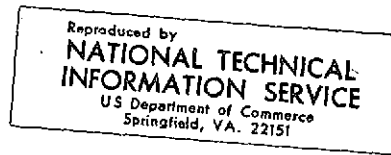


THEORETICAL PREDICTION OF THICK WING AND
PYLON-FUSELAGE-FANPOD-NACELLE
AERODYNAMIC CHARACTERISTICS
AT SUBCRITICAL SPEEDS

PART II - COMPUTER PROGRAM DESCRIPTION

By J. Kojima and J.R. Tulinius

July 24, 1974



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

~~PRICES SUBJECT TO CHANGE~~

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 N75-11936
OF THICK WING AND
PYLON-FUSELAGE-FANPOD-NACELLE AERODYNAMIC
(Rockwell International Corp., Los Angeles) 340 p HC \$9.50 CSCL 01C G3/05 Unclass 04298

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

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

Prepared under Contract No. 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

SUMMARY	1
INTRODUCTION	5
PROGRAM SETUP	6
DESCRIPTION OF SUBROUTINES	11
DESCRIPTION OF INPUT DATA	15
SAMPLE INPUT DATA SHEETS	39
SAMPLE INPUT DATA	79
DESCRIPTION OF OUTPUT	89
SAMPLE OUTPUT	91
LISTING OF PROGRAM	134

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.

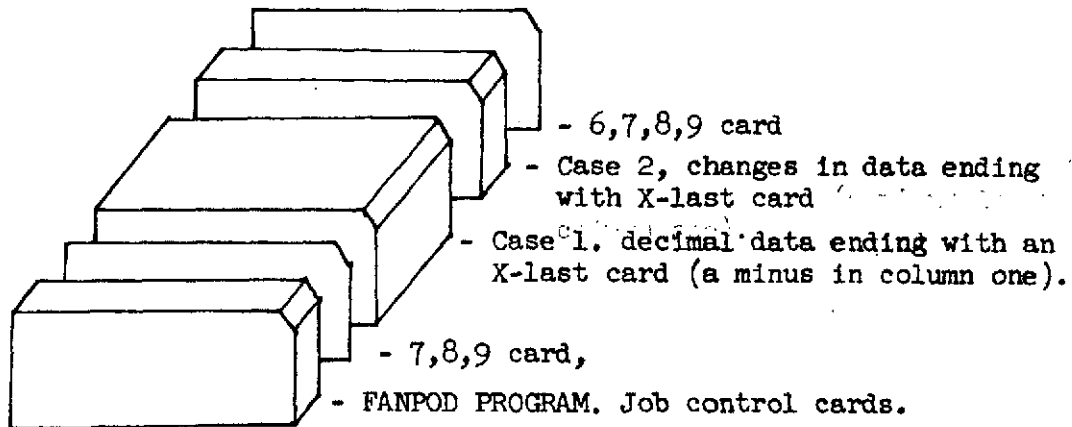
CASE II:

- 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 LG0, 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 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(135000,50000)

LGO,LC=10000.

(7-8-9) CARD

•
•
•
•
•
•
•

(INPUT DATA)

10

•
•
•
•
•
•

(6-7-8-9) CARD

DESCRIPTION OF SUBROUTINES

NUMBER	SYMBOL	DESCRIPTION
1	MAIN	Main program controls general flow of program.
2	CODIM	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	MSOLX	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	GEOM	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 COS 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	C O DMT	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	AQVWV	Calculate matrix A with Q on wing and vortices on wing.
32	AQVWVF	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	AQFVP	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/ϕ 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	SϕLU	Control program for matrix solution routines.
50	MSϕL	Special Householder method for solving a set of linear simultaneous equations with matrix A on various I/ϕ units.
51	MSϕLP	Special Householder method for solving a set of linear simultaneous equations with the results save on various I/ϕ 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 ϕ^* .
62	SPF	Calculate the derivatives of the wing F matrix.
63	BL OAD	Calculate the load coefficients for the fuselage.
64	FL OAD	Calculate the load coefficients for the fanpod.
65	WL OAD	Calculate the load coefficients for the wing.
66	PL OAD	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	C OD	Special C OD DIM.

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 \perp to the camber line described in loc. 670-704 or \perp to the fanpod X axis. Use 0.0 if sections are to be placed \perp to the camber line and 1.0 if they are to be placed \perp to the X axis.
- 17 FLØI This input indicates the type of vortex grid input to be used to describe the fanpod 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 ϕ or X, input a 1.0 here and list the X stations in loc. 1460-1609.
- 18 FTTHI 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 FNKS Input the number of chordwise stations where the fanpod cross sections are to be described. The maximum number is 29.
- 21-49 XS₁, XS₂, ... 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	$\Theta_1, \Theta_2, \dots$	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	F ϕ X	X location of the nose of the starboard fanpod.
88	F ϕ Y	Y location of the nose of the starboard fanpod.
89	F ϕ Z	Z location of the nose of the starboard fanpod.
90-564	R ₁ , R ₂ , ...	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	FNDKM	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 axes 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 , ...	List of chordwise stations where the fanpod multiplication factors and camber are defined.
600-634	YMC ₁ , YMC ₂ , ...	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, \dots	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, \dots	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	$X/C_1, X/C_2, \dots$	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	η_1, η_2, \dots	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 n 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 $s\eta$ 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 $s\eta_0 = 2(1 - \eta_{01}) / (WNVS\phi - 1)$, 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, $\eta_{01} + s\eta_0$ in loc. 1213, $\eta_{01} + 2s\eta_0$ in loc. 1216, and $\eta_{01} + 3s\eta_0$ in loc. 1019. The X distances of the wing leading edge from the wing apex at η_{01} , $\eta_{01} + s\eta_0$, $\eta_{01} + 2s\eta_0$, and $\eta_{01} + 3s\eta_0$ 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 $\eta_{01} + 3s\eta_0$ 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 $\eta_{01} + 3s\eta_0$. 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 \mathcal{Z}_{01} to $\mathcal{Z}_{01} + 3\mathcal{M}_0$, 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 $\mathcal{M}_1 = 2\mathcal{Z}_N / (\text{WNVSI} - 1)$, where \mathcal{Z}_N is the \mathcal{Z} 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 $\mathcal{Z}_N - 3\mathcal{M}_1$. At $\mathcal{Z} = \mathcal{Z}_N - 3\mathcal{M}_1$ the X distances from the wing apex to the leading and trailing edges are input. At $\mathcal{Z} = \mathcal{Z}_N - 2\mathcal{M}_1$ and $\mathcal{Z} = \mathcal{Z}_N - \mathcal{M}_1$ 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'_{01} and S'_{02} for the outboard fanpod fillet. If there are no fanpod fillets on the wing trailing edge the S' values are equal to zero. At \mathcal{Z}_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 | WNVV | 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 ζ 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 $WNJC = 2 \text{ WNU}$.
- 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 $WNJS = 2 \text{ WNW}$. 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 = 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 $\text{COT}^e/2$, SIN^e , $\text{SIN}2^e, \dots$ 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 $r^W \sqrt{1-r^2}$, 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, \dots

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, \dots

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	$\cot \phi / 2$	$(x/c)_1$
2.0	$\cot(90-\phi/2)$	$(x/c)_2$
3.0	$\sin \phi$	$(x/c)_3$
4.0	$\cos \phi$	$(x/c)_4$
5.0	$\sin 2\phi$	$(x/c)_5$
6.0	$\cos 2\phi$	$(x/c)_6$
.	.	.
.	.	.
.	.	.

