sand grain height in feet.
4985	TAU	Turbulence intensity. If TRID=-1.0, this input is not required.
4990-4994	$XTRL_1, XTRL_2 \cdots$	The x transition point over length (x/c) for the fuselage, fanpod, wing, pylon and nacelle respectively. If TRID=0.0, these data are not required.
4995-4999	$TCM_1, TCM_2, \cdots$	The maximum thickness over length (t/c) for the fuselage, fanpod, wing, pylon and nacelle respectively.

SAMPLE INPUT DATA SHEETS

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

FANPOD PROGRAM
BODY-FANPOD-WING-PYLON-NACELLE

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	1		
13			AR ~ WING ASPECT RATIO
25			SPAN ~ WING SPAN
37			MAC ~ MEAN AERODYNAMIC CHORD
49		73 80	MACH ~ MACH NO.
61			
1	5		
13			FCD ~ FANPOD CHORD
25			FAA ~ FANPOD ANGLE OF ATTACK (DEG.)
37			XCG ~ Xc.g. FOR MOMENT
49		73 80	WAA ~ WING ANGLE OF ATTACK (DEG.)
61			WAAI ~ { 0.0 WING IS FLAT 1.0 WING HAS CAMBER OR TWIST.
	10		
13			WADI ~ WING CAMBER TABLE INDICATOR { 0.0 IF LOCAL ANGLE OF ATTACK IN RADIANS, 1.0 IF DEFLECTIONS IN Z/c.
25			NOT USED
37			PRII ~ PRINT INDICATOR { 0.0 - PRINT MATRIX B. 1.0 PRINT MATRIX B AND LEAST SQ. B.
49		73 80	NOT USED
61			WSJC ~ FIRST WING CONTROL POINT IN CHORDWISE DIRECTION TO BE USED IN LIFTING SOLUTION, J VALUE FROM 1660-1689.
1	15		
13			FMFI ~ FANPOD MULTI. FACTOR INDICATOR { 0.0 $Y_M = Z_M$ (Z CAMBER), -2.0 $Y_M = Z_M$ (Y CAMBER) 1.0 $Y_M \neq Z_M$ (" "), -1.0 $Y_M \neq Z_M$ (" ")
25			FCPI ~ FANPOD CAMBER INDICATOR { 0.0 FANPOD SECTION PUT ⊥ TO CAMBER LINE, 1.0 " " " ⊥ " X_F AXIS.
37			FLØI ~ FANPOD CHORDWISE VORTEX GRID INDI. { 0.0 GRID AT EQUAL $\Delta\phi$ WHERE $\phi = \cos^{-1}(1-2x/c)$ 1.0 GIVEN IN LOC. 1460-1609. -1.0 EVEN $\Delta x/c$
49		73 80	FTHI ~ FANPOD LATERAL VORTEX GRID INDI. { 0.0 GRID AT EQUAL θ. 1.0 GIVEN θ AT LOC. 1610-1659.
61			PVPC ~ PNVS MINUS NUMBER OF SPAN STATIONS WHERE PRESSURES ARE TO BE COMPUTED ON THE PYLON.

FORM 114-C-17 REV. 7-58

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

FANPOD PROGRAM
BODY-FANPOD-WING-PYLON-NACELLE

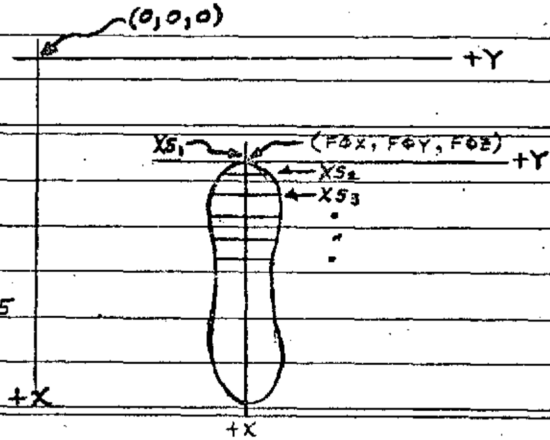
	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	1		
13			AR ~ WING ASPECT RATIO
25			SPAN ~ WING SPAN
37			MAC ~ MEAN AERODYNAMIC CHORD
49		73 80	MACH ~ MACH NO.
61			
1	5		
13			FCD ~ FANPOD CHORD
25			FAA ~ FANPOD ANGLE OF ATTACK (DEG.)
37			XCG ~ Xc.g. FOR MOMENT
49		73 80	WAA ~ WING ANGLE OF ATTACK (DEG.)
61			WAAI ~ { 0.0 WING IS FLAT; 1.0 WING HAS CAMBER OR TWIST.
	10		
13			WADI ~ WING CAMBER TABLE INDICATOR { 0.0 IF LOCAL ANGLE OF ATTACK IN RADIANS. 1.0 IF DEFLECTIONS IN Z/C.
25			NOT USED
37			PRII ~ PRINT INDICATOR { 0.0 - PRINT MATRIX B. 1.0 PRINT MATRIX B AND LEAST SQ. B.
49		73 80	NOT USED
61			WSJC ~ FIRST WING CONTROL POINT IN CHORDWISE DIRECTION TO BE USED IN LIFTING SOLUTION, J VALUE FROM 1660 - 1689.
1	15		
13			FMFI ~ FANPOD MULTI. FACTOR INDICATOR { 0.0 $Y_M = Z_M$ (Z CAMBER). -2.0 $Y_M = Z_M$ (Y CAMBER) 1.0 $Y_M \neq Z_M$ (" "). -1.0 $Y_M \neq Z_M$ (" ")
25			FCPI ~ FANPOD CAMBER INDICATOR { 0.0 FANPOD SECTION PUT ⊥ TO CAMBER LINE. 1.0 " " " ⊥ " X_F AXIS.
37			FLOI ~ FANPOD CHORDWISE VORTEX GRID INDI. { 0.0 GRID AT EQUAL $\Delta\phi$ WHERE $\phi = \cos^{-1}(1-2X/C)$ 1.0 GIVEN IN LOC. 1460-1609. -1.0 EVEN $\Delta X/C$
49		73 80	FTHI ~ FANPOD LATERAL VORTEX GRID INDI { 0.0 GRID AT EQUAL θ. 1.0 GIVEN θ AT LOC. 1610 - 1659.
61			PVPC ~ PNVS MINUS NUMBER OF SPAN STATIONS WHERE PRESSURES ARE TO BE COMPUTED ON THE PYLON.

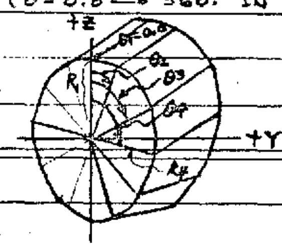
FORM 114-C-17 REV. 7-58

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ___ of ___ JOB NO. ______

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
20		FANPOD COORDINATE TABLE
		FNXS ~ No. of X (cross-section) in list below. (Max. 29)
		XS_1 $XS_1 < XS_2 < XS_3 \cdots$ (Relative to FANPOD origin.)
		XS_2
		XS_3
		.
25		
		.
		.
		NOTE: FNXS*FNTY < 475
50		
		FNTY ~ No. of θ at each cross-section in list below. (Max. 36)
		θ_1 $\theta_1 < \theta_2 < \theta_3 \cdots$ (θ = 0.0 → 360. in Deg.)
		θ_2
		θ_3
		:
87		
		FΦX ~ FANPOD origin X
		FΦY ~ " " Y } Relative to origin (0,0,0)
		FΦZ ~ " " Z





FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH	
1	90			FANPOD COORDINATE TABLE (CON'T)
13			R_1	LIST OF RADIUS
25			R_2	
37			R_3	
49		73 80	.	FOR XS_1 AT EACH θ
61			.	
1	− 95			
13			.	
25			.	
37			R_{NTY+1}	NOTE: NTY = NO. OF θ IN LOC. 50.
49		73 80	R_{NTY+2}	
61			R_{NTY+3}	
1	100			
13			.	FOR XS_2 AT EACH θ
25			.	
37			.	
49		73 80	.	
61			$R_{NTY+NTY+1}$	
1	105			
13			$R_{NTY+NTY+2}$	
25			.	FOR XS_3 AT EACH θ
37			.	
49		73 80	.	
61			.	

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO.________ PROGRAMMER____________ DATE________ PAGE_____of_____ JOB NO.________

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
565		FANPOD MULTIPLICATION FACTOR AND CAMBER TABLE
		FNXM ~ NO. OF X STATIONS IN LIST BELOW. (MAX 34)
		XMC_1 LIST OF X STATIONS WHERE MULTIPLICATION
		XMC_2 FACTORS AND CAMBERS ARE GIVEN.
	73 80	XMC_3 $XMC_1 < XMC_2 < XMC_3 < \ldots$
		.
-570		
		. NOTE:
		. IF FANPOD HAS NO CAMBER AND THE MULT. FACTORS ARE NOT REQ'D., SET FNXM=0.0
		XMC_{NXM}
	73 80	
600		
		YM_1 LIST OF MULT. FACTORS YM
		YM_2
		YM_3
	73 80	.
		.
605		
		.
		.
		YM_{NXM}
	73 80	

43

114-C-17 VELLUM

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
6.35	73 80	FANPOD MULT. FACTOR AND CAMBER TABLE (CON'T.)
		ZM_1 LIST OF MULT. FACTORS ZM
		ZM_2
		ZM_3 NOTE 1
		• LIST OF ZM IS NOT REQUIRED, IF $Y_M = Z_M$ SET FMFI=0.0 (LOC. 15)
		•
-640	73 80	
		•
		•
		ZM_{NXM}
670	73 80	
		FZC_1 LIST OF FANPOD CAMBER (Z)
		FZC_2
		FZC_3
		•
		•
675	73 80	
		•
		•
		FZC_{NXM}

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ________ PROGRAMMER ____________ DATE ________ PAGE ____ of ____ JOB NO. ________

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	705		WING CAMBER TABLE
13			WNX ~ No. OF X/C IN LIST BELOW. (MAX. 23)
25			x/c_1 $x/c_1 < x/c_2 < x/c_3 \cdots$
37			x/c_2
49		73 80	x/c_3
61			.
1	710		
13			. NOTE:
25			. THE WING CAMBER TABLE IS NOT REQUIRED IF WAAI = 0.0 (LOC. 9)
37			x/c_{NX}
49		73 80	
61			
1	730		
13			WNE ~ No. OF η IN LIST BELOW. (MAX. 15)
25			η_1 $\eta = \frac{y}{b/2}$
37			η_2 $\eta_1 < \eta_2 < \eta_3 \cdots$
49		73 80	η_3
61			.
1	735		
13			. NOTE: THIS LIST OF η'S ARE REQ'D ALSO FOR TWIST
25			. IN LOC. 1970 → 1984 AND AZLE IN LOC. 1985 → 1999.
37			η_{NE}
49		73 80	
61			

FORTRAN FIXED 10 DIGIT DECIMAL DATA

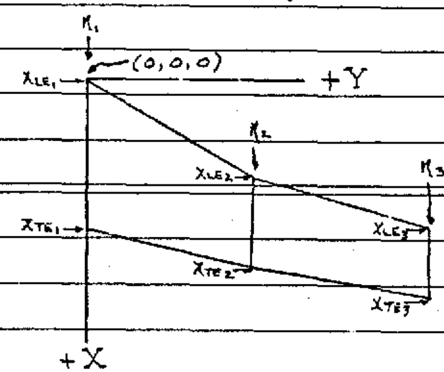
DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	750		WING CAMBER TABLE (CON'T.)
13			WAD1 — LIST OF;
25			WAD2 — 1.) IF WADI=0.0 (LOC. 10), ANGLE OF ATTACKS (α IN RAD.).
37			WAD3 — 2.) IF WADI=1.0 (LOC. 10), DEFLECTIONS (Z/C).
49		73 80	.
61			. — FOR η_1 AT EACH X/C
1	-755		
13			.
25			.
37			WADNX+1
49		73 80	WADNX+2
61			WADNX+3
1	760		
13			. — FOR η_2 AT EACH X/C
25			.
37			. — NOTE:
49		73 80	. — NX = NO. OF X/C IN LOC. 705
61			WADNX+NX+1
1	765		
13			.
25			. — FOR η_3 AT EACH X/C
37			.
49		73 80	
61			

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO.______ PROGRAMMER______ DATE______ PAGE____of____ JOB NO.______

Col	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	1200		
13			WΦX ~ WING ORIGIN X } RELATIVE TO AIRCRAFT ORIGIN (0,0,0)
25	0.0		WΦY ~ " " Y
37			WΦZ ~ " " Z
49		73 … 80	WITΦ ~ I^TH θ OF FANPOD AT WHICH OUTER WING IS ATTACHED.
61			WITI ~ " " " " " " INNER " " "
1	1205		
13			WNVSΦ ~ No. OF VORTICES SPANWISE FOR BOTH SIDES OF OUTER WING. IF WING ONLY (ODD INTEGER), OTHERWISE (EVEN INTEGER), (LIM. 11→50)
25			WNVSI ~ No. OF VORTICES SPANWISE FOR BOTH SIDES OF INNER WING, NO BODY (ODD INTEGER), WITH BODY (EVEN INTEGER), (LIM. 11→50)
37			WITB ~ I^TH θ OF BODY AT WHICH WING IS ATTACHED.
49		73 … 80	XNEΦ ~ No. OF η IN OUTER WING TABLE: (LOC. 1210 → 1239)
61			XNEI ~ " " " " INNER " " . (LOC. 1240 → 1269)
1	1210		WING PLANFORM TABLE
13			η_1 ΦUTER WING (WING ONLY)
25			X_{LE_1}
37			X_{TE_1}
49		73 … 80	η_2
61			X_{LE_2}
1	1215		
13			X_{TE_2}
25			η_3
37			X_{LE_3}
49		73 … 80	X_{TE_3}
61			

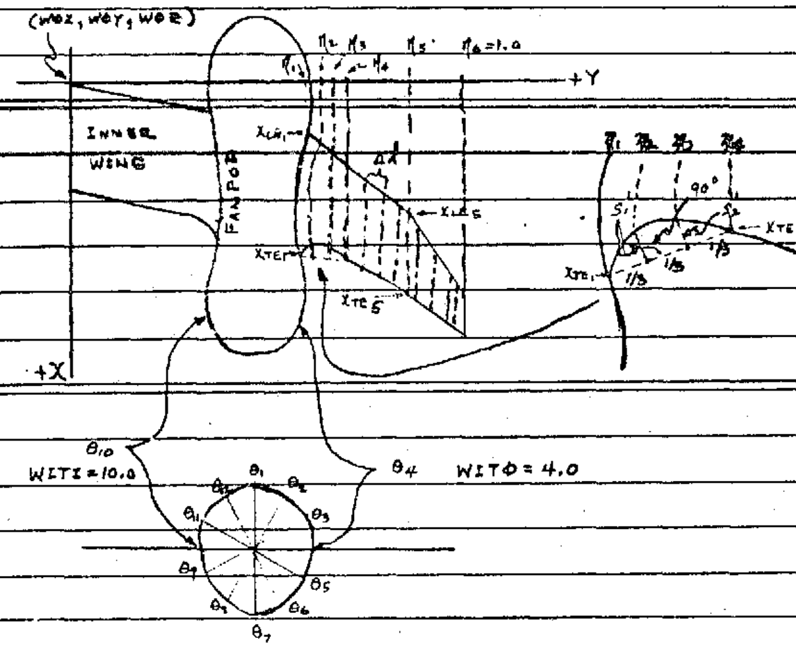


FORM 114-C-17 REV. 7-58

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

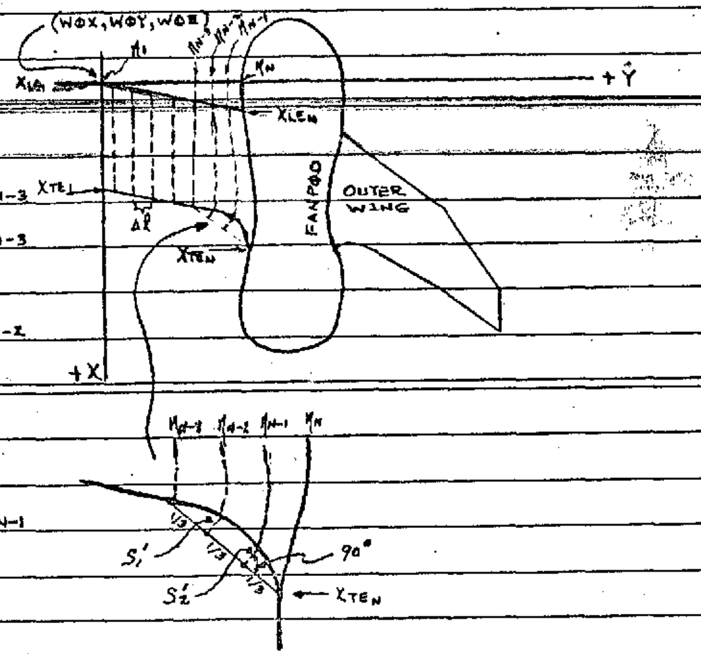
	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	1.2.1.0		WING PLANFORM TABLE
13			η_1 OUTER WING (FANPOD – WING)
25			X_{LE_1} NOTE:
37			X_{TE_1} 1. $\eta = \frac{Y}{b/2}$.
49		73 80	η_2 2. MAX. 10 η
61			X_{LE_2} 3. $(\eta_2-\eta_1)\approx(\eta_3-\eta_2)\approx(\eta_4-\eta_3)\approx\Delta l = \frac{1.-\eta_1}{.5*WNVS\Phi-.5}$
1	1.2.1.5		4. η, XLE, XTE RELATIVE TO ORIGIN (WΦX, WΦY, WΦZ)
13			S'_1
25			η_3
37			X_{LE_3}
49		73 80	S'_2
61			η_4
1	1220		
13			X_{LE_4}
25			X_{TE_4}
37			η_5
49		73 80	X_{LE_5}
61			X_{TE_5}
1	1225		
13			η_6
25			.
37			.
49		73 80	.
61			.



FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

Field	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	1240		WING PLANFORM TABLE (CON'T.)
13			η_1 INNER WING (FANPOD - WING)
25			X_{LE_1} NOTE:
37			X_{TE_1} 1. $\eta = \frac{Y}{b/2}$.
49		73 80	η_2 2. MAX. 10 η
61			X_{LE_2} 3. $(\eta_{N-2}-\eta_{N-3}) \approx (\eta_{N-1}-\eta_{N-2}) \approx (\eta_N-\eta_{N-1}) \approx \Delta\eta = \frac{\eta_N}{.5*WNVSI-.5}$
1	1245		4. η, X_{LE}, X_{TE} RELATIVE TO ORIGIN (WOX, WOY, WOZ)
13			X_{TE_2}
25			•
37			•
49		73 80	•
61			η_{N-3}
1	1250		
13			$X_{LE_{N-3}}$
25			$X_{TE_{N-3}}$
37			η_{N-2}
49		73 80	$X_{LE_{N-2}}$
61			S_1'
1	1255		
13			η_{N-1}
25			$X_{LE_{N-1}}$
37			S_2'
49		73 80	η_N
61			X_{LE_N}

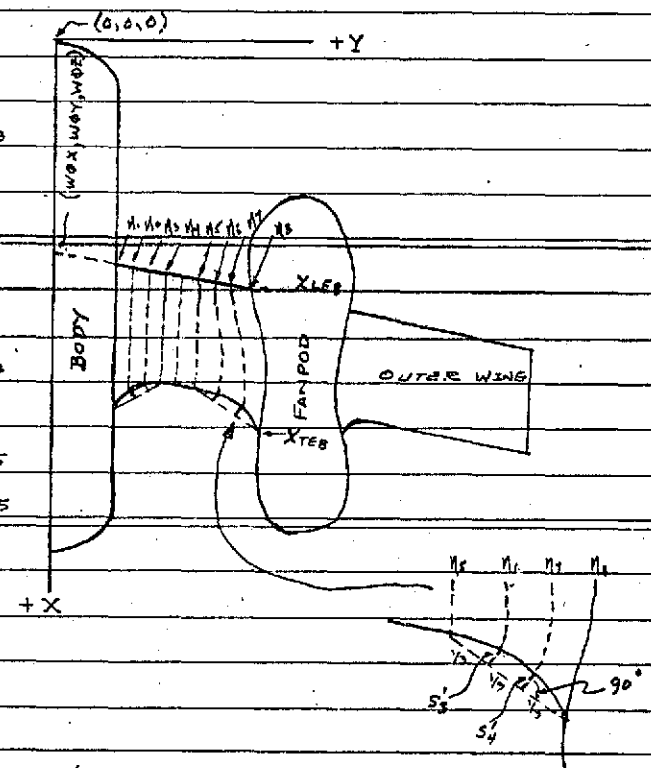


X_{TE}

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ________ PROGRAMMER ____________ DATE ________ PAGE _____ of _____ JOB NO. ________

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	1240		WING PLANFORM TABLE
13			η_1 INNER WING (BODY – FANPOD – WING)
25			X_{LE_1}
37			X_{TE_1} NOTE: OUTER WING IS THE SAME AS ILLUSTRATED
49		73 … 80	η_2 FANPOD – WING COMBINATION.
61			X_{LE_2}
1	-1245		
13			S'_1
25			η_3
37			X_{LE_3}
49		73 … 80	S'_2
61			η_4
1	1250		
13			X_{LE_4}
25			X_{TE_4}
37			η_5
49		73 … 80	X_{LE_5}
61			X_{TE_5}
1	1255		
13			η_6
25			X_{LE_6}
37			S'_3
49		73 … 80	η_7
61			X_{LE_7}, S'_4, η_8, X_{TE_8}, X_{LE_8}



FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO.________ PROGRAMMER ______________ DATE ________ PAGE ____ of ____ JOB NO. ________

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	1270		
13			WNVC ~ No. of vortices chordwise on the wing. (MAX 40)
25			Not used
37			WNPC ~ No. of Cp calc. per chord of wing. (MAX. 30) List of x/c at Loc. 1870.
49		73 80	WNPS ~ * No. of Cp calc. per semi-span of wing. (MAX. 30) List of η at Loc. 1900.
61			FNVX ~ No. of X stations in longit. direction on fanpod. List of X at Loc 1460, if FLΦI = 1.0 (MAX. 100)
1	1275		
13			FNVY ~ No. of vortices in lateral direction on fanpod. List of θ at Loc 1610, if FTHI = 1.0 (MAX. 40)
25			FNDV ~ No. of divisions per lateral vortex on fanpod. (odd integer)
37			WNJC ~ No. of control points per chord on the wing. List of J at Loc. 1660. (MAX. 30)
49		73 80	WNJS ~ * No. of control points for semi-span on the wing. List of J at Loc. 1690. (MAX. 30)
61			FNJX ~ No. of control points in longit. direction on fanpod. List of J at Loc. 1720. (MAX 75)
1	1280		
13			WNU ~ No. of chordwise assumed load shapes for the wing. (MAX. 10)
25			WNW ~ * No. of spanwise assumed load shapes for the wing. List of W at Loc. 1285. (MAX. 10)
37			FNF ~ No. of longitudinal assumed load shapes for fanpod. List of F at Loc. 1295. (MAX. 25)
49		73 80	Not used * If WNW = 0.0, the discrete vortices are used
61			Not used spanwise also WNPS and WNJS are not req'd
1	1285		List of spanwise assumed load shapes for wing
13			W_1 0.0
25			W_2 Standard η^w 2. 4.
37			W_3 6. 8.
49		73 80	• Special R-func 1930. 1935.
61			• Additinal data at specified loc. 1940. :

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

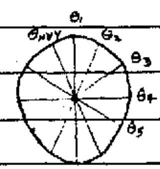
NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1295		LIST OF LONGITUDINAL ASSUMED LOAD SHAPES FOR FANPOD.
		F_1 F FOR TRIG. FUNC. F FOR LINEAR FUNC. (MAX. 25)
		F_2 1. ~ COT $\phi/2$ $(x/c)_1$
		F_3 2. ~ COT $(90 - \phi/2)$ $(x/c)_2$
		F_4 3. ~ SIN ϕ $(x/c)_3$
		F_5 4. ~ COS ϕ $(x/c)_4$
1300		5. ~ SIN 2ϕ $(x/c)_5$
		F_6 6. ~ COS 2ϕ $(x/c)_6$
		. . .
		. . .
		. . .
		.
1305		COMMENT ON x/c
		. 1. SET $(x/c)_1 = 0.0$
		. 2. SET FOLLOWING x/c BETWEEN
		. CONTROL POINTS (FJX'S).
		F_{NR} 3. LAST x/c NOT EQUAL TO 1.0
1310		

FORM 114-C-17 REV. 7-58

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO.________ PROGRAMMER ____________ DATE ________ PAGE ______ of ______ JOB NO. ________

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	1460		LIST OF X STATIONS IN LONGIT. DIRECTION ON FANPOD.
13			$FVX_1 = 0.0$ (MAX. 100)
25			FVX_2 LIST IS REQ'D. ONLY IF FLOX (LOC. 17) = 1.0
37			FVX_3 $FVX_1 < FVX_2 < FVX_3 < \cdots$
49		73 80	.
61			.
1	+465		
13			.
25			.
37			FVX_{NVX} = FCD (FANPOD CHORD)
49		73 80	
61			
1	1610		LIST OF LATERAL VORTICES ON FANPOD (MAX. 40)
13			$FVT_1 \sim \theta_1 = 0.0$ (θ IN DEGREES)
25			FVT_2 LIST IS REQ'D. ONLY IF FTHI (LOC. 18) = 1.0
37			FVT_3 $\theta_1 < \theta_2 < \theta_3 < \cdots$
49		73 80	.
61			.
1	1615		
13			.
25			.
37			$FVT_{NVY} \sim \theta_{NVY} \not> 360.$
49		73 80	
61			



FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	1660		LIST OF J (CONTROL POINTS) PER CHORD ON WING. (MAX. 30*)
13			WJC_1 START FROM LEADING EDGE.
25			WJC_2
37			:
49		73 80	:
61			WJC_{NJC} < WNVC (Loc. 1270)
1	+690		LIST OF J (CONTROL POINTS) FOR SEMI-SPAN ON WING. (MAX. 30*)
13			WJS_1 START FROM WING TIP
25			WJS_2
37			: * RESTRICTION : NJC * NJS ≤ 350
49		73 80	:
61			WJS_{NJS} < .5 * (WNVS + 1.)
1	1720		LIST OF J (CONTROL POINTS) IN LONGIT. DIRECTION ON FANPOD.
13			$FJX_1 \neq 1.$ (MAX. 75)
25			FJX_2 NOTE: DO NOT SET $FJX_1 = 1.0$
37			FJX_3
49		73 80	.
61			.
1	1725		
13			.
25			.
37			FJX_{NJX}
49		73 80	
61			

FORTRAN FIXED 10 DIGIT DECIMAL DATA

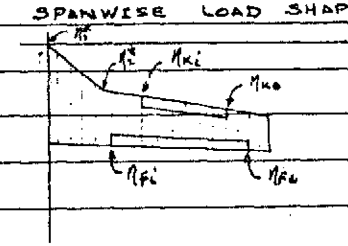
DECK NO.______ PROGRAMMER______ DATE______ PAGE____ of____ JOB NO.______

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1870		LIST OF X/c FOR Cp CALC. ON WING. (MAX. 30)
		$WPC_1 = (X/c)_1$
		WPC_2
		WPC_3
		•
		•
1875		
		•
		•
		WPC_{NPC}
1900		LIST of η (SPANWISE) FOR Cp CALC. ON WING. (MAX. 30)
		$WPS_1 = \eta_1$
		WPS_2 LIST IS NOT REQ'D., IF WNW=0.0 (DISCRETE)
		WPS_3
		•
		•
1905		
		•
		•
		WPS_{NPS}

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

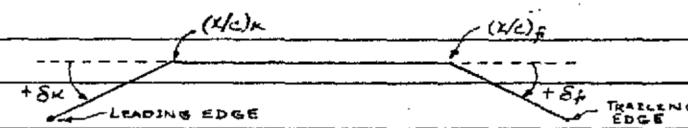
	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	1930		SPECIAL WING SPANWISE LOAD SHAPE #1.
13			TYPE INDI. ~ P-FUNC 0.0 FLAP −1.0 KRUEGER 1.0
25			η^* NOT USED NOT USED
37			RL ($\Delta\eta_i$) η_{Fi} η_{Ki}
49		73 80	RR ($\Delta\eta_o$) η_{Fo} η_{Ko}
61			NOT USED NOT USED NOT USED
1	1935		SPECIAL WING SPANWISE LOAD SHAPE #2.
13			TYPE INDI. ~
25			
37			
49		73 80	
61			
1	1940		COMMENTS:
13			1. THE P-FUNC η^* SHOULD BE AT A "CONTROL POINT" η.
25			2. THE FLAP AND KRUEGER, η_i AND η_o, SHOULD BE AT
37			A "TRAILING VORTEX" η.
49		73 80	3. ONLY ONE FLAP AND/OR KRUEGER ON WING.
61			4. THE P-FUNC, FLAP AND KRUEGER LOC. (1930, 1935...)
1	1945		MUST BE LISTED LAST, RESPECTIVELY, IN LIST OF W.
13			
25			
37			
49		73 80	
61			



FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO.______ PROGRAMMER______ DATE______ PAGE___ of___ JOB NO.______

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	1960		WING CHORDWISE LOAD SHAPE FOR FLAP,
13			δ_f ~ FLAP DEFLECTION ANGLE, (RADIAN)
25			$(x/c)_f$ ~ FLAP PIVOT x/c,
37			η_{fi}
49		73 80	η_{fo}
61			NOT USED
1	1965		WING CHORDWISE LOAD SHAPE FOR KRUEGER.
13			δ_K ~ KRUEGER DEFLECTION ANGLE. (RADIAN)
25			$(x/c)_K$ ~ KRUEGER PIVOT x/c.
37			η_{Ki} COMMENT:
49		73 80	η_{Ko} 1. INCREASE NO. OF CHORDWISE LOAD SHAPES (WNU) BY 1 OR 2 IF δ_f AND/OR δ_K ARE REQ'D.
61			NOT USED 2. $(x/c)_f$ AND $(x/c)_K$ SHOULD BE EQUAL TO A "BOUND VORTEX" x/c.
1	1970		*WING TWIST PER η IN LOC. 731 → 745.
13			ε_1 ~ TWIST AT η_1 (RADIAN)
25			ε_2 ~ " " η_2
37			ε_3 ~ " " η_3
49		73 80	•
61			•
1	1985		ΔZ OF WING LEADING EDGE PER η IN LOC. 731 → 745.
13			ΔZ_{LE_1} * NOTE:
25			ΔZ_{LE_2} TWISTS ARE ADDED TO WING ANGLE OF
37			ΔZ_{LE_3} ATTACK ONLY IF WAAI (LOC. 9) = 1.0
49		73 80	•
61			•



FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ___ of ___ JOB NO. ______

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	2000		WING THICKNESS TABLE (Z/C VS η) VS X/C
13			NXC ~ NO. OF X/C IN LIST BELOW. (MAX. 23)
25			$(X/C)_1 = 0.0$
37			$(X/C)_2$ $(X/C)_1 < (X/C)_2 < (X/C)_3 \cdots$
49		73 80	$(X/C)_3$ NOTE: THE THICKNESS CALCULATIONS
61			· ARE BYPASSD IF NXC = 0.0
1	2005		
13			·
25			·
37			$(X/C)_{NXC}$
49		73 80	
61			
1	2025		
13			NETA ~ NO. OF η IN LIST BELOW. (MAX. 19)
25			η_1
37			η_2 $\eta_1 < \eta_2 < \eta_3 \cdots$
49		73 80	η_3
61			·
1	2030		
13			·
25			·
37			η_{NETA}
49		73 80	
61			

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ________ PROGRAMMER ____________ DATE ________ PAGE ____ of ____ JOB NO. ________

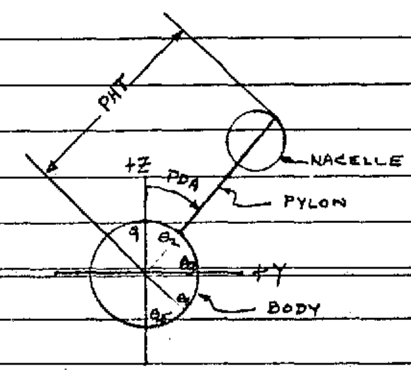
	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	2050		WING THICKNESS TABLE (CON'T.)
13			$(Z/c)_1$
25			$(Z/c)_2$
37			$(Z/c)_3$
49		73 80	. FOR η_1 AT EACH X/C
61			.
1	2055		
13			.
25			.
37			$(Z/c)_{NXC+1}$
49		73 80	$(Z/c)_{NXC+2}$
61			$(Z/c)_{NXC+3}$
1	2060		
13			. FOR η_2 AT EACH X/C
25			.
37			.
49		73 80	.
61			$(Z/c)_{2NXC+1}$
1	2065		
13			$(Z/c)_{2NXC+2}$
25			$(Z/c)_{2NXC+3}$ FOR η_3 AT EACH X/C
37			.
49		73 80	.
61			.

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ________ PROGRAMMER ______________ DATE ________ PAGE ____ of ____ JOB NO. __________

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	2490		
13			ATP ~ PYLON ATTACH TO { 1.0 ~ BODY; 2.0 ~ FANPOD; 3.0 ~ WING }
25			PITH ~ Ith θ OF BODY OR FANPOD AT WHICH PYLON IS ATTACHED,
37			PDA ~ PYLON DIHEDRAL ANGLE. (DEG.)
49		73 80	NOT USED
61			" "

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1			
13			
25			ILLUSTRATION:
37			ATP = 1.0
49		73 80	PITH = 2.0
61			PDA = 45.0
1			
13			
25			
37			NOTE:
49		73 80	IF PYLON IS ATTACHED TO
61			THE WING, ATTACH PYLON ON
1			A WING "TRAILING" VORTEX.
13			
25			
37			
49		73 80	
61			

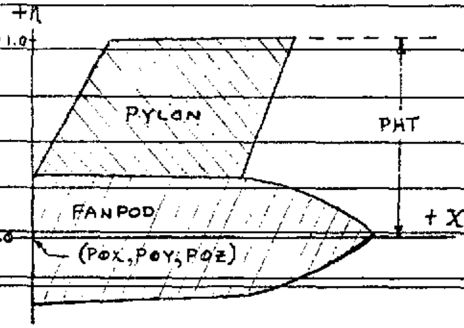


FORM 114-C-17 REV. 7-58

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ________ PROGRAMMER ________ DATE ________ PAGE ____ of ____ JOB NO. ________

Col.	NUMBER	IDENTIFICATION (73–80)	DESCRIPTION DO NOT KEY PUNCH
1	2500		PYLON DATA
13			PHT ~ PYLON HEIGHT
25			PΦX } PYLON ORIGIN
37			PΦY } RELATIVE TO (0,0,0)
49			PΦZ }
61			PAA ~ PYLON ANGLE OF ATTACK (DEG.)
1	2505		
13			PAAI ~ { 0.0 ~ PYLON IS FLAT. 1.0 ~ PYLON ANGLE OF ATTACK = PAA + CAMBER + TWIST
25			PADI ~ PYLON CAMBER TABLE (2530→2939) { 0.0 ~ ANGLE (α IN RADIAN) 1.0 ~ DEFLECTION (Z/C)
37			PNVC ~ No. OF VORTICES CHORDWISE ON PYLON. (MAX. 19)
49			PNVS ~ " " " SPANWISE " " (MAX. 20)
61			PUN ~ No. OF CHORDWISE ASSUMED LOAD SHAPES FOR PYLON. (MAX. 10)
1	2510		
13			PSJC ~ PYLON STARTING J CHORDWISE (NOT USED IN THICKNESS CALC.) SET EQUAL TO A J IN LIST BELOW. (APPROX. 7% OF CHORD)
25			PNJC ~ No. OF CONTROL POINTS PER CHORD ON PYLON IN LIST BELOW. (MAX. 18)
37			PJC_1 LIST OF PYLON J
49			PJC_2
61			PJC_3
1	2515		
13			PJC_4
25			.
37			.
49			PJC_{NJC}
61			



FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	2530		PYLON CAMBER TABLE
13			PNX ~ No. of x/c in list below. (Max. 19)
25			PXC1
37			PXC2 $x/c_1 < x/c_2 < x/c_3 \cdots$
49		73 80	PXC3
61			.
1	2535		
13			. Note:
25			. The pylon camber table is not required, if PAAI = 0.0 (Loc. 2505)
37			
49		73 80	
61			
1	2550		
13			PNE ~ No. of η in list below. (Max. 19)
25			PET1 = η_1 $\eta = \frac{"Y"}{b/2}$.
37			PET2 = η_2 $\eta_1 < \eta_2 < \eta_3 \cdots$
49		73 80	PET3 = η_3
61			.
1	2555		
13			.
25			.
37			
49		73 80	
61			

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ________ PROGRAMMER ________ DATE ________ PAGE ____ of ____ JOB NO. ________

Col.	NUMBER	IDENTIFICATION	DESCRIPTION	DO NOT KEY PUNCH
1	2570			PYLON CAMBER TABLE (CON'T)
13			PAD_1	LIST OF:
25			PAD_2	1.) IF PADI = 0.0 (LOC. 2506), ANGLE OF ATTACKS (α IN RAD.)
37			PAD_3	2.) IF PADI = 1.0 (LOC. 2506), DEFLECTIONS (Z/c)
49		73 80	.	
61			.	FOR η_1 AT EACH X/C
1	2575			
13			.	NOTE:
25			.	NK = NO. OF X/C IN LOC. 2530
37			PAD_{NX+1}	
49		73 80	PAD_{NX+2}	
61			PAD_{NX+3}	
1	2580			
13			.	FOR η_2 AT EACH X/C
25			.	
37			.	
49		73 80	.	
61			$PAD_{NK+NK+1}$	
1	2585			
13			$PAD_{NK+NK+2}$	
25			.	FOR η_3 AT EACH X/C
37			.	
49		73 80	.	
61			.	

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ________ PROGRAMMER ______________ DATE ________ PAGE ____ of ____ JOB NO. ________

Col.	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	2940		PYLON TWIST PER η IN LOC. 2551 → 2569
13			ε_1 ~ TWIST AT η_1
25			ε_2 ~ " " η_2
37			ε_3 ~ " " η_3
49		73 80	·
61			·
1	2945		
13			· NOTE:
25			· TWISTS ARE ADDED TO PYLON ANGLE OF
37			ε_{PNE} ATTACK ONLY IF PAAI (LOC. 2505) = 1.0
49		73 80	
61			
1	2960		PYLON PLANFORM TABLE
13			η_1
25			X_{LE_1} NOTE:
37			X_{TE_1} SEE WING PLANFORM TABLE (OUTER WING)
49		73 80	η_2 DISCRIPTIONS.
61			X_{LE_2}
1	2965		
13			S_1'
25			η_3
37			X_{LE_3}
49		73 80	S_2'
61			η_4

114-C-17 VELLUM

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ________ PROGRAMMER ________________ DATE ________ PAGE _____ of _____ JOB NO. ____________

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	3000		Pylon Thickness Table (Z/c vs η) vs X/c
13			NXC ~ No. of X/c in list below. (Max. 19)
25			$(X/c)_1 = 0.0$
37			$(X/c)_2$ $(X/c)_1 < (X/c)_2 < (X/c)_3 \cdots$
49		73 80	$(X/c)_3$ Note: The thickness calculations
61			. are bypass, if NXC = 0.0
1	3005		
13			.
25			.
37			$(X/c)_{NXC}$
49		73 80	
61			
1	3020		
13			NETA ~ No. of η in list below. (Max. 18)
25			η_1
37			η_2 $\eta_1 < \eta_2 < \eta_3 \cdots$
49		73 80	η_3
61			.
1	3025		
13			.
25			.
37			η_{NETA}
49		73 30	
61			

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ___ of ___ JOB NO. ______

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH	
1	3040		PYLON THICKNESS TABLE (CON'T.)	
13			$(z/c)_1$	
25			$(z/c)_2$	
37			$(z/c)_3$	
49		73 80	.	For η_1 at each X/c
61			.	
1	3045			
13			.	
25			.	
37			$(z/c)_{NXC+1}$	
49		73 80	$(z/c)_{NXC+2}$	
61			$(z/c)_{NXC+3}$	
1	3050			
13			.	For η_2 at each X/c
25			.	
37			.	
49		73 80	.	
61			$(z/c)_{2NXC+1}$	
1	3055			
13			$(z/c)_{2NXC+2}$	
25			$(z/c)_{2NXC+2}$	For η_3 at each X/c
37			.	
49		73 80	.	
61			.	

FORTRAN FIXED 10 DIGIT DECIMAL DATA

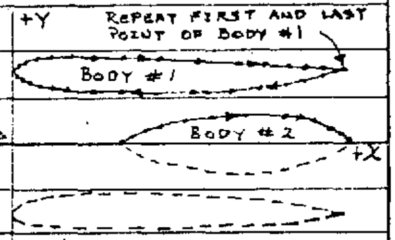
DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ___ of ___ JOB NO. ______

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	3380		* AREA OF INFLUENCE : ELIMINATE VORTICES BEYOND THE FOLLOWING LIMITS
13			XLFX ~ No. OF VORTICES IN THE X DIRECTION ON FANPOD.
25			XLFY ~ " " " " " Y " " FANPOD.
37			XLW ~ " " " " " Y " " WING.
49		73 80	XLP ~ " " " " " "Y" " " PYLON.
61			XLNC ~ " " " " " X " " NACELLE.
1	3385		
13			XLNR ~ " " " " " "Y" " " NACELLE.
25			XLBX ~ " " " " " X " " BODY.
37			XLBY ~ " " " " " Y " " BODY.
49		73 80	NOT USED * NOTE :
61			" " INITIALLY SET TO 1000000.0
1			
13			
25			
37			
49		73 80	
61			
1			
13			
25			
37			
49		73 80	
61			

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	3390		NACELLE DATA
13			NB ~ NO. OF BODIES
25			XNΦ } NACELLE ORIGIN
37			YNΦ } RELATIVE TO (0,0,0)
49		73 … 80	ZNΦ }
61			XNP1 ~ NO. OF COORD. ON BODY #1
1	3395		NOTE: $XNP1 + XNP2 \le 140$
13			XNP2 ~ NO. OF COORD. ON BODY #2
25			NOT USED NOTE:
37			" " 1. FOR BODY #1 READ COORD. FROM TRAILING EDGE→CLOCKWISE.
49		73 … 80	" " 2. " #2 " " " LEADING " "
61			" " 3. ALL OF THE Y COORD. MUST BE POSITIVE AND NO TWO CONSECUTIVE Y VALUES MAY BE ZEROS.
1	3400		X COORD. OF BODY #1 (NACELLE) (MAX 140)
13			$XB1_1$
25			$XB1_2$
37			⋮
49		73 … 80	⋮
61			⋮
1	3550		Y COORD. OF BODY #1 (NACELLE) (MAX 140)
13			$YB1_1$
25			$YB1_2$
37			⋮
49		73 … 80	
61			



FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	3700		X COORD. OF BODY #2 (NACELLE) (MAX. 140)
13			$XB2_1$
25			$XB2_2$
37			⋮
49		73 80	⋮
61			⋮
1	3850		Y COORD. OF BODY #2 (NACELLE) (MAX. 140)
13			$YB2_1$
25			$YB2_2$
37			⋮
49		73 80	⋮
61			⋮
1			
13			
25			
37			
49		73 80	
61			
1			
13			
25			
37			
49		73 80	
61			

FORTRAN FIXED 10 DIGIT DECIMAL DATA

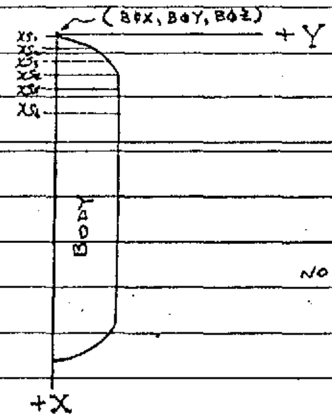
DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ___ of ___ JOB NO. ______

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	4000		
13			IDB ~ Body Indicator } 0.0 ~ Component does not exist.
25			IDF ~ Fanpod " } 1.0 ~ Calc. comp. infl. matrix A,
37			IDW ~ Wing " } 2.0 ~ Use comp. infl. matrix A from prior case.
49		73 80	IDP ~ Pylon " } Note: No 2.'s can follow 1.'s.
61			IDN ~ Nacelle " } Legal: 2., 2., 1., 1., 1. Illegal: 2., 2., 1., 2., 2.
1	4005		BØDY INPUT
13			BCD ~ Body chord
25			BAA ~ Body angle of attack (Deg.)
37			BMFI ~ Body Multi. Factor Indi. { 0.0 $Y_M = Z_M$ (Z camber). −2.0 $Y_M = Z_M$ (Y camber); 1.0 $Y_M \neq Z_M$ (" "). −1.0 $Y_M \neq Z_M$ (" ")
49		73 80	BCPI ~ Body Camber Indicator { 0.0 Body section put $\perp$ to camber line. 1.0 " " " $\perp$ " X_B axis.
61			BLØI ~ Body Chordwise Vortex Grid Indi. { 0.0 Grid at equal $\Delta\Phi$ where $\Phi = \cos^{-1}(1 - 2x/c)$; 1.0 Given in loc. 4735–4884. −1.0 even $\Delta x/c$
1	4010		
13			BTHI ~ Body Lateral Vortex Grid Indi. { 0.0 Grid at equal θ. 1.0 Given θ at loc. 4885–4904.
25			Not used
37			" "
49		73 80	" "
61			" "
1	4015		
13			BØX ~ Body origin X } Relative to origin (0,0,0)
25			BØY ~ " " Y } Relative to origin (0,0,0)
37			BØZ ~ " " Z } Relative to origin (0,0,0)
49		73 80	Not used
61			" "

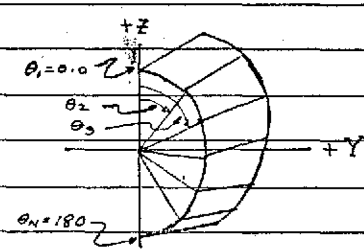
FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO.______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	4020		BODY COORDINATE TABLE
13			BNXS ~ NO. OF X CROSS-SECTIONS IN LIST BELOW. (MAX. 2[illegible].)
25			XS_1 $XS_1 < XS_2 < XS_3 \cdots$ RELATIVE TO BODY ORIGIN,
37			XS_2
49		73 80	XS_3
61			.
1	4025		
13			.
25			. NOTE:
37			
49		73 80	
61			



	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	4050		
13			BNTY ~ NO. OF θ AT EACH CROSS-SECTION IN LIST BELOW. (MAX. 3[illegible])
25			θ_1 $\theta_1 < \theta_2 < \theta_3 \cdots$ (θ = 0.0 ⟶ 180. IN DEG.)
37			θ_2
49		73 80	θ_3
61			.
1	4055		
13			.
25			.
37			
49		73 80	
61			



FORM 114-C-17 REV. 7-68

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO.________ PROGRAMMER________________ DATE________ PAGE______ of______ JOB NO.____________

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	4090		Body Coordinate Table (con't.)
13			R_1
25			R_2
37			R_3
49		73 … 80	. For X_{S_1} at each θ
61			.
1	-4095		
13			.
25			.
37			R_{NTY+1} Note: NTY = No. of θ in Loc. 50.
49		73 … 80	R_{NTY+2}
61			R_{NTY+3}
1	4100		
13			. For X_{S_2} at each θ
25			.
37			.
49		73 … 80	.
61			$R_{NTY+NTY+1}$
1	4105		
13			$R_{NTY+NTY+2}$
25			. For X_{S_3} at each θ
37			.
49		73 … 80	.
61			.

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO.________ PROGRAMMER ______________ DATE ________ PAGE ____ of ____ JOB NO. ________

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	4565		BODY MULTIPLICATION FACTOR AND CAMBER TABLE
13			BNXM ~ No. OF X STATIONS IN LIST BELOW. (MAX. 34)
25			XMC_1 LIST OF X STATIONS WHERE MULTIPLICATION
37			XMC_2 FACTORS AND CAMBERS ARE GIVEN.
49		73 80	XMC_3 $XMC_1 < XMC_2 < XMC_3$ . . . .
61			.
1	4570		
13			. NOTE :
25			. IF BODY HAS NO CAMBER AND THE MULTI.
37			XMC_{NXM} FACTORS ARE NOT REQ'D., SET BNXM = 0.0
49		73 80	
61			
1	4600		
13			YM_1 LIST OF MULTI. FACTORS YM
25			YM_2
37			YM_3
49		73 80	.
61			.
1	4605		
13			.
25			.
37			YM_{NXM}
49		73 80	
61			

FORM ... 17 ... 7-58

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	4635		BODY MULTI. FACTOR AND CAMBER TABLE (CON'T.)
13			ZM_1 LIST OF MULTI. FACTORS ZM
25			ZM_2
37			ZM_3 NOTE: LIST OF ZM IS NOT REQ'D, IF YM = ZM
49		73 80	. SET FMFI=0.0 IN LOC. 15
61			.
1	4640		
13			.
25			.
37			ZM_{NKM}
49		73 80	
61			
1	4670		
13			BZC_1 LIST OF BODY CAMBER Z
25			BZC_2
37			BZC_3
49		73 80	.
61			.
1	4675		
13			.
25			.
37			BZC_{NKM}
49		73 80	
61			

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ________ PROGRAMMER ________ DATE ________ PAGE ____ of ____ JOB NO. ________

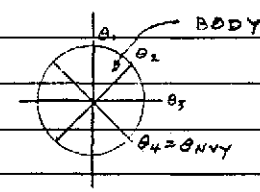
	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH		
1	.4705				
13			BNVX ~	No. of X stations in longit. direction on body. List of X at loc. 4735, if BLΦI (Loc. 4007) = 1.0	(Max. 150)
25			BNVY ~	No. of vortices in lateral direction on body. List of θ at loc. 4885, if BTHI (Loc. 4010) = 1.0	(Max. 20)
37			BNDV ~	No. of divisions per lateral vortex on body. (Odd integer)	
49		73 80	BNJX ~	No. of control points in longit. direction on body. List of J at loc. 4905.	(Max. 75)
61			BNF ~	No. of longit. assumed load shapes for body. List of F at loc. 4710.	(Max. 25)
1	-4710			List of longitudinal assumed load shapes for body.	
13			F_1	F for trig. func.	F for linear func.
25			F_2	1. ~ cot $\Phi/2$	$(x/c)_1$
37			F_3	2. ~ cot $(90-\Phi/2)$	$(x/c)_2$
49		73 80	F_4	3. ~ sin Φ	$(x/c)_3$
61			F_5	4. ~ cos Φ	$(x/c)_4$
1	4715			5. ~ sin 2Φ	$(x/c)_5$
13			F_6	6. ~ cos 2Φ	$(x/c)_6$
25			.	.	.
37			.	.	.
49		73 80	.	.	.
61			.		
1	4720				Comment on x/c
13			.		1. Set $(x/c)_1 = 0.0$
25			F_{NF}		2. Set following x/c between
37					control points (BJX's).
49		73 80			3. Last x/c not equal to 1.0.
61					

FORM 114-C-17 REV. 7-58

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	4735		LIST OF X STATIONS IN LONGIT. DIRECTION ON BODY.
13			$BVX_1 = 0.0$ (MAX. 150)
25			BVX_2 LIST IS REQ'D. ONLY IF BLBI(LOC. 4009)=1.0
37			BVX_3 $BVX_1 < BVX_2 < BVX_3 < \cdots$
49		73 80	.
61			.
1	-4740		
13			.
25			.
37			$BVX_{NVX} = BCD$ (BODY CHORD)
49		73 80	
61			
1	4885		LIST OF LATERAL VORTICES ON BODY. (MAX. 20)
13			$BVT_1 = \theta_1 = 0.0$ (θ IN DEGREES)
25			BVT_2 LIST IS REQ'D. ONLY IF BTHI (LOC. 4010)=1.0
37			BVT_3 $\theta_1 < \theta_2 < \theta_3 < \cdots$
49		73 80	.
61			.
1	4890		
13			.
25			.
37			$BVT_{NVY} \sim \theta_{NVY} \doteq 180.$
49		73 80	
61			



FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ___ of ___ JOB NO. ______

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	49.05		LIST OF J (CONTROL POINTS) IN LONGIT. DIRECTION ON BODY.
13			$BJX_1 \neq 1.0$ (MAX. 75)
25			BJX_2 NOTE: DO NOT SET $BJX_1 = 1.0$
37			BJX_3
49		73 80	.
61			.
1	4910		
13			.
25			.
37			BJX_{NJX}
49		73 80	
61			
1	49.80		SKIN FRICTION INPUT DATA
13			* T_∞ ~ FREE STREAM STATIC TEMPERATURE. (°R)
25			P_∞ ~ " " " PRESSURE. (#/FT²)
37			CK ~ THICKNESS CORRECTIONS (AIRFOILS) $K = 1 + CK(t/c) + 60(t/c)^4$; $C_D = C_F(S/S_{REF})K$
49		73 80	TRID ~ TRANSITION INDICATOR { -1.0 ~ TRANS. POINTS (XTRL) ARE INPUT. 0.0 ~ FLAT PLATE NATURAL TRANSITION.
61			KS ~ EQUIVALENT SAND GRAIN HEIGHT (FT.)
1	49.85		
13			T ~ TURBULENCE INTENSITY (NOT REQ'D. IF TRID = -1.0)
25			CONV ~ CONVERSION TO FEET. EXAMPLE: CONV = 12.0, IF CASE DATA ARE IN INCHES.
37			NOT USED * NOTE:
49		73 80	" " IF T_∞ IS SET TO 0.0, THE SKIN FRICTION
61			" " DRAG CALCULATION IS BYPASSED.

FORM 114-C-17 REV. 7-68

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ of ____ JOB NO. ______

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH	
1	4990		SKIN FRICTION INPUT DATA (CON'T.)	
13			XTRL1 ~ X TRANSTION / LENGTH FOR BODY.	Reqd. only if TRID = -1.0
25			XTRL2 ~ " " " " FANPOD.	
37			XTRL3 ~ " " " " WING.	
49		73 80	XTRL4 ~ " " " " PYLON.	
61			XTRL5 ~ " " " " NACELLE.	
1	4995			
13			TCM1 ~ MAX. THICKNESS/LENGTH FOR BODY.	
25			TCM2 ~ " " " " FANPOD.	If flat plate set TCM = 0.0
37			TCM3 ~ " " " " WING.	
49		73 80	TCM4 ~ " " " " PYLON.	
61			TCM5 ~ " " " " NACELLE.	
1	- 5000		LAST DATA CARD OF A CASE WITH A MINUS IN COL. 1.	
13				
25				
37				
49		73 80		
61				
1				
13				
25				
37				
49		73 80		
61				

SAMPLE INPUT DATA

```
A04NZXX,5,1400,60000.482511,RCST5786P FANPOD                          CAS10010
*NOTAPES,PSS                                                          CAS10020
LIBCOPY,A04RCS,LIB,OBJECT.                                            CAS10030
COPY,LIB/RX,LGO.                                                      CAS10040
RFL(135000,580000)                                                    CAS10050
LGO.                                                                  CAS10060
'                                                                     CAS10070
         1 4.0       24.0      6.125     0.6                          CAS10080
        10                                          3.0               CAS10090
      1200 0.0       0.0       0.0       0.0                          CAS10100
      1205 18.0                4.0       5.0                          CAS10110
      1210 0.1315    14.30     21.4060   0.2335     15.6              CAS10120
      1215 0.0       0.3355    16.9      0.0        0.4375            CAS10130
      1220 18.2      24.388    1.0       25.35      29.855            CAS10140
      1270 20.0      21.0      4.0                                    CAS10150
      1275                     11.0                                   CAS10160
      1280 5.0       0.0                                              CAS10170
      1660 1.0       2.0       3.0       5.0        7.0               CAS10180
      1665 9.0       11.0      13.0      15.0       17.0              CAS10190
      1670 19.0                                                       CAS10200
      1870 0.2       0.4       0.6       0.8                          CAS10210
      2000 20.0      0.0       0.005     0.0075     0.0125            CAS10220
      2005 0.025     0.05      0.1       0.15       0.20              CAS10230
      2010 0.25      0.30      0.35      0.4        0.45              CAS10240
      2015 0.5       0.55      0.6       0.7        0.8               CAS10250
      2020 1.0                                                        CAS10260
      2025 2.0       0.0       1.0                                    CAS10270
      2050 0.0       0.00464   0.00563   0.00718    0.00981           CAS10280
      2055 0.01313   0.01824   0.02194   0.02474    0.02687           CAS10290
      2060 0.02842   0.02945   0.02996   0.02992    0.02925           CAS10300
      2065 0.02793   0.02602   0.02087   0.01437    0.00013           CAS10310
      2070 0.0       0.00464   0.00563   0.00718    0.00981           CAS10320
      2075 0.01313   0.01824   0.02194   0.02474    0.02687           CAS10330
      2080 0.02842   0.02945   0.02996   0.02992    0.02925           CAS10340
      2085 0.02793   0.02602   0.02087   0.01437    0.00013           CAS10350
      4005 33.33     0.0       0.0       1.0        1.0               CAS10360
      4010 0.0                                                        CAS10370
      4015 0.0       0.0       0.0                                    CAS10380
      4565 23.0      0.0       0.2       0.3        0.5               CAS10390
      4570 1.0       2.0       3.0       4.0        6.0               CAS10400
      4575 8.0       10.0      12.0      14.0       16.0              CAS10410
      4580 18.0      20.0      22.0      24.0       26.0              CAS10420
      4585 28.0      30.0      32.0      33.33                        CAS10430
```

4050	2.0	0.0	180.0			CAS10440
4090	40.0	40.0	40.0	40.0		CAS10450
4020	2.0	0.0	33.33			CAS10460
4705	150.0	6.0	3.0	50.0	25.0	CAS10470
4710	0.0	0.00197	0.00789	0.01771	0.03142	CAS10480
4715	0.04894	0.07022	0.09517	0.12369	0.15567	CAS10490
4720	0.19098	0.22949	0.27103	0.31545	0.36258	CAS10500
4725	0.41221	0.46417	0.51825	0.57422	0.63187	CAS10510
4730	0.69098	0.75131	0.81262	0.87466	0.93721	CAS10520
4735	0.0	0.00731	0.01645	0.02924	0.04568	CAS10530
4740	0.06577	0.08951	0.11689	0.14792	0.18258	CAS10540
4745	0.22088	0.26281	0.30837	0.35755	0.41034	CAS10550
4750	0.46675	0.52676	0.59036	0.65756	0.72834	CAS10560
4755	0.80269	0.88061	0.96209	1.04712	1.13569	CAS10570
4760	1.22779	1.32341	1.42254	1.52517	1.63128	CAS10580
4765	1.74087	1.85393	1.97044	2.09039	2.21376	CAS10590
4770	2.34055	2.47073	2.60430	2.74123	2.88152	CAS10600
4775	3.02516	3.17211	3.32237	3.47592	3.63275	CAS10610
4780	3.79283	3.95615	4.12269	4.29244	4.46537	CAS10620
4785	4.64146	4.82070	5.00307	5.18854	5.37710	CAS10630
4790	5.56873	5.76339	5.96109	6.16178	6.36545	CAS10640
4795	6.57208	6.78164	6.99412	7.20948	7.42771	CAS10650
4800	7.64878	7.87266	8.09934	8.32878	8.56097	CAS10660
4805	8.79587	9.03346	9.27371	9.51661	9.76212	CAS10670
4810	10.01020	10.26085	10.51403	10.76971	11.02786	CAS10680
4815	11.28846	11.55147	11.81687	12.08464	12.35473	CAS10690
4820	12.62711	12.90177	13.17867	13.45779	13.73908	CAS10700
4825	14.02252	14.30808	14.59573	14.88543	15.17715	CAS10710
4830	15.47085	15.76650	16.06410	16.36360	16.66496	CAS10720
4835	16.96812	17.27310	17.57985	17.88829	18.19846	CAS10730
4840	18.51027	18.82368	19.13870	19.45529	19.77341	CAS10740
4845	20.09299	20.41403	20.73651	21.06033	21.38553	CAS10750
4850	21.71204	22.03981	22.36880	22.69902	23.03041	CAS10760
4855	23.36290	23.69650	24.03117	24.36684	24.70351	CAS10770
4860	25.04112	25.37961	25.71901	26.05920	26.40024	CAS10780
4865	26.74199	27.08450	27.42769	27.77150	28.11597	CAS10790
4870	28.46098	28.80653	29.15256	29.49907	29.84601	CAS10800
4875	30.19328	30.54095	30.88892	31.23711	31.58557	CAS10810
4880	31.93422	32.283	32.63193	32.98091	33.33	CAS10820
4905	3.0	6.0	9.0	12.0	15.0	CAS10830
4910	18.0	21.0	24.0	27.0	30.0	CAS10840
4915	33.0	36.0	39.0	42.0	45.0	CAS10850
4920	48.0	51.0	54.0	57.0	60.0	CAS10860

```
4925 63.0          66.0          69.0          72.0          75.0          CAS10870
4930 78.0          81.0          84.0          87.0          90.0          CAS10880
4935 93.0          96.0          99.0          102.0         105.0         CAS10890
4940 108.0         111.0         114.0         117.0         120.0         CAS10900
4945 123.0         126.0         129.0         132.0         135.0         CAS10910
4950 138.0         141.0         144.0         147.0         149.0         CAS10920
4600 0.0           0.00216       0.00298       0.00437       0.00722       CAS10930
4605 0.01205       0.01613       0.01971       0.02593       0.03090       CAS10940
4610 0.03465       0.03741       0.03933       0.04063       0.04143       CAS10950
4615 0.04167       0.04130       0.04024       0.03842       0.03562       CAS10960
4620 0.03128       0.02526       0.02083                                   CAS10970
4000 1.                          1.                                        CAS10980
5000                                                                       CAS10990
   8 4.                                                                    CAS11000
4006 4.                                                                    CAS11010
4000 2.                          2.                                        CAS11020
5000                                                                       CAS11030
                                                                           CAS11040
```

```
A04NZXX,5,5000,60000.482511,RCST5786P FANPOD                                  CAS20010
*NOTAPES,PSS                                                                  CAS20020
LIBCOPY,A04RCS,LIB,OBJECT.                                                    CAS20030
COPY,LIB/RX,LGO.                                                              CAS20040
RFL(135000,580000)                                                            CAS20050
LGO.                                                                          CAS20060
'                                                                             CAS20070
         1 5.9744      55.92         9.918332     0.6                         CAS20080
         5 26.398      0.0           1.0          3.0          1.0            CAS20090
        10 1.0        -1.0           0.0          0.0          3.0            CAS20100
        15 1.0         1.0           1.0          1.0          13.0           CAS20110
       705 20.0        0.0           0.0003       0.001        0.002          CAS20120
       710 0.0035      0.005         0.0075       0.0125       0.025          CAS20130
       715 0.05        0.1           0.2          0.3          0.4            CAS20140
       720 0.5         0.6           0.7          0.8          0.9            CAS20150
       725 1.0                                                                CAS20160
       730 5.0         0.0           0.3262       0.5988       0.7495         CAS20170
       735 1.000                                                              CAS20180
       750 0.0         0.000283      0.000730     0.001095     0.001494       CAS20190
       755 0.001857    0.002324      0.002881     0.004212     0.005075       CAS20200
       760 0.00635     0.0084        0.00945      0.0096       0.00895        CAS20210
       765 0.0077      0.00605       0.0041       0.00195      0.0            CAS20220
       770 0.0         0.000257      0.000447     0.0006705    0.0009566      CAS20230
       775 0.00121     0.00168       0.00242      0.00401      0.00648        CAS20240
       780 0.00881     0.01252       0.01583      0.01851      0.01865        CAS20250
       785 0.01737     0.01455       0.01138      0.00633      0.0            CAS20260
       790 0.0         0.0000939     0.000225     0.000358     0.000552       CAS20270
       795 0.00072     0.000978      0.001481     0.00268      0.00511        CAS20280
       800 0.00872     0.01399       0.01729      0.01959      0.02150        CAS20290
       805 0.02081     0.01901       0.01537      0.01116      0.0            CAS20300
       810 0.0         0.000185      0.000485     0.000802     0.001207       CAS20310
       815 0.001582    0.00220       0.00326      0.00516      0.00791        CAS20320
       820 0.01058     0.01546       0.01951      0.02232      0.02380        CAS20330
       825 0.02334     0.02144       0.01821      0.01304      0.0            CAS20340
       830 0.0         0.000650      0.001194     0.001730     0.00246        CAS20350
       835 0.00308     0.00400       0.00545      0.00794      0.01073        CAS20360
       840 0.01314     0.01634       0.01871      0.02034      0.02011        CAS20370
       845 0.01869     0.01575       0.01160      0.00656      0.0            CAS20380
      1270 20.0        41.0          12.0         10.0         37.0           CAS20390
      1275 12.0        3.0           10.0         10.0         19.0           CAS20400
      1280 6.0         0.0           10.0                                     CAS20410
      1660 2.0         3.0           5.0          7.0          9.0            CAS20420
      1665 11.0        13.0          15.0         17.0         19.0           CAS20430
```

1870	0.025	0.05	0.1	0.2	0.3	CAS20440
1875	0.4	0.5	0.6	0.7	0.8	CAS20450
1880	0.9	0.95				CAS20460
2000	20.0	0.0	0.0003	0.001	0.002	CAS20470
2005	0.0035	0.005	0.0075	0.0125	0.025	CAS20480
2010	0.05	0.1	0.2	0.3	0.4	CAS20490
2015	0.5	0.6	0.7	0.8	0.9	CAS20500
2020	1.0					CAS20510
2025	5.0	0.0	0.3262	0.5988	0.7495	CAS20520
2030	1.0					CAS20530
2050	0.0	0.00325	0.00515	0.00715	0.00925	CAS20540
2055	0.010875	0.01315	0.016675	0.02196	0.03025	CAS20550
2060	0.03940	0.04755	0.050	0.04909	0.04486	CAS20560
2065	0.03767	0.02837	0.01810	0.00873	0.0	CAS20570
2070	0.0	0.00273	0.00441	0.005797	0.00740	CAS20580
2075	0.00876	0.010466	0.01316	0.01750	0.02406	CAS20590
2080	0.03341	0.04531	0.05125	0.05249	0.04879	CAS20600
2085	0.04121	0.03175	0.02050	0.01000	0.0	CAS20610
2090	0.0	0.00265	0.00429	0.005638	0.00719	CAS20620
2095	0.00852	0.010178	0.01280	0.01702	0.02330	CAS20630
2100	0.03308	0.04510	0.05101	0.05168	0.04679	CAS20640
2105	0.03864	0.03032	0.01960	0.00930	0.0	CAS20650
2110	0.0	0.00249	0.00403	0.00530	0.00676	CAS20660
2115	0.00801	0.009568	0.01203	0.01600	0.02230	CAS20670
2120	0.03120	0.04218	0.04770	0.04900	0.04698	CAS20680
2125	0.04147	0.03262	0.02166	0.01020	0.0	CAS20690
2130	0.0	0.00245	0.00395	0.00520	0.00664	CAS20700
2135	0.00786	0.009388	0.01181	0.01570	0.02265	CAS20710
2140	0.03140	0.04130	0.04656	0.04867	0.04806	CAS20720
2145	0.04405	0.03662	0.02631	0.01380	0.0	CAS20730
4005	65.0	-5.0	1.0	1.0	0.0	CAS20740
4010	0.0					CAS20750
4015	0.0	0.0	0.0			CAS20760
4020	2.0	0.0	65.0			CAS20770
4050	2.0	0.0	180.0			CAS20780
4090	3.059	3.059	3.059	3.059		CAS20790
4565	24.0	0.0	0.764	3.059	6.105	CAS20800
4570	9.160	12.21	15.29	18.34	21.40	CAS20810
4575	24.45	27.50	30.59	33.6	36.69	CAS20820
4580	39.75	42.76	45.85	48.90	51.95	CAS20830
4585	55.0	58.05	61.1	64.15	65.0	CAS20840
4600	0.0	0.675	1.225	1.69	1.9	CAS20850
4605	1.99	2.0	2.0	2.0	2.0	CAS20860

4610	2.0	2.0	2.0	2.0	1.99	CAS20870
4615	1.98	1.94	1.85	1.68	1.44	CAS20880
4620	1.19	0.79	0.5	0.0		CAS20890
4635	0.0	0.34	0.85	1.28	1.45	CAS20900
4640	1.525	1.555	1.555	1.555	1.555	CAS20910
4645	1.555	1.555	1.545	1.5	1.43	CAS20920
4650	1.345	1.215	1.07	0.9	0.72	CAS20930
4655	0.545	0.355	0.175	0.12		CAS20940
4670	0.0	0.075	0.38	0.65	0.44	CAS20950
4675	0.15	-0.155	-0.44	-0.72	-0.99	CAS20960
4680	-1.25	-1.50	-1.72	-1.85	-1.90	CAS20970
4685	-1.875	-1.78	-1.63	-1.4	-1.125	CAS20980
4690	-0.78	-0.44	-0.1	0.0		CAS20990
4705	121.0	6.0	3.0	40.0	20.0	CAS21000
4710	0.0	0.006155	0.02447	0.054495	0.09549	CAS21010
4715	0.146445	0.206105	0.273005	0.34549	0.421785	CAS21020
4720	0.5	0.578215	0.65451	0.726995	0.793895	CAS21030
4725	0.853555	0.90451	0.945505	0.97553	0.993845	CAS21040
4905	2.0	5.0	8.0	11.0	13.0	CAS21050
4910	16.0	19.0	22.0	25.0	28.0	CAS21060
4915	31.0	34.0	37.0	40.0	43.0	CAS21070
4920	46.0	49.0	52.0	55.0	58.0	CAS21080
4925	61.0	64.0	67.0	70.0	73.0	CAS21090
4930	76.0	79.0	82.0	85.0	88.0	CAS21100
4935	91.0	94.0	97.0	100.0	103.0	CAS21110
4940	106.0	109.0	112.0	115.0	118.0	CAS21120
20	6.0	0.0	1.25	2.5	4.0	CAS21130
25	5.5	26.398				CAS21140
50	33.0	0.0	20.0	32.5	40.0	CAS21150
55	45.0	50.0	57.5	70.0	90.0	CAS21160
60	110.0	122.5	130.0	135.0	140.0	CAS21170
65	147.5	160.0	180.0	200.0	212.5	CAS21180
70	220.0	225.0	230.0	237.5	250.0	CAS21190
75	270.0	290.0	302.5	310.0	315.0	CAS21200
80	320.0	327.5	340.0	360.0		CAS21210
87	8.818	12.960	3.3751			CAS21220
90	1.0	1.0	1.0	1.0	1.0	CAS21230
95	1.0	1.0	1.0	1.0	1.0	CAS21240
100	1.0	1.0	1.0	1.0	1.0	CAS21250
105	1.0	1.0	1.0	1.0	1.0	CAS21260
110	1.0	1.0	1.0	1.0	1.0	CAS21270
115	1.0	1.0	1.0	1.0	1.0	CAS21280
120	1.0	1.0	1.0	1.0	1.0093	CAS21290

125	1.0269	1.0443	1.0600	1.0443	1.0269	CAS21300
130	1.0093	1.0	1.0093	1.0269	1.0443	CAS21310
135	1.06	1.0443	1.0269	1.0093	1.0	CAS21320
140	1.0093	1.0269	1.0443	1.06	1.0443	CAS21330
145	1.0269	1.0093	1.0	1.0093	1.0269	CAS21340
150	1.0443	1.06	1.0443	1.0269	1.0093	CAS21350
155	1.0	1.0	1.0349	1.1009	1.1659	CAS21360
160	1.225	1.1659	1.1009	1.0349	1.0	CAS21370
165	1.0349	1.1009	1.1659	1.225	1.1659	CAS21380
170	1.1009	1.0349	1.0	1.0349	1.1009	CAS21390
175	1.1659	1.225	1.1659	1.1009	1.0349	CAS21400
180	1.0	1.0349	1.1009	1.1659	1.225	CAS21410
185	1.1659	1.1009	1.0349	1.0	1.0	CAS21420
190	1.0582	1.1682	1.2764	1.3749	1.2764	CAS21430
195	1.1682	1.0582	1.0	1.0582	1.1682	CAS21440
200	1.2764	1.3749	1.2764	1.1682	1.0582	CAS21450
205	1.0	1.0582	1.1682	1.2764	1.3749	CAS21460
210	1.2764	1.1682	1.0582	1.0	1.0582	CAS21470
215	1.1682	1.2764	1.3749	1.2764	1.1682	CAS21480
220	1.0582	1.0	1.0	1.0642	1.1857	CAS21490
225	1.3054	1.4142	1.3054	1.1857	1.0642	CAS21500
230	1.0	1.0642	1.1857	1.3054	1.4142	CAS21510
235	1.3054	1.1857	1.0642	1.0	1.0642	CAS21520
240	1.1857	1.3054	1.4142	1.3054	1.1857	CAS21530
245	1.0642	1.0	1.0642	1.1857	1.3054	CAS21540
250	1.4142	1.3054	1.1857	1.0642	1.0	CAS21550
255	1.0	1.0642	1.1857	1.3054	1.4142	CAS21560
260	1.3054	1.1857	1.0642	1.0	1.0642	CAS21570
265	1.1857	1.3054	1.4142	1.3054	1.1857	CAS21580
270	1.0642	1.0	1.0642	1.1857	1.3054	CAS21590
275	1.4142	1.3054	1.1857	1.0642	1.0	CAS21600
280	1.0642	1.1857	1.3054	1.4142	1.3054	CAS21610
285	1.1857	1.0642	1.0			CAS21620
565	34.0	0.0	0.003	0.0075	0.014	CAS21630
570	0.023	0.036	0.06	0.1	0.16	CAS21640
575	0.22	0.28	0.34	0.4	0.6	CAS21650
580	0.9	1.2	1.6	2.0	3.0	CAS21660
585	4.4	6.0	13.36	14.5	16.5	CAS21670
590	18.5	20.0	22.0	24.0	25.0	CAS21680
595	26.0	26.2	26.3	26.37	26.398	CAS21690
600	0.0	0.04	0.1	0.164	0.23	CAS21700
605	0.315	0.435	0.575	0.728	0.85	CAS21710
610	0.96	1.06	1.139	1.37	1.68	CAS21720

```
 615 1.949       2.225       2.44        2.785       3.02        CAS21730
 620 3.065       3.065       3.065       3.065       3.065       CAS21740
 625 3.065       3.065       3.065       3.065       3.065       CAS21750
 630 3.065       3.065       3.065       3.065                   CAS21760
 635 0.0         0.03        0.075       0.12        0.17        CAS21770
 640 0.225       0.31        0.41        0.52        0.615       CAS21780
 645 0.695       0.76        0.83        1.02        1.245       CAS21790
 650 1.42        1.575       1.66        1.69        1.69        CAS21800
 655 1.69        1.69        1.645       1.495       1.275       CAS21810
 660 1.085       0.805       0.505       0.355       0.186       CAS21820
 665 0.155       0.105       0.060       0.0                     CAS21830
1295 1.0         3.0         5.0         7.0         9.0         CAS21840
1300 11.0        13.0        15.0        17.0        19.0        CAS21850
1460 0.0         0.0911      0.362       0.804       1.403       CAS21860
1465 2.142       3.0         3.945       4.96        6.0         CAS21870
1470 7.0         8.0         9.0         10.0        11.0        CAS21880
1475 12.0        13.0        14.0        15.0        16.0        CAS21890
1480 17.0        18.0        18.877      19.745      20.595      CAS21900
1485 21.42       22.2        22.94       23.625      24.24       CAS21910
1490 24.8        25.28       25.66       25.99       26.21       CAS21920
1495 26.34       26.398                                          CAS21930
1610 0.0         26.566      45.0        90.0        135.0       CAS21940
1615 153.434     180.0       206.566     225.0       270.0       CAS21950
1620 315.0       333.434                                         CAS21960
1720 2.0         3.0         4.0         5.0         6.0         CAS21970
1725 7.0         8.0         10.0        12.0        14.0        CAS21980
1730 16.0        18.0        20.0        22.0        24.0        CAS21990
1735 26.0        28.0        30.0        32.0                    CAS22000
1970 0.0523      0.0489      0.0456      0.0366     -0.00785     CAS22010
1985 0.3949      0.175      -0.14       -0.4        -0.775       CAS22020
1210 0.11        17.64       30.09       0.1874      18.82       CAS22030
1215 0.0         0.2648      19.98       0.0         0.3422      CAS22040
1220 21.15       31.76       1.0         31.018      36.418      CAS22050
1200 0.          0.          3.7051                              CAS22060
1205 24.                     2.          5.                      CAS22070
   5                                     0.04                    CAS22080
4006-4.96                                                        CAS22090
4000 1.                      1.                                  CAS22100
5000                                                             CAS22110
   8 3.06                                                        CAS22120
4006-1.94                                                        CAS22130
4000 2.                      2.                                  CAS22140
5000                                                             CAS22150
```

```
1240 0.11         17.64        30.09        0.1447       18.172       CAS22160
1245 0.0          0.1794       18.695       0.0          0.2141       CAS22170
1250 19.22        30.845       0.2488       19.74        31.09        CAS22180
1255 0.2835       20.26        0.0          0.3182       20.78        CAS22190
1260 0.0          0.3529       21.3         31.825                    CAS22200
1210 0.574        24.62        33.395       0.6305       25.475       CAS22210
1215 0.0          0.6875       26.32        0.0          0.7449       CAS22220
1220 27.19        34.6         1.0          31.018       36.418       CAS22230
  87 8.756        12.96        3.7051                                 CAS22240
1200 0.           0.           3.7051       4.           10.          CAS22250
1205 16.          14.          2.           5.           8.           CAS22260
   5              0.18                      0.18                      CAS22270
4006-4.82                                                             CAS22280
4000 2.           1.           1.                                     CAS22290
5000                                                                  CAS22300
   5              3.19                      3.19                      CAS22310
4006-1.81                                                             CAS22320
4000 2.           2.           2.                                     CAS22330
5000                                                                  CAS22340
                                                                      CAS22350
```

DESCRIPTION OF OUTPUT

The program output is printed in the following sequence;

1. Input data
2. Boundary conditions
3. Longitudinal and lateral surface velocities and pressures at the control points on the fuselage
4. Total and section loads on the fuselage
5. Fanpod results in an analogous manner to the fuselage
6. Linear net pressures on the wing at specified locations
7. Nonlinear surface velocities and pressures on the wing at the control points
8. Total and section loads on the wing
9. Pylon results in an analogous manner to the wing
10. Total configuration forces and pitching moment.