1460-1609 FVX_1, FVX_2, \dots

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, \dots

This is the list of lateral vortex grid stations. This list of \ominus 's refer to the fanpod body before the multiplication factors and camber are applied. The \ominus 's are input from 0.0 to sum value less than 360 degrees. The first \ominus must be zero. The maximum number is 40.

1660-1689 WJC_1, WJC_2, \dots

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 WJS₁,WJS₂,...

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 $(WNVS\phi+WNVSI+1)/2$. This input is unnecessary if a discrete solution is done in the spanwise direction on the wing.

1720-1869 FJX₁,FJX₂,...

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 WPC₁,WPC₂,...

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 WPS₁,WPS₂,...

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 ζ 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_0 , Y_0	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 ζ 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, ...	Wing twist table. Wing twist in radians is input for the stations listed in loc. 731-745.
1985-1999	$\Delta ZLE_1, \Delta ZLE_2, \dots$	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	$(X/C)_1, (X/C)_2, \dots$	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	η_1, η_2, \dots	List of η stations where wing thickness is described.
2050-2499	$(Z/C)_1, (Z/C)_2, \dots$	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 , ...	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 , ...	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 ζ stations on the pylon at which the pylon camber and twist are described. The maximum number is 19.
2551-2569	ζ_1, ζ_2, \dots	The list of ζ 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 , ...	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 ζ stations given at loc. 2531-2549 and loc. 2551-2569, respectively. The camber is input for the first ζ station and all of the (X/C)'s, then the second ζ station and all of the (X/C)'s, etc. This table is unnecessary if wing is flat.
2940-2959	$\epsilon_1, \epsilon_2, \dots$	Pylon twist table. The pylon twist is given in radians at the ζ stations listed at loc. 2551-2569.

2960-2965	ζ, 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 $P\phi X$. The pylon ζ 's are relative to $P\phi Z$ and equal to Y_p/PHT .
3000	NXC	Number of X/C stations on the pylon where the pylon thickness is defined. The maximum number is 19.
3001-3019	$(X/C)_1, (X/C)_2, \dots$	List of X/C stations on the pylon where the pylon thickness is defined.
3020	NETA	Number of ζ stations on the pylon where the pylon thickness is defined. The maximum number is 18.
3021-3039	ζ_1, ζ_2, \dots	List of ζ stations on the pylon where the pylon thickness is defined.
3040-3369	$(Z/C)_1, (Z/C)_2, \dots$	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 ζ station and all of the X/C stations, then the second ζ , 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 range 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	XNP2	Number of points used to define second nacelle body contour. Maximum number is XNP2=140-XNP1.
3400-3549	XB1 ₁ ,XB1 ₂ ,...	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 ,...	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 ,...	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 ,...	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 \perp to the camber line described in loc. 4670-4704 or \perp to the fuselage X axis. Use 0.0 if sections are to be placed \perp to the camber line and 1.0 if they are to be placed to the X axis.
4009	BL \emptyset I	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 \emptyset 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 \emptyset , 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 , ...	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	$\Theta_1, \Theta_2, \dots$	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 , ...	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 axes 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 , ...	List of chordwise stations where the fuselage multiplication factors and camber are defined.
4600-4634	YMC ₁ , YMC ₂ , ...	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 , ...	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

F₁, F₂, ...

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	COT ϕ	(x/c) ₁
2.0	COT(90- ϕ /2)	(x/c) ₂
3.0	SIN ϕ	(x/c) ₃
4.0	COS ϕ	(x/c) ₄
5.0	SIN2 ϕ	(x/c) ₅
6.0	COS2 ϕ	(x/c) ₆
.	.	.
.	.	.
.	.	.

4735-4884

BVX₁, BVX₂, ...

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

BVT₁, BVT₂, ...

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

BJX₁, BJB₂, ...

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 ...	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 ,...	The maximum thickness over length (t/c) for the fuselage, fanpod, wing, pylon and nacelle respectively.

SAMPLE INPUT DATA SHEETS

FORTRAN FIXED 16 DIGIT DECIMAL DATA

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

NUMBER	IDENTIFICATION	DESCRIPTION	DO NOT KEY PUNCH	FANPOD PROGRAM BODY-FANPOD-WING-PYLON-NACELLE
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
61			WAAI ~ { 0.0 WING IS FLAT 1.0 WING HAS CAMBER OR TWIST.	
13	10			
25			WADI ~ WING CAMBER TABLE INDICATOR { 0.0 IF LOCAL ANGLE OF ATTACK IN RADIANS, 1.0 IF DEFLECTIONS IN Z/C.	
37			NOT USED	
49			PRII ~ PRINT INDICATOR { 0.0 - PRINT MATRIX B. 1.0 PRINT MATRIX B AND LEAST SQ. B.	
61		73	80	NOT USED
1	15			
13			FMFI ~ FANPOD MULTI. FACTOR INDICATOR { 0.0 Ym=Zm (Z CAMBER), -2.0 Ym=Zm (Y CAMBER) 1.0 Ym≠Zm (" "), -1.0 Ym≠Zm (" ")	
25			FCPI ~ FANPOD CAMBER INDICATOR { 0.0 FANPOD SECTION PUT ⊥ TO CAMBER LINE, 1.0 " " " ⊥ " XF AXIS.	
37			FLΦI ~ FANPOD CHORDWISE VORTEX GRID INDI. { 0.0 GRID AT EQUAL ΔΦ WHERE Φ = COS ⁻¹ (1-2X/L) 1.0 GIVEN IN L&C. 1480-1609. -1.0 EVEN ΔX/C	
49		73	80	FTHI ~ FANPOD LATERAL VORTEX GRID INDI. { 0.0 GRID AT EQUAL θ, 1.0 GIVEN θ AT L&C. 1610-1659.
61			PVPC ~ PIVS MINUS NUMBER OF SPAN STATIONS WHERE PRESSURES ARE TO BE COMPUTED ON THE PYLON.	

FORTRAN FIXED 16 DIGIT DECIMAL DATA

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

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.
13	10	
25		WADI ~ WING CAMBER TABLE INDICATOR { 0.0 IF LOCAL ANGLE OF ATTACK IN RADIANS, 1.0 IF DEFLECTIONS IN $\frac{1}{2}^\circ$.
37		NOT USED
49	73	80 PRIL ~ PRINT INDICATOR { 0.0 - PRINT MATRIX B. 1.0 PRINT MATRIX B AND LEAST SQ. B.
61		NOT USED
1	15	
13		FMFI ~ FANPOD MULTI-FACOR INDICATOR { 0.0 Ym=Zm (Z CAMBER), -2.0 Ym=Zm (Y CAMBER) 1.0 Ym=Zm (" "), -1.0 Ym=Zm (" ")
25		FCPI ~ FANPOD CAMBER INDICATOR { 0.0 FANPOD SECTION PUT \perp TO CAMBER LINE. 1.0 " " " \perp " X \bar{c} AXIS.
37		FLOI ~ FANPOD CHORDWISE VORTEX GRID INDI. { 0.0 GRID AT EQUAL $\Delta\phi$ WHERE $\phi = \cos^{-1}(1-2X/L)$ 1.0 GIVEN IN LOC. 1480-1609. -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 ~ PNV3 MINUS NUMBER OF SPAN STATIONS WHERE PRESSURES ARE TO BE COMPUTED ON THE PYLON.

FORTRAN FIXED 10 DIGIT DECIMAL DATA

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

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	20	FANPOD COORDINATE TABLE
13		FNXS ~ No. of X (CROSS-SECTION) IN LIST BELOW. (MAX. 29)
25		$X_{S_1} \quad X_{S_1} < X_{S_2} < X_{S_3} \dots$ (RELATIVE TO FANPOD ORIGIN.)
37		X_{S_2}
49	73	X_{S_3}
61	80	
1	25	<div style="display: flex; align-items: center;"> <div style="margin-right: 20px;"> <p>(0,0,0)</p> </div> <div> <p>(F_{OX}, F_{OY}, F_{OZ})</p> </div> </div>
13		
25		
37		
49	73	NOTE: FNXS * FNTY < 475
61	80	
1	50	FNTY ~ No. of θ AT EACH CROSS-SECTION IN LIST BELOW. (MAX. 36)
13		$\theta_1 \quad \theta_1 < \theta_2 < \theta_3 \dots$ ($0 = 0.0 \rightarrow 360$. IN DEG.)
25		θ_2
37		θ_3
49	73	
61	80	
1	87	<div style="display: flex; align-items: center;"> <div style="margin-right: 20px;"> <p>F_{OX} ~ FANPOD ORIGIN X</p> <p>F_{OY} ~ " " Y</p> <p>F_{OZ} ~ " " Z</p> </div> <div> <p>} RELATIVE TO ORIGIN (0,0,0)</p> </div> </div>
13		
25		
37		
49	73	
61	80	

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 (CONT.)
13		R ₁ LIST OF RADIUS
25		R ₂
37		R ₃
49	73 80	FOR X _{S1} AT EACH θ
61		.
1	- 95	
13		.
25		.
37		
49	73 80	R _{NTY+1} NOTE: NTY = No. OF θ IN LOC. 50.
61		R _{NTY+2}
		R _{NTY+3}
1	1.00	
13		FOR X _{S2} AT EACH θ
25		.
37		.
49	73 80	.
61		R _{NTY+NTY+1}
1	1.05	
13		R _{NTY+NTY+2}
25		FOR X _{S3} 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	73 .80	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. XMC ₃ XMC ₁ < XMC ₂ < XMC ₃ < . . .
13		
25		
37		
49		
61		
1	73 .80	-570 . . NOTE: IF FANPOD HAS NO CAMBER AND THE MULT. FACTORS ARE NOT READ, SET FNXM=0.0 XMC _{NXM}
13		
25		
37		
49		
61		
1	73 .80	600 YM ₁ LIST OF MULT. FACTORS YM YM ₂ YM ₃ . . YM _{NXM}
13		
25		
37		
49		
61		
1	73 .80	605 . . YM _{NXM}
13		
25		
37		
49		
61		

FORTRAN FIXED 10 DIGIT DECIMAL DATA

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

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH	
1	635	FANPOD MULT. FACTOR AND CAMBER TABLE (CON'T.)	
13		ZM ₁ LIST OF MULT. FACTORS ZM	
25		ZM ₂	
37		ZM ₃ NOTE 1	
49		73 80	LIST OF ZM IS NOT REQUIRED, IF Y _M = Z _M
61			SET FMFI = 0.0 (LOC. 15)
1	-640		
13			
25			
37			
49		73 80	ZM _{NXM}
61			
1	670		
13			
25			
37			
49		73 80	FZC ₁ LIST OF FANPOD CAMBER (Z)
61			FZC ₂
1	675		
13			
25			
37			
49		73 80	FZC ₃
61			FZC _{NXM}

FORTRAN FIXED 10 DIGIT DECIMAL DATA

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

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	7.05		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
37			X/C ₂
49		73 80	X/C ₃
61			.
1	7.10		
13			.
25			NOTE:
37			THE WING CAMBER TABLE IS NOT
49			REQUIRED IF WAAI = 0.0 (LOC. 9)
61			X/C _{NX}
1	7.30		
13			WNE ~ NO. OF η IN LIST BELOW. (MAX. 15)
25			η_1 $\eta = \frac{X}{272}$
37			η_2 $\eta_1 < \eta_2 < \eta_3$
49		73 80	η_3
61			.
1	7.35		
13			.
25			NOTE:
37			THIS LIST OF η 's ARE REQ'D ALSO FOR TWIST
49			IN LOC. 1970 → 1984 AND AZLE IN LOC. 1985 → 1999.
61			η_{NE}

FORTRAN FIXED 10 DIGIT DECIMAL DATA

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

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	7.50	WING CAMBER TABLE (CON'T.)
13		
25		WAD ₁ } LIST OF;
37		WAD ₂ } 1.) IF WADI=0.0 (LOC. 10), ANGLE OF ATTACKS (α IN RAD.).
49		WAD ₃ } 2.) IF WADI=1.0 (LOC. 10), DEFLECTIONS (Z/C).
61	73 80	•
1		• } FOR η ₁ AT EACH X/C
1	- 7.55	
13		
25		
37		
49		WAD _{NX+1} }
61	73 80	WAD _{NX+2}
94		WAD _{NX+3}
1	7.60	
13		• } FOR η ₂ AT EACH X/C
25		•
37		•
49		•
61	73 80	•
		NOTE:
		NX=NO. OF X/C IN LOC. 705
		WAD _{NX+NX+1} }
1	7.65	
13		
25		• } FOR η ₃ AT EACH X/C
37		•
49		•
61	73 80	•

FORTRAN FIXED 10 DIGIT DECIMAL DATA

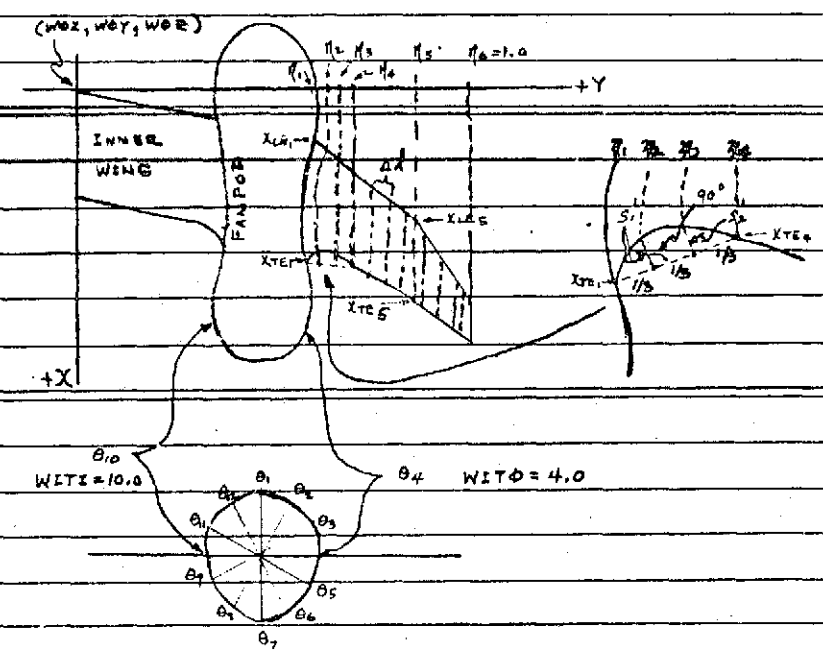
DECK NO. _____ PROGRAMMER _____ DATE _____ PAGE _____ of _____ JOB NO. _____