Description of output nomenclature list in the order that it is printed:

Matrix B - Total configuration boundary conditions

(X_Q, Y_Q, Z_Q)-Fuselage or fanpod control point locations

VT/VFS - Lateral surface velocity on the fuselage or fanpod

WM/VFS - Longitudinal surface velocity on the fuselage or fanpod

CP - Surface pressure coefficient on the fuselage or fanpod

CL - Lift coefficient

CD - Drag coefficient

CMXY - Pitching moment due to lifting forces

CMYZ - Pitching moment due to drag forces

CM - Pitching moment due to all forces

CLW/WA - Section lift coefficient on the fuselage or fanpod

CDW/WA - Section drag coefficient on the fuselage or fanpod

X/C - Percent chord station

ETA - Percent wing or pylon semi-span station

U/V UP - Longitudinal velocity on upper surface of wing or pylon

U/V LOW - Longitudinal velocity on lower surface of wing or pylon

V/V UP - Lateral velocity on upper surface of wing or pylon

V/V LOW - Lateral velocity on lower surface of wing or pylon

CP LOW - Lower surface pressure coefficient on wing or pylon

CP UP - Upper surface pressure coefficient on wing or pylon

CP NETL - Linear net pressure on wing or pylon

CP L-U - Nonlinear net pressure on wing or pylon

CNC/CA - Section lift on wing or pylon

CDC/CA - Section zero suction drag on wing or pylon

CMC/CA - Section pitching moment on wing or pylon

X/C CP - Section center of pressure on the wing or pylon

WCL - Wing lift coefficient

PCL - Pylon lift coefficient

WCD - Wing zero suction drag coefficient

PCD - Pylon zero suction drag coefficient

SAMPLE OUTPUT
(CASE I)

** SUHSON   INTERFERENCE PRESSURE PROGRAM * INPUT D   **

1	5.974400	55.920000	9.918332	.600000	
5	26.398000	0.000000	1.000000	.040000	1.000000
10	1.000000	-1.000000	0.000000	0.000000	3.000000
15	1.000000	1.000000	1.000000	1.000000	13.000000
20	6.000000	0.000000	1.250000	2.500000	4.000000
25	5.500000	26.398000	0.000000	0.000000	0.000000
50	33.000000	0.000000	20.000000	32.500000	40.000000
55	45.000000	50.000000	57.500000	70.000000	90.000000
60	110.000000	122.500000	130.000000	135.000000	140.000000
65	147.500000	160.000000	180.000000	200.000000	212.500000
70	220.000000	225.000000	230.000000	237.500000	250.000000
75	270.000000	290.000000	302.500000	310.000000	315.000000
80	320.000000	327.500000	340.000000	360.000000	0.000000
85	0.000000	0.000000	8.818000	12.960000	3.375100
90	1.000000	1.000000	1.000000	1.000000	1.000000
95	1.000000	1.000000	1.000000	1.000000	1.000000
100	1.000000	1.000000	1.000000	1.000000	1.000000
105	1.000000	1.000000	1.000000	1.000000	1.000000
110	1.000000	1.000000	1.000000	1.000000	1.000000
115	1.000000	1.000000	1.000000	1.000000	1.000000
120	1.000000	1.000000	1.000000	1.000000	1.009300
125	1.026900	1.044300	1.060000	1.044300	1.026900
130	1.009300	1.000000	1.009300	1.026900	1.044300
135	1.060000	1.044300	1.026900	1.009300	1.000000
140	1.009300	1.026900	1.044300	1.060000	1.044300
145	1.026900	1.009300	1.000000	1.009300	1.026900
150	1.044300	1.060000	1.044300	1.026900	1.009300
155	1.000000	1.000000	1.034900	1.100900	1.165900
160	1.225000	1.165900	1.100900	1.034900	1.000000
165	1.034900	1.100900	1.165900	1.225000	1.165900
170	1.100900	1.034900	1.000000	1.034900	1.100900
175	1.165900	1.225000	1.165900	1.100900	1.034900
180	1.000000	1.034900	1.100900	1.165900	1.225000
185	1.165900	1.100900	1.034900	1.000000	1.000000
190	1.058200	1.168200	1.276400	1.374900	1.276400
195	1.168200	1.058200	1.000000	1.058200	1.168200
200	1.276400	1.374900	1.276400	1.168200	1.058200
205	1.000000	1.058200	1.168200	1.276400	1.374900
210	1.276400	1.168200	1.058200	1.000000	1.058200
215	1.168200	1.276400	1.374900	1.276400	1.168200
220	1.058200	1.000000	1.000000	1.064200	1.185700

225	1.305400	1.414200	1.305400	1.185700	1.064200
230	1.000000	1.064200	1.185700	1.305400	1.414200
235	1.305400	1.185700	1.064200	1.000000	1.064200
240	1.185700	1.305400	1.414200	1.305400	1.185700
245	1.064200	1.000000	1.064200	1.185700	1.305400
250	1.414200	1.305400	1.185700	1.064200	1.000000
255	1.000000	1.064200	1.185700	1.305400	1.414200
260	1.305400	1.185700	1.064200	1.000000	1.064200
265	1.185700	1.305400	1.414200	1.305400	1.[illegible]700
270	1.064200	1.000000	1.064200	1.185700	1.305400
275	1.414200	1.305400	1.185700	1.064200	1.000000
280	1.064200	1.185700	1.305400	1.414200	1.305400
285	1.185700	1.064200	1.000000	0.000000	0.000000
565	34.000000	0.000000	.003000	.007500	.014000
570	.023000	.036000	.060000	.100000	.160000
575	.220000	.280000	.340000	.400000	.600000
580	.900000	1.200000	1.600000	2.000000	3.000000
585	4.400000	6.000000	13.360000	14.500000	16.500000
590	18.500000	20.000000	22.000000	24.000000	25.000000
595	26.000000	26.200000	26.300000	26.370000	26.398000
600	0.000000	.040000	.100000	.164000	.230000
605	.315000	.435000	.575000	.728000	.850000
610	.960000	1.060000	1.139000	1.370000	1.680000
615	1.949000	2.225000	2.440000	2.785000	3.020000
620	3.065000	3.065000	3.065000	3.065000	3.065000
625	3.065000	3.065000	3.065000	3.065000	3.065000
630	3.065000	3.065000	3.065000	3.065000	3.065000
635	0.000000	.030000	.075000	.120000	.170000
640	.225000	.310000	.410000	.520000	.615000
645	.695000	.760000	.830000	1.020000	1.245000
650	1.420000	1.575000	1.660000	1.690000	1.690000
655	1.690000	1.690000	1.645000	1.495000	1.275000
660	1.085000	.805000	.505000	.355000	.186000
665	.155000	.105000	.060000	0.000000	0.000000
705	20.000000	0.000000	.000300	.001000	.002000
710	.003500	.005000	.007500	.012500	.025000
715	.050000	.100000	.200000	.300000	.400000
720	.500000	.600000	.700000	.800000	.900000
725	1.000000	0.000000	0.000000	0.000000	0.000000
730	5.000000	0.000000	.326200	.598800	.749500
735	1.000000	0.000000	0.000000	0.000000	0.000000
750	0.000000	.000283	.000730	.001095	.001494

755	.001857	.002324	.002881	.004212	.005075
760	.006350	.008400	.009450	.009600	.008950
765	.007700	.006050	.004100	.001950	0.000000
770	0.000000	.000257	.000447	.000671	.000957
775	.001210	.001680	.002420	.004010	.006480
780	.008810	.012520	.015830	.018510	.018650
785	.017370	.014550	.011380	.006330	0.000000
790	0.000000	.000094	.000225	.000358	.000552
795	.000720	.000978	.001481	.002680	.005110
800	.008720	.013990	.017290	.019590	.021500
805	.020810	.019010	.015370	.011160	0.000000
810	0.000000	.000185	.000485	.000802	.001207
815	.001582	.002200	.003260	.005160	.007910
820	.010580	.015460	.019510	.022320	.023800
825	.023340	.021440	.018210	.013040	0.000000
830	0.000000	.000650	.001194	.001730	.002460
835	.003080	.004000	.005450	.007940	.010730
840	.013140	.016340	.018710	.020340	. 110
845	.018690	.015750	.011600	.006560	0.000000
1200	0.000000	0.000000	3.705100	0.000000	0.000000
1205	24.000000	0.000000	2.000000	5.000000	0.000000
1210	.110000	17.640000	30.090000	.187400	18.820000
1215	0.000000	.264800	19.080000	0.000000	.342200
1220	21.150000	31.760000	1.000000	31.018000	36.418000
1270	20.000000	41.000000	12.000000	10.000000	37.000000
1275	12.000000	3.000000	10.000000	10.000000	19.000000
1280	6.000000	0.000000	10.000000	0.000000	0.000000
1295	1.000000	3.000000	5.000000	7.000000	9.000000
1300	11.000000	13.000000	15.000000	17.000000	19.000000
1460	0.000000	.091100	.362000	.804000	1.403000
1465	2.142000	3.000000	3.945000	4.960000	6.000000
1470	7.000000	8.000000	9.000000	10.000000	11.000000
1475	12.000000	13.000000	14.000000	15.000000	16.000000
1480	17.000000	18.000000	18.877000	19.745000	20.595000
1485	21.420000	22.200000	22.940000	23.625000	24.240000
1490	24.800000	25.280000	25.660000	25.990000	26.210000
1495	26.340000	26.398000	0.000000	0.000000	0.000000
1610	0.000000	26.566000	45.000000	90.000000	135.000000
1615	153.434000	180.000000	206.566000	225.000000	270.000000
1620	315.000000	333.434000	0.000000	0.000000	0.000000
1660	2.000000	3.000000	5.000000	7.000000	9.000000
1665	11.000000	13.000000	15.000000	17.000000	19.000000

1720	2.000000	3.000000	4.000000	5.000000	6.000000
1725	7.000000	8.000000	10.000000	12.000000	14.000000
1730	16.000000	18.000000	20.000000	22.000000	24.000000
1735	26.000000	28.000000	30.000000	32.000000	0.000000
1870	.025000	.050000	.100000	.200000	.300000
1875	.400000	.500000	.600000	.700000	.800000
1880	.900000	.050000	0.000000	0.000000	0.000000
1970	.052360	.049026	.049393	.037769	-.008132
1985	.394900	.175000	-.140000	-.400000	-.775000
2000	20.000000	0.000000	.000300	.001000	.002000
2005	.003500	.005000	.007500	.012500	.025000
2010	.050000	.100000	.200000	.300000	.400000
2015	.500000	.600000	.700000	.800000	.900000
2020	1.000000	0.000000	0.000000	0.000000	0.000000
2025	5.000000	0.000000	.326200	.598800	.749500
2030	1.000000	0.000000	0.000000	0.000000	0.000000
2050	0.000000	.003250	.005150	.007150	.009250
2055	.010875	.013150	.016675	.021960	.030250
2060	.039400	.047550	.050000	.049090	.044860
2065	.037670	.028370	.018100	.008730	0.000000
2070	0.000000	.002730	.004410	.005797	.007400
2075	.008760	.010466	.013160	.017500	.024060
2080	.033410	.045310	.051250	.052490	.048790
2085	.041210	.031750	.020500	.010000	0.000000
2090	0.000000	.002650	.004290	.005638	.007100
2095	.008520	.010178	.012800	.017020	.023300
2100	.033080	.045100	.051010	.051680	.[illegible]790
2105	.038640	.030320	.019600	.009300	0.000000
2110	0.000000	.002490	.004030	.005300	.006760
2115	.008010	.009568	.012030	.016000	.022300
2120	.031200	.042180	.047700	.049000	.046980
2125	.041470	.032620	.021660	.010200	0.000000
2130	0.000000	.002450	.003950	.005200	.006640
2135	.007860	.009388	.011810	.015700	.022650
2140	.031400	.041300	.046560	.048670	.048060
2145	.044050	.036620	.026310	.013800	0.000000
3380	1000000.000000	1000000.000000	1000000.000000	1000000.000000	1000000.000000
3385	1000000.000000	1000000.000000	1000000.000000	0.000000	0.000000
4000	1.000000	0.000000	1.000000	0.000000	0.000000
4005	65.000000	-4.960000	1.000000	1.000000	0.000000
4020	2.000000	0.000000	65.000000	0.000000	0.000000
4050	2.000000	0.000000	180.000000	0.000000	0.000000
4090	3.059000	3.059000	3.059000	3.059000	0.000000

4565	24.000000	0.000000	.764000	3.059000	6.105000
4570	9.160000	12.210000	15.290000	18.340000	21.400000
4575	24.450000	27.500000	30.590000	33.600000	36.690000
4580	39.750000	42.760000	45.850000	48.900000	51.950000
4585	55.000000	58.050000	61.100000	64.150000	65.000000
4600	0.000000	.675000	1.225000	1.690000	1.900000
4605	1.990000	2.000000	2.000000	2.000000	2.000000
4610	2.000000	2.000000	2.000000	2.000000	1.990000
4615	1.980000	1.940000	1.850000	1.680000	1.440000
4620	1.190000	.790000	.500000	0.000000	0.000000
4635	0.000000	.340000	.850000	1.280000	1.450000
4640	1.525000	1.555000	1.555000	1.555000	1.555000
4645	1.555000	1.555000	1.545000	1.500000	1.430000
4650	1.345000	1.215000	1.070000	.900000	.720000
4655	.545000	.355000	.175000	.120000	0.000000
4670	0.000000	.075000	.380000	.650000	.440000
4675	.150000	-.155000	-.440000	-.720000	-.990000
4680	-1.250000	-1.500000	-1.720000	-1.850000	-1.900000
4685	-1.875000	-1.780000	-1.630000	-1.400000	-1.125000
4690	-.780000	-.440000	-.100000	0.000000	0.000000
4705	121.000000	6.000000	3.000000	40.000000	20.000000
4710	0.000000	.006155	.024470	.054495	.095400
4715	.146445	.206105	.273005	.345490	.421785
4720	.500000	.578215	.654510	.726995	.793805
4725	.853555	.904510	.945505	.975530	.993845
4905	2.000000	5.000000	8.000000	11.000000	13.000000
4910	16.000000	19.000000	22.000000	25.000000	28.000000
4915	31.000000	34.000000	37.000000	40.000000	43.000000
4920	46.000000	49.000000	52.000000	55.000000	58.000000
4925	61.000000	64.000000	67.000000	70.000000	73.000000
4930	76.000000	79.000000	82.000000	85.000000	88.000000
4935	91.000000	94.000000	97.000000	100.000000	103.000000
4940	106.000000	109.000000	112.000000	115.000000	118.000000

** COMPLE) MATRIX 1

** COMPLETED MATRIX 3

** COMPLETED MATRIX 11

** COMPLETED MATRIX 13

* MATRIX- A

.82252	.81836	.79882	.64546	.59091	.55474	.50708	.42855	.25239	.12284
.07295	.03798	.01181	-.00626	-.00320	-.00173	-.00043	.00101	.00326	.00587
.00224	.00107	.00097	.00912	-.00214	-.00687	-.00601	-.00725	-.00681	-.00249
.01919	.01669	-.00091	.00275	.01211	.01732	.01466	-.02210	-.06758	-.27200
.86810	.86408	.84445	.62373	.56818	.53702	.49078	.41205	.25445	.13209
.08132	.04091	.00795	-.00039	.05553	.06043	.01144	-.03208	-.05540	-.03433
-.00260	.00641	.00825	.01762	.01052	.00264	-.00393	-.01578	-.02610	-.02552
-.00495	-.00659	-.05405	-.04356	-.01486	.00674	.01077	-.26311	-.50591	-.71795
.91876	.91545	.89867	.58334	.51637	.49011	.44603	.36640	.24889	.14354
.09371	.04625	.00348	-.00089	.00487	.01543	.01337	.00145	-.00905	-.00981
-.00313	.00651	.00523	.01152	.00558	-.01128	-.03369	-.07066	-.10463	-.11596
-.10472	-.10842	-.21584	-.19266	-.12895	-.07756	-.06424	-.67523	-.78925	-.82670
.88499	.88087	.85992	.47865	.40348	.38659	.35373	.27957	.21654	.13780
.09570	.05008	.00792	-.00173	-.00310	-.00134	.00225	.00454	.00393	.00077
-.00381	-.01544	-.03046	-.03798	-.05749	-.08804	-.12294	-.17872	-.22268	-.24445
-.25339	-.26149	-.36206	-.34116	-.28638	-.24738	-.24788	-.76505	-.82820	-.83396
.77340	.76728	.73730	.40955	.32596	.30161	.27447	.20518	.17705	.11881
.08652	.05060	.01989	-.00060	-.00291	-.00246	-.00013	.00278	.00343	-.00016
-.01203	-.04945	-.07713	-.09707	-.13111	-.16335	-.19032	-.23459	-.25958	-.27419
-.28527	-.28741	-.32951	-.32028	-.29603	-.27960	-.27989	-.51628	-.67183	-.75461
.69340	.68669	.65508	.40698	.31323	.27523	.24515	.17511	.15721	.10689
.07979	.05061	.02850	.00050	-.00172	-.00147	-.00006	.00193	.00206	-.00210
-.01845	-.07092	-.10437	-.12941	-.17009	-.20003	-.21967	-.25507	-.26912	-.27980
-.29076	-.29125	-.30551	-.30224	-.29393	-.28950	-.29228	-.33281	-.36950	-.46262
-.13354	-.04985	-.02874	-.02120	-.00516	.01159	.02844	.03807	.05921	.06331
-.12070	-.04187	-.02497	-.01713	-.00127	.01711	.03356	.04474	.07841	.08489
-.10155	-.02932	-.01815	-.00989	.00573	.02631	.04222	.05551	.10622	.11594
-.08372	-.01698	-.01056	-.00195	.01346	.03587	.05133	.06639	.13124	.14365
-.06170	-.01952	-.00620	.00665	.01847	.04248	.05954	.06996	.13395	.14629
-.04058	-.02512	-.00276	.01485	.02260	.04801	.06700	.07169	.13127	.14286
-.04181	-.01980	.00139	.01571	.02650	.05022	.06715	.07457	.11989	.12880
-.04711	-.01310	.00517	.01506	.02989	.05151	.06585	.07728	.10796	.11408
-.05178	-.00720	.00848	.01447	.03287	.05264	.06469	.07966	.09743	.10110
-.05196	-.00140	.01192	.01593	.03566	.05326	.06328	.07923	.08779	.08970
-.04308	.00509	.01615	.02182	.03858	.05303	.06124	.07314	.07862	.07988
-.03500	.01101	.02002	.02718	.04126	.05288	.05946	.06773	.07043	.07110

** SUBSONIC INTERFERENCE PRESS PROGRAM ** FUSELAGE OUTPUT **

NO.	XQ	YQ	ZQ	VT/VFS	VM/VFS	CP
1	.01416	.01075	.02101	.02704	-.28092	.99913
2	.18366	.13806	.27105	.03243	-.00943	1.09191
3	.55616	.40163	.80035	.05369	.37904	.92103
4	1.12873	.64054	1.44892	.07254	.59575	.67751
5	1.61998	.73148	1.85440	.07030	.69537	.53550
6	2.51773	.87940	2.51172	.06412	.80181	.36434
7	3.60346	1.04814	3.22736	.05400	.93520	.12384
8	4.87035	1.20960	3.91875	.04154	1.10320	-.21450
9	6.31046	1.34877	4.46039	.02786	1.27824	-.59923
10	7.91476	1.44525	4.64878	.01842	1.20360	-.43113
11	9.67319	1.51603	4.71361	.01343	1.15140	-.31645
12	11.57474	1.56003	4.66623	.01178	1.11490	-.23786
13	13.60747	1.57496	4.55352	.01228	1.08692	-.17859
14	15.75864	1.57743	4.37770	-.00270	1.05711	-.11625
15	18.01477	1.57743	4.16729	-.05637	1.05237	-.10956
16	20.36171	1.57743	3.95139	-.01435	1.10227	-.21106
17	22.78475	1.57743	3.73350	.00822	1.12197	-.25291
18	25.26870	1.57743	3.51643	.02832	1.10538	-.21825
19	27.79799	1.57743	3.30254	.03935	1.07095	-.14651
20	30.35675	1.57743	3.09507	.03066	1.03254	-.06668
21	32.92895	1.57743	2.88399	-.01681	1.01837	-.03723
22	35.49847	1.57743	2.66640	-.02884	1.01714	-.03529
23	38.04918	1.57396	2.44962	-.03120	1.01625	-.03364
24	40.56510	1.56742	2.24717	-.03169	1.02328	-.04789
25	43.03046	1.55952	2.05933	-.03356	1.03707	-.07611
26	45.42981	1.53614	1.83545	-.03380	1.03026	-.06223
27	47.74809	1.49156	1.62334	-.03149	1.02938	-.05824
28	49.97077	1.41846	1.42309	-.02921	1.02528	-.05181
29	52.08393	1.31667	1.23927	-.02839	1.01965	-.04034
30	54.07430	1.19431	1.06513	-.03076	1.00477	-.01050
31	55.92942	1.07682	.94126	-.03618	1.00570	-.01273
32	57.63766	.96726	.84856	-.04216	1.02508	-.05232
33	59.18829	.82371	.74342	-.04689	1.01881	-.04003
34	60.57161	.67392	.64162	-.06915	.99638	.00245
35	61.77894	.56720	.56115	-.10820	.99178	.00467
36	62.80271	.49824	.49963	-.14088	1.02243	-.06484
37	63.63650	.45203	.44985	-.15397	1.06137	-.14820
38	64.27509	.34313	.40619	-.15465	1.00955	-.04294
39	64.71447	.14199	.37427	-.30926	1.00825	-.11108
40	64.95188	.02429	.35678	-.82505	1.09899	-.81970

** SUBSONIC INTERFERENCE PRESS PROGRAM ** FUSELAGE OUTPUT **

NO.	XQ	YQ	ZQ	VT/VFS	VM/VFS	CP
1	.01416	.02936	.01574	.09077	-.18117	1.04458
2	.18386	.37718	.20307	.10695	.05465	1.07612
3	.55616	1.09727	.60027	.16732	.42368	.85065
4	1.12873	1.74999	1.09246	.22238	.63983	.56804
5	1.61998	1.99843	1.40728	.21001	.73113	.43756
6	2.51773	2.40255	1.92234	.18628	.81815	.30390
7	3.60346	2.86357	2.48075	.15533	.93256	.10722
8	4.87035	3.30468	3.02172	.11903	1.08438	-.18682
9	6.31046	3.68492	3.43744	.07769	1.24192	-.52185
10	7.91476	3.94849	3.54787	.05163	1.18206	-.38573
11	9.67319	4.14187	3.55609	.04095	1.13982	-.29280
12	11.57474	4.26207	3.47261	.03706	1.11573	-.24082
13	13.60747	4.30287	3.33655	.03366	1.08276	-.17082
14	15.75864	4.30962	3.15125	.02221	.99121	.01703
15	18.01477	4.30962	2.94084	.07254	.83222	.31046
16	20.36171	4.30962	2.72494	.11017	1.01785	-.04796
17	22.78475	4.30962	2.50705	.06533	1.08771	-.18424
18	25.26870	4.30962	2.28997	.03735	1.04880	-.10045
19	27.79799	4.30962	2.07609	.04238	.98558	.02689
20	30.35675	4.30962	1.86875	.05884	.95005	.09475
21	32.92895	4.30962	1.66244	.02432	.98351	.03222
22	35.49847	4.30962	1.46715	.01216	1.00745	-.01509
23	38.04918	4.30014	1.28923	.01020	1.02247	-.04537
24	40.56510	4.28228	1.13605	.01156	1.02728	-.05517
25	43.03046	4.26070	1.00680	.01318	1.04011	-.08140
26	45.42981	4.19682	.86242	.02158	1.04105	-.08362
27	47.74809	4.07503	.73442	.03681	1.04194	-.08632
28	49.97077	3.87529	.62487	.05448	1.03391	-.07147
29	52.08393	3.59720	.53364	.07134	1.02150	-.04835
30	54.07430	3.26292	.45433	.08444	.99828	-.00369
31	55.92942	2.94194	.41562	.09305	.99393	.00345
32	57.63766	2.64261	.39974	.09725	1.01907	-.04775
33	59.18829	2.25041	.36922	.11845	.98948	.00690
34	60.57161	1.84118	.33602	.10496	.94624	.09441
35	61.77894	1.54963	.31317	.04928	.94542	.10473
36	62.80271	1.36121	.29872	.00112	1.00292	-.00584
37	63.63650	1.23497	.28718	-.02331	1.06731	-.13794
38	64.27509	.93744	.27447	.05049	.85541	.27213
39	64.71447	.38794	.26495	.02929	.71965	.50245
40	64.95188	.06636	.25966	-.73927	.52262	.18328

** SUBSONIC INTERFERENCE PRESS PROGRAM ** FUSELAGE OUTPUT **

NO.	XO	YO	ZO	VT/VFS	VM/VFS	CP
1	.01416	.04011	.00660	.17797	-.10748	1.04202
2	.18386	.51524	.08534	.19974	.11596	1.03009
3	.55616	1.49889	.25372	.26525	.49516	.72767
4	1.12873	2.39053	.47506	.31336	.72387	.39085
5	1.61998	2.72991	.63286	.29087	.79954	.28307
6	2.51773	3.28195	.90150	.25138	.85464	.21027
7	3.60346	3.91171	1.18757	.20968	.93968	.07352
8	4.87035	4.51428	1.46802	.16388	1.05993	-.14829
9	6.31046	5.03369	1.66563	.10599	1.18447	-.39900
10	7.91476	5.39373	1.64103	.07156	1.15502	-.32896
11	9.67319	5.65790	1.55120	.05864	1.13192	-.27746
12	11.57474	5.82210	1.40520	.05353	1.10964	-.22926
13	13.60747	5.87783	1.22870	.04848	1.07283	-.15121
14	15.75864	5.88705	1.02698	.05487	1.02126	-.04578
15	18.01477	5.88705	.81656	.09755	.97593	.03818
16	20.36171	5.88705	.60066	.13716	.99919	-.01716
17	22.78475	5.88705	.38278	.11981	1.02904	-.07280
18	25.26870	5.88705	.16570	.09320	1.02755	-.06416
19	27.79799	5.88705	-.04819	.09164	1.00881	-.02604
20	30.35675	5.88705	-.25529	.11524	.99051	.00562
21	32.92895	5.88705	-.45334	.14408	.99522	-.01121
22	35.49847	5.88705	-.61000	.16474	1.01407	-.05521
23	38.04918	5.87411	-.72062	.18081	1.02647	-.08565
24	40.56510	5.84970	-.78846	.19659	1.03334	-.10542
25	43.03046	5.82022	-.81622	.21308	1.04844	-.14275
26	45.42981	5.73296	-.82293	.23523	1.05617	-.16822
27	47.74809	5.56659	-.80525	.26182	1.06058	-.19003
28	49.97077	5.29375	-.75770	.29254	1.05034	-.18561
29	52.08393	4.91387	-.68507	.31763	1.03704	-.17356
30	54.07430	4.45723	-.60361	.32604	1.01437	-.13360
31	55.92942	4.01876	-.49482	.35198	1.00370	-.12977
32	57.63766	3.60987	-.37762	.38326	1.02308	-.19024
33	59.18829	3.07411	-.27891	.43391	.97495	-.13707
34	60.57161	2.51510	-.19330	.46635	.90563	-.03752
35	61.77894	2.11684	-.11634	.47349	.91047	-.05289
36	62.80271	1.85945	-.04927	.46989	1.00259	-.22144
37	63.63650	1.68701	.00543	.43532	1.05693	-.29824
38	64.27509	1.28057	.04633	.47024	.79194	.15379
39	64.71447	.52993	.07558	.56500	.52147	.42410
40	64.95188	.09065	.09143	.46708	.25607	.76363

** SUBSONIC INTERFERENCE PRESSI PROGRAM ** FUSELAGE OUTPUT **

NO.	XO	YO	ZO	VT/VFS	VM/VFS	CP
1	.01416	.04011	-.00395	.17438	-.05758	1.05326
2	.18386	.51524	-.05060	.18011	.18182	1.01576
3	.55616	1.49889	-.14645	.19498	.58254	.65831
4	1.12873	2.39053	-.23785	.18194	.83325	.27935
5	1.61998	2.72991	-.26137	.19152	.88004	.19207
6	2.51773	3.28195	-.27726	.18164	.90449	.15091
7	3.60346	3.91171	-.30566	.16224	.96830	.03619
8	4.87035	4.51428	-.32604	.13647	1.05080	-.12146
9	6.31046	5.03369	-.38029	.09328	1.13181	-.28222
10	7.91476	5.39373	-.56079	.06522	1.12795	-.26970
11	9.67319	5.65790	-.76384	.05301	1.11974	-.25076
12	11.57474	5.82210	-.98204	.04705	1.10369	-.21602
13	13.60747	5.87783	-1.20523	.04496	1.07300	-.15125
14	15.75864	5.88705	-1.42593	.05448	1.04045	-.08485
15	18.01477	5.88705	-1.63634	.07346	1.01418	-.03386
16	20.36171	5.88705	-1.85224	.09199	1.01466	-.03786
17	22.78475	5.88705	-2.07013	.09761	1.01964	-.04897
18	25.26870	5.88705	-2.28720	.09678	1.02132	-.05221
19	27.79799	5.88705	-2.50109	.10059	1.01805	-.04636
20	30.35675	5.88705	-2.70793	.11776	1.01850	-.05097
21	32.92895	5.88705	-2.89644	.14816	1.02792	-.07801
22	35.49847	5.88705	-3.00850	.18414	1.03811	-.11046
23	38.04918	5.87411	-3.04140	.21533	1.04411	-.13487
24	40.56510	5.84970	-3.01070	.24546	1.04721	-.15470
25	43.03046	5.82022	-2.92127	.27585	1.05740	-.19083
26	45.42981	5.73296	-2.76900	.30574	1.05947	-.21178
27	47.74809	5.56659	-2.58310	.33305	1.05865	-.22687
28	49.97077	5.29375	-2.35415	.36209	1.03939	-.20745
29	52.08393	4.91387	-2.09232	.38900	1.01369	-.17602
30	54.07430	4.45723	-1.82522	.41554	.97581	-.12347
31	55.92942	4.01876	-1.54610	.43950	.95725	-.10840
32	57.63766	3.60987	-1.27525	.44514	.97508	-.14695
33	59.18829	3.07411	-1.02731	.43259	.93792	-.06642
34	60.57161	2.51510	-.80451	.46031	.86465	.04064
35	61.77894	2.11684	-.61229	.50842	.86360	-.00430
36	62.80271	1.85945	-.45110	.53287	.93925	-.16367
37	63.63650	1.68701	-.31991	.52036	.96669	-.20150
38	64.27509	1.28057	-.21711	.37403	.73784	.32477
39	64.71447	.52993	-.14307	.22042	.45431	.79632
40	64.95188	.09065	-.10282	.13292	.18776	1.03058

** SUBSONIC INTERFERENCE PRESS PROGRAM ** FUSELAGE OUTPUT **

NO.	XQ	YQ	ZQ	VT/VFS	VM/VFS	CP
1	.01416	.02936	-.01308	.08532	-.01932	1.08417
2	.18386	.37718	-.16834	.07379	.26014	1.00680
3	.55616	1.09727	-.49300	.04750	.69016	.54636
4	1.12873	1.74999	-.85525	.01695	.93575	.12548
5	1.61998	1.99843	-1.03579	.04411	.95731	.08222
6	2.51773	2.40255	-1.29810	.06469	.96355	.06779
7	3.60346	2.86357	-1.59884	.07143	1.01647	-.03819
8	4.87035	3.30468	-1.87974	.06832	1.06414	-.13538
9	6.31046	3.68492	-2.15210	.05246	1.10111	-.21105
10	7.91476	3.94849	-2.46763	.03943	1.10850	-.22559
11	9.67319	4.14187	-2.76873	.03202	1.10725	-.22242
12	11.57474	4.26207	-3.04945	.02700	1.09484	-.19585
13	13.60747	4.30287	-3.31308	.02605	1.07501	-.15414
14	15.75864	4.30962	-3.55020	.03391	1.05078	-.10429
15	18.01477	4.30962	-3.76062	.04273	1.02704	-.05635
16	20.36171	4.30962	-3.97651	.05100	1.02247	-.04783
17	22.78475	4.30962	-4.19440	.05657	1.02057	-.04459
18	25.26870	4.30962	-4.41148	.06062	1.02130	-.04652
19	27.79799	4.30962	-4.62537	.06574	1.02307	-.05077
20	30.35675	4.30962	-4.83198	.07735	1.03606	-.07884
21	32.92895	4.30962	-5.01222	.09879	1.05613	-.12377
22	35.49847	4.30962	-5.08566	.12647	1.06243	-.14288
23	38.04918	4.30014	-5.05125	.15076	1.06274	-.15008
24	40.56510	4.28228	-4.93522	.17323	1.05912	-.14969
25	43.03046	4.26070	-4.74430	.19441	1.06058	-.16027
26	45.42981	4.19682	-4.45434	.20979	1.04665	-.13775
27	47.74809	4.07503	-4.12276	.21819	1.03533	-.11824
28	49.97077	3.87529	-3.73672	.22492	1.00784	-.06594
29	52.08393	3.59720	-3.31104	.23020	.97407	-.00181
30	54.07430	3.26292	-2.88316	.23940	.93701	.06508
31	55.92942	2.94194	-2.45653	.25373	.90922	.11001
32	57.63766	2.64261	-2.05262	.25604	.89089	.12604
33	59.18829	2.25041	-1.67544	.23103	.86664	.19902
34	60.57161	1.84118	-1.33383	.23628	.82035	.27788
35	61.77894	1.54963	-1.04180	.27221	.81199	.27303
36	62.80271	1.36121	-.79910	.28110	.84131	.21730
37	63.63650	1.23497	-.60167	.24967	.84760	.22359
38	64.27509	.93744	-.44526	.12857	.68157	.54362
39	64.71447	.38794	-.33243	-.06294	.48200	.81766
40	64.95188	.06636	-.27104	-.18718	.25226	.97683

** SUBSONIC INTERFERENCE PRESS PROGRAM ** FUSELAGE OUTPUT **

NO.	XQ	YQ	ZQ	VT/VFS	VM/VFS	CP
1	.01416	.01075	-.01835	.02474	.00894	1.09244
2	.18386	.13806	-.23631	.01389	.34226	.95503
3	.55616	.40163	-.69308	-.00798	.83308	.31443
4	1.12873	.64054	-1.21171	-.01558	1.01009	-.02049
5	1.61998	.73148	-1.48290	-.00236	1.02566	-.05175
6	2.51773	.87940	-1.88749	.01040	1.01811	-.03654
7	3.60346	1.04814	-2.34545	.01623	1.05206	-.10606
8	4.87035	1.20960	-2.77677	.01745	1.08164	-.16765
9	6.31046	1.34877	-3.17506	.01509	1.09653	-.19894
10	7.91476	1.44525	-3.56855	.01245	1.10197	-.21039
11	9.67319	1.51603	-3.92625	.01012	1.09937	-.20483
12	11.57474	1.56003	-4.24307	.00813	1.08888	-.18264
13	13.60747	1.57496	-4.53005	.00791	1.07795	-.15968
14	15.75864	1.57743	-4.77665	.01083	1.05717	-.11647
15	18.01477	1.57743	-4.98707	.01358	1.03272	-.06629
16	20.36171	1.57743	-5.20297	.01605	1.02648	-.05365
17	22.78475	1.57743	-5.42085	.01800	1.02278	-.04620
18	25.26870	1.57743	-5.63793	.01973	1.02333	-.04739
19	27.79799	1.57743	-5.85182	.02177	1.02711	-.05516
20	30.35675	1.57743	-6.05830	.02591	1.04720	-.09644
21	32.92895	1.57743	-6.23377	.03358	1.07479	-.15411
22	35.49847	1.57743	-6.28491	.04345	1.07988	-.16550
23	38.04918	1.57396	-6.21164	.05200	1.07710	-.16048
24	40.56510	1.56742	-6.04634	.05966	1.06959	-.14564
25	43.03046	1.55952	-5.79682	.06661	1.06589	-.13878
26	45.42981	1.53614	-5.42738	.07079	1.04023	-.08640
27	47.74809	1.49156	-5.01169	.07191	1.02219	-.04983
28	49.97077	1.41846	-4.53494	.07227	.99183	.01107
29	52.08393	1.31667	-4.01466	.07232	.95639	.08067
30	54.07430	1.19431	-3.49397	.07382	.92376	.14303
31	55.92942	1.07682	-2.98218	.07719	.89423	.19781
32	57.63766	.96726	-2.50143	.07746	.87257	.23753
33	59.18829	.82371	-2.04964	.06973	.84350	.29095
34	60.57161	.67392	-1.63943	.07067	.81530	.34023
35	61.77894	.56720	-1.28978	.08208	.80345	.35874
36	62.80271	.49824	-1.00001	.08432	.80756	.35131
37	63.63650	.45203	-.76434	.07124	.79854	.36890
38	64.27509	.34313	-.57698	.02816	.70790	.52081
39	64.71447	.14199	-.44176	-.07883	.59107	.68267
40	64.95188	.02429	-.36816	-.28684	.49492	.71451

FUSELAGE LOADS ** REF. FUSELAGE ARE 677.89822

* CL= .01568 * CD= .00028 * CMXY= -.09697 * CMYZ= .00284 * CM= -.09413

X	CLW/WA	CDW/WA	X	CLW/WA	CDW/WA
.00050	.00626	.08931	.52243	-.14526	.00599
.00201	.01973	.34800	.54482	-.13975	.01225
.00453	-.00191	.64964	.56712	-.13306	.01722
.00804	-.06038	.82346	.58928	-.12387	.02097
.01254	-.13964	.62818	.61126	-.11176	.02249
.01802	-.22807	.34623	.63302	-.09749	.02614
.02447	-.21956	.27653	.65451	-.08124	.03344
.03188	-.19827	.24600	.67569	-.06246	.03607
.04024	-.17374	.20898	.69651	-.04279	.03682
.04952	-.14808	.12120	.71694	-.02577	.03751
.05970	-.10509	.01054	.73693	-.00830	.03749
.07078	-.01219	-.12860	.75645	.01800	.03386
.08271	.12334	-.23806	.77545	.04312	.02659
.09549	.29939	-.29955	.79389	.06730	.01552
.10908	.24762	-.22594	.81174	.08212	.00214
.12346	.17325	-.13962	.82897	.08910	-.01250
.13860	.11820	-.10532	.84553	.10136	-.02295
.15447	.08045	-.07527	.86140	.11719	-.02949
.17103	.05912	-.04949	.87654	.13902	-.02846
.18826	.03931	-.02815	.89092	.15420	-.03054
.20611	.02206	-.01210	.90451	.15172	-.03882
.22455	-.01621	-.00424	.91729	.13941	-.04710
.24355	-.06126	-.00137	.92922	.12308	-.05568
.26307	-.10968	-.00275	.94030	.11022	-.05247
.28306	-.11817	-.00275	.95048	.10062	-.04385
.30349	.03372	.00071	.95976	.10240	-.03393
.32431	.13354	.00280	.96812	.11159	-.02631
.34549	.18155	.00353	.97553	.12920	-.02280
.36698	.16941	.00302	.98198	.12737	-.05134
.38874	.12574	.00210	.98746	.10314	-.10335
.41072	.06597	.00103	.99196	.07609	-.14224
.43288	.00181	.00004	.99547	.04889	-.17051
.45518	-.06652	-.00066	.99799	.02260	-.17021
.47757	-.11970	-.00049	.99950	.00564	-.16706
.50000	-.14256	.00090			

** WING L EAR PRESSURE COEF.

** LIST OF X/C

.0250 .0500 .1000 .2000 .3000 .4000 .5000 .6000 .7000 .8000
.9000 .9500

* ETA ** LISTS OF CP AT ABOVE X/C

ETA	.0250	.0500	.1000	.2000	.3000	.4000	.5000	.6000	.7000	.8000	.9000	.9500
.149	.8746	.6018	.3983	.2474	.1962	.1937	.2169	.2443	.2528	.2210	.1372	.0781
.226	.7850	.5528	.3996	.3181	.3014	.2989	.2966	.2878	.2674	.2298	.1659	.1166
.303	.7732	.5590	.4236	.3598	.3514	.3506	.3446	.3295	.3035	.2639	.2007	.1481
.381	.7427	.5796	.4729	.4073	.3906	.3874	.3813	.3645	.3353	.2955	.2373	.1843
.458	.7662	.6133	.5093	.4363	.4154	.4117	.4058	.3890	.3629	.3338	.2923	.2404
.536	.7879	.6457	.5437	.4609	.4346	.4307	.4260	.4109	.3904	.3755	.3539	.3033
.613	.8030	.6660	.5644	.4757	.4466	.4429	.4398	.4280	.4154	.4163	.4139	.3636
.690	.8196	.6444	.5305	.4628	.4479	.4452	.4402	.4330	.4334	.4488	.4486	.3899
.768	.7960	.6004	.4831	.4358	.4337	.4331	.4272	.4230	.4314	.4547	.4543	.3912
.845	.6826	.5348	.4441	.3997	.3947	.3959	.3935	.3881	.3863	.3910	.3754	.3186
.923	.5042	.4236	.3705	.3302	.3187	.3190	.3187	.3120	.3003	.2870	.2598	.2160
.971	.3360	.2861	.2446	.2012	.1877	.1906	.1948	.1907	.1779	.1619	.1427	.1195

105-

**WING NC .INEAR VELOCITY AND PRESSURE COFFFICIENT

ETA = .9710

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.16710	.90412	-.02667	.08066	-.35114	.17887	.53001	.33793
.05450	1.19383	.99272	-.03918	.04933	-.41062	.01209	.42271	.25088
.14645	1.18525	1.06164	-.04301	.03210	-.39201	-.12664	.26537	.22008
.27300	1.16435	1.06088	-.05989	.03105	-.34783	-.12501	.22282	.18902
.42178	1.15484	1.05104	-.08830	.02906	-.33109	-.10452	.22656	.19188
.57822	1.14139	1.03414	-.11492	.03021	-.30709	-.06992	.23717	.19247
.72700	1.10036	.99368	-.12888	.04215	-.22278	.01084	.23362	.17359
.85355	1.04569	.94670	-.13100	.06145	-.10953	.10088	.21041	.15304
.94550	.98065	.89450	-.12304	.08343	.02323	.19629	.17306	.12265
.99384	.85986	.80645	-.10047	.10814	.25624	.34835	.09211	.05011

ETA = .9226

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.20420	.86952	-.06428	.07845	-.43596	.24291	.67888	.50734
.05450	1.21470	.95282	-.07245	.04448	-.46030	.09088	.55119	.41596
.14645	1.21816	1.03000	-.07927	.01731	-.46896	-.06086	.40810	.34650
.27300	1.21541	1.04514	-.08575	.01779	-.46380	-.09187	.37192	.32007
.42178	1.20757	1.04004	-.09518	.02351	-.44796	-.08162	.36634	.31929
.57822	1.18865	1.02180	-.10304	.03211	-.40761	-.04492	.36268	.31406
.72700	1.13892	.97423	-.10164	.05001	-.29905	.04858	.34763	.29688
.85355	1.07537	.91905	-.09610	.06987	-.16320	.15252	.31572	.27490
.94550	1.00111	.86835	-.08694	.08650	-.00978	.24365	.25343	.22204
.99384	.87171	.79882	-.06451	.10341	.24102	.36244	.12142	.08901

ETA = .8452

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.25484	.83673	-.08511	.08543	-.55204	.30037	.85241	.68827
.05450	1.23904	.92582	-.08344	.04950	-.51623	.14220	.65843	.52079
.14645	1.23757	1.01063	-.08773	.01635	-.51360	-.02159	.49201	.41402
.27300	1.24254	1.03173	-.09420	.01093	-.52582	-.06421	.46161	.39468
.42178	1.23678	1.03012	-.09759	.01425	-.51349	-.06101	.45248	.39578
.57822	1.21482	1.01146	-.09588	.02322	-.46418	-.02355	.44063	.38923
.72700	1.16260	.95657	-.08628	.04296	-.34762	.08376	.43137	.38719
.85355	1.10183	.89326	-.07661	.06365	-.21557	.20159	.41716	.38985
.94550	1.02868	.84637	-.06542	.07796	-.06210	.28458	.34668	.32701
.99384	.89352	.79870	-.04091	.08947	.20357	.36549	.16192	.13283

ETA = .7678

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.29115	.81826	-.09420	.08920	-.63581	.33196	.96777	.80345
.05450	1.25524	.91171	-.08512	.05249	-.55293	.16852	.72145	.58182
.14645	1.25081	1.00190	-.08897	.01695	-.54355	-.00408	.53946	.44878
.27300	1.25940	1.02617	-.09724	.00844	-.56431	-.05284	.51146	.43335
.42178	1.25014	1.02462	-.09816	.00941	-.54360	-.04972	.49388	.43215
.57822	1.22358	1.00429	-.09155	.01834	=.48296	-.00893	.47403	.42320
.72700	1.17184	.94396	-.07803	.03825	-.36652	.10851	.47503	.43665
.85355	1.11720	.87464	-.06589	.05736	-.24678	.23658	.48336	.46373
.945	1.04749	.83074	-.05224	06820	-.09906	.31369	.4127	.40109
.99384	.90959	.79975	-.02532	.07497	.17469	.36625	.19156	.16456

ETA = .6904

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.29863	.82166	-.09257	.08833	-.65261	.32623	.97884	.82625
.05450	1.26352	.90213	-.08390	.05713	-.57145	.18593	.75738	.62742
.14645	1.26691	.99645	-.09062	.01959	-.58016	.00670	.58686	.48725
.27300	1.27319	1.03076	-.09795	.00566	-.59563	-.06215	.53348	.44971
.42178	1.26091	1.02974	-.09750	.00364	-.56777	-.06005	.50771	.44454
.57822	1.21560	.99304	-.08386	.01597	-.46394	.01364	.47758	.43433
.72700	1.16759	.94199	-.06807	.03168	-.35588	.11278	.46866	.43619
.85355	1.11335	.87654	-.05222	.04660	-.23704	.23429	.47134	.45653
.94550	1.04933	.83597	-.03681	.05365	-.10150	.30637	.40787	.39927
.99384	.91703	.80982	-.00950	.05788	.16125	.35142	.19017	.16591

ETA = .6131

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.29248	.83298	-.08958	.08663	-.63811	.30676	.94487	.80916
.05450	1.26452	.89637	-.08344	.06208	-.57364	.19602	.76966	.65229
.14645	1.27794	.99361	-.09135	.02152	-.60530	.01229	.61759	.51201
.27300	1.28176	1.03901	-.09479	.00237	-.61453	-.07898	.53555	.45061
.42178	1.26935	1.03962	-.09235	-.00088	-.58508	-.08022	.50576	.44283
.57822	1.20418	.98520	-.07290	.01417	-.43701	.02925	.46626	.43113
.72700	1.15838	.94528	-.05763	.02423	-.33457	.10687	.44144	.41396
.85355	1.10274	.88533	-.04037	.03609	-.21344	.21908	.43252	.42026
.94550	1.04511	.84706	-.02472	.04120	-.09209	.28796	.38005	.37188
.99384	.92108	.82132	.00174	.04475	.15368	.33295	.17927	.15715

ETA = .5357

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.29062	.83628	-.08819	.08766	-.63365	.30076	.93441	.79333
.05450	1.26208	.90336	-.08151	.06098	-.55783	.18316	.75099	.63156
.14645	1.27304	.99885	-.08640	.01914	-.59340	.00193	.59533	.49690
.27300	1.27985	1.04368	-.08855	-.00043	-.60918	-.08855	.52063	.43833
.42178	1.26976	1.04622	-.08439	-.00355	-.58565	-.09378	.49188	.43034
.57822	1.20298	.99191	-.06353	.01066	-.43317	.01602	.44920	.41504
.72700	1.14991	.95085	-.04756	.01856	-.31519	.09637	.41156	.38549
.85355	1.08928	.89817	-.02950	.02736	-.18426	.19591	.38017	.36846
.94550	1.03093	.86447	-.01307	.03045	-.06263	.25752	.32016	.31087
.99384	.91523	.83221	.01204	.03499	.16459	.31474	.15014	.12971

ETA = .4583

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.28704	.83952	-.08526	.08801	-.62502	.29497	.91999	.77203
.05450	1.25716	.91332	-.07778	.05813	-.55620	.16486	.72106	.59832
.14645	1.26346	1.00602	-.08011	.01696	-.57077	-.01235	.55842	.46698
.27300	1.27315	1.04765	-.08224	-.00159	-.59302	-.09671	.49631	.41829
.42178	1.26563	1.05168	-.07743	-.00515	-.57529	-.10506	.47024	.41112
.57822	1.20289	1.00185	-.05669	.00748	-.43222	-.00376	.42846	.39361
.72700	1.14278	.95803	-.03876	.01554	-.29903	.08254	.38157	.35514
.85[illegible]	1.07673	.91186	-.01960	02226	-.15745	.17058	.3280	.31579
.945[illegible]	1.01601	.88224	-.00282	.02417	-.03220	.22551	.25770	.24728
.99384	.90759	.84209	.02068	.03019	.17865	.29762	.11897	.09851

ETA = .3809

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.28199	.84291	-.08439	.08925	-.61340	.28875	.90215	.74887
.05450	1.25070	.92271	-.07640	.05683	-.54144	.14729	.68873	.56374
.14645	1.25206	1.01287	-.07577	.01621	-.54441	-.02610	.51831	.43172
.27300	1.26355	1.05152	-.07656	-.00175	-.57046	-.10469	.46577	.39289
.42178	1.25838	1.05687	-.07033	-.00597	-.55795	-.11578	.44217	.39670
.57822	1.20077	1.01140	-.04935	.00517	-.42680	-.02291	.40389	.36922
.72700	1.13603	.96520	-.02934	.01385	-.28387	.06861	.35248	.32561
.85355	1.06596	.92464	-.00958	.01956	-.13469	.14655	.28125	.26833
.94550	1.00326	.89866	.00673	.02131	-.00657	.19530	.20187	.19064
.99384	.90102	.85128	.02813	.02922	.19055	.28132	.09076	.07161

ETA = .3035

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.25630	.80474	-.07230	.10286	-.55351	.35246	.90596	.78124
.05450	1.23544	.91838	-.06866	.05957	-.50612	.15516	.66127	.53817
.14645	1.23455	1.02103	-.06741	.01591	-.50398	-.04260	.46138	.37983
.27300	1.24821	1.05650	-.06846	-.00127	-.53480	-.11499	.41981	.35206
.42178	1.24579	1.06240	-.06079	-.00637	-.52845	-.12725	.40120	.34991
.57822	1.19333	1.02147	-.03853	.00337	-.40947	-.04324	.36622	.33369
.72700	1.12734	.97255	-.01648	.01302	-.26462	.05424	.31886	.29433
.85355	1.05606	.93319	.00373	.01858	-.11409	.13031	.24440	.23435
.94550	.99053	.90685	.02017	.02101	.01847	.18003	.16156	.15399
.99384	.88840	.85092	.04174	.03179	.21297	.28180	.06883	.05420

ETA = .2261

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.26088	.80775	-.06869	.10650	-.56339	.34649	.90988	.79364
.05450	1.24096	.92868	-.07017	.05795	-.51871	.13582	.65453	.52973
.14645	1.22337	1.02917	-.06423	.01846	-.47860	-.05922	.41938	.34600
.27300	1.22748	1.06248	-.05713	.00358	-.48699	-.12739	.35960	.30356
.42178	1.22405	1.06786	-.04334	-.00488	-.47806	-.13858	.33947	.29869
.57822	1.17936	1.03018	-.01891	.00211	-.37766	-.06094	.31672	.29051
.72700	1.11811	.98297	.00212	.01180	-.24458	.03373	.27831	.25921
.85355	1.05105	.94778	.01803	.01491	-.10405	.10242	.20646	.19974
.94550	.98826	.92336	.03017	.01321	.02248	.14920	.12673	.12188
.99384	.88755	.86055	.04781	.02117	.21397	.26509	.05112	.04023

ETA = .1487

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.27812	.78587	-.14876	.03433	-.61793	.39450	1.01243	.88443
.05450	1.20998	.88335	-.11505	.00681	-.45715	.22401	.68116	.57329
.14645	1.22853	1.05422	-.08252	-.01921	-.49256	-.11063	.38193	.30832
.27300	1.22292	1.10840	-.04820	-.00880	-.47594	-.22397	.25191	.20413
.42178	1.21653	1.10985	-.05903	-.02447	-.46275	-.22754	.23521	.19725
.57822	1.14060	1.01667	-.03644	.00094	-.29415	-.03351	.26064	.23919
.727	1.10095	.97140	-.01122	02431	-.20819	.05607	.2642	.24890
.85355	1.04023	.94585	-.01337	.01009	-.08164	.10627	.18791	.18260
.94550	.99741	.95110	.00756	.01852	.00512	.09588	.09076	.08375
.99384	.88653	.87104	.00469	.00862	.21820	.24650	.02830	.01850

** WING LOADS **

	ETA	CNC/CA	CDC/CA	CMC/CA	X/C CP
1	.14870	.38438	.00436	-.17316	.45050
2	.22610	.41840	.00697	-.18961	.45319
3	.30350	.45034	.00590	-.19738	.43830
4	.38089	.46572	.00464	-.19728	.42361
5	.45828	.47267	.00600	-.19266	.40760
6	.53567	.47330	.00733	-.18495	.39076
7	.61306	.46456	.00781	-.17328	.37299
8	.69045	.43885	.00509	-.15514	.35352
9	.76784	.39625	.00155	-.13140	.33161
10	.84522	.32678	-.00388	-.09849	.30140
11	.92261	.23825	-.00806	-.06423	.26959
12	.97098	.14797	-.00955	-.03693	.24960

** TOTAL WING LOADS *

WCL	WCD	WCM	X/C
.35435	.00267	-8.21076	23.17154

** TOTAL CONFIGURATION PARAMETE   *

CL	CD	CM	X/C
.37003	.00295	-8.30490	22.44391

** INDUCED DRAG =   .01174

SAMPLE OUTPUT
(CASE II)

** SUBSON INTERFERENCE PRESSURE PROGRAM * INPUT D/ **

1	5.974400	55.920000	9.918332	.600000	
5	26.398000	0.000000	1.000000	3.060000	1.000000
10	1.000000	-1.000000	0.000000	0.000000	3.000000
15	1.000000	1.000000	1.000000	1.000000	13.000000
20	6.000000	0.000000	1.250000	2.500000	4.000000
25	5.500000	26.398000	0.000000	0.000000	0.000000
50	33.000000	0.000000	20.000000	32.500000	40.000000
55	45.000000	50.000000	57.500000	70.000000	90.000000
60	110.000000	122.500000	130.000000	135.000000	140.000000
65	147.500000	160.000000	180.000000	200.000000	212.500000
70	220.000000	225.000000	230.000000	237.500000	250.000000
75	270.000000	290.000000	302.500000	310.000000	315.000000
80	320.000000	327.500000	340.000000	360.000000	0.000000
85	0.000000	0.000000	8.818000	12.960000	3.375100
90	1.000000	1.000000	1.000000	1.000000	1.000000
95	1.000000	1.000000	1.000000	1.000000	1.000000
100	1.000000	1.000000	1.000000	1.000000	1.000000
105	1.000000	1.000000	1.000000	1.000000	1.000000
110	1.000000	1.000000	1.000000	1.000000	1.000000
115	1.000000	1.000000	1.000000	1.000000	1.000000
120	1.000000	1.000000	1.000000	1.000000	1.009300
125	1.026900	1.044300	1.060000	1.044300	1.026900
130	1.009300	1.000000	1.009300	1.026900	1.044300
135	1.060000	1.044300	1.026900	1.009300	1.000000
140	1.009300	1.026900	1.044300	1.060000	1.044300
145	1.026900	1.009300	1.000000	1.009300	1.026900
150	1.044300	1.060000	1.044300	1.026900	1.009300
155	1.000000	1.000000	1.034900	1.100900	1.165900
160	1.225000	1.165900	1.100900	1.034900	1.000000
165	1.034900	1.100900	1.165900	1.225000	1.165900
170	1.100900	1.034900	1.000000	1.034900	1.100900
175	1.165900	1.225000	1.165900	1.100900	1.034900
180	1.000000	1.034900	1.100900	1.165900	1.225000
185	1.165900	1.100900	1.034900	1.000000	1.000000
190	1.058200	1.168200	1.276400	1.374900	1.276400
195	1.168200	1.058200	1.000000	1.058200	1.168200
200	1.276400	1.374900	1.276400	1.168200	1.058200
205	1.000000	1.058200	1.168200	1.276400	1.374900
210	1.276400	1.168200	1.058200	1.000000	1.058200
215	1.168200	1.276400	1.374900	1.276400	1.168200
220	1.058200	1.000000	1.000000	1.064200	1.185700

225	1.305400	1.414200	1.305400	1.185700	1.064200
230	1.000000	1.064200	1.185700	1.305400	1.414200
235	1.305400	1.185700	1.064200	1.000000	1.064200
240	1.185700	1.305400	1.414200	1.305400	1.185700
245	1.064200	1.000000	1.064200	1.185700	1.305400
250	1.414200	1.305400	1.185700	1.064200	1.000000
255	1.000000	1.064200	1.185700	1.305400	1.414200
260	1.305400	1.185700	1.064200	1.000000	1.064200
265	1.185700	1.305400	1.414200	1.305400	1.[illegible]700
270	1.064200	1.000000	1.064200	1.185700	1.305400
275	1.414200	1.305400	1.185700	1.064200	1.000000
280	1.064200	1.185700	1.305400	1.414200	1.305400
285	1.185700	1.064200	1.000000	0.000000	0.000000
565	34.000000	0.000000	.003000	.007500	.014000
570	.023000	.036000	.060000	.100000	.160000
575	.220000	.280000	.340000	.400000	.600000
580	.900000	1.200000	1.600000	2.000000	3.000000
585	4.400000	6.000000	13.360000	14.500000	16.500000
590	18.500000	20.000000	22.000000	24.000000	25.000000
595	26.000000	26.200000	26.300000	26.370000	26.398000
600	0.000000	.040000	.100000	.164000	.230000
605	.315000	.435000	.575000	.728000	.850000
610	.960000	1.060000	1.139000	1.370000	1.680000
615	1.949000	2.225000	2.440000	2.785000	3.020000
620	3.065000	3.065000	3.065000	3.065000	3.065000
625	3.065000	3.065000	3.065000	3.065000	3.065000
630	3.065000	3.065000	3.065000	3.065000	0.000000
635	0.000000	.030000	.075000	.120000	.170000
640	.225000	.310000	.410000	.520000	.615000
645	.695000	.760000	.830000	1.020000	1.245000
650	1.420000	1.575000	1.660000	1.690000	1.690000
655	1.690000	1.690000	1.645000	1.495000	1.275000
660	1.085000	.805000	.505000	.355000	.186000
665	.155000	.105000	.060000	0.000000	0.000000
705	20.000000	0.000000	.000300	.001000	.002000
710	.003500	.005000	.007500	.012500	.025000
715	.050000	.100000	.200000	.300000	.400000
720	.500000	.600000	.700000	.800000	.900000
725	1.000000	0.000000	0.000000	0.000000	0.000000
730	5.000000	0.000000	.326200	.598800	.749500
735	1.000000	0.000000	0.000000	0.000000	0.000000
750	0.000000	.000283	.000730	.001095	.001404

755	.001857	.002324	.002881	.004212	.005075
760	.006350	.008400	.009450	.009600	.008950
765	.007700	.006050	.004100	.001950	0.000000
770	0.000000	.000257	.000447	.000671	.000957
775	.001210	.001680	.002420	.004010	.006480
780	.008810	.012520	.015830	.018510	.018650
785	.017370	.014550	.011380	.006330	0.000000
790	0.000000	.000094	.000225	.000358	.000552
795	.000720	.000978	.001481	.002680	.005110
800	.008720	.013990	.017290	.019590	.021500
805	.020810	.019010	.015370	.011160	0.000000
810	0.000000	.000185	.000485	.000802	.001207
815	.001582	.002200	.003260	.005160	.007910
820	.010580	.015460	.019510	.022320	.023800
825	.023340	.021440	.018210	.013040	0.000000
830	0.000000	.000650	.001194	.001730	.002460
835	.003080	.004000	.005450	.007940	.010730
840	.013140	.016340	.018710	.020340	.[illegible]110
845	.018690	.015750	.011600	.006560	0.000000
1200	0.000000	0.000000	3.705100	0.000000	0.000000
1205	24.000000	0.000000	2.000000	5.000000	0.000000
1210	.110000	17.640000	30.090000	.187400	18.820000
1215	0.000000	.264800	19.980000	0.000000	.342200
1220	21.150000	31.760000	1.000000	31.018000	36.418000
1270	20.000000	41.000000	12.000000	10.000000	37.000000
1275	12.000000	3.000000	10.000000	10.000000	19.000000
1280	6.000000	0.000000	10.000000	0.000000	0.000000
1295	1.000000	3.000000	5.000000	7.000000	9.000000
1300	11.000000	13.000000	15.000000	17.000000	19.000000
1460	0.000000	.091100	.362000	.804000	1.403000
1465	2.142000	3.000000	3.945000	4.960000	6.000000
1470	7.000000	8.000000	9.000000	10.000000	11.000000
1475	12.000000	13.000000	14.000000	15.000000	16.000000
1480	17.000000	18.000000	18.877000	19.745000	20.595000
1485	21.420000	22.200000	22.940000	23.625000	24.240000
1490	24.800000	25.280000	25.660000	25.990000	26.210000
1495	26.340000	26.398000	0.000000	0.000000	0.000000
1610	0.000000	26.566000	45.000000	90.000000	135.000000
1615	153.434000	180.000000	206.566000	225.000000	270.000000
1620	315.000000	333.434000	0.000000	0.000000	0.000000
1660	2.000000	3.000000	5.000000	7.000000	9.000000
1665	11.000000	13.000000	15.000000	17.000000	19.000000
1690	1.000000	2.000000	3.000000	4.000000	5.000000

1695	6.000000	7.000000	8.000000	9.000000	10.000000
1700	11.000000	12.000000	0.000000	0.000000	0.000000
1720	2.000000	3.000000	4.000000	5.000000	6.000000
1725	7.000000	8.000000	10.000000	12.000000	14.000000
1730	16.000000	18.000000	20.000000	22.000000	24.000000
1735	26.000000	28.000000	30.000000	32.000000	0.000000
1870	.025000	.050000	.100000	.200000	.300000
1875	.400000	.500000	.600000	.700000	.800000
1880	.900000	.950000	0.000000	0.000000	0.000000
1900	.148700	.226100	.303500	.380894	.458282
1905	.535671	.613059	.690447	.767835	.845224
1910	.922612	.970979	0.000000	0.000000	0.000000
1970	.052360	.049026	.049393	.037769	-.008132
1985	.394900	.175000	-.140000	-.400000	-.775000
2000	20.000000	0.000000	.000300	.001000	.002000
2005	.003500	.005000	.007500	.012500	.025000
2010	.050000	.100000	.200000	.300000	.400000
2015	.500000	.600000	.700000	.800000	.900000
2020	1.000000	0.000000	0.000000	0.000000	0.000000
2025	5.000000	0.000000	.326200	.598800	.749500
2030	1.000000	0.000000	0.000000	0.000000	0.000000
2050	0.000000	.003250	.005150	.007150	.009250
2055	.010875	.013150	.016675	.021960	.030250
2060	.039400	.047550	.050000	.049090	.044860
2065	.037670	.028370	.018100	.008730	0.000000
2070	0.000000	.002730	.004410	.005797	.[illegible]400
2075	.008760	.010466	.013160	.017500	.024060
2080	.033410	.045310	.051250	.052490	.048790
2085	.041210	.031750	.020500	.010000	0.000000
2090	0.000000	.002650	.004290	.005638	.007190
2095	.008520	.010179	.012800	.017020	.023300
2100	.033080	.045100	.051010	.051680	.046790
2105	.038640	.030320	.019600	.009300	0.000000
2110	0.000000	.002490	.004030	.005300	.006760
2115	.008010	.009568	.012030	.016000	.022300
2120	.031200	.042180	.047700	.049000	.046980
2125	.041470	.032620	.021660	.010200	0.000000
2130	0.000000	.002450	.003950	.005200	.006640
2135	.007860	.009388	.011810	.015700	.022650
2140	.031400	.041300	.046560	.048670	.048060
2145	.044050	.036620	.026310	.013800	0.000000
3380	1000000.000000	1000000.000000	1000000.000000	1000000.000000	1000000.000000
3385	1000000.000000	1000000.000000	1000000.000000	0.000000	0.000000

4000	2.000000	0.000000	2.000000	0.000000	0.000000
4005	65.000000	-1.940000	1.000000	1.000000	0.000000
4020	2.000000	0.000000	65.000000	0.000000	0.000000
4050	2.000000	0.000000	180.000000	0.000000	0.000000
4090	3.059000	3.059000	3.059000	3.059000	0.000000
4565	24.000000	0.000000	.764000	3.059000	6.105000
4570	9.160000	12.210000	15.290000	18.340000	21.400000
4575	24.450000	27.500000	30.590000	33.600000	36.690000
4580	39.750000	42.760000	45.850000	48.900000	51.950000
4585	55.000000	58.050000	61.100000	64.150000	65.000000
4600	0.000000	.675000	1.225000	1.690000	1.900000
4605	1.990000	2.000000	2.000000	2.000000	2.000000
4610	2.000000	2.000000	2.000000	2.000000	1.990000
4615	1.980000	1.940000	1.850000	1.680000	1.440000
4620	1.190000	.790000	.500000	0.000000	0.000000
4635	0.000000	.340000	.850000	1.280000	1.450000
4640	1.525000	1.555000	1.555000	1.555000	1.555000
4645	1.555000	1.555000	1.545000	1.500000	1.430000
4650	1.345000	1.215000	1.070000	.900000	.720000
4655	.545000	.355000	.175000	.120000	0.000000
4670	0.000000	.075000	.380000	.650000	.440000
4675	.150000	-.155000	-.440000	-.720000	-.990000
4680	-1.250000	-1.500000	-1.720000	-1.850000	-1.900000
4685	-1.875000	-1.780000	-1.630000	-1.400000	-1.125000
4690	-.780000	-.440000	-.100000	0.000000	0.000000
4705	121.000000	6.000000	3.000000	40.000000	20.000000
4710	0.000000	.006155	.024470	.054495	.095490
4715	.146445	.206105	.273005	.345490	.421785
4720	.500000	.578215	.654510	.726995	.793895
4725	.853555	.904510	.945505	.975530	.993845
4735	0.000000	.002831	.025470	.070759	.138639
4740	.229073	.341996	.477331	.634984	.814843
4745	1.016784	1.240667	1.486335	1.753616	2.042325
4750	2.352261	2.683207	3.034933	3.407194	3.[illegible]730
4755	4.212267	4.644519	5.096184	5.566947	6.056480
4760	6.564442	7.090478	7.634223	8.195298	8.773311
4765	9.367859	9.978529	10.604894	11.246518	11.902954
4770	12.573745	13.258422	13.956509	14.667520	15.390958
4775	16.126320	16.873092	17.630756	18.398782	19.176636
4780	19.963774	20.759650	21.563707	22.375387	23.194122