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH	
1	1200			
13				
25	0.0		$W\phi X$ ~ WING ORIGIN X } $W\phi Y$ ~ " " " Y } RELATIVE TO AIRCRAFT ORIGIN (0,0,0) $W\phi Z$ ~ " " " Z }	
37				
49				
61			$WIT\phi$ ~ I TH θ OF FANPOD AT WHICH OUTER WING IS ATTACHED. $WITI$ ~ " " " " " " " " INNER " " " "	
1	-1205			
13				
25			$WNVS\phi$ ~ NO. OF VORTICES SPANWISE FOR BOTH SIDES OF OUTER WING. IF WING ONLY (ODD INTEGER), OTHERWISE (EVEN INTEGER), (LIM. 11-50). $WNVSI$ ~ NO. OF VORTICES SPANWISE FOR BOTH SIDES OF INNER WING, NO BODY (ODD INTEGER), WITH BODY (EVEN INTEGER), (LIM. 11-50).	
37				
49			$WITB$ ~ I TH θ OF BODY AT WHICH WING IS ATTACHED. $XNE\phi$ ~ No. OF N IN OUTER WING TABLE: (LOC. 1210 → 1239) $XNEI$ ~ " " " " INNER " " " (LOC. 1240 → 1269)	
61				
1	1210		WING PLANFORM TABLE	
13				
25			N_1 OUTER WING (WING ONLY)	
37				
49				
61				
1	1215			
13				
25				N_3 XLE_3 XTE_3
37				
49				
61				

FORTRAN FIXED 10 DIGIT DECIMAL DATA

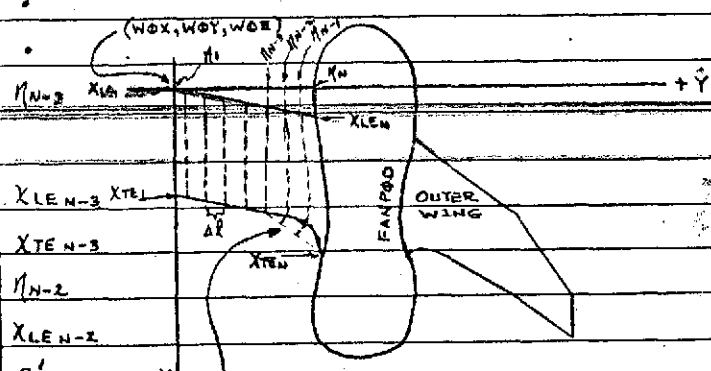
DECK NO. _____ PROGRAMMER _____ DATE _____ PAGE _____ of _____ JOB NO. _____

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	1210	WING PLANFORM TABLE
13		η_1 OUTER WING (FANPOD-WING)
25		X_{LE1} NOTE:
37		X_{TE1} 1. $\eta = \frac{Y}{b/2}$.
49		η_2 2. MAX. 10 η
61		X_{LE2} 3. $(\eta_2 - \eta_1) \approx (\eta_3 - \eta_2) \approx (\eta_4 - \eta_3) \approx \Delta \eta = \frac{1 - \eta_1}{.5 * WNV50 - .5}$
1	1215	4. η, X_{LE}, X_{TE} RELATIVE TO ORIGIN (W_{OX}, W_{OY}, W_{OE})
13		S'_1
25		η_3
37		X_{LE3}
49		S'_2 (W_{OX}, W_{OY}, W_{OE})
61		η_4
1	1220	INNER WING
13		X_{LE4}
25		X_{TE4}
37		η_5
49		X_{LE5}
61		X_{TE5} +X
1	1225	η_6 WITZ = 10.0
13		θ_1
25		θ_2
37		θ_3
49		θ_4 WITD = 4.0
61		θ_5



FORTRAN FIXED 10 DIGIT DECIMAL DATA

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

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	1.240		WING PLANFORM TABLE (CON'T.)
13			η_1 INNER WING (FANPOD - WING)
25			XLE_1 NOTE:
37			XTE_1 1. $\eta = \frac{Y}{b/2}$.
49		73 80	η_2 2. MAX. 10 η
61			XLE_2 3. $(\eta_{N-2} - \eta_{N-2}) \approx (\eta_{N-1} - \eta_{N-2}) \approx (\eta_N - \eta_{N-1}) \approx \Delta \eta = \frac{\eta_N}{.5 * WINGSI - .5}$
1	1.245		4. η, XLE, XTE RELATIVE TO ORIGIN (WOX, WOY, WOE)
13			XTE_2
25			.
37			.
49		73 80	.
61			
1	1.250		η_{N-2} XLE_{N-3} XTE_{N-3} η_{N-2} XLE_{N-2} S'_1 +X
13			XLE_{N-3} XTE_{N-3}
25			XTE_{N-3}
37			η_{N-2}
49		73 80	XLE_{N-2}
61			S'_1 +X
1	1.255		η_{N-1} XLE_{N-1} S'_2 η_N XLE_N
13			η_{N-1}
25			XLE_{N-1}
37			S'_2
49		73 80	η_N
61			XLE_N

Y =

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		* No. OF Cp CALC. PER SEMI-SPAN OF WING. (MAX. 30) WNPS ~ LIST OF Y AT LOC. 1900.
61		73 80 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		* No. OF CONTROL POINTS FOR SEMI-SPAN ON THE WING. WNJS ~ LIST OF J AT LOC. 1690. (MAX. 30)
61		73 80 FNXJ ~ 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		* No. OF SPANWISE ASSUMED LOAD SHAPES FOR THE WING. WNW ~ 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 SPANWISE ALSO WNPS AND WNJS ARE NOT REQ'D
61		NOT USED
1	1285	LIST OF SPANWISE ASSUMED LOAD SHAPES FOR WING
13		W1
25		W2
37		W3
49		73 80 * SPECIAL REFUNC ADDITIONAL DATA AT SPECIFIED LOC.
61		

}

0.0
2.
4.
6.
8.
1930.
1935.
1940.

STANDARD η^w

FORTRAN FIXED 10 DIGIT DECIMAL DATA

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

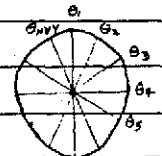
NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH	
1	73 80	LIST OF LONGITUDINAL ASSUMED LOAD SHAPES FOR FANPOD.	
13			1295
25			F ₁ F FOR TRIG. FUNC. F FOR LINEAR FUNC. (MAX. 25)
37			F ₂ 1. ~ $\cos \phi/2$ (x/c) ₁
49			F ₃ 2. ~ $\cos (90 - \phi/2)$ (x/c) ₂
61			F ₄ 3. ~ $\sin \phi$ (x/c) ₃
1	73 80	5. ~ $\sin 2\phi$ (x/c) ₅	
13			1300
25			F ₅ 4. ~ $\cos \phi$ (x/c) ₄
37			F ₆ 6. ~ $\cos 2\phi$ (x/c) ₆
49			.
61			.
1	73 80	COMMENT ON x/c	
13			1305
25			1. SET (x/c) ₁ = 0.0
37			2. SET FOLLOWING x/c BETWEEN
49			CONTROL POINTS (FJX'S).
61			F _{4F} 3. LAST x/c NOT EQUAL TO 1.0
1	73 80		
13			1310
25			
37			
49			
61			

52

FORTRAN FIXED 10 DIGIT DECIMAL DATA

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

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	73 80	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 FLOI (LOC. 17) = 1.0
37		FVX ₃ FVX ₁ < FVX ₂ < FVX ₃ < . . .
49		.
61		.
1	73 80	+465
13		.
25		.
37		FVX _{NVX} = FCD (FANPOD CHORD)
49		.
61		.
1	73 80	1610 LIST OF LATERAL VORTICES ON FANPOD (MAX. 40)
13		FVT ₁ - θ ₁ = 0.0 (θ IN DEGREES)
25		FVT ₂ LIST IS REQ'D. ONLY IF FTHI (LOC. 18) = 1.0
37		FVT ₃ θ ₁ < θ ₂ < θ ₃ <
49		.
61		.
1	73 80	1615
13		.
25		.
37		FVT _{NVY} ~ θ _{NVY} * 360.
49		.
61		.



53

FORTRAN FIXED 10 DIGIT DECIMAL DATA

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

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	73 80	1.660 LIST OF J (CONTROL POINTS) PER CHORD ON WING. (MAX. 30)*
13		WJC ₁ START FROM LEADING EDGE.
25		WJC ₂
37		:
49		:
61		WJC _{NJC} < WNVC (Loc. 1270)
1	73 80	+690 LIST OF J (CONTROL POINTS) FOR SEMI-SPAN ON WING. (MAX. 30)*
13		WJS ₁ START FROM WING TIP
25		WJS ₂
37		:
49		:
61		WJS _{NJS} < .5*(WNVS+1)
1	73 80	1720 LIST OF J (CONTROL POINTS) IN LONGIT. DIRECTION ON FANPOD.
13		FJX ₁ ≠ 1. (MAX. 75)
25		FJX ₂ NOTE: DO NOT SET. FJX ₁ = 1.0
37		FJX ₃
49		.
61		.
1	73 80	1725
13		.
25		.
37		.
49		FJX _{NJX}
61		.

FORTRAN FIXED 10 DIGIT DECIMAL DATA

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

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	1870		LIST OF X/C FOR C_p CALC. ON WING. (MAX. 30)
13			$WPC_1 = (X/C)_1$
25			WPC_2
37			WPC_3
49		73 80	.
61			.
1	1875		
13			.
25			.
37			WPC_{NPC}
49		73 80	
61			
1	1900		LIST OF η (SPANWISE) FOR C_p CALC. ON WING. (MAX. 30)
13			$WPS_1 = \eta_1$
25			WPS_2 LIST IS NOT REQ'D., IF $WNW=0.0$ (DISCRETE)
37			WPS_3
49		73 80	.
61			.
1	1905		
13			.
25			.
37			WPS_{NPS}
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	1930	SPECIAL WING SPANWISE LOAD SHAPE #1.
13		P-FUNC FLAP KRUEGER
25		TYPE INDI. ~ 0.0 -1.0 1.0
37		η^* NOT USED NOT USED
49		RL ($\Delta\eta_i$) η_{fi} η_{ri}
61		RR ($\Delta\eta_o$) η_{fo} η_{ro}
61	NOT USED NOT USED NOT USED	
1	1935	SPECIAL WING SPANWISE LOAD SHAPE #2.
13		TYPE INDI. ~
25		
37		
49		
61		
61		
1	1940	COMMENTS:
13		1. THE P-FUNC η^* SHOULD BE AT A "CONTROL POINT" η .
25		2. THE FLAP AND KRUEGER, η_{fi} AND η_{ro} , SHOULD BE AT
37		A "TRAILING VORTEX" η .
49		3. ONLY ONE FLAP AND/OR KRUEGER ON WING.
61		4. THE P-FUNC, FLAP AND KRUEGER LOC. (1930, 1935...)
61	MUST BE LISTED LAST, RESPECTIVELY, IN LIST OF W.	
1	1945	
13		
25		
37		
49		
61		

95

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		$\delta_f \sim$ FLAP DEFLECTION ANGLE, (RADIAN)
25		$(x/c)_f \sim$ FLAP PIVOT x/c ,
37		η_{fl}
49		η_{fo}
61		NOT USED
1	1965	WING CHORDWISE LOAD SHAPE FOR KRUEGER.
13		$\delta_k \sim$ KRUEGER DEFLECTION ANGLE. (RADIAN)
25		$(x/c)_k \sim$ KRUEGER PIVOT x/c .
37		η_{ki} COMMENT:
49		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 \rightarrow 745.
13		$E_1 \sim$ TWIST AT η_1 (RADIAN)
25		$E_2 \sim$ " " η_2
37		$E_3 \sim$ " " η_3
49		.
61		.
1	1985	ΔZ OF WING LEADING EDGE PER η IN LOC. 731 \rightarrow 745.
13		ΔZ_{LE1} *NOTE!
25		ΔZ_{LE2} TWISTS ARE ADDED TO WING ANGLE OF
37		ΔZ_{LE3} ATTACK ONLY IF WAAZ (LOC. 9) = 1.0
49		.
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 . . .
49		73 80	(X/C) ₃ NOTE: THE THICKNESS CALCULATIONS
61			• ARE BYPASSED 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 $\eta_1 < \eta_2 < \eta_3$. . .
49		73 80	η_3
61			•
1	2030		
13			-
25			-
37			η_{NETA}
49		73 80	
61			

85

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 (CONT.)
13			(Z/C) ₁
25			(Z/C) ₂
37			(Z/C) ₃
49		73	80
61			- } FOR η_1 AT EACH X/C
1	2055		
13			-
25			-
37			(Z/C) _{NXC+1}
49		73	80
61			(Z/C) _{NXC+2}
1	2060		(Z/C) _{NXC+3}
13			-
25			-
37			-
49		73	80
61			-
1	2065		(Z/C) _{2NXC+1}
13			(Z/C) _{2NXC+2}
25			(Z/C) _{2NXC+3} } FOR η_3 AT EACH X/C
37			-
49		73	80
61			-

59

FORTRAN FIXED 10 DIGIT DECIMAL DATA

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

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

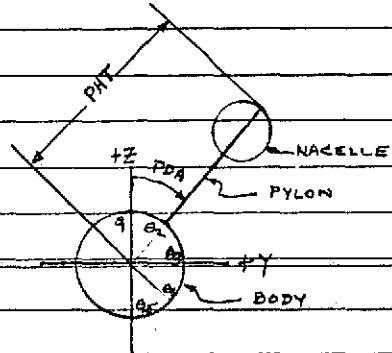


ILLUSTRATION:

ATP = 1.0
 PITH = 2.0
 PDA = 45.0

NOTE:
 IF PYLON IS ATTACHED TO
 THE WING, ATTACH PYLON ON
 A WING "TRAILING" VORTEX.