4785	24.019342	24.850473	25.686935	26.528146	27.373518
4790	28.222462	29.074389	29.928701	30.784807	31.642107
4795	32.500006	33.357904	34.215205	35.071310	35.925623
4800	36.777549	37.626494	38.471866	39.313076	40.149538
4805	40.980668	41.805880	42.624624	43.436303	44.240361
4810	45.036236	45.823375	46.601228	47.369254	48.126917
4815	48.873690	49.609052	50.332490	51.043500	51.741597
4820	52.426264	53.097055	53.753491	54.395115	55.021479
4825	55.632149	56.226697	56.804710	57.365784	57.909529
4830	58.435565	58.943527	59.433060	59.903822	60.355487
4835	60.787738	61.200276	61.592811	61.965072	62.316797
4840	62.647743	62.957679	63.246388	63.513669	63.759336
4845	63.983218	64.185159	64.365019	64.522670	64.658005
4850	64.770929	64.861362	64.929242	64.974521	64.997169
4855	65.000000	0.000000	0.000000	0.000000	0.000000
4905	2.000000	5.000000	8.000000	11.000000	13.000000
4910	16.000000	19.000000	22.000000	25.000000	28.000000
4915	31.000000	34.000000	37.000000	40.000000	43.000000
4920	46.000000	49.000000	52.000000	55.000000	58.000000
4925	61.000000	64.000000	67.000000	70.000000	73.000000
4930	76.000000	79.000000	82.000000	85.000000	88.000000
4935	91.000000	94.000000	97.000000	100.000000	103.000000
4940	106.000000	109.000000	112.000000	115.000000	118.000000

MATRIX- A

.79811	.79370	.77306	.61311	.55689	.51979	.47101	.39087	.21222	.08175
.03170	-.00332	-.02948	-.04753	-.04447	-.04300	-.04171	-.04026	-.03801	-.03547
-.03904	-.04025	-.04039	-.03230	-.04362	-.04843	-.04764	-.04893	-.04854	-.04426
-.02262	-.02516	-.04276	-.03910	-.02976	-.02460	-.02731	-.06400	-.10873	-.29989
.84910	.84484	.82404	.59482	.53853	.50749	.46091	.38125	.22199	.09885
.04792	.00743	-.02544	-.03370	.02222	.02712	-.02187	-.06539	-.08871	-.06764
-.03596	-.02717	-.02570	-.01679	-.02445	-.03299	-.04016	-.05253	-.06328	-.06305
-.04285	-.04484	-.09237	-.08192	-.05349	-.03237	-.02891	-.30096	-.53484	-.72640
.91215	.90871	.89125	.56934	.50267	.47726	.43348	.35367	.23569	.12992
.07995	.03238	-.01035	-.01466	-.00889	.00167	-.00040	-.01232	-.02282	-.02357
-.01693	-.00752	-.00919	-.00340	-.00999	-.02775	-.05105	-.08888	-.12361	-.13559
-.12507	-.12955	-.23706	-.21399	-.15099	-.10094	-.08947	-.69375	-.79909	-.82996
.89552	.89153	.87122	.49543	.41962	.40167	.36823	.29382	.23064	.15192
.10979	.06411	.02178	.01204	.01067	.01243	.01602	.01830	.01769	.01453
.01000	-.00142	-.01608	-.02312	-.04202	-.07179	-.10594	-.16112	-.20455	-.22570
-.23408	-.24145	-.34253	-.32141	-.26559	-.22502	-.22370	-.75035	-.82188	-.83410
.79708	.79123	.76251	.44308	.35994	.33502	.30753	.23837	.21016	.15215
.11994	.08407	.05327	.03271	.03040	.03085	.03318	.03609	.03673	.03315
.02132	-.01592	-.04329	-.06282	-.09645	-.12819	-.15473	-.19882	-.22360	-.23800
-.24885	-.25067	-.29314	-.28377	-.25900	-.24192	-.24164	-.48231	-.64722	-.74931
.72289	.71645	.68606	.44471	.35247	.31489	.28509	.21568	.19787	.14787
.12089	.09181	.06973	.04177	.03954	.03980	.04121	.04321	.04333	.03917
.02284	-.02967	-.06316	-.08825	-.12908	-.15915	-.17888	-.21456	-.22871	-.23947
-.25053	-.25099	-.26541	-.26210	-.25366	-.24913	-.25190	-.29298	-.33086	-.43741
-.09137	-.00768	.01343	.02097	.03700	.05376	.07061	.08023	.10138	.10548
-.07853	.00030	.01720	.02504	.04090	.05927	.07573	.08691	.12058	.12706
-.05938	.01284	.02402	.03228	.04790	.06848	.08438	.09768	.14839	.15811
-.04155	.02519	.03161	.04022	.05563	.07804	.09350	.10855	.17340	.18582
-.01954	.02265	.03597	.04882	.06063	.08465	.10171	.11212	.17612	.18845
.00159	.01705	.03941	.05702	.06477	.09018	.10917	.11386	.17343	.18503
.00035	.02236	.04355	.05788	.06867	.09239	.10931	.11673	.16206	.17097
-.00494	.02907	.04733	.05722	.07206	.09368	.10801	.11944	.15012	.15625
-.00961	.03497	.05065	.05664	.07503	.09480	.10686	.12182	.13959	.14327
-.00980	.04076	.05409	.05810	.07783	.09543	.10545	.12140	.12996	.13187
-.00091	.04726	.05832	.06399	.08074	.09520	.10340	.11531	.12079	.12204
.00717	.05318	.06219	.06935	.08342	.09505	.10162	.10990	.11260	.11327

** SUBSONIC INTERFERENCE PRESS PROGRAM ** FUSELAGE OUTPUT **

NO.	XQ	YQ	ZQ	VT/VFS	VM/VFS	CP
1	.01416	.01075	.02101	.01730	-.22446	1.03322
2	.18386	.13806	.27105	.02319	.05536	1.08898
3	.55616	.40163	.80035	.04379	.45726	.84662
4	1.12873	.64054	1.44892	.06030	.66579	.58116
5	1.61998	.73148	1.85440	.05555	.75632	.44140
6	2.51773	.87940	2.51172	.04649	.85456	.27406
7	3.60346	1.04814	3.22736	.03526	.98124	.03603
8	4.87035	1.20960	3.91875	.02237	1.14361	-.29988
9	6.31046	1.34877	4.46039	.00796	1.31224	-.67632
10	7.91476	1.44525	4.64878	-.00257	1.22835	-.48597
11	9.67319	1.51603	4.71361	-.00945	1.16894	-.35458
12	11.57474	1.56003	4.66623	-.01240	1.12895	-.26796
13	13.60747	1.57496	4.55352	-.01172	1.09088	-.20594
14	15.75864	1.57743	4.37770	-.03106	1.07674	-.15803
15	18.01477	1.57743	4.16729	-.09571	1.08353	-.18020
16	20.36171	1.57743	3.95139	-.02120	1.13692	-.28539
17	22.78475	1.57743	3.73350	.00518	1.15596	-.32622
18	25.26870	1.57743	3.51643	.02759	1.13078	-.27248
19	27.79799	1.57743	3.30254	.03887	1.08691	-.17989
20	30.35675	1.57743	3.09507	.02936	1.04219	-.08634
21	32.92895	1.57743	2.88399	-.08907	1.02316	-.05452
22	35.49847	1.57743	2.66640	-.10340	1.01922	-.04928
23	38.04918	1.57396	2.44962	-.10743	1.01631	-.04426
24	40.56510	1.56742	2.24717	-.10930	1.02214	-.05642
25	43.03046	1.55952	2.05933	-.11244	1.03492	-.08307
26	45.42981	1.53614	1.83545	-.11452	1.02703	-.06749
27	47.74809	1.49156	1.62334	-.11489	1.02412	-.06167
28	49.97077	1.41846	1.42309	-.11674	1.02016	-.05408
29	52.08393	1.31667	1.23727	-.12210	1.01374	-.04242
30	54.07430	1.19431	1.06513	-.13306	.99841	-.01450
31	55.92942	1.07682	.94126	-.14847	.99863	-.01927
32	57.63766	.96726	.84856	-.16595	1.01691	-.06130
33	59.18829	.82371	.74342	-.19039	1.01000	-.05606
34	60.57161	.67392	.64162	-.24198	.98721	-.03305
35	61.77894	.56720	.56115	-.31141	.98219	-.06134
36	62.80271	.49824	.49963	-.37107	1.01161	-.15873
37	63.63650	.45203	.44985	-.40741	1.04697	-.25599
38	64.27509	.34313	.40619	-.48527	.98079	-.21104
39	64.71447	.14199	.37427	-1.04886	.97050	-.96231
40	64.95188	.02429	.35678	-2.69567	1.05677	-3.68916

** SUBSONIC INTERFERENCE PRESS PROGRAM ** FUSELAGE OUTPUT **

NO.	XO	YO	ZO	VT/VFS	VM/VFS	CP
1	.01416	.02936	.01574	.05765	-.14890	1.06300
2	.18386	.37718	.20307	.07172	.09631	1.07613
3	.55616	1.09727	.60027	.12381	.47710	.81004
4	1.12873	1.74999	1.09246	.17259	.69354	.51113
5	1.61998	1.99843	1.40728	.15607	.77493	.38796
6	2.51773	2.40255	1.92234	.12878	.85703	.25454
7	3.60346	2.86357	2.48075	.09660	.96873	.05248
8	4.87035	3.30468	3.02172	.05959	1.11633	-.24417
9	6.31046	3.68492	3.43744	.01629	1.26702	-.57331
10	7.91476	3.94849	3.54787	-.01239	1.19769	-.41788
11	9.67319	4.14187	3.55609	-.02603	1.14837	-.31035
12	11.57474	4.26207	3.47261	-.03336	1.12668	-.26400
13	13.60747	4.30287	3.33655	-.04149	1.09458	-.19626
14	15.75864	4.30962	3.15125	-.06256	.98386	.02817
15	18.01477	4.30962	2.94084	-.02017	.76319	.43303
16	20.36171	4.30962	2.72494	.08118	.92307	.14316
17	22.78475	4.30962	2.50705	.05298	.99076	.01561
18	25.26870	4.30962	2.28997	.03828	.98823	.02198
19	27.79799	4.30962	2.07609	.04351	.95706	.08275
20	30.35675	4.30962	1.86875	.05399	.93982	.11499
21	32.92895	4.30962	1.66244	-.05824	.98196	.03247
22	35.49847	4.30962	1.46715	-.07590	1.00594	-.01764
23	38.04918	4.30014	1.28923	-.08128	1.02082	-.04847
24	40.56510	4.28228	1.13605	-.08235	1.02516	-.05743
25	43.03046	4.26070	1.00680	-.08266	1.03759	-.08279
26	45.42981	4.19682	.86242	-.07672	1.03817	-.08305
27	47.74809	4.07503	.73442	-.06488	1.03870	-.08248
28	49.97077	3.87529	.62487	-.05230	1.03062	-.06453
29	52.08393	3.59720	.53364	-.04285	1.01910	-.03823
30	54.07430	3.26292	.45433	-.03950	.99482	.00878
31	55.92942	2.94194	.41562	-.04153	.99013	.01796
32	57.63766	2.64261	.39974	-.04916	1.01412	-.03078
33	59.18829	2.25041	.36922	-.04857	.98491	.02765
34	60.57161	1.84118	.33602	-.09265	.94148	.10602
35	61.77894	1.54963	.31317	-.17932	.93097	.08494
36	62.80271	1.36121	.29872	-.25378	.99561	-.05536
37	63.63650	1.23497	.28718	-.29924	1.05656	-.20209
38	64.27509	.93744	.27447	-.30195	.84658	.19548
39	64.71447	.38794	.26495	-.73506	.70594	-.03853
40	64.95188	.06636	.25966	-2.62397	.50524	-3.45360

** SUBSONIC INTERFERENCE PRESS PROGRAM ** FUSELAGE OUTPUT **

NO.	XO	YO	ZO	VT/VFS	VM/VFS	CP
1	.01416	.04011	.00660	.11173	-.09661	1.06736
2	.18386	.51524	.08534	.12584	.13079	1.05418
3	.55616	1.49889	.25372	.17248	.51500	.75091
4	1.12873	2.39053	.47506	.21298	.74799	.40941
5	1.61998	2.72991	.63286	.19104	.82055	.29787
6	2.51773	3.28195	.90150	.15420	.87391	.21660
7	3.60346	3.91171	1.18757	.11461	.95789	.06975
8	4.87035	4.51428	1.46802	.06954	1.07569	-.15959
9	6.31046	5.03369	1.66563	.01041	1.19378	-.40920
10	7.91476	5.39373	1.64103	-.02685	1.15048	-.33453
11	9.67319	5.65790	1.55120	-.04234	1.13281	-.27782
12	11.57474	5.82210	1.40520	-.05149	1.10897	-.22765
13	13.60747	5.87783	1.22870	-.06373	1.06946	-.14585
14	15.75864	5.88705	1.02698	-.06323	1.00762	-.01926
15	18.01477	5.88705	.81656	-.01625	.94189	.11372
16	20.36171	5.88705	.60066	.05020	.94108	.11298
17	22.78475	5.88705	.38278	.06621	.96587	.06307
18	25.26870	5.88705	.16570	.06263	.98035	.03510
19	27.79799	5.88705	-.04819	.06980	.97788	.03901
20	30.35675	5.88705	-.25529	.09392	.97257	.04547
21	32.92895	5.88705	-.45334	.11876	.98660	.01253
22	35.49847	5.88705	-.61000	.13558	1.00872	-.03578
23	38.04918	5.87411	-.72062	.14914	1.02303	-.06841
24	40.56510	5.84970	-.78846	.16339	1.03043	-.08778
25	43.03046	5.82022	-.81622	.17906	1.04573	-.12419
26	45.42981	5.73296	-.82293	.20010	1.05322	-.14731
27	47.74809	5.56659	-.80525	.22527	1.05738	-.16624
28	49.97077	5.29375	-.75770	.25356	1.04737	-.15895
29	52.08393	4.91387	-.68507	.27460	1.03471	-.14412
30	54.07430	4.45723	-.60361	.27709	1.01361	-.10320
31	55.92942	4.01876	-.49482	.29862	1.00375	-.09585
32	57.63766	3.60987	-.37762	.32761	1.02227	-.15028
33	59.18829	3.07411	-.27891	.37506	.97474	-.09005
34	60.57161	2.51510	-.19330	.40296	.90581	.01715
35	61.77894	2.11684	-.11634	.40603	.91050	.00612
36	62.80271	1.85945	-.04927	.40113	1.00175	-.16200
37	63.63650	1.68701	.00543	.36961	1.05475	-.24357
38	64.27509	1.28057	.04633	.41130	.79135	.20840
39	64.71447	.52993	.07558	.51032	.51014	.49030
40	64.95188	.09065	.09143	.42530	.25252	.80811

** SUBSONIC INTERFERENCE PRESS : PROGRAM ** FUSELAGE OUTPUT **

NO.	XO	YO	ZO	VT/VFS	VM/VFS	CP
1	.01416	.04011	-.00395	.10807	-.06579	1.07425
2	.18386	.51524	-.05060	.10568	.17089	1.04540
3	.55616	1.49889	-.14645	.10152	.56868	.70722
4	1.12873	2.39053	-.23785	.08076	.82027	.33000
5	1.61998	2.72991	-.26137	.09064	.87349	.23355
6	2.51773	3.28195	-.27726	.08312	.90029	.18558
7	3.60346	3.91171	-.30566	.06571	.96465	.06552
8	4.87035	4.51428	-.32604	.04064	1.04761	-.09826
9	6.31046	5.03369	-.38029	-.00301	1.12554	-.26051
10	7.91476	5.39373	-.56079	-.03313	1.12016	-.25003
11	9.67319	5.65790	-.76384	-.04774	1.11122	-.23208
12	11.57474	5.82210	-.98204	-.05658	1.09442	-.19735
13	13.60747	5.87783	-1.20523	-.06199	1.06176	-.12964
14	15.75864	5.88705	-1.42593	-.05451	1.02485	-.05303
15	18.01477	5.88705	-1.63634	-.03263	.99107	.01675
16	20.36171	5.88705	-1.85224	-.00211	.98281	.03418
17	22.78475	5.88705	-2.07013	.02263	.98417	.03098
18	25.26870	5.88705	-2.28720	.03870	.99068	.01707
19	27.79799	5.88705	-2.50109	.05259	.99374	.00973
20	30.35675	5.88705	-2.70793	.07433	1.00164	-.00879
21	32.92895	5.88705	-2.89644	.10523	1.01724	-.04567
22	35.49847	5.88705	-3.00850	.13995	1.03148	-.08291
23	38.04918	5.87411	-3.04140	.16960	1.04033	-.10995
24	40.56510	5.84970	-3.01070	.19851	1.04514	-.13017
25	43.03046	5.82022	-2.92127	.22822	1.05654	-.16582
26	45.42981	5.73296	-2.76900	.25757	1.05906	-.18479
27	47.74809	5.56659	-2.58310	.28493	1.05846	-.19790
28	49.97077	5.29375	-2.35415	.31416	1.03915	-.17568
29	52.08393	4.91387	-2.09232	.34092	1.01355	-.14167
30	54.07430	4.45723	-1.82522	.36634	.97654	-.08713
31	55.92942	4.01876	-1.54610	.38823	.95916	-.07026
32	57.63766	3.60987	-1.27525	.39184	.97840	-.10970
33	59.18829	3.07411	-1.02731	.37646	.94181	-.02866
34	60.57161	2.51510	-.80451	.39998	.86908	.08537
35	61.77894	2.11684	-.61229	.44410	.86864	.04845
36	62.80271	1.85945	-.45110	.46713	.94558	-.11121
37	63.63650	1.68701	-.31991	.45734	.97468	-.15689
38	64.27509	1.28057	-.21711	.31698	.74404	.35682
39	64.71447	.52993	-.14307	.16624	.46089	.81335
40	64.95188	.09065	-.10282	.09097	.19399	1.03885

** SUBSONIC INTERFERENCE PRESS PROGRAM ** FUSELAGE OUTPUT **

XO	YO	ZO	VT/VFS	VM/VFS	CP
.01416	.02936	-.01308	.05212	-.04932	1.08714
.18386	.37718	-.16834	.03799	.22126	1.03356
.55616	1.09727	-.49300	.00307	.64130	.62058
1.12873	1.74999	-.85525	-.03409	.88887	.21271
1.61998	1.99843	-1.03579	-.01139	.92356	.14886
2.51773	2.40255	-1.29810	.00524	.93638	.12454
3.60346	2.86357	-1.59884	.01047	.99238	.01508
4.87035	3.30468	-1.87974	.00653	1.04337	-.08797
6.31046	3.68492	-2.15210	-.01035	1.08222	-.16868
7.91476	3.94849	-2.46763	-.02481	1.09097	-.18758
9.67319	4.14187	-2.76873	-.03360	1.09147	-.18913
11.57474	4.26207	-3.04945	-.03998	1.08006	-.16560
13.60747	4.30287	-3.31308	-.04224	1.06033	-.12465
15.75864	4.30962	-3.55020	-.03490	1.03514	-.07225
18.01477	4.30962	-3.76062	-.02452	1.00938	-.01942
20.36171	4.30962	-3.97651	-.01159	1.00204	-.00422
22.78475	4.30962	-4.19440	.00128	.99889	.00222
25.26870	4.30962	-4.41148	.01261	1.00124	-.00263
27.79799	4.30962	-4.62537	.02328	1.00545	-.01146
30.35675	4.30962	-4.83198	.03836	1.02235	-.04647
32.92895	4.30962	-5.01222	.06127	1.04616	-.09735
35.49847	4.30962	-5.08566	.08928	1.05605	-.12186
38.04918	4.30014	-5.05125	.11343	1.05936	-.13348
40.56510	4.28228	-4.93522	.13581	1.05807	-.13624
43.03046	4.26070	-4.74430	.15712	1.06151	-.14943
45.42981	4.19682	-4.45434	.17284	1.04906	-.12887
47.74809	4.07503	-4.12276	.18196	1.03901	-.11151
49.97077	3.87529	-3.73672	.18972	1.01228	-.06037
52.08393	3.59720	-3.31104	.19623	.97914	.00279
54.07430	3.26292	-2.88316	.20666	.94273	.06898
55.92942	2.94194	-2.45653	.22220	.91575	.11316
57.63766	2.64261	-2.05262	.22563	.90770	.12660
59.18829	2.25041	-1.67544	.20060	.87468	.19812
60.57161	1.84118	-1.33383	.20493	.82897	.27749
61.77894	1.54963	-1.04180	.24059	.82148	.27378
62.80271	1.36121	-.79910	.25039	.85282	.21399
63.63650	1.23497	-.60167	.22103	.86261	.21095
64.27509	.93744	-.44526	.09909	.69518	.53045
64.71447	.38794	-.33243	-.10268	.49955	.79050
64.95188	.06636	-.27104	-.23249	.27260	.94218

** SUBSONIC INTERFERENCE PRESS PROGRAM ** FUSELAGE OUTPUT **

XQ	YQ	ZQ	VT/VFS	VM/VFS	CP
.01416	.01075	-.01835	.01496	-.04762	1.09030
.18386	.13806	-.23631	.00454	.27598	1.00320
.55616	.40163	-.69308	-.01799	.75327	.44934
1.12873	.64054	-1.21171	-.02830	.94191	.11314
1.61998	.73148	-1.48290	-.01774	.97045	.05821
2.51773	.87940	-1.88749	-.00792	.97377	.05194
3.60346	1.04814	-2.34545	-.00331	1.01592	-.03201
4.87035	1.20960	-2.77677	-.00262	1.05026	-.10211
6.31046	1.34877	-3.17506	-.00558	1.06968	-.14238
7.91476	1.44525	-3.56855	-.00884	1.07879	-.16147
9.67319	1.51603	-3.92625	-.01160	1.07985	-.16373
11.57474	1.56003	-4.24307	-.01395	1.07147	-.14627
13.60747	1.57496	-4.53005	-.01453	1.06206	-.12671
15.75864	1.57743	-4.77665	-.01172	1.04167	-.08456
18.01477	1.57743	-4.98707	-.00852	1.01733	-.03492
20.36171	1.57743	-5.20297	-.00487	1.01022	-.02054
22.78475	1.57743	-5.42085	-.00114	1.00624	-.01251
25.26870	1.57743	-5.63793	.00245	1.00776	-.01556
27.79799	1.57743	-5.85182	.00607	1.01289	-.02593
30.35675	1.57743	-6.05830	.01133	1.03557	-.07206
32.92895	1.57743	-6.23377	.01963	1.06592	-.13489
35.49847	1.57743	-6.28491	.02982	1.07408	-.15241
38.04918	1.57396	-6.21164	.03855	1.07428	-.15340
40.56510	1.56742	-6.04634	.04640	1.06932	-.14370
43.03046	1.55952	-5.79682	.05358	1.06790	-.14144
45.42981	1.53614	-5.42738	.05808	1.04427	-.09309
47.74809	1.49156	-5.01169	.05958	1.02814	-.06029
49.97077	1.41846	-4.53494	.06039	.99919	-.00202
52.08393	1.31667	-4.01466	.06096	.96497	.06549
54.07430	1.19431	-3.49397	.06300	.93311	.12675
55.92942	1.07682	-2.98218	.06695	.90428	.18066
57.63766	.96726	-2.50143	.06777	.88360	.21883
59.18829	.82371	-2.04964	.06027	.85520	.27138
60.57161	.67392	-1.63943	.06128	.82745	.32040
61.77894	.56720	-1.28978	.07303	.81616	.33839
62.80271	.49824	-1.00001	.07571	.82160	.32852
63.63650	.45203	-.76434	.06289	.81632	.33956
64.27509	.34313	-.57698	.01884	.73158	.48418
64.71447	.14199	-.44176	-.09426	.62389	.63519
64.95188	.02429	-.36816	-.31853	.54190	.63853

* FUSELAGE LOADS ** REF. FUSELAGE AR 677.89822

* CL= .04998 * CD= .00139 * CMXY= -.14537 * CMYZ= .00090 * CM= -.14446

X	CLW/WA	CDW/WA	X	CLW/WA	CDW/WA
.00050	.00252	.09104	.52243	-.11389	.00072
.00201	.00694	.35496	.54482	-.11262	.00632
.00453	-.01251	.66531	.56712	-.11027	.01110
.00804	-.05263	.84747	.58928	-.10531	.01502
.01254	-.10398	.64071	.61126	-.09680	.01708
.01802	-.16037	.34243	.63302	-.08592	.02125
.02447	-.15043	.26600	.65451	-.07288	.02919
.03188	-.12601	.23323	.67569	-.05715	.03256
.04024	-.09545	.19388	.69651	-.04039	.03395
.04952	-.06115	.10379	.71694	-.02657	.03525
.05970	-.00569	-.00698	.73693	-.01204	.03575
.07078	.09834	-.14414	.75645	.01217	.03278
.08271	.23912	-.24567	.77545	.03582	.02619
.09549	.41725	-.29016	.79389	.05902	.01580
.10908	.35366	-.21398	.81174	.07417	.00292
.12346	.26491	-.12642	.82897	.08191	-.01136
.13860	.19701	-.09381	.84553	.09489	-.02109
.15447	.15076	-.06538	.86140	.11113	-.02666
.17103	.12608	-.04121	.87654	.13228	-.02438
.18826	.10366	-.02121	.89092	.14716	-.02637
.20611	.08462	-.00631	.90451	.14540	-.03562
.22455	.04502	-.00056	.91729	.13699	-.04439
.24355	-.00870	-.00043	.92922	.12515	-.05286
.26307	-.07771	-.00534	.94030	.11794	-.04984
.28306	-.10276	-.00674	.95048	.11482	-.04058
.30349	.03072	.00195	.95976	.12074	-.02962
.32431	.12033	.00761	.96812	.13197	-.02101
.34549	.16829	.01038	.97553	.14890	-.01626
.36698	.16812	.01013	.98198	.14633	-.04276
.38874	.13759	.00811	.98746	.12533	-.09951
.41072	.08907	.00502	.99196	.11961	-.13067
.43288	.03110	.00175	.99547	.11814	-.14504
.45518	-.03680	-.00170	.99799	.08755	-.06709
.47757	-.08692	-.00404	.99950	.06515	-.00750
.50000	-.10823	-.00374			

** WING L[illegible] PRESSURE COEF.

** LIST OF X/C

* ETA ** LISTS OF CP AT ABOVE X/C

ETA	.0250	.0500	.1000	.2000	.3000	.4000	.5000	.6000	.7000	.8000	.9000	.9500
.149	1.5100	1.0413	.7272	.5460	.4878	.4582	.4367	.4132	.3755	.3080	.1963	.1209
.226	1.6777	1.1549	.8095	.6202	.5603	.5201	.4793	.4347	.3839	.3195	.2263	.1574
.303	1.8259	1.2568	.8783	.6675	.6012	.5585	.5159	.4696	.4181	.3557	.2643	.1915
.381	1.8125	1.3249	.9849	.7490	.6526	.5982	.5540	.5056	.4485	.3819	.2947	.2237
.458	1.8805	1.3906	1.0434	.7908	.6850	.6270	.5809	.5311	.4761	.4197	.3491	.2794
.536	1.9372	1.4473	1.0941	.8252	.7104	.6498	.6033	.5542	.5040	.4612	.4104	.3421
.613	1.9790	1.4861	1.1269	.8468	.7264	.6642	.6179	.5711	.5282	.5011	.4694	.4016
.690	2.0136	1.4765	1.1004	.8371	.7283	.6654	.6161	.5732	.5431	.5306	.5019	.4262
.768	1.9948	1.4347	1.0526	.8064	.7084	.6460	.5950	.5550	.5333	.5298	.5027	.4242
.845	1.8605	1.3519	.9974	.7525	.6497	.5885	.5413	.5017	.4723	.4533	.4150	.3455
.923	1.5947	1.1723	.8643	.6252	.5179	.4607	.4227	.3896	.3579	.3280	.2859	.2338
.971	1.1677	.7938	.5231	.3422	.2837	.2615	.2467	.2287	.2066	.1839	.1564	.1272

**WING NO .INEAR VELOCITY AND PRESSURE COEFFICIENT!

ETA = .9710

125

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.40235	.72910	-.10845	.13436	-.89520	.46892	1.36412	1.18118
.05450	1.33241	.87962	-.09305	.09090	-.73022	.22232	.95253	.75511
.14645	1.24429	1.00806	-.08684	.07261	-.52855	-.02141	.50715	.41186
.27300	1.19875	1.02922	-.11011	.07920	-.43127	-.06518	.36609	.29407
.42178	1.17831	1.02957	-.14457	.08362	-.39446	-.06661	.32785	.25819
.57822	1.15715	1.02005	-.17496	.08874	-.35747	-.04818	.30929	.23307
.72700	1.11089	.98463	-.19178	.10367	-.26432	.01978	.28410	.20045
.85355	1.05256	.94088	-.19589	.12512	-.14435	.09997	.24432	.17111
.94550	.98428	.89137	-.18938	.14868	-.00467	.18639	.19106	.13094
.99384	.86155	.80455	-.16851	.17525	.23410	.33142	.09732	.05175

ETA = .9226

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.51596	.64169	-.18230	.15227	-1.17940	.59437	1.77377	1.61022
.05450	1.41957	.78977	-.15249	.10223	-.94498	.37801	1.32299	1.12903
.14645	1.32894	.93310	-.13318	.06337	-.73007	.12673	.85681	.72537
.27300	1.28176	.98594	-.13194	.05961	-.62201	.02442	.64643	.54013
.42178	1.24893	1.00301	-.13941	.06482	-.54968	-.01023	.53945	.45145
.57822	1.21517	.99830	-.14756	.07439	-.47645	-.00213	.47432	.39672
.72700	1.15521	.96016	-.14710	.09363	-.34488	.06975	.41463	.34968
.85355	1.08482	.91103	-.14232	.11455	-.19362	.15914	.35276	.30978
.94550	1.00624	.86403	-.13364	.13186	-.03029	.24112	.27142	.24062
.99384	.87384	.79685	-.11179	.14952	.22845	.35337	.12492	.09514

ETA = .8452

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.60034	.58851	-.21680	.16442	-1.38863	.66276	2.05139	1.87933
.05450	1.46494	.74853	-.16996	.10927	-1.05590	.44448	1.50038	1.30058
.14645	1.36569	.90049	-.14039	.05859	-.81658	.18881	1.00539	.85083
.27300	1.32524	.95914	-.13375	.04633	-.72170	.07863	.80033	.67142
.42178	1.29152	.98168	-.13112	.04364	-.64399	.03450	.67849	.57746
.57822	1.25069	.97982	-.12674	.05102	-.55059	.03748	.58806	.50960
.72700	1.18461	.93749	-.11608	.07032	-.40138	.11739	.51877	.46644
.85355	1.11452	.88248	-.10593	.09094	-.24764	.21708	.46472	.44012
.94550	1.03522	.84089	-.09428	.10507	-.07999	.28909	.36907	.35513
.99384	.89570	.79686	-.06947	.11647	.19627	.36271	.16644	.14192

ETA = .7678

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.64879	.56349	-.23064	.16013	-1.50693	.69327	2.20019	2.01555
.05450	1.48726	.73084	-.17382	.11243	-1.10943	.47203	1.58146	1.37860
.14645	1.38472	.88790	-.14174	.05773	-.86108	.21224	1.07331	.90242
.27300	1.34825	.94890	-.13540	.03921	-.77507	.09891	.87399	.72930
.42178	1.31050	.97153	-.12866	.03484	-.68676	.05519	.74194	.63419
.57822	1.26409	.96865	-.11790	.04089	-.57887	.06038	.63925	.56244
.72700	1.19728	.92196	-.10214	.05026	-.42646	.14842	.57488	.53112
.85355	1.13219	.86199	-.08867	.07749	-.28223	.25669	.53893	.52478
.945	1.05530	.82424	-.07396	08760	-.11785	.32186	.4397	.43557
.99364	.91203	.79773	-.04622	.09380	.16857	.36630	.19774	.17576

ETA = .6904

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.65500	.56540	-.22847	.16803	-1.52078	.69126	2.21205	2.03345
.05450	1.49459	.72228	-.17286	.11627	-1.12643	.48457	1.61100	1.42231
.14645	1.40275	.88188	-.14335	.05876	-.90345	.22317	1.12662	.94305
.27300	1.36440	.95163	-.13539	.03465	-.81242	.09398	.90640	.75091
.42178	1.32354	.97463	-.12654	.02675	-.71617	.04960	.76576	.65403
.57822	1.25727	.95632	-.10800	.03557	-.56147	.08484	.64631	.58178
.72700	1.19453	.91857	-.08932	.04911	-.41812	.15597	.57408	.53817
.85355	1.12953	.86275	-.07164	.06265	-.27395	.25749	.53143	.52337
.94550	1.05789	.82878	-.05481	.06861	-.12079	.31706	.43786	.43733
.99384	.91967	.80765	-.02642	.07200	.15564	.35321	.19757	.17831

ETA = .6131

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.64131	.57823	-.22246	.16501	-1.48674	.67595	2.16268	1.99717
.05450	1.49104	.71966	-.17150	.11957	-1.11769	.48781	1.60550	1.43579
.14645	1.41327	.88007	-.14370	.05867	-.92811	.22651	1.15462	.96366
.27300	1.37325	.95940	-.13159	.02955	-.83210	.07924	.91133	.75063
.42178	1.33263	.98369	-.12044	.02021	-.73576	.03203	.76779	.65368
.57822	1.24558	.94854	-.09565	.03138	-.53290	.10017	.63307	.58135
.72700	1.18573	.92129	-.07712	.03891	-.39686	.15174	.54860	.51909
.85355	1.11941	.87098	-.05776	.04909	-.25056	.24417	.49473	.48982
.94550	1.05407	.83951	-.04053	.05290	-.11156	.30021	.41177	.41170
.99384	.92386	.81908	-.01296	.05546	.14825	.33571	.18746	.17016

ETA = .5357

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.63078	.58675	-.21877	.16301	-1.46084	.66559	2.12644	1.95527
.05450	1.48317	.72948	-.16831	.11606	-1.09831	.47327	1.57158	1.39746
.14645	1.40489	.88699	-.13796	.05408	-.90718	.21433	1.12152	.93654
.27300	1.36957	.96485	-.12472	.02450	-.82205	.06888	.89093	.73364
.42178	1.33239	.99038	-.11179	.01526	-.73348	.01894	.75242	.63939
.57822	1.24425	.95509	-.08539	.02540	-.52822	.08784	.61606	.56524
.72700	1.17723	.92673	-.06595	.03055	-.37671	.14201	.51873	.49147
.85355	1.10608	.88360	-.04563	.03749	-.22096	.22216	.44311	.43903
.94550	1.04004	.85678	-.02754	.03915	-.08184	.27074	.35258	.35142
.99384	.91810	.83003	-.00117	.04259	.15933	.31795	.15862	.14197

ETA = .4583

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.61544	.59723	-.21362	.15886	-1.42317	.65323	2.07640	1.89862
.05450	1.47072	.74320	-.16345	.10962	-1.06775	.45301	1.52076	1.34099
.14645	1.39008	.89650	-.13105	.04857	-.87112	.19734	1.06845	.89394
.27300	1.35980	.97019	-.11796	.02007	-.79801	.05863	.85664	.70664
.42178	1.32675	.99634	-.10439	.01053	-.71910	.00720	.72631	.61675
.57822	1.24380	.96493	-.07805	.01912	-.52613	.06897	.59510	.54253
.72700	1.16994	.93394	-.05661	.02438	-.35969	.12862	.48831	.46096
.853	1.09360	.89724	-.03517	02916	-.19373	.19753	.3912	.38662
.945[illegible]	1.02527	.87459	-.01674	.02953	-.05123	.23921	.29044	.28809
.99384	.91062	.84009	.00804	.03435	.17335	.30088	.12753	.11186

ETA = .3809

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.59504	.60966	-.21303	.15468	-1.37496	.63799	2.01294	1.83054
.05450	1.45390	.75787	-.16364	.10364	-1.02828	.43062	1.45890	1.27580
.14645	1.37152	.90646	-.12761	.04277	-.82721	.17932	1.00653	.84372
.27300	1.34579	.97561	-.11226	.01446	-.76449	.04818	.81267	.67244
.42178	1.31725	1.00195	-.09716	.00478	-.69601	-.00393	.69208	.58842
.57822	1.24107	.97429	-.07079	.01238	-.51903	.05083	.56986	.51684
.72700	1.16319	.94105	-.04754	.01850	-.34406	.11526	.45932	.43154
.85355	1.08322	.90996	-.02601	.02257	-.17133	.17413	.34546	.33978
.94550	1.01298	.89113	-.00849	.02299	-.02615	.20919	.23534	.23189
.99384	.90447	.84961	.01384	.02989	.18473	.28426	.09953	.08509

ETA = .3035

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.56220	.57197	-.20265	.15503	-1.29439	.68759	1.98198	1.84698
.05450	1.42341	.76053	-.15369	.09045	-.95425	.42903	1.38328	1.20003
.14645	1.33789	.92284	-.11575	.02666	-.74694	.14963	.89657	.74327
.27300	1.32288	.98238	-.10322	.00347	-.71002	.03503	.74504	.61476
.42178	1.30209	1.00639	-.08806	-.00359	-.65980	-.01282	.64698	.54950
.57822	1.23256	.98383	-.06062	.00388	-.49872	.03216	.53088	.48001
.72700	1.15500	.94792	-.03561	.01123	-.32530	.10226	.42756	.40281
.85355	1.07520	.91761	-.01366	.01473	-.15406	.16004	.31409	.31239
.94550	1.00172	.89924	.00431	.01506	-.00347	.19446	.19793	.19954
.99384	.89260	.85007	.02737	.02396	.20624	.28377	.07754	.05850

ETA = .2261

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.51268	.60637	-.19085	.12454	-1.17415	.65181	1.82596	1.69704
.05450	1.39916	.78736	-.15397	.05718	-.89771	.38975	1.28746	1.10283
.14645	1.31272	.93404	-.11533	.00286	-.68901	.12903	.81804	.68782
.27300	1.29463	.98391	-.09497	-.01094	-.64388	.03190	.67577	.57275
.42178	1.27725	1.00625	-.07306	-.01452	-.60104	-.01274	.58830	.51143
.57822	1.21847	.98962	-.04312	-.00668	-.46561	.02064	.48625	.44493
.72700	1.14659	.95867	-.01898	.00019	-.30621	.08155	.38776	.36829
.85355	1.07129	.93410	-.00110	-.00050	-.14571	.12892	.27462	.27472
.94550	1.00101	.91812	.01244	-.00539	-.00217	.15927	.16144	.16470
.99384	.89361	.86205	.03211	-.00044	.20406	.26286	.05880	.05359

ETA = .1487

X/C	U/V UP	U/V LOW	V/V UP	V/V LOW	CP UP	CP LOW	CP L-U	CP NETL
.02447	1.45007	.63881	-.25307	.04164	-1.04933	.62221	1.67153	1.52725
.05450	1.32055	.77687	-.19670	.00233	-.72897	.41081	1.13978	.99437
.14645	1.29960	.96380	-.14996	-.02768	-.66706	.07077	.73783	.61251
.27300	1.28580	1.01553	-.11044	-.01605	-.62656	-.03148	.59508	.49916
.42178	1.27331	1.03138	-.10958	-.03062	-.59804	-.06430	.53373	.45320
.57822	1.18970	.97403	-.07948	-.01453	-.40594	.05128	.45723	.41900
.727	1.14054	.95510	-.05275	.00172	-.29540	.08847	.3838	.36097
.85355	1.07190	.94282	-.05641	-.02406	-.15008	.11161	.26169	.25436
.94550	1.02281	.95564	-.03684	-.02089	-.04729	.08698	.13427	.12931
.99384	.90453	.88232	-.03878	-.03319	.18326	.22481	.04155	.03419

** WING LOADS **

	ETA	CNC/CA	CDC/CA	CMC/CA	X/C CP
1	.14870	.70079	.05003	-.30121	.42981
2	.22610	.72153	.04595	-.29418	.40772
3	.30350	.75959	.04206	-.29248	.38506
4	.38089	.77033	.04017	-.28330	.36776
5	.45828	.76776	.04060	-.27004	.35172
6	.53567	.75593	.04086	-.25385	.33581
7	.61306	.73155	.03953	-.23354	.31925
8	.69045	.68704	.03253	-.20662	.30074
9	.76784	.62076	.02392	-.17365	.27973
10	.84522	.52033	.01183	-.13068	.25114
11	.92261	.38683	.00126	-.08531	.22053
12	.97098	.23631	-.00407	-.04874	.20624