09

FORTRAN FIXED 10 DIGIT DECIMAL DATA

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

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH	
1	2500	PYLON DATA ⁺¹ _{1.0}	
13		PHT ~ PYLON HEIGHT	
25			
37			PΦX } PYLON ORIGIN
49			PΦY } RELATIVE TO (0,0,0)
61		PΦZ } PAA ~ PYLON ANGLE OF ATTACK (DEG.)	
1	2505	0.0 ~ PYLON IS FLAT.	
13		PAAI ~ 1.0 ~ PYLON ANGLE OF ATTACK = PAA + CAMBER + TWIST	
25		PADI ~ TABLE (2530-2939) { 0.0 ~ ANGLE (α IN RADIAN)	
37		{ 1.0 ~ DEFLECTION (Z/C)	
49		PNVC ~ No. OF VORTICES CHORDWISE ON PYLON. (MAX. 19)	
61		PNVs ~ " " " SPANWISE " " (MAX. 20)	
19		PUN ~ No. OF CHORDWISE ASSUMED LOAD SHAPES FOR PYLON. (MAX. 10)	
1	2510	PYLON STARTING J CHORDWISE (NOT USED IN THICKNESS CALC.)	
13		PSJC ~ SET EQUAL TO A J IN LIST BELOW. (APPROX. 7% OF CHORD)	
25		No. OF CONTROL POINTS PER CHORD ON PYLON IN	
37		PNJC ~ LIST BELOW. (MAX. 18)	
49		PJC ₁ LIST OF PYLON J	
61		PJC ₂	
1		PJC ₃	
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/R IN LIST BELOW. (MAX. 19)
25			PXC ₁
37			PXC ₂ x/c ₁ < x/c ₂ < x/c ₃ . . .
49		73 . . . 80	PXC ₃
61			.
1	2535		
13			.
25			NOTE:
37			THE PYLON CAMBER TABLE IS NOT
49		73 . . . 80	REQUIRED, IF PAAL = 0.0 (LOC. 2505)
61			.
1	2550		
13			PNE ~ NO. OF η IN LIST BELOW. (MAX. 19)
25			PET ₁ = η ₁ η = $\frac{Y}{b/2}$
37			PET ₂ = η ₂ η ₁ < η ₂ < η ₃
49		73 . . . 80	PET ₃ = 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. _____

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH	
1	73 80	PYLON CAMBER TABLE (CONT)	
13			
25		PAD ₁	LIST OF:
37		PAD ₂	1.) IF PADI=0.0 (LOC. 2506), ANGLE OF ATTACKS (α IN RAD)
49		PAD ₃	2.) IF PADI=1.0 (LOC. 2506), DEFLECTIONS (E/C)
61			FOR η_1 AT EACH X/C
1	73 80		
13			
25			NOTE:
37			NK# NO. OF X/E IN LOC. 2530
49		PAD _{NK+1}	
61		PAD _{NK+2}	
1	73 80		
13			
25			FOR η_2 AT EACH X/C
37			
49			
61		PAD _{NK+NK+1}	
1	73 80		
13			
25			FOR η_3 AT EACH X/C
37			
49			
61		PAD _{NK+NK+2}	

59

FORTRAN FIXED 10 DIGIT DECIMAL DATA

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

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH		
1	2940		PYLON TWIST PER η IN LOC. 2551 \rightarrow 2569		
13			$E_1 \sim$ TWIST AT η_1 .		
25			$E_2 \sim$ " " η_2		
37			$E_3 \sim$ " " η_3		
49			73	80	.
61			.		
1	2945		.		
13			NOTE:		
25			TWISTS ARE ADDED TO PYLON ANGLE OR		
37			E_{PNE} ATTACK ONLY IF $PA_{AZ}(\text{LOC}, 2505) = 1.0$		
49			73	80	.
61			.		
1	2960		PYLON PLANFORM TABLE		
13			η_1		
25			XLE_1 NOTE:		
37			XTE_1 SEE WING PLANFORM TABLE (OUTER WING)		
49			73	80	η_2 DISCRPTIONS.
61			XLE_2		
1	2965				
13			S_1		
25			η_3		
37			XLE_3		
49			73	80	S_2
61			η_4		

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		$(X/c)_2$ $(X/c)_1 < (X/c)_2 < (X/c)_3 \dots$
49	73	$(X/c)_3$ NOTE: THE THICKNESS CALCULATIONS
61	80	• ARE BYPASS, IF NXC = 0.0
1	3005	
13		•
25		•
37		$(X/c)_{NXC}$
49	73	
61	80	
1	3020	
13		NETA ~ No. OF η IN LIST BELOW. (MAX. 18)
25		η_1
37		η_2 $\eta_1 < \eta_2 < \eta_3 \dots$
49	73	η_3
61	80	•
1	3025	
13		•
25		•
37		η_{NETA}
49	73	
61	80	

59

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				.

99

FORTRAN FIXED 10 DIGIT DECIMAL DATA

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

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	73 80	* 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		XLP ~ " " " " " "Y" " " " PYLON.
61		XLNC ~ " " " " " X " " " NACELLE.
1	73 80	XLNR ~ " " " " " "Y" " " " NACELLE.
13		XLBX ~ " " " " " X " " " BODY.
25		XLBY ~ " " " " " Y " " " BODY.
37		NOT USED
49		* NOTE :
61		" " " " " INITIALLY SET TO 1000000.0
1	73 80	
13		
25		
37		
49		
61		
1	73 80	
13		
25		
37		
49		
61		

19

FORTRAN FIXED 10 DIGIT DECIMAL DATA

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

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1		NACELLE DATA
13		NB ~ No. of BODIES
25		XNO
37		YNO
49		ZNO
61		XNPI ~ No. of COORD. ON BODY #1
1		NOTE: XNPI + XNP2 ≤ 140
13		XNP2 ~ No. of COORD. ON BODY #2
25		NOT USED
37		NOTE:
49		" " 1. FOR BODY #1 READ COORD. FROM TRAILING EDGE → CLOCKWISE.
61		" " 2. " " #2 " " " LEADING " "
1		X COORD. OF BODY #1 (NACELLE) (MAX 140)
13		XB1
25		XB12
37		:
49		:
61		:
1		Y COORD. OF BODY #1 (NACELLE) (MAX 140)
13		YB1
25		YB12
37		:
49		:
61		:

+Y REPEAT FIRST AND LAST POINT OF BODY #1

Body #1

Body #2

NACELLE ORIGIN RELATIVE TO (0,0,0)

(XNO, YNO, ZNO)

73 80

XNPI ~ No. of COORD. ON BODY #1

NOTE: XNPI + XNP2 ≤ 140

XNP2 ~ No. of COORD. ON BODY #2

NOT USED NOTE:

" " 1. FOR BODY #1 READ COORD. FROM TRAILING EDGE → CLOCKWISE.