** TOTAL WING LOADS *

WCL	WCD	WCM	X/C
.58050	.02843	-13.34778	22.99340

** TOTAL CONFIGURATION PARAMETE[illegible] *

CL	CD	CM	X/C
.63049	.02982	-13.49224	21.39966

** INDUCED DRAG = .02620

LISTING OF PROGRAM

```
A04NZXX,5,250,70000.482511,RCST5786P FANPOD
*NOTAPES,PSS
UPDATE(N)
RUN76(SA,I=COMPILE,NL50000)
REWIND(NEWPL,LGO)
COPYPSS.
CXIT.
REWIND(NEWPL,LGO)
COPYPSS.
,
*DECK MAIN
      PROGRAM FANPOD(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE9,TAPE10,
     1TAPE19,TAPE20,TAPE21,TAPE22,TAPE23,TAPE11,TAPE12,TAPE16,TAPE24)
C                     ** MAIN **
C*                    HIGH-SPEED INTERFERENCE EFFECTS ON V/STOL AIRCRAFT
C*                      (SUBSONIC INTERFERENCE PRESSURE PROGRAM)
C
      COMMON/DAT/ DA(5000)
      COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP
      COMMON/CFG/ DG(15)
      COMMON/CPB/ DB(13)
      COMMON/CPF/ DF(13)
      COMMON/CPW/ DW(18)
      COMMON/CPP/ DP(12)
      COMMON/CPN/ DN(21)
      COMMON/CFK/ DK(14)
      COMMON/CSF/ DS(90)
C
      LARGE      DM1(91812)
      LARGE      DM2(103466)
C
C*                    INITIAL
      PI=3.141593
      PI4=12.56637
      RC=57.29578
      DO 10 I=1,5000
   10 DA(I)=0.0
      DO 12 I=3380,3387
   12 DA(I)=1000000.
C*                   READ,PRINT AND TEST INPUT DATA
   20 CALL RDATA
C*                   CALC GEOMETRY
      CALL GEOM
```

```
C*                    CALC INFLUENCE MATRIX A
      CALL MATA
C*                    CALC BOUNDARY CONDITION MATRIX B
      CALL MATB
C*                    SOLVE LINEAR SIMULTANEOUS EQUATIONS (AX=B)
      CALL SOLU
C*                    CALC PRESSURE COEF. , INTEGRATED LOADS AND INDUCED DRAG
      CALL PLCAL
C*                    CALC SKIN FRICTION
      CALL SKINF
C*                    GO TO NEXT CASE
      GO TO 20
      END
*DECK CODIM
      SUBROUTINE CODIM (XI,YI,NI,T,ANS,NA)
C
C****                 A CONTROLLED DEVIATION ITERPOLATION METHOD
C
      LARGE       XI(1)      ,YI(1)      ,T(1)       ,ANS(1)
C
      DATA XK/0.5/
C
      N=NI
      DO 910 IE=1,NA
      X=T(IE)
  100 IF(N-2)110,120,200
  110 Y = YI(N)
      GO TO 900
  120 Y = (YI(2)-YI(1))/(XI(2)-XI(1))* (X-XI(1)) +YI(1)
      GO TO 900
  200 J = 1
  210 IF(XI(J)-X)230,220,250
  220 Y =YI(J)
      GO TO 900
  230 J = J+1
      IF(J-N)210,210,250
  250 IF(J-2)120,155,260
  155 J = 3
      JJ = 1
      GO TO 285
  260 IF(J-N)280,265,270
  265 J = N-1
      JJ = 2
```

```
      GO TO 285
  270 Y= (YI(N)-YI(N-1))/(XI(N)-XI(N-1))* (X-XI(N-1))+YI(N-1)
      GO TO 900
  280 JJ = 3
  285 IF(N-3)290,290,295
  290 J = 3
  295 K = J-1
      M = K-1
      L = J+1
      A1 = X-XI(M)
      A2 = X-XI(K)
      A3 = X-XI(J)
      AL = (X-XI(K))/(XI(J)-XI(K))
      S = AL*YI(J)+(1.0-AL)*YI(K)
      C1= A3*A2/((XI(M)-XI(K))*(XI(M)-XI(J)))
      C2= A1*A3/((XI(K)-XI(M))*(XI(K)-XI(J)))
      C3= A2*A1/((XI(J)-XI(M))*(XI(J)-XI(K)))
      P1 = C1*YI(M)+C2*YI(K)+C3*YI(J)
      IF(N-3)305,305,310
  305 P2 = P1
      GO TO 315
  310 A4 = X-XI(L)
      C4= A4*A3/((XI(K)-XI(J))*(XI(K)-XI(L)))
      C5= A2*A4/((XI(J)-XI(K))*(XI(J)-XI(L)))
      C6= A3*A2/((XI(L)-XI(K))*(XI(L)-XI(J)))
      P2 = C4*YI(K)+C5*YI(J)+C6*YI(L)
  315 GO TO (320,330,350),JJ
  320 P2 = P1
      AL = (X-XI(1))/(XI(2)-XI(1))
      S =  AL*YI(2)+ (1.0-AL)*YI(1)
      P1=  S + XK*(P2-S)
      GO TO 350
  330 P1 = P2
      AL = (X-XI(N-1))/(XI(N)-XI(N-1))
      S  = AL*YI(N) +(1.0-AL)*YI(N-1)
      P2 = S+ XK*(P1-S)
  350 E1 =  ABS(P1-S)
      E2 =  ABS(P2-S)
      IF(E1+E2)400,400,410
  400 Y = S
      GO TO 900
  410 BT =  (E1*AL)/(E1*AL+(1.0-AL)*E2)
      Y  = BT*P2+(1.0-BT)*P1
```

```
  900 ANS(IE)=Y
  910 CONTINUE
      RETURN
      END
*DECK WINGD
      SUBROUTINE WINGD (NETA,ETA,XL,XTXL,CR)
C
C*                   TABLE LOOKUP OF PLANAR PLANFORM FOR X LE AND CHORD
C
      LARGE        ETA(1),XL(1),XTXL(1),CR(1)
C
      DO 40 I=1,NETA
      L=1
      ET=ETA(I)
      IF(ET.LT.0.0) ET=ABS(ET)
      IF(ET.LE.CR(L)) GO TO 38
   32 IF(ET-CR(L)) 37,38,34
   34 L=L+3
      IF(L-60) 32,42,42
   37 RATIO=(CR(L)-ET)/(CR(L)-CR(L-3))
      XL(I)=CR(L+1)-(CR(L+1)-CR(L-2))*RATIO
      XTXL(I)=CR(L+2)-(CR(L+2)-CR(L-1))*RATIO-XL(I)
      GO TO 40
   38 XL(I)=CR(L+1)
      XTXL(I)=CR(L+2)-XL(I)
   40 CONTINUE
      RETURN
   42 WRITE             ( 6,1000)(ETA(K),K=1,NETA)
      CALL EXIT
      STOP
 1000 FORMAT(23H1 ** WINGD * ETA VALUES/(1H010F10.3))
      END
*DECK PFUNC
      SUBROUTINE PFUNC (IW,NEV,EV,W,S)
C
C*                   CALC. MATRIX S OF P FUNCTION
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION  W(1)
C
      LARGE        EV(1),S(51,1)
C
```

```
      COMMON/CRG/ PI,PI4,RC,BETA
C
      IC=W(IW)+.5
      IND=DA(IC)
      IF(IND) 20,30,20
   20 EL=DA(IC+2)
      ER=DA(IC+3)
      GO TO 201
C                  P-FUNCTION TYPE LOAD SHAPE
   30 ETASO  =DA(IC+1)
      RL=DA(IC+2)
      RR=DA(IC+3)
      ETAS=ETASO
   33 IF(ETASO-RL) 40,41,41
   41 IF(1.0-RR-ETASO) 42,56,56
   40 A10=1.0-ETASO/RL
      B10=1.0/RL
      C10=-(RL+RR)*(1.0-ETASO)/(RR*RL)
      E10=((RL+RR)/(RR*RL)-1.0/RL)*(1.0-(ETASO+RR))
      D10=0.0
      GO TO 43
   56 D10=(1.0-(ETASO-RL))/RL
      C10=-(1.0/RR+1.0/RL)*(1.0-ETASO)
      E10=(1.0-(ETASO+RR))/RR
      A10=0.0
      B10=0.0
      GO TO 43
   42 D10=(1.0-(ETASO-RL))/RL
      C10=-(1.0/(1.0-ETASO)+1.0/RL)*(1.0-ETASO)
      A10=0.0
      B10=0.0
      E10=0.0
   43 DO 200 IE=1,NEV
      ETA=EV(IE)
      IF(ETA.NE.0.0) GO TO 27
   10 ETA=ETA-0.0000001
   27 D=SQRT(1.0-ETA**2)
      IF(D.EQ.1.) D=.9999999
   28 ETAS = ETASO
      IF(ETASO-RL) 44,45,45
   44 ETAS=0.0
      C   = SQRT(1.0-ETAS**2)
      E   =(ETA*C   -ETAS*D   )/(ETA*C   +ETAS*D   )
```

```
      G   =(1.0+ETA*ETAS+D   *C   )/(1.0-ETA*ETAS-D   *C   )
      IF(E)  9,10,11
    9 E   = ABS(E   )
   11 IF(G) 12,10,14
   12 G   = ABS(G   )
   14 EMTPH=((ETAS-ETA)*ALOG(E)-ETAS*ALOG(G)+2.*( ACOS(ETAS))*D)/PI
   45 L=0
   49 K=0
    5 K=K+1
      C   = SQRT(1.0-ETAS**2)
      H   = SQRT( ABS((1.0-ETA*ETAS-D   *C   )/(1.0-ETA*ETAS+D   *C  )))
      O   = SQRT( ABS((1.0+ETA*ETAS-D   *C   )/(1.0+ETA*ETAS+D   *C  )))
      IF(H) 16,10,16
   16 IF(O) 18,10,18
   18 PS   =-(((ETAS-ETA)**2)* ALOG(H   )+((ETAS+ETA)**2)* ALOG(O   )
     1       +(4.*ETAS*( ACOS(ETAS)      )-2.*C)*D)/(6.28318*(1.-ETAS))
   39 IF(ETAS0-RL) 46,47,47
   46 IF(L) 48,48,47
   48 PSZERO   =PS
      ETAS=ETAS0
      L=L+1
      GO TO 49
   47 IF(K-2) 2,3,4
    2 PS0   =PS
      IF(ETAS0-RL) 50,51,51
   51 IF(1.0-RR-ETAS0) 52,50,50
   50 ETAS=ETAS0+RR
      GO TO 5
   52 ETAS=ETAS0-RL
      GO TO 5
    3 IF(ETAS0-RL) 53,55,55
   55 IF(1.0-RR-ETAS0) 57,58,58
   58 PSRR   =PS
      ETAS=ETAS0-RL
      GO TO 5
   53 PSRR   =PS
      PSRL   =0.0
      GO TO 59
   57 PSRL   =PS
      PSRR   =0.0
      EMTPH   =0.0
      PSZERO   =0.0
      GO TO 59
```

```
    4 PSRL   =PS
      EMTPH   =0.0
      PSZERO   =0.0
   59 PS    =A10*EMTPH    +B10*PSZERO    +C10*PSO    +D10*PSRL    +E10*PSRR
  200 S(IE,IW)=PS
      GO TO 300
C                  FLAP OR KRUEGER TYPE LOAD SHAPE
  201 PHII= ACOS(EL)
      PHIO= ACOS(ER)
      DO 250 IE=1,NEV
      ETA=EV(IE)
      PHI= ACOS(ETA)
      P1=ABS(PHI-PHII)/2.0
      P2=(PHI+PHII)/2.0
      IF(P1.NE.0.0) GO TO 202
      C1=0.0
      GO TO 204
  202 C1=(EL-ETA)*ALOG(SIN(P1)/SIN(P2))
  204 C2=(EL+ETA)*ALOG(COS(P2)/COS(P1))
      P1=ABS(PHI-PHIO)/2.0
      P2=(PHI+PHIO)/2.0
      IF(PI.NE.0.0) GO TO 206
      C3=0.0
      GO TO 208
  206 C3=(ER-ETA)*ALOG(SIN(P1)/SIN(P2))
  208 C4=(ER+ETA)*ALOG(COS(P2)/COS(P1))
  210 S(IE,IW)=(C1+C2-C3-C4+2.*SIN(PHI)*(PHII-PHIO))/PI
  250 CONTINUE
  300 RETURN
      END
*DECK MSOLX
      SUBROUTINE MSOLX (NKT,NQT,NCB,AA,B,ID)
C
C*                 HOUSEHOLDER METHOD FOR SOLVING SET OF LINEAR
C*                 SIMULTANEOUS EQUATIONS
C
      LARGE        B(ID,1),AA(ID,1)
C
      LARGE         DUM(91812),A(102,103),AR(204)    ,IL(102)
C
      NKTP=NKT+NCB
      DO 10 J=1,NKT
      DO 10 K=J,NKTP
```

```
   10 A(J,K) = 0.0
      DO 60 K=1,NQT
   20 DO 22 I=1,NKT
   22 AR(I)=AA(K,I)
      DO 24 J=1,NCB
      NKTJ=NKT+J
   24 AR(NKTJ)=B(K,J)
C
   30 DO 50 I=1,NKT
      R =  SQRT(A(I,I) ** 2  + AR(I) **2)
      IF(R .EQ. 0.0) GO TO 50
      C = A(I,I) / R
      S = AR(I) / R
      DO 40 J=I,NKTP
      T2 = C * A(I,J) + S * AR(J)
      AR(J) = -S * A(I,J) + C * AR(J)
   40 A(I,J) = T2
   50 CONTINUE
   60 CONTINUE
      II = 1
      DO 80 I=1,NKT
      IF(A(I,I) .LE. 0.0000001) GO TO 70
      IL(I) = II
      II = II + 1
      GO TO 80
   70 IL(I) = 0
   80 CONTINUE
      DO 250 J=1,NCB
      NKTJ=NKT+J
      DO 90 I=1,NKT
   90 AR(I)=0.0
      II=NKT
      DO 210 I=1,NKT
      IF(IL(II) .LE. 0) GO TO 210
      JI = IL(II)
      IF(II - NKT) 170, 200, 220
  170 IK = II + 1
      DO 180 K=IK,NKT
  180 AR(II) = AR(II) - A(JI,K) * AR(K)
  200 AR(II)=(AR(II)+A(JI,NKTJ))/A(JI,II)
  210 II = II - 1
  220 CONTINUE
      DO 240 I=1,NKT
```

```
  240 AA(I,J)=AR(I)
  250 CONTINUE
      RETURN
      END
*DECK RDATA
      SUBROUTINE RDATA
C
C*               READ,PRINT AND TEST INPUT DATA
C
      COMMON/DAT/ DA(5000)
      DIMENSION   WJS(1)
      EQUIVALENCE      (DA(4),XMACH)     ,(DA(12),PRII)    ,(DA(1205),WNVSO)
     2,(DA(1206),WNVSI),(DA(1690),WJS)  ,(DA(1270),WNVC),(DA(1271),WNVS)
     3,(DA(1272),WNPC),(DA(1273),WNPS)  ,(DA(1274),FNVX),(DA(1275),FNVY)
     4,(DA(1276),FNDV),(DA(1277),WNJC)  ,(DA(1278),WNJS),(DA(1279),FNJX)
     5,(DA(1280),WNU) ,(DA(1281),WNW)   ,(DA(1282),FNF) ,(DA(2492),PDA)
     6,(DA(2507),PNVC),(DA(2508),PNVS)  ,(DA(2509),PNU) ,(DA(2511),PNJC)
     7,(DA(3390),XNB) ,(DA(3394),XNP1)  ,(DA(3395),XNP2),(DA(4705),BNVX)
     8,(DA(4706),BNVY),(DA(4707),BNDV)  ,(DA(4708),BNJX),(DA(4709),BNF)

      COMMON/CFG/ IDB,IDF,IDW,IDP,IDN,NQB,NQF,NQW,NQP,NQN
     1           ,NKB,NKF,NKW,NKP,NKN
      COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP
      COMMON/CPB/ NVXB,NVXMB,NVYB,NDVB,NVBB,NVBPB,NFB,NJXB
      COMMON/CPF/ NVXF,NVXMF,NVYF,NDVF,NVBF,NVBPF,NFF,NJXF
      COMMON/CPW/ DW1(9),NU,NW,NVC,NVS,NJC,NJS,NVSO,NVSI
      COMMON/CPP/ DP1(7),NVCP,NVSP,NJCP,NUP
      COMMON/CPN/ NB,NP1,NP2,NOBP,NT
C
C*               READ INPUT DATA
      CALL DECRD (DA(1))
C*               PRINT INPUT DATA
      J=1
      WRITE (6,10) J,(DA(I),I=1,4)
   10 FORMAT (59H1 ** SUBSONIC INTERFERENCE PRESSURE PROGRAM * INPUT DAT
     1A **//1H I6,4F19.6)
      DO 20 I=5,4996,5
      IF(DA(I  ).NE.0.0) GO TO 12
      IF(DA(I+1).NE.0.0) GO TO 12
      IF(DA(I+2).NE.0.0) GO TO 12
      IF(DA(I+3).NE.0.0) GO TO 12
      IF(DA(I+4).NE.0.0) GO TO 12
      GO TO 20
```

```
   12 WRITE (6,14) I,DA(I),DA(I+1),DA(I+2),DA(I+3),DA(I+4)
   14 FORMAT (1H I6,5F19.6)
   20 CONTINUE
C*                 INITIAL
      NQB=0
      NQF=0
      NQW=0
      NQP=0
      NQN=0
      NKB=0
      NKF=0
      NKW=0
      NKP=0
      NKN=0
      IDB=DA(4000)
      IDF=DA(4001)
      IDW=DA(4002)
      IDP=DA(4003)
      IDN=DA(4004)
      REWIND 12
C*                 INITIAL FUSELAGE
      IF(IDB.EQ.0) GO TO 40
      REWIND 19
      REWIND 23
      NVXB=BNVX
      NVXMB=NVXB-1
      NVYB=BNVY
      NDVB=BNDV
      NVBB=NDVB*NVYB
      NVBPB=NVBB+1
      NFB=BNF
      NJXB=BNJX
      NKB=NVYB*NFB
      NQB=NJXB*NVYB
      NPT=NVXMB*NVBPB
      IF(NPT.LE.4000) GO TO 40
      WRITE (6,32)
   32 FORMAT (31H0 ** FUSELAGE * TOO MANY POINTS)
      GO TO 200
C*                 INITIAL FANPOD
   40 IF(IDF.EQ.0) GO TO 50
      REWIND 20
      REWIND 24
```

```
      NVXF=FNVX
      NVXMF=NVXF-1   NVYF=FNVY   4325
      NDVF=FNDV
      NVBF=NDVF*NVYF
      NVBPF=NVBF+1
      NFF=FNF
      NJXF=FNJX
      NKF=NVYF*NFF
      NQF=NJXF*NVYF
      NPT=NVXMF*NVBPF
      IF(NPT.LE.4000) GO TO 50
      WRITE (6,42)
   42 FORMAT (29H0 ** FANPOD * TOO MANY POINTS)
      GO TO 200
C*                INITIAL WING
   50 IF(IDW.EQ.0) GO TO 60
      REWIND 21
      NVC=WNVC
      NVSO=WNVSO
      NVSI=WNVSI
      NVS=NVSO
      IF(IDF.NE.0) NVS=NVSO+NVSI
      IF(NVS.EQ.0) NVS=WNVS
      NJC=WNJC
      NJS=WNJS
      NU=WNU
      NW=WNW
      IF(NW.NE.0) GO TO 54
      NW=(NVS+1)/2
      NJS=NW
      WJS(1)=1.0
      DO 52 I=2,NJS
   52 WJS(I)=WJS(I-1)+1.0
   54 NKW=NU*NW
      NQW=NJC*NJS
C*                INITIAL PYLON
   60 IF(IDP.EQ.0) GO TO 70
      REWIND 22
      NVCP=PNVC
      NVSP=PNVS
      NJCP=PNJC
      NUP=PNU
      NKP=NUP*NVSP
```

```
      NQP=NJCP*NVSP
C*                  INITIAL NACELLE
   70 IF(IDN.EQ.0) GO TO 100
      REWIND 9
      REWIND 10
      REWIND 11
      REWIND 16
      NB=XNB
      NP1=XNP1
      NT=NP1-1
      IF(NB.EQ.1) GO TO 71
      NP2=XNP2
      NT=NT+NP2-1
   71 NKN=NT
      NQN=NT
      IF(NT.LE.140) GO TO 100
      WRITE (6,72)
   72 FORMAT (30H0 ** NACELLE * TOO MANY POINTS)
      GO TO 200
C*                  TEST TOTAL POINTS
  100 NKT=NKB+NKF+NKW+NKP
      IF(NKT.LE.450) GO TO 110
      WRITE (6,102)
  102 FORMAT (30H0 ** RDATA * TOO MANY UNKNOWNS)
      GO TO 200
  110 NOBP=NQB+NQF+NQW+NQP
      IF(NOBP.LE.860) GO TO 120
      WRITE (6,112)
  112 FORMAT (30H0 ** RDATA * TOO MANY EQUATIONS)
      GO TO 200
  120 IF(XMACH.LT.1.) GO TO 210
      WRITE (6,130)
  130 FORMAT (33H0 ** MACH GREATER OR EQUAL TO 1.0)
  200 CALL EXIT
  210 BETA=SQRT(1.-XMACH**2)
C                   INITIAL PYLON DIHEDRAL ANGLE
      SINDP=0.0
      COSDP=1.0
      IF(IDP.EQ.0) GO TO 220
      PDAR=PDA/RC
      SINDP=SIN(PDAR)
      COSDP=COS(PDAR)
  220 RETURN
```

```
      END
*DECK DECRD
      SUBROUTINE DECRD (DATA)
C
C THE FUNCTION OF THIS ROUTINE IS TO READ SINGLE-PRECISION REAL NUMBERS
C FROM CARD COLUMNS 13-72 WITH A 5E12.0 FORMAT. THE DATA IS STORED INTO
C THE ADDRESS OF THE FIRST ARGUMENT MINUS ONE PLUS THE CONTENTS OF CARD
C COLUMNS 2-12 AND THE FOLLOWING FOUR LOCATIONS. IF ANY DATA ITEM
C CONTAINS 12 BLANKS, NO DATA IS STORED. CARDS ARE READ AND DATA IS
C STORED UPTO AND INCLUDING A CARD WITH A MINUS IN COLUMN ONE.
C
C
      DIMENSION DATA(1),ADATA(5),IDATA(17),IIDATA(8)
      DATA IBLANK/10H          /
C     READ A CARD AND TEST IADD. *************************************
   15 READ(5,16)IIDATA
   16 FORMAT(8A10)
      IF(EOF(5)) 60,19,60
   19 DECODE(72,17,IIDATA)IADD,ADATA
   17 FORMAT(I12,5G12.0)
      DECODE(80,18,IIDATA)IDATA
   18 FORMAT(12X,17A4)
      J=IADD
      IF(IADD) 22,40,24
   22 J=-J
C     TEST FOR BLANK FIELDS AND STORE NON-BLANK FIELDS. **************
   24 DO 30 I=1,5
      L=3*I
      K=L-2
      DO 26 M=K,L
      IF(IDATA(M)-IBLANK)28,26,28
   26 CONTINUE
      GO TO 30
   28 DATA(J)=ADATA(I)
   30 J=J+1
      IF(IADD)100,40,15
C     ERROR PRINT OF BAD CARD IMAGE. *********************************
   40 WRITE(6,50)IADD,IDATA
   50 FORMAT(17H0DECRD ER. CARD=(,1I12,17A4,2H).)
   60 CALL EXIT
  100 RETURN
      END
*DECK GEOM
```

```
      SUBROUTINE GEOM
C
C*                 CONTROL PROGRAM FOR GEOMETRY ROUTINES
C
      COMMON/DAT/ DA(5000)
      DIMENSION  WJC(1),WJS(1)
      EQUIVALENCE       (DA(1660),WJC)   ,(DA(1690),WJS)
     1,(DA(87),FOX)    ,(DA(1460),FVX)   ,(DA(2490),ATP)
     2,(DA(4015),BOX)  ,(DA(4735),BVX)
C
      COMMON/CFG/ IDB,IDF,IDW,IDP,IDN
      COMMON/CPB/ NVXB,NVXMB,NVYB,NDVB,NVBB,NVBPB,NB
      COMMON/CPF/ NVXF,NVXMF,NVYF,NDVF,NVBF,NVBPF,NF
      COMMON/CPW/ DW1(13),NJC,NJS
C
      LARGE        BOU(3000) ,DB1(13000),XBB(4000) ,YBB(4000) ,ZBB(4000)
     *            ,DF1(11700),XBF(4000) ,YBF(4000) ,ZBF(4000)
     1            ,EV(52)    ,XLV(52)   ,XTLV(52)
     2            ,DM1(1550) ,EP(30)    ,XTLP(30)  ,XOCP(30)
     3            ,DM2(10850),CFJ(50)
C
C*                START
      IF(IDB.NE.1) GO TO 10
      CALL BXYZ
      CALL BQPTS
      IF(NB.GT.0) CALL BCLS
C
   10 IF(IDF.NE.1) GO TO 20
      CALL FXYZ
      CALL FQPTS
      IF(NF.GT.0) CALL FCLS
C
   20 IF(IDW.NE.1) GO TO 30
      CALL WXYZ
      CALL RTWI
      CALL MATCS
      DO 22 J=1,NJS
      IE=WJS(J)
      EP(J)=EV(IE)
   22 XTLP(J)=XTLV(IE)
      DO 24 J=1,NJC
      IJ=WJC(J)
   24 XOCP(J)=CFJ(IJ+1)
```

```
      IF(DA(2000).NE.0.0) CALL WTZCS
C
   30 IF(IDP.NE.1) GO TO 40
      IF(ATP.GT.1.0) GO TO 34
      CALL PXYZ (NVXB,NDVB,NVBPB,BOX,BVX,XBB,YBB,ZBB)
      GO TO 36
   34 CALL PXYZ (NVXF,NDVF,NVBPF,FOX,FVX,XBF,YBF,ZBF)
   36 IF(DA(3000).NE.0.0) CALL PTZCS
C
   40 IF(IDN.NE.1) GO TO 50
      CALL NXYZ
C
   50 RETURN
      END
*DECK BXYZ
      SUBROUTINE BXYZ
C
C*                 CALC. COORDINATE OF BODY VORTICES (XB,YB,ZB)
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION   XS(1),TYS(1),RZS(1),FVX(1),FVT(1),FJX(1),YYM(1)
     1            ,XMC(1),ZZM(1),FZC(1)
      EQUIVALENCE       (DA(4005),FCD),(DA(4007),FMFI),(DA(4008),FCPI)
     1,(DA(4009),FLOI),(DA(4010),FTHI) ,(DA(4015),FOX) ,(DA(4016),FOY)
     2,(DA(4017),FOZ) ,(DA(4020),FNXS) ,(DA(4021),XS)  ,(DA(4050),FNTY)
     3,(DA(4051),TYS) ,(DA(4090),RZS)  ,(DA(4565),FNXM),(DA(4566),XMC)
     4,(DA(4600),YYM) ,(DA(4635),ZZM)  ,(DA(4670),FZC) ,(DA(4705),FNVX)
     5,(DA(4707),FNDV),(DA(4735),FVX)  ,(DA(4885),FVT) ,(DA(4905),FJX)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CPB/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX,IFBE,SINA,COSA,FAREA
     *            ,SA
C
      LARGE       BOU(3000) ,DM1(11700),AXY(650)  ,AYZ(650)  ,XB(4000)
     *,YB(4000)   ,ZB(4000)  ,FAN(23700),WIN(23430),PYL(6061) ,CEL(3700)
     1,STH(400)   ,CTH(400)  ,TH(400)
     2,STC(150)   ,CTC(150)  ,VX1(150)  ,VX2(150)  ,ZC1(150)  ,ZC2(150)
     3,RS(30)     ,STS(30)   ,ZM(150)   ,YM(150)   ,ZC(150)
     4,YMAX(150)  ,FVXL(150) ,XMCL(34)  ,YMCL(34)  ,ZZML(34)  ,FZCL(34)
C
C                 INITIAL
      NXS=FNXS
```

```
      NTY=FNTY
      NXM=FNXM
      FNVXM=FNVX-1.
      FNVB=NVB
C*               CALC LONGITUDINAL PARAMETER X
      FVX(1)=0.0
      FVX(NVX)=FCD
      IF(FLOI) 30,10,50
C                EVEN DELTA PHI
   10 DPHI=PI/(FNVXM-1.)
      PHX=DPHI/2.
      FCD2=FCD/2.
      DO 20 I=2,NVXM
      FVX(I)=FCD2-FCD2*COS(PHX)
   20 PHX=PHX+DPHI
      GO TO 50
C                EVEN DELTA X
   30 DX=FCD/(FNVXM-1.)
      FVX(2)=DX/2.
      DO 40 I=3,NVXM
   40 FVX(I)=FVX(I-1)+DX
C*               CALC LATERAL PARAMETERS
   50 IF(FTHI) 120,120,60
C                INPUT THETA LIST
   60 IF(FNDV.GT.1.) GO TO 80
      DO 70 I=1,NVY
   70 TH(I)=FVT(I)
      TH(NVY+1)=180.
      GO TO 140
   80 IC=0
      FVT(NVY+1)=180.
      DO 100 I=1,NVY
      DTH=(FVT(I+1)-FVT(I))/FNDV
      IC=IC+1
      TH(IC)=FVT(I)
      DO 100 J=2,NDV
      IC=IC+1
  100 TH(IC)=TH(IC-1)+DTH
      TH(IC+1)=180.
      GO TO 140
C                EVEN DELTA THETA
  120 DTH=180./FNVB
      TH(1)=0.
```

```
      DO 130 I=2,NVBP
  130 TH(I)=TH(I-1)+DTH
C                SIN-COS THETA
  140 DO 150 I=1,NVBP
      THR=TH(I)/RC
      STH(I)=SIN(THR)
  150 CTH(I)=COS(THR)
C*               SHIFT ARRAYS TO LCM FOR CODIM
      DO 151 I=1,NVX
  151 FVXL(I)=FVX(I)
      DO 152 I=1,NXM
      XMCL(I)=XMC(I)
      YMCL(I)=YYM(I)
      ZZML(I)=ZZM(I)
  152 FZCL(I)=FZC(I)
C*               CALC CAMBER AND MUTIPLICATION FACTORS
      IF(NXM.NE.0) GO TO 154
      DO 153 I=1,NVX
      YM(I)=1.
      ZM(I)=1.
  153 ZC(I)=0.0
      GO TO 192
  154 CALL CODIM (XMCL,YMCL,NXM,FVXL,YM,NVX)
      IF(ABS(FMFI).EQ.1.0) GO TO 180
  160 DO 170 I=1,NVX
  170 ZM(I)=YM(I)
      GO TO 190
  180 CALL CODIM (XMCL,ZZML,NXM,FVXL,ZM,NVX)
  190 CALL CODIM (XMCL,FZCL,NXM,FVXL,ZC,NVX)
  192 IF(FCPI) 220,220,200
C                PERPENDICULAR TO X-AXIS
  200 DO 210 I=2,NVX
      STC(I)=0.0
  210 CTC(I)=1.
      GO TO 250
C                PERPENDICULAR TO CAMBER LINE
  220 DX=FCD/1000.
      DXH=DX/2.
      DO 230 I=2,NVX
      VX1(I-1)=FVX(I)+DXH
  230 VX2(I-1)=FVX(I)-DXH
      CALL CODIM (XMCL,FZCL,NXM,VX1,ZC1,NVXM)
      CALL CODIM (XMCL,FZCL,NXM,VX2,ZC2,NVXM)
```

```
      DX2=DX*DX
      DO 240 I=2,NVX
      DZ=ZC2(I-1)-ZC1(I-1)
      DH=SQRT(DX2+DZ**2)
      STC(I)=DZ/DH
  240 CTC(I)=DX/DH
C*                TABLE LOOKUP FOR RADIUS AT EACH POINT
C                 INPUT (R VS THETA) VS X
  250 CALL CODMT (NXS,NTY,NVB,TYS,RZS,TH,XB)
      L=1
      DO 350 I=2,NVX
      DO 310 J=2,NXS
      IF(FVX(I)-XS(J)) 320,320,310
  310 CONTINUE
      RAT=0.0
      GO TO 330
  320 RAT=(XS(J)-FVX(I))/(XS(J)-XS(J-1))
  330 K=J
      DO 340 IT=1,NVB
      L=L+1
      ZB(L)=XB(K)-RAT*(XB(K)-XB(K-1))
  340 K=K+NXS
      L=L+1
      L1=L-NVB
      ZB(L)=ZB(L1)
  350 CONTINUE
C*                CALC XB,YB,ZB AND INTEGRATED LOAD PARAMETERS
      XB(1)=0.0
      YB(1)=0.0
      ZB(1)=0.0
      YMAX(1)=0.0
      FAREA=0.0
      L=1
      DO 390 IX=2,NVX
      C1=ZM(IX)*CTC(IX)
      IF(FMFI.LT.0.0) C1=YM(IX)*CTC(IX)
      YMAX(IX)=0.0
      DO 360 IY=1,NVBP
      L=L+1
      C2=ZB(L)*CTH(IY)
      IF(FMFI.GE.0.0) GO TO 352
      YB(L)=ZC(IX)+C1*ZB(L)*STH(IY)
      ZB(L)=C2*ZM(IX)
```

```
      GO TO 354
  352 YB(L)=YM(IX)*ZB(L)*STH(IY)
      ZB(L)=ZC(IX)+C1*C2
  354 XB(L)=FVX(IX)-C2*STC(IX)
  356 IF(YB(L).LE.YMAX(IX)) GO TO 360
      YMAX(IX)=YB(L)
  360 CONTINUE
      FAREA=(YMAX(IX)+YMAX(IX-1))*(FVX(IX)-FVX(IX-1))+FAREA
  390 CONTINUE
      IFBE=0
      IF(YMAX(NVX).GT.0.001) IFBE=1
C*               SETUP FOR COEFFICIENTS
C*               AXY AND AYZ=2*PROJECTED AREA
      IJ=1
      JX=FJX(1)
      NC=0
      SA=0.0
      M2=2
C
      DO 460 IX=1,NVXM
      IF(IX.LE.JX) GO TO 410
      IF(IJ.GE.NJX) GO TO 410
      IJ=IJ+1
      JX=FJX(IJ)
C
  410 DO 450 IY=1,NVY
      IF(IX.NE.JX) GO TO 420
      NC=NC+1
      AXY(NC)=0.0
      AYZ(NC)=0.0
C
  420 DO 450 I=1,NDV
      IF(IX.GT.1) GO TO 430
      AX=YB(M2)*XB(M2+1)-YB(M2+1)*XB(M2)
      AY=YB(M2)*ZB(M2+1)-YB(M2+1)*ZB(M2)
      AZ=XB(M2)*ZB(M2+1)-XB(M2+1)*ZB(M2)
      GO TO 440
C
  430 M1=M2-NVBP
      X4=XB(M2+1)-XB(M1)
      X3=XB(M1+1)-XB(M2)
      Y4=YB(M2+1)-YB(M1)
      Y3=YB(M1+1)-YB(M2)
```

```
      Z4=ZB(M2+1)-ZB(M1)
      Z3=ZB(M1+1)-ZB(M2)
      AX=Y4*X3-Y3*X4
      AY=Y4*Z3-Y3*Z4
      AZ=X4*Z3-X3*Z4
C
  440 IF(IX.NE.JX) GO TO 445
      AXY(NC)=AXY(NC)+AX
      AYZ(NC)=AYZ(NC)+AY
  445 SA=SQRT(AX**2+AY**2+AZ**2)+SA
C
      M2=M2+1
  450 CONTINUE
  460 M2=M2+1
C*                 COMPRESS. EFFECT AND TRANSLATE COORDINATES
  500 DO 510 I=1,L
      XB(I)=XB(I)+FOX
      YB(I)=BETA*(YB(I)+FOY)
  510 ZB(I)=BETA*(ZB(I)+FOZ)
      RETURN
      END
*DECK BQPTS
      SUBROUTINE BQPTS
C                  FUSELAGE
C*                 CALC DIRECTION MATRICES AND COORDINATE OF CONTROL
C*                 POINTS Q
C
      COMMON/DAT/ DA(5000)
      DIMENSION   FJX(1)
      EQUIVALENCE       (DA(4905),FJX)
C
      COMMON/CPB/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX
C
      LARGE       BOU(3000)
     *            ,XQ(650)   ,YQ(650)   ,ZQ(650)
     1            ,TMX(650)  ,TMY(650)  ,TMZ(650)
     2            ,TTX(650)  ,TTY(650)  ,TTZ(650)
     3            ,XN(650)   ,YN(650)   ,ZN(650)
     4            ,DM1(5200) ,XB(4000)  ,YB(4000)  ,ZB(4000)
C
C
C*                 START
      NDVH=NDV/2
```

```
      IC=0
      K1=NVBP+NDVH-1
      DO 60 IY=1,NVY
      K2=IY*NDV-K1
      DO 50 IX=1,NJX
      JX=FJX(IX)
      J2=JX*NVBP+K2
      J4=J2+1
      IF(JX-1) 10,10,20
   10 J3=1
      J1=1
      GO TO 30
   20 J1=J2-NVBP
      J3=J1+1
   30 IC=IC+1
      XQ(IC)=(XB(J1)+XB(J2)+XB(J3)+XB(J4))/4.
      YQ(I         J1)+YB(J2)+YB(J3)+YB(J4))/4.
      ZQ(IC)=(ZB(J1)+ZB(J2)+ZB(J3)+ZB(J4))/4.
C
      TM1=XB(J2)-XB(J1)+XB(J4)-XB(J3)
      TM2=YB(J2)-YB(J1)+YB(J4)-YB(J3)
      TM3=ZB(J2)-ZB(J1)+ZB(J4)-ZB(J3)
      TT1=XB(J3)-XB(J1)+XB(J4)-XB(J2)
      TT3=ZB(J3)-ZB(J1)+ZB(J4)-ZB(J2)
      TT2=YB(J3)-YB(J1)+YB(J4)-YB(J2)
C
      S2143    =SQRT(TM1**2+TM2**2+TM3**2)
      S3142=SQRT(TT1**2+TT2**2+TT3**2)
C
      TMX(IC)=TM1/S2143
      TMY(IC)=TM2/S2143
      TMZ(IC)=TM3/S2143
      TTX(IC)=TT1/S3142
      TTY(IC)=TT2/S3142
      TTZ(IC)=TT3/S3142
C
      TX=TMY(IC)*TTZ(IC)-TMZ(IC)*TTY(IC)
      TY=TMZ(IC)*TTX(IC)-TMX(IC)*TTZ(IC)
      TZ=TMX(IC)*TTY(IC)-TMY(IC)*TTX(IC)
      SRXYZ=SQRT(TX**2+TY**2+TZ**2)
      XN(IC)=TX/SRXYZ
      YN(IC)=TY/SRXYZ
   50 ZN(IC)=TZ/SRXYZ
```

```
   60 CONTINUE
      RETURN
      END
*DECK BCLS
      SUBROUTINE BCLS
C
C*                  CALC CHORDWISE LOAD SHAPES FOR THE FUSELAGE
C
      COMMON/DAT/ DA(5000)
      DIMENSION  FVX(1),F(1)
      EQUIVALENCE       (DA(4710),F)    ,(DA(4735),FVX)  ,(DA(4005),FCD)
C
      COMMON/CPB/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF
C
      LARGE        BOU(3000) ,DM1(7800) ,PM(150,25)
     1            ,DM2(70341),PHI(150)  ,CPH(150)   ,XOC(150)
C*                  START
C                   CALC  X/C , COS(PHI) AND PHI
      DO 10 I=2,NVXM
      XOC(I)=FVX(I)/FCD
      CPH(I)=1.-2.*XOC(I)
   10 PHI(I)= ACOS(CPH(I))
      IF(XOC(NVXM).LT.1.0) GO TO 30
      WRITE (6,20)
   20 FORMAT (32H1 ** BCLS * X/C GREATER THAN 1.0)
      CALL EXIT
C*                  NO. OF LOAD SHAPE LOOP
   30 DO 200 J=1,NF
      DO 90 I=2,NVXM
      IF(F(1).GE.1.0) GO TO 100
C                   LINEAR LOAD SHAPES
      DXC=XOC(I)-F(J)
      IF(DXC.GT.0.0) GO TO 70
      PM(I,J)=0.0
      GO TO 90
   70 PM(I,J)=DXC/(1.-F(J))
   90 CONTINUE
      GO TO 200
C                   TRIG. LOAD SHAPE
  100 DO 190 I=2,NVXM
      IF(F(J)-2.) 110,120,130
  110 PM(I,J)=(1.+CPH(I))/SIN(PHI(I))
      GO TO 190
```

```
  120 PM(I,J)=(1.-CPH(I))/SIN(PHI(I))
      GO TO 190
  130 IF(AMOD(F(J),2.)) 150,150,140
  140 PM(I,J)=SIN((F(J)-1.)/2.*PHI(I))
      GO TO 190
  150 PM(I,J)=COS((F(J)-2.)/2.*PHI(I))
  190 CONTINUE
  200 CONTINUE
      RETURN
      END
*DECK FXYZ
      SUBROUTINE FXYZ
C
C*                 CALC. COORDINATE OF BODY VORTICES (XB,YB,ZB)
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION   XS(29)     ,TYS(36)     ,RZS(475)   ,FVX(150)   ,FVT(50)
     1            ,FJX(149)   ,YYM(17)
     1            ,XMC(1),ZZM(1),FZC(1)
C
      EQUIVALENCE      (DA(5),FCD)        ,(DA(17),FLOI)    ,(DA(18),FTHI)
     1,(DA(19),FCTI)  ,(DA(21),XS)        ,(DA(51),TYS)     ,(DA(90),RZS)
     2,(DA(566),XMC)  ,(DA(600),YYM)      ,(DA(635),ZZM)    ,(DA(670),FZC)
     3,(DA(1276),FNDV),(DA(1460),FVX)     ,(DA(1610),FVT)   ,(DA(15),FMFI)
     4,(DA(16),FCPI)  ,(DA(1720),FJX)
     5,(DA(87),FOX)   ,(DA(88),FOY)       ,(DA(89),FOZ)
     6,(DA(20),FNXS)  ,(DA(50),FNTY)      ,(DA(565),FNXM)   ,(DA(1274),FNVX)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CPF/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX,IFBE,SINA,COSA,FAREA
     *            ,SA
C
      LARGE       BOU(3000) ,FUS(25000),DM1(10400),AXY(650)   ,AYZ(650)
     *,XB(4000)   ,YB(4000)   ,ZB(4000)   ,WIN(23430),PYL(6061) ,CEL(3700)
     1,STH(400)   ,CTH(400)   ,TH(400)
     2,STC(150)   ,CTC(150)   ,VX1(150)   ,VX2(150)   ,ZC1(150)   ,ZC2(150)
     3,RS(30)     ,STS(30)    ,ZM(150)    ,YM(150)    ,ZC(150)    ,YMIN(150)
     4,YMAX(150)  ,FVXL(150)  ,XMCL(34)   ,YMCL(34)   ,ZZML(34)   ,FZCL(34)
C
C                 INITIAL
      NXS=FNXS
      NTY=FNTY
```

```
      NXM=FNXM
      FNVXM=FNVX-1.
      FNVB=NVB
C*              CALC LONGITUDINAL PARAMETER X
      FVX(1)=0.0
      FVX(NVX)=FCD
      IF(FLOI) 30,10,50
C               EVEN DELTA PHI
   10 DPHI=PI/(FNVXM-1.)
      PHX=DPHI/2.
      FCD2=FCD/2.
      DO 20 I=2,NVXM
      FVX(I)=FCD2-FCD2*COS(PHX)
   20 PHX=PHX+DPHI
      GO TO 50
C               EVEN DELTA X
   30 DX=FCD/(FNVXM-1.)
      FVX(2)=DX/2.
      DO 40 I=3,NVXM
   40 FVX(I)=FVX(I-1)+DX
C*              CALC LATERAL PARAMETERS
   50 IF(FTHI) 120,120,60
C               INPUT THETA LIST
   60 IF(FNDV.GT.1.) GO TO 80
      DO 70 I=1,NVY
   70 TH(I)=FVT(I)
      TH(NVY+1)=360.
      GO TO 140
   80 IC=0
      FVT(NVY+1)=360.
      DO 100 I=1,NVY
      DTH=(FVT(I+1)-FVT(I))/FNDV
      IC=IC+1
      TH(IC)=FVT(I)
      DO 100 J=2,NDV
      IC=IC+1
  100 TH(IC)=TH(IC-1)+DTH
      TH(IC+1)=360.
      GO TO 140
C               EVEN DELTA THETA
  120 DTH=360./FNVB
      TH(1)=0.
      DO 130 I=2,NVBP
```

```
  130 TH(I)=TH(I-1)+DTH
C                 SIN-COS THETA
  140 DO 150 I=1,NVBP
      THR=TH(I)/RC
      STH(I)=SIN(THR)
  150 CTH(I)=COS(THR)
C*                SHIFT ARRAYS TO LCM FOR CODIM
      DO 151 I=1,NVX
  151 FVXL(I)=FVX(I)
      DO 152 I=1,NXM
      XMCL(I)=XMC(I)
      YMCL(I)=YYM(I)
      ZZML(I)=ZZM(I)
  152 FZCL(I)=FZC(I)
C*                CALC CAMBER AND MUTIPLICATION FACTORS
      IF(NXM.NE.0) GO TO 154
      DO 153 I=1,NVX
      YM(I)=1.
      ZM(I)=1.
  153 ZC(I)=0.0
      GO TO 192
  154 CALL CODIM (XMCL,YMCL,NXM,FVXL,YM,NVX)
      IF(ABS(FMFI).EQ.1.0) GO TO 180
  160 DO 170 I=1,NVX
  170 ZM(I)=YM(I)
      GO TO 190
  180 CALL CODIM (XMCL,ZZML,NXM,FVXL,ZM,NVX)
  190 CALL CODIM (XMCL,FZCL,NXM,FVXL,ZC,NVX)
  192 IF(FCPI) 220,220,200
C                 PERPENDICULAR TO X-AXIS
  200 DO 210 I=2,NVX
      STC(I)=0.0
  210 CTC(I)=1.
      GO TO 250
C                 PERPENDICULAR TO CAMBER LINE
  220 DX=FCD/1000.
      DXH=DX/2.
      DO 230 I=2,NVX
      VX1(I-1)=FVX(I)+DXH
  230 VX2(I-1)=FVX(I)-DXH
      CALL CODIM (XMCL,FZCL,NXM,VX1,ZC1,NVXM)
      CALL CODIM (XMCL,FZCL,NXM,VX2,ZC2,NVXM)
      DX2=DX*DX
```

```
      DO 240 I=2,NVX
      DZ=ZC2(I-1)-ZC1(I-1)
      DH=SQRT(DX2+DZ**2)
      STC(I)=DZ/DH
  240 CTC(I)=DX/DH
C*                 TABLE LOOKUP FOR RADIUS AT EACH POINT
C                  INPUT (R VS THETA) VS X
  250 CALL CODMT (NXS,NTY,NVB,TYS,RZS,TH,XB)
      L=1
      DO 350 I=2,NVX
      DO 310 J=2,NXS
      IF(FVX(I)-XS(J)) 320,320,310
  310 CONTINUE
      RAT=0.0
      GO TO 330
  320 RAT=(XS(J)-FVX(I))/(XS(J)-XS(J-1))
  330 K=J
      DO 340 IT=1,NVB
      L=L+1
      ZB(L)=XB(K)-RAT*(XB(K)-XB(K-1))
  340 K=K+NXS
      L=L+1
      L1=L-NVB
      ZB(L)=ZB(L1)
  350 CONTINUE
C*                 CALC XB,YB,ZB AND INTEGRATED LOAD PARAMETERS
      XB(1)=0.0
      YB(1)=0.0
      ZB(1)=0.0
      YMIN(1)=0.0
      YMAX(1)=0.0
      FAREA=0.0
      L=1
      DO 390 IX=2,NVX
      C1=ZM(IX)*CTC(IX)
      IF(FMFI.LT.0.0) C1=YM(IX)*CTC(IX)
      YMIN(IX)=1000000.
      YMAX(IX)=0.0
      DO 360 IY=1,NVBP
      L=L+1
      C2=ZB(L)*CTH(IY)
      IF(FMFI.GE.0.0) GO TO 352
      YB(L)=ZC(IX)+C1*ZB(L)*STH(IY)
```

```
      ZB(L)=C2*ZM(IX)
      GO TO 354
  352 YB(L)=YM(IX)*ZB(L)*STH(IY)
      ZB(L)=ZC(IX)+C1*C2
  354 XB(L)=FVX(IX)-C2*STC(IX)
      IF(YB(L).GT.YMAX(IX)) YMAX(IX)=YB(L)
      IF(YB(L).LT.YMIN(IX)) YMIN(IX)=YB(L)
  360 CONTINUE
      DY1=YMAX(IX)-YMIN(IX)
      DY2=YMAX(IX-1)-YMIN(IX-1)
      FAREA=(DY1+DY2)    *(FVX(IX)-FVX(IX-1))+FAREA
  390 CONTINUE
      IFBE=0
      IF(DY1.GT.0.001) IFBE=1
C*                SETUP FOR COEFFICIENTS
C*                AXY AND AYZ = PROJECTED AREA
      IJ=1
      JX=FJX(1)
      NC=0
      SA=0.0
      M2=2
C
      DO 460 IX=1,NVXM
      IF(IX.LE.JX) GO TO 410
      IF(IJ.GE.NJX) GO TO 410
      IJ=IJ+1
      JX=FJX(IJ)
C
  410 DO 450 IY=1,NVY
      IF(IX.NE.JX) GO TO 420
      NC=NC+1
      AXY(NC)=0.0
      AYZ(NC)=0.0
C
  420 DO 450 I=1,NDV
      IF(IX.GT.1) GO TO 430
      AX=YB(M2)*XB(M2+1)-YB(M2+1)*XB(M2)
      AY=YB(M2)*ZB(M2+1)-YB(M2+1)*ZB(M2)
      AZ=XB(M2)*ZB(M2+1)-XB(M2+1)*ZB(M2)
      GO TO 440
C
  430 M1=M2-NVBP
      X4=XB(M2+1)-XB(M1)
```

```
      X3=XB(M1+1)-XB(M2)
      Y4=YB(M2+1)-YB(M1)
      Y3=YB(M1+1)-YB(M2)
      Z4=ZB(M2+1)-ZB(M1)
      Z3=ZB(M1+1)-ZB(M2)
      AX=Y4*X3-Y3*X4
      AY=Y4*Z3-Y3*Z4
      AZ=X4*Z3-X3*Z4
C
  440 IF(IX.NE.JX) GO TO 445
      AXY(NC)=AXY(NC)+AX
      AYZ(NC)=AYZ(NC)+AY
  445 SA=SQRT(AX**2+AY**2+AZ**2)+SA
C
      M2=M2+1
  450 CONTINUE
  460 M2=M2+1
C*                 COMPRESS. EFFECT AND TRANSLATE COORDINATES
  500 DO 510 I=1,L
      XB(I)=XB(I)+FOX
      YB(I)=BETA*(YB(I)+FOY)
  510 ZB(I)=BETA*(ZB(I)+FOZ)
      RETURN
      END
*DECK FQPTS
      SUBROUTINE FQPTS
C                  FANPOD
C*                 CALC DIRECTION MATRICES AND COORDINATE OF CONTROL
C*                 POINTS Q
C
      COMMON/DAT/ DA(5000)
      DIMENSION   FJX(1)
      EQUIVALENCE      (DA(1720),FJX)
C
      COMMON/CPF/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX
C
      LARGE        BOU(3000) ,FUS(25000)
     *             ,XQ(650)   ,YQ(650)    ,ZQ(650)
     1             ,TMX(650)  ,TMY(650)   ,TMZ(650)
     2             ,TTX(650)  ,TTY(650)   ,TTZ(650)
     3             ,XN(650)   ,YN(650)    ,ZN(650)
     4             ,DM1(3900) ,XB(4000)   ,YB(4000)   ,ZB(4000)
C
```

```
C*                START
      NDVH=NDV/2
      IC=0
      K1=NVBP+NDVH-1
      DO 60 IY=1,NVY
      K2=IY*NDV-K1
      DO 50 IX=1,NJX
      JX=FJX(IX)
      J2=JX*NVBP+K2
      J4=J2+1
      IF(JX-1) 10,10,20
   10 J3=1
      J1=1
      GO TO 30
   20 J1=J2-NVBP
      J3=J1+1
   30 IC=IC+1
      XQ(IC)=(XB(J1)+XB(J2)+XB(J3)+XB(J4))/4.
      YQ(IC)=(YB(J1)+YB(J2)+YB(J3)+YB(J4))/4.
      ZQ(IC)=(ZB(J1)+ZB(J2)+ZB(J3)+ZB(J4))/4.
C
      TM1=XB(J2)-XB(J1)+XB(J4)-XB(J3)
      TM2=YB(J2)-YB(J1)+YB(J4)-YB(J3)
      TM3=ZB(J2)-ZB(J1)+ZB(J4)-ZB(J3)
      TT1=XB(J3)-XB(J1)+XB(J4)-XB(J2)
      TT3=ZB(J3)-ZB(J1)+ZB(J4)-ZB(J2)
      TT2=YB(J3)-YB(J1)+YB(J4)-YB(J2)
C
      S2143    =SQRT(TM1**2+TM2**2+TM3**2)
      S3142=SQRT(TT1**2+TT2**2+TT3**2)
C
      TMX(IC)=TM1/S2143
      TMY(IC)=TM2/S2143
      TMZ(IC)=TM3/S2143
      TTX(IC)=TT1/S3142
      TTY(IC)=TT2/S3142
      TTZ(IC)=TT3/S3142
C
      TX=TMY(IC)*TTZ(IC)-TMZ(IC)*TTY(IC)
      TY=TMZ(IC)*TTX(IC)-TMX(IC)*TTZ(IC)
      TZ=TMX(IC)*TTY(IC)-TMY(IC)*TTX(IC)
      SRXYZ=SQRT(TX**2+TY**2+TZ**2)
      XN(IC)=TX/SRXYZ
```

```
      YN(IC)=TY/SRXYZ
   50 ZN(IC)=TZ/SRXYZ
   60 CONTINUE
      RETURN
      END
*DECK FCLS
      SUBROUTINE FCLS
C
C                 CALC CHORDWISE LOAD SHAPES FOR THE FANPOD
C
      COMMON/DAT/ DA(5000)
      DIMENSION  FVX(1),F(1)
      EQUIVALENCE     (DA(1295),F)      ,(DA(1460),FVX),(DA(5),FCD)
C
      COMMON/CPF/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF
C
      LARGE       BOU(3000) ,FUS(25000),DM1(7800) ,PM(100,25)
     1            ,DM2(46591),PHI(150)  ,CPH(150)  ,XOC(150)
C*                START
C                 CALC  X/C , COS(PHI) AND PHI
      DO 10 I=2,NVXM
      XOC(I)=FVX(I)/FCD
      CPH(I)=1.-2.*XOC(I)
   10 PHI(I)= ACOS(CPH(I))
      IF(XOC(NVXM).LT.1.0) GO TO 30
      WRITE (6,20)
   20 FORMAT (32H1 ** FCLS * X/C GREATER THAN 1.0)
      CALL EXIT
C*                NO. OF LOAD SHAPE LOOP
   30 DO 200 J=1,NF
      IF(F(1).GE.1.0) GO TO 100
C                 LINEAR LOAD SHAPES
      DO 90 I=2,NVXM
      DXC=XOC(I)-F(J)
      IF(DXC.GT.0.0) GO TO 70
      PM(I,J)=0.0
      GO TO 90
   70 PM(I,J)=DXC/(1.-F(J))
   90 CONTINUE
      GO TO 200
C                 TRIG. LOAD SHAPE
  100 DO 190 I=2,NVXM
      IF(F(J)-2.) 110,120,130
```

```
  110 PM(I,J)=(1.+CPH(I))/SIN(PHI(I))
      GO TO 190
  120 PM(I,J)=(1.-CPH(I))/SIN(PHI(I))
      GO TO 190
  130 IF(AMOD(F(J),2.)) 150,150,140
  140 PM(I,J)=SIN((F(J)-1.)/2.*PHI(I))
      GO TO 190
  150 PM(I,J)=COS((F(J)-2.)/2.*PHI(I))
  190 CONTINUE
  200 CONTINUE
      RETURN
      END
*DECK WXYZ
      SUBROUTINE WXYZ
C
C*                 CALC. COORDINATE OF WING VORTICES AND CONTROL POINTS
C
      COMMON/DAT/ DA(5000)
      DIMENSION   WPI(1),WPO(1)
C
      EQUIVALENCE      (DA(2),WSPAN)     ,(DA(1203),WITO),(DA(1204),WITI)
     1,(DA(1210),WPO) ,(DA(1240),WPI)   ,(DA(1200),WOX) ,(DA(1202),WOZ)
     2,(DA(1208),XNEO),(DA(1209),XNEI)  ,(DA(1270),WNVC),(DA(1281),WNW)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CFG/ IDB,IDF
      COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3
     1           ,NE5,NTB,NU,NW,NVC,NVS,NJC,NJS,NVSO,NVSI,IWX
C
      LARGE       BOU(3000) ,FUS(25000),FAN(23700)
     1           ,EV(52)    ,XLV(52)   ,XTLV(52)
     2           ,DM1(810)  ,WPE(30)   ,WPT(60)   ,DM2(11540)
     3           ,CFI(50)   ,CFJ(50)   ,STH(50)   ,ACT(50)
     4           ,ET(52)    ,XLT(52)   ,XTLT(52)
C
C*               START
      DTH=PI/WNVC
      DTH2=DTH/2.
      TH=DTH2
      DO 2 I=1,NVC
      CFI(I)=.5*(1.-COS(TH))
      STH(I)=SIN(TH)
      ACT(I)=TH-DTH2
```

```
      CFJ(I)=.5*(1.-COS(ACT(I)))
    2 TH=TH+DTH
      HWS=WSPAN/2.
      IWO=3.*XNEO-2.
C                SETUP FOR WING ONLY
      IF(IDB.NE.0) GO TO 18
      IF(IDF.NE.0) GO TO 18
      DO 10 I=1,IWO,3
      WPT(I)=WPO(I)
      WPT(I+1)=WPO(I+1)
      WPT(I+2)=WPO(I+2)
   10 CONTINUE
      NPT=I
      NVSH=NVS/2
      WNVSH=NVSH
      DL=1./WNVSH
       NE2=NVSH+1
      EV(2)=1.-DL
      EV(1)=EV(2)+.625*DL
      ET(1)=1.-DL/2.
      ET(2)=ET(1)-DL
      DO 14 I=3,NE2
      ET(I)=ET(I-1)-DL
   14 EV(I)=EV(I-1)-DL
      NE5=100
      NE3=100
      DLO=DL
      DL2=DL
      GO TO 82
   18 NVSH=NVSO/2
      NE3=NVSH-3
C                SETUP WING PLANFORM TABLE
      IC=1
      IF(IDF.EQ.0) GO TO 44
      IWX=3.*XNEI-2.
      IC=IWX
      DO 40 I=1,IWX,3
      WPE(IC)=WPI(I)
      WPE(IC+1)=WPI(I+1)+WOX
      WPE(IC+2)=WPI(I+2)+WOX
      WPT(I)=WPE(IC)
      WPT(I+1)=WPE(IC+1)
      WPT(I+2)=WPE(IC+2)
```

```
 40 IC=IC-3
    WPE(6)=WPE(6)-WOX
    WPE(9)=WPE(9)-WOX
    WPT(IWX-4)=WPT(IWX-4)-WOX
    WPT(IWX-1)=WPT(IWX-1)-WOX
    CALL FRXYZ (WITI,NTI,1,IWX)
    IF(IDB.EQ.0) GO TO 42
    NE2=NVS/2
    NE5=NE2-3
    WPT(6)=WPT(6)-WOX
    WPT(9)=WPT(9)-WOX
    CALL BRXYZ
 42 IC=IWX+3
 44 DO 50 I=1,IWO,3
    WPE(I)=WPO(I)
    WPE(I+1)=WPO(I+1)+WOX
    WPE(I+2)=WPO(I+2)+WOX
    IF(I.EQ.4) WPE(I+2)=WPO(I+2)
    IF(I.EQ.7) WPE(I+2)=WPO(I+2)
    WPT(IC)=WPE(I)
    WPT(IC+1)=WPE(I+1)
    WPT(IC+2)=WPE(I+2)
    IC=IC+3
 50 CONTINUE
 60 NPT=IC-3
C*               CALC VORTEX LATTICE SPACING SPANWISE
    WNVSH=NVSH
    DLO=(1.-WPO(10))/(WNVSH-3.5)
    EV(NE3+1)=(WPO(10)+WPO(7))/2.
    ET(NE3+1)=WPO(7)
    EV(NE3+2)=(WPO(7)+WPO(4))/2.
    ET(NE3+2)=WPO(4)
    EV(NE3+3)=(WPO(4)+WPO(1))/2.
    ET(NE3+3)=WPO(1)
    EV(2)=1.-DLO
    EV(1)=EV(2)+.625*DLO
    ET(1)=1.-DLO/2.
    ET(2)=ET(1)-DLO
    DO 70 I=3,NE3
    ET(I)=ET(I-1)-DLO
 70 EV(I)=EV(I-1)-DLO
    IF(IDF.NE.0) GO TO 72
    NE5=NE3
```

```
      NE2=NE5+3
      NE3=100
      CALL BRXYZ
      ET(NE2+1)=1.
      GO TO 84
   72 CALL FRXYZ (WITO,NTO,0,IWX)
C                INNER WING
      EV(NE3+4)=(WPI(IWX-3)+WPI(IWX))/2.
      ET(NE3+4)=WPI(IWX)
      EV(NE3+5)=(WPI(IWX-6)+WPI(IWX-3))/2.
      ET(NE3+5)=WPI(IWX-3)
      EV(NE3+6)=(WPI(IWX-9)+WPI(IWX-6))/2.
      ET(NE3+6)=WPI(IWX-6)
      NE4=NE3+7
      NE5=100
      IF(IDB.EQ.0) GO TO 76
      NVSH=NVSI/2
      WNVSH=NVSH
      DLI=(WPI(IWX)-WPI(1))/WNVSH
      NE2=NVS/2
      NE5=NE2-3
      EV(NE5+1)=(WPI(10)+WPI(7))/2.
      ET(NE5+1)=WPI(10)
      EV(NE5+2)=(WPI(7)+WPI(4))/2.
      ET(NE5+2)=WPI(7)
      EV(NE2)=(WPI(4)+WPI(1))/2.
      ET(NE2)=WPI(4)
      ET(NE2+1)=WPI(1)
      IF(NE4.GT.NE5) GO TO 84
      DO 74 I=NE4,NE5
      ET(I)=ET(I-1)-DLI
   74 EV(I)=EV(I-1)-DLI
      GO TO 84
   76 NVSH=NVSI/2+1
      WNVSH=NVSH
      DLI=WPI(IWX-9)/(WNVSH-3.5)
      NE2=NVS/2+1
      DL2=DLI/2.
      DO 80 I=NE4,NE2
      ET(I)=EV(I-1)-DL2
   80 EV(I)=EV(I-1)-DLI
   82 EV(NE2)=0.
      ET(NE2+1)=-DL2
```

```
   84 NE1=NE2+1
      EV(NE1)=1.
C                FIND X LEADING EDGE AND CHORD AT EACH ETA STATIONS
      CALL WINGD (NE1,ET,XLT,XTLT,WPT)
      CALL WINGD (NE1,EV,XLV,XTLV,WPT)
C                TEST WING PLANFORM TABLE FOR NACELLE TYPE BREAK
      IF(WNW.NE.0.0) GO TO 100
      DO 90 I=1,NPT,3
      IF(WPT(I+3).GE.1.0) GO TO 100
      IF(ABS(WPT(I)-WPT(I+3)).GT.0.01) GO TO 90
      DO 85 J=1,NE2
      IF(ABS(WPT(I)-ET(J)).GT.0.005) GO TO 85
      ET(J)=-ET(J)
      XLT(J)=WPT(I+1)
      XTLT(J)=WPT(I+2)-WPT(I+1)
   85 CONTINUE
   90 CONTINUE
  100 HWS=HWS*BETA
      RETURN
      END
*DECK BRXYZ
      SUBROUTINE BRXYZ
C
C*               CALC. COORDINATES OF ROOT SECTION
C
      COMMON/DAT/ DA(5000)
      DIMENSION   FVX(150)  ,WJC(30)   ,WJS(30)
C
      EQUIVALENCE      (DA(1202),ZCON) ,(DA(1207),WITH),(DA(1660),WJC)
     1,(DA(1690),WJS) ,(DA(4015),FOX)  ,(DA(4735),FVX) ,(DA(1270),WNVC)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CPB/ NVX,NVXM,NVY,NDV,NVB,NVBP
      COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3
     1           ,NE5,NVT,NU,NW,NVC,NVS,NJC,NJS
C
      LARGE       BOU(3000) ,FUS(13000)
     1           ,XB(4000)  ,YB(4000)  ,ZB(4000)  ,FAN(23700)
     2           ,EV(52)    ,XLV(52)   ,XTLV(52)
     3           ,XD(270)   ,YD(270)   ,ZD(270)
     4           ,DM1(30)   ,WPE(60)   ,DM2(6140)
     5           ,XR(1800)  ,YR(1800)  ,ZR(1800)
     6           ,CFI(50)   ,DM3(10784),PYL(6061) ,CEL(3700)
```

```
     7              ,XRF(50)    ,YRF(50)    ,ZRF(50)    ,TAN1(150) ,TAN2(150)
C
C*                  SEARCH LIST OF FUSELAGE X FOR XLE AND XTE OF WING ROOT
      DO 50 I=1,NVX
      FVXT=FVX(I)+FOX
      IF(WPE(2).LT.FVXT) GO TO 60
   50 CONTINUE
   60 IS=I
      DO 70 I=IS,NVX
      FVXT=FVX(I)+FOX
      IF(WPE(3).LT.FVXT) GO TO 90
   70 CONTINUE
C
C*                  SHIFT VORTEX COORD. OF FUSELAGE TO THE MERGING ROOT
   90 IE=I
      ITV=WITH-1.
      IC=ITV*NDV+2+NVBP*(IS-2)
      IC1=IC-NVBP
      IF(IC1.LE.0) IC1=IC
      YCON=HWS*WPE(10)
      NPT=IE-IS+2
      DO 92 I=1,NPT
      XD(I)=XB(IC1)
      YD(I)=YB(IC1)/BETA
      ZD(I)=ZB(IC1)/BETA
   92 IC1=IC1+NVBP
      CD=WPE(3)-WPE(2)
      DO 94 I=1,NVC
   94 XRF(I)=WPE(2)+CD*CFI(I)
      CALL CODIM (XD,YD,NPT,XRF,YRF,NVC)
      CALL CODIM (XD,ZD,NPT,XRF,ZRF,NVC)
      IS=0
      DO 96 I=1,NVC
      IS=IS+4
      XR(IS)=XRF(I)
      YR(IS)=YRF(I)
   96 ZR(IS)=ZRF(I)
      XRS=XR(IS)
      NVT=NVX-IE+1
      IC=IC1-NVBP
      DO 100 I=1,NVT
      IS=IS+4
      XR(IS)=XB(IC)
```

```
      YR(IS)=YB(IC)/BETA
      ZR(IS)=ZB(IC)/BETA
  100 IC=IC+NVBP
C
C*                CALC VORTEX COORD. OF MERGING ROOT TO PLANAR WING
      XRE=XR(IS)
      WNVT=NVT
      CD=WPE(12)-WPE(11)
      IS=-3
      DO 110 I=1,NVC
      IS=IS+4
      XR(IS)=WPE(11)+CD*CFI(I)
      YR(IS)=YCON
  110 ZR(IS)=ZCON
      XRR=XR(IS)
      DXF=XRE-XRS
      DXR=XRE-XRR
      DO 120 I=1,NVT
      IS=IS+4
      XR(IS)=(XR(IS+3)-XRS)/DXF*DXR+XRR
      YR(IS)=YCON
  120 ZR(IS)=ZCON
C
C*                CALC COORD. OF THE TWO CENTER ROOT VORTEX LINES
      TTH=.6666667
      IX=4*NVC
      X1=WPE(3)
      Y1=(YR(IX)+YR(IX+4))/2.
      X2=WPE(12)
      Y2=YCON
      DX=X2-X1
      DY=Y2-Y1
      DZ=ZCON-(ZR(IX)+ZR(IX+4))/2.
      SQD=SQRT(DX**2+DY**2+DZ**2)
      XP1=X1+DX/3.-WPE(6)*DY/SQD
      YP1=Y1+DY/3.+WPE(6)*DX/SQD
      XP2=X1+TTH*DX-WPE(9)*DY/SQD
      YP2=Y1+TTH*DY+WPE(9)*DX/SQD
      TT1=(XP1-X1)/(YP1-Y1)
      TT2=(X2-XP2)/(Y2-YP2)
      DY=(WPE(4)-WPE(1))*HWS
      TL1=(WPE(5)-WPE(2))/DY
      TL2=(WPE(11)-WPE(8))/DY
```

```
C
      DT1=(TT1-TL1)/WNVC
      DT2=(TT2-TL2)/WNVC
      TAN1(1)=TL1+DT1/2.
      TAN2(1)=TL2+DT2/2.
      DO 130 I=2,NVC
      TAN1(I)=TAN1(I-1)+DT1
  130 TAN2(I)=TAN2(I-1)+DT2
C
      IS=1
      DO 140 I=1,NVC
      DX=XR(IS)-XR(IS+3)
      DY=YR(IS)-YR(IS+3)
      DZ=ZR(IS)-ZR(IS+3)
      DXDY=DX/DY
      TC1=(TAN1(I)-DXDY)/(1.+DXDY*TAN1(I))/3.
      TC2=(TAN2(I)-DXDY)/(1.+DXDY*TAN2(I))/3.
      XR(IS+2)=XR(IS+3)+DX/3.+DY*TC1
      YR(IS+2)=YR(IS+3)+DY/3.-DX*TC1
      ZR(IS+2)=ZR(IS+3)+DZ/3.
      XR(IS+1)=XR(IS+3)+TTH*DX-DY*TC2
      YR(IS+1)=YR(IS+3)+TTH*DY+DX*TC2
      ZR(IS+1)=ZR(IS+3)+TTH*DZ
  140 IS=IS+4
C*              CALC XR,YR,ZR FOR TRAILING VORTICES
      DT1=TAN1(NVC)/WNVT
      DT2=TAN2(NVC)/WNVT
      TAN1(1)=TAN1(NVC)-DT1
      TAN2(1)=TAN2(NVC)-DT2
      DO 150 I=2,NVT
      TAN1(I)=TAN1(I-1)-DT1
  150 TAN2(I)=TAN2(I-1)-DT2
C
      DO 160 I=1,NVT
      DX=XR(IS)-XR(IS+3)
      DY=YR(IS)-YR(IS+3)
      DZ=ZR(IS)-ZR(IS+3)
      DXDY=DX/DY
      TC1=(TAN1(I)-DXDY)/(1.+DXDY*TAN1(I))/3.
      TC2=(TAN2(I)-DXDY)/(1.+DXDY*TAN2(I))/3.
      XR(IS+2)=XR(IS+3)+DX/3.+DY*TC1
      YR(IS+2)=YR(IS+3)+DY/3.-DX*TC1
      ZR(IS+2)=ZR(IS+3)+DZ/3.
```

```
      XR(IS+1)=XR(IS+3)+TTH*DX-DY*TC2
      YR(IS+1)=YR(IS+3)+TTH*DY+DX*TC2
      ZR(IS+1)=ZR(IS+3)+TTH*DZ
  160 IS=IS+4
C*                 CALC CHORDS FOR WING ROOT SECTION
      WPE(3)=SQRT((WPE(2)-XR(4))**2+(WPE(1)*HWS-YR(4))**2)+WPE(2)
      WPE(6)=SQRT((WPE(5)-XR(3))**2+(WPE(4)*HWS-YR(3))**2)+WPE(5)
      WPE(9)=SQRT((WPE(8)-XR(2))**2+(WPE(7)*HWS-YR(2))**2)+WPE(8)
      IS=4
      DO 190 I=2,NVC
      IS=IS+4
      WPE(3)=SQRT((XR(IS-4)-XR(IS))**2+(YR(IS-4)-YR(IS))**2)+WPE(3)
      WPE(6)=SQRT((XR(IS-5)-XR(IS-1))**2+(YR(IS-5)-YR(IS-1))**2)+WPE(6)
  190 WPE(9)=SQRT((XR(IS-6)-XR(IS-2))**2+(YR(IS-6)-YR(IS-2))**2)+WPE(9)
      WPE(3)=SQRT((XR(IS)-X1)**2+(YR(IS)-Y1)**2)+WPE(3)
      WPE(6)=SQRT((XR(IS-1)-XP1)**2+(YR(IS-1)-YP1)**2)+WPE(6)
      WPE(9)=SQRT((XR(IS-2)-XP2)**2+(YR(IS-2)-YP2)**2)+WPE(9)
C*                 COMPRESS. EFFECT ON WING ROOT PARAMETERS
      IS=4*(NVC+NVT)
      IF(BETA.EQ.1.) GO TO 196
      DO 192 I=1,IS
      YR(I)=YR(I)*BETA
  192 ZR(I)=ZR(I)*BETA
C                  SHIFT ANS.
  196 IC=1200
      DO 200 I=1,IS
      IC=IC+1
      XR(IC)=XR(I)
      YR(IC)=YR(I)
  200 ZR(IC)=ZR(I)
C*                 CALC CONTROL POINTS AT SPECIFIED WJC AND WJS
      IS=180
      DO 230 IJS=1,NJS
      IY=WJS(IJS)
      IE=IY-NE5
      IF(IE.LE.0) GO TO 230
      IF(IE.GT.3) GO TO 230
      DO 225 IJC=1,NJC
      IX=WJC(IJC)
      IC=4*IX+IE+1200
      IS=IS+1
      XD(IS)=(XR(IC)+XR(IC+1)+XR(IC-4)+XR(IC-3))/4.
      YD(IS)=(YR(IC)+YR(IC+1)+YR(IC-4)+YR(IC-3))/4.
```

```
  225 ZD(IS)=(ZR(IC)+ZR(IC+1)+ZR(IC-4)+ZR(IC-3))/4.
  230 CONTINUE
      RETURN
      END
*DECK FRXYZ
      SUBROUTINE FRXYZ (WITH,NVT,IST,IXX)
C
C*                 CALC. COORDINATES OF ROOT SECTION
C
      COMMON/DAT/ DA(5000)
      DIMENSION   FVX(150)   ,WJC(30)    ,WJS(30)
C
      EQUIVALENCE       (DA(1202),ZCON) ,(DA(1460),FVX) ,(DA(1660),WJC)
     1,(DA(1690),WJS) ,(DA(87),FOX)     ,(DA(1270),WNVC)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CPF/ NVX,NVXM,NVY,NDV,NVB,NVBP
      COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3
     1            ,NE5,NDD,NU,NW,NVC,NVS,NJC,NJS
C
      LARGE        BOU(3000) ,FUS(25000),DM1(11700)
     1            ,XB(4000)  ,YB(4000)  ,ZB(4000)
     2            ,EV(52)    ,XLV(52)   ,XTLV(52)
     3            ,XD(270)   ,YD(270)   ,ZD(270)
     4            ,WPE(30)   ,WPT(60)   ,DM2(6140)
     5            ,XR(1800)  ,YR(1800)  ,ZR(1800)
     6            ,CFI(50)   ,DM3(10784),PYL(6061) ,CEL(3700)
     7            ,XRF(50)   ,YRF(50)   ,ZRF(50)    ,TAN1(150) ,TAN2(150)
C
C*                SEARCH LIST OF FUSELAGE X FOR XLE AND XTE OF WING ROOT
      DO 50 I=1,NVX
      FVXT=FVX(I)+FOX
      IF(WPE(2).LT.FVXT) GO TO 60
   50 CONTINUE
   60 IS=I
      DO 70 I=IS,NVX
      FVXT=FVX(I)+FOX
      IF(WPE(3).LT.FVXT) GO TO 90
   70 CONTINUE
C
C*                SHIFT VORTEX COORD. OF FUSELAGE TO THE MERGING ROOT
   90 IE=I
      ITV=WITH-1.
```

```
      IC=ITV*NDV+2+NVBP*(IS-2)
      IC1=IC-NVBP
      IF(IC1.LE.0) IC1=IC
      YCON=HWS*WPE(10)
      NPT=IE-IS+2
      DO 92 I=1,NPT
      XD(I)=XB(IC1)
      YD(I)=YB(IC1)/BETA
      ZD(I)=ZB(IC1)/BETA
   92 IC1=IC1+NVBP
      CD=WPE(3)-WPE(2)
      DO 94 I=1,NVC
   94 XRF(I)=WPE(2)+CD*CFI(I)
      CALL CODIM (XD,YD,NPT,XRF,YRF,NVC)
      CALL CODIM (XD,ZD,NPT,XRF,ZRF,NVC)
      IS=0
      DO 96 I=1,NVC
      IS=IS+4
      XR(IS)=XRF(I)
      YR(IS)=YRF(I)
   96 ZR(IS)=ZRF(I)
      XRS=XR(IS)
      NVT=NVX-IE+1
      IC=IC1-NVBP
      DO 100 I=1,NVT
      IS=IS+4
      XR(IS)=XB(IC)
      YR(IS)=YB(IC)/BETA
      ZR(IS)=ZB(IC)/BETA
  100 IC=IC+NVBP
C
C*              CALC VORTEX COORD. OF MERGING ROOT TO PLANAR WING
      XRE=XR(IS)
      WNVT=NVT
      CD=WPE(12)-WPE(11)
      IS=-3
      DO 110 I=1,NVC
      IS=IS+4
      XR(IS)=WPE(11)+CD*CFI(I)
      YR(IS)=YCON
  110 ZR(IS)=ZCON