" " 2. " " #2 " " " LEADING " "

" " 3. ALL OF THE YCOORD. MUST BE POSITIVE AND NO TWO CONSECUTIVE Y VALUES MAY BE ZEROS.

X COORD. OF BODY #1 (NACELLE) (MAX 140)

XB1

XB12

:

:

:

Y COORD. OF BODY #1 (NACELLE) (MAX 140)

YB1

YB12

:

:

:

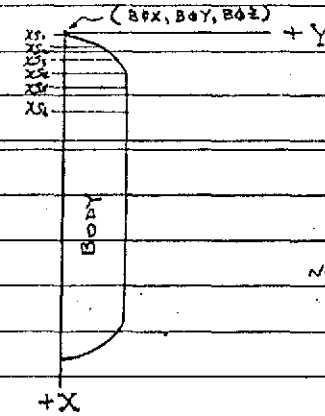
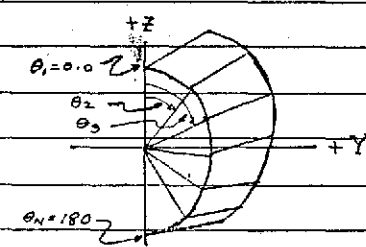
FORTRAN FIXED 10 DIGIT DECIMAL DATA

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

	NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	37.00		X COORD. OF BODY #2 (NACELLE) (MAX.140)
13			XB2 ₁
25			XB2 ₂
37			:
49		73 80	:
61			:
1	38.50		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	4020	BODY COORDINATE TABLE
13		BNXS ~ No. of X CROSS-SECTIONS IN LIST BELOW. (MAX. 29.)
25		$X_{S_1} \quad X_{S_1} < X_{S_2} < X_{S_3} \dots$ RELATIVE TO BODY ORIGIN,
37		($B_{S_1}, B_{S_2}, B_{S_3}$)
49	73	X_{S_2} 
61	80	X_{S_3}
1	4025	
13		
25		
37		NOTE :
49	73	
61	80	
1	4050	
13		BNTY ~ No. of θ AT EACH CROSS-SECTION IN LIST BELOW. (MAX. 36)
25		$\theta_1 \quad \theta_1 < \theta_2 < \theta_3 \dots$ ($\theta = 0.0 \rightarrow 180.$ IN DEG.)
37		θ_2
49	73	θ_3
61	80	
1	4055	
13		
25		$\theta_1 = 0.0$
37		θ_2
49	73	θ_3
61	80	$\theta_N = 180$

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 ₁ AT EACH Θ
61			•
1	-4095		
13			•
25			•
37			R _{NTY+1}
49		73. 80.	R _{NTY+2}
61			R _{NTY+3}
1	4100		
13			• FOR X ₂ AT EACH Θ
25			•
37			•
49		73. 80.	•
61			R _{NTY+NTY+1}
1	4105		
13			R _{NTY+NTY+2}
25			• FOR X ₃ AT EACH Θ
37			•
49		73. 80.	•
61			•

NOTE:
NTY = No. OF Θ IN LOG: 50.

22

FORTRAN FIXED 16 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 CAMBER'S ARE GIVEN.
49		73	XMC ₃ XMC ₁ < XMC ₂ < XMC ₃ - - -
61		80	.
1	4570		
13			.
25			NOTE :
37			IF BODY HAS NO CAMBER AND THE MULTI.
49			XMCNXM FACTORS ARE NOT REQ'D. ; SET BNXM = 0,0
61		73	80
1	4600		
13			YM ₁ LIST OF MULTI. FACTORS Y _m
25			YM ₂
37			YM ₃
49		73	.
61		80	.
1	4605		
13			.
25			.
37			YMNXM
49		73	80
61			

73

FORTRAN FIXED 16 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 (CONT.)
13			ZM ₁ LIST OF MULTI. FACTORS ZM
25			ZM ₂
37			ZM ₃ NOTE: LIST OF ZM IS NOT REQ'D, IF YN = ZM
49		73 80	• SET FMFI = 0.0 IN LOC. 15
61			•
1	4640		
13			•
25			•
37			ZM _{NXM}
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 _{NXM}
49		73 80	
61			

FORTRAN FIXED 10 DIGIT DECIMAL DATA

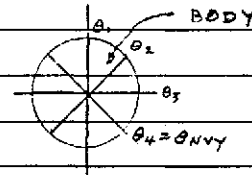
DECK NO. _____ PROGRAMMER _____ DATE _____ PAGE _____ of _____ JOB NO. _____

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	4705	
13		
25		
37		
49		
61		
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		F ₄ 3. ~ $\sin \phi$ (X/C) ₃
61		F ₅ 4. ~ $\cos \phi$ (X/C) ₄
1	4715	5. ~ $\sin 2\phi$ (X/C) ₅
13		F ₆ 6. ~ $\cos 2\phi$ (X/C) ₆
25		.
37		.
49		.
61		.
1	4720	COMMENT ON X/C
13		1. SET (X/C) ₁ = 0.0
25		F _{NP} 2. SET FOLLOWING X/C BETWEEN
37		CONTROL POINTS (BJX'S).
49		3. LAST X/C NOT EQUAL TO 1.0.
61		

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 B1G1(LOC. 4009) = 1.0
37		BVX ₃ BVX ₁ < BVX ₂ < BVX ₃ < . . .
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 = θ ₁ 0.0 (θ IN DEGREES)
25		BVT ₂ LIST IS REQ'D. ONLY IF BTHI(LOC. 4010) = 1.0
37		BVT ₃ θ ₁ < θ ₂ < θ ₃ < . . .
49		73 80
61		.
1	4890	
13		.
25		.
37		BVT _{NVY} ~ θ _{NVY} ≠ 180.
49		73 80
61		.



76

FORTRAN FIXED 16 DIGIT DECIMAL DATA

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

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
1	73 80	49.05
13		LIST OF J (CONTROL POINTS) IN LONGIT. DIRECTION ON BODY,
25		BJX ₁ ≠ 1.0 (MAX. 75)
37		NOTE: DO NOT SET BJX ₁ = 1.0
49		BJX ₂
61		BJX ₃
1	73 80	49.10
13		.
25		.
37		BJX _{NJX}
49		
61		
1	73 80	49.80
13		SKIN FRICTION INPUT DATA
25		* T _∞ ~ FREE STREAM STATIC TEMPERATURE. (°R)
37		P _∞ ~ " " " PRESSURE. (#/ft ²)
49		CK ~ THICKNESS CORRECTIONS (AIRFOILS) $K = 1 + \frac{CK}{2} + 60 \frac{CK}{S} + C_D \frac{CK}{S^2}$
61		TRID ~ TRANSITION INDICATOR { -1.0 ~ TRANS. POINTS (X/T _∞) ARE INPUT, 0.0 ~ FLAT PLATE NATURAL TRANSITION.
1	73 80	49.85
13		K _s ~ EQUIVALENT SAND GRAIN HEIGHT (FT.)
25		T ~ TURBULENCE INTENSITY (NOT REQ'D, IF TRID = -1.0)
37		CONV ~ CONVERSION TO FEET. EXAMPLE: CONV = 12.0, IF CASE DATA ARE IN INCHES.
49		NOT USED * NOTE:
61		" " IF T _∞ IS SET TO 0.0, THE SKIN FRICTION DRAG CALCULATION IS BYPASSED.