      XRR=XR(IS)
      DXF=XRE-XRS
```

```
      DXR=XRE-XRR
      DO 120 I=1,NVT
      IS=IS+4
      XR(IS)=(XR(IS+3)-XRS)/DXF*DXR+XRR
      YR(IS)=YCON
  120 ZR(IS)=ZCON
C
C*                 CALC COORD. OF THE TWO CENTER ROOT VORTEX LINES
      TTH=.6666667
      IX=4*NVC
      X1=WPE(3)
      Y1=(YR(IX)+YR(IX+4))/2.
      X2=WPE(12)
      Y2=YCON
      DX=X2-X1
      DY=Y2-Y1
      DZ=ZCON-(ZR(IX)+ZR(IX+4))/2.
      SQD=SQRT(DX**2+DY**2+DZ**2)
      XP1=X1+DX/3.-WPE(6)*DY/SQD
      YP1=Y1+DY/3.+WPE(6)*DX/SQD
      XP2=X1+TTH*DX-WPE(9)*DY/SQD
      YP2=Y1+TTH*DY+WPE(9)*DX/SQD
      TT1=(XP1-X1)/(YP1-Y1)
      TT2=(X2-XP2)/(Y2-YP2)
      DY=(WPE(4)-WPE(1))*HWS
      TL1=(WPE(5)-WPE(2))/DY
      TL2=(WPE(11)-WPE(8))/DY
C
      DT1=(TT1-TL1)/WNVC
      DT2=(TT2-TL2)/WNVC
      TAN1(1)=TL1+DT1/2.
      TAN2(1)=TL2+DT2/2.
      DO 130 I=2,NVC
      TAN1(I)=TAN1(I-1)+DT1
  130 TAN2(I)=TAN2(I-1)+DT2
C
      IS=1
      DO 140 I=1,NVC
      DX=XR(IS)-XR(IS+3)
      DY=YR(IS)-YR(IS+3)
      DZ=ZR(IS)-ZR(IS+3)
      DXDY=DX/DY
      TC1=(TAN1(I)-DXDY)/(1.+DXDY*TAN1(I))/3.
```

```
      TC2=(TAN2(I)-DXDY)/(1.+DXDY*TAN2(I))/3.
      XR(IS+2)=XR(IS+3)+DX/3.+DY*TC1
      YR(IS+2)=YR(IS+3)+DY/3.-DX*TC1
      ZR(IS+2)=ZR(IS+3)+DZ/3.
      XR(IS+1)=XR(IS+3)+TTH*DX-DY*TC2
      YR(IS+1)=YR(IS+3)+TTH*DY+DX*TC2
      ZR(IS+1)=ZR(IS+3)+TTH*DZ
  140 IS=IS+4
C*               CALC XR,YR,ZR FOR TRAILING VORTICES
      DT1=TAN1(NVC)/WNVT
      DT2=TAN2(NVC)/WNVT
      TAN1(1)=TAN1(NVC)-DT1
      TAN2(1)=TAN2(NVC)-DT2
      DO 150 I=2,NVT
      TAN1(I)=TAN1(I-1)-DT1
  150 TAN2(I)=TAN2(I-1)-DT2
C
      DO 160 I=1,NVT
      DX=XR(IS)-XR(IS+3)
      DY=YR(IS)-YR(IS+3)
      DZ=ZR(IS)-ZR(IS+3)
      DXDY=DX/DY
      TC1=(TAN1(I)-DXDY)/(1.+DXDY*TAN1(I))/3.
      TC2=(TAN2(I)-DXDY)/(1.+DXDY*TAN2(I))/3.
      XR(IS+2)=XR(IS+3)+DX/3.+DY*TC1
      YR(IS+2)=YR(IS+3)+DY/3.-DX*TC1
      ZR(IS+2)=ZR(IS+3)+DZ/3.
      XR(IS+1)=XR(IS+3)+TTH*DX-DY*TC2
      YR(IS+1)=YR(IS+3)+TTH*DY+DX*TC2
      ZR(IS+1)=ZR(IS+3)+TTH*DZ
  160 IS=IS+4
C*               CALC CHORDS FOR WING ROOT SECTION
      WPE(3)=SQRT((WPE(2)-XR(4))**2+(WPE(1)*HWS-YR(4))**2)+WPE(2)
      WPE(6)=SQRT((WPE(5)-XR(3))**2+(WPE(4)*HWS-YR(3))**2)+WPE(5)
      WPE(9)=SQRT((WPE(8)-XR(2))**2+(WPE(7)*HWS-YR(2))**2)+WPE(8)
      IS=4
      DO 190 I=2,NVC
      IS=IS+4
      WPE(3)=SQRT((XR(IS-4)-XR(IS))**2+(YR(IS-4)-YR(IS))**2)+WPE(3)
      WPE(6)=SQRT((XR(IS-5)-XR(IS-1))**2+(YR(IS-5)-YR(IS-1))**2)+WPE(6)
  190 WPE(9)=SQRT((XR(IS-6)-XR(IS-2))**2+(YR(IS-6)-YR(IS-2))**2)+WPE(9)
      WPE(3)=SQRT((XR(IS)-X1)**2+(YR(IS)-Y1)**2)+WPE(3)
      WPE(6)=SQRT((XR(IS-1)-XP1)**2+(YR(IS-1)-YP1)**2)+WPE(6)
```

```
      WPE(9)=SQRT((XR(IS-2)-XP2)**2+(YR(IS-2)-YP2)**2)+WPE(9)
C*                 COMPRESS. EFFECT ON WING ROOT PARAMETERS
      IS=4*(NVC+NVT)
      IF(BETA.EQ.1.) GO TO 196
      DO 192 I=1,IS
      YR(I)=YR(I)*BETA
  192 ZR(I)=ZR(I)*BETA
C                  SHIFT ANS.
  196 IF(IST.EQ.0) GO TO 220
      IX=0
      DO 200 I=1,IS,4
      IC=I+603
      DO 200 J=1,4
      IX=IX+1
      XR(IC)=XR(IX)
      YR(IC)=YR(IX)
      ZR(IC)=ZR(IX)
  200 IC=IC-I
C*                 CALC CONTROL POINTS AT SPECIFIED WJC AND WJS
      IS=90
      DO 210 IJS=1,NJS
      IY=WJS(IJS)
      IE=IY-NE3
      IF(IE.LE.3) GO TO 210
      IF(IE.GT.6) GO TO 210
      DO 205 IJC=1,NJC
      IX=WJC(IJC)
      IC=4*IX+IE+597
      IS=IS+1
      XD(IS)=(XR(IC)+XR(IC+1)+XR(IC-4)+XR(IC-3))/4.
      YD(IS)=(YR(IC)+YR(IC+1)+YR(IC-4)+YR(IC-3))/4.
  205 ZD(IS)=(ZR(IC)+ZR(IC+1)+ZR(IC-4)+ZR(IC-3))/4.
  210 CONTINUE
      WPT(IXX+2)=WPE(3)
      WPT(IXX-1)=WPE(6)
      WPT(IXX-4)=WPE(9)
      GO TO 300
  220 IS=0
      DO 230 IJS=1,NJS
      IY=WJS(IJS)
      IE=IY-NE3
      IF(IE.LE.0) GO TO 230
      IF(IE.GT.3) GO TO 230
```

```
      DO 225 IJC=1,NJC
      IX=WJC(IJC)
      IC=4*IX+IE
      IS=IS+1
      XD(IS)=(XR(IC)+XR(IC+1)+XR(IC-4)+XR(IC-3))/4.
      YD(IS)=(YR(IC)+YR(IC+1)+YR(IC-4)+YR(IC-3))/4.
  225 ZD(IS)=(ZR(IC)+ZR(IC+1)+ZR(IC-4)+ZR(IC-3))/4.
  230 CONTINUE
      WPT(IXX+5)=WPE(3)
      WPT(IXX+8)=WPE(6)
      WPT(IXX+11)=WPE(9)
  300 RETURN
      END
*DECK RTWI
      SUBROUTINE RTWI
C
C*                ROTATE WING ROOT COORDINATES TO THE ACTUAL CHORD LINES.
C
      COMMON/DAT/ DA(5000)
      DIMENSION   WET(1),TWI(1),TDZL(1),DZLE(52),EX(52)
      EQUIVALENCE       (DA(731),WET)    ,(DA(1970),TWI)   ,(DA(1985),TDZL)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CFG/ IDB,IDF
      COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3,NE5,NTB,NU,NW,NVC
C
      LARGE       DM1(51700),EV(52)     ,DM2(7144)
     1            ,XB1(600)   ,XB2(600)   ,XB3(600)   ,DM3(1800)
     2            ,ZB1(600)   ,ZB2(600)   ,ZB3(600)   ,DM4(200)
     3            ,ET(52)     ,X          ,DM5(6722)
     4            ,COSTT(52)  ,SINTT(52)  ,COSTV(52)  ,SINTV(52)
     5            ,XB(600)    ,XBI(600)   ,XBR(600)
     6            ,ZB(600)    ,ZBI(600)   ,ZBR(600)
C
C                 LOOKUP FOR TWIST
      NEW=DA(730)
      NVSH=NE2+1
      CALL TWIST (NEW,WET,TWI,NVSH,EV,COSTV,SINTV)
      CALL TWIST (NEW,WET,TWI,NVSH,ET,COSTT,SINTT)
C                 LOOKUP FOR DZLE AT TRAIL VORTICES
      IF(NEW.LT.1) NEW=1
      DO 5 I=1,NVSH
    5 EX(I)=ET(I)
```

```
      CALL COD (NEW,WET,IDZL,NVSH,EX,DZLE)
      DO 10 I=1,NVSH
      IF(DA(730).LT.1.0) DZLE(I)=0.0
   10 DZLE(I)=DZLE(I)*BETA
C                 ROTATE XB AND ZB
      NES=NE5
      NEE=NE2
      IF(IDF.EQ.0) GO TO 40
      NE4=NE3+7
      NE6=NE3+3
      DZLE(NE6)=0.0
      COSTT(NE6)=1.0
      SINTT(NE6)=0.0
      NVO=NVC+NTO
      NVI=NVC+NTI
      IS=0
      DO 20 IX=1,NVO
      DO 20 IE=NE3,NE6
      IS=IS+1
      DXX=XB1(IS)-XLT(IE)
      XB(IS)=XLT(IE)+DXX*COSTT(IE)
   20 ZB(IS)=ZB1(IS)-DXX*SINTT(IE)+DZLE(IE)
C
      NEX=NE3+4
      DZLE(NEX)=0.0
      COSTT(NEX)=1.0
      SINTT(NEX)=0.0
      IS=0
      DO 30 IX=1,NVI
      DO 30 IE=NEX,NE4
      IS=IS+1
      DXX=XB2(IS)-XLT(IE)
      XBI(IS)=XLT(IE)+DXX*COSTT(IE)
   30 ZBI(IS)=ZB2(IS)-DXX*SINTT(IE)+DZLE(IE)
C
      NES=NES+1
      NEE=NEE+1
   40 IF(IDB.EQ.0) GO TO 80
      NVO=NVC+NTB
      DZLE(NEE)=0.0
      COSTT(NEE)=1.0
      SINTT(NEE)=0.0
      IS=0
```

```
      DO 60 IX=1,NVO
      DO 60 IE=NES,NEE
      IS=IS+1
      DXX=XB3(IS)-XLT(IE)
      XBR(IS)=XLT(IE)+DXX*COSTT(IE)
   60 ZBR(IS)=ZB3(IS)-DXX*SINTT(IE)+DZLE(IE)
C
   80 CONTINUE
C                LOOKUP FOR DZLE AT CENTER OF VORTICES
      DO 85 I=1,NVSH
   85 EX(I)=EV(I)
      CALL COD (NEW,WET,TDZL,NVSH,EX,DZLE)
      DO 90 I=1,NVSH
      IF(DA(730).LT.1.0) DZLE(I)=0.0
   90 SINTT(I)=DZLE(I)
      RETURN
      END
*DECK TWIST
      SUBROUTINE TWIST (NEI,EI,TI,NEO,EO,CT,ST)
C
C*               LOOKUP TWIST ANGLE ANDCALC COS AND SIN OF TWIST
C
      LARGE      EO(1),CT(1),ST(1)
      DIMENSION  EI(1),TI(1)
C
C
      DO 60 I=1,NEO
      IF(NEI.NE.0) GO TO 5
      CT(I)=1.0
      ST(I)=0.0
      GO TO 60
    5 ETA=ABS(EO(I))
      IF(ETA.GT.EI(1)) GO TO 10
      TW=TI(1)
      GO TO 50
   10 DO 20 J=2,NEI
      IF(ETA-EI(J)) 40,30,20
   20 CONTINUE
      TW=TI(NEI)
      GO TO 50
   30 TW=TI(J)
      GO TO 50
   40 RATY=(EI(J)-ETA)/(EI(J)-EI(J-1))
```

```
      TW=TI(J)-RATY*(TI(J)-TI(J-1))
   50 CT(I)=COS(TW)
      ST(I)=SIN(TW)
   60 CONTINUE
      RETURN
      END
*DECK MATCS
      SUBROUTINE MATCS
C
C*                   CALC MATRICES C AND S
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION   W(10)
C
      EQUIVALENCE      (DA(1285),W)     ,(NI,NVC)         ,(DA(1281),WNW)
     1,(DA(1960),DAF) ,(DA(1961),PXCF) ,(DA(1965),DAK) ,(DA(1966),PXCK)
     2,(DA(2),WS)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CPW/ DW1(9),NU,NW,NVC,NVS
      COMMON/CFK/ NDA,NUT,THEF,SINTF,SINTF2,COSTF,THEK,SINTK,SINTK2
     1               ,COSTK,SINDF,COSDF,SINDK,COSDK
C
      LARGE          BOU(3000) ,FUS(25000),FAN(23700),EV(52)     ,DM1(12544)
     1               ,CFI(50)   ,CFJ(50)   ,STH(50)   ,ACT(50)   ,DM2(156)
     2               ,C(40,40)  ,S(51,10)  ,CSEC(40,10),DM3(7968)
     3               ,PYL(6061) ,CEL(3700) ,BC(40,10)
C*                   CALC MATRIX C
C                    FLAP SETUP
      NDA=0
      IF(DAF.NE.0.0) GO TO 4
      THEF=0.0
      SINTF=0.0
      SINTF2=0.0
      GO TO 10
    4 NDA=1
      COSTF=1.-2.*PXCF
      THEF= ACOS(COSTF)
      SINTF=SIN(THEF)
      SINTF2=SIN(2.*THEF)
      SINDF=SIN(DAF)
      COSDF=COS(DAF)
```

```
      NUS=NU-1
      IF(DAK.EQ.0.0) NUS=NU
      BC(1,NUS)=SINTF/2.
      PIT=THEF-PI
      DO 5 I=2,NI
      IF(THEF.LE.ACT(I)) GO TO 6
    5 BC(I,NUS)=PIT
    6 KF=I
      DO 8 J=KF,NI
    8 BC(J,NUS)=THEF
C                KRUEGER SETUP
   10 IF(DAK.NE.0.0) GO TO 12
      THEK=0.0
      SINTK=0.0
      SINTK2=0.0
      GO TO 20
   12 NDA=NDA+1
      COSTK=1.-2.*PXCK
      THEK= ACOS(COSTK)
      SINTK=SIN(THEK)
      SINTK2=SIN(2.*THEK)
      SINDK=SIN(DAK)
      COSDK=COS(DAK)
      BC(1,NU)=SINTK/2.
      PIT=THEK-PI
      DO 14 I=2,NI
      IF(THEK.LT.ACT(I)) GO TO 16
   14 BC(I,NU)=PIT
   16 KK=I
      DO 18 J=KK,NI
   18 BC(J,NU)=THEK
   20 NUT=NU-NDA
C*
      XI=0.0
      DO 22 I=2,NUT
      XI=XI+1.
      BC(1,I)=0.0
      DO 22 J=2,NI
   22 BC(J,I)=-COS(XI*ACT(J))
      BC(1,2)=.25
      DO 24 I=1,NI
      C(1,I)=1.
      BC(I,1)=1.
```

```
      DO 24 J=2,NI
   24 C(J,I)=1./(CFJ(J)-CFI(I))
      BC(1,1)=.5
      CALL MSOLX (NI,NI,NU,C,BC,40)
C*                 CALC MATRIX CSEC
      DO 40 I=1,NU
      CSEC(1,I)=C(1,I)*WS
      DO 40 K=2,NVC
   40 CSEC(K,I)=C(K,I)*WS+CSEC(K-1,I)
C
      IF(WNW.EQ.0.0) GO TO 200
C*                 CALC MATRIX S
      NEV=(NVS+1)/2
      DO 160 I=1,NW
      L=W(I)
      IF(L.LT.1000) GO TO 100
      CALL PFUNC (I,NEV,EV,W,S)
      GO TO 150
  100 DO 140 J=1,NEV
      ETA=EV(J)
      IF(ETA.GT.0.0000001) GO TO 130
      IF(L.EQ.0) GO TO 120
      S(J,I)=0.0
      GO TO 140
  120 S(J,I)=1.0
      GO TO 140
  130 S(J,I)=SQRT(1.-ETA**2)*ETA**L
  140 CONTINUE
  150 S(1,I)=4.*S(1,I)
  160 CONTINUE
  200 RETURN
      END
*DECK WTZCS
      SUBROUTINE WTZCS
C
C*                 CALC. THICKNESS SLOPESFROM DEFLECTIONS (Z/C).
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION   WJC(30)    ,ED(19)     ,XOCD(24)   ,TZC(750)
C
      EQUIVALENCE      (DA(2001),XOCD)  ,(DA(2026),ED)    ,(DA(2050),TZC)
     1,(NI,NVC)       ,(NEP,NJS)        ,(NJ,NJC)         ,(DA(1660),WJC)
```

```
C
      COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3,DW1(4),NVC,NVS,NJC,NJS
C
      LARGE       BOU(3000) ,FUS(25000),FAN(23700)
     1,EV(52)     ,XLV(52)   ,XTLV(52)  ,DPI(840)   ,WPT(60)    ,BS(650)
     2            ,EP(30)    ,XTLP(30)  ,XOCP(30)
     3            ,CSJ(30,30),TSJ(30,30),DM1(9000)
     4            ,CFI(50)   ,CFJ(50)   ,DM2(2766)
     5            ,TSI(52,40),TZI(52,40),DM3(3808)  ,PYL(6061)  ,CEL(3700)
     6            ,CHD(24)   ,ZD(24)    ,ZCJ(50)    ,ZCI(50)    ,ZJ(50,20)
     7            ,ZI(50,20) ,EL(20)    ,XOCL(24)
C
C*                INITIAL
      LD=0
      NDE=DA(2025)
      NXD=DA(2000)
C*                SHIFT TO LCM FOR WINGD AND CODIM
      DO 10 I=1,NDE
   10 EL(I)=ED(I)
      DO 20 I=1,NXD
   20 XOCL(I)=XOCD(I)
      CALL WINGD (NDE,EL,ZD,CHD,WPT)
      NI1=NI+1
      CFJ(NI1)=1.
      DO 100 I=1,NDE
   30 DO 40 IX=1,NXD
      LD=LD+1
   40 ZD(IX)=TZC(LD)*CHD(I)
      CALL CODIM (XOCL,ZD,NXD,CFJ,ZCJ,NI1)
      CALL CODIM (XOCL,ZD,NXD,CFI,ZCI,NI)
      DO 60 K=1,NI
      ZJ(K,I)=ZCJ(K)
   60 ZI(K,I)=ZCI(K)
      ZJ(NI1,I)=ZCJ(NI1)
  100 CONTINUE
C
C               LINEAR INTERPOLATION FOR Z/C SPANWISE AND CALC SLOPE
      DO 200 IE=1,NEP
      IF(EP(IE).GT.ED(1)) GO TO 120
      II=1
      GO TO 140
  120 DO 130 IS=2,NDE
      II=IS
```

```
      IF(EP(IE)-ED(IS)) 150,140,130
  130 CONTINUE
  140 DO 145 J=1,NJ
      IC=WJC(J)+1.
  145 TSJ(IE,J)=(ZI(IC,II)-ZI(IC-1,II))/(CFI(IC)-CFI(IC-1))/XTLP(IE)
      GO TO 200
  150 RATY=(ED(II)-EP(IE))/(ED(II)-ED(II-1))
  160 DO 170 J=1,NJ
      IC=WJC(J)+1.
      Z1=ZI(IC-1,II)-RATY*(ZI(IC-1,II)-ZI(IC-1,II-1))
      Z2=ZI(IC,II)-RATY*(ZI(IC,II)-ZI(IC,II-1))
  170 TSJ(IE,J)=(Z2-Z1)/(CFI(IC)-CFI(IC-1))/XTLP(IE)
  200 CONTINUE
      DO 300 IE=1,NE2
      IF(EV(IE).GT.ED(1)) GO TO 220
      II=1
      GO TO 240
  220 DO 230 IS=2,NDE
      II=IS
      IF(EV(IE)-ED(IS)) 250,240,230
  230 CONTINUE
  240 DO 245 I=1,NI
      TZI(IE,I)=ZI(I,II)
  245 TSI(IE,I)=(ZJ(I+1,II)-ZJ(I,II))/(CFJ(I+1)-CFJ(I))
      GO TO 300
  250 RATY=(ED(II)-EV(IE))/(ED(II)-ED(II-1))
  260 DO 270 I=1,NI
      Z1=ZJ(I,II)-RATY*(ZJ(I,II)-ZJ(I,II-1))
      Z2=ZJ(I+1,II)-RATY*(ZJ(I+1,II)-ZJ(I+1,II-1))
      TZI(IE,I)=ZI(I,II)-RATY*(ZI(I,II)-ZI(I,II-1))
  270 TSI(IE,I)=(Z2-Z1)/(CFJ(I+1)-CFJ(I))
  300 CONTINUE
      LT=NE2+1
      DO 320 I=1,NI
      TZI(LT,I)=ZI(I,NDE)
  320 TSI(LT,I)=(ZJ(I+1,NDE)-ZJ(I,NDE))/(CFJ(I+1)-CFJ(I))
      RETURN
      END
*DECK PXYZ
      SUBROUTINE PXYZ (NVX,NDV,NVBP,FOX,FVX,XB,YB,ZB)
C
C*                 CALC. COORDINATES OF PYLON VORTICES AND CONTROL POINTS
C
```

```
      COMMON/DAT/ DA(5000)
C
      DIMENSION   PPT(1),FVX(1),PJC(1)
      LARGE       XB(1),YB(1),ZB(1)
C
      EQUIVALENCE       (DA(2490),ATP)   ,(DA(2491),PITH),(DA(2492),PDA)
     1,(DA(2500),PHT) ,(DA(2501),POX)   ,(DA(2502),POY) ,(DA(2503),POZ)
     2,(DA(2507),PNVC),(DA(2508),PNVS)  ,(DA(2512),PJC) ,(DA(2960),PPT)
C
      COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP
      COMMON/CPP/ HPS,NVSH,NVT,DL,NEVP,ZCON,YCON,NVC,NVS,NJC,NUP
C
      LARGE          DM1(75130)
     1               ,EV(22)    ,XLV(22)   ,XTLV(22)   ,WPE(39)
     2               ,XD(60)    ,YD(60)    ,ZD(60)
     3               ,XR(400)   ,YR(400)   ,ZR(400)
     4               ,CFI(20)   ,CFJ(20)   ,STH(20)    ,ACT(20)
     5               ,ET(22)    ,XLT(22)   ,XTLT(22)   ,C(20,20)
     6               ,DM2(2980) ,CSEC(21,10),DM3(840)  ,CEL(3700)
     7               ,XRF(50)   ,YRF(50)   ,ZRF(50)    ,TAN1(150) ,TAN2(150)
     8               ,BC(20,10)
C
C*               START
      DTH=PI/PNVC
      DTH2=DTH/2.
      TH=DTH2
      DO 2 I=1,NVC
      CFI(I)=.5*(1.-COS(TH))
      STH(I)=SIN(TH)
      ACT(I)=TH-DTH2
      CFJ(I)=.5*(1.-COS(ACT(I)))
    2 TH=TH+DTH
C                CALC. CHORDWISE LOAD COEFF.
      XI=0.0
      DO 4 I=2,NUP
      XI=XI+1.
      BC(1,I)=0.0
      DO 4 J=2,NVC
    4 BC(J,I)=-COS(XI*ACT(J))
      BC(1,2)=.25
      DO 6 I=1,NVC
      C(1,I)=1.
      BC(I,1)=1.
```

```
      DO 6 J=2,NVC
    6 C(J,I)=1./(CFJ(J)-CFI(I))
      BC(1,1)=.5
      CALL MSOLX (NVC,NVC,NUP,C,BC,20)
C
      SPAN=PHT+PHT
      DO 8 I=1,NUP
      CSEC(1,I)=C(1,I)*SPAN
      DO 8 K=2,NVC
    8 CSEC(K,I)=C(K,I)*SPAN+CSEC(K-1,I)
C                SHIFT PYLON PLANFORM TABLE
      HPS=PHT
      DO 10 I=1,37,3
      WPE(I)=PPT(I)
      WPE(I+1)=PPT(I+1)
      WPE(I+2)=PPT(I+2)
      IF(PPT(I).GE.1.) GO TO 12
   10 CONTINUE
   12 NPP=I-2
C*               CALC VORTEX LATTICE SPACE SPANWISE
      DL=(1.-WPE(10))/(PNVS-3.5)
      NEVP=NVS-3
      EV(NEVP+1)=(WPE(10)+WPE(7))/2.
      EV(NEVP+2)=(WPE( 7)+WPE(4))/2.
      EV(NEVP+3)=(WPE( 4)+WPE(1))/2.
   30 EV(2)=1.-DL
      EV(1)=EV(2)+.625*DL
      ET(1)=1.-DL/2.
      ET(2)=ET(1)-DL
      DO 40 I=3,NEVP
      ET(I)=ET(I-1)-DL
   40 EV(I)=EV(I-1)-DL
      NVSH=NVS+1
      EV(NVSH)=1.0
      CALL WINGD (NEVP,ET,XLT,XTLT,WPE)
C                TEST WING PLANFORM FOR NACELLE TYPE BREAK
      DO 44 I=1,NPP,3
      IF(ABS(WPE(I)-WPE(I+3)).GT.0.01) GO TO 44
      DO 42 J=1,NEVP
      IF(ABS(WPE(I)-ET(J)).GT.0.005) GO TO 42
      ET(J)=-ET(J)
      XLT(J)=DA(I+1)
      XTLT(J)=DA(I+2)-DA(I+1)
```

```
   42 CONTINUE
   44 CONTINUE
C
      IF(ATP.EQ.3.0) GO TO 102
C*             SEARCH LIST OF FUSELAGE X FOR XLE AND XTE OF WING ROOT
      DO 50 I=1,NVX
      FVXT=FVX(I)+FOX-POX
      IF(WPE(2).LT.FVXT) GO TO 60
   50 CONTINUE
   60 IS=I
      DO 70 I=IS,NVX
      FVXT=FVX(I)+FOX-POX
      IF(WPE(3).LT.FVXT) GO TO 90
   70 CONTINUE
C
C*             SHIFT VORTEX COORD. OF FUSELAGE TO THE MERGING ROOT
   90 IE=I
      ITV=PITH-1.0
      IC=(IS-2)*NVBP+2+ITV*NDV
      IC1=IC-NVBP
      IF(IC1.LE.0) IC1=IC
      NPT=IE-IS+2
      DO 92 I=1,NPT
      XD(I)=XB(IC1)-POX
      YP=YB(IC1)/BETA-POY
      ZP=ZB(IC1)/BETA-POZ
      YD(I)=YP*SINDP+ZP*COSDP
      ZD(I)=ZP*SINDP-YP*COSDP
   92 IC1=IC1+NVBP
      CD=WPE(3)-WPE(2)
      DO 94 I=1,NVC
   94 XRF(I)=WPE(2)+CD*CFI(I)
      CALL CODIM (XD,YD,NPT,XRF,YRF,NVC)
      CALL CODIM (XD,ZD,NPT,XRF,ZRF,NVC)
      IS=0
      DO 96 I=1,NVC
      IS=IS+4
      XR(IS)=XRF(I)
      YR(IS)=YRF(I)
   96 ZR(IS)=ZRF(I)
      XRS=XR(IS)
      NVT=NVX-IE+1
      IC=IC1-NVBP
```

```
      DO 100 I=1,NVT
      IS=IS+4
      XR(IS)=XB(IC)-POX
      YP=YB(IC)/BETA-POY
      ZP=ZB(IC)/BETA-POZ
      YR(IS)=YP*SINDP+ZP*COSDP
      ZR(IS)=ZP*SINDP-YP*COSDP
  100 IC=IC+NVBP
      GO TO 108
C                     PYLON ATTACH TO WING
  102 CD=WPE(3)-WPE(2)
      YP=WPE(1)*HPS
      IS=0
      DO 104 I=1,NVC
      IS=IS+4
      XR(IS)=WPE(2)+CD*CFI(I)
      YR(IS)=YP
  104 ZR(IS)=0.0
      XRS=XR(IS)
      DX=XRS-XR(IS-4)
      NVT=2
      DO 106 I=1,NVT
      IS=IS+4
      XR(IS)=XR(IS-4)+DX
      YR(IS)=YP
  106 ZR(IS)=0.0
C
C*                    CALC VORTEX COORD. OF MERGING ROOT TO PLANAR WING
  108 ZCON=0.0
      YCON=HPS*WPE(10)
      XRE=XR(IS)
      WNVT=NVT
      CD=WPE(12)-WPE(11)
      IS=-3
      DO 110 I=1,NVC
      IS=IS+4
      XR(IS)=WPE(11)+CD*CFI(I)
      YR(IS)=YCON
  110 ZR(IS)=ZCON
      XRR=XR(IS)
      DXF=XRE-XRS
      DXR=XRE-XRR
      DO 120 I=1,NVT
```

```
      IS=IS+4
      XR(IS)=(XR(IS+3)-XRS)/DXF*DXR+XRR
      YR(IS)=YCON
  120 ZR(IS)=ZCON
C
C*                CALC COORD. OF THE TWO CENTER ROOT VORTEX LINES
      TTH=.6666667
      IX=4*NVC
      X1=WPE(3)
      Y1=(YR(IX)+YR(IX+4))/2.
      X2=WPE(12)
      Y2=YCON
      DX=X2-X1
      DY=Y2-Y1
      DZ=ZCON-(ZR(IX)+ZR(IX+4))/2.
      SQD=SQRT(DX**2+DY**2+DZ**2)
      XP1=X1+DX/3.-WPE(6)*DY/SQD
      YP1=Y1+DY/3.+WPE(6)*DX/SQD
      XP2=X1+TTH*DX-WPE(9)*DY/SQD
      YP2=Y1+TTH*DY+WPE(9)*DX/SQD
      TT1=(XP1-X1)/(YP1-Y1)
      TT2=(X2-XP2)/(Y2-YP2)
      DY=(WPE(4)-WPE(1))*HPS
      TL1=(WPE(5)-WPE(2))/DY
      TL2=(WPE(11)-WPE(8))/DY
C
      DT1=(TT1-TL1)/PNVC
      DT2=(TT2-TL2)/PNVC
      TAN1(1)=TL1+DT1/2.
      TAN2(1)=TL2+DT2/2.
      DO 130 I=2,NVC
      TAN1(I)=TAN1(I-1)+DT1
  130 TAN2(I)=TAN2(I-1)+DT2
C
      IS=1
      DO 140 I=1,NVC
      DX=XR(IS)-XR(IS+3)
      DY=YR(IS)-YR(IS+3)
      DZ=ZR(IS)-ZR(IS+3)
      DXDY=DX/DY
      TC1=(TAN1(I)-DXDY)/(1.+DXDY*TAN1(I))/3.
      TC2=(TAN2(I)-DXDY)/(1.+DXDY*TAN2(I))/3.
      XR(IS+2)=XR(IS+3)+DX/3.+DY*TC1
```

```
      YR(IS+2)=YR(IS+3)+DY/3.-DX*TC1
      ZR(IS+2)=ZR(IS+3)+DZ/3.
      XR(IS+1)=XR(IS+3)+TTH*DX-DY*TC2
      YR(IS+1)=YR(IS+3)+TTH*DY+DX*TC2
      ZR(IS+1)=ZR(IS+3)+TTH*DZ
  140 IS=IS+4
C*               CALC XR,YR,ZR FOR TRAILING VORTICES
      DT1=TAN1(NVC)/WNVT
      DT2=TAN2(NVC)/WNVT
      TAN1(1)=TAN1(NVC)-DT1
      TAN2(1)=TAN2(NVC)-DT2
      DO 150 I=2,NVT
      TAN1(I)=TAN1(I-1)-DT1
  150 TAN2(I)=TAN2(I-1)-DT2
C
      DO 160 I=1,NVT
      DX=XR(IS)-XR(IS+3)
      DY=YR(IS)-YR(IS+3)
      DZ=ZR(IS)-ZR(IS+3)
      DXDY=DX/DY
      TC1=(TAN1(I)-DXDY)/(1.+DXDY*TAN1(I))/3.
      TC2=(TAN2(I)-DXDY)/(1.+DXDY*TAN2(I))/3.
      XR(IS+2)=XR(IS+3)+DX/3.+DY*TC1
      YR(IS+2)=YR(IS+3)+DY/3.-DX*TC1
      ZR(IS+2)=ZR(IS+3)+DZ/3.
      XR(IS+1)=XR(IS+3)+TTH*DX-DY*TC2
      YR(IS+1)=YR(IS+3)+TTH*DY+DX*TC2
      ZR(IS+1)=ZR(IS+3)+TTH*DZ
  160 IS=IS+4
C
C*               CALC CONTROL POINTS AT SPECIFIED WJC AND WJS
      IS=0
      DO 180 IY=1,NVS
      IE=IY-NEVP
      IF(IE.LE.0) GO TO 180
      DO 170 IJC=1,NJC
      IX=PJC(IJC)
      IC=4*IX+IE
      IS=IS+1
      XD(IS)=(XR(IC)+XR(IC+1)+XR(IC-4)+XR(IC-3))/4.
      YD(IS)=(YR(IC)+YR(IC+1)+YR(IC-4)+YR(IC-3))/4.
  170 ZD(IS)=(ZR(IC)+ZR(IC+1)+ZR(IC-4)+ZR(IC-3))/4.
  180 CONTINUE
```

```
      NTD=IS
C*                 CALC CHORDS FOR WING ROOT SECTION
      WPE(3)=SQRT((WPE(2)-XR(4))**2+(WPE(1)*HPS-YR(4))**2)+WPE(2)
      WPE(6)=SQRT((WPE(5)-XR(3))**2+(WPE(4)*HPS-YR(3))**2)+WPE(5)
      WPE(9)=SQRT((WPE(8)-XR(2))**2+(WPE(7)*HPS-YR(2))**2)+WPE(8)
      IS=4
      DO 190 I=2,NVC
      IS=IS+4
      WPE(3)=SQRT((XR(IS-4)-XR(IS))**2+(YR(IS-4)-YR(IS))**2)+WPE(3)
      WPE(6)=SQRT((XR(IS-5)-XR(IS-1))**2+(YR(IS-5)-YR(IS-1))**2)+WPE(6)
  190 WPE(9)=SQRT((XR(IS-6)-XR(IS-2))**2+(YR(IS-6)-YR(IS-2))**2)+WPE(9)
      WPE(3)=SQRT((XR(IS)-X1)**2+(YR(IS)-Y1)**2)+WPE(3)
      WPE(6)=SQRT((XR(IS-1)-XP1)**2+(YR(IS-1)-YP1)**2)+WPE(6)
      WPE(9)=SQRT((XR(IS-2)-XP2)**2+(YR(IS-2)-YP2)**2)+WPE(9)
C*                                 COMPRESS. EFFECT ON PYLON ROOT PARAMETERS
      IF(BETA.EQ.1.) GO TO 200
      ZCON=ZCON*BETA
      YCON=YCON*BETA
      IS=4*(NVC+NVT)
      DO 192 I=1,IS
      YR(I)=YR(I)*BETA
  192 ZR(I)=ZR(I)*BETA
      DO 194 I=1,NTD
      YD(I)=YD(I)*BETA
  194 ZD(I)=ZD(I)*BETA
C    *             FIND X LEADING AND CHORD AT ETA V
  200 CALL WINGD (NVSH,EV,XLV,XTLV,WPE)
      HPS=HPS*BETA
      RETURN
      END
*DECK PTZCS
      SUBROUTINE PTZCS
C
C*                 CALC. THICKNESS SLOPESFROM DEFLECTIONS (Z/C).
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION   PJC(18)    ,ED(18)     ,XOCD(19)   ,TZC(342)
C
      EQUIVALENCE       (DA(3001),XOCD) ,(DA(3021),ED)   ,(DA(3040),TZC)
     1                  ,(DA(2512),PJC)
C
      COMMON/CPP/ HPS,NVSH,DP1(5),NVC,NVS,NJC
```

```
C
      LARGE       DM1(75130)
     1            ,EV(22)     ,XLV(22)    ,XTLV(22)   ,WPE(39)
     2            ,DP2(1380)  ,CFI(20)    ,CFJ(20)
     3            ,DP3(506)   ,CSJ(20,20) ,TSJ(20,20)
     4            ,DP4(2390)  ,TSI(21,20) ,TZI(21,20) ,CEL(3700)
     5            ,CHD(20)    ,ZD(20)     ,ZCJ(20)
     6            ,ZCI(20)    ,ZJ(20,20)  ,ZI(20,20)
     7            ,EL(20)     ,XOCL(20)
C
C                 INITIAL
      LD=0
      NDE=DA(3020)
      NXD=DA(3000)
C                 SHIFT TO LCM FOR WINGD AND CODIM
      DO 10 I=1,NDE
   10 EL(I)=ED(I)
      DO 20 I=1,NXD
   20 XOCL(I)=XOCD(I)
C
      CALL WINGD (NDE,EL,ZD,CHD,WPE)
      NI1=NVC+1
      CFJ(NI1)=1.
      DO 100 I=1,NDE
   30 DO 40 IX=1,NXD
      LD=LD+1
   40 ZD(IX)=TZC(LD)*CHD(I)
      CALL CODIM (XOCL,ZD,NXD,CFJ,ZCJ,NI1)
      CALL CODIM (XOCL,ZD,NXD,CFI,ZCI,NVC)
      DO 60 K=1,NVC
      ZJ(K,I)=ZCJ(K)
   60 ZI(K,I)=ZCI(K)
      ZJ(NI1,I)=ZCJ(NI1)
  100 CONTINUE
C
C                 LINEAR INTERPOLATION FOR Z/C SPANWISE AND CALC SLOPE
      DO 200 IE=1,NVS
      IF(EV(IE).GT.ED(1)) GO TO 120
      II=1
      GO TO 140
  120 DO 130 IS=2,NDE
      II=IS
      IF(EV(IE)-ED(IS)) 150,140,130
```

```
  130 CONTINUE
  140 DO 145 J=1,NJC
      IC=PJC(J)+1.
  145 TSJ(IE,J)=(ZI(IC,II)-ZI(IC-1,II))/(CFI(IC)-CFI(IC-1))/XTLV(IE)
      GO TO 200
  150 RATY=(ED(II)-EV(IE))/(ED(II)-ED(II-1))
  160 DO 170 J=1,NJC
      IC=PJC(J)+1.
      Z1=ZI(IC-1,II)-RATY*(ZI(IC-1,II)-ZI(IC-1,II-1))
      Z2=ZI(IC,II)-RATY*(ZI(IC,II)-ZI(IC,II-1))
  170 TSJ(IE,J)=(Z2-Z1)/(CFI(IC)-CFI(IC-1))/XTLV(IE)
  200 CONTINUE
      NEVH=NVSH-1
      DO 300 IE=1,NEVH
      IF(EV(IE).GT.ED(1)) GO TO 220
      II=1
      GO TO 240
  220 DO 230 IS=2,NDE
      II=IS
      IF(EV(IE)-ED(IS)) 250,240,230
  230 CONTINUE
  240 DO 245 I=1,NVC
      TZI(IE,I)=ZI(I,II)
  245 TSI(IE,I)=(ZJ(I+1,II)-ZJ(I,II))/(CFJ(I+1)-CFJ(I))
      GO TO 300
  250 RATY=(ED(II)-EV(IE))/(ED(II)-ED(II-1))
  260 DO 270 I=1,NVC
      Z1=ZJ(I,II)-RATY*(ZJ(I,II)-ZJ(I,II-1))
      Z2=ZJ(I+1,II)-RATY*(ZJ(I+1,II)-ZJ(I+1,II-1))
      TZI(IE,I)=ZI(I,II)-RATY*(ZI(I,II)-ZI(I,II-1))
  270 TSI(IE,I)=(Z2-Z1)/(CFJ(I+1)-CFJ(I))
  300 CONTINUE
      LT=NVSH
      DO 320 I=1,NVC
      TZI(LT,I)=ZI(I,NDE)
  320 TSI(LT,I)=(ZJ(I+1,NDE)-ZJ(I,NDE))/(CFJ(I+1)-CFJ(I))
      RETURN
      END
*DECK NXYZ
      SUBROUTINE NXYZ
C
C*                 SETUP NACELLE COORDINATES AND PARAMETERS
C
```

```
      COMMON/DAT/ DA(5000)
C
      DIMENSION  XB1(1),YB1(1),XB2(1),YB2(1)
C
      EQUIVALENCE      (DA(3400),XB1)   ,(DA(3550),YB1)  ,(DA(3700),XB2)
     1,(DA(3850),YB2)  ,(ND(1),NP1)     ,(ND(2),NP2)
C
      COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP
      COMMON/CFG/ ID1(5),NQB,NQF,NQW,NQP,NQN
      COMMON/CPN/ NB,ND(3),NT
C
      LARGE        DM1(81191),DM2(3000)
     1             ,X1(140)    ,Y1(140)    ,SINAN(140),COSAN(140),DELS(140)
C
C
      ND(NB+1)=NQB+NQF+NQW+NQP
      DO 10 I=1,NP1
      X1(I)=XB1(I)
   10 Y1(I)=YB1(I)*BETA
C
      IF(NB.EQ.1) GO TO 30
      IC=NP1
      DO 20 I=1,NP2
      IC=IC+1
      X1(IC)=XB2(I)
   20 Y1(IC)=YB2(I)*BETA
C
   30 N1=0
      J1=0
      DO 50 K=1,NB
      M1=N1+1
      N1=N1+ND(K)-1
      DO 40 J=M1,N1
      J1=J1+1
      T1=X1(J1+1)-X1(J1)
      T2=Y1(J1+1)-Y1(J1)
      DELS(J)=SQRT(T1*T1+T2*T2)
      COSAN(J)=T1/DELS(J)
   40 SINAN(J)=T2/DELS(J)
   50 J1=J1+1
C
      RETURN
      END
```

```
*DECK CODMT
      SUBROUTINE CODMT (NXS,NTY,NVB,XI,YI,TH,RZ)
C
C*                   A CONTROLLED DEVIATION ITERPOLATION METHOD
C*                   WITH MULTIPLE TABLES
C
      DIMENSION  XI(1),YI(1)
C
      LARGE      TH(1),RZ(1)
C
      DATA XK/0.5/
C*                   START
      N=NTY
      NM3=N-3
      IC=0
      DO 400 IT=1,NVB
      X=TH(IT)
      J=1
      IF(N-2) 10,30,50
   10 DO 20 IX=1,NXS
      IC=IC+1
      RZ(IC)=YI(J)
   20 J=J+N
      GO TO 400
   30 RXI=(X-XI(1))/(XI(2)-XI(1))
      DO 40 IX=1,NXS
      IC=IC+1
      RZ(IC)=(YI(J+1)-YI(J))*RXI+YI(J)
   40 J=J+N
      GO TO 400
   50 IF(XI(J)-X) 60,10,70
   60 J=J+1
      IF(J-N) 50,50,70
   70 IF(J-2) 10,80,90
   80 J=3
      JJ=1
      GO TO 120
   90 IF(J-N) 100,110,10
  100 JJ=3
      GO TO 120
  110 J=N-1
      JJ=2
  120 IF(NM3) 130,130,140
```

```
130 J=3
140 K=J-1
    M=K-1
    L=J+1
    A1=X-XI(M)
    A2=X-XI(K)
    A3=X-XI(J)
    C1=A3*A2/((XI(M)-XI(K))*(XI(M)-XI(J)))
    C2=A1*A3/((XI(K)-XI(M))*(XI(K)-XI(J)))
    C3=A2*A1/((XI(J)-XI(M))*(XI(J)-XI(K)))
    IF(NM3) 160,160,150
150 A4=X-XI(L)
    C4=A4*A3/((XI(K)-XI(J))*(XI(K)-XI(L)))
    C5=A2*A4/((XI(J)-XI(K))*(XI(J)-XI(L)))
    C6=A3*A2/((XI(L)-XI(K))*(XI(L)-XI(J)))
160 IF(JJ-2) 170,180,190
170 AL=(X-XI(1))/(XI(2)-XI(1))
    GO TO 200
180 AL=(X-XI(N-1))/(XI(N)-XI(N-1))
    GO TO 200
190 AL=(X-XI(K))/(XI(J)-XI(K))
200 DO 300 IX=1,NXS
    P1=C1*YI(M)+C2*YI(K)+C3*YI(J)
    IF(NM3) 210,210,220
210 P2=P1
    GO TO 230
220 P2=C4*YI(K)+C5*YI(J)+C6*YI(L)
230 IF(JJ-2) 240,250,260
240 P2=P1
    IA=(IX-1)*N+1
    S=AL*YI(IA+1)+(1.-AL)*YI(IA)
    P1=S+XK*(P2-S)
    GO TO 270
250 P1=P2
    IA=IX*N
    S=AL*YI(IA)+(1.-AL)*YI(IA-1)
    P2=S+XK*(P1-S)
    GO TO 270
260 S=AL*YI(J)+(1.-AL)*YI(K)
270 E1=ABS(P1-S)
    E2=ABS(P2-S)
    IC=IC+1
    IF(E1+E2) 280,280,290
```

```
  280 RZ(IC)=S
      GO TO 295
  290 BT=(E1*AL)/(E1*AL+(1.-AL)*E2)
      RZ(IC)=BT*P2+(1.-BT)*P1
  295 J=J+N
      K=K+N
      L=L+N
      M=M+N
  300 CONTINUE
  400 CONTINUE
      RETURN
      END
*DECK MATA
      SUBROUTINE MATA
C
C*                CONTROL ROUTINE FOR THE CALC OF THE INFLUENCE MATRIX-A
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION  WJS(1),PJC(1),XB1(1),YB1(1),XB2(1),YB2(1)
C
      EQUIVALENCE      (DA(1202),WOZ)   ,(DA(1690),WJS)  ,(DA(2501),POX)
     1,(DA(2502),POY)  ,(DA(2503),POZ)   ,(DA(2517),PJC)  ,(DA(3391),XNO)
     2,(DA(3392),YNO)  ,(DA(3393),ZNO)   ,(DA(3400),XB1)  ,(DA(3550),YB1)
     3,(DA(3700),XB2)  ,(DA(3850),YB2)
C
      COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP
      COMMON/CFG/ IDB,IDF,IDW,IDP,IDN,NQB,NQF,NQW,NQP,NQN
      COMMON/CPW/ HWS,IW1(5),NE3,NE5,IW2(5),NJC,NJS
      COMMON/CPP/ HPS,IP1(3),NEVP,IP2(3),NVSP,NJCP
      COMMON/CPN/ NB,NP1,NP2
C
      LARGE       DM1(3000)  ,XQB(650)   ,YQB(650)   ,ZQB(650)   ,DM2(23050)
     1            ,XQF(650)   ,YQF(650)   ,ZQF(650)   ,DM3(21750)
     2            ,EV(52)     ,XLV(52)    ,XTLV(52)
     3            ,XD(270)    ,YD(270)    ,ZD(270)    ,DM4(740)
     4            ,EP(30)     ,XTLP(30)   ,XOCP(30)   ,DM5(21634)
     5            ,EVP(22)    ,XLVP(22)   ,XTLVP(22)   ,WPE(39)
     6            ,XDP(60)    ,YDP(60)    ,ZDP(60)    ,DM6(1220)  ,CFJP(20)
     7            ,DM7(4536)  ,XOB(1000)  ,YOB(1000)  ,ZOB(1000)  ,DM8(700)
     8            ,XA(650)    ,YA(650)    ,ZA(650)
C
C*                INITIAL
```

```
      ICN=0
      I19=2
      I20=2
      I21=2
      I22=2
C
C*                 CONTROL POINTS Q ON FUSELAGE
      IF(IDB.EQ.0) GO TO 100
      DO 30 I=1,NQB
      XA(I)=XQB(I)
      YA(I)=YQB(I)
   30 ZA(I)=ZQB(I)
C                  CALC INFL OF FUSELAGE
      IF(IDB.EQ.1) GO TO 44
      DO 42 I=1,NQB
   42 READ (19)
      GO TO 50
   44 I19=1
      CALL AQXVB (NQB,1)
C                  CALC INFL OF FANPOD
   50 IF(IDF.EQ.0) GO TO 60
      IDX=IDB+IDF
      IF(IDX.NE.4) GO TO 54
      DO 52 I=1,NQB
   52 READ (20)
      GO TO 60
   54 I20=1
      CALL AQXVF(NQB,2)
C                  CALC INFL OF WING
   60 IF(IDW.EQ.0) GO TO 70
      IDX=IDB+IDW
      IF(IDX.NE.4) GO TO 64
      DO 62 I=1,NQB
   62 READ (21)
      GO TO 70
   64 I21=1
      CALL AQXVW(NQB,3)
C                  CALC INFL OF PYLON
   70 IF(IDP.EQ.0) GO TO 80
      IDX=IDB+IDP
      IF(IDX.NE.4) GO TO 74
      DO 72 I=1,NQB
   72 READ (22)
```

```
      GO TO 80
   74 I22=1
      CALL AQXVP(NQB,4)
C                SETUP NACELLE OFF BODY POINTS
   80 IF(IDN.EQ.0) GO TO 100
      DO 82 I=1,NQB
      ICN=ICN+1
      XOB(ICN)=XA(I)
      YOB(ICN)=YA(I)
   82 ZOB(ICN)=ZA(I)
C
C*               CONTROL POINTS Q ON FANPOD
  100 IF(IDF.EQ.0) GO TO 200
      DO 130 I=1,NQF
      XA(I)=XQF(I)
      YA(I)=YQF(I)
  130 ZA(I)=ZQF(I)
C                CALC INFL OF FUSELAGE
      IF(IDB.EQ.0) GO TO 150
      IDX=IDF+I19+IDB
      IF(IDX.NE.6) GO TO 144
      DO 142 I=1,NQF
  142 READ (19)
      GO TO 150
  144 I19=1
      CALL AQXVB(NQF,6)
C                CALC INFL OF FANPOD
  150 IDX=IDF+I20
      IF(IDX.NE.4) GO TO 154
      DO 152 I=1,NQF
  152 READ (20)
      GO TO 160
  154 I20=1
      CALL AQXVF(NQF,7)
C                CALC INFL OF WING
  160 IF(IDW.EQ.0) GO TO 170
      IDX=IDF+I21+IDW
      IF(IDX.NE.6) GO TO 164
      DO 162 I=1,NQF
  162 READ (21)
      GO TO 170
  164 I21=1
      CALL AQXVW(NQF,8)
```

```
C                  CALC INFL OF PYLON
  170 IF(IDP.EQ.0) GO TO 180
      IDX=IDF+I22+IDP
      IF(IDX.NE.6) GO TO 174
      DO 172 I=1,NQF
  172 READ (22)
      GO TO 180
  174 I22=1
      CALL AQXVP(NQF,9)
C                  SETUP NACELLE OFF BODY POINTS
  180 IF(IDN.EQ.0) GO TO 200
      DO 182 I=1,NQF
      ICN=ICN+1
      XOB(ICN)=XA(I)
      YOB(ICN)=YA(I)
  182 ZOB(ICN)=ZA(I)
C
C*                 CONTROL POINTS Q ON WING
  200 IF(IDW.EQ.0) GO TO 300
      IR=0
      IQ=0
      NE4=NE3+7
      NET=NE3+4
      NE6=NE5+1
      DO 220 I=1,NJS
      IE=WJS(I)
      Y=HWS*EP(I)
      DO 220 J=1,NJC
      IQ=IQ+1
      IF(IDF.EQ.0) GO TO 210
      IF(IE.LE.NE3) GO TO 214
      IF(IE.GE.NE4) GO TO 210
      IF(IE.EQ.NET.AND.J.EQ.1) IR=90
      GO TO 212
  210 IF(IE.LE.NE5) GO TO 214
      IF(IE.EQ.NE6.AND.J.EQ.1) IR=180
  212 IR=IR+1
      XA(IQ)=XD(IR)
      YA(IQ)=YD(IR)
      ZA(IQ)=ZD(IR)
      GO TO 220
  214 XA(IQ)=XLV(IE)+XTLV(IE)*XOCP(J)
      YA(IQ)=Y
```

```
      ZA(IQ)=WOZ*BETA
  220 CONTINUE
C                CALC INFL OF FUSELAGE
      IF(IDB.EQ.0) GO TO 250
      IDX=IDW+I19+IDB
      IF(IDX.NE.6) GO TO 244
      DO 242 I=1,NQW
  242 READ (19)
      GO TO 250
  244 I19=1
      CALL AQXVB(NQW,11)
C                CALC INFL OF FANPOD
  250 IF(IDF.EQ.0) GO TO 260
      IDX=IDW+I20+IDF
      IF(IDX.NE.6) GO TO 254
      DO 252 I=1,NQW
  252 READ (20)
      GO TO 260
  254 I20=1
      CALL AQXVF(NQW,12)
C                CALC INFL OF WING
  260 IDX=IDW+I21
      IF(IDX.NE.4) GO TO 264
      DO 262 I=1,NQW
  262 READ (21)
      GO TO 270
  264 I21=1
      CALL AQWVW(13)
C                CALC INFL OF PYLON
  270 IF(IDP.EQ.0) GO TO 280
      IDX=IDW+I22+IDP
      IF(IDX.NE.6) GO TO 274
      DO 272 I=1,NQW
  272 READ (22)
      GO TO 280
  274 I22=1
      CALL AQXVP(NQW,14)
C                SETUP NACELLE OFF BODY POINTS
  280 IF(IDN.EQ.0) GO TO 300
      DO 282 I=1,NQW
      ICN=ICN+1
      XOB(ICN)=XA(I)
      YOB(ICN)=YA(I)
```

```
  282 ZOB(ICN)=ZA(I)
C
C*                 CONTROL POINTS Q ON PYLON
  300 IF(IDP.EQ.0) GO TO 400
      IR=0
      IQ=0
      PYB=POY*BETA
      PZB=POZ*BETA
      DO 320 I=1,NVSP
      Y=HPS*EVP(I)
      YC=PYB+Y*SINDP
      ZC=PZB+Y*COSDP
      DO 320 J=1,NJCP
      IJ=PJC(J)
      IQ=IQ+1
      IF(I.LE.NEVP) GO TO 310
      IR=IR+1
      XA(IQ)=XDP(IR)+POX
      YA(IQ)=(YDP(IR)*SINDP-ZDP(IR)*COSDP)+PYB
      ZA(IQ)=(YDP(IR)*COSDP+ZDP(IR)*SINDP)+PZB
      GO TO 320
  310 XA(IQ)=XLVP(I)+XTLVP(I)*CFJP(IJ+1)+POX
      YA(IQ)=YC
      ZA(IQ)=ZC
  320 CONTINUE
C                  CALC INFL OF FUSELAGE
      IF(IDB.EQ.0) GO TO 350
      IDX=IDP+I19+IDB
      IF(IDX.NE.6) GO TO 344
      DO 342 I=1,NQP
  342 READ (19)
      GO TO 350
  344 I19=1
      CALL AQXVB(NQP,16)
C                  CALC INFL OF FANPOD
  350 IF(IDF.EQ.0) GO TO 360
      IDX=IDP+I20+IDF
      IF(IDX.NE.6) GO TO 354
      DO 352 I=1,NQP
  352 READ (20)
      GO TO 360
  354 I20=1
      CALL AQXVF(NQP,17)
```

```
C                   CALC INFL OF WING
  360 IF(IDW.EQ.0) GO TO 370
      IDX=IDP+I21+IDW
      IF(IDX.NE.6) GO TO 364
      DO 362 I=1,NQP
  362 READ (21)
      GO TO 370
  364 I21=1
      CALL AQXVW(NQP,18)
C                   SETUP NACELLE OFF BODY POINTS
  370 IF(IDN.EQ.0) GO TO 380
      DO 372 I=1,NQP
      ICN=ICN+1
      XOB(ICN)=XA(I)
      YOB(ICN)=YA(I)
  372 ZOB(ICN)=ZA(I)
C                   CALC INFL OF PYLON
  380 IDX=IDP+I22
      IF(IDX.NE.4) GO TO 384
      DO 382 I=1,NQP
  382 READ (22)
      GO TO 400
  384 I22=1
      CALL AQPVP
      CALL AQXVP(NQP,19)
C
C*                  CONTROL POINTS Q ON NACELLE
  400 IF(IDN.EQ.0) GO TO 500
      YNOB=YNO*BETA
      ZNOB=ZNO*BETA
      DO 410 I=2,NP1
      XA(I-1)=XNO+(XB1(I)+XB1(I-1))/2.
      YN=(YB1(I)+YB1(I-1))/2.*BETA
      YA(I-1)=YNOB-YN*SINDP
  410 ZA(I-1)=ZNOB-YN*COSDP
      IF(NB.EQ.1) GO TO 430
      IC=NP1-1
      DO 420 I=2,NP2
      IC=IC+1
      XA(IC)=XNO+(XB2(I)+XB2(I-1))/2.
      YN=(YB2(I)+YB2(I-1))/2.
      YA(IC)=YNOB-YN*SINDP
  420 ZA(IC)=ZNOB-YN*COSDP
```

```
C                  SETUP NACELLE OFF BODY POINTS
  430 DO 440 I=1,NQN
      ICN=ICN+1
      XOB(ICN)=XA(I)
      YOB(ICN)=YA(I)
  440 ZOB(ICN)=ZA(I)
C                  CALC INFL OF FUSELAGE
      IF(IDB.EQ.0) GO TO 450
      IDX=IDN+I19+IDB
      IF(IDX.NE.6) GO TO 444
      DO 442 I=1,NQN
  442 READ (19)
      GO TO 450
  444 I19=1
      CALL AQXVB(NQN,21)
C                  CALC INFL OF FANPOD
  450 IF(IDF.EQ.0) GO TO 460
      IDX=IDN+I20+IDF
      IF(IDX.NE.6) GO TO 454
      DO 452 I=1,NQN
  452 READ (20)
      GO TO 460
  454 I20=1
      CALL AQXVF(NQN,22)
C                  CALC INFL OF WING
  460 IF(IDW.EQ.0) GO TO 470
      IDX=IDN+I21+IDW
      IF(IDX.NE.6) GO TO 464
      DO 462 I=1,NQN
  462 READ (21)
      GO TO 470
  464 I21=1
      CALL AQXVW(NQN,23)
C                  CALC INFL OF PYLON
  470 IF(IDP.EQ.0) GO TO 500
      IDX=IDN+I22+IDP
      IF(IDX.NE.6) GO TO 474
      DO 472 I=1,NQN
  472 READ (22)
      GO TO 500
  474 I22=1
      CALL AQXVP(NQN,24)
C
```

```
C*              REIND I/O UNITS
  500 IF(IDB.NE.0) REWIND 19
      IF(IDF.NE.0) REWIND 20
      IF(IDW.NE.0) REWIND 21
      IF(IDP.NE.0) REWIND 22
C
C*              CALC INFL OF NACELLE ON OTHER COMPONENTS
      IF(IDN.NE.0) CALL NACELL
C
C*              COMBINE INFLUENCE MATRICES AND SETUP FOR MATRIX SOLU.
      CALL MATAPE
C
      RETURN
      END
*DECK AQXVB
      SUBROUTINE AQXVB(NQ,ITYP)
C
C*              CALC MATRIX-A (Q ON X-COMPONENT AND VORTICES ON FUSE.)
C
      COMMON/DAT/ DA(5000)
C
      EQUIVALENCE      (DA(3386),XLFX),(DA(3387),XLFY),(DA(4707),FNDV)
C
      COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP
      COMMON/CFG/ IG1(10),NKB
      COMMON/CPB/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX,IFBE
C
      LARGE       DM1(10800)
     *            ,PM(150,25),DFM(150)   ,DM2(1300)
     *            ,XB(4000)   ,YB(4000)   ,ZB(4000)   ,DM3(56891)
     1            ,XQ(650)    ,YQ(650)    ,ZQ(650)
     2            ,AX(250)    ,AY(250)    ,AZ(250)
     3            ,QBX(51)    ,QBY(51)    ,QBZ(51)
     4            ,QBXP(51)   ,QBYP(51)   ,QBZP(51)
     5            ,QTX(51)    ,QTY(51)    ,QTZ(51)
     6            ,DL(51)     ,DLP(51)    ,SL(51)     ,PLM(51,25)
C
C*              START
      XLFY2=XLFY*XLFY
C               NO. OF CONTROL POINTS Q LOOP
      DO 400 IQ=1,NQ
C               CLEAR SUMMATION MATRICES
      IF(NF.NE.0) GO TO 11
```

```
      DO 10 I=1,NVY
      QBXP(I)=0.0
      QBYP(I)=0.0
   10 QBZP(I)=0.0
      GO TO 14
   11 DO 12 I=1,NKB
      AX(I)=0.0
      AY(I)=0.0
   12 AZ(I)=0.0
C                  NVX AND NVY LOOPS
   14 IPAS=0
      DO 300 IX=1,NVXM
      LI=(IX-1)*NVBP+2
      LF=LI+1
      NVYX=NVY+NVY
C                  TEST TO BYPASS X-SEGMENT
      IF(IQ.EQ.1) GO TO 15
      IF(IX.GE.(NVXM-1)) GO TO 15
      TT=ABS(XQ(IQ)-XB(LI))/DFM(IX)
      IF(TT.GT.XLFX) GO TO 270
   15 DZ2=(ZB(LI)-ZQ(IQ))**2
      DYB=YB(LI)-YB(LI+1)
      IF(DYB.EQ.0.0) GO TO 16
      DF2=(FNDV*DYB)**2
   16 TT=((-YB(LI)-YQ(IQ))**2+DZ2)/DF2
      IF(TT.GT.XLFY2) NVYX=NVY
      DO 180 IS=1,NVYX
      K=NDV
      IF(IX.NE.NVXM) GO TO 18
      K=0
      LF=LI+NDV
      LI=LF-NVBP
   18 ISS=IS-NVY
      IF(ISS) 20,20,32
   20 JS=IS
      QBX(JS)=0.0
      QBY(JS)=0.0
      QBZ(JS)=0.0
   30 TY1=YB(LF)-YB(LI)
      TY2=YB(LF)-YQ(IQ)
      TY3=YB(LI)-YQ(IQ)
      GO TO 50
   32 JS=ISS
```

```
   40 TY1=YB(LI)-YB(LF)
      TY2=-YB(LF)-YQ(IQ)
      TY3=-YB(LI)-YQ(IQ)
   50 TX1=XB(LF)-XB(LI)
      TX2=XB(LF)-XQ(IQ)
      TX3=XB(LI)-XQ(IQ)
      TZ1=ZB(LF)-ZB(LI)
      TZ2=ZB(LF)-ZQ(IQ)
      TZ3=ZB(LI)-ZQ(IQ)
      HN=TX2*TX1+TY2*TY1+TZ2*TZ1
      HD=TX1**2+TY1**2+TZ1**2
      SQHD=SQRT(HD)
      IF(SQHD.GT.0.00001) GO TO 52
   51 QSR=0.0
      GO TO 54
   52 HT=HN/HD
      HX=TX2-HT*TX1
      HY=TY2-HT*TY1
      HZ=TZ2-HT*TZ1
      SHP=SQRT(HX**2+HY**2+HZ**2)
      IF(SHP.LT.0.00001) GO TO 51
      SRF=HN/SQRT(TX2**2+TY2**2+TZ2**2)
      HI=TX1*TX3+TY1*TY3+TZ1*TZ3
      SRI=HI/SQRT(TX3**2+TY3**2+TZ3**2)
      QT=(SRF-SRI)/(SQHD*SHP*PI4)
      TX=TY2*TZ1-TZ2*TY1
      TY=TZ2*TX1-TX2*TZ1
      TZ=TX2*TY1-TY2*TX1
      QSR=QT/SQRT(TX**2+TY**2+TZ**2)
      IF(ISS.GT.0) QSR=-QSR
   54 K=K-1
      IF(K) 140,60,60
C*                 LATERAL VORTICES
   60 QBX(JS)=QSR*TX+QBX(JS)
      QBY(JS)=QSR*TY+QBY(JS)
      QBZ(JS)=QSR*TZ+QBZ(JS)
      IF(K) 100,100,70
C                  SETUP FOR ADDITIONAL LATERAL
   70 LI=LF
      LF=LF+1
      IF(ISS) 30,30,40
C                  SETUP FOR LONGITUDINAL
  100 LI=LF-NVBP
```

```
      IF(IX.EQ.1) LI=1
      IF(ISS) 30,30,40
C*                 LONGITUDINAL VORTICES
  140 IF(ISS) 160,155,150
  150 QTX(JS)=QSR*TX+QTX(JS)
      QTY(JS)=QSR*TY+QTY(JS)
      QTZ(JS)=QSR*TZ+QTZ(JS)
      GO TO 170
  155 LF=LI+1
  160 QTX(JS)=QSR*TX
      QTY(JS)=QSR*TY
      QTZ(JS)=QSR*TZ
      DL(JS)=SQHD
  170 LI=LF
      LF=LF+1
  180 CONTINUE
      IF(IPAS.EQ.0) GO TO 190
      IPAS=0
      READ (23)
      GO TO 255
  190 IF(IX.NE.NVXM) GO TO 198
      IF(IFBE.EQ.0) GO TO 198
C    *             COMPUTE TRAIL VORTICES
  191 DO 196 I=1,NVY
      PASS=1.
      HY=YB(LI)-YQ(IQ)
      HX=XB(LI)-XQ(IQ)
      HZ=ZB(LI)-ZQ(IQ)
  192 DD1=HY*HY+HZ*HZ
      IF(DD1.LT.0.00001) GO TO 193
      DD2=SQRT(HX*HX+DD1)
      IF(DD2.LT.0.00001) GO TO 193
      QSR=(1.-HX/DD2)/DD1*PASS/PI4
      QTY(JS)=QTY(JS)+QSR*HZ
      QTZ(JS)=QTZ(JS)-QSR*HY
  193 IF(PASS) 195,195,194
  194 PASS=-1.
      HY=-YB(LI)-YQ(IQ)
      GO TO 192
  195 LI=LI-NDV
  196 JS=JS-1
C*                 CALC MATRIX L
  198 IF(NF.LE.0) GO TO 210
```

```
      IF(ITYP.NE.1) GO TO 200
      IF(IQ.EQ.1) GO TO 201
  200 IF(IX.EQ.1) GO TO 255
      READ (23) ((PLM(J,I),J=1,NVY),I=1,NF)
      GO TO 210
  201 DFM(IX)=DL(1)
      IF(IX.NE.1) GO TO 204
      DLP(1)=DL(1)
      DO 202 J=2,NVY
  202 DLP(J)=(DL(J-1)+DL(J))/2.
      DO 203 J=1,NVY
      DO 203 I=1,NF
  203 PLM(J,I)=0.0
      GO TO 255
  204 DLX=DL(1)/2.
      SL(1)=DLP(1)+DLX
      DLP(1)=DLX
      DO 205 J=2,NVY
      DLX=(DL(J-1)+DL(J))/4.
      SL(J)=DLP(J)+DLX
  205 DLP(J)=DLX
      IF(IX.NE.NVXM) GO TO 207
      DO 206 J=1,NVY
  206 SL(J)=SL(J)+DLP(J)
  207 DO 208 J=1,NVY
      DO 208 I=1,NF
  208 PLM(J,I)=PM(IX,I)*SL(J)+PLM(J,I)
      WRITE (23) ((PLM(J,I),J=1,NVY),I=1,NF)
C*                  SUM FOR AX,AY AND AZ
  210 TX=QBXP(1)-QBX(1)+QTX(1)-QTX(NVY)
      TY=QBYP(1)-QBY(1)+QTY(1)-QTY(NVY)
      TZ=QBZP(1)-QBZ(1)+QTZ(1)-QTZ(NVY)
      IF(NF) 212,212,214
  212 AX(IX)=TX
      AY(IX)=TY
      AZ(IX)=TZ
      GO TO 230
  214 DO 220 I=1,NF
      AX(I)=TX*PLM( 1,I)+AX(I)
      AY(I)=TY*PLM( 1,I)+AY(I)
  220 AZ(I)=TZ*PLM( 1,I)+AZ(I)
      IC=NF
  230 DO 250 J=2,NVY
```

```
      TX=QBXP(J)-QBX(J)+QTX(J)-QTX(J-1)
      TY=QBYP(J)-QBY(J)+QTY(J)-QTY(J-1)
      TZ=QBZP(J)-QBZ(J)+QTZ(J)-QTZ(J-1)
      IF(NF) 232,232,234
  232 IC=(J-1)*NVXM+IX
      AX(IC)=TX
      AY(IC)=TY
      AZ(IC)=TZ
      GO TO 250
  234 DO 240 I=1,NF
      IC=IC+1
      AX(IC)=TX*PLM( J,I)+AX(IC)
      AY(IC)=TY*PLM( J,I)+AY(IC)
  240 AZ(IC)=TZ*PLM( J,I)+AZ(IC)
  250 CONTINUE
  255 DO 260 I=1,NVY
      QBXP(I)=QBX(I)
      QBYP(I)=QBY(I)
  260 QBZP(I)=QBZ(I)
      GO TO 300
  270 IPAS=1
      IF(IX.EQ.1) GO TO 300
      READ (23)
  300 CONTINUE
      REWIND 23
C
      IF(ITYP.LT.16) GO TO 320
      DO 310 I=1,NKB
      AYT=AY(I)
      AY(I)=AYT*SINDP+AZ(I)*COSDP
  310 AZ(I)=-AYT*COSDP+AZ(I)*SINDP
C
  320 WRITE (19) (AX(IC),AY(IC),AZ(IC),IC=1,NKB)
  400 CONTINUE
      WRITE (6,450) ITYP
  450 FORMAT (21H0 ** COMPLETED MATRIX,I4)
      RETURN
      END
*DECK AQXVF
      SUBROUTINE AQXVF(NQ,ITYP)
C
C*                 CALC MATRIX-A (Q ON X-COMPONENT AND VORTICES ON FANP.)
C
```

```
      COMMON/DAT/ DA(5000)
      EQUIVALENCE      (DA(3380),XLFX) ,(DA(3381),XLFY) ,(DA(1276),FNDV)
C
      COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP
      COMMON/CFG/ IG1(11),NKF
      COMMON/CPF/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX,IFBE
C
      LARGE       DM1(35800)
     *            ,PM(100,25),DFM(100)   ,DM2(1300)
     *            ,XB(4000)   ,YB(4000)   ,ZB(4000)   ,DM3(33191)
     1            ,XQ(650)    ,YQ(650)    ,ZQ(650)
     2            ,AX(250)    ,AY(250)    ,AZ(250)
     3            ,QBX(51)    ,QBY(51)    ,QBZ(51)
     4            ,QBXP(51)   ,QBYP(51)   ,QBZP(51)
     5            ,QTX(51)    ,QTY(51)    ,QTZ(51)
     6            ,DL(51)     ,DLP(51)    ,SL(51)     ,PLM(51,25)
C
C*                START
      PI4=4.*PI
      XLFY2=XLFY*XLFY
C                 NO. OF CONTROL POINTS Q LOOP
      DO 400 IQ=1,NQ
C                 CLEAR SUMMATION MATRICES
      IF(NF.NE.0) GO TO 11
      DO 10 I=1,NVY
      QBXP(I)=0.0
      QBYP(I)=0.0
   10 QBZP(I)=0.0
      GO TO 14
   11 DO 12 I=1,NKF
      AX(I)=0.0
      AY(I)=0.0
   12 AZ(I)=0.0
C                 NVX AND NVY LOOPS
   14 IPAS=0
      DO 300 IX=1,NVXM
      LI=(IX-1)*NVBP+2
      LF=LI+1
      NVYX=NVY+NVY
C                 TEST TO BYPASS X-SEGMENT
      IF(IQ.EQ.1) GO TO 15
      IF(IX.GE.(NVXM-1)) GO TO 15
      TT=ABS(XQ(IQ)-XB(LI))/DFM(IX)
```

```
      IF(TT.GT.XLFX) GO TO 270
   15 DZ2=(ZB(LI)-ZQ(IQ))**2
      DYB=YB(LI)-YB(LI+1)
      IF(DYB.EQ.0.0) GO TO 16
      DF2=(FNDV*DYB)**2
   16 TT=((-YB(LI)-YQ(IQ))**2+DZ2)/DF2
      IF(TT.GT.XLFY2) NVYX=NVY
      DO 180 IS=1,NVYX
      K=NDV
      IF(IX.NE.NVXM) GO TO 18
      K=0
      LF=LI+NDV
      LI=LF-NVBP
   18 ISS=IS-NVY
      IF(ISS) 20,20,32
   20 JS=IS
      QBX(JS)=0.0
      QBY(JS)=0.0
      QBZ(JS)=0.0
   30 TY1=YB(LF)-YB(LI)
      TY2=YB(LF)-YQ(IQ)
      TY3=YB(LI)-YQ(IQ)
      GO TO 50
   32 JS=ISS
   40 TY1=YB(LI)-YB(LF)
      TY2=-YB(LF)-YQ(IQ)
      TY3=-YB(LI)-YQ(IQ)
   50 TX1=XB(LF)-XB(LI)
      TX2=XB(LF)-XQ(IQ)
      TX3=XB(LI)-XQ(IQ)
      TZ1=ZB(LF)-ZB(LI)
      TZ2=ZB(LF)-ZQ(IQ)
      TZ3=ZB(LI)-ZQ(IQ)
      HN=TX2*TX1+TY2*TY1+TZ2*TZ1
      HD=TX1**2+TY1**2+TZ1**2
      SQHD=SQRT(HD)
      IF(SQHD.GT.0.00001) GO TO 52
   51 QSR=0.0
      GO TO 54
   52 HT=HN/HD
      HX=TX2-HT*TX1
      HY=TY2-HT*TY1
      HZ=TZ2-HT*TZ1
```

```fortran
      SHP=SQRT(HX**2+HY**2+HZ**2)
      IF(SHP.LT.0.00001) GO TO 51
      SRF=HN/SQRT(TX2**2+TY2**2+TZ2**2)
      HI=TX1*TX3+TY1*TY3+TZ1*TZ3
      SRI=HI/SQRT(TX3**2+TY3**2+TZ3**2)
      QT=(SRF-SRI)/(SQHD*SHP*PI4)
      TX=TY2*TZ1-TZ2*TY1
      TY=TZ2*TX1-TX2*TZ1
      TZ=TX2*TY1-TY2*TX1
      QSR=QT/SQRT(TX**2+TY**2+TZ**2)
      IF(ISS.GT.0) QSR=-QSR
   54 K=K-1
      IF(K) 140,60,60
C*                 LATERAL VORTICES
   60 QBX(JS)=QSR*TX+QBX(JS)
      QBY(JS)=QSR*TY+QBY(JS)
      QBZ(JS)=QSR*TZ+QBZ(JS)
      IF(K) 100,100,70
C                  SETUP FOR ADDITIONAL LATERAL
   70 LI=LF
      LF=LF+1
      IF(ISS) 30,30,40
C                  SETUP FOR LONGITUDINAL
  100 LI=LF-NVBP
      IF(IX.EQ.1) LI=1
      IF(ISS) 30,30,40
C*                 LONGITUDINAL VORTICES
  140 IF(ISS) 160,155,150
  150 QTX(JS)=QSR*TX+QTX(JS)
      QTY(JS)=QSR*TY+QTY(JS)
      QTZ(JS)=QSR*TZ+QTZ(JS)
      GO TO 170
  155 LF=LI+1
  160 QTX(JS)=QSR*TX
      QTY(JS)=QSR*TY
      QTZ(JS)=QSR*TZ
      DL(JS)=SQHD
  170 LI=LF
      LF=LF+1
  180 CONTINUE
      IF(IPAS.EQ.0) GO TO 190
      IPAS=0
      READ (24)
```

```
      GO TO 255
  190 IF(IX.NE.NVXM) GO TO 198
      IF(IFBE.EQ.0) GO TO 198
C     *          COMPUTE TRAIL VORTICES
  191 DO 196 I=1,NVY
      PASS=1.
      HY=YB(LI)-YQ(IQ)
      HX=XB(LI)-XQ(IQ)
      HZ=ZB(LI)-ZQ(IQ)
  192 DD1=HY*HY+HZ*HZ
      IF(DD1.LT.0.00001) GO TO 193
      DD2=SQRT(HX*HX+DD1)
      IF(DD2.LT.0.00001) GO TO 193
      QSR=(1.-HX/DD2)/DD1*PASS/PI4
      QTY(JS)=QTY(JS)+QSR*HZ
      QTZ(JS)=QTZ(JS)-QSR*HY
  193 IF(PASS) 195,195,194
  194 PASS=-1.
      HY=-YB(LI)-YQ(IQ)
      GO TO 192
  195 LI=LI-NDV
  196 JS=JS-1
C*               CALC MATRIX L
  198 IF(NF.LE.0) GO TO 210
      IF(ITYP.GT.7) GO TO 200
      IF(IQ.EQ.1) GO TO 201
  200 IF(IX.EQ.1) GO TO 255
      READ (24) ((PLM(J,I),J=1,NVY),I=1,NF)
      GO TO 210
  201 DFM(IX)=DL(1)
      IF(IX.NE.1) GO TO 204
      DLP(1)=(DL(1)+DL(NVY))/2.
      DO 202 J=2,NVY
  202 DLP(J)=(DL(J-1)+DL(J))/2.
      DO 203 J=1,NVY
      DO 203 I=1,NF
  203 PLM(J,I)=0.0
      GO TO 255
  204 DLX=(DL(1)+DL(NVY))/4.
      SL(1)=DLP(1)+DLX
      DLP(1)=DLX
      DO 205 J=2,NVY
      DLX=(DL(J-1)+DL(J))/4.
```

```
      SL(J)=DLP(J)+DLX
  205 DLP(J)=DLX
      IF(IX.NE.NVXM) GO TO 207
      DO 206 J=1,NVY
  206 SL(J)=SL(J)+DLP(J)
  207 DO 208 J=1,NVY
      DO 208 I=1,NF
  208 PLM(J,I)=PM(IX,I)*SL(J)+PLM(J,I)
      WRITE (24) ((PLM(J,I),J=1,NVY),I=1,NF)
C*                 SUM FOR AX,AY AND AZ
  210 TX=QBXP(1)-QBX(1)+QTX(1)-QTX(NVY)
      TY=QBYP(1)-QBY(1)+QTY(1)-QTY(NVY)
      TZ=QBZP(1)-QBZ(1)+QTZ(1)-QTZ(NVY)
      IF(NF) 212,212,214
  212 AX(IX)=TX
      AY(IX)=TY
      AZ(IX)=TZ
      GO TO 230
  214 DO 220 I=1,NF
      AX(I)=TX*PLM( 1,I)+AX(I)
      AY(I)=TY*PLM( 1,I)+AY(I)
  220 AZ(I)=TZ*PLM( 1,I)+AZ(I)
      IC=NF
  230 DO 250 J=2,NVY
      TX=QBXP(J)-QBX(J)+QTX(J)-QTX(J-1)
      TY=QBYP(J)-QBY(J)+QTY(J)-QTY(J-1)
      TZ=QBZP(J)-QBZ(J)+QTZ(J)-QTZ(J-1)
      IF(NF) 232,232,234
  232 IC=(J-1)*NVXM+IX
      AX(IC)=TX
      AY(IC)=TY
      AZ(IC)=TZ
      GO TO 250
  234 DO 240 I=1,NF
      IC=IC+1
      AX(IC)=TX*PLM( J,I)+AX(IC)
      AY(IC)=TY*PLM( J,I)+AY(IC)
  240 AZ(IC)=TZ*PLM( J,I)+AZ(IC)
  250 CONTINUE
  255 DO 260 I=1,NVY
      QBXP(I)=QBX(I)
      QBYP(I)=QBY(I)
  260 QBZP(I)=QBZ(I)
```

```
      GO TO 300
  270 IPAS=1
      IF(IX.EQ.1) GO TO 300
      READ (24)
  300 CONTINUE
      REWIND 24
C
      IF(ITYP.LT.17) GO TO 320
      DO 310 I=1,NKF
      AYT=AY(I)
      AY(I)=AYT*SINDP+AZ(I)*COSDP
  310 AZ(I)=-AYT*COSDP+AZ(I)*SINDP
C
  320 WRITE (20) (AX(IC),AY(IC),AZ(IC),IC=1,NKF)
  400 CONTINUE
      WRITE (6,450) ITYP
  450 FORMAT (21H0 ** COMPLETED MATRIX,I4)
      RETURN
      END
*DECK AQXVW
      SUBROUTINE AQXVW(NQ,ITYP)
C
C*               CALC MATRIX-A (Q ON X-COMPONENT AND VORTICES ON WING)
C
      COMMON/DAT/ DA(5000)
C
      EQUIVALENCE                       (DA(1281),WNW)  ,(DA(1270),WNVC)
     1,(DA(1202),WOZ)  ,(DA(3382),XLW)
C
      COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP
      COMMON/CFG/ IDB,IDF,IDD(10),NKW
      COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3
     1            ,NE5,NTB,NU,NW,NVC,NVS,NJC,NJS
C
      LARGE       DM1(51700)
     1            ,EV(52)     ,XLV(52)    ,XTLV(52)    ,DM2(8840)
     2            ,YB(600)    ,YBI(600)   ,YBR(600)    ,DM3(1800)
     3            ,CFI(50)    ,CFJ(50)    ,STH(50)     ,ACT(50)
     4            ,ET(52)     ,XLT(52)    ,XTLT(52)
     5            ,C(40,40)   ,S(51,10)   ,DM4(400)    ,TSI(52,40),DM5(2080)
     6            ,COSTT(52)  ,DZLE(52)   ,COSTV(52)   ,SINTV(52)
     7            ,XB(600)    ,XBI(600)   ,XBR(600)
     8            ,ZB(600)    ,ZBI(600)   ,ZBR(600)    ,DM6(9761)
```

```
     9              ,XQ(650)    ,YQ(650)    ,ZQ(650)
     *              ,AX(250)    ,AY(250)    ,AZ(250)
     *              ,SEU(30)    ,SEV(30)    ,SEW(30)
C
C*                  INITIAL AND CALC MATRIX A
      XLW2=4.*XLW*XLW
      NE1=NE2
      NVSH=NE2+1
      NE4=NE3+7
      NE6=NE3+3
      INDT=DA(2000)
      IF(INDT.NE.0) NE1=NVSH
      DTH=PI/WNVC
      DO 300 IQ=1,NQ
      DO 12 I=1,NKW
      AX(I)=0.0
      AY(I)=0.0
   12 AZ(I)=0.0
      IF(IDF.EQ.0) GO TO 14
      CALL AQXVWR(IQ,NTO,NE3,XB,YB,ZB)
      CALL AQXVWR(IQ,NTI,NE6,XBI,YBI,ZBI)
   14 IF(IDB.NE.0) CALL AQXVWR(IQ,NTB,NE5,XBR,YBR,ZBR)
C*                  CHORDWISE LOOP
      TUV=0.
      TVV=0.
      TWV=0.
      DO 250 I=1,NVC
      IR=4*I-3
      IF(IDF.EQ.0) IR=IR+1200
      DO 15 IW=1,NW
      SEU(IW)=0.0
      SEV(IW)=0.0
   15 SEW(IW)=0.0
C*                  SPANWISE LOOP
      XV3=XLT(1)+XTLT(1)*CFI(I)*COSTT(1)
      XV2=XLV(1)+XTLV(1)*CFI(I)*COSTV(1)
      XVT=XLV(NVSH)+XTLV(NVSH)*CFI(I)*COSTV(NVSH)
      DO 200 IE=1,NE1
      IF(IE.NE.1) GO TO 22
      TL=2./DLO
      TPT=(XVT-XV3)*TL/HWS
      GO TO 24
   22 IF(IE.NE.(NE3+4)) GO TO 26
```

```
      TL=2./DLI
   24 DL4=2.*TL/BETA
      TDY=HWS/TL
      SC=TDY
      DTX=DTH/SC
   26 KT=IE
      WZV=WOZ-XTLV(IE)*CFI(I)*SINTV(IE)+DZLE(IE)
      ZVW=(ZQ(IQ)-WZV*BETA)/SC
      IF(INDT.NE.0) GO TO 40
      IF(IE.GT.NE5) GO TO 200
      IF(IE.LE.NE3) GO TO 48
      IF(IE.GE.NE4) GO TO 48
      GO TO 200
   40 IF(IE.NE.NE1) GO TO 42
      TPI=TPT
      TPO=-TPT
      TL=2./DLO
      DL4=2.*TL/BETA
      TDY=HWS/TL
      SC=TDY
      DTX=DTH/SC
      YV=HWS
      XVW=(XQ(IQ)-XVT)/SC
      GO TO 60
   42 IF(IE.GT.NE5) GO TO 43
      IF(IE.LE.NE3) GO TO 48
      IF(IE.GE.NE4) GO TO 48
   43 TPO=(XB(IR)-XB(IR+1))/(YB(IR)-YB(IR+1))
      TPI=TPO
   46 YV=.5*(YB(IR)+YB(IR+1))
      XVW=(XQ(IQ)-(XB(IR)+XB(IR+1))/2.)/SC
      ZVW=(ZQ(IQ)-(ZB(IR)+ZB(IR+1))/2.)/SC
      IR=IR+1
      IF(IE.EQ.NE6) IR=IR+597
      IF(IE.EQ.(NE3+6)) IR=IR+597
      GO TO 60
   48 YV=HWS*EV(IE)
      IF(IE.NE.1) GO TO 50
      TPI=TPT
      TPO=TPT
      XVW=4.*(XQ(IQ)-XV2)/SC
      ZVW=4.*ZVW
      GO TO 60
```

```
50 IF(IE-NE4) 58,52,54
52 XV1=XLT(IE)+XTLT(IE)*CFI(I)*COSTT(IE)
54 KT=KT+1
58 XV2=XLV(IE)+XTLV(IE)*CFI(I)*COSTV(IE)
   XV3=XLT(KT)+XTLT(KT)*CFI(I)*COSTT(KT)
   XVW=(XQ(IQ)-XV2)/SC
   TPI=(XV2-XV3)/TDY
   TPO=(XV1-XV2)/TDY
   IF(IE.EQ.NE2) GO TO 60
   IF(ET(KT).LT.0.0) TPI=TPO
60 ZV2=ZVW**2
   XZ2=XVW**2+ZV2
   ZL4=ZVW*DL4
   IPASS=1
70 YVW=(YQ(IQ)-YV)/SC
   EVI=0.
   EVO=0.
   IF((YVW**2+ZVW**2).GT.XLW2) GO TO 120
   TI2=TPI**2+1.
   TO2=TPO**2+1.
   XTI=XVW+TPI
   XTO=XVW-TPO
   IF(IE.EQ.1) YVW=4.*YVW
   YCP=YVW+1.
   YP2=YCP**2
   YCM=YVW-1.
   YM2=YCM**2
   SQO=SQRT(XTO**2+YM2+ZV2)
   SQI=SQRT(XTI**2+YP2+ZV2)
   SQR=SQRT(XZ2+YVW**2)
   XYTI=XVW-YVW*TPI
   XYTO=XVW-YVW*TPO
   R12=(XTI/SQI+1.)/(YP2+ZV2)
   R22=XYTI**2+ZV2*TI2
   R32=XYTO**2+ZV2*TO2
   R42=(XTO/SQO+1.)/(YM2+ZV2)
   EX=0.0
   EY=R12-R42
   EZ=YCM*R42-YCP*R12
   IF(R22/TI2.LE.1.E-10) GO TO 80
   YXTI=YVW+XVW*TPI
   EX=((YXTI+TI2)/SQI-YXTI/SQR)/R22
   EY=EY-EX*TPI
```

```
      EVI=EX*XYTI
      EZ=EZ-EVI
   80 IF(R32/TO2.LE.1.E-10) GO TO 90
      YXTO=YVW+XVW*TPO
      EXX=(YXTO/SQR+(TO2-YXTO)/SQO)/R32
      EX=EX+EXX
      EY=EY-EXX*TPO
      EVO=-EXX*XYTO
      EZ=EZ+EVO
   90 IF(INDT.EQ.0) GO TO 99
      IF(IE.EQ.1) GO TO 98
      STS=1./SQR
      EUI=1./SQI-STS
      EUO=1./SQO-STS
      EWI=0.
      IF(R22/TI2.LE.1.E-10) GO TO 92
      PII=SQRT(TI2)
      EWI=(YXTI/SQR-(YXTI+TI2)/SQI)*ZVW*PII/R22
   92 EWO=0.
      IF(R32/TO2.LE.1.E-10) GO TO 94
      POI=SQRT(TO2)
      EWO=(YXTO/SQR-(YXTO-TO2)/SQO)*ZVW*POI/R32
   94 ST=TSI(IE,I)*STH(I)*DTX
      IF(IE.NE.NE1) GO TO 96
      IF(IPASS.EQ.0.0) GO TO 97
   96 C4=ST/TI2
      TUV=TUV-(EUI*TPI-EVI)*C4
      TVV=TVV-(EVI*TPI+EUI)*C4
      TWV=TWV-EWI*PII*C4
      IF(IE.EQ.NE1) GO TO 130
   97 C1=ST/TO2
      TUV=TUV+(EUO*TPO-EVO)*C1
      TVV=TVV+(EVO*TPO+EUO)*C1
      TWV=TWV+EWO*POI*C1
      IF(IE.EQ.NE1) GO TO 200
   98 IF(IE.GT.NE5) GO TO 120
      IF(IE.GE.NE4) GO TO 99
      IF(IE.GT.NE3) GO TO 120
   99 EX=EX*ZL4
      EY=EY*ZL4
      EZ=EZ*DL4
      IF(WNW.NE.0.0) GO TO 100
      SEU(IE)=EX+SEU(IE)
```

```
      SEV(IE)=EY+SEV(IE)
      SEW(IE)=EZ+SEW(IE)
      GO TO 120
  100 DO 110 IW=1,NW
      SEU(IW)=SEU(IW)+EX*S(IE,IW)
      SEV(IW)=SEV(IW)+EY*S(IE,IW)
  110 SEW(IW)=SEW(IW)+EZ*S(IE,IW)
  120 IF(IPASS.EQ.0) GO TO 190
      IF(IDB.NE.0) GO TO 130
      IF(IE.EQ.NE2) GO TO 200
  130 IPASS=0
      TPX=-TPI
      TPI=-TPO
      TPO=TPX
      YV=-YV
      GO TO 70
  190 XV1=XV3
  200 CONTINUE
      IF(WNW.NE.0.0) GO TO 204
      SEU(1)=4.*SEU(1)
      SEV(1)=4.*SEV(1)
      SEW(1)=4.*SEW(1)
  204 MY=0
      DO 210 IU=1,NU
      DO 210 IW=1,NW
      MY=MY+1
      AX(MY)=AX(MY)+SEU(IW)*C(I,IU)
      AY(MY)=AY(MY)+SEV(IW)*C(I,IU)
  210 AZ(MY)=AZ(MY)+SEW(IW)*C(I,IU)
  250 CONTINUE
      TUV=TUV/PI4/BETA
      TVV=TVV/PI4
      TWV=TWV/PI4
C
      IF(ITYP.LT.18) GO TO 290
      TVT=TVV
      TVV=TVT*SINDP+TWV*COSDP
      TWV=-TVT*COSDP+TWV*SINDP
      DO 260 I=1,NKW
      AYT=AY(I)
      AY(I)=AYT*SINDP+AZ(I)*COSDP
  260 AZ(I)=-AYT*COSDP+AZ(I)*SINDP
C
```

```
  290 WRITE (21) (AX(IU),AY(IU),AZ(IU),IU=1,NKW),TUV,TVV,TWV
  300 CONTINUE
      WRITE (6,450) ITYP
  450 FORMAT (21H0 ** COMPLETED MATRIX,I4)
      RETURN
      END
*DECK AQXVWR
      SUBROUTINE AQXVWR(IQ,NVT,NEVX,XB,YB,ZB)
C
C*                CALC MATRIX-A (Q ON X-COMPONENT AND VORTICES ON ROOT)
C
      COMMON/DAT/ DA(5000)
C
      LARGE      XB(1),YB(1),ZB(1)
C
      EQUIVALENCE       (DA(1281),WNW)   ,(DA(3382),XLW)
C
      COMMON/CPW/ IDD(9),NU,NW,NVC
C
      LARGE       DM1(51700),DM2(14552)
     1            ,S(51,10)   ,CSEC(40,10),DM3(17729)
     2            ,XQ(650)    ,YQ(650)    ,ZQ(650)
     3            ,AX(250)    ,AY(250)    ,AZ(250)
     4            ,QBX(600)   ,QBY(600)   ,QBZ(600)
     5            ,QTX(600)   ,QTY(600)   ,QTZ(600)
C
C                 INITIAL
      NVCT=NVC+NVT-1
      NPAS=0
      IF(ABS((-YB(4)-YQ(IQ))/(YB(1)-YB(2))).GT.XLW) NPAS=1
C                 NVX AND NVY LOOPS
      LI=1
      LF=5
      JS=0
      DO 300 IX=1,NVCT
      K=0
      IF(IX.GT.NVC) JS=JS-4
      DO 200 IS=1,4
      JS=JS+1
   20 IPASS=1
      IF(IX.GT.NVC) IPASS=3
   30 TY1=YB(LF)-YB(LI)
      TY2=YB(LF)-YQ(IQ)
```

```
      TY3=YB(LI)-YQ(IQ)
      GO TO 50
   40 IPASS=2
      IF(NPAS.EQ.0) GO TO 44
      IF(K.EQ.0) GO TO 160
      GO TO 70
   44 TY1=YB(LI)-YB(LF)
      TY2=-YB(LF)-YQ(IQ)
      TY3=-YB(LI)-YQ(IQ)
   50 TX1=XB(LF)-XB(LI)
      TX2=XB(LF)-XQ(IQ)
      TX3=XB(LI)-XQ(IQ)
      TZ1=ZB(LF)-ZB(LI)
      TZ2=ZB(LF)-ZQ(IQ)
      TZ3=ZB(LI)-ZQ(IQ)
      HN=TX2*TX1+TY2*TY1+TZ2*TZ1
      HD=TX1**2+TY1**2+TZ1**2
      SQHD=SQRT(HD)
      IF(SQHD.GT.0.00001) GO TO 52
      QSR=0.0
      GO TO 54
   52 HT=HN/HD
      HX=TX2-HT*TX1
      HY=TY2-HT*TY1
      HZ=TZ2-HT*TZ1
      SHP=SQRT(HX**2+HY**2+HZ**2)
      SRF=HN/SQRT(TX2**2+TY2**2+TZ2**2)
      HI=TX1*TX3+TY1*TY3+TZ1*TZ3
      SRI=HI/SQRT(TX3**2+TY3**2+TZ3**2)
      QT=(SRF-SRI)/SQHD/SHP
      TX=TY2*TZ1-TZ2*TY1
      TY=TZ2*TX1-TX2*TZ1
      TZ=TX2*TY1-TY2*TX1
      QSR=QT/SQRT(TX**2+TY**2+TZ**2)
   54 IF(K.EQ.0) GO TO 140
C*                 LATERAL VORTICES
      IF(IPASS.EQ.2) GO TO 60
      QBX(JS)=QSR*TX
      QBY(JS)=QSR*TY
      QBZ(JS)=QSR*TZ
      LF=LI
      LI=LF+1
      GO TO 40
```

```
   60 QBX(JS)=QSR*TX+QBX(JS)
      QBY(JS)=QSR*TY+QBY(JS)
      QBZ(JS)=QSR*TZ+QBZ(JS)
   70 LF=LI+4
      K=0
      GO TO 20
C*                 LONGITUDINAL  VORTICES
  140 IF(IPASS.GE.2) GO TO 150
      QTX(JS)=QSR*TX
      QTY(JS)=QSR*TY
      QTZ(JS)=QSR*TZ
  145 LF=LI
      LI=LF+4
      GO TO 40
  150 QTX(JS)=QSR*TX+QTX(JS)
      QTY(JS)=QSR*TY+QTY(JS)
      QTZ(JS)=QSR*TZ+QTZ(JS)
      IF(IPASS.EQ.3) GO TO 145
  160 IF(IX.LE.NVC) GO TO 190
      LI=LF+1
      LF=LI+4
      GO TO 200
  190 LI=LF
      LF=LI+1
      K=1
  200 CONTINUE
      IF(IX.GT.NVC) GO TO 300
      LI=LI+1
      LF=LI+4
  300 CONTINUE
C
C*                 CALC CONTRIBUTION FROM TRAIL VORTICES
      JS=JS-4
      DO 320 IS=1,4
      JS=JS+1
      PASS=1.
      HY=YB(LI)-YQ(IQ)
      HX=XB(LI)-XQ(IQ)
      HZ=ZB(LI)-ZQ(IQ)
  310 DD1=HY*HY+HZ*HZ
      IF(DD1.LT.0.00001) GO TO 315
      DD2=SQRT(HX*HX+DD1)
      IF(DD2.LT.0.00001) GO TO 315
```

```
      QSR=(1.-HX/DD2)/DD1*PASS
      QTY(JS)=QTY(JS)+QSR*HZ
      QTZ(JS)=QTZ(JS)-QSR*HY
  315 IF(NPAS.EQ.1) GO TO 320
      IF(PASS.LT.0.) GO TO 320
      PASS=-1.
      HY=-YB(LI)-YQ(IQ)
      GO TO 310
  320 LI=LI+1
C
C*                 SUM FOR AX,AY AND AZ
      N=4*NVC+1
      DO 340 I=1,3
      N=N+1
      QBX(N)=0.0
      QBY(N)=0.0
  340 QBZ(N)=0.0
      N=0
      DO 360 J=1,NVC
      N=N+1
      DO 360 I=1,3
      N=N+1
      EX=QTX(N-1)-QTX(N)+QBX(N+4)-QBX(N)
      EY=QTY(N-1)-QTY(N)+QBY(N+4)-QBY(N)
      EZ=QTZ(N-1)-QTZ(N)+QBZ(N+4)-QBZ(N)
      IE=NEVX+I
      IF(WNW.NE.0.0) GO TO 348
      K=IE
      DO 342 IU=1,NU
      AX(K)=EX*CSEC(J,IU)+AX(K)
      AY(K)=EY*CSEC(J,IU)+AY(K)
      AZ(K)=EZ*CSEC(J,IU)+AZ(K)
  342 K=K+NW
      GO TO 360
  348 K=0
      DO 350 IU=1,NU
      DO 350 IW=1,NW
      K=K+1
      SCS=S(IE,IW)*CSEC(   J,IU)
      AX(K)=EX*SCS+AX(K)
      AY(K)=EY*SCS+AY(K)
  350 AZ(K)=EZ*SCS+AZ(K)
  360 CONTINUE
```

```
      RETURN
      END
*DECK AQWVW
      SUBROUTINE AQWVW (ITYP)
C
C*                CALC MATRIX-A (Q ON WING AND VORTICES ON WING)
C
      COMMON/DAT/ DA(5000)
C
      EQUIVALENCE      (DA(1270),WNVC) ,(DA(1281),WNW) ,(DA(3382),XLW)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CFG/ IDB,IDF,IDD(10),NKW
      COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3
     1           ,NE5,NTB,NU,NW,NVC,NVS,NJC,NJS
      LARGE      DM1(51700)
     1           ,EV(52)     ,XLV(52)    ,XTLV(52)   ,DM2(3440)
     2           ,UV(30,30)  ,UVP(30,30) ,VV(30,30)  ,VVP(30,30)
     3           ,XB(600)    ,XBI(600)   ,XBR(600)
     4           ,YB(600)    ,YBI(600)   ,YBR(600)
     5           ,ZB(600)    ,ZBI(600)   ,ZBR(600)
     6           ,CFI(50)    ,CFJ(50)    ,STH(50)    ,ACT(50)
     7           ,ET(52)     ,XLT(52)    ,XTLT(52)
     8           ,C(40,40)   ,S(51,10)  ,CSEC(40,10)
     9           ,TSI(52,40),TZI(52,40),DM3(13569)
     *           ,XQ(650)    ,YQ(650)    ,ZQ(650)
     *           ,AX(250)    ,AY(250)    ,AZ(250)    ,SUMES(30)
C
C
      NE1=NE2+1
      NVSH=NE1
      NE4=NE3+7
      NE6=NE3+3
      INDT=DA(2000)
      IF(INDT.EQ.0) NE1=NE2
      DTH=PI/WNVC*.5
C*                Q ON WING LOOP
      IC=0
      DO 350 KP=1,NJS
      DO 340 J=1,NJC
      PUV=0.
      PVV=0.
      TUV=0.
```

```
      TVV=0.
      DO 12 I=1,NKW
      AX(I)=0.0
      AY(I)=0.0
   12 AZ(I)=0.0
      IC=IC+1
      XP=XQ(IC)
      EVP=YQ(IC)/HWS
      IF(IDF.EQ.0.AND.IDB.EQ.0) GO TO 40
      IF(IDF.EQ.0) GO TO 14
      CALL AQXVWR(IC,NTO,NE3,XB,YB,ZB)
      CALL AQXVWR(IC,NTI,NE6,XBI,YBI,ZBI)
   14 IF(IDB.NE.0) CALL AQXVWR(IC,NTB,NE5,XBR,YBR,ZBR)
      DO 32 I=1,NKW
   32 AZ(I)=-AZ(I)
C        I LOOP
   40 DO 330 I=1,NVC
      IR=4*I-3
      IF(IDF.EQ.0) IR=IR+1200
      DO 100 NS=1,NW
  100 SUMES(NS)=0.0
C        ETAV LOOP
      XV3=XLT(1)+XTLT(1)*CFI(I)
      XV2=XLV(1)+XTLV(1)*CFI(I)
      XVT=XLV(NVSH)+XTLV(NVSH)*CFI(I)
      DO 310 KV=1,NE1
      IF(KV.NE.1) GO TO 102
      TL=2./DLO
      TPT=(XVT-XV3)*TL/HWS
      GO TO 104
  102 IF(KV.NE.(NE3+4)) GO TO 106
      TL=2./DLI
  104 DL4=2.*TL/BETA
      X2=TL/HWS
      DTX=X2*DTH
      TDY=HWS/TL
  106 KT=KV
      IPASS=1
      EVY=EV(KV)
      IF(INDT.NE.0) GO TO 110
      IF(KV.GT.NE5) GO TO 310
      IF(KV.LE.NE3) GO TO 140
      IF(KV.GE.NE4) GO TO 140
```

```
      GO TO 310
  110 IF(KV.NE.NE1) GO TO 120
      TPI=TPT
      TPO=-TPT
      TL=2./DLO
      DL4=2.*TL/BETA
      X2=TL/HWS
      DTX=X2*DTH
      TDY=HWS/TL
      XC=X2*(XP-XVT)
      GO TO 190
  120 IF(KV.GT.NE5) GO TO 121
      IF(KV.LE.NE3) GO TO 140
      IF(KV.GE.NE4) GO TO 140
  121 TPO=(XB(IR)-XB(IR+1))/(YB(IR)-YB(IR+1))
  130 EVY=(YB(IR)+YB(IR+1))/2./HWS
      XC=X2*(XP-(XB(IR)+XB(IR+1))/2.)
      TPI=TPO
      IR=IR+1
      IF(KV.EQ.NE6) IR=IR+597
      IF(KV.EQ.(NE3+6)) IR=IR+597
      GO TO 190
  140 IF(KV.NE.1) GO TO 150
      TPI=TPT
      TPO=TPT
      XC=4.*X2*(XP-XV2)
      GO TO 190
  150 IF(KV-NE4) 180,160,170
  160 XV1=XLT(KV)+XTLT(KV)*CFI(I)
  170 KT=KT+1
  180 XV2=XLV(KV)+XTLV(KV)*CFI(I)
      XV3=XLT(KT)+XTLT(KT)*CFI(I)
      TPI=(XV2-XV3)/TDY
      TPO=(XV1-XV2)/TDY
      IF(KV.EQ.NE2) GO TO 182
      IF(ET(KT).LT.0.0) TPI=TPO
  182 XC=X2*(XP-XV2)
  190 YC=TL*(EVP-EVY)
      EVI=0.
      EVO=0.
      IF(ABS(YC/2.).GT.XLW) GO TO 302
      IF(KV.EQ.1) YC=4.*YC
      YCP=YC+1.
```

```
      YCM=YC-1.
      XTO=XC-TPO
      XTI=XC+TPI
      SQO=SQRT(XTO**2+YCM**2)
      SQI=SQRT(XTI**2+YCP**2)
      EC=(XTI/SQI+1.)/YCP-(XTO/SQO+1.)/YCM
      SQC=SQRT(XC**2+YC**2)
      C1=TPI**2+1.
      C4=C1
      C2=XC-YC*TPI
      IF(C2**2/C1.LE.1.E-10) GO TO 230
      C3=YC+XC*TPI
      EVI=((C3+C1)/SQI-C3/SQC)/C2
  230 C1=TPO**2+1.
      C2=XC-YC*TPO
      IF(C2**2/C1.LE.1.E-10) GO TO 250
      C3=YC+XC*TPO
  240 EVO=((C3-C1)/SQO-C3/SQC)/C2
  250 IF(INDT.EQ.0) GO TO 290
      IF(KV.EQ.1) GO TO 290
      STS=1./SQC
      EUI=1./SQI-STS
      EUO=1./SQO-STS
      AYC=ABS(YC/X2)
      ST=TSI(KV,I)*STH(I)*DTX
      SP=TZI(KV,I)/STH(I)*DTX*2.0
      TVI=ST*EVI
      TVO=ST*EVO
      KX=KV*IPASS
  260 IF(KV.NE.NE1) GO TO 280
      IF(KX.NE.NE1) GO TO 270
  262 PUV=PUV-SP*(EUI*TPI-EVI)/C4
      PVV=PVV-SP*(EVI*TPI+EUI)/C4
      EUI=ST*EUI
      TUV=TUV-(EUI*TPI-TVI)/C4
      TVV=TVV-(TVI*TPI+EUI)/C4
      GO TO 305
  270 PUV=PUV+SP*(EUO*TPO-EVO)/C1
      PVV=PVV+SP*(EVO*TPO+EUO)/C1
      EUO=ST*EUO
      TUV=TUV+(EUO*TPO-TVO)/C1
      TVV=TVV+(TVO*TPO+EUO)/C1
      GO TO 310
```

```
  280 PUV=PUV+SP*((EUO*TPO-EVO)/C1-(EUI*TPI-EVI)/C4)
      PVV=PVV+SP*((EVO*TPO+EUO)/C1-(EVI*TPI+EUI)/C4)
      EUI=ST*EUI
      EUO=ST*EUO
      TUV=TUV+(EUO*TPO-TVO)/C1-(EUI*TPI-TVI)/C4
      TVV=TVV+(TVO*TPO+EUO)/C1-(TVI*TPI+EUI)/C4
  290 IF(KV.GT.NE5) GO TO 302
      IF(KV.GE.NE4) GO TO 291
      IF(KV.GT.NE3) GO TO 302
  291 EC=(EC+EVI-EVO)*DL4
      IF(WNW.NE.0.0) GO TO 292
      SUMES(KV)=SUMES(KV)+EC
      GO TO 302
  292 DO 300 NS=1,NW
  300 SUMES(NS)=SUMES(NS)+EC*S(KV,NS)
  302 IF(IPASS.EQ.0) GO TO 308
      IF(IDB.NE.0) GO TO 305
      IF(KV.EQ.NE2) GO TO 310
  305 IPASS=0
      EVY=-EVY
      TPX=-TPI
      TPI=-TPO
      TPO=TPX
      GO TO 190
  308 XV1=XV3
  310 CONTINUE
      IF(WNW.EQ.0.0) SUMES(1)=4.*SUMES(1)
      MY=0
  312 DO 314 NC=1,NU
      DO 314 NS=1,NW
      MY=MY+1
  314 AZ(MY)=AZ(MY)+SUMES(NS)*C(I,NC)
  330 CONTINUE
      WRITE (21) (AZ(MY),MY=1,NKW)
      UVP(KP,J)=PUV/PI4/BETA
      VVP(KP,J)=PVV/PI4
      UV(KP,J)=TUV/PI4/BETA
  340 VV(KP,J)=TVV/PI4
  350 CONTINUE
      IF(INDT.NE.0) CALL AQWVWT
C
      WRITE (6,450) ITYP
  450 FORMAT (21H0 ** COMPLETED MATRIX,I4)
```

```
      RETURN
      END
*DECK AQWVWT
      SUBROUTINE AQWVWT
C
C*               CALC ADDITIONAL INFLUENCE OF WING THICKNESS ON
C                VELOCITIES DUE TO BREAKS,ETC.
C
      COMMON/DAT/ DA(5000)
C
      EQUIVALENCE      (DA(1270),WNVC) ,(DA(3382),XLW)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CFG/ IDB,IDF
      COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3
     1           ,NE5,NTB,NU,NW,NVC,NVS,NJC,NJS
C
      LARGE       DM1(51700)
     1           ,EV(52)     ,XLV(52)    ,XTLV(52)   ,DM2(3440)
     2           ,UV(30,30)  ,UVP(30,30) ,VV(30,30)  ,VVP(30,30)
     3           ,XB(1800)   ,YB(1800)   ,ZB(1800)
     4           ,CFI(50)    ,CFJ(50)    ,STH(50)    ,ACT(50)
     5           ,ET(52)     ,XLT(52)    ,XTLT(52)
     6           ,DM3(2510)  ,TSI(52,40) ,TZI(52,40)
     7           ,DM4(13569) ,XQ(650)    ,YQ(650)    ,ZQ(650)
C
C
      NE1=NE2+1
      NVSH=NE1
      NE4=NE3+7
      NE6=NE3+3
      DTH=PI/WNVC*.5
C*               Q ON WING LOOP
      IC=0
      DO 350 KP=1,NJS
      DO 340 J=1,NJC
      PUV=0.
      PVV=0.
      TUV=0.
      TVV=0.
      IC=IC+1
      XP=XQ(IC)
      EVP=YQ(IC)/HWS
```

```
C         I LOOP
   40 DO 330 I=2,NVC
      STHJ=SIN(ACT(I))
      IR=4*I-7
      IF(IDF.EQ.0) IR=IR+1200
C         ETAV LOOP
      XV3=XLT(1)+XTLT(1)*CFJ(I)
      XV2=XLV(1)+XTLV(1)*CFJ(I)
      XVT=XLV(NVSH)+XTLV(NVSH)*CFJ(I)
      DO 310 KV=1,NE1
      IF(KV.NE.1) GO TO 102
      TL=2./DLO
      TPT=(XVT-XV3)*TL/HWS
      GO TO 104
  102 IF(KV.NE.(NE3+4)) GO TO 106
      TL=2./DLI
  104 DL4=2.*TL/BETA
      X2=TL/HWS
      DTX=X2*DTH
      TDY=HWS/TL
  106 KT=KV
      IPASS=1
      EVY=EV(KV)
  110 IF(KV.NE.NE1) GO TO 120
      TPI=TPT
      TPO=-TPT
      TL=2./DLO
      DL4=2.*TL/BETA
      X2=TL/HWS
      DTX=X2*DTH
      TDY=HWS/TL
      XC=X2*(XP-XVT)
      GO TO 190
  120 IF(KV.GT.NE5) GO TO 130
      IF(KV.LE.NE3) GO TO 140
      IF(KV.GE.NE4) GO TO 140
  130 XB1=(XB(IR)+XB(IR+4))/2.0
      XB2=(XB(IR+1)+XB(IR+5))/2.0
      YB1=(YB(IR)+YB(IR+4))/2.0
      YB2=(YB(IR+1)+YB(IR+5))/2.0
      TPO=(XB1-XB2)/(YB1-YB2)
      EVY=(YB1+YB2)/2.0/HWS
      XC=X2*(XP-(XB1+XB2)/2.0)
```

```
      TPI=TPO
      IR=IR+1
      IF(KV.EQ.NE6) IR=IR+597
      IF(KV.EQ.(NE3+6)) IR=IR+597
      GO TO 190
  140 IF(KV.NE.1) GO TO 150
      TPI=TPT
      TPO=TPT
      XC=4.*X2*(XP-XV2)
      GO TO 190
  150 IF(KV-NE4) 180,160,170
  160 XV1=XLT(KV)+XTLT(KV)*CFJ(I)
  170 KT=KT+1
  180 XV2=XLV(KV)+XTLV(KV)*CFJ(I)
      XV3=XLT(KT)+XTLT(KT)*CFJ(I)
      TPI=(XV2-XV3)/TDY
      TPO=(XV1-XV2)/TDY
      IF(KV.EQ.NE2) GO TO 182
      IF(ET(KT).LT.0.0) TPI=TPO
  182 XC=X2*(XP-XV2)
  190 YC=TL*(EVP-EVY)
      EVI=0.
      EVO=0.
      IF(ABS(YC/2.).GT.XLW) GO TO 290
      IF(KV.EQ.1) YC=4.*YC
      YCP=YC+1.
      YCM=YC-1.
      XTO=XC-TPO
      XTI=XC+TPI
      SQO=SQRT(XTO**2+YCM**2)
      SQI=SQRT(XTI**2+YCP**2)
      SQC=SQRT(XC**2+YC**2)
      C1=TPI**2+1.
      C4=C1
      C2=XC-YC*TPI
      IF(C2**2/C1.LE.1.E-10) GO TO 230
      C3=YC+XC*TPI
      EVI=((C3+C1)/SQI-C3/SQC)/C2
  230 C1=TPO**2+1.
      C2=XC-YC*TPO
      IF(C2**2/C1.LE.1.E-10) GO TO 250
      C3=YC+XC*TPO
  240 EVO=((C3-C1)/SQO-C3/SQC)/C2
```

```
250 IF(KV.EQ.1) GO TO 290
    IF(SQC.GT.1.E-10) GO TO 254
    STS=0.0
    GO TO 256
254 STS=1./SQC
256 EUI=1./SQI-STS
    EUO=1./SQO-STS
    AYC=ABS(YC/X2)
    SLOPE=(TSI(KV,I)+TSI(KV,I-1))/2.0
    DEFL=(TZI(KV,I)+TZI(KV,I-1))/2.0
    ST=SLOPE*STHJ*DTX
    SP=DEFL/STHJ*DTX*2.0
    TVI=ST*EVI
    TVO=ST*EVO
    KX=KV*IPASS
260 IF(KV.NE.NE1) GO TO 280
    IF(KX.NE.NE1) GO TO 270
262 PUV=PUV-SP*(EUI*TPI-EVI)/C4
    PVV=PVV-SP*(EVI*TPI+EUI)/C4
    EUI=ST*EUI
    TUV=TUV-(EUI*TPI-TVI)/C4
    TVV=TVV-(TVI*TPI+EUI)/C4
    GO TO 305
270 PUV=PUV+SP*(EUO*TPO-EVO)/C1
    PVV=PVV+SP*(EVO*TPO+EUO)/C1
    EUO=ST*EUO
    TUV=TUV+(EUO*TPO-TVO)/C1
    TVV=TVV+(TVO*TPO+EUO)/C1
    GO TO 310
280 PUV=PUV+SP*((EUO*TPO-EVO)/C1-(EUI*TPI-EVI)/C4)
    PVV=PVV+SP*((EVO*TPO+EUO)/C1-(EVI*TPI+EUI)/C4)
    EUI=ST*EUI
    EUO=ST*EUO
    TUV=TUV+(EUO*TPO-TVO)/C1-(EUI*TPI-TVI)/C4
    TVV=TVV+(TVO*TPO+EUO)/C1-(TVI*TPI+EUI)/C4
290 IF(IPASS.EQ.0) GO TO 308
    IF(IDB.NE.0) GO TO 305
    IF(KV.EQ.NE2) GO TO 310
305 IPASS=0
    EVY=-EVY
    TPX=-TPI
    TPI=-TPO
    TPO=TPX
```

```
      GO TO 190
  308 XV1=XV3
  310 CONTINUE
  330 CONTINUE
      UVP(KP,J)=PUV/PI4/BETA+UVP(KP,J)
      VVP(KP,J)=PVV/PI4+VVP(KP,J)
      UV(KP,J)=TUV/PI4/BETA+UV(KP,J)
  340 VV(KP,J)=TVV/PI4+VV(KP,J)
  350 CONTINUE
      RETURN
      END
*DECK AQXVP
      SUBROUTINE AQXVP(NQ,ITYP)
C
C*              CALC MATRIX-A * Q ON COMPONENTS AND VORTICES ON PYLON
C
      COMMON/DAT/ DA(5000)
C
      EQUIVALENCE      (DA(2500),PHT)   ,(DA(2507),PNVC),(DA(3383),XLP)
     1,(DA(2501),POX) ,(DA(2502),POY)   ,(DA(2503),POZ) ,(DA(2492),PDA)
C
      COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP
      COMMON/CFG/ IDG(13),NKP
      COMMON/CPP/ HPS,NVSH,NVT,DL,NEVP,ZCON,YCON,NVC,NVS,NJC,NUP
C
      LARGE       DM1(75130)
     1            ,EV(22)     ,XLV(22)    ,XTLV(22)   ,WPE(39)
     2            ,XD(60)     ,YD(60)     ,ZD(60)
     3            ,XB(400)    ,YB(400)    ,ZB(400)
     4            ,CFI(20)    ,CFJ(20)    ,STH(20)    ,ACT(20)
     5            ,ET(22)     ,XLT(22)    ,XTLT(22)   ,C(20,20)
     6            ,DM2(3190)  ,TSI(21,20),TZI(21,20),CEL(3700)
     7            ,XQ(650)    ,YQ(650)    ,ZQ(650)
     8            ,AX(250)    ,AY(250)    ,AZ(250)    ,AZU(250)
     9            ,SEU(30)    ,SEV(30)    ,SEW(30)
C
C
C*              INITIAL AND CALC MATRIX A
      PYB=POY*BETA
      PZB=POZ*BETA
      IMAGE=1
      IF(POY.EQ.0.0.AND.PDA.EQ.0.0) IMAGE=0
      XLP2=4.*XLP*XLP
```

```
      DL4=4./DL/BETA
      TDY=DL*HPS/2.
      SC=TDY
      NE1=NEVP
      NE2=NVSH-1
      INDT=DA(3000)
      IF(INDT.NE.0) NE1=NVSH
      DTH=PI/PNVC
      DTX=DTH/SC
      DO 300 IQ=1,NQ
C*                 CHORDWISE LOOP
      TVT=0.0
      TWT=0.0
      TUV=0.
      YQS=YQ(IQ)
      SDS=SINDP
      IPASS=1
      XQ(IQ)=XQ(IQ)-POX
      YQT=YQS-PYB
      ZQT=ZQ(IQ)-PZB
   20 YQ(IQ)=YQT*SDS+ZQT*COSDP
      ZQ(IQ)=ZQT*SDS-YQT*COSDP
      CALL AQXVPR (IQ,IPASS,SDS)
      TVV=0.
      TWV=0.
      IF(IPASS.EQ.0) GO TO 30
      IF(ITYP.EQ.19) GO TO 251
   30 IR=1
      DO 250 I=1,NVC
      DO 40 IW=1,NVS
      SEU(IW)=0.0
      SEV(IW)=0.0
   40 SEW(IW)=0.0
C*                 SPANWISE LOOP
      XV3=XLT(1)+XTLT(1)*CFI(I)
      XV2=XLV(1)+XTLV(1)*CFI(I)
      XVT=XLV(NVSH)+XTLV(NVSH)*CFI(I)
      TPT=(XVT-XV3)/TDY
      DO 200 IE=1,NE1
      ZVW=(ZQ(IQ)-ZCON)/SC
      IF(INDT.EQ.0.0) GO TO 48
      EVI=0.
      EVO=0.
```

```
      IF(IE.NE.NE1) GO TO 42
      TPI=TPT
      TPO=-TPT
      YV=HPS
      XVW=(XQ(IQ)-XVT)/SC
      GO TO 60
   42 IF(IE.LE.NEVP) GO TO 48
      TPO=(XB(IR)-XB(IR+1))/(YB(IR)-YB(IR+1))
   44 TPI=TPO
      YV=.5*(YB(IR)+YB(IR+1))
      XVW=(XQ(IQ)-(XB(IR)+XB(IR+1))/2.)/SC
      ZVW=(ZQ(IQ)-(ZB(IR)+ZB(IR+1))/2.)/SC
      IR=IR+1
      IF(IE.EQ.NE2) IR=IR+1
      GO TO 60
   48 YV=HPS*EV(IE)
      IF(IE.NE.1) GO TO 50
      TPI=TPT
      TPO=TPT
      XVW=4.*(XQ(IQ)-XV2)/SC
      ZVW=4.*ZVW
      GO TO 60
   50 XV2=XLV(IE)+XTLV(IE)*CFI(I)
      XV3=XLT(IE)+XTLT(IE)*CFI(I)
      XVW=(XQ(IQ)-XV2)/SC
      TPI=(XV2-XV3)/TDY
      TPO=(XV1-XV2)/TDY
      IF(IE.EQ.NE2) GO TO 60
      IF(ET(IE).LT.0.0) TPI=TPO
   60 IF(ITYP.EQ.24) ZVW=2.
      YVW=(YQ(IQ)-YV)/SC
      TI2=TPI**2+1.
      TO2=TPO**2+1.
      XTI=XVW+TPI
      XTO=XVW-TPO
      IF(IE.EQ.1) YVW=4.*YVW
      YCP=YVW+1.
      YP2=YCP**2
      YCM=YVW-1.
      YM2=YCM**2
   70 ZV2=ZVW**2
      IF((YVW**2+ZV2).GT.XLP2) GO TO 190
      XZ2=XVW**2+ZV2
```

```
      ZL4=ZVW*DL4
      SQO=SQRT(XTO**2+YM2+ZV2)
      SQI=SQRT(XTI**2+YP2+ZV2)
      SQR=SQRT(XZ2+YVW**2)
      XYTI=XVW-YVW*TPI
      XYTO=XVW-YVW*TPO
      R12=(XTI/SQI+1.)/(YP2+ZV2)
      R22=XYTI**2+ZV2*TI2
      R32=XYTO**2+ZV2*TO2
      R42=(XTO/SQO+1.)/(YM2+ZV2)
      EX=0.0
      EY=R12-R42
      EZ=YCM*R42-YCP*R12
      IF(R22/TI2.LE.1.E-10) GO TO 80
      YXTI=YVW+XVW*TPI
      EX=((YXTI+TI2)/SQI-YXTI/SQR)/R22
      EY=EY-EX*TPI
      EVI=EX*XYTI
      EZ=EZ-EVI
   80 IF(R32/TO2.LE.1.E-10) GO TO 90
      YXTO=YVW+XVW*TPO
      EXX=(YXTO/SQR+(TO2-YXTO)/SQO)/R32
      EX=EX+EXX
      EY=EY-EXX*TPO
      EVO=-EXX*XYTO
      EZ=EZ+EVO
   90 IF(INDT.EQ.0) GO TO 98
      IF(IE.EQ.1) GO TO 98
      STS=1./SQR
      EUI=1./SQI-STS
      EUO=1./SQO-STS
      EWI=0.
      IF(R22/TI2.LE.1.E-10) GO TO 92
      PII=SQRT(TI2)
      EWI=(YXTI/SQR-(YXTI+TI2)/SQI)*ZVW*PII/R22
   92 EWO=0.
      IF(R32/TO2.LE.1.E-10) GO TO 94
      POI=SQRT(TO2)
      EWO=(YXTO/SQR-(YXTO-TO2)/SQO)*ZVW*POI/R32
   94 ST=TSI(IE,I)*STH(I)*DTX
      IF(IE.NE.NE1) GO TO 96
      IF(IPASS.EQ.0.0) GO TO 97
   96 C4=ST/TI2
```

```
      TUV=TUV-(EUI*TPI-EVI)*C4
      TVV=TVV-(EVI*TPI+EUI)*C4
      TWV=TWV-EWI*PII*C4
      IF(IE.EQ.NE1) GO TO 190
   97 C1=ST/TO2
      TUV=TUV+(EUO*TPO-EVO)*C1
      TVV=TVV+(EVO*TPO+EUO)*C1
      TWV=TWV+EWO*POI*C1
      IF(IE.EQ.NE1) GO TO 200
   98 IF(IE.GT.NEVP) GO TO 190
      IF(ITYP.NE.24) GO TO 100
      IF(IPASS.EQ.1) GO TO 190
  100 EX=EX*ZL4
      EY=EY*ZL4
      EZ=EZ*DL4
      SEU(IE)=EX+SEU(IE)
      SEV(IE)=EY+SEV(IE)
      SEW(IE)=EZ+SEW(IE)
  190 XV1=XV3
  200 CONTINUE
      SEU(1)=4.*SEU(1)
      SEV(1)=4.*SEV(1)
      SEW(1)=4.*SEW(1)
      MY=0
      DO 210 IU=1,NUP
      DO 210 IW=1,NVS
      MY=MY+1
      AX(MY)=AX(MY)+SEU(IW)*C(I,IU)
      SEVT=SEV(IW)*SDS-SEW(IW)*COSDP
      IF(IPASS.EQ.0) SEVT=-SEVT
      AY(MY)=AY(MY)+SEVT    *C(I,IU)
      SEWT=SEV(IW)*COSDP+SEW(IW)*SDS
  210 AZ(MY)=AZ(MY)+SEWT    *C(I,IU)
  250 CONTINUE
C
      IF(IPASS.EQ.0) GO TO 252
  251 IPASS=0
      IF(IMAGE.EQ.0) GO TO 252
      TVT=TVV*SDS-TWV*COSDP
      TWT=TVV*COSDP+TWV*SDS
      YQT=-YQS-PYB
      GO TO 20
C
```

```
  252 TUV=TUV/PI4/BETA
      TVY=TVV
      TVVT=TVY*SDS-TWV*COSDP
      IF(IMAGE.NE.0) TVVT=-TVVT
      TVV=(TVVT+TVT)/PI4
      TWV=(TVY*COSDP+TWV*SDS+TWT)/PI4
C
      IF(ITYP.LT.19) GO TO 264
      TVY=TVV
      TVV=TVY*SINDP+TWV*COSDP
      TWV=-TVY*COSDP+TWV*SINDP
      DO 260 I=1,NKP
      AYT=AY(I)
      AY(I)=AYT*SINDP+AZ(I)*COSDP
  260 AZ(I)=-AYT*COSDP+AZ(I)*SINDP
C
  264 IF(ITYP.NE.19) GO TO 280
      READ (9) (AZU(IU),IU=1,NKP)
      DO 270 IU=1,NKP
  270 AZ(IU)=AZU(IU)-AZ(IU)
  280 WRITE (22) (AX(IU),AY(IU),AZ(IU),IU=1,NKP),TUV,TVV,TWV
C                 RESET Q COORDINATES
      XQ(IQ)=XQ(IQ)+POX
      YQ(IQ)=YQS
      ZQ(IQ)=ZQT+PZB
  300 CONTINUE
C
      REWIND 9
      WRITE (6,350) ITYP
  350 FORMAT (21H0 ** COMPLETED MATRIX,I4)
      RETURN
      END
*DECK AQXVPR
      SUBROUTINE AQXVPR (IQ,IPASS,SDS)
C
C*          CALC MATRIX-A *Q ON COMPONENTS AND VORTICES ON PYLON ROOT
C
      COMMON/DAT/ DA(5000)
C
      EQUIVALENCE      (DA(3383),XLP)
C
      COMMON/CRG/ RDG(5),COSDP
      COMMON/CFG/ IDG(13),NKP
```

```
      COMMON/CPP/ HPS,NVSH,NVT,DL,NEVP,ZCON,YCON,NVC,NVS,NJC,NUP
      LARGE        DM1(75130)
     1             ,EV(22)      ,XLV(22)     ,XTLV(22)    ,WPE(39)
     2             ,XD(60)      ,YD(60)      ,ZD(60)
     3             ,XB(400)     ,YB(400)     ,ZB(400)
     4             ,DM2(3526)   ,CSEC(21,10) ,DM3(840)    ,CEL(3700)
     5             ,XQ(650)     ,YQ(650)     ,ZQ(650)
     6             ,AX(250)     ,AY(250)     ,AZ(250)     ,A(250)
     7             ,QBX(600)    ,QBY(600)    ,QBZ(600)
     8             ,QTX(600)    ,QTY(600)    ,QTZ(600)
C
C*                 CONTROL POINT INDEX
      NVCT=NVC+NVT-1
      IF(IPASS.EQ.1) GO TO 10
      IF(ABS((YB(4)-YQ(IQ))/(YB(1)-YB(2))).GT.XLP) GO TO 400
      GO TO 14
   10 DO 12 I=1,NKP
      AX(I)=0.0
      AY(I)=0.0
   12 AZ(I)=0.0
C                  NVX AND NVY LOOPS
   14 LI=1
      LF=5
      JS=0
      DO 300 IX=1,NVCT
      K=0
      IF(IX.LE.NVC) GO TO 20
      JS=JS-4
      K=-1
   20 DO 200 IS=1,4
      JS=JS+1
   30 TZ1=ZB(LF)-ZB(LI)
      TZ2=ZB(LF)-ZQ(IQ)
      TZ3=ZB(LI)-ZQ(IQ)
   50 TX1=XB(LF)-XB(LI)
      TX2=XB(LF)-XQ(IQ)
      TX3=XB(LI)-XQ(IQ)
      TY1=YB(LF)-YB(LI)
      TY2=YB(LF)-YQ(IQ)
      TY3=YB(LI)-YQ(IQ)
      HN=TX2*TX1+TY2*TY1+TZ2*TZ1
      HD=TX1**2+TY1**2+TZ1**2
      SQHD=SQRT(HD)
```

```
      IF(SQHD.GT.0.00001) GO TO 52
      QSR=0.0
      GO TO 54
   52 HT=HN/HD
      HX=TX2-HT*TX1
      HY=TY2-HT*TY1
      HZ=TZ2-HT*TZ1
      SHP=SQRT(HX**2+HY**2+HZ**2)
      SRF=HN/SQRT(TX2**2+TY2**2+TZ2**2)
      HI=TX1*TX3+TY1*TY3+TZ1*TZ3
      SRI=HI/SQRT(TX3**2+TY3**2+TZ3**2)
      QT=(SRF-SRI)/SQHD/SHP
      TX=TY2*TZ1-TZ2*TY1
      TY=TZ2*TX1-TX2*TZ1
      TZ=TX2*TY1-TY2*TX1
      QSR=QT/SQRT(TX**2+TY**2+TZ**2)
   54 IF(K) 150,140,70
C*                 LATERAL VORTICES
   70 QBX(JS)=QSR*TX
      QBY(JS)=QSR*TY
      QBZ(JS)=QSR*TZ
      LI=LF
      LF=LI+4
      K=0
      GO TO 30
C*                 LONGITUDINAL VORTICES
  140 QTX(JS)=QSR*TX
      QTY(JS)=QSR*TY
      QTZ(JS)=QSR*TZ
      LF=LI+1
      K=1
      GO TO 200
  150 QTX(JS)=QSR*TX+QTX(JS)
      QTY(JS)=QSR*TY+QTY(JS)
      QTZ(JS)=QSR*TZ+QTZ(JS)
      LI=LI+1
      LF=LI+4
  200 CONTINUE
      IF(IX.GT.NVC) GO TO 300
      LI=LI+1
      LF=LI+4
  300 CONTINUE
C
```

```
C*                    CALC CONTRIBUTION FROM TRAIL VORTICES
      JS=JS-4
      DO 320 IS=1,4
      JS=JS+1
      HY=YB(LI)-YQ(IQ)
      HX=XB(LI)-XQ(IQ)
      HZ=ZB(LI)-ZQ(IQ)
      DD1=HY*HY+HZ*HZ
      IF(DD1.LT.0.00001) GO TO 320
      DD2=SQRT(HX*HX+DD1)
      IF(DD2.LT.0.00001) GO TO 320
      QTY(JS)=QTY(JS)+QSR*HZ    QSR=(1.-HX/DD2)/DD1    47395
      QTZ(JS)=QTZ(JS)-QSR*HY
  320 LI=LI+1
C
C*                    SUM FOR AX,AY AND AZ
      N=4*NVC+1
      DO 340 I=1,3
      N=N+1
      QBX(N)=0.0
      QBY(N)=0.0
  340 QBZ(N)=0.0
      N=0
      DO 360 J=1,NVC
      N=N+1
      DO 360 I=1,3
      N=N+1
      EX=QTX(N-1)-QTX(N)+QBX(N+4)-QBX(N)
      EY=QTY(N-1)-QTY(N)+QBY(N+4)-QBY(N)
      EZ=QTZ(N-1)-QTZ(N)+QBZ(N+4)-QBZ(N)
      EYT=EY
      EY=EYT*SDS-EZ*COSDP
      IF(IPASS.EQ.0) EY=-EY
      EZ=EYT*COSDP+EZ*SDS
      IE=NEVP+I
      K=IE
      DO 342 IU=1,NUP
      AX(K)=EX*CSEC(J,IU)+AX(K)
      AY(K)=EY*CSEC(J,IU)+AY(K)
      AZ(K)=EZ*CSEC(J,IU)+AZ(K)
  342 K=K+NVS
  360 CONTINUE
  400 RETURN
```

```
      END
*DECK AQPVP
      SUBROUTINE AQPVP
C
C*                CALC MATRIX-A Q ON PYLON AND VERTICES ON PYLON
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION   PJC(20)
C
      EQUIVALENCE      (DA(2507),PNVC) ,(DA(3383),XLP)   ,(DA(2512),PJC)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CFG/ IDG(13),NKP
      COMMON/CPP/ HPS,NVSH,NVT,DL,NEVP,ZCON,YCON,NVC,NVS,NJC,NUP
C
      LARGE       DM1(75130)
     1            ,EV(22)      ,XLV(22)    ,XTLV(22)    ,WPE(39)
     2            ,XD(60)      ,YD(60)     ,ZD(60)
     3            ,XB(400)     ,YB(400)    ,ZB(400)
     4            ,CFI(20)     ,CFJ(20)    ,STH(20)     ,ACT(20)
     5            ,ET(22)      ,XLT(22)    ,XTLT(22)    ,C(20,20)
     6            ,CSJ(20,20),TSJ(20,20),UV(20,20)
     7            ,UVP(20,20),VV(20,20)  ,VVP(20,20)
     8            ,DM2(790)    ,TSI(21,20),TZI(21,20),CEL(3700)
     9            ,XQ(650)     ,YQ(650)    ,ZQ(650)     ,AZ(250)    ,SUMES(30)
C
C
      REWIND 9
      NE1=NEVP
      NE2=NVSH-1
      INDT=DA(3000)
      IF(INDT.NE.0) NE1=NVSH
   10 DL4=4./DL/BETA
      TL=2./DL
      X2=TL/HPS
      DTH=PI/PNVC*.5
      DTX=X2*DTH
      TDY=HPS/TL
C*               Q ON PYLON LOOP
      IP=0
      IC=0
      DO 350 KP=1,NE2
```

```
      DO 340 J=1,NJC
      JX=PJC(J)
      PUV=0.
      PVV=0.
      TUV=0.
      TVV=0.
      IC=IC+1
      IF(KP.LE.NEVP) GO TO 20
      IP=IP+1
      XP=XD(IP)
      EVP=YD(IP)/HPS
      GO TO 30
   20 XP=XLV(KP)+XTLV(KP)*CFJ(JX+1)
      EVP=EV(KP)
   30 DO 32 I=1,NKP
   32 AZ(I)=0.0
C       I LOOP
   40 IR=1
      DO 330 I=1,NVC
      DO 100 NS=1,NVS
  100 SUMES(NS)=0.0
C       ETAV LOOP
      XV3=XLT(1)+XTLT(1)*CFI(I)
      XV2=XLV(1)+XTLV(1)*CFI(I)
      XVT=XLV(NVSH)+XTLV(NVSH)*CFI(I)
      TPT=(XVT-XV3)/TDY
      DO 310 KV=1,NE1
      EVI=0.
      EVO=0.
      EVY=EV(KV)
      IF(INDT.EQ.0   ) GO TO 140
      IF(KV.NE.NE1) GO TO 120
      TPI=TPT
      TPO=-TPT
      XC=X2*(XP-XVT)
      GO TO 190
  120 IF(KV.LE.NEVP) GO TO 140
      TPO=(XB(IR)-XB(IR+1))/(YB(IR)-YB(IR+1))
  130 EVY=(YB(IR)+YB(IR+1))/2./HPS
      XC=X2*(XP-(XB(IR)+XB(IR+1))/2.)
      TPI=TPO
      IR=IR+1
      IF(KV.EQ.NE2) IR=IR+1
```

```
      GO TO 190
  140 IF(KV.NE.1) GO TO 180
      TPI=TPT
      TPO=TPT
      XC=4.*X2*(XP-XV2)
      GO TO 190
  180 XV2=XLV(KV)+XTLV(KV)*CFI(I)
      XV3=XLT(KV)+XTLT(KV)*CFI(I)
      TPI=(XV2-XV3)/TDY
      TPO=(XV1-XV2)/TDY
      IF(KV.EQ.NE2) GO TO 182
      IF(ET(KV).LT.0.0) TPI=TPO
  182 XC=X2*(XP-XV2)
  190 YC=TL*(EVP-EVY)
      IF(ABS(YC/2.).GT.XLP) GO TO 300
      IF(KV.EQ.1) YC=4.*YC
      YCP=YC+1.
      YCM=YC-1.
      XTO=XC-TPO
      XTI=XC+TPI
      SQO=SQRT(XTO**2+YCM**2)
      SQI=SQRT(XTI**2+YCP**2)
      EC=(XTI/SQI+1.)/YCP-(XTO/SQO+1.)/YCM
      SQC=SQRT(XC**2+YC**2)
      C1=TPI**2+1.
      C4=C1
      C2=XC-YC*TPI
      IF(C2**2/C1.LE.1.E-10) GO TO 230
      C3=YC+XC*TPI
      EVI=((C3+C1)/SQI-C3/SQC)/C2
  230 C1=TPO**2+1.
      C2=XC-YC*TPO
      IF(C2**2/C1.LE.1.E-10) GO TO 250
      C3=YC+XC*TPO
  240 EVO=((C3-C1)/SQO-C3/SQC)/C2
  250 IF(INDT.EQ.0) GO TO 290
      IF(KV.EQ.1) GO TO 290
      STS=1./SQC
      EUI=1./SQI-STS
      EUO=1./SQO-STS
      AYC=ABS(YC/X2)
      ST=TSI(KV,I)*STH(I)*DTX
      SP=TZI(KV,I)/STH(I)*DTX*2.0
```

```
      TVI=ST*EVI
      TVO=ST*EVO
  260 IF(KV.NE.NE1) GO TO 280
      PUV=PUV-SP*(EUI*TPI-EVI)/C4
      PVV=PVV-SP*(EVI*TPI+EUI)/C4
      EUI=ST*EUI
      TUV=TUV-(EUI*TPI-TVI)/C4
      TVV=TVV-(TVI*TPI+EUI)/C4
      GO TO 300
  280 PUV=PUV+SP*((EUO*TPO-EVO)/C1-(EUI*TPI-EVI)/C4)
      PVV=PVV+SP*((EVO*TPO+EUO)/C1-(EVI*TPI+EUI)/C4)
      EUI=ST*EUI
      EUO=ST*EUO
      TUV=TUV+(EUO*TPO-TVO)/C1-(EUI*TPI-TVI)/C4
      TVV=TVV+(TVO*TPO+EUO)/C1-(TVI*TPI+EUI)/C4
  290 IF(KV.GT.NEVP) GO TO 300
      EC=(EC+EVI-EVO)*DL4
      SUMES(KV)=SUMES(KV)+EC
  300 XV1=XV3
  310 CONTINUE
      SUMES(1)=4.*SUMES(1)
      MY=0
  312 DO 314 NC=1,NUP
      DO 314 NS=1,NVS
      MY=MY+1
  314 AZ(MY)=AZ(MY)+SUMES(NS)*C(I,NC)
  330 CONTINUE
      WRITE (9)  (AZ(MY),MY=1,NKP)
      UVP(KP,J)=PUV/PI4/BETA
      VVP(KP,J)=PVV/PI4
      UV(KP,J)=TUV/PI4/BETA
  340 VV(KP,J)=TVV/PI4
  350 CONTINUE
      REWIND 9
      IF(INDT.NE.0) CALL AQPVPT
      RETURN
      END
*DECK AQPVPT
      SUBROUTINE AQPVPT
C
C*                 CALC ADDITIONAL INFLUENCE OF PYLON THICKNESS
C                  DUE TO BREAKS ETC.
C
```

```
      COMMON/DAT/ DA(5000)
C
      DIMENSION   PJC(20)
C
      EQUIVALENCE       (DA(2507),PNVC) ,(DA(3383),XLP)  ,(DA(2512),PJC)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CPP/ HPS,NVSH,NVT,DL ,NEVP,ZCON,YCON,NVC,NVS,NJC,NUP
C
      LARGE        DM1(75130)
     1             ,EV(22)     ,XLV(22)    ,XTLV(22)   ,WPE(39)
     2             ,XD(60)     ,YD(60)     ,ZD(60)
     3             ,XB(400)    ,YB(400)    ,ZB(400)
     4             ,CFI(20)    ,CFJ(20)    ,STH(20)    ,ACT(20)
     5             ,ET(22)     ,XLT(22)    ,XTLT(22)   ,C(20,20)
     6             ,CSJ(20,20),TSJ(20,20),UV(20,20)
     7             ,UVP(20,20),VV(20,20) ,VVP(20,20)
     8             ,DM2(790)   ,TSI(21,20),TZI(21,20),CEL(3700)
     9             ,XQ(650)    ,YQ(650)    ,ZQ(650)    ,AZ(250)    ,SUMES(30)
C
C
      NE1=NVSH
      NE2=NVSH-1
   10 DL4=4./DL/BETA
      TL=2./DL
      X2=TL/HPS
      DTH=PI/PNVC*.5
      DTX=X2*DTH
      TDY=HPS/TL
C*            Q ON PYLON LOOP
      IP=0
      IC=0
      DO 350 KP=1,NE2
      DO 340 J=1,NJC
      JX=PJC(J)
      PUV=0.
      PVV=0.
      TUV=0.
      TVV=0.
      IC=IC+1
      IF(KP.LE.NEVP) GO TO 20
      IP=IP+1
      XP=XD(IP)
```

```
      EVP=YD(IP)/HPS
      GO TO 40
   20 XP=XLV(KP)+XTLV(KP)*CFJ(JX+1)
      EVP=EV(KP)
C       I LOOP
   40 IR=1
      DO 330 I=2,NVC
C       ETAV LOOP
      STHJ=SIN(ACT(I))
      XV3=XLT(1)+XTLT(1)*CFJ(I)
      XV2=XLV(1)+XTLV(1)*CFJ(I)
      XVT=XLV(NVSH)+XTLV(NVSH)*CFJ(I)
      TPT=(XVT-XV3)/TDY
      DO 310 KV=1,NE1
      EVI=0.
      EVO=0.
      EVY=EV(KV)
      IF(KV.NE.NE1) GO TO 120
      TPI=TPT
      TPO=-TPT
      XC=X2*(XP-XVT)
      GO TO 190
  120 IF(KV.LE.NEVP) GO TO 140
      XB1=(XB(IR)+XB(IR+4))/2.
      XB2=(XB(IR+1)+XB(IR+5))/2.
      YB1=(YB(IR)+YB(IR+4))/2.
      YB2=(YB(IR+1)+YB(IR+5))/2.
      TPO=(XB1-XB2)/(YB1-YB2)
      EVY=(YB1+YB2)/2./HPS
      XC=X2*(XP-(XB1+XB2)/2.)
      TPI=TPO
      IR=IR+1
      IF(KV.EQ.NE2) IR=IR+1
      GO TO 190
  140 IF(KV.NE.1) GO TO 180
      TPI=TPT
      TPO=TPT
      XC=4.*X2*(XP-XV2)
      GO TO 190
  180 XV2=XLV(KV)+XTLV(KV)*CFJ(I)
      XV3=XLT(KV)+XTLT(KV)*CFJ(I)
      TPI=(XV2-XV3)/TDY
      TPO=(XV1-XV2)/TDY
```

```
      IF(KV.EQ.NE2) GO TO 182
      IF(ET(KV).LT.0.0) TPI=TPO
  182 XC=X2*(XP-XV2)
  190 YC=TL*(EVP-EVY)
      IF(ABS(YC/2.).GT.XLP) GO TO 300
      IF(KV.EQ.1) YC=4.*YC
      YCP=YC+1.
      YCM=YC-1.
      XTO=XC-TPO
      XTI=XC+TPI
      SQO=SQRT(XTO**2+YCM**2)
      SQI=SQRT(XTI**2+YCP**2)
      SQC=SQRT(XC**2+YC**2)
      C1=TPI**2+1.
      C4=C1
      C2=XC-YC*TPI
      IF(C2**2/C1.LE.1.E-10) GO TO 230
      C3=YC+XC*TPI
      EVI=((C3+C1)/SQI-C3/SQC)/C2
  230 C1=TPO**2+1.
      C2=XC-YC*TPO
      IF(C2**2/C1.LE.1.E-10) GO TO 250
      C3=YC+XC*TPO
  240 EVO=((C3-C1)/SQO-C3/SQC)/C2
  250 IF(KV.EQ.1) GO TO 300
      IF(SQC.GT.1.E-10) GO TO 254
      STS=0.0
      GO TO 256
  254 STS=1./SQC
  256 EUI=1./SQI-STS
      EUO=1./SQO-STS
      AYC=ABS(YC/X2)
      SLOPE=(TSI(KV,I)+TSI(KV,I-1))/2.0
      DEFL=(TZI(KV,I)+TZI(KV,I-1))/2.0
      ST=SLOPE*STHJ*DTX
      SP=DEFL/STHJ*DTX*2.0
      TVI=ST*EVI
      TVO=ST*EVO
  260 IF(KV.NE.NE1) GO TO 280
      PUV=PUV-SP*(EUI*TPI-EVI)/C4
      PVV=PVV-SP*(EVI*TPI+EUI)/C4
      EUI=ST*EUI
      TUV=TUV-(EUI*TPI-TVI)/C4
```

```
      TVV=TVV-(TVI*TPI+EUI)/C4
      GO TO 300
  280 PUV=PUV+SP*((EUO*TPO-EVO)/C1-(EUI*TPI-EVI)/C4)
      PVV=PVV+SP*((EVO*TPO+EUO)/C1-(EVI*TPI+EUI)/C4)
      EUI=ST*EUI
      EUO=ST*EUO
      TUV=TUV+(EUO*TPO-TVO)/C1-(EUI*TPI-TVI)/C4
      TVV=TVV+(TVO*TPO+EUO)/C1-(TVI*TPI+EUI)/C4
  300 XV1=XV3
  310 CONTINUE
  330 CONTINUE
      UVP(KP,J)=PUV/PI4/BETA+UVP(KP,J)
      VVP(KP,J)=PVV/PI4+VVP(KP,J)
      UV(KP,J)=TUV/PI4/BETA+UV(KP,J)
  340 VV(KP,J)=TVV/PI4+VV(KP,J)
  350 CONTINUE
      RETURN
      END
*DECK NACELL
      SUBROUTINE NACELL
C
C*                 CONTROL ROUTINE TO CALC. INFLUENCE MATRICES OF NACELLE
C                  ON OTHER COMPONENTS AND NACELLE ON NACELLE
C
      COMMON/DAT/ DA(5000)
C
      EQUIVALENCE       (DA(2492),PDA)   ,(DA(3391),XNO)  ,(DA(3392),YNO)
     1,(DA(3393),ZNO)
C
      COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP
      COMMON/CFG/ ID1(5),NQB,NQF,NQW,NQP,NQN
C
      LARGE        DM1(81191),XOB(1000) ,YOB(1000) ,ZOB(1000) ,DM2(700)
     1             ,X2(1000)  ,Y2(1000)  ,SDT(1000) ,CDT(1000)
C
C                  INITIAL
      PDAR=PDA/RC
      NQT=NQB+NQF+NQW+NQP
      YNB=YNO*BETA
      ZNB=ZNO*BETA
C                  INFLUENCE OF RIGHT SIDE NACELLE
      L1=NQT
      DO 10 I=1,L1
```

```
      X2(I)=XOB(I)-XNO
      TY=YOB(I)-YNB
      TZ=ZOB(I)-ZNB
      YNQ=-TY*SINDP-TZ*COSDP
      ZNQ=TY*COSDP-TZ*SINDP
      Y2(I)=SQRT(YNQ**2+ZNQ**2)
      THE=ATAN2(ZNQ,YNQ)
      ANG=PDAR-THE
      SDT(I)=SIN(ANG)
   10 CDT(I)=COS(ANG)
      CALL AQXVN(1,L1)
C                TEST FOR IMAGE
      SYM=YNO+PDA
      IF(SYM.EQ.0.0) GO TO 30
C                INFLUENCE OF LEFT SIDE NACELLE
      L1=NQT+NQN
      DO 20 I=1,L1
      X2(I)=XOB(I)-XNO
      TY=YOB(I)+YNB
      TZ=ZOB(I)-ZNB
      YNQ=TY*SINDP-TZ*COSDP
      ZNQ=TY*COSDP+TZ*SINDP
      Y2(I)=SQRT(YNQ**2+ZNQ**2)
      THE=ATAN2(ZNQ,YNQ)
      ANG=-PDAR-THE
      SDT(I)=SIN(ANG)
   20 CDT(I)=COS(ANG)
      CALL AQXVN (2,L1)
C
   30 REWIND 9
      WRITE (6,50)
   50 FORMAT (44H0 ** COMPLETED INFLUENCE MATRICES OF NACELLE)
C
      RETURN
      END
*DECK AQXVN
      SUBROUTINE AQXVN (IPASS,L1)
C
C*               CALCULATE INFLUENCE MATRICES OF NACELLE
C
      COMMON/DAT/ DA(5000)
C
      EQUIVALENCE     (DA(2492),PDA)  ,(DA(3392),YNO)  ,(DA(3385),XLNR)
```

```
     1,(DA(3384),XLNC)
C
      COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP
      COMMON/CFG/ ID1(5),NQB,NQF,NQW,NQP,NQN
      COMMON/CPN/ NB,ND(3),NT,BON,YZERO,IAC,I,J,J1
C
      LARGE       DM1(81191),XOB(1000) ,YOB(1000) ,ZOB(1000)
     1            ,X1(140)   ,Y1(140)   ,SINA(140) ,COSA(140) ,DELS(140)
     2            ,X2(1000)  ,Y2(1000)  ,SDT(1000) ,CDT(1000)
     3            ,A(140)    ,B(140)    ,C(140)    ,AX(140)   ,AY(140)
C
C          * START
      IS=NQB+NQF+NQW
      IE=IS+NQP
      SYM=YNO+PDA
      BON=1.0
      YZERO=0.0
      XLNR2=2.0*XLNR
      XLNC2=2.0*XLNC
C          *  I MIDPOINT LOOP
   10 CONTINUE
      DO 400 I=1,LI
C          *  J ELEMENT LOOP
C             J1 IS THE COORDINATE COUNTER
C             J IS THE ELEMENT COUNTER
      J1=0
      N1=0
      DO 110 K=1,NB
      M1=N1+1
      N1=N1+ND(K)-1
      DO 100 J=M1,N1
      J1=J1+1
C          * COMPUTE X,Y,Z MATRICES
      IF(BON.EQ.0.0) GO TO 50
      X11=0.5*(X1(J1)+X1(J1+1))
      Y11=0.5*(Y1(J1)+Y1(J1+1))
   40 IF(Y11.EQ.0.0) GO TO 45
      IF(ABS(Y2(I)/Y11   ).GT.XLNR2) GO TO 45
      IF(ABS((X11   -X2(I))/Y11   ).LE.XLNC2) GO TO 50
   45 AX(J)=0.0
      AY(J)=0.0
      GO TO 100
   50 CALL XYZ
```

```
  100 CONTINUE
  110 J1=J1+1
      NT=N1
C                SAVE A,B,C MATRICES ONTAPE
      IF(IPASS.EQ.2) GO TO 140
      DO 130 IK=1,NT
      A(IK)=AX(IK)
      B(IK)=-AY(IK)*SDT(I)
  130 C(IK)=-AY(IK)*CDT(I)
      IF(SYM.EQ.0.0) GO TO 160
      WRITE (11) (A(IK),B(IK),C(IK),IK=1,NT)
      GO TO 400
C
  140 READ (11) (A(IK),B(IK),C(IK),IK=1,NT)
      DO 150 IK=1,NT
      A(IK)=A(IK)+AX(IK)
      B(IK)=B(IK)-AY(IK)*SDT(I)
  150 C(IK)=C(IK)-AY(IK)*CDT(I)
C
  160 IF(BON.EQ.0.0) GO TO 180
      IF(I.LE.IS) GO TO 200
      IF(I.GT.IE) GO TO 180
      DO 170 IK=1,NT
      YT=B(IK)
      B(IK)=YT*SINDP+C(IK)*COSDP
  170 C(IK)=-YT*COSDP+C(IK)*SINDP
      GO TO 200
C
  180 DO 190 IK=1,NT
      YT=B(IK)
      B(IK)=-YT*SINDP-C(IK)*COSDP
  190 C(IK)=YT*COSDP-C(IK)*SINDP
C
  200 WRITE (9) (A(IK),B(IK),C(IK),IK=1,NT)
C
  400 CONTINUE
      IF(IPASS.EQ.2) GO TO 500
C               * TEST IF OFF BODY COMPLETED
  410 IF(BON) 420,500,420
  420 BON=0.0
      L1=NT
      N1=0
      J1=0
```

```
      DO 450 K=1,NB
      M1=N1+1
      N1=N1+ND(K)-1
      DO 440 J=M1,N1
      J1=J1+1
      X2(J)=(X1(J1+1)+X1(J1))/2.
      Y2(J)=(Y1(J1+1)+Y1(J1))/2.
  440 CONTINUE
  450 CONTINUE
      GO TO 10
 500  CONTINUE
      REWIND 11
      RETURN
      END
*DECK XYZ
      SUBROUTINE XYZ
C
C                   * CONTROL FOR X,Y,Z MATRICES COMPUTATION
C
C
      COMMON/CPN/ NB,ND(3),NT,BON,YZERO,IAC,I,J,J1
     1            ,SJ,DS,DX,DY,NI,XJ,YJ,XK,EEK,EKK
C
      LARGE       DM1(81191),XOB(1000) ,YOB(1000) ,ZOB(1000)
     1            ,X1(140)   ,Y1(140)   ,SINA(140) ,COSA(140) ,DELS(140)
     2            ,X2(1000)  ,Y2(1000)  ,SDT(1000) ,CDT(1000)
     3            ,A(140)    ,B(140)    ,C(140)    ,AX(140)   ,AY(140)
C
C                   * START
      IF (BON) 100,10,100
   10 IF (J-I) 110,20,110
C                   * J EQUAL I PATH
   20 T1=.5*DELS(J)
      SJ=T1/Y2(J)
      IF (SJ-.08) 30,30,40
   30 CALL XYZ1
      GO TO 1000
   40 SJ=.08
      CALL XYZ1
      NI=33
      T2=.08*Y2(J)
      DS=(T1-T2)/32.
      DX=DS*COSA(J)
```

```
      DY=DS*SINA(J)
      XJ=X2(J)+T2*COSA(J)-DX
      YJ=Y2(J)+T2*SINA(J)-DY
      CALL XYZ2
      GO TO 300
C              * INITIAL Y COORDINATE MID-POINT FOR ZERO TEST
  100 YZERO=Y2(I)-.000001
C              * J NOT EQUAL I PATH
C              * COMPUTE MINIMUM DISTANCE TO I MIDPOINT
  110 D1=(X2(I)-X1(J1))**2+(Y2(I)-Y1(J1))**2
      X22=(X1(J1+1)+X1(J1))*0.5
      Y22=(Y1(J1+1)+Y1(J1))*0.5
      D2=(X2(I)-X22)**2+(Y2(I)-Y22)**2
      D3=(X2(I)-X1(J1+1))**2+(Y2(I)-Y1(J1+1))**2
      IF (D1-D2) 130,130,120
  120 IF (D2-D3) 150,150,140
  130 IF (D1-D3) 160,160,140
  140 DM=SQRT(D3)
      GO TO 170
  150 DM=SQRT(D2)
      GO TO 170
  160 DM=SQRT(D1)
C              * COMPUTE NO. OF INTERVALS(NI) AND DELTA S (DS)
C                FOR SIMPSON RULE INTEGRATION
  170 IF (DM.EQ.0.0) GO TO 200
      NI=8.*DELS(J)/DM+0.9
      IF (NI) 180,180,190
  180 NI=3
      DS=DELS(J)/2.
      GO TO 220
  190 NI=NI+NI
      IF (NI-128) 210,200,200
  200 NI=129
      DS=DELS(J)/128.
      GO TO 220
  210 XNI=NI
      DS=DELS(J)/XNI
      NI=NI+1
  220 DX=DS*COSA(J)
      DY=DS*SINA(J)
  300 XJ=X1(J1)-DX
      YJ=Y1(J1)-DY
      CALL XYZ2
```

```
 1000 RETURN
      END
*DECK XYZ1
      SUBROUTINE XYZ1
C
C              * COMPUTE X,Y,Z MATRICES FOR SJ LESS THAN OR EQUAL .08
C
C
      COMMON/CPN/ NB,ND(3),NT,BON,YZERO,IAC,I,J,J1
     1           ,SJ,DS,DX,DY,NI,XJ,YJ,XK,EEK,EKK
C
      LARGE       DM1(81191),XOB(1000) ,YOB(1000) ,ZOB(1000)
     1           ,X1(140)   ,Y1(140)   ,SINA(140) ,COSA(140) ,DELS(140)
     2           ,X2(1000)  ,Y2(1000)  ,SDT(1000) ,CDT(1000)
     3           ,A(140)    ,B(140)    ,C(140)    ,AX(140)   ,AY(140)
C
C              * START
C              * INITIALIZE
      T1=SJ*SJ
      T2=ALOG(SJ/8.)
      T3=SINA(J)*SINA(J)
      T4=T2+T3
      T5=.666666667*T3
      T6=T5*T3
      T7=SJ+SJ
      T8=T7+T7
      T9=6.2831853*COSA(J)
      T10=6.2831853*SINA(J)
C              * AXIS FLOW
   20 T11=T1*SJ
      AX(J)=T10+SINA(J)*COSA(J)*(T7+(T4+2.16666667)*T11/12.)
      AY(J)=T7*T4-T9-(1.+T2-T3-T6)*T11/8.
  100 RETURN
      END
*DECK XYZ2
      SUBROUTINE XYZ2
C
C              * COMPUTE X,Y,Z MATRICES USING SIMPSON RULE INTEGRATION
C
C
      COMMON/CPN/ NB,ND(3),NT,BON,YZERO,IAC,I,J,J1
     1           ,SJ,DS,DX,DY,NI,XJ,YJ,XK,EEK,EKK
C
```

```
      LARGE         DM1(81191),XOB(1000) ,YOB(1000) ,ZOB(1000)
     1              ,X1(140)    ,Y1(140)    ,SINA(140) ,COSA(140) ,DELS(140)
     2              ,X2(1000)   ,Y2(1000)   ,SDT(1000) ,CDT(1000)
     3              ,A(140)     ,B(140)     ,C(140)    ,AX(140)   ,AY(140)
C
C                * START
C                * INITIALIZE
      S2=.666666667*DS
      T1=Y2(I)*Y2(I)
C                * NO. OF INTERVAL LOOP
      DO 1000 IS=1,NI
      XJ=XJ+DX
      YJ=YJ+DY
      T2=YJ*YJ
      T3=X2(I)-XJ
      T4=T3*T3
      T5=(Y2(I)+YJ)**2
      T6*T4+T5
      T7=SQRT(T6)

      T8=T2+T4
      T9=(Y2(I)-YJ)**2
      T10=T9+T4
C                * COMPUTE ELLIPIC INTEGRAL
      XK=(T4+T9)/T6
      CALL ELIP
C                * AXIS FLOW
   20 T11=YJ/T7
      F1=T11*T3*EEK/T10
      IF (Y2(I).NE.0.0) GO TO 25
      F2=0.0
      GO TO 26
   25 F2=T11*(EKK+EEK*(T1-T8)/T10)/Y2(I)
   26 F3=T11*EKK
C                * SIMPSON RULE INTEGRATION
      IF (IS-1) 30,30,40
C                * FIRST PASS
   30 AXS=F1
      AYS=F2
      IA=0
      GO TO 1000
   40 IF (IS-NI) 50,80,50
   50 IF (IA) 70,60,70
```

```
C                 * EVEN PASS
   60 AXS=AXS+4.*F1
      AYS=AYS+4.*F2
      IA=1
      GO TO 1000
C                 * ODD PASS
   70 AXS=AXS+F1+F1
      AYS=AYS+F2+F2
      IA=0
      GO TO 1000
C                 * LAST PASS
   80 S4=S2+S2
      IF (J-I) 100,90,100
   90 IF (BON.NE.0.0) GO TO 100
      AX(J)=AX(J)-S4*(AXS+F1)
      AY(J)=AY(J)-S2*(AYS+F2)
      GO TO 1000
  100 AX(J)=-S4*(AXS+F1)
      AY(J)=-S2*(AYS+F2)
 1000 CONTINUE
      RETURN
      END
*DECK ELIP
      SUBROUTINE ELIP
C
C                 * HASTINGS APPROXIMATION FOR ELLIPTIC INTERGALS
C
      COMMON/CPN/ NB,ND(3),NT,BON,YZERO,IAC,I,J,J1
     1           ,SJ,DS,DX,DY,NI,XJ,YJ,XK,EEK,EKK
C
C                 * START
   10 ETA=XK
      IF (ETA) 20,20,40
   20 WRITE (6,30) ETA,I,J
   30 FORMAT ( 1H0 36H *** ERROR IN SUBROUTINE ELIP * ETA= F15.8 /
     1        6H  * I=I6,5X,5H * J=I6,5X,27H * ALOG(ETA) SET TO -88.028)
      ELN=-88.028
      ETA=0.0
      GO TO 50
   40 ELN=ALOG(ETA)
   50 EKK=1.38629436112+ETA*(0.09666344259+ETA*(0.03590092383+ETA*(
     1    0.03742563713+ETA*0.01451196212)))-ELN*(0.5+ETA*(
     2    0.12498593597+ETA*(0.06880248576+ETA*(0.03328355346+ETA*
```

```
     3    0.00441787012])))
      EEK=1.+ETA*(0.44325141463+ETA*(0.06260601220+ETA*(0.04757383546
     1    +ETA*0.01736506451)))-ELN*(ETA*(0.24998368310+ETA*(
     2    0.09200180037+ETA*(0.04069697526+ETA*0.00526449639))))
      RETURN
      END
*DECK MATAPE
      SUBROUTINE MATAPE
C
C*              COMBINE INFLUENCE MATRICES AND SETUP FOR MATRIX SOLU.
C*              ALSO CALC INFLUENCE OF WING AND PYLON THICKNESS ON
C*              BOUNDARY CONDITION (MATRIX-B)
C
      COMMON/DAT/ DA(5000)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CFG/ IDB,IDF,IDW,IDP,IDN
     1           ,NQB,NQF,NQW,NQP,NQN
     2           ,NKB,NKF,NKW,NKP,NKN
C
      LARGE       BT(1000)   ,BW(1000)   ,BP(1000)   ,DM1(5850)
     1           ,BXN(650)   ,BYN(650)   ,BZN(650)   ,DM2(23050)
     2           ,FXN(650)   ,FYN(650)   ,FZN(650)   ,DM3(48671)
     3           ,SINAN(140),COSAN(140),DM4(140)
     4           ,AX(250)    ,AY(250)    ,AZ(250)    ,A(250)
C
C*              INITIAL
      NQT=NQB+NQF+NQW+NQP+NQN
      DO 10 I=1,NQT
      BW(I)=0.0
   10 BP(I)=0.0
      IX=0
C
C*              CONTROL POINTS Q ON FUSELAGE
      IF(IDB.EQ.0) GO TO 100
      DO 90 IQ=1,NQB
      IC=0
C               COMBINE INFL OF FUSELAGE
      READ (19) (AX(I),AY(I),AZ(I),I=1,NKB)
      DO 20 I=1,NKB
      IC=IC+1
   20 A(IC)=AX(I)*BXN(IQ)+AY(I)*BYN(IQ)+AZ(I)*BZN(IQ)
C               COMBINE INFL OF FANPOD
```

```
      IF(IDF.EQ.0) GO TO 40
      READ (20) (AX(I),AY(I),AZ(I),I=1,NKF)
      DO 30 I=1,NKF
      IC=IC+1
   30 A(IC)=AX(I)*BXN(IQ)+AY(I)*BYN(IQ)+AZ(I)*BZN(IQ)
C                COMBINE INFL OF WING
   40 IX=IX+1
      IF(IDW.EQ.0) GO TO 60
      READ (21) (AX(I),AY(I),AZ(I),I=1,NKW),TUV,TVV,TWV
      BW(IX)=BETA*(BETA*TUV*BXN(IQ)+TVV*BYN(IQ)+TWV*BZN(IQ))
      DO 50 I=1,NKW
      IC=IC+1
   50 A(IC)=AX(I)*BXN(IQ)+AY(I)*BYN(IQ)+AZ(I)*BZN(IQ)
C                COMBINE INFL OF PYLON
   60 IF(IDP.EQ.0) GO TO 80
      READ (22) (AX(I),AY(I),AZ(I),I=1,NKP),TUV,TVV,TWV
      BP(IX)=BETA*(BETA*TUV*BXN(IQ)+TVV*BYN(IQ)+TWV*BZN(IQ))
      DO 70 I=1,NKP
      IC=IC+1
   70 A(IC)=AX(I)*BXN(IQ)+AY(I)*BYN(IQ)+AZ(I)*BZN(IQ)
C                WRITE INFL MATRIX ON UNIT 12
   80 WRITE (12) (A(I),I=1,IC)
   90 CONTINUE
C
C*               CONTROL POINTS Q ON FANPOD
  100 IF(IDF.EQ.0) GO TO 200
      DO 190 IQ=1,NQF
      IC=0
C                COMBINE INFL OF FUSELAGE
      IF(IDB.EQ.0) GO TO 130
      READ (19) (AX(I),AY(I),AZ(I),I=1,NKB)
      DO 120 I=1,NKB
      IC=IC+1
  120 A(IC)=AX(I)*FXN(IQ)+AY(I)*FYN(IQ)+AZ(I)*FZN(IQ)
  130 CONTINUE
C                COMBINE INFL OF FANPOD
      READ (20) (AX(I),AY(I),AZ(I),I=1,NKF)
      DO 140 I=1,NKF
      IC=IC+1
  140 A(IC)=AX(I)*FXN(IQ)+AY(I)*FYN(IQ)+AZ(I)*FZN(IQ)
      IX=IX+1
C                COMBINE INFL OF WING
      IF(IDW.EQ.0) GO TO 160
```

```
      READ (21) (AX(I),AY(I),AZ(I),I=1,NKW),TUV,TVV,TWV
      BW(IX)=BETA*(BETA*TUV*FXN(IQ)+TVV*FYN(IQ)+TWV*FZN(IQ))
      DO 150 I=1,NKW
      IC=IC+1
  150 A(IC)=AX(I)*FXN(IQ)+AY(I)*FYN(IQ)+AZ(I)*FZN(IQ)
C                 COMBINE INFL OF PYLON
  160 IF(IDP.EQ.0) GO TO 180
      READ (22) (AX(I),AY(I),AZ(I),I=1,NKP),TUV,TVV,TWV
      BP(IX)=BETA*(BETA*TUV*FXN(IQ)+TVV*FYN(IQ)+TWV*FZN(IQ))
      DO 170 I=1,NKP
      IC=IC+1
  170 A(IC)=AX(I)*FXN(IQ)+AY(I)*FYN(IQ)+AZ(I)*FZN(IQ)
C                 WRITE INFL MATRIX ON UNIT 12
  180 WRITE (12) (A(I),I=1,IC)
  190 CONTINUE
C
C*                CONTROL POINTS Q ON WING
  200 IF(IDW.EQ.0) GO TO 300
      DO 290 IQ=1,NQW
      IC=0
C                 COMBINE INFL OF FUSELAGE
      IF(IDB.EQ.0) GO TO 230
      READ (19) (AX(I),AY(I),AZ(I),I=1,NKB)
      DO 220 I=1,NKB
      IC=IC+1
  220 A(IC)=-AZ(I)
C                 COMBINE INFL OF FANPOD
  230 IF(IDF.EQ.0) GO TO 250
      READ (20) (AX(I),AY(I),AZ(I),I=1,NKF)
      DO 240 I=1,NKF
      IC=IC+1
  240 A(IC)=-AZ(I)
  250 IX=IX+1
C                 COMBINE INFL OF WING
      READ (21) (AZ(I),I*1,NKW)
      DO 260 I=1,NKW
      IC=IC+1
  260 A(IC)=AZ(I)
C                 COMBINE INFL OF PYLON
      IF(IDP.EQ.0) GO TO 280
      READ (22) (AX(I),AY(I),AZ(I),I=1,NKP),TUV,TVV,TWV
      BP(IX)=-TWV*BETA
      DO 270 I=1,NKP
```

```
      IC=IC+1
  270 A(IC)=-AZ(I)
C                 WRITE INFL MATRIX ON UNIT 12
  280 WRITE (12) (A(I),I=1,IC)
  290 CONTINUE
C
C*                CONTROL POINTS Q ON PYLON
  300 IF(IDP.EQ.0) GO TO 400
      DO 390 IQ=1,NQP
      IC=0
C                 COMBINE INFL OF FUSELAGE
      IF(IDB.EQ.0) GO TO 330
      READ (19) (AX(I),AY(I),AZ(I),I=1,NKB)
      DO 320 I=1,NKB
      IC=IC+1
  320 A(IC)=-AZ(I)
C                 COMBINE INFL OF FANPOD
  330 IF(IDF.EQ.0) GO TO 350
      READ (20) (AX(I),AY(I),AZ(I),I=1,NKF)
      DO 340 I=1,NKF
      IC=IC+1
  340 A(IC)=-AZ(I)
  350 IX=IX+1
C                 COMBINE INFL OF WING
      IF(IDW.EQ.0) GO TO 370
      READ (21) (AX(I),AY(I),AZ(I),I=1,NKW),TUV,TVV,TWV
      BW(IX)=-BETA*TWV
      DO 360 I=1,NKW
      IC=IC+1
  360 A(IC)=-AZ(I)
  370 CONTINUE
C                 COMBINE INFL OF PYLON
      READ (22) (AX(I),AY(I),AZ(I),I=1,NKP),TUV,TVV,TWV          56195
      DO 380 I=1,NKP  BP(IX)=-BETA*TWV
      IC=IC+1
  380 A(IC)=AZ(I)
C                 WRITE INFL MATRIX ON UNIT 12
      WRITE (12) (A(I),I=1,IC)
  390 CONTINUE
C
C*                CONTROL POINTS Q ON NACELLE
  400 IF(IDN.EQ.0) GO TO 500
C*                THE MINUS SIGN BEFORE COSAN IS DUE TO AY BEING IN THE
```

```
C*                PYLON COORD. SYS.. AY IS EQUAL TO -AY IN NACELLE SYS.
      DO 490 IQ=1,NQN
      IC=0
C                 COMBINE INFL OF FUSELAGE
      IF(IDB.EQ.0) GO TO 420
      READ (19) (AX(I),AY(I),AZ(I),I=1,NKB)
      DO 410 I=1,NKB
      IC=IC+1
  410 A(IC)=-SINAN(IQ)*AX(I)-COSAN(IQ)*AY(I)
C                 COMBINE INFL OF FANPOD
  420 IF(IDF.EQ.0) GO TO 440
      READ (20) (AX(I),AY(I),AZ(I),I=1,NKF)
      DO 430 I=1,NKF
      IC=IC+1
  430 A(IC)=-SINAN(IQ)*AX(I)-COSAN(IQ)*AY(I)
  440 IX=IX+1
C                 COMBINE INFL OF WING
      IF(IDW.EQ.0) GO TO 460
      READ (21) (AX(I),AY(I),AZ(I),I=1,NKW),TUV,TVV,TWV
      BW(IX)=BETA*(-SINAN(IQ)*TUV*BETA-COSAN(IQ)*TVV)
      DO 450 I=1,NKW
      IC=IC+1
  450 A(IC)=-SINAN(IQ)*AX(I)-COSAN(IQ)*AY(I)
C                 COMBINE INFL OF PYLON
  460 IF(IDP.EQ.0) GO TO 480
      READ (22) (AX(I),AY(I),AZ(I),I=1,NKP),TUV,TVV,TWV
      BP(IX)=BETA*(-SINAN(IQ)*TUV*BETA-COSAN(IQ)*TVV)
      DO 470 I=1,NKP
      IC=IC+1
  470 A(IC)=-SINAN(IQ)*AX(I)-COSAN(IQ)*AY(I)
C                 WRITE INFL MATRIX ON UNIT 12
  480 WRITE (12) (A(I),I=1,IC)
  490 CONTINUE
C
C*                REWIND I/O UNITS
  500 REWIND 12
      IF(IDB.NE.0) REWIND 19
      IF(IDF.NE.0) REWIND 20
      IF(IDW.NE.0) REWIND 21
      IF(IDP.NE.0) REWIND 22
C
C*                SETUP ON UNIT 10 THE INFLUENCE MATRICES OF NACELLE
      IF(IDN.EQ.0) GO TO 700
```

```
C                 Q ON FUSELAGE
      IF(IDB.EQ.0) GO TO 530
      DO 520 IQ=1,NQB
      READ (9) (AX(I),AY(I),AZ(I),I=1,NKN)
      DO 510 I=1,NKN
  510 A(I)=AX(I)*BXN(IQ)+AY(I)*BYN(IQ)+AZ(I)*BZN(IQ)
  520 WRITE (10) (A(I),I=1,NKN)
C                 Q ON FANPOD
  530 IF(IDF.EQ.0) GO TO 560
      DO 550 IQ=1,NQF
      READ (9) (AX(I),AY(I),AZ(I),I=1,NKN)
      DO 540 I=1,NKN
  540 A(I)=AX(I)*FXN(IQ)+AY(I)*FYN(IQ)+AZ(I)*FZN(IQ)
  550 WRITE (10) (A(I),I=1,NKN)
C                 Q ON WING
  560 IF(IDW.EQ.0) GO TO 590
      DO 580 IQ=1,NQW
      READ (9) (AX(I),AY(I),AZ(I),I=1,NKN)
      DO 570 I=1,NKN
  570 A(I)=-AZ(I)
  580 WRITE (10) (A(I),I=1,NKN)
  590 IF(IDP.EQ.0) GO TO 620
      DO 610 IQ=1,NQP
C                 Q ON PYLON
      READ (9) (AX(I),AY(I),AZ(I),I=1,NKN)
      DO 600 I=1,NKN
  600 A(I)=-AZ(I)
  610 WRITE (10) (A(I),I=1,NKN)
C                 Q ON NACELLE
  620 DO 640 IQ=1,NQN
      READ (9) (AX(I),AY(I),AZ(I),I=1,NKN)
      DO 630 I=1,NKN
  630 A(I)=-SINAN(IQ)*AX(I)+COSAN(IQ)*AY(I)
  640 WRITE (10) (A(I),I=1,NKN)
C
      REWIND 9
      REWIND 10
C
  700 RETURN
      END
*DECK MATB
      SUBROUTINE MATB
C
```

```
C*                CALC MATRIX-B
C
      COMMON/DAT/ DA(5000)
C
      EQUIVALENCE     (DA(6),FAA)        ,(DA(8),WAA)    ,(DA(10),WADI)
     1,(DA(9),WAAI)    ,(DA(4006),BAA)
     2,(DA(1960),DAF) ,(DA(1961),PXCF) ,(DA(1962),ETFI),(DA(1963),ETFO)
     3,(DA(1965),DAK) ,(DA(1966),PXCK) ,(DA(1967),ETKI),(DA(1968),ETKO)
     4,(DA(2504),PAA) ,(DA(2505),PAAI) ,(DA(2506),PADI)
C
      COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP
      COMMON/CFG/ IDB,IDF,IDW,IDP,IDN,NQB,NQF,NQW,NQP,NQN
      COMMON/CPB/ IB1(9),SIBA,COBA
      COMMON/CPF/ IF1(9),SIFA,COFA
      COMMON/CPW/ IW1(13),NJC,NJS
      COMMON/CFK/ NDA
C
      LARGE        BT(1000)   ,BW(1000)   ,BP(1000)   ,DM1(5850)
     1            ,BXN(650)   ,BYN(650)   ,BZN(650)   ,DM2(23050)
     2            ,FXN(650)   ,FYN(650)   ,FZN(650)   ,DM3(16956)
     3            ,BS(650)    ,EP(30)     ,XTLP(30)   ,XOCP(30)   ,DM4(26285)
     4            ,BSP(360)   ,DM5(4330)  ,SINAN(140)
C
C*                START
C                 SETUP SB DUE TO THICKNESS
      NQT=NQB+NQF+NQW+NQP+NQN
      DO 10 I=1,NQT
   10 BT(I)=BW(I)+BP(I)
C*                CALC MATRIX-B OF THE BODY
      IF(IDB.EQ.0) GO TO 25
      BAAB=BAA/RC*BETA
      COBA=COS(BAAB)
      SIBA=SIN(BAAB)
      DO 20 I=1,NQB
   20 BT(I)=-COBA*BXN(I)-SIBA*BZN(I)-BT(I)
   25 IS=NQB
C*                CALC MATRIX-B OF THE FANPOD
      IF(IDF.EQ.0) GO TO 40
      FAAB=FAA/RC*BETA
      COFA=COS(FAAB)
      SIFA=SIN(FAAB)
      DO 30 I=1,NQF
      IS=IS+1
```

```
   30 BT(IS)=-COFA*FXN(I)-SIFA*FZN(I)-BT(IS)
C*                 CALC MATRIX-B OF THE WING
   40 IF(IDW.EQ.0) GO TO 150
      WAAR=WAA*BETA/RC
      IF(WAAI) 70,70,90
   70 DO 80 I=1,NQW
      IS=IS+1
      BS(I)=WAAR/BETA
   80 BT(IS)=WAAR-BT(IS)
      GO TO 140
   90 IF(WADI) 100,100,110
  100 CALL WCAS
      GO TO 120
  110 CALL WCZCS
  120 DO 130 I=1,NQW
      IS=IS+1
      BS(I)=BS(I)+WAAR/BETA
  130 BT(IS)=BS(I) *BETA     -BT(IS)
  140 CONTINUE
C*                 ADD FLAP AND/OR KRUEGER DEFLECTION
      IF(NDA.EQ.0) GO TO 150
      IB=NQB+NQF
      AF=ATAN(BETA*TAN(DAF))
      BCDF=COS(AF)
      BSDF=SIN(AF)
      AK=ATAN(BETA*TAN(DAK))
      BCDK=COS(AK)
      BSDK=SIN(AK)
      DO 146 IS=1,NJS
      DO 144 IC=1,NJC
      XC=XOCP(IC)
      IB=IB+1
      IF(DAF.EQ.0.0) GO TO 142
      IF(EP(IS).LT.ETFI) GO TO 142
      IF(EP(IS).GT.ETFO) GO TO 142
      IF(XC.LE.PXCF) GO TO 142
      BT(IB)=BT(IB)*BCDF+BSDF
  142 IF(DAK.EQ.0.0) GO TO 144
      IF(EP(IS).LT.ETKI) GO TO 144
      IF(EP(IS).GT.ETKO) GO TO 144
      IF(XC.GE.PXCK) GO TO 144
      BT(IB)=BT(IB)*BCDK-BSDK
  144 CONTINUE
```

```
  146 CONTINUE
  150 IF(IDP.EQ.0) GO TO 240
C*                 CALC MATRIX-B OF THE PYLON
      PAAR=PAA*BETA/RC*SINDP
      IF(PAAI) 170,170,190
  170 DO 180 I=1,NQP
      IS=IS+1
      BSP(I)=PAAR/BETA
  180 BT(IS)=PAAR-BT(IS)
      GO TO 240
  190 IF(PADI) 200,200,210
  200 CALL PCAS
      GO TO 220
  210 CALL PCZCS
  220 DO 230 I=1,NQP
      IS=IS+1
      BSP(I)=BSP(I)+PAAR/BETA
  230 BT(IS)=BSP(I) *BETA     -BT(IS)
  240 IF(IDN.EQ.0) GO TO 280
C*                 CALC MATRIX-B OF THE NACELLE
      DO 250 I=1,NQN
      IS=IS+1
  250 BT(IS)=SINAN(I)-BT(IS)
C*                 PRINT MATRIX-B
  280 IF(DA(12).LT.0.0) GO TO 300
      WRITE (6,290) (BT(IS),IS=1,NQT)
  290 FORMAT (12H1 * MATRIX-B/(1H 10F10.5))
  300 RETURN
      END
*DECK WCAS
      SUBROUTINE WCAS
C
C     SUBR. TLU TABLE LOOKUP FOR THE LOCAL ANGLE OF ATTACK AT SPECIFIED
C     ETA AND X/C.
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION  BI(1)     ,WXC(24)   ,WET(19)    ,WJC(30)    ,WJS(30)
     1           ,TWI(19)
C
      EQUIVALENCE     (DA(706),WXC)    ,(DA(731),WET)   ,(DA(1660),WJC)
     1,(DA(1690),WJS) ,(DA(1970),TWI)   ,(DA(1270),WNVC)
     2,(DA(750),BI)
```

```
C
      COMMON/CPW/ DWI(13),NJC,NJS
C
      LARGE        BOU(3000) ,FUS(25000),FAN(23700)
     1             ,EV(52)    ,DM1(1004) ,BS(650)    ,DM2(60)    ,XOCP(30)
C
C                  START
      NXW=DA(705)
      NEW=DA(730)
      IB=0
      DO 620 IJS=1,NJS
      IE=WJS(IJS)
      IF(EV(IE)-WET(1)) 510,510,515
  510 J=1
      GO TO 530
  515 DO 520 NC=2,NEW
      J=NC
      IF(EV(IE)-WET(J)) 540,530,520
  520 CONTINUE
  530 RATY=0.0
      GO TO 550
  540 RATY=(WET(J)-EV(IE))/(WET(J)-WET(J-1))
  550 TWB=TWI(J)-RATY*(TWI(J)-TWI(J-1))
      DO 610 IJC=1,NJC
      XC=XOCP(IJC)
      IF(XC-WXC(1)) 560,560,565
  560 I=1
      GO TO 580
  565 DO 570 NC=2,NXW
      I=NC
      IF(XC-WXC(I)) 590,580,570
  570 CONTINUE
  580 RATX=0.0
      GO TO 600
  590 RATX=(WXC(I)-XC)/(WXC(I)-WXC(I-1))
  600 NPX=I+NXW*(J-1)
      AX=BI(NPX)-RATX*(BI(NPX)-BI(NPX-1))
      NPX=NPX-NXW
      BX=BI(NPX)-RATX*(BI(NPX)-BI(NPX-1))
      IB=IB+1
  610 BS(IB)=AX-RATY*(AX-BX)+TWB
  620 CONTINUE
      RETURN
```

```
      END
*DECK WCZCS
      SUBROUTINE WCZCS
C
C*                CALC. LOCAL ANGLES OF ATTACK FROM DEFLECTION INPUT(Z/C)
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION    WXC(24)    ,WET(19)    ,WAD(456)  ,TWI(19)    ,WJC(30)
     1             ,WJS(30)
C
      EQUIVALENCE       (DA(706),WXC)    ,(DA(731),WET)   ,(DA(750),WAD)
     1,(DA(1970),TWI) ,(DA(1660),WJC)   ,(DA(1690),WJS)  ,(DA(1270),WNVC)
C
      COMMON/CPW/ DW1(13),NJC,NJS
C
      LARGE        BOU(3000)  ,FUS(25000),FAN(23700)
     1             ,DM1(996)   ,WPT(60)    ,BS(650)
     2             ,EP(30)     ,XTLP(30)   ,XOCP(30)
     3             ,DM2(21634),PYL(6061)  ,CEL(3700)
     4             ,XLP(19)    ,CHD(19)    ,ZI(24)
     5             ,XI(24)     ,XO(48)     ,ZO(48)     ,Z(48,19)
     6             ,WEL(20)
C
C                  SET FOR CODIM AND CURVE FIT FOR DEFLECTIONS (Z)
      NXW=DA(705)
      NEW=DA(730)
C                  SHIFT TO LCM FOR WINGD
      DO 10  I=1,NEW
   10 WEL(I)=WET(I)
      CALL WINGD (NEW,WEL,XLP,CHD,WPT)
      LD=0
      DXI=.5*(1.-COS(1.5708/WNVC))
      DO 100  IE=1,NEW
      DO 20  IX=1,NXW
      LD=LD+1
      ZI(IX)=WAD(LD)*CHD(IE)
   20 XI(IX)=WXC(IX)
      IC=0
      DO 40  IJC=1,NJC
      IC=IC+2
      XO(IC-1)=XOCP(IJC)-DXI
   40 XO(IC)=XOCP(IJC)+DXI
```

```
      CALL CODIM (XI,ZI,NXW,XO,ZO,IC)
      DO 60 IX=1,IC
   60 Z(IX,IE)=ZO(IX)
  100 CONTINUE
C                LINEAR INTERPOLATE FOR Z SPANWISE AND CALC SLOPE
      KK=0
      DXT=DXI+DXI
      DO 200 IE=1,NJS
      DX=DXT*XTLP(IE)
      IF(EP(IE).GT.WET(1)) GO TO 120
      II=1
      GO TO 140
  120 DO 130 IS=2,NEW
      II=IS
      IF(EP(IE)-WET(IS)) 150,140,130
  130 CONTINUE
  140 RATY=0.0
      GO TO 160
  150 RATY=(WET(II)-EP(IE))/(WET(II)-WET(II-1))
  160 TWB=TWI(II)-RATY*(TWI(II)-TWI(II-1))
      IC=0
      DO 170 IJC=1,NJC
      IC=IC+1
      Z1=Z(IC,II)-RATY*(Z(IC,II)-Z(IC,II-1))
      IC=IC+1
      Z2=Z(IC,II)-RATY*(Z(IC,II)-Z(IC,II-1))
      KK=KK+1
  170 BS(KK)=(Z1-Z2)/DX+TWB
  200 CONTINUE
      RETURN
      END
*DECK PCAS
      SUBROUTINE PCAS
C
C     SUBR. TLU TABLE LOOKUP FOR THE LOCAL ANGLE OF ATTACK AT SPECIFIED
C     ETA AND X/C.
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION   BI(361)    ,PXC(19)    ,PET(19)    ,PJC(18)    ,TWI(19)
C
      EQUIVALENCE       (DA(2531),PXC)   ,(DA(2551),PET)  ,(DA(2512),PJC)
     1                 ,(DA(2940),TWI)   ,(DA(2507),PNVC),(DA(2570),BI)
```

```
C
      COMMON/CPP/ DP1(8),NVS,NJC
C
      LARGE       DM1(75130),EV(22)     ,DM2(1483) ,CFJ(20)
     1            ,DM3(3126) ,BS(360)
C
C                 START
      NXW=DA(2530)
      NEW=DA(2550)
      IB=0
      DO 620 IE=1,NVS
      IF(EV(IE)-PET(1)) 510,510,515
  510 J=1
      GO TO 530
  515 DO 520 NC=2,NEW
      J=NC
      IF(EV(IE)-PET(J)) 540,530,520
  520 CONTINUE
  530 RATY=0.0
      GO TO 550
  540 RATY=(PET(J)-EV(IE))/(PET(J)-PET(J-1))
  550 TWB=TWI(J)-RATY*(TWI(J)-TWI(J-1))
      DO 610 IJC=1,NJC
      IJ=PJC(IJC)+1.0
      XC=CFJ(IJ)
      IF(XC-PXC(1)) 560,560,565
  560 I=1
      GO TO 580
  565 DO 570 NC=2,NXW
      I=NC
      IF(XC-PXC(I)) 590,580,570
  570 CONTINUE
  580 RATX=0.0
      GO TO 600
  590 RATX=(PXC(I)-XC)/(PXC(I)-PXC(I-1))
  600 NPX=I+NXW*(J-1)
      AX=BI(NPX)-RATX*(BI(NPX)-BI(NPX-1))
      NPX=NPX-NXW
      BX=BI(NPX)-RATX*(BI(NPX)-BI(NPX-1))
      IB=IB+1
  610 BS(IB)=AX-RATY*(AX-BX)+TWB
  620 CONTINUE
      RETURN
```

```
      END
*DECK PCZCS
      SUBROUTINE PCZCS
C
C*                 CALC. LOCAL ANGLES OF ATTACK FROM DEFLECTION INPUT(Z/C)
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION   PXC(19)    ,PET(19)    ,PAD(361)  ,TWI(19)    ,PJC(18)
C
      EQUIVALENCE       (DA(2531),PXC) ,(DA(2551),PET)  ,(DA(2570),PAD)
     1                  ,(DA(2940),TWI) ,(DA(2507),PNVC) ,(DA(2512),PJC)
C
      COMMON/CPP/ DP1(8),NVS,NJC
C
      LARGE       DM1(75130)
     1            ,EV(22)     ,XLV(22)    ,XTLV(22)   ,WPE(39)
     2            ,DP2(1400)  ,CFJ(20)    ,DP3(3126)  ,BS(360)
     3            ,DM2(4750)  ,XLP(19)    ,CHD(19)    ,ZI(24)
     4            ,XI(24)     ,XO(48)     ,ZO(48)     ,Z(48,19)
     5            ,PEL(20)
C
C                  SET FOR CODIM AND CURVE FIT FOR DEFLECTIONS
      LD=0
      NXW=DA(2530)
      NEW=DA(2550)
C                  SHIFT TO LCM FOR WINGD
      DO 10 I=1,NEW
   10 PEL(I)=PET(I)
C
      CALL WINGD (NEW,PEL,XLP,CHD,WPE)
      DXI=.5*(1.-COS(1.5708/PNVC))
      DO 100 IE=1,NEW
      DO 20 IX=1,NXW
      LD=LD+1
      ZI(IX)=PAD(LD)*CHD(IE)
   20 XI(IX)=PXC(IX)
      IC=0
      DO 40 IJC=1,NJC
      IC=IC+2
      IJ=PJC(IJC)+1.0
      XO(IC-1)=CFJ(IJ)-DXI
   40 XO(IC)=CFJ(IJ)+DXI
```

```
      CALL CODIM (XI,ZI,NXW,XO,ZO,IC)
      DO 60 IX=1,IC
   60 Z(IX,IE)=ZO(IX)
  100 CONTINUE
C                LINEAR INTERPOLATE FOR Z SPANWISE AND CALC SLOPE
      KK=0
      DXT=DXI+DXI
      DO 200 IE=1,NVS
      DX=DXT*XTLV(IE)
      IF(EV(IE).GT.PET(1)) GO TO 120
      II=1
      GO TO 140
  120 DO 130 IS=2,NEW
      II=IS
      IF(EV(IE)-PET(IS)) 150,140,130
  130 CONTINUE
  140 RATY=0.0
      GO TO 160
  150 RATY=(PET(II)-EV(IE))/(PET(II)-PET(II-1))
  160 TWB=TWI(II)-RATY*(TWI(II)-TWI(II-1))
      IC=0
      DO 170 IJC=1,NJC
      IC=IC+1
      Z1=Z(IC,II)-RATY*(Z(IC,II)-Z(IC,II-1))
      IC=IC+1
      Z2=Z(IC,II)-RATY*(Z(IC,II)-Z(IC,II-1))
      KK=KK+1
  170 BS(KK)=(Z1-Z2)/DX+TWB
  200 CONTINUE
      RETURN
      END
*DECK SOLU
      SUBROUTINE SOLU
C
C*               SOLVE LINEAR SIMULTANEOUS EQUATIONS (AX=B)
C
      COMMON/DAT/ DA(5000)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CFG/ IDB,IDF,IDW,IDP,IDN,NQB,NQF,NQW,NQP,NQN
     1           ,NKB,NKF,NKW,NKP,NKN
C
      LARGE       B(1000)    ,DM1(90812),A(1000)    ,AR(590)
```

```
C
C
      L=NKN
      M=NQB+NQF+NQW+NQP
      N=NKB+NKF+NKW+NKP
      NQT=M+L
      NKT=N+L
      IF(L.NE.0) GO TO 10
C*                 NO NACELLE
      CALL MSOL (N,M,12)
      GO TO 30
   10 IF(N.NE.0) GO TO 20
C*                 NACELLE ONLY
      CALL MSOL (L,L,10)
      GO TO 30
C*                 WITH NACELLE
   20 CALL PARTM (L,M,N)
C*                 PRINT LEAST SQUARE MATRIX-B
   30 IF(DA(12).EQ.0.0) GO TO 110
      L1=N+1
      DO 50 K=1,NQT
      A(K)=0.0
      IF(N.NE.0) READ (12) (AR(I),I=1,N)
      IF(L.NE.0) READ (10) (AR(I),I=L1,NKT)
      DO 50 I=1,NKT
   50 A(K)=A(K)+B(I)*AR(I)
C
      IF(L.NE.0) REWIND 10
      IF(N.NE.0) REWIND 12
C
      WRITE (6,100) (A(K),K=1,NQT)
  100 FORMAT (26H1 ** LEAST SQUARE MATRIX-B/(1H 10F10.5))
C
C*                 COMPRESS. EFFECT ON COEFFICIENTS
  110 IF(BETA.EQ.1.) GO TO 140
      DO 120 I=1,NKT
  120 B(I)=B(I)/BETA
C
  140 RETURN
      END
*DECK MSOL
      SUBROUTINE MSOL (NKX,NQX,MT)
C
```

```
C*                 SOLVES AX=B FOR X, GIVEN A AND B
C
      COMMON/DAT/ DA(5000)
      DIMENSION   WJC(1)     ,PJC(1)
C
      EQUIVALENCE      (DA(14),WSJC)    ,(DA(1277),WNJC),(DA(1660),WJC)
     1                ,(DA(2510),PSJC) ,(DA(2511),PNJC),(DA(2512),PJC)
C
      COMMON/CFG/ IDB,IDF,IDW,IDP,IDN,NQB,NQF,NQW,NQP,NQN
C
      LARGE       B(1000)     ,DM1(90812)
      LARGE       A(101975) ,AR(591)     ,IXC(450)   ,IL(450)
C
C
      NJC=WNJC
      NJCP=PNJC
      NKTP=NKX+1
      N=(NKX*(NKX+3))/2
      DO 10 I=1,N
   10 A(I)=0.0
      NKT2=NKX+2
      NK1=NQB+NQF+1
      NK2=NK1+NQW
      NK3=NK2+NQP
      JC=0
      JCP=0
      DO 60 K=1,NQX
      READ (MT) (AR(L),L=1,NKX)
      IF(IDW.EQ.0) GO TO 32
      IF(K.LT.NK1) GO TO 32
      IF(K.GE.NK2) GO TO 32
      IF(JC.GE.NJC) JC=0
      JC=JC+1
      IF(WSJC.GT.WJC(JC)) GO TO 60
      GO TO 36
   32 IF(IDP.EQ.0) GO TO 36
      IF(K.LT.NK2) GO TO 36
      IF(K.GE.NK3) GO TO 36
      IF(JCP.GE.NJCP) JCP=0
      JCP=JCP+1
      IF(PSJC.GT.PJC(JCP)) GO TO 60
   36 AR(NKX+1)=B(K)
      IXI=1
```

```
      DO 50 I=1,NKX
      R=SQRT(A(IXI)**2+AR(I)**2)
      IF(R.EQ.0.0) GO TO 50
      C=A(IXI)/R
      S=AR(I)/R
      IXJ=IXI
      DO 40 J=I,NKTP
      T2=C*A(IXJ)+S*AR(J)
      AR(J)=-S*A(IXJ)+C*AR(J)
      A(IXJ)=T2
   40 IXJ=IXJ+1
   50 IXI=IXI+NKT2-I
   60 CONTINUE
      REWIND MT
      II=1
      IXI=1
      DO 80 I=1,NKX
      IXC(I)=IXI
      B(I)=0.0
      IL(I)=0
      IF(A(IXI).LE.0.0000001) GO TO 80
      IL(I)=II
      II=II+1
   80 IXI=IXI+NKT2-I
      II=NKX
      DO 210 J=1,NKX
      IF(IL(II).LE.0) GO TO 210
      JI=IL(II)
      JS=IXC(JI)-JI
      JXI=JS+II
      JXN=JS+NKTP
      IF(II-NKX) 170,200,220
  170 IK=II+1
      JXK=JXI
      DO 180 K=IK,NKX
      JXK=JXK+1
  180 B(II)=B(II)-A(JXK)*B(K)
  200 B(II)=(B(II)+A(JXN))/A(JXI)
  210 II=II-1
  220 CONTINUE
      RETURN
      END
*DECK MSOLP
```

```
      SUBROUTINE MSOLP(NKT,NQT,NCB,M)
C
C*                 HOUSEHOLDER METHOD FOR SOLVING SET OF LINEAR
C*                 SIMULTANEOUS EQUATIONS
C
      LARGE       B(1000)    ,DM1(90812)
      LARGE       A(101975) ,AR(591)    ,KRR(450)   ,IL(450)
C
C
      N1=NCB-1
      KR=0
      NKTP=NKT+NCB
      DO 10 J=1,NKT
      KRR(J)=KR+J-1
      KR=KRR(J)
      DO 10 K=J,NKTP
      LL=(J-1)*NKTP+K-KRR(J)
 10   A(LL)=0.0
      DO 60 K=1,NQT
      READ (10) (AR(I),I=1,NKT)
      KR=NKT+1
      LL=NKT+N1
      READ (12) (AR(I),I=KR,LL)
      AR(NKT+NCB)=B(M+K)
C
   30 DO 50 I=1,NKT
      LL=(I-1)*NKTP+I-KRR(I)
      R=SQRT(A(LL)**2+AR(I)**2)
      IF(R .EQ. 0.0) GO TO 50
      C=A(LL)/R
      S = AR(I) / R
      DO 40 J=I,NKTP
      LL=(I-1)*NKTP+J-KRR(I)
      T2=C*A(LL)+S*AR(J)
      AR(J)=-S*A(LL)+C*AR(J)
 40   A(LL)=T2
   50 CONTINUE
   60 CONTINUE
      REWIND 10
      REWIND 12
      II = 1
      DO 80 I=1,NKT
      LL=(I-1)*NKTP+I-KRR(I)
```

```
      IF(A(LL)-0.0000001)70,92,92
   92 CONTINUE
      IL(I) = II
      II = II + 1
      GO TO 80
   70 IL(I) = 0
   80 CONTINUE
      DO 250 J=1,NCB
      NKTJ=NKT+J
      DO 90 I=1,NKT
   90 AR(I)=0.0
      II=NKT
      DO 210 I=1,NKT
      IF(IL(II) .LE. 0) GO TO 210
      JI = IL(II)
      LX=(JI-1)*NKTP+NKTJ-KRR(JI)
      LM=(JI-1)*NKTP+II-KRR(JI)
      IF(II - NKT) 170, 200, 220
  170 IK = II + 1
      DO 180 K=IK,NKT
      LL=(JI-1)*NKTP+K-KRR(JI)
  180 AR(II)=AR(II)-A(LL)*AR(K)
  200 AR(II)=(AR(II)+A(LX))/A(LM)
  210 II = II - 1
  220 CONTINUE
      WRITE (16) (AR(I),I=1,NKT)
  250 CONTINUE
      REWIND 16
      RETURN
      END
*DECK PARTM
      SUBROUTINE PARTM(L,M,N)
C
C*                  PARTISION-HOUSEHOLDER MATRIX SOLUTION
C
      LARGE       B(1000)     ,DM1(90812)
      LARGE       A(38835),SAVE(140,451),AR(591)    ,A11(450)   ,IL(450)
C   SIZE OF SAVE=(L,N+1)
C
      NCB=N+1
      DO 10 I=1,M
      READ (10)
   10 READ (12)
```

```
      CALL MSOLP (L,L,NCB,M)
      DO 20 I=1,NCB
   20 READ (16) (SAVE(J,I),J=1,L)
      REWIND 16
      DO 70 I=1,M
      READ (10) (A11(LL),LL=1,L)
      DO 50 J=1,NCB
      AR(J)=0.0
      DO 40 K=1,L
   40 AR(J)=AR(J)+A11(K)*SAVE(K,J)
   50 CONTINUE
      READ (12) (A11(LL),LL=1,N)
      DO 60 J=1,N
   60 A11(J)=AR(J)-A11(J)
      WRITE (11) (A11(J),J=1,N)
   70 B(I)=AR(N+1)-B(I)
      REWIND 10
      REWIND 11
      REWIND 12
      CALL MSOL (N,M,11)
      DO 80 I=1,NCB
   80 READ (16) (SAVE(J,I),J=1,L)
      REWIND 16
      DO 100 K=1,L
      KK=N+K
      TEMP=0.0
      DO 90 J=1,N
   90 TEMP=TEMP+B(J)*SAVE(K,J)
  100 B(KK)=SAVE(K,NCB)-TEMP
      RETURN
      END
*DECK PLCAL
      SUBROUTINE PLCAL
C
C*                   CALC PRESSURE COEFFICIENTS AND INTEGRATED LOADS
C
      COMMON/CFG/ IDB,IDF,IDW,IDP
C
C                    INITIAL
      BCL=0.0
      BCD=0.0
      BCM=0.0
      FCL=0.0
```

```
      FCD=0.0
      FCM=0.0
      WCL=0.0
      WCD=0.0
      WCM=0.0
      PCL=0.0
      PCD=0.0
      PCM=0.0
C                FUSELAGE
      IF(IDB.EQ.0) GO TO 10
      CALL BVRCP
      CALL BLOAD (BCL,BCD,BCM)
C                FANPOD
   10 IF(IDF.EQ.0) GO TO 20
      CALL FVRCP
      CALL FLOAD (FCL,FCD,FCM)
C                WING
   20 IF(IDW.EQ.0) GO TO 30
      CALL WDPQ
      CALL WVAP
      CALL WLOAD (WCL,WCD,WCM)
C                PYLON
   30 IF(IDP.EQ.0) GO TO 40
      CALL PDPQ
      CALL PVAP
      CALL PLOAD (PCL,PCD,PCM)
C
C                SUM AND PRINT
   40 TCL=BCL+FCL+WCL+PCL
      TCD=BCD+FCD+WCD+PCD
      TCM=BCM+FCM+WCM+PCM
      TXC=-TCM/TCL
C
      WRITE (6,50) TCL,TCD,TCM,TXC
   50 FORMAT (1H1,20X,36H ** TOTAL CONFIGURATION PARAMETERS */1H0,40X,
     1      ,2HCL,13X,2HCD,13X,2HCM,13X,3HX/C/1H ,30X,4F15.5)
C
      IF(IDW.EQ.0) GO TO 70
      CALL IDRAG (BCL,FCL,CDI)
      WRITE (6,60) CDI
   60 FORMAT (1H0,20X,18H ** INDUCED DRAG =,1F10.5)
C                REWIND I/O
   70 IF(IDB.NE.0) REWIND 19
```

```
      IF(IDF.NE.0) REWIND 20
      IF(IDW.NE.0) REWIND 21
      IF(IDP.NE.0) REWIND 22
C
      RETURN
      END
*DECK BVRCP
      SUBROUTINE BVRCP
C
C*                 CALC. VELOCITY RATIO AND PRESSURE COEFFICIENT AT
C*                 CONTROL POINTS ON THE FUSELAGE AND PRINT
C
      COMMON/DAT/ DA(5000)
      DIMENSION   FJX(1),FVX(1)
C
      EQUIVALENCE      (DA(2),WSPAN)    ,(DA(4),XMACH)   ,(DA(1207),WITB)
     1                ,(DA(4015),FOX)   ,(DA(4735),FVX)  ,(DA(4905),FJX)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CFG/ IDB,IDF,IDW,IDP,IDN,NOB,NOF,NOW,NOP,NON
     1           ,NKB,NKF,NKW,NKP,NKN
      COMMON/CPB/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX,IFBE,SINA,COSA
      COMMON/CPW/ DW1(7),NE5,NTB,NU
      COMMON/CFK/ NDA,NUT,THEF,SINTF,SINTF2,COSTF,THEK,SINTK,SINTK2
C
      LARGE       CKV(1000)  ,DM1(2000)
     *           ,XQ(650)    ,YQ(650)    ,ZQ(650)
     1           ,TMX(650)   ,TMY(650)   ,TMZ(650)
     2           ,TTX(650)   ,TTY(650)   ,TTZ(650)
     3           ,XN(650)    ,YN(650)    ,ZN(650)
     4           ,PM(150,25),DB1(13450),FAN(23700),EV(52)     ,DM2(944)
     5           ,WPI(60)    ,DM3(22374),PYL(6061) ,CEL(3700)
     6           ,XF(51,10)  ,DM4(561)
     7           ,AX(250)    ,AY(250)    ,AZ(250)
     8           ,CK(650)    ,CP(650)    ,XL(650)    ,VMV(650)   ,VTV(650)
     9           ,ETA(2)     ,PLM(51,25)
C
C*                START
C*                CALC. XKV IF WING-BODY
      IF(IDW.EQ.0) GO TO 30
      WP2=2.*WSPAN*PI*BETA
      I=NE5+3
      XF(I,NUT+1)=0.0
```

```
      CALL SFC (I,EV)
      XKV=WP2*(XF(I,1)-XF(I,2)/2.+XF(I,NUT+1)*SINTF+XF(I,NU)*SINTK)
C*                CALC CONTRIBUTION TO FLOW FROM ADJACENT VORTICES
   30 NC=0
      DO 36 IY=1,NVY
      DO 36 IJ=1,NJX
      J=FJX(IJ)
      K=J+1
      IF(J.EQ.NVXM) K=J
      NC=NC+1
      KC=(IY-1)*NF
      VMV(NC)=0.0
      DO 35 I=1,NF
      KC=KC+1
   35 VMV(NC)=CKV(KC)*(PM(J,I)+PM(K,I))+VMV(NC)
   36 VMV(NC)=VMV(NC)/4.
      IX=1
      DO 40 IJ=1,NJX
      JX=FJX(IJ)
   37 IX=IX+1
      IF(IX.GE.JX) GO TO 38
      READ  (23)
      GO TO 37
   38 READ  (23) ((PLM(J,I),J=1,NVY ),I=1,NF)
      NC=IJ
      KC=0
      DO 40 J=1,NVY
      CK(NC)=0.0
      DO 39 I=1,NF
      KC=KC+1
   39 CK(NC)=CKV(KC)*PLM(J,I)+CK(NC)
   40 NC=NC+NJX
      REWIND 23
      DO 50 I=1,NJX
   50 XL(I)=YQ(I)+YQ(I)
      N1=NJX
      DO 53 J=2,NVY
      DO 53 I=1,NJX
      N1=N1+1
      N2=N1-NJX
   53 XL(N1)=SQRT((XQ(N2)-XQ(N1))**2+(YQ(N2)-YQ(N1))**2+(ZQ(N2)-ZQ(N1))
     1      **2)
      NC=0
```

```
      ITW=WITB
      WLE=WPT(2)
      WTE=WPT(3)
      DO 57 J=1,NVY
   55 DO 57 I=1,NJX
      NC=NC+1
      N1=NC-NJX
      IF(J.EQ.1) N1=NC
      N2=NC+NJX
      IF(J.EQ.NVY) N2=NC
      C1=.25
      C2=.25
      XK1=0.
      XK2=0.
      IF(IDW.EQ.0) GO TO 57
      JX=FJX(I)
      XX=FVX(JX)+FOX
      IF(XX.LT.WLE) GO TO 57
      JW=J-ITW
      IF(XX.GE.WTE) GO TO 56
      IF(JW.EQ.-1) C2=0.0
      IF(JW.EQ.0) C1=0.0
      GO TO 57
   56 IF(JW.EQ.-1) XK2=XKV
      IF(JW.EQ.0) XK1=XKV
   57 VTV(NC)=C1*(CK(NC)-CK(N1)+XK1)/XL(NC)+C2*(CK(N2)-CK(NC)+XK2)
     1        /XL(N2)
C
C*                   CALC. VELOCITY RATIO AND CP
   58 DO 60 J=1,NQB
      READ(19) (AX(IS),AY(IS),AZ(IS),IS=1,NKB)
      VTV(J)=VTV(J)+(TTX(J)*COSA*BETA+TTZ(J)*SINA)
      VMV(J)=VMV(J)+(TMX(J)*COSA*BETA+TMZ(J)*SINA)
      DO 60 I=1,NKB
      VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(I)+VTV(J)
   60 VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(I)+VMV(J)
      IF(IDF.EQ.0) GO TO 62
      DO 61 J=1,NQB
      READ(20) (AX(IS),AY(IS),AZ(IS),IS=1,NKF)
      II=NKB
      DO 61 I=1,NKF
      II=II+1
      VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(II)+VTV(J)
```

```
   61 VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(II)+VMV(J)
C
   62 IF(IDW.EQ.0) GO TO 65
      DO 64 J=1,NQB
      READ(21) (AX(IS),AY(IS),AZ(IS),IS=1,NKW),UV,VV,WV
      IF(DA(2000).EQ.0.0) GO TO 63
      UB=UV*BETA
      VTV(J)=TTX(J)*UB+TTY(J)*VV+TTZ(J)*WV+VTV(J)
      VMV(J)=TMX(J)*UB+TMY(J)*VV+TMZ(J)*WV+VMV(J)
   63 II=NKB+NKF
      DO 64 I=1,NKW
      II=II+1
      VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(II)+VTV(J)
   64 VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(II)+VMV(J)
   65 IF(IDP.EQ.0) GO TO 72
      DO 70 J=1,NQB
      READ (22) (AX(IS),AY(IS),AZ(IS),IS=1,NKP),UV,VV,WV
      IF(DA(3000).EQ.0.0) GO TO 66
      UB=UV*BETA
      VTV(J)=TTX(J)*UB+TTY(J)*VV+TTZ(J)*WV+VTV(J)
      VMV(J)=TMX(J)*UB+TMY(J)*VV+TMZ(J)*WV+VMV(J)
   66 II=NKB+NKF+NKW
      DO 70 I=1,NKP
      II=II+1
      VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J) AZ(I))*CKV(II)+VTV(J)
   70 VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J) AZ(I))*CKV(II)+VMV(J)
   72 IF(IDN.EQ.0) GO TO 80
      DO 74 J=1,NQB
      READ (9) (AX(IS),AY(IS),AZ(IS),IS=1,NKN)
      II=NKB+NKF+NKW+NKP
      DO 74 I=1,NKN
      II=II+1
      VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(II)+VTV(J)
   74 VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(II)+VMV(J)
C
   80 DO 89 J=1,NQB
      IF(XMACH.LE.0.0) GO TO 88
      VTV(J)=VTV(J)/SQRT((TTX(J)*BETA)**2+TTY(J)**2+TTZ(J)**2)
      VMV(J)=VMV(J)/SQRT((TMX(J)*BETA)**2+TMY(J)**2+TMZ(J)**2)
      XMS=XMACH**2
      XM1=1.42857/XMS
      CMN=1.+0.2*XMS*(1.-VTV(J)**2-VMV(J)**2)
      IF(CMN.GT.0.0) GO TO 87
```

```
      CP(J)=-XM1
      GO TO 89
   87 CP(J)=XM1*(CMN**3.5-1.)
      GO TO 89
   88 CP(J)=1.-VTV(J)**2-VMV(J)**2
   89 CONTINUE
C*                 PRINT OUTPUT
      NC=0
      DO 110 J=1,NVY
      WRITE (6,90)
   90 FORMAT (1H1,20X,65H **  SUBSONIC INTERFERENCE PRESSURE PROGRAM **
     1FUSELAGE OUTPUT **/6H0  NO.,7X,2HXQ,10X,2HYQ,10X,2HZQ,8X,6HVT/VFS
     2,6X,6HVM/VFS,8X,2HCP/)
      DO 110 I=1,NJX
      NC=NC+1
      ZP=ZQ(NC)/BETA
      YP=YQ(NC)/BETA
      WRITE (6,100) I,XQ(NC),YP,ZP,VTV(NC),VMV(NC),CP(NC)
  100 FORMAT (1H ,I5,6F12.5)
  110 CONTINUE
      RETURN
      END
*DECK BLOAD
      SUBROUTINE BLOAD (FCL,CDF,FCM)
C
C*                 CALCULATE FUSELAGE LOCAL INTEGRATED LOADS
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION  FVX(1),FJX(1)
      EQUIVALENCE      (DA(4005),FCD)   ,(DA(4735),FVX)  ,(DA(4905),FJX)
     1,(DA(1),AR)     ,(DA(2),WSPAN)    ,(DA(3),XMAC)    ,(DA(4006),FAA)
     2,(DA(7),XCG)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CFG/ IDB,IDF,IDW,IDP,IDN
      COMMON/CPB/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX,IFBE,SINA,COSA,FAREA
C
      LARGE          BOU(3000) ,XQ(650)    ,YQ(650)    ,ZQ(650)
     1               ,DM1(9750) ,AXY(650)   ,AYZ(650)   ,DM2(68891),DM3(2471)
     2               ,CP(650)   ,FDX(150)   ,CLW(150)   ,CDW(150)   ,CMX(150)
     3               ,FX(71)    ,CLWA(71)   ,CMXA(71)   ,CMZ(150)   ,CMZA(71)
     4               ,CDWA(71)
```

```
C                    START
C*                   CALC C(L)*W/W(AVG) AT DOWNWASH X STATION
      IY=1
      DO 40 I=1,NJX
      JX=FJX(I)
      DX=FVX(JX+1)-FVX(JX)
      FDX(I)=(FVX(JX)+FVX(JX+1))/2.
      IX=I
      CLW(I)=0.0
      CMX(I)=0.0
      CMZ(I)=0.0
      CDW(I)=0.0
      DO 30 J=1,NVY
      CLW(I)=CP(IX)*(AXY(IY)*COSA +AYZ(IY)*SINA )+CLW(I)
      CMX(I)=CP(IX)*(XQ(IX)-XCG)*AXY(IY)+CMX(I)
      CMZ(I)=CP(IX)*ZQ(IX)*AYZ(IY)+CMZ(I)
      CDW(I)=CP(IX)*(AXY(IY)*SINA -AYZ(IY)*COSA )+CDW(I)
      IX=IX+NJX
   30 IY=IY+1
      CAX=FAREA*DX
      CSX=FCD/CAX
      CLW(I)=CLW(I)*CSX
      CMX(I)=-CMX(I)/CAX
      CMZ(I)=-CMZ(I)/CAX
   40 CDW(I)=CDW(I)*CSX
C     *              CALC C*W/W(AVG) AT EQUAL INTERVAL OF PHI
      DX=PI/70.
      FX(1)=0.0
      SPHI=0.0
      FCD2=FCD/2.
      DO 50 I=2,70
      SPHI=SPHI+DX
   50 FX(I)=FCD2*(1.-COS(SPHI))
      CALL CODIM (FDX,CLW,NJX,FX,CLWA,70)
      CALL CODIM (FDX,CMX,NJX,FX,CMXA,70)
      CALL CODIM (FDX,CMZ,NJX,FX,CMZA,70)
      CALL CODIM (FDX,CDW,NJX,FX,CDWA,70)
      SPHI=0.0
      DO 52 I=2,70
      FX(I) = FX(I) / FCD
      SPHI=SPHI+DX
      HSP=SIN(SPHI)/2.
      CLW(I)=HSP*CLWA(I)
```

```
      CMX(I)=HSP*CMXA(I)
      CMZ(I)=HSP*CMZA(I)
   52 CDW(I)=HSP*CDWA(I)
C*                  SIMPSON RULE INTEGRATION
      SL1=CLW(70)
      SL2=0.0
      XM1=CMX(70)
      XM2=0.0
      ZM1=CMZ(70)
      ZM2=0.0
      SD1=CDW(70)
      SD2=0.0
      DO 60 I=2,69,2
      SL1= CLW(I)+SL1
      SL2= CLW(I+1)+SL2
      XM1= CMX(I)+XM1
      XM2= CMX(I+1)+XM2
      ZM1= CMZ(I)+ZM1
      ZM2= CMZ(I+1)+ZM2
      SD1= CDW(I)+SD1
   60 SD2= CDW(I+1)+SD2
      DX3=DX/3.
      IDT=IDF+IDW+IDP+IDN
      IF(IDT.NE.0) GO TO 70
      SR=1.
      CR=1.
      GO TO 80
   70 SR=FAREA*AR/WSPAN**2
      CR=FCD/XMAC
   80 FCMXY=DX3*(4.*XM1+2.*XM2)*SR*CR
      FCL=DX3*(4.*SL1+2.*SL2)*SR
      FCMYZ=DX3*(4.*ZM1+2.*ZM2)*SR*CR
      CDF=DX3*(4.*SD1+2.*SD2)*SR
      FCM=FCMXY+FCMYZ
C*                  PRINT OUTPUT
      WRITE (6,100) FAREA,FCL,CDF,FCMXY,FCMYZ,FCM,(FX(I),CLWA(I),CDWA(I)
     1              ,FX(I+35),CLWA(I+35),CDWA(I+35),I=2,35)
     2              ,FX(36),CLWA(36),CDWA(36)
  100 FORMAT (1H1,10X,45H ** FUSELAGE LOADS      ** REF. FUSELAGE AREA=,
     1        F14.5/1H0,10X,   5H* CL=F10.5,5X,  5H* CD=F10.5,3X,
     2        7H* CMXY=F11.5,3X,7H* CMYZ=F11.5,5X, 5H* CM=F11.5/
     3        1H0,12X,1HX,15X,6HCLW/WA,11X,6HCDW/WA,12X,1HX,15X,6HCLW/WA
     4       ,11X,6HCDW/WA/ 1H0,6F17.5 /(1H ,6F17.5))
```

```
      RETURN
      END
*DECK FVRCP
      SUBROUTINE FVRCP
C
C*                 CALC. VELOCITY RATIO AND PRESSURE COEFFICIENT AT
C*                 CONTROL POINTS ON THE FANPOD
C
      COMMON/DAT/ DA(5000)
      DIMENSION   FJX(1),FVX(1)
      EQUIVALENCE       (DA(4),XMACH)     ,(DA(11),FWCI)    ,(DA(1720),FJX)
     1                  ,(DA(2),WSPAN)    ,(DA(1281),WNW)  ,(DA(1460),FVX)
     2                  ,(DA(87),FOX)     ,(DA(1203),WITO),(DA(1204),WITI)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CFG/ IDB,IDF,IDW,IDP,IDN,NQB,NQF,NQW,NQP,NQN
     1           ,NKB,NKF,NKW,NKP,NKN
      COMMON/CFK/ NDA,NUT,THEF,SINTF,SINTF2,COSTF,THEK,SINTK,SINTK2
      COMMON/CPF/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX,IFBE,SINA,COSA
      COMMON/CPW/ DWI(6),NE3,NE5,NTB,NU,DW2(5),NVSO,NVSI,IWX
C
      LARGE       CKV(1000) ,DM1(2000) ,FUS(25000)
     *            ,XQ(650)  ,YQ(650)   ,ZQ(650)
     1            ,TMX(650) ,TMY(650)  ,TMZ(650)
     2            ,TTX(650) ,TTY(650)  ,TTZ(650)
     3            ,XN(650)  ,YN(650)   ,ZN(650)
     4            ,PM(100,25),DFI(13400),EV(52)     ,DM2(944)
     5            ,WPT(60)  ,DM3(22374),PYL(6061)  ,CEL(3700)
     6            ,XF(51,10) ,DM4(561)
     7            ,AX(250)  ,AY(250)   ,AZ(250)
     8            ,CK(650)  ,CP(650)   ,XL(650)    ,VMV(650)   ,VTV(650)
     9            ,ETA(2)   ,PLM(51,25)
C
C*                 START
C*                 CALC. XKV IF WING-BODY
      IF(IDW.EQ.0) GO TO 30
      WP2=2.*WSPAN*PI*BETA
      J=NE3+3
      I=J+1
      XF(I,NUT+1)=0.0
      XF(J,NUT+1)=0.0
      CALL SFC (I,EV)
      XKI=WP2*(XF(I,1)-XF(I,2)/2.+XF(I,NUT+1)*SINTF+XF(I,NU)*SINTK)
```

```
      XKO=WP2*(XF(J,1)-XF(J,2)/2.+XF(J,NUT+1)*SINTF+XF(J,NU)*SINTK)
C*                CALC CONTRIBUTION TO FLOW FROM ADJACENT VORTICES
   30 NC=0
      DO 36 IY=1,NVY
      DO 36 IJ=1,NJX
      J=FJX(IJ)
      K=J+1
      IF(J.EQ.NVXM) K=J
      NC=NC+1
      KC=(IY-1)*NF+NKB
      VMV(NC)=0.0
      DO 35 I=1,NF
      KC=KC+1
   35 VMV(NC)=CKV(KC)*(PM(J,I)+PM(K,I))+VMV(NC)
   36 VMV(NC)=VMV(NC)/4.
      IX=1
      DO 40 IJ=1,NJX
      JX=FJX(IJ)
   37 IX=IX+1
      IF(IX.GE.JX) GO TO 38
      READ (24)
      GO TO 37
   38 READ  (24) ((PLM(J,I),J=1,NVY ),I=1,NF)
      NC=IJ
      KC=NKB
      DO 40 J=1,NVY
      CK(NC)=0.0
      DO 39 I=1,NF
      KC=KC+1
   39 CK(NC)=CKV(KC)*PLM(J,I)+CK(NC)
   40 NC=NC+NJX
      REWIND 24
   49 NC=0
      N1=0
      NCC=NJX*(NVY-1)
      DO 53 J=1,NVY
      DO 53 I=1,NJX
      N1=N1+1
      N2=N1-NJX
      IF(J.EQ.1) N2=N1+NCC
   53 XL(N1)=SQRT((XQ(N2)-XQ(N1))**2+(YQ(N2)-YQ(N1))**2+(ZQ(N2)-ZQ(N1))
     1     **2)
      NVYH=NVY/2+1
```

```
      DO 57 J=1,NVY
      IF(IDW.EQ.0) GO TO 55
      IF(J.NE.1) GO TO 54
      ITW=WITO
      WLE=WPT(IWX+4)
      WTE=WPT(IWX+5)
      XKV=XKO
      GO TO 55
   54 IF(J.NE.NVYH) GO TO 55
      ITW=WITI
      WLE=WPT(IWX+1)
      WTE=WPT(IWX+2)
      XKV=-XKI
   55 DO 57 I=1,NJX
      NC=NC+1
      N1=NC-NJX
      IF(J.EQ.1) N1=NC+NCC
      N2=NC+NJX
      IF(J.EQ.NVY) N2=I
      C1=.25
      C2=.25
      XK1=0.
      XK2=0.
      IF(IDW.EQ.0) GO TO 57
      JX=FJX(I)
      XX=FVX(JX)+FOX
      IF(XX.LT.WLE) GO TO 57
      JW=J-ITW
      IF(XX.GE.WTE) GO TO 56
      IF(JW.EQ.-1) C2=0.0
      IF(JW.EQ.0) C1=0.0
      GO TO 57
   56 IF(JW.EQ.-1) XK2=XKV
      IF(JW.EQ.0) XK1=XKV
   57 VTV(NC)=C1*(CK(NC)-CK(N1)+XK1)/XL(NC)+C2*(CK(N2)-CK(NC)+XK2)
     1       /XL(N2)
C
C*                 CALC. VELOCITY RATIO AND CP
   58 DO 60 J=1,NQF
      READ(20) (AX(IS),AY(IS),AZ(IS),IS=1,NKF)
      VTV(J)=VTV(J)+(TTX(J)*COSA*BETA+TTZ(J)*SINA)
      VMV(J)=VMV(J)+(TMX(J)*COSA*BETA+TMZ(J)*SINA)
      N=NKB
```

```
      DO 60 I=1,NKF
      N=N+1
      VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(N)+VTV(J)
   60 VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(N)+VMV(J)
      IF(IDB.EQ.0) GO TO 62
      DO 61 J=1,NQF
      READ(19) (AX(IS),AY(IS),AZ(IS),IS=1,NKB)
      DO 61 I=1,NKB
      VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(I)+VTV(J)
   61 VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(I)+VMV(J)
C
   62 IF(IDW.EQ.0) GO TO 65
      DO 64 J=1,NQF
      READ(21) (AX(IS),AY(IS),AZ(IS),IS=1,NKW),UV,VV,WV
      IF(DA(2000).EQ.0.0) GO TO 63
      UB=UV*BETA
      VTV(J)=TTX(J)*UB+TTY(J)*VV+TTZ(J)*WV+VTV(J)
      VMV(J)=TMX(J)*UB+TMY(J)*VV+TMZ(J)*WV+VMV(J)
   63 II=NKB+NKF
      DO 64 I=1,NKW
      II=II+1
      VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(II)+VTV(J)
   64 VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(II)+VMV(J)
   65 IF(IDP.EQ.0) GO TO 72
      DO 70 J=1,NQF
      READ (22) (AX(IS),AY(IS),AZ(IS),IS=1,NKP),UV,VV,WV
      IF(DA(3000).EQ.0.0) GO TO 66
      UB=UV*BETA
      VTV(J)=TTX(J)*UB+TTY(J)*VV+TTZ(J)*WV+VTV(J)
      VMV(J)=TMX(J)*UB+TMY(J)*VV+TMZ(J)*WV+VMV(J)
   66 II=NKB+NKF+NKW
      DO 70 I=1,NKP
      II=II+1
      VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(II)+VTV(J)
   70 VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(II)+VMV(J)
   72 IF(IDN.EQ.0) GO TO 80
      DO 74 J=1,NQF
      READ (9) (AX(IS),AY(IS),AZ(IS),IS=1,NKN)
      II=NKB+NKF+NKW+NKP
      DO 74 I=1,NKN
      II=II+1
      VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(II)+VTV(J)
   74 VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(II)+VMV(J)
```

```
C
   80 DO 89 J=1,NQF
      IF(XMACH.LE.0.0) GO TO 88
      VTV(J)=VTV(J)/SQRT((TTX(J)*BETA)**2+TTY(J)**2+TTZ(J)**2)
      VMV(J)=VMV(J)/SQRT((TMX(J)*BETA)**2+TMY(J)**2+TMZ(J)**2)
      XMS=XMACH**2
      XM1=1.42857/XMS
      CMN=1.+0.2*XMS*(1.-VTV(J)**2-VMV(J)**2)
      IF(CMN.GT.0.0) GO TO 87
      CP(J)=-XM1
      GO TO 89
   87 CP(J)=XM1*(CMN**3.5-1.)
      GO TO 89
   88 CP(J)=1.-VTV(J)**2-VMV(J)**2
   89 CONTINUE
C*                PRINT OUTPUT
      NC=0
      DO 110 J=1,NVY
      WRITE (6,90)
   90 FORMAT (1H1,20X,65H **  SUBSONIC INTERFERENCE PRESSURE PROGRAM **
     1 FAN-POD OUTPUT **/6H0  NO.,7X,2HXQ,10X,2HYQ,10X,2HZQ,8X,6HVT/VFS
     2,6X,6HVM/VFS,8X,2HCP/)
      DO 110 I=1,NJX
      NC=NC+1
      ZP=ZQ(NC)/BETA
      YP=YQ(NC)/BETA
      WRITE (6,100) I,XQ(NC),YP,ZP,VTV(NC),VMV(NC),CP(NC)
  100 FORMAT (1H ,I5,6F12.5)
  110 CONTINUE
      RETURN
      END
*DECK FLOAD
      SUBROUTINE  FLOAD (FCL,CDF,FCM)
C
C*                CALCULATE FANPOD LOCAL INTEGRATED LOADS
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION  FVX(1),FJX(1)
C
      EQUIVALENCE      (DA(5),FCD)      ,(DA(1460),FVX)  ,(DA(1720),FJX)
     1,(DA(1),AR)      ,(DA(2),WSPAN)   ,(DA(3),XMAC)    ,(DA(6),FAA)
     2,(DA(7),XCG)
```

```
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CFG/ IDB,IDF,IDW,IDP,IDN
      COMMON/CPF/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX,IFBE,SINA,COSA,FAREA
C
      LARGE       BOU(3000) ,FUS(25000),XQ(650)   ,YQ(650)   ,ZQ(650)
     1           ,DM1(8450) ,AXY(650)  ,AYZ(650)  ,DM2(45191),DM3(2471)
     2           ,CP(650)   ,FDX(150)  ,CLW(150)  ,CDW(150)  ,CMX(150)
     3           ,FX(71)    ,CLWA(71)  ,CMXA(71)  ,CMZ(150)  ,CMZA(71)
     4           ,CDWA(71)
C                START
C*               CALC C(L)*W/W(AVG) AT DOWNWASH X STATION
      IY=1
      DO 40 I=1,NJX
      JX=FJX(I)
      DX=FVX(JX+1)-FVX(JX)
      FDX(I)=(FVX(JX)+FVX(JX+1))/2.
      IX=I
      CLW(I)=0.0
      CMX(I)=0.0
      CMZ(I)=0.0
      CDW(I)=0.0
      DO 30 J=1,NVY
      CLW(I)=CP(IX)*(AXY(IY)*COSA +AYZ(IY)*SINA )+CLW(I)
      CMX(I)=CP(IX)*(XQ(IX)-XCG)*AXY(IY)+CMX(I)
      CMZ(I)=CP(IX)*ZQ(IX)*AYZ(IY)+CMZ(I)
      CDW(I)=CP(IX)*(AXY(IY)*SINA -AYZ(IY)*COSA )+CDW(I)
      IX=IX+NJX
   30 IY=IY+1
      CAX=FAREA*DX
      CSX=FCD/CAX
      CLW(I)=CLW(I)*CSX
      CMX(I)=-CMX(I)/CAX
      CMZ(I)=-CMZ(I)/CAX
   40 CDW(I)=CDW(I)*CSX
C    *           CALC C*W/W(AVG) AT EQUAL INTERVAL OF PHI
      DX=PI/70.
      FX(1)=0.0
      SPHI=0.0
      FCD2=FCD/2.
      DO 50 I=2,70
      SPHI=SPHI+DX
   50 FX(I)=FCD2*(1.-COS(SPHI))
```

```
      CALL CODIM (FDX,CLW,NJX,FX,CLWA,70)
      CALL CODIM (FDX,CMX,NJX,FX,CMXA,70)
      CALL CODIM (FDX,CMZ,NJX,FX,CMZA,70)
      CALL CODIM (FDX,CDW,NJX,FX,CDWA,70)
      SPHI=0.0
      DO 52 I=2,70
      FX(I) = FX(I) / FCD
      SPHI=SPHI+DX
      HSP=SIN(SPHI)/2.
      CLW(I)=HSP*CLWA(I)
      CMX(I)=HSP*CMXA(I)
      CMZ(I)=HSP*CMZA(I)
   52 CDW(I)=HSP*CDWA(I)
C*                 SIMPSON RULE INTEGRATION
      SL1=CLW(70)
      SL2=0.0
      XM1=CMX(70)
      XM2=0.0
      ZM1=CMZ(70)
      ZM2=0.0
      SD1=CDW(70)
      SD2=0.0
      DO 60 I=2,69,2
      SL1= CLW(I)+SL1
      SL2= CLW(I+1)+SL2
      XM1= CMX(I)+XM1
      XM2= CMX(I+1)+XM2
      ZM1= CMZ(I)+ZM1
      ZM2= CMZ(I+1)+ZM2
      SD1= CDW(I)+SD1
   60 SD2= CDW(I+1)+SD2
      DX3=DX/3.
      IDT=IDB+IDW+IDP+IDN
      IF(IDT.NE.0) GO TO 70
      SR=1.
      CR=1.
      GO TO 80
   70 SR=FAREA*AR/WSPAN**2
      CR=FCD/XMAC
   80 FCMXY=DX3*(4.*XM1+2.*XM2)*SR*CR
      FCL=DX3*(4.*SL1+2.*SL2)*SR
      FCMYZ=DX3*(4.*ZM1+2.*ZM2)*SR*CR
      CDF=DX3*(4.*SD1+2.*SD2)*SR
```

```
      FCM=FCMXY+FCMYZ
C*                 PRINT OUTPUT
      WRITE (6,100) FAREA,FCL,CDF,FCMXY,FCMYZ,FCM,(FX(I),CLWA(I),CDWA(I)
     1              ,FX(I+35),CLWA(I+35),CDWA(I+35),I=2,35)
     2              ,FX(36),CLWA(36),CDWA(36)
  100 FORMAT (1H1,10X,45H ** FANPOD LOADS            ** REF. FANPOD AREA=,
     1        F14.5/1H0,10X,   5H* CL=F10.5,5X,   5H* CD=F10.5,3X,
     2        7H* CMXY=F11.5,3X,7H* CMYZ=F11.5,5X, 5H* CM=F11.5/
     3        1H0,12X,1HX,15X,6HCLW/WA,11X,6HCDW/WA,12X,1HX,15X,6HCLW/WA
     4       ,11X,6HCDW/WA/ 1H0,6F17.5 /(1H ,6F17.5))
      RETURN
      END
*DECK WDPQ
      SUBROUTINE WDPQ
C
C*                 CALC WING PRESSURE COEFFICIENTS (DP/Q)
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION   WPE(60)    ,WPC(30)    ,WPS(30)
C
      EQUIVALENCE      (DA(2),WSPAN)    ,(DA(4),XMACH)   ,(DA(11),FWCI)
     1                ,(DA(1210),WPE)   ,(DA(1870),WPC)  ,(DA(1900),WPS)
     2                ,(DA(1960),DAF)   ,(DA(1965),DAK)  ,(DA(1270),WNVC)
     3                ,(DA(1281),WNW)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CPW/ DW1(9),NU,NW
      COMMON/CFG/ DF1(10),NKB,NKF
      COMMON/CFK/ NDA,NUT,THEF,SINTF,SINTF2,COSTF,THEK,SINTK,SINTK2
C
      LARGE       CKV(1000) ,BOU(2000) ,FUS(25000),FAN(23700)
     1           ,EV(52)    ,DM1(944)  ,WPT(60)
     2           ,DM2(22374),PYL(6061),CEL(3700)
     3           ,XF(51,10) ,DM3(561)  ,TA(30,10)
     4           ,DPQR(30)  ,XLE(30)   ,XTLE(30)   ,THE(30)
     5           ,EPS(30)
C
C
      NPC=DA(1272)
      NPS=DA(1273)
C                  SHIFT TO LCM FOR SFC AND WINGD
      DO 10 I=1,NPS
```

```
   10 EPS(I)=WPS(I)
C                CALC MATRIX-F
      IF(WNW.NE.0.0) GO TO 570
      NPS=NW
      IC=NW
      DO 400 IE=1,NPS
      WPS(IC)=EV(IE)
      EPS(IC)=EV(IE)
  400 IC=IC-1
      IC=NKB+NKF
      DO 410 IU=1,NU
      IX=NPS
      DO 410 IE=1,NPS
      IC=IC+1
      XF(IX,IU)=CKV(IC)/BETA
  410 IX=IX-1
      GO TO 580
  570 CALL SFC (NPS,EPS)
C                CALC CHORD LOAD SHAPES FUNCTION
  580 DO 594 I=1,NPC
      THE(I)= ACOS(1.-2.*WPC(I))
      THED2=THE(I)/2.
      TA(I,1)= COS(THED2)/ SIN(THED2)
      ANC=0.0
      DO 590 J=2,NUT
      ANC=ANC+1.
  590 TA(I,J) = SIN(ANC * THE(I))
      IF(DAF.EQ.0.0) GO TO 592
      SINTT=SIN(.5*(THE(I)-THEF))
      IF(ABS(SINTT).LT.0.001) SINTT=0.001
      TA(I,NUT+1)=ALOG(ABS(SIN(.5*(THE(I)+THEF))/SINTT))
  592 IF(DAK.EQ.0.0) GO TO 594
      SINTT=SIN(.5*(THE(I)-THEK))
      IF(ABS(SINTT).LT.0.001) SINTT=0.001
      TA(I,NU)=ALOG(ABS(SIN(.5*(THE(I)+THEK))/SINTT))
  594 CONTINUE
      X3=8.*WSPAN
      CALL WINGD (NPS,EPS,XLE,XTLE,WPT)
      WRITE (6,1000) (WPC(I),I=1,NPC)
      WRITE (6,2000)
      DO 620 NE=1,NPS
  595 EBC=X3/XTLE(NE)
      DO 610 I=1,NPC
```

```
      DPQR(I)=0.0
      DO 600 J=1,NU
  600 DPQR(I)=DPQR(I)+XF(NE,J)*TA(I,J)
  610 DPQR(I)=EBC*DPQR(I)
      WRITE           ( 6,3000)WPS(NE),(DPQR(I),I=1,NPC)
  620 CONTINUE
  640 RETURN
 1000 FORMAT (31H1 ** WING LINEAR PRESSURE COEF./1H0,7X,14H** LIST OF X/
     1C/1H0,6X,10F10.4/(1H ,6X,10F10.4))
 2000 FORMAT(35H0* ETA  ** LISTS OF CP AT ABOVE X/C)
 3000 FORMAT(1H0F6.3,10F10.4/(1H 6X,10F10.4))
      END
*DECK WVAP
      SUBROUTINE WVAP
C
C     *VAP* CALC. VELOCITIES AND PRESSURES WITH THICKNESS
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION   AJ(30)
C
      EQUIVALENCE      (DA(2),WS)       ,(DA(4),XMACH)   ,(DA(1270),XNI)
     1,(DA(1660),AJ)   ,(DA(1960),DAF)   ,(DA(1965),DAK)
     2,(NJC,NJ)        ,(NJS,NEP)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CFG/ IDB,IDF,IDW,IDP,IDN,NQB,NQF,NQW,NQP,NQN
     1            ,NKB,NKF,NKW,NKP,NKN
      COMMON/CPW/ DW1(9),NU,NW,NVC,NVS,NJC,NJS
      COMMON/CFK/ NDA,NUT,THEF,SINTF,SINTF2,COSTF,THEK,SINTK,SINTK2
     1            ,COSTK,SINDF,COSDF,SINDK,COSDK
C
      LARGE       CKV(1000) ,BOU(2000) ,FUS(25000),FAN(23700)
     *            ,DM1(996)  ,WPT(60)   ,BS(650)
     1            ,EP(30)    ,XTLP(30)  ,XOCP(30)
     2            ,CSJ(30,30),TSJ(30,30),UV(30,30)
     3            ,UVP(30,30),VV(30,30) ,VVP(30,30)
     4            ,DM2(25995)
     5            ,XF(51,10) ,SRE(51)   ,S(51,10)
     6            ,BE(30,30) ,CF(30,30) ,CA(30,10) ,TA(30,11) ,TB(30,11)
     7            ,TPTE(30)  ,TPLE(30)  ,PF(30,10)
     8            ,UVU(30)   ,UVL(30,30),VVU(30)   ,VVL(30,30)
     9            ,CPU(30)   ,CPL(30)   ,CPN(30)   ,CPO(30)
```

```
     *              ,AX(250)    ,AY(250)    ,AZ(250)
C
C*                INITIAL THETA AND TRIG. FUNC.
      DTH=PI/XNI
      DO 40 J=1,NJ
      THE=DTH*AJ(J)
      THE2=THE/2.
      TA(J,1)=COS(THE2)/SIN(THE2)
      CF(J,1)=1.
      C=0.
      DO 10 N=1,NUT
      C=C+1.
      CA(J,N)=COS(C*THE)
      CF(J,N+1)=CA(J,N)
   10 TA(J,N+1)=SIN(C*THE)
      TB(J,1)=TA(J,2)+THE
      TB(J,2)=THE2-TA(J,3)/4.
      DO 20 N=3,NUT
      D2=N+N
      D1=D2-4.
   20 TB(J,N)=TA(J,N-1)/D1-TA(J,N+1)/D2
      IF(DAF.EQ.0.0) GO TO 30
      SINTT=SIN(.5*(THE-THEF))
      IF(ABS(SINTT).LT.0.001) SINTT=.001
      AST=ABS(SIN(.5*(THE+THEF))/SINTT)
      TA(J,NUT+1)=ALOG(AST)
      TB(J,NUT+1)=(COS(THE)-COSTF)*ALOG(1./AST)+THE*SINTF
   30 IF(DAK.EQ.0.0) GO TO 40
      SINTT=SIN(.5*(THE-THEK))
      IF(ABS(SINTT).LT.0.001) SINTT=.001
      AST=ABS(SIN(.5*(THE+THEK))/SINTT)
      TA(J,NU)=ALOG(AST)
      TB(J,NU)=(COS(THE)-COSTK)*ALOG(1./AST)+THE*SINTK
   40 CONTINUE
      IF(DA(2000).EQ.0.0) GO TO 200
C*                CALC SLOPE OF CAMBER SURFACE
      NA=0
      DO 41 I=1,NEP
      DO 41 J=1,NJ
      NA=NA+1
   41 BE(J,I)=BS(NA)
      CALL MSOLX(NUT,NJ,NEP,CF,BE,30)
      DO 44 IE=1,NEP
```

```
      DO 44 J=1,NJ
      SUMA=0.0
      DO 42 N=2,NUT
   42 SUMA=CF(N,IE)*CA(J,N-1)+SUMA
      SUMB=0.0
      C=0.0
      DO 43 N=3,NUT,2
      C=C+1.
   43 SUMB=CF(N,IE)/(4.*C**2-1.)+SUMB
   44 CSJ(IE,J)=-SUMA-SUMB
C*                TPS FOR TAN PHI STAR
      HWS=WS/2.
      CALL TPS (HWS,WPT,NEP,EP,TPTE,TPLE)
C*                SPF FOR DERIVATIVE OF F MATRIX
      CALL SFC (NEP,EP)
      CALL SPF
C*                CALC. VELOCITIES AND PRESSURES
      DO 45 IE=1,NEP
      DO 45 J=1,NJ
      UVL(IE,J)=0.0
   45 VVL(IE,J)=0.0
      IF(IDB.EQ.0) GO TO 47
C*                CALC INFLUENCE OF OTHER COMPONENTS ON WING
      DO 46 IE=1,NEP
      DO 46 J=1,NJ
      READ (19) (AX(IS),AY(IS),AZ(IS),IS=1,NKB)
      DO 46 IS=1,NKB
      UVL(IE,J)=UVL(IE,J)+AX(IS)*CKV(IS)
   46 VVL(IE,J)=VVL(IE,J)+AY(IS)*CKV(IS)
   47 IF(IDF.EQ.0) GO TO 49
      DO 48 IE=1,NEP
      DO 48 J=1,NJ
      READ (20) (AX(IS),AY(IS),AZ(IS),IS=1,NKF)
      IC=NKB
      DO 48 IS=1,NKF
      IC=IC+1
      UVL(IE,J)=UVL(IE,J)+AX(IS)*CKV(IC)
   48 VVL(IE,J)=VVL(IE,J)+AY(IS)*CKV(IC)
   49 IF(IDP.EQ.0) GO TO 51
      DO 50 IE=1,NEP
      DO 50 J=1,NJ
      READ (22) (AX(IS),AY(IS),AZ(IS),IS=1,NKP),PUV,PVV
      IC=NKB+NKF+NKW
```

```
      UVL(IE,J)=UVL(IE,J)+PUV
      VVL(IE,J)=VVL(IE,J)+PVV
      DO 50 IS=1,NKP
      IC=IC+1
      UVL(IE,J)=UVL(IE,J)+AX(IS)*CKV(IC)
   50 VVL(IE,J)=VVL(IE,J)+AY(IS)*CKV(IC)
   51 IF(IDN.EQ.0) GO TO 55
      DO 52 IE=1,NEP
      DO 52 J=1,NJ
      READ (9) (AX(IS),AY(IS),AZ(IS),IS=1,NKN)
      IC=NKB+NKF+NKW+NKP
      DO 52 IS=1,NKN
      IC=IC+1
      UVL(IE,J)=UVL(IE,J)+AX(IS)*CKV(IC)
   52 VVL(IE,J)=VVL(IE,J)+AY(IS)*CKV(IC)
C
   55 WS8=8.*WS
      BSQ=BETA**2
      WRITE (6,100)
      DO 90 IE=1,NEP
      C1=WS8/XTLP(IE)
      C2=C1/4.
      DTP=TPTE(IE)-TPLE(IE)
      DO 80 J=1,NJ
      READ (21)
      UVB=UVL(IE,J)/BETA
      VVB=VVL(IE,J)
   56 TP=TPLE(IE)+DTP*XOCP(J)
      TPX=TP*TP
      TP1=1.+TPX
      TP2=SQRT(TP1)
      TPU=TP2*(UV(IE,J)-UVP(IE,J))+1.
      TPV=TP2*(VV(IE,J)-VVP(IE,J))+1.
      DZU=TP1*(TSJ(IE,J)-CSJ(IE,J))**2
      DZL=TP1*(TSJ(IE,J)+CSJ(IE,J))**2
      DSU=SQRT(1.+DZU)
      DSL=SQRT(1.+DZL)
      CPS=0.
      VVS=0.
      DO 58 N=2,NU
      CPS=CPS+XF(IE,N)*TA(J,N)
   58 VVS=VVS+PF(IE,N)*TB(J,N)
      CP1=XF(IE,1)*TA(J,1)
```

```
      CPO(    J)=C1*(CP1+CPS)
      UVS=C2*(CP1*TPU+CPS)
      UVU(    J)=(UV(IE,J)+UVS+(1.+TPX*DSU)/TP1)/DSU+UVB/DSU
      UVL(IE,J)=(UV(IE,J)-UVS+(1.+TPX*DSL)/TP1)/DSL+UVB/DSL
      VV1=2.*PF(IE,1)*TB(J,1)-C2*CP1*TP
      VVS=VV1*TPV+2.*VVS-C2*CPS*TP
      TTP=TP/TP1
      VVU(    J)=(VV(IE,J)+VVS-TTP*(1.-DSU))/DSU*BETA+VVB/DSU
      VVL(IE,J)=(VV(IE,J)-VVS-TTP*(1.-DSL))/DSL*BETA+VVB/DSL
      CPU(    J)=1.-UVU(    J)**2-VVU(    J)**2
      CPL(    J)=1.-UVL(IE,J)**2-VVL(IE,J)**2
C                 COMPRESSIBILITY EFFECTS
      IF(XMACH.LE.0.0) GO TO 78
      XMS=XMACH**2
      XM1=1.42857/XMS
      XM2=0.2*XMS
      IF(XMACH.GT.0.01) GO TO 62
      CPU(J)=CPU(J)+XMS*CPU(J)/4.0
      CPL(J)=CPL(J)+XMS*CPL(J)/4.0
      GO TO 70
   62 CPU(J)=XM1*(1.+XM2*CPU(J))**3.5-XM1
      CPL(J)=XM1*(1.+XM2*CPL(J))**3.5-XM1
C
   70 SINP=TP/TP2
      COSP=1./TP2
      CP2=COSP**2
      SP2=SINP**2
      SCP=SINP*COSP
      CPMA=XMACH*COSP
      CPM2=CPMA**2
      SRCM=SQRT(1.-CPM2)
      CSR=.5*(1.-SRCM)
      CPU(J)=CPU(J)*SRCM/(1.-CPU(J)*CSR)
      CPL(J)=CPL(J)*SRCM/(1.-CPL(J)*CSR)
      BWU=SQRT(1.-CPM2*(1.-CPU(J)*CPMA))
      BWL=SQRT(1.-CPM2*(1.-CPL(J)*CPMA))
      DVMU=UV(IE,J)+UVS+UVB
      DVML=UV(IE,J)-UVS+UVB
      DVTU=(VV(IE,J)+VVS)*BETA+VVB
      DVTL=(VV(IE,J)-VVS)*BETA+VVB
      SBDU=SQRT(BWU**2+DZU)
      SBDL=SQRT(BWL**2+DZL)
      UVU(J)=(BWU*CP2+(BSQ*DVMU*CP2-BETA*DVTU*SCP)/SRCM )/SBDU
```

```
     *       +(1.+DVMU)*SP2+DVTU*SCP
      UVL(IE,J)=(BWL*CP2+(BSQ*DVML*CP2-BETA*DVTL*SCP)/SRCM )/SBDL
     *           +(1.+DVML)*SP2+DVTL*SCP
      VVU(J)=-(BWU*SCP+(BSQ*DVMU*SCP-BETA*DVTU*SP2)/SRCM )/SBDU
     *       +(1.+DVMU)*SCP+DVTU*CP2
      VVL(IE,J)=-(BWL*SCP+(BSQ*DVML*SCP-BETA*DVTL*SP2)/SRCM )/SBDL
     *           +(1.+DVML)*SCP+DVTL*CP2
      CPU(J)=1.-UVU(J)**2-VVU(J)**2
      CPL(J)=1.-UVL(IE,J)**2-VVL(IE,J)**2
      IF(XMACH.GT.0.01) GO TO 72
      CPU(J)=CPU(J)+XMS*CPU(J)/4.0
      CPL(J)=CPL(J)+XMS*CPL(J)/4.0
      GO TO 78
   72 CPU(J)=XM1*(1.+XM2*CPU(J))**3.5-XM1
      CPL(J)=XM1*(1.+XM2*CPL(J))**3.5-XM1
C                SAVE CP FOR LOAD CALC
   78 BE(IE,J)=CPU(J)
      CF(IE,J)=CPL(J)
   80 CPN(    J)=CPL(    J)-CPU(    J)
   90 WRITE (6,120) EP(IE),(XOCP(J),UVU(    J),UVL(IE,J),VVU(    J),
     1         VVL(IE,J),CPU(    J),CPL(    J),CPN(    J),CPO(    J),J=1,NJ)
  100 FORMAT (54H1 **WING NON-LINEAR VELOCITY AND PRESSURE COEFFICIENTS)
  120 FORMAT (8H0  ETA =,F7.4/1H ,7X,3HX/C,7X,6HU/V UP,6X,7HU/V LOW,5X,
     1        6HV/V UP,6X,7HV/V LOW,6X,5HCP UP,7X,6HCP LOW,6X,6HCP L-U,
     2        5X,7HCP NETL/(1H ,9F12.5))
      GO TO 300
C                CALC LINEAR PRESSURE COEFFICIENTS AND SKIP RECORDS
  200 WS8=8.*WS
      CALL SFC (NEP,EP)
      DO 220 IE=1,NEP
      C1=WS8/XTLP(IE)
      DO 220 J=1,NJ
      READ (21)
      IF(IDB.NE.0) READ (19)
      IF(IDF.NE.0) READ (20)
      IF(IDP.NE.0) READ (22)
      IF(IDN.NE.0) READ (9)
C
      CPS=0.0
      DO 210 N=1,NU
  210 CPS=CPS+XF(IE,N)*TA(J,N)
  220 BE(IE,J)=C1*CPS
C
```

```
  300 RETURN
      END
*DECK WLOAD
      SUBROUTINE WLOAD(WCL,WCD,WCM)
C
C*                   CALC. WING LOAD PARAMETERS
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION  WJS(1),WJC(1)
C
      EQUIVALENCE      (DA(1),AR)       ,(DA(2),SPAN)    ,(DA(3),XMAC)
     1,(DA(7),XCG)     ,(DA(1270),WNVC) ,(DA(1660),WJC)  ,(DA(1690),WJS)
     2,(DA(1960),DAF)  ,(DA(1961),PXCF) ,(DA(1962),ETFI),(DA(1963),ETFO)
     3,(DA(1965),DAK)  ,(DA(1966),PXCK) ,(DA(1967),ETKI),(DA(1968),ETKO)
C
      COMMON/CRG/ PI
      COMMON/CFG/ IDB,IDF
      COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3,NE5,IW1(3),NVC,NVS,NJC,NJS
      COMMON/CFK/ IK1(10),SINDF,COSDF,SINDK,COSDK
C
      LARGE         BOU(3000)  ,FUS(25000),FAN(23700)
     1              ,EV(52)     ,XLV(52)    ,XTLV(52)   ,DM1(900)   ,BS(650)
     2              ,EP(30)     ,XTLP(30)   ,XOCP(30)   ,DM2(900)   ,TSJ(30,30)
     3              ,DM3(3600)  ,XR(1800)   ,YR(1800)   ,ZR(1800)
     4              ,DM4(10834),PYL(6061)  ,CEL(3700)  ,DM5(1071)
     5              ,CPU(30,30),CPL(30,30),DSS(30,30)
     6              ,THI(32)    ,CNI(32)    ,CMI(32)    ,CDI(32)    ,STH(30)
     7              ,THO(40)    ,CNO(40)    ,CMO(40)    ,CDO(40)
     8              ,EPI(31)    ,CNCCI(31)  ,CMCCI(31)  ,CDCCI(31)  ,XCCP(52)
     9              ,EVO(52)    ,XLE(52)    ,CNCCO(52)  ,CMCCO(52)  ,CDCCO(52)
C
C*                   CALC DS/DSLE OF ROOT SECTIONS
      IF(IDB.NE.0) DSLEB=SQRT((YR(1201)-YR(1202))**2+(ZR(1201)-ZR(1202))
     1 **2)
      IF(IDF.EQ.0) GO TO 10
      DSLE1=SQRT((YR(1)-YR(2))**2+(ZR(1)-ZR(2))**2)
      DSLE2=SQRT((YR(601)-YR(602))**2+(ZR(601)-ZR(602))**2)
C                    SPANWISE LOOP
   10 DO 50 IJS=1,NJS
      IY=WJS(IJS)
      IEB=IY-NE5
      IEF=IY-NE3
```

```
C               CHORDWISE LOOP
      DO 50 IJC=1,NJC
      IX=WJC(IJC)
      DSS(IJS,IJC)=1.0
      IF(IDF.EQ.0) GO TO 30
      IF(IEF.LE.0) GO TO 50
      IF(IEF.GT.3) GO TO 20
      DSLE=DSLE1
      IC=4*IX+IEF
      GO TO 40
C               INBOARD ROOT OF FANPOD
   20 IF(IEF.GT.6) GO TO 30
      DSLE=DSLE2
      IC=4*IX+IEF+597
      GO TO 40
C               ROOT OF FUSELAGE
   30 IF(IDB.EQ.0) GO TO 50
      IF(IEB.LE.0) GO TO 50
      IF(IEB.GT.3) GO TO 50
      DSLE=DSLEB
      IC=4*IX+IEB+1200
C
   40 DY=(YR(IC)-YR(IC+1)+YR(IC-4)-YR(IC-3))/2.0
      DZ=(ZR(IC)-ZR(IC+1)+ZR(IC-4)-ZR(IC-3))/2.0
      DS=SQRT(DY**2+DZ**2)
      DSS(IJS,IJC)=DS/DSLE
   50 CONTINUE
C               OUTBOARD ROOT OF FANPOD
C               CALC CHORDWISE THETAS
      DTH=PI/WNVC
      CONS=.5*AR/SPAN*DTH
      THO(1)=DTH
      DO 60 I=2,NVC
   60 THO(I)=THO(I-1)+DTH
C
      THI(1)=0.0
      DO 70 IC=1,NJC
      THI(IC+1)=WJC(IC)*DTH
   70 STH(IC)=SIN(THI(IC+1))
      NJI=NJC+2
      THI(NJI)=PI
C
C*              CALC LIFT,DRAG AND MOMENT COEFFICIENTS
```

```
      IV=NJS
      NA=0
      CNI(1)=0.0
      CMI(1)=0.0
      CDI(1)=0.0
      CNI(NJI)=0.0
      CMI(NJI)=0.0
      CDI(NJI)=0.0
C
      DO 160 IE=1,NJS
      DO 140 IC=1,NJC
      TD=0.0
C                CALC TAN OF FLAP DEFLECTION
      IF(DAF.EQ.0.0) GO TO 100
      IF(EP(IE).LE.ETFI) GO TO 100
      IF(EP(IE).GE.ETFO) GO TO 100
      IF(XOCP(IC).LE.PXCF) GO TO 100
      TD=SINDF/COSDF
      GO TO 110
C                CALC TAN OF KRUGER DEFLECTION
  100 IF(DAK.EQ.0) GO TO 110
      IF(EP(IE).LE.ETKI) GO TO 110
      IF(EP(IE).GE.ETKO) GO TO 110
      IF(XOCP(IC).LE.PXCK) GO TO 110
      TD=SINDK/COSDK
C                INCLUDE THICKNESS SLOPES,CAMBER AND TWIST
  110 NA=NA+1
      IF(DA(2000).EQ.0.0) GO TO 120
      STCTU=-TSJ(IE,IC)+BS(NA)
      STCTL=TSJ(IE,IC)+BS(NA)
      TAU=(TD+STCTU)/(1.-TD*STCTU)
      TAL=(TD+STCTL)/(1.-TD*STCTL)
      CPN=CPL(IE,IC)-CPU(IE,IC)
      CPD=TAL*CPL(IE,IC)-TAU*CPU(IE,IC)
      GO TO 130
  120 CPN=CPU(IE,IC)
      CPD=(TD+BS(NA))/(1.-TD*BS(NA))*CPU(IE,IC)
  130 CPM=CPN*XOCP(IC)*XTLP(IE)
      DST=DSS(IE,IC)*STH(IC)
      CNI(IC+1)=DST*CPN
      CMI(IC+1)=DST*CPM
  140 CDI(IC+1)=DST*CPD
C                CHORDWISE EXPANSION
```

```
      CALL CODIM(THI,CNI,NJI,THO,CNO,NVC)
      CALL CODIM(THI,CMI,NJI,THO,CMO,NVC)
      CALL CODIM(THI,CDI,NJI,THO,CDO,NVC)
C                SUM CN,CM AND CD CHORDWISE
      CN=0.0
      CM=0.0
      CD=0.0
      DO 150 I=1,NVC
      CN=CN+CNO(I)
      CM=CM+CMO(I)
  150 CD=CD+CDO(I)
C                SET PARAMETERS IN ASCENDING ORDER
      EPI(IV)=EP(IE)
      CC=CONS*XTLP(IE)
      CNCCI(IV)=CC*CN
      CMCCI(IV)=-CC*CM/XMAC
      CDCCI(IV)=CC*CD
  160 IV=IV-1
C
C*               INTEGRATE SPANWISE
      NSI=NJS+1
      EPI(NSI)=1.0
      CNCCI(NSI)=0.0
      CMCCI(NSI)=0.0
      CDCCI(NSI)=0.0
      IV=NE2
      DO 170 I=1,NE2
      EVO(IV)=EV(I)
      XLE(IV)=XLV(I)
  170 IV=IV-1
C                SPANWISE EXPANSION
      CALL CODIM(EPI,CNCCI,NSI,EVO,CNCCO,NE2)
      CALL CODIM(EPI,CMCCI,NSI,EVO,CMCCO,NE2)
      CALL CODIM(EPI,CDCCI,NSI,EVO,CDCCO,NE2)
C                CALC TOTAL WING LIFT,DRAG AND MOMENT
      WCL=0.0
      WCM=0.0
      WCD=0.0
      NEE=NE2-NE3-3
      DO 240 I=1,NE2
      IF(ABS(CNCCO(I)).LT.0.000001) CNCCO(I)=0.000001
      XCCP(I)=-CMCCO(I)/CNCCO(I)
      CMCG=CMCCO(I)-XLE(I)*CNCCO(I)+XCG*CNCCO(I)
```

```
      IF(IDF.EQ.0) GO TO 200
      IF(I.GT.NEE) GO TO 200
      CNX=DLI*CNCCO(I)
      CMX=DLI*CMCG
      CDX=DLI*CDCCO(I)
      GO TO 210
  200 CNX=DLO*CNCCO(I)
      CMX=DLO*CMCG
      CDX=DLO*CDCCO(I)
  210 IF(IDB.NE.0) GO TO 220
      IF(I.NE.1) GO TO 220
      CNX=CNX/2.
      CMX=CMX/2.
      CDX=CDX/2.
      GO TO 230
  220 IF(I.NE.NE2) GO TO 230
      CNX=CNX/3.
      CMX=CMX/3.
      CDX=CDX/3.
  230 WCL=WCL+CNX
      WCM=WCM+CMX
  240 WCD=WCD+CDX
      IF(ABS(WCL).LT.0.000001) WCL=0.000001
      WXC=-WCM/WCL
C                PRINT OUTPUT
C
      WRITE (6,300) (I,EVO(I),CNCCO(I),CDCCO(I),CMCCO(I),XCCP(I),
     1  I=1,NE2)
  300 FORMAT (1H1,20X,17H ** WING LOADS **/1H ,24X,3HETA,12X,6HCNC/CA,
     1       9X,6HCDC/CA,9X,6HCMC/CA,9X,6HX/C CP/(1H ,I15,5F15.5))
C
      WRITE (6,310) WCL,WCD,WCM,WXC
  310 FORMAT (1H0,20X,22H ** TOTAL WING LOADS */1H ,39X,3HWCL,12X,3HWCD,
     1       12X,3HWCM,12X,3HX/C/1H ,30X,4F15.5)
C
      RETURN
      END
*DECK PDPQ
      SUBROUTINE PDPQ
C
C*               CALC PYLON PRESSURE COEFFICIENTS (DP/Q)
C
      COMMON/DAT/ DA(5000)
```

```
C
      DIMENSION   PJC(18)
C
      EQUIVALENCE      (DA(2500),PHT)  ,(DA(2512),PJC)  ,(DA(19),PVPC)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CFG/ IG1(10),NKB,NKF,NKW,NKP
      COMMON/CPP/ HPS,NVSH,NVT,DL ,NEVP,ZCON,YCON,NVC,NVS,NJC,NUP
C
      LARGE        CKV(1000) ,DM1(74130)
     1             ,EV(22)     ,XLE(22)    ,XTLE(22)   ,WPE(39)
     2             ,XD(60)     ,YD(60)     ,ZD(60)
     3             ,DPW1(1220),CFJ(20)     ,STH(20)    ,ACT(20)
     4             ,DM2(4496)  ,CEL(3700)  ,XF(20,10)  ,TA(20,11)
     5             ,DPQR(20)   ,WPS(20)    ,WPC(20)    ,XTL(20)
     6             ,YC(90)     ,ZC(90)
C
C              PRINT PYLON ROOT COORDINATES
      WRITE (6,700)
      NTD=3*NJC
      DO 100 IM=1,NTD
      YC(IM)=YD(IM)/BETA
  100 ZC(IM)=ZD(IM)/BETA
      IC=NVS
      DO 105 IE=1,NVS
      XTL(IC)=XTLE(IE)
      WPS(IC)=EV(IE)
  105 IC=IC-1
      IS=NTD-NJC+1
      IE=NTD
      DO 110 NE=1,3
      WRITE (6,3000)WPS(NE),(XD(K),K=IS,IE)
      WRITE (6,720) (YC(K),K=IS,IE)
      WRITE (6,720) (ZC(K),K=IS,IE)
      IS=IS-NJC
  110 IE=IE-NJC
C              CALC MATRIX-F
      IC=NKB+NKF+NKW
      DO 410 IU=1,NUP
      IX=NVS
      DO 410 IE=1,NVS
      IC=IC+1
      XF(IX,IU)=CKV(IC)/BETA
```

```
  410 IX=IX-1
C                CALC CHORD LOAD SHAPES FUNCTION
      DO 590 I=1,NJC
      IJ=PJC(I)+1.
      WPC(I)=CFJ(IJ)
      THED2=ACT(IJ)/2.
      TA(I,1)= COS(THED2)/ SIN(THED2)
      ANC=0.0
      DO 590 J=2,NUP
      ANC=ANC+1.
  590 TA(I,J) = SIN(ANC *ACT(IJ))
      X3=16.*PHT
      WRITE (6,1000) (WPC(I),I=1,NJC)
      WRITE (6,2000)
      IC=PVPC
      IE=NVS-IC
      DO 620 NE=1,IE
  595 EBC=X3/XTL(NE)
      DO 610 I=1,NJC
      DPQR(I)=0.0
      DO 600 J=1,NUP
  600 DPQR(I)=DPQR(I)+XF(NE,J)*TA(I,J)
  610 DPQR(I)=EBC*DPQR(I)
      WRITE (6,3000) WPS(NE),(DPQR(I),I=1,NJC)
  620 CONTINUE
  640 RETURN
  700 FORMAT (27H1 ** PYLON ROOT COORDINATES/23H0 * ETA * LIST OF X,Y,Z)
  720 FORMAT (1H0,6X,10F10.4/(1H ,6X,10F10.4))
 1000 FORMAT (31H1 **PYLON LINEAR PRESSURE COEF./1H0,7X,14H** LIST OF X/
     1C/1H0,6X,10F10.4/(1H ,6X,10F10.4))
 2000 FORMAT(35H0* ETA  ** LISTS OF CP AT ABOVE X/C)
 3000 FORMAT(1H0F6.3,10F10.4/(1H 6X,10F10.4))
      END
*DECK PVAP
      SUBROUTINE PVAP
C
C     *VAP* CALC. VELOCITIES AND PRESSURES WITH THICKNESS
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION   PJC(18)
C
      EQUIVALENCE     (DA(4),XMACH)   ,(DA(2500),PHT) ,(DA(2512),PJC)
```

```
     1,(DA(19),PVPC)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CFG/ IDB,IDF,IDW,IDP,IDN,IG1(5),NKB,NKF,NKW,NKP,NKN
      COMMON/CPP/ DPP(3),DL,DPP1(3),NVC,NVS,NJC,NUP
C
      LARGE        CKV(1000)  ,DM1(74130)
     1             ,EV(22)    ,XLV(22)    ,XTLV(22)   ,WPE(39)    ,DP1(1400)
     2             ,CFJ(20)   ,STH(20)    ,ACT(20)    ,DP2(466)
     3             ,CSJ(20,20),TSJ(20,20),UV(20,20)  ,UVP(20,20)
     4             ,VV(20,20) ,VVP(20,20),DPP2(220)  ,BS(360)
     5             ,DM2(4750) ,XF(20,10)  ,BE(20,20)  ,CF(20,20)  ,CA(20,10)
     6             ,TA(20,11) ,TB(20,10)  ,TPTE(20)   ,TPLE(20)   ,PF(20,10)
     7             ,UVU(20,20),UVL(20,20),VVU(20,20),VVL(20,20)
     8             ,CPU(20,20),CPL(20,20),CPN(20,20),CPO(20,20)
     9             ,AX(250)   ,AY(250)    ,AZ(250)    ,XOCP(20)
C
C*                 INITIAL THETA AND TRIG. FUNC.
      DO 20 J=1,NJC
      IJ=PJC(J)+1.
      THE=ACT(IJ)
      THE2=THE/2.0
      XOCP(J)=CFJ(IJ)
      TA(J,1)=COS(THE2)/SIN(THE2)
      CF(J,1)=1.0
      C=0.
      DO 10 N=1,NUP
      C=C+1.
      TA(J,N+1)=SIN(C*THE)
      CA(J,N)=COS(C*THE)
   10 CF(J,N+1)=CA(J,N)
      TB(J,1)=TA(J,2)+THE
      TB(J,2)=THE2-TA(J,3)/4.
      DO 20 N=3,NUP
      D2=N+N
      D1=D2-4.
   20 TB(J,N)=TA(J,N-1)/D1-TA(J,N+1)/D2
      IF(DA(3000).EQ.0.0) GO TO 200
C                  CALC SLOPE OF CAMBER SURFACE
      NA=0
      DO 30 I=1,NVS
      DO 30 J=1,NJC
      NA=NA+1
```

```
   30 BE(J,I)=BS(NA)
      CALL MSOLX (NUP,NJC,NVS,CF,BE,20)
C*                 CALC DERIVATIVES OF F VALUES
      IC=NKB+NKF+NKW
      DO 36 IU=1,NUP
      DO 36 IE=1,NVS
      IC=IC+1
   36 XF(IE,IU)=CKV(IC)/BETA
      DLA=.625*DL
      DLB=DLA+DL
      DLC=DL+DL
      K=NKB+NKF+NKW
      DO 40 IU=1,NUP
      K=K+2
      PF(1,IU)=(CKV(K-1)-CKV(K))/DLA
      PF(2,IU)=(CKV(K-1)-CKV(K+1))/DLB
      DO 40 IE=3,NVS
      K=K+1
      IF(IE.NE.NVS) GO TO 38
      PF(IE,IU)=0.0
      GO TO 40
   38 PF(IE,IU)=(CKV(K-1)-CKV(K+1))/DLC
   40 CONTINUE
C*                 CALC SLOPE OF CAMBER SURFACE
      DO 46 IE=1,NVS
      DO 46 J=1,NJC
      SUMA=0.0
      DO 42 N=2,NUP
   42 SUMA=CF(N,IE)*CA(J,N-1)+SUMA
      SUMB=0.0
      C=0.0
      DO 44 N=3,NUP,2
      C=C+1.
   44 SUMB=CF(N,IE)/(4.*C**2-1.)+SUMB
   46 CSJ(IE,J)=-SUMA-SUMB
C*                 TPS FOR TAN PHI STAR
      CALL TPS (PHT,WPE,NVS,EV,TPTE,TPLE)
C*                 CALC INFLUENCE OF OTHER COMPONENTS ON PYLON
      DO 147 IE=1,NVS
      DO 147 J=1,NJC
      UVL(IE,J)=0.0
  147 VVL(IE,J)=0.0
C
```

```
      IF(IDB.EQ.0) GO TO 148
      DO 47 IE=1,NVS
      DO 47 J=1,NJC
      UVL(IE,J)=0.0
      VVL(IE,J)=0.0
      READ (19) (AX(IS),AY(IS),AZ(IS),IS=1,NKB)
      DO 47 IS=1,NKB
      UVL(IE,J)=UVL(IE,J)+AX(IS)*CKV(IS)
   47 VVL(IE,J)=VVL(IE,J)+AY(IS)*CKV(IS)
C
  148 IF(IDF.EQ.0) GO TO 150
      DO 48 IE=1,NVS
      DO 48 J=1,NJC
      READ (20) (AX(IS),AY(IS),AZ(IS),IS=1,NKF)
      IC=NKB
      DO 48 IS=1,NKF
      IC=IC+1
      UVL(IE,J)=UVL(IE,J)+AX(IS)*CKV(IC)
   48 VVL(IE,J)=VVL(IE,J)+AY(IS)*CKV(IC)
C
  150 IF(IDW.EQ.0) GO TO 151
      DO 50 IE=1,NVS
      DO 50 J=1,NJC
      READ (21) (AX(IS),AY(IS),AZ(IS),IS=1,NKW),WUV,WVV
      IC=NKB+NKF
      UVL(IE,J)=UVL(IE,J)+WUV
      VVL(IE,J)=VVL(IE,J)+WVV
      DO 50 IS=1,NKW
      IC=IC+1
      UVL(IE,J)=UVL(IE,J)+AX(IS)*CKV(IC)
   50 VVL(IE,J)=VVL(IE,J)+AY(IS)*CKV(IC)
C
  151 IF(IDP.EQ.0) GO TO 152
      DO 51 IE=1,NVS
      DO 51 J=1,NJC
      READ (22) (AX(IS),AY(IS),AZ(IS),IS=1,NKP),PUV,PVV
      IC=NKB+NKF+NKW
      UVL(IE,J)=UVL(IE,J)+PUV
      VVL(IE,J)=VVL(IE,J)+PVV
      DO 51 IS=1,NKP
      IC=IC+1
      UVL(IE,J)=UVL(IE,J)+AX(IS)*CKV(IC)
   51 VVL(IE,J)=VVL(IE,J)+AY(IS)*CKV(IC)
```

```
C
  152 IF(IDN.EQ.0) GO TO 55
      DO 52 IE=1,NVS
      DO 52 J=1,NJC
      READ (9) (AX(IS),AY(IS),AZ(IS),IS=1,NKN)
      IC=NKB+NKF+NKW+NKP
      DO 52 IS=1,NKN
      IC=IC+1
      UVL(IE,J)=UVL(IE,J)+AX(IS)*CKV(IC)
   52 VVL(IE,J)=VVL(IE,J)+AY(IS)*CKV(IC)
C*                 CALC. VELOCITIES AND PRESSURES
   55 PS8=16.*PHT
      BSQ=BETA**2
      DO 80 IE=1,NVS
      C1=PS8/XTLV(IE)
      C2=C1/4.
      DTP=TPTE(IE)-TPLE(IE)
      DO 80 J=1,NJC
      UVB=UVL(IE,J)/BETA
      VVB=VVL(IE,J)
      TP=TPLE(IE)+DTP*XOCP(J)
      TPX=TP*TP
      TP1=1.+TPX
      TP2=SQRT(TP1)
      TPU=TP2*(UV(IE,J)-UVP(IE,J))+1.
      TPV=TP2*(VV(IE,J)-VVP(IE,J))+1.
      DZU=TP1*(TSJ(IE,J)-CSJ(IE,J))**2
      DZL=TP1*(TSJ(IE,J)+CSJ(IE,J))**2
      DSU=SQRT(1.+DZU)
      DSL=SQRT(1.+DZL)
      CPS=0.
      VVS=0.
      DO 58 N=2,NUP
      CPS=CPS+XF(IE,N)*TA(J,N)
   58 VVS=VVS+PF(IE,N)*TB(J,N)
      CP1=XF(IE,1)*TA(J,1)
      CPO(IE,J)=C1*(CP1+CPS)
      UVS=C2*(CP1*TPU+CPS)
      UVU(IE,J)=(UV(IE,J)+UVS+(1.+TPX*DSU)/TP1)/DSU+UVB/DSU
      UVL(IE,J)=(UV(IE,J)-UVS+(1.+TPX*DSL)/TP1)/DSL+UVB/DSL
      VV1=2.*PF(IE,1)*TB(J,1)-C2*CP1*TP
      VVS=VV1*TPV+2.*VVS-C2*CPS*TP
      TTP=TP/TP1
```

```
      VVU(IE,J)=(VV(IE,J)+VVS-TTP*(1.-DSU))/DSU*BETA+VVB/DSU
      VVL(IE,J)=(VV(IE,J)-VVS-TTP*(1.-DSL))/DSL*BETA+VVB/DSL
      CPU(IE,J)=1.-UVU(IE,J)**2-VVU(IE,J)**2
      CPL(IE,J)=1.-UVL(IE,J)**2-VVL(IE,J)**2
C              COMPRESSIBILITY EFFECTS
      IF(XMACH.LE.0.0) GO TO 78
      XMS=XMACH**2
      XM1=1.42857/XMS
      XM2=0.2*XMS
      IF(XMACH.GT.0.01) GO TO 62
      CPU(IE,J)=CPU(IE,J)+XMS*CPU(IE,J)/4.0
      CPL(IE,J)=CPL(IE,J)+XMS*CPL(IE,J)/4.0
      GO TO 70
   62 CPU(IE,J)=XM1*(1.+XM2*CPU(IE,J))**3.5-XM1
      CPL(IE,J)=XM1*(1.+XM2*CPL(IE,J))**3.5-XM1
C
   70 SINP=TP/TP2
      COSP=1./TP2
      CP2=COSP**2
      SP2=SINP**2
      SCP=SINP*COSP
      CPMA=XMACH*COSP
      CPM2=CPMA**2
      SRCM=SQRT(1.-CPM2)
      CSR=.5*(1.-SRCM)
      CPU(IE,J)=CPU(IE,J)*SRCM/(1.-CPU(IE,J)*CSR)
      CPL(IE,J)=CPL(IE,J)*SRCM/(1.-CPL(IE,J)*CSR)
      BWU=SQRT(1.-CPM2*(1.-CPU(IE,J)*CPMA))
      BWL=SQRT(1.-CPM2*(1.-CPL(IE,J)*CPMA))
      DVMU=UV(IE,J)+UVS+UVB
      DVML=UV(IE,J)-UVS+UVB
      DVTU=(VV(IE,J)+VVS)*BETA+VVB
      DVTL=(VV(IE,J)-VVS)*BETA+VVB
      SBDU=SQRT(BWU**2+DZU)
      SBDL=SQRT(BWL**2+DZL)
      UVU(IE,J)=(BWU*CP2+(BSQ*DVMU*CP2-BETA*DVTU*SCP)/SRCM )/SBDU
     *        +(1.+DVMU)*SP2+DVTU*SCP
      UVL(IE,J)=(BWL*CP2+(BSQ*DVML*CP2-BETA*DVTL*SCP)/SRCM )/SBDL
     *            +(1.+DVML)*SP2+DVTL*SCP
      VVU(IE,J)=-(BWU*SCP+(BSQ*DVMU*SCP-BETA*DVTU*SP2)/SRCM )/SBDU
     *        +(1.+DVMU)*SCP+DVTU*CP2
      VVL(IE,J)=-(BWL*SCP+(BSQ*DVML*SCP-BETA*DVTL*SP2)/SRCM )/SBDL
     *            +(1.+DVML)*SCP+DVTL*CP2
```

```
      CPL(IE,J)=1.-UVL(IE,J)**2-VVL(IE,J)**2
      CPU(IE,J)=1.-UVU(IE,J)**2-VVU(IE,J)**2
      IF(XMACH.GT.0.01) GO TO 72
      CPU(IE,J)=CPU(IE,J)+XMS*CPU(IE,J)/4.0
      CPL(IE,J)=CPL(IE,J)+XMS*CPL(IE,J)/4.0
      GO TO 78
   72 CPU(IE,J)=XM1*(1.+XM2*CPU(IE,J))**3.5-XM1
      CPL(IE,J)=XM1*(1.+XM2*CPL(IE,J))**3.5-XM1
   78 CPN(IE,J)=CPL(IE,J)-CPU(IE,J)
   80 CONTINUE
C                 PRINT
      WRITE (6,100)
      IS=PVPC+1.0
      DO 90 IE=IS,NVS
   90 WRITE (6,120) EV(IE),(XOCP(J),UVU(IE,J),UVL(IE,J),VVU(IE,J),
     1         VVL(IE,J),CPU(IE,J),CPL(IE,J),CPN(IE,J),CPO(IE,J),J=1,NJC)
  100 FORMAT (54H1 *PYLON NON-LINEAR VELOCITY AND PRESSURE COEFFICIENTS)
  120 FORMAT (8H0  ETA =,F7.4/1H ,7X,3HX/C,7X,6HU/V UP,6X,7HU/V LOW,5X,
     1         6HV/V UP,6X,7HV/V LOW,6X,5HCP UP,7X,6HCP LOW,6X,6HCP L-U,
     2         5X,7HCP NETL/(1H ,9F12.5))
      GO TO 300
C                 CALC LINEAR PRESSURE COEFFICIENTS AND SKIP RECORDS
  200 PS8=16.*PHT
      DO 220 IE=1,NVS
      C1=PS8/XTLV(IE)
      DO 220 J=1,NJC
C
      CPS=0.0
      DO 210 N=1,NUP
  210 CPS=CPS+XF(IE,N)*TA(J,N)
  220 CPU(IE,J)=C1*CPS
C
  300 RETURN
      END
*DECK PLOAD
      SUBROUTINE PLOAD (PCL,PCD,PCM)
C
C*                CALC PYLON LOAD PARAMETERS
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION  PJC(1)
C
```

```
      EQUIVALENCE      (DA(1),AR)        ,(DA(2),SPAN)    ,(DA(3),XMAC)
     1,(DA(7),XCG)     ,(DA(19),PVPC)    ,(DA(2507),PNVC),(DA(2512),PJC)
     2,(DA(2492),PDA)  ,(DA(2501),POX)
C
      COMMON/CRG/ PI,PI4,RC
      COMMON/CPP/ HPS,NVSH,NVT,DL,NEVP,ZCON,YCON,NVC,NVS,NJC,NUP
C
      LARGE        DM1(75130),EV(22)       ,XLV(22)     ,XILV(22)    ,DM2(219)
     1             ,XR(400)    ,YR(400)     ,ZR(400)     ,CFI(20)     ,CFJ(20)
     2             ,DM3(906)   ,TSJ(20,20),DM4(1820) ,BS(360)     ,DM5(4950)
     3             ,DSS(20,20),DM6(2860)  ,CPU(20,20),CPL(20,20)
     4             ,THI(22)    ,CNI(22)     ,CMI(22)     ,CDI(22)     ,STH(20)
     5             ,THO(20)    ,CNO(20)     ,CMO(20)     ,CDO(20)     ,XCCP(20)
     6             ,EVO(20)    ,XLE(20)     ,CNCC(20)    ,CMCC(20)    ,CDCC(20)
C
C*               CALC DS/DSLE OF ROOT SECTION
      DSLE=SQRT((YR(1)-YR(2))**2+(ZR(1)-ZR(2))**2)
      DO 10 IJS=1,NVS
      IEP=IJS-NEVP
      DO 10 IJC=1,NJC
      DSS(IJS,IJC)=1.0
      IF(IEP.LE.0) GO TO 10
      IX=PJC(IJC)
      IC=4*IX+IEP
      DY=(YR(IC)-YR(IC+1)+YR(IC-4)-YR(IC-3))/2.
      DZ=(ZR(IC)-ZR(IC+1)+ZR(IC-4)-ZR(IC-3))/2.
      DSS(IJS,IJC)=SQRT(DY**2+DZ**2)/DSLE
   10 CONTINUE
C                CALC CHORDWISE THETAS
      DTH=PI/PNVC
      CONS=.5*AR/SPAN*DTH
      THO(1)=DTH
      DO 20 I=2,NVC
   20 THO(I)=THO(I-1)+DTH
C
      THI(1)=0.0
      DO 30 IC=1,NJC
      THI(IC+1)=PJC(IC)*DTH
   30 STH(IC)=SIN(THI(IC+1))
      NJI=NJC+2
      THI(NJI)=PI
C
C*               CALC LIFT,DRAG AND MOMENT COEFFICIENTS
```

```
      IV=NVS-PVPC
      NA=0
      CNI(1)=0.0
      CMI(1)=0.0
      CDI(1)=0.0
      CNI(NJI)=0.0
      CMI(NJI)=0.0
      CDI(NJI)=0.0
C
      DO 90 IE=1,NVS
      IF(IE.GT.PVPC) GO TO 40
      NA=NA+NJC
      GO TO 90
   40 DO 70 IC=1,NJC
      NA=NA+1
C                 CALC TAN ALPHA AND CP NET
      IF(DA(3000).EQ.0.0) GO TO 50
      TAU=-TSJ(IE,IC)+BS(NA)
      TAL=TSJ(IE,IC)+BS(NA)
      CPN=CPL(IE,IC)-CPU(IE,IC)
      CPD=TAL*CPL(IE,IC)-TAU*CPU(IE,IC)
      GO TO 60
C
   50 CPN=CPU(IE,IC)
      CPD=BS(NA)*CPU(IE,IC)
   60 IX=PJC(IC)
      CPM=CPN*CFJ(IX+1)*XTLV(IE)
      DST=DSS(IE,IC)*STH(IC)
      CNI(IC+1)=DST*CPN
      CMI(IC+1)=DST*CPM
   70 CDI(IC+1)=DST*CPD
C                 CHORDWISE EXPANSION
      CALL CODIM (THI,CNI,NJI,THO,CNO,NVC)
      CALL CODIM (THI,CMI,NJI,THO,CMO,NVC)
      CALL CODIM (THI,CDI,NJI,THO,CDO,NVC)
C                 SUM,CN,CM AND CD CHORDWISE
      CN=0.0
      CM=0.0
      CD=0.0
      DO 80 I=1,NVC
      CN=CN+CNO(I)
      CM=CM+CMO(I)
   80 CD=CD+CDO(I)
```

```
C                   SET PARAMETERS IN ASCENDING ORDER
      EVO(IV)=EV(IE)
      CC=CONS*XTLV(IE)
      CNCC(IV)=CC*CN
      CMCC(IV)=-CC*CM/XMAC
      CDCC(IV)=CC*CD
      XLE(IV)=XLV(IE)
      IV=IV-1
   90 CONTINUE
C                   CALC TOTAL PYLON LIFT ,DRAG AND MOMENT
      PCL=0.0
      PCM=0.0
      PCD=0.0
      NEE=NVS-PVPC
      CDA=COS((90.-PDA)/RC)
C
      DO 110 I=1,NEE
      IF(ABS(CNCC(I)).LT.0.000001) CNCC(I)=0.000001
      XCCP(I)=-CMCC(I)/CNCC(I)
      XARM=XCG-POX-XLE(I)
      CMCG=CMCC(I)+XARM*CNCC(I)
      CNX=CNCC(I)*DL
      CMX=CMCG*DL
      CDX=CDCC(I)*DL
      IF(I.NE.NVS) GO TO 100
      CNX=CNX/3.
      CMX=CMX/3.
      CDX=CDX/3.
  100 PCL=PCL+CNX*CDA
      PCM=PCM+CMX
  110 PCD=PCD+CDX
      IF(ABS(PCL).LT.0.000001) PCL=0.000001
      PXC=-PCM/PCL
C*                  PRINT OUTPUT
C
      WRITE (6,300) (I,EVO(I),CNCC(I),CDCC(I),CMCC(I),XCCP(I),I=1,NEE)
  300 FORMAT (1H1,20X,18H ** PYLON LOADS **/1H ,24X,3HETA,12X,6HCNC/CA,
     1       9X,6HCDC/CA,9X,6HCMC/CA,9X,6HX/C CP/(1H ,I15,5F15.5))
C
      WRITE (6,310) PCL,PCD,PCM,PXC
  310 FORMAT (1H0,20X,23H ** TOTAL PYLON LOADS **/1H ,39X,3HPCL,12X,3HPCD
     1       ,12X,3HPCM,12X,3HX/C/1H ,30X,4F15.5)
C
```

```
      RETURN
      END
*DECK SFC
      SUBROUTINE SFC (NETA,ETA)
C
C*                 CALC MATRIX F
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION  W(1)
C
      LARGE      ETA(1)
C
      EQUIVALENCE      (DA(1285),W)     ,(DA(1281),WNW)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CFG/ DF1(10),NKB,NKF
      COMMON/CPW/ DW1(9),NU,NW
      LARGE        CKV(1000) ,BOU(2000) ,FUS(25000),FAN(23700)
     1            ,WIN(23430),PYL(6061) ,CEL(3700)
     2            ,XF(51,10) ,SRE(51)    ,S(51,10)
C
      DO 10  J=1,NETA
   10 SRE(J)=SQRT(1.-ETA(J)**2)
      IF(WNW.NE.0.0) GO TO 500
      K=NKB+NKF
      DO 20 J=1,NU
      DO 20 NE=1,NETA
      K=K+1
   20 XF(NE,J)=CKV(K)/BETA
      GO TO 600
  500 DO 550 I=1,NW
      L=W(I)
      IF(L.LT.1000) GO TO 510
      CALL PFUNC (I,NETA,ETA,W,S)
      GO TO 550
  510 DO 540 J=1,NETA
      IF(ETA(J).GT.0.00001) GO TO 518
      S(J,I)=0.0
      IF(L.EQ.0) S(J,I)=1.0
      GO TO 540
  518 S(J,I)=SRE(J)*ETA(J)**L
  540 CONTINUE
```

```
  550 CONTINUE
C       CALC F MATRIX
  560 DO 575 NE=1,NETA
      K=NKB+NKF
      DO 572 J=1,NU
      XF(NE,J)=0.0
      DO 570 I=1,NW
      K=K+1
  570 XF(NE,J)=XF(NE,J)+CKV(K)*S(NE,I)
  572 XF(NE,J)=XF(NE,J)/BETA
  575 CONTINUE
  600 RETURN
      END
*DECK TPS
      SUBROUTINE TPS (HWS,WT,NEP,EP,TPTE,TPLE)
C
C*    *TPS* CALC. LEADING AND TRAILING EDGE TAN PHI STAR
C
      LARGE      WT(1),EP(1),TPTE(1),TPLE(1)
C
      LARGE      DMI(90000)
     1           ,YB(30)    ,CB(30)    ,PBL(30)    ,PBT(30)
C
C*               CALC. YB,CB AND PHI FROM      PLANFORM
      IC=0
      DO 10 L=1,59,3
      IC=IC+1
      YB(IC)=HWS*WT(L)
      CB(IC)=WT(L+2)-WT(L+1)
      IF(WT(L).GE.1.0) GO TO 20
      DY=HWS*(WT(L+3)-WT(L))
      PBL(IC)=ATAN((WT(L+4)-WT(L+1))/DY)
   10 PBT(IC)=ATAN((WT(L+5)-WT(L+2))/DY)
   20 PBL(IC)=-PBL(IC-1)
      PBT(IC)=-PBT(IC-1)
      NB=IC
C*               CALC. TPLE AND TPTE
      DO 110 IP=1,NEP
      YP=HWS*EP(IP)
      DO 40 IB=2,NB
      IF(YP.LE.YB(IB)) GO TO 50
   40 CONTINUE
   50 IX=IB
```

```
      IF(IX.GT.2) GO TO 60
      APL=0.
      APT=0.
      GO TO 70
   60 APL=(PBL(IX-1)+PBL(IX-2))/2.
      APT=(PBT(IX-1)+PBT(IX-2))/2.
   70 DY=YB(IX)-YB(IX-1)
      IF(CB(IX-1).GE.DY) GO TO 90
      ALL=PBL(IX-1)
      ALT=PBT(IX-1)
      YPC=YB(IX-1)+CB(IX-1)
      IF(YP.GT.YPC) GO TO 80
      C=CB(IX-1)
      YBE=YB(IX-1)
      GO TO 100
   80 C=DY-CB(IX-1)
      YBE=YB(IX)
      APL=(PBL(IX-1)+PBL(IX))/2.
      APT=(PBT(IX-1)+PBT(IX))/2.
      GO TO 100
   90 C=DY
      YBE=YB(IX-1)
      ALL=(PBL(IX-1)+PBL(IX))/2.
      ALT=(PBT(IX-1)+PBT(IX))/2.
  100 AYC=6.283185*ABS((YP-YBE)/C)
      IF(ALL.NE.0.0) GO TO 102
      TLY=AYC
      GO TO 104
  102 TLY=TAN(ALL)/ALL*AYC
  104 SLB=SQRT(1.+TLY**2)-TLY
      CL=COS(ALL)
      CLP=COS(SLB*APL)
      CL2=COS(SLB*ALL)**2
      TPLE(IP)=SQRT(CL2-CL**2*CLP**2)/(CL*CLP)
      IF(ALT.NE.0.0) GO TO 106
      TLY=AYC
      GO TO 108
  106 TLY=TAN(ALT)/ALT*AYC
  108 SLB=SQRT(1.+TLY**2)-TLY
      CL=COS(ALT)
      CLP=COS(SLB*APT)
      CL2=COS(SLB*ALT)**2
  110 TPTE(IP)=SQRT(CL2-CL**2*CLP**2)/(CL*CLP)
```

```
      RETURN
      END
*DECK SPF
      SUBROUTINE SPF
C
C*    *SPF* CALC. THE DERIVATIVES OF F VALUES
C
      COMMON/DAT/ DA(5000)
      DIMENSION   W(10)       ,WD(25)
C
      EQUIVALENCE      (DA(1285),W)     ,(DA(1281),WNW) ,(DA(1930),WD)
     1,(NEP,NJS)
C
      COMMON/CRG/ PI,PI4,RC,BETA
      COMMON/CFG/ DF1(10),NKB,NKF
      COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3
     1            ,NE5,NIB,NU,NW,NVC,NVS,NJC,NJS
C
      LARGE        CKV(1000) ,BOU(2000) ,FUS(25000),FAN(23700)
     1            ,DM1(1706) ,EP(30)     ,DM2(21694),PYL(6061) ,CEL(3700)
     2            ,DM3(510)  ,SRE(51)
     3            ,DM4(3330) ,PF(30,10) ,EPO(30),ES(4),CS(4),S(30,10)
C
C*                CALC SLOPES OF SPANWISE LOAD SHAPES
      IF(WNW.EQ.0.0) GO TO 170
      IC=1
      DO 150 IW=1,NW
      L=W(IW)+1.1
      IF(L.GT.1000) GO TO 30
C
      DO 20 IE=1,NEP
      EPL=EP(IE)**L
      S(IE,IW)=-EPL/SRE(IE)
      IF(IW.EQ.1) GO TO 20
      S(IE,IW)=S(IE,IW)+SRE(IE)*W(IW)*EPO(IE)
   20 EPO(IE)=EPL
      GO TO 150
C
   30 IF(WD(IC).EQ.0.0) GO TO 60
      ES(1)=WD(IC+2)
      ES(2)=WD(IC+3)
      CS(1)=1.
      CS(2)=-1.
```

```
      GO TO 90
   60 ETAB=WD(IC+1)
      RI=WD(IC+2)
      RO=WD(IC+3)
      ES(1)=ETAB-RI
      ES(2)=ETAB
      ES(3)=ETAB+RO
      EB1=1.-ETAB
      ERO=EB1-RO
      RR=(RI+RO)/(RI*RO)
      IF(ES(1).GE.0.0) GO TO 70
      ES(1)=0.0
      ES(4)=0.0
      CS(1)=1./RI
      CS(2)=-RR*EB1
      CS(3)=(RR-CS(1))*ERO
      CS(4)=1.-ETAB/RI
      NP=4
      GO TO 100
   70 IF(ERO.LT.0.0) GO TO 80
      CS(1)=EB1/RI+1.
      CS(2)=-RR*EB1
      CS(3)=EB1/RO-1.
      NP=3
      GO TO 100
   80 CS(1)=EB1/RI+1.
      CS(2)=-CS(1)
   90 NP=2
  100 DO 145 IE=1,NEP
      S(IE,IW)=0.0
      ETA=EP(IE)
      IF(ETA.EQ.0.0) GO TO 145
      ES2=2.*ETA/SRE(IE)
      DO 140 K=1,NP
      AES= ACOS(ES(K))
      SES=SQRT(1.-ES(K)**2)
      IF(SES.EQ.SRE(IE)) SES=SES+.000001
      E1=ES(K)*SRE(IE)
      E2=ETA   *SES
      E=ABS((E1-E2)/(E1+E2))
      IF(K.EQ.4) GO TO 110
      IF(WD(IC).EQ.0.0) GO TO 120
  110 PS=(ALOG(1./E)-ES2*AES)/PI
```

```
      GO TO 140
  120 IF(ABS(ES(K)-ETA).GT.0.000001) GO TO 130
      EX=-ES(K)*ALOG(ES(K)**2)
      GO TO 135
  130 EX=ES(K)*ALOG(E)+ETA*ALOG(ABS((SES+SRE(IE))/(SES-SRE(IE))))
  135 PS=(EX+ES2*(ES(K)*AES-SES))/(PI*(1.-ES(K)))
  140 S(IE,IW)=S(IE,IW)+CS(K)*PS
  145 CONTINUE
      IC=IC+5
  150 CONTINUE
C*                 CALC THE DERIVATIVES OF F VALUES
      DO 160 IE=1,NEP
      K=NKB+NKF
      DO 160 IU=1,NU
      PF(IE,IU)=0.0
      DO 160 IW=1,NW
      K=K+1
  160 PF(IE,IU)=PF(IE,IU)+CKV(K)*S(IE,IW)
      GO TO 220
C                  FIND SLOPE OF COEF. IF SPANWISE DISCRETE VORTEX
  170 DLA=.625*DLO
      DLB=DLA+DLO
      NE7=NE3+4
      NE6=NE3+3
      K=NKB+NKF
      DO 214 IU=1,NU
      K=K+2
      PF(1,IU)=(CKV(K-1)-CKV(K))/DLA
      PF(2,IU)=(CKV(K-1)-CKV(K+1))/DLB
      DO 212 IE=3,NJS
      K=K+1
      IF(IE.NE.NJS) GO TO 200
      PF(IE,IU)=0.0
      GO TO 212
  200 IF(IE-NE6) 210,208,202
  202 IF(IE.NE.NE7) GO TO 204
      PF(IE,IU)=(CKV(K)-CKV(K+1))/DLI
      GO TO 212
  204 PF(IE,IU)=(CKV(K-1)-CKV(K+1))/(DLI+DLI)
      GO TO 212
  208 PF(IE,IU)=(CKV(K-1)-CKV(K))/DLO
      GO TO 212
  210 PF(IE,IU)=(CKV(K-1)-CKV(K+1))/(DLO+DLO)
```

```
  212 CONTINUE
  214 CONTINUE
  220 RETURN
      END
*DECK IDRAG
      SUBROUTINE IDRAG (BCL,FCL,CDI)
C
C*                 CALC INDUCED DRAG
C
      COMMON/DAT/ DA(5000)
C
      EQUIVALENCE      (DA(1),AR)      ,(DA(2),SPAN)
C
      COMMON/CRG/ PI
      COMMON/CFG/ IDB,IDF
      COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3
C
      LARGE       CM1(64496),ET(52)    ,CM2(24712),CNCCO(52) ,CM3(104)
     1            ,CLS(104)   ,SS(104)    ,C(102)     ,CM4(2086)
      LARGE       CM5(10812),EOC(102)  ,Q(102)     ,P(102,102)
C
C*                 INITIAL
      B2=SPAN**2
      RAREA=B2/AR
      HRA = RAREA
C
      NVT=NE2
      IF(IDB.NE.0) NVT=NVT+1
      IF(IDF.NE.0) NVT=NVT+1
      IV=NVT
      NEF=NE2-NE3-2
C*                 SETUP SECTION LOADS AND ABSCCISA ON LEFT SIDE
      DO 60 I=1,NE2
      IF(IDB.EQ.0) GO TO 10
      IF(I.NE.1) GO TO 10   CLS(IV)= BCL * RAREA
      IV=IV-1
C
   10 IF(IDF.EQ.0) GO TO 40
      IF(I-NEF) 20,30,40    20 CLS(IV)= CNCCNΦ(I) * DLI * HRA
      GO TO 50
C                           30 CLS(IV)= FCL* HRA
      IV=IV-1
C                           40 CLS(IV)= CNCCΦ(I)*DLΦ * HRA
      IF(I.EQ.NE2) CLS(IV)=CLS(IV)/3.
```

```
C
   50 IV=IV-1
   60 CONTINUE
C
      NVT=NVT-1
      DO 70 I=1,NVT
   70 EOC(I)=(1.-ET(I))/2.
C*                 SETUP SECTION LOADS AND ABSCCISA ON RIGHT SIDE
      NN=NVT+NVT
      IV=NVT
      IS=NVT+1
      DO 80 I=IS,NN
      EOC(I)=1.-EOC(IV)
      CLS(I+1)=CLS(IV)
   80 IV=IV-1
C*                 CALC EMINTON-LORD Q AND P MATRICES
      CALL EMLQP (NN)
C*                 INTEGRATE SECTION LOADS CURVE
      NX=NN+1
      SS(1)=CLS(1)
      DO 90 I=2,NX
   90 SS(I)=SS(I-1)+CLS(I)
      DSS=SS(NX)
C*                 CALC INDUCED DRAG COEF
      DO 100 N=1,NN
  100 C(N)=SS(N)-DSS*Q(N)
C
      SUMB=0.0
      DO 120 M=1,NN
      SUMC=0.0
      DO 110 N=1,NN
  110 SUMC=SUMC+P(M,N)*C(N)
  120 SUMB=SUMB+SUMC*C(M)
C
      CDI=(4.*DSS**2/PI+SUMB*PI)/B2*.25/RAREA
C
      RETURN
      END
*DECK EMLQP
      SUBROUTINE EMLQP(NN)
C
C    *             COMPUTES COEFFICIENTS FOR EMLORD EQUATION
C
```

```
      COMMON/CRG/ PI
C
      LARGE        DUM(102624)
      LARGE        XOC(102)    ,Q(102)     ,P(102,102),B(102,102)
C
C     *            START
C     *            SETUP MATRIES P AND B
      DO 100 N=1,NN
      X=XOC(N)
      XV=1.-2.*X
      ANG=ATAN2(SQRT(1.-XV**2),XV)
      Q(N)=(ANG-2.*XV*SQRT(X-X**2))/PI
      DO 70 M=N,NN
      Y=XOC(M)
      IF(M-N) 30,20,30
   20 B(M,N)=1.0
      GO TO 40
   30 B(M,N)=0.0
   40 E=(X-Y)**2
      E1=X+Y-2.*X*Y
      E2=2.*SQRT(X*Y*(1.-X)*(1.-Y))
      IF(E) 60,50,60
   50 P(M,N)=E1*E2
      GO TO 70
   60 P(M,N)=E/2.*ALOG((E1-E2)/(E1+E2))+E1*E2
   70 CONTINUE
      NK=N-1
      IF(NK) 100,100,80
   80 DO 90 M=1,NK
      E=P(N,M)
      P(M,N)=E
   90 B(M,N)=0.0
  100 CONTINUE
C     *            SOLVE SIMULTANEOUS EQUATIONS FOR COEFFICIENTS
      CALL MSOLX (NN,NN,NN,P,B,102)
      RETURN
      END
*DECK SKINF
      SUBROUTINE SKINF
C
C*                 CALC SKIN FRICTION DRAG
C
      COMMON/DAT/ DA(5000)
```

```
C
      DIMENSION  XTRL(1),TCM(1)
C
      EQUIVALENCE      (DA(1),AR)        ,(DA(2),SPAN)    ,(DA(4),XMACH)
     1,(DA(4980),TFS) ,(DA(4981),PFS)   ,(DA(4982),CKAF),(DA(4983),TRID)
     2,(DA(4984),XKS) ,(DA(4985),TAU)   ,(DA(4990),XTRL),(DA(4995),TCM)
     3,(DA(5),TLF)    ,(DA(4005),TLB)   ,(DA(4986),CONV)
C
C
      COMMON/CSF/ CID(5),NS(5),SA(5),TLW(52),TLP(22),TLN
C                 TABLE DATA
      DIMENSION   TAUI(10)  ,DRTI(10)  ,RKI(10)   ,RXTRI(10)
C
      DATA     TAUI/ .001      ,.002      ,.003      ,.004
     1,.008     ,.012      ,.016      ,.020      ,.024      ,.028/
C
      DATA     DRTI/ 950.      ,720.      ,590.      ,525.
     1,365.     ,230.      ,115.      ,55.       ,25.       ,10./
C
      DATA     RKI / 0.0       ,50.       ,100.      ,120.
     1,150.     ,175.      ,200.      ,250.      ,300.      ,350./
C
      DATA   RXTRI/ 670000.  ,670000.   ,670000.   ,660000.
     1,640000.  ,570000.   ,460000.   ,280000.   ,140000.   ,50000./
C
C*               INITIAL
      IF(TFS.EQ.0.0) GO TO 400
      CALL SETSF
      SREF=SPAN**2/AR
C
C*               FREESTREAM PARAMETERS
   20 RTFS=1716.*TFS
      RHO=PFS/RTFS
      XM=XMACH
      IF(XM.EQ.0.0) XM=.05
      UFS=SQRT(1.4*RTFS)*XM
      XMU=2.270E-8*TFS*SQRT(TFS)/(TFS+198.6)
      R=RHO*UFS/XMU
      GM1=.2*XM
      GM2=GM1*XM
C*               TURBULENT PARAMETERS
      TRTT=1.+.88*GM1
      A2=GM2/TRTT
```

```
      B=(1.+GM2)/TRTT-1.
      DENO=SQRT(B**2+4.*A2)
      ALP=(2.*A2-B)/DENO
      BET=B/DENO
      C1=.242*( ASIN(ALP)+ ASIN(BET))/SQRT(GM2)
      C2=.76*ALOG10(TRTT)
C*             LAMINAR PARAMETERS
      TRTL=1.+.851*GM1
      TST=1.+.72*(TRTL-1.)
      TS=TST*TFS
      XMUS=2.270E-8*TS*SQRT(TS)/(TS+198.6)
      CS=XMUS/XMU/TST
      SRCS=SQRT(CS)
C*             FLAT PLATE NATURAL TRANSITION POINT
      IF(TRID.LT.0.0) GO TO 60
C              SMOOTH
      CALL COD (10,TAUI,DRTI,1,TAU,DRT)
      RTTR=163.+DRT
      RXTR=(RTTR/.664)**2/CS
      IF(XKS.EQ.0.0) GO TO 60
C              ROUGH
      RK=R*XKS
      CALL COD (10,RKI,RXTRI,1,RK,RXTX)
      RXTR=RXTR*RXTX/670000.
      RTTR=.664*SQRT(RXTR)*SRCS
C
C*             NO. OF COMPONENTS LOOP
   60 SAT=0.0
      CFT=0.0
      DO 300 I=1,5
      IF(CID(I).EQ.0.0) GO TO 300
C*             CALC FORM CORRECTION AND AREA RATIO
      XK=1.0
      IF(CID(I)-2.) 80,70,75
   70 XK=1.+CKAF*TCM(I)+60.*TCM(I)**4
      GO TO 80
   75 XK=1.+1.5*TCM(I)*SQRT(TCM(I))+7.*TCM(I)**3
   80 SRK=SA(I)/SREF*XK
C
C*             NO. OF SEGMENTS PER COMPONENT
      IF(I.EQ.1) TL=TLB
      IF(I.EQ.2) TL=TLF
      IF(I.EQ.5) TL=TLN
```

```
      NSC=NS(I)
      DO 290 IS=1,NSC
      IF(I.EQ.3) TL=TLW(IS)
      IF(I.EQ.4) TL=TLP(IS)
      TL=TL/CONV
      IF(TRID.GE.0.0) GO TO 90
      RXTR=R*XTRL(I)*TL
      RTTR=.664*SQRT(RXTR)*SRCS
C
   90 XTR=RXTR/R
      CFRDX=2.*RTTR
C*                 ITERATE FOR CFS (SMOOTH)
      FP=0.0
      FM=0.0
      IP=1
      CF=.001
      CO=100.
C
      SL=TL
      IF(CFRDX.LE.0.0) GO TO 95
      CFDO=(C1/(ALOG10(CFRDX)-C2))**2
      RDX=CFRDX/CFDO
      DX=RDX/R
      SL=TL-XTR+DX
   95 RLS=R*SL
  100 IF(ABS(1.-CF/CO).LT.0.001) GO TO 170
      F=ALOG10(CF*RLS)-C1/SQRT(CF)-C2
      IF(F) 110,170,120
  110 FM=F
      CM=CF
      GO TO 130
  120 FP=F
      CP=CF
  130 IF(IP.EQ.0) GO TO 140
      IP=0
      CF=.007
      GO TO 100
  140 IF((FP*FM).LT.0.0) GO TO 160
      WRITE (6,150) I
  150 FORMAT (51H0 ** SKINF * NO SOLUTION WITHIN CFS LIMITS * COMP.=,I5)
      CFS=0.0
      GO TO 180
```

```
  160 CO=CF
      CF=(CM+CP)/2.
      GO TO 100
C
  170 CFS=CF*SL/TL*SRK
      CFR=0.0
  180 IF(XKS.EQ.0.0) GO TO 280
C*                ITERATE FOR CFR (ROUGH)
      DX=0.0
      IF((XTR/TL    ).LT.0.0001) GO TO 270
      IP=1
      DX=XTR/2.0
C
  200 F1=1.89+1.62*ALOG10(DX/XKS)
      F2=SQRT(F1)
      CF=1./(TRTT*F2*F1**2)
      DXO=DX
      RDX=CFRDX/CF
      DX=RDX/R
      IF(ABS(1.-DX/DXO).LT.0.001) GO TO 270
      IP=IP+1
      IF(IP.LE.100) GO TO 200
      WRITE (6,250) I
  250 FORMAT (51H0 ** SKINF * NO SOLUTION WITHIN CFR LIMITS * COMP.=,I5)
      CFR=0.0
      GO TO 280
C
  270 SL=TL-XTR+DX
      F1=1.89+1.62*ALOG10(SL/XKS)
      F2=SQRT(F1)
      CF=1./(TRTT*F2*F1**2)
      CFR=CF*SL/TL*SRK
C*                SUM SKIN FRICTION DRAG
  280 CFU=CFR
      IF(CFR.LT.CFS) CFU=CFS
      CFT=CFT+CFU
  290 CONTINUE
      SAT=SAT+SA(I)
C
  300 CONTINUE
C
      WRITE (6,310) XMACH,TFS,PFS,XKS,TAU,SREF,SAT,CFT
  310 FORMAT (23H1 ** SKIN FRICTION DRAG/1H0,4X,5HMACH=,F7.3,4X,5HTEMP=,
```

```
     1    F9.3,4X,6HPRESS=,F10.3,4X,3HKS=,F9.6,4X,4HTAU=,F7.4/
     2     19H0 * REFERENCE AREA=,F10.3/21H0 * WET SURFACE AREA=,F10.3/
     3     29H0 * TOTAL SKIN FRICTION DRAG=,F10.5)
C
  400 RETURN
      END
*DECK SETSF
      SUBROUTINE SETSF
C
C*                 SETUP SKIN FRICTION DRAG INPUT DATA
C
      COMMON/DAT/ DA(5000)
C
      DIMENSION  XB1(1),YB1(1)
C
      EQUIVALENCE       (DA(2),WS)        ,(DA(19),PVPC)   ,(DA(2500),PHT)
     1,(DA(2502),POY) ,(DA(3392),YNO)   ,(DA(3400),XB1)  ,(DA(3550),YB1)
     2,(DA(2492),PDA)
C
      COMMON/CFG/ IDB,IDF,IDW,IDP,IDN
      COMMON/CPB/ DB1(12),SAB
      COMMON/CPF/ DF1(12),SAF
      COMMON/CPW/ DW1(3),DLI,DLO,NE2,NE3,DW2(5),NVSW
      COMMON/CPP/ DP1(3),DLP,DP2(4),NVSP
      COMMON/CPN/ DN1,NP1
      COMMON/CSF/ CID(5),NS(5),SA(5),TLW(52),TLP(22),TLN
C
      LARGE       DM1(51804) ,XTLVW(52) ,DM2(12744),XTLTW(52)
     1            ,DM3(10522),XTLVP(22)
C
C                 FUSELAGE SETUP
      CID(1)=0.0
      IF(IDB.EQ.0) GO TO 10
      CID(1)=3.0
      NS(1)=1
      SA(1)=SAB
C                 FANPOD SETUP
   10 CID(2)=0.0
      IF(IDF.EQ.0) GO TO 20
      CID(2)=3.0
      NS(2)=1
      SA(2)=SAF
C                 WING SETUP
```

```
   20 CID(3)=0.0
      IF(IDW.EQ.0) GO TO 40
      CID(3)=2.0
      NS(3)=NE2
      NE6=NE3+3
      NE1=NE2+1
      NVSH=NVSW/2
      TWS=WS+WS
      DS4=TWS*DLO
      TLW(1)=(XTLVW(NE1)+XTLTW(1))/2.0
      SA(3)=TLW(1)*DS4/2.0
      DO 30 I=2,NVSH
      IF(I.GT.NE6) DS4=TWS*DLI
      TLW(I)=XTLVW(I)
      SA(3)=SA(3)+TLW(I)*DS4
   30 CONTINUE
      IF(IDB.NE.0) GO TO 40
      TLW(NE2)=(XTLVW(NE2)+XTLTW(NE2))/2.0
      SA(3)=SA(3)+TLW(NE2)*DS4/2.0
C                 PYLON SETUP
   40 CID(4)=0.0
      IF(IDP.EQ.0) GO TO 60
      CID(4)=2.0
      NVPC=PVPC
      NS(4)=NVSP-NVPC
      DS4=4.*PHT*DLP
      NSX=NVPC+1
      IC=0
      SA(4)=0.0
      DO 50 I=NSX,NVSP
      IC=IC+1
      TLP(IC)=XTLVP(I)
      SA(4)=SA(4)+TLP(IC)*DS4
   50 CONTINUE
      IF(POY.EQ.0.0.AND.PDA.EQ.0.0) SA(4)=SA(4)/2.0
C                 NACELLE SETUP
   60 CID(5)=0.0
      IF(IDN.EQ.0) GO TO 70
      CID(5)=2.0
      NS(5)=1
      TLN=XB1(1)
      IF(XB1(NP1).GT.XB1(1)) TLN=XB1(NP1)
      SA(5)=12.56637*YB1(1)*TLN
```

```
      IF(YNO.EQ.0.0.AND.PDA.EQ.0.0) SA(5)=SA(5)/2.0
C
   70 RETURN
      END
*DECK COD
      SUBROUTINE COD (NI,XI,YI,NA,XO,ANS)
C
C****               A CONTROLLED DEVIATION ITERPOLATION METHOD
C
      DIMENSION  XI(1)     ,YI(1)     ,XO(1)     ,ANS(1)
C
      DATA XK/0.5/
C
      N=NI
      DO 910 IE=1,NA
      X=XO(IE)
  100 IF(N-2)110,120,200
  110 Y = YI(N)
      GO TO 900
  120 Y = (YI(2)-YI(1))/(XI(2)-XI(1))* (X-XI(1)) +YI(1)
      GO TO 900
  200 J = 1
  210 IF(XI(J)-X)230,220,250
  220 Y =YI(J)
      GO TO 900
  230 J = J+1
      IF(J-N)210,210,250
  250 IF(J-2)220,155,260
  155 J = 3
      JJ = 1
      GO TO 285
  260 IF(J-N)280,265,110
  265 J = N-1
      JJ = 2
      GO TO 285
  280 JJ = 3
  285 IF(N-3)290,290,295
  290 J = 3
  295 K = J-1
      M = K-1
      L = J+1
      A1 = X-XI(M)
      A2 = X-XI(K)
```

```
      A3 = X-XI(J)
      AL = (X-XI(K))/(XI(J)-XI(K))
      S = AL*YI(J)+(1.0-AL)*YI(K)
      C1= A3*A2/((XI(M)-XI(K))*(XI(M)-XI(J)))
      C2= A1*A3/((XI(K)-XI(M))*(XI(K)-XI(J)))
      C3= A2*A1/((XI(J)-XI(M))*(XI(J)-XI(K)))
      P1 = C1*YI(M)+C2*YI(K)+C3*YI(J)
      IF(N-3)305,305,310
  305 P2 = P1
      GO TO 315
  310 A4 = X-XI(L)
      C4= A4*A3/((XI(K)-XI(J))*(XI(K)-XI(L)))
      C5= A2*A4/((XI(J)-XI(K))*(XI(J)-XI(L)))
      C6= A3*A2/((XI(L)-XI(K))*(XI(L)-XI(J)))
      P2 = C4*YI(K)+C5*YI(J)+C6*YI(L)
  315 GO TO (320,330,350),JJ
  320 P2 = P1
      AL = (X-XI(1))/(XI(2)-XI(1))
      S =  AL*YI(2)+ (1.0-AL)*YI(1)
      P1=  S + XK*(P2-S)
      GO TO 350
  330 P1 = P2
      AL = (X-XI(N-1))/(XI(N)-XI(N-1))
      S  = AL*YI(N) +(1.0-AL)*YI(N-1)
      P2 = S+ XK*(P1-S)
  350 E1 =  ABS(P1-S)
      E2 =  ABS(P2-S)
      IF(E1+E2)400,400,410
  400 Y = S
      GO TO 900
  410 BT =  (E1*AL)/(E1*AL+(1.0-AL)*E2)
      Y  = BT*P2+(1.0-BT)*P1
  900 ANS(IE)=Y
  910 CONTINUE
      RETURN
      END
*
GROUP=121,AERO
OWNER=R.C.SMITH
ACCOUNT=482511
WRITE LIBRARY=A04RCS
REPLACE SUBSET=UPDATPL
COPY 1 F FROM FILE=NEWPL
```

```
REPLACE SUBSET=OBJECT
COPY 1 F FROM FILE=LGO
END.
```