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 (CONT.)
13			XTRL ₁ ~ X TRANSION / LENGTH FOR BODY.
25			XTRL ₂ ~ " " " " FANPOD.
37			XTRL ₃ ~ " " " " WING.
49		73 80	XTRL ₄ ~ " " " " PYLON.
61			XTRL ₅ ~ " " " " NACELLE.
1	4995		
13			TCM ₁ ~ MAX. THICKNESS / LENGTH FOR BODY.
25			TCM ₂ ~ " " " " FANPOD.
37			TCM ₃ ~ " " " " WING.
49		73 80	TCM ₄ ~ " " " " PYLON.
61			TCM ₅ ~ " " " " NACELLE.
1	5000		LAST DATA CARD OF A CASE WITH A MINUS IN COL. 1.
13			
25			
37			
49		73 80	
61			
1			
12			
25			
37			
49		73 80	
61			

READ ONLY
IF TRID = -1.0

IF PLAT PLATE
SET TCMFO, 0

78

SAMPLE INPUT DATA

A04NZXX,5,1400,60000,482511,RCST5786P FANPOD
 *NOTAPES,PSS
 LIBCOPY,A04RCS,LIB,OBJECT.
 COPY,LIB/RX,LGO.
 RFL(135000,580000)
 LGO.

CAS10010
 CAS10020
 CAS10030
 CAS10040
 CAS10050
 CAS10060
 CAS10070
 CAS10080
 CAS10090
 CAS10100
 CAS10110
 CAS10120
 CAS10130
 CAS10140
 CAS10150
 CAS10160
 CAS10170
 CAS10180
 CAS10190
 CAS10200
 CAS10210
 CAS10220
 CAS10230
 CAS10240
 CAS10250
 CAS10260
 CAS10270
 CAS10280
 CAS10290
 CAS10300
 CAS10310
 CAS10320
 CAS10330
 CAS10340
 CAS10350
 CAS10360
 CAS10370
 CAS10380
 CAS10390
 CAS10400
 CAS10410
 CAS10420
 CAS10430

1	4.0	24.0	6.125	0.6	
10					3.0
1200	0.0	0.0	0.0	0.0	
1205	18.0		4.0	5.0	
1210	0.1315	14.30	21.4060	0.2335	15.6
1215	0.0	0.3355	16.9	0.0	0.4375
1220	18.2	24.388	1.0	25.35	29.855
1270	20.0	21.0	4.0		
1275			11.0		
1280	5.0	0.0			
1660	1.0	2.0	3.0	5.0	7.0
1665	9.0	11.0	13.0	15.0	17.0
1670	19.0				
1870	0.2	0.4	0.6	0.8	
2000	20.0	0.0	0.005	0.0075	0.0125
2005	0.025	0.05	0.1	0.15	0.20
2010	0.25	0.30	0.35	0.4	0.45
2015	0.5	0.55	0.6	0.7	0.8
2020	1.0				
2025	2.0	0.0	1.0		
2050	0.0	0.00464	0.00563	0.00718	0.00981
2055	0.01313	0.01824	0.02194	0.02474	0.02687
2060	0.02842	0.02945	0.02996	0.02992	0.02925
2065	0.02793	0.02602	0.02087	0.01437	0.00013
2070	0.0	0.00464	0.00563	0.00718	0.00981
2075	0.01313	0.01824	0.02194	0.02474	0.02687
2080	0.02842	0.02945	0.02996	0.02992	0.02925
2085	0.02793	0.02602	0.02087	0.01437	0.00013
4005	33.33	0.0	0.0	1.0	1.0
4010	0.0				
4015	0.0	0.0	0.0		
4565	23.0	0.0	0.2	0.3	0.5
4570	1.0	2.0	3.0	4.0	6.0
4575	8.0	10.0	12.0	14.0	16.0
4580	18.0	20.0	22.0	24.0	26.0
4585	28.0	30.0	32.0	33.33	

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
 *NOTAPES,PSS
 LIBCOPY,A04RCS,LIB,OBJECT.
 COPY,LIB/RX,LGO.
 RFL(135000,580000)
 LGO.

CAS20010
 CAS20020
 CAS20030
 CAS20040
 CAS20050
 CAS20060
 CAS20070
 CAS20080
 CAS20090
 CAS20100
 CAS20110
 CAS20120
 CAS20130
 CAS20140
 CAS20150
 CAS20160
 CAS20170
 CAS20180
 CAS20190
 CAS20200
 CAS20210
 CAS20220
 CAS20230
 CAS20240
 CAS20250
 CAS20260
 CAS20270
 CAS20280
 CAS20290
 CAS20300
 CAS20310
 CAS20320
 CAS20330
 CAS20340
 CAS20350
 CAS20360
 CAS20370
 CAS20380
 CAS20390
 CAS20400
 CAS20410
 CAS20420
 CAS20430

1	5.9744	55.92	9.918332	0.6		
5	26.398	0.0	1.0	3.0	1.0	
10	1.0	-1.0	0.0	0.0	3.0	
15	1.0	1.0	1.0	1.0	13.0	
705	20.0	0.0	0.0003	0.001	0.002	
710	0.0035	0.005	0.0075	0.0125	0.025	
715	0.05	0.1	0.2	0.3	0.4	
720	0.5	0.6	0.7	0.8	0.9	
725	1.0					
730	5.0	0.0	0.3262	0.5988	0.7495	
735	1.000					
750	0.0	0.000283	0.000730	0.001095	0.001494	
755	0.001857	0.002324	0.002881	0.004212	0.005075	
760	0.00635	0.0084	0.00945	0.0096	0.00895	
765	0.0077	0.00605	0.0041	0.00195	0.0	
770	0.0	0.000257	0.000447	0.0006705	0.0009566	
775	0.00121	0.00168	0.00242	0.00401	0.00648	
780	0.00881	0.01252	0.01583	0.01851	0.01865	
785	0.01737	0.01455	0.01138	0.00633	0.0	
790	0.0	0.0000939	0.000225	0.000358	0.000552	
795	0.00072	0.000978	0.001481	0.00268	0.00511	
800	0.00872	0.01399	0.01729	0.01959	0.02150	
805	0.02081	0.01901	0.01537	0.01116	0.0	
810	0.0	0.000185	0.000485	0.000802	0.001207	
815	0.001582	0.00220	0.00326	0.00516	0.00791	
820	0.01058	0.01546	0.01951	0.02232	0.02380	
825	0.02334	0.02144	0.01821	0.01304	0.0	
830	0.0	0.000650	0.001194	0.001730	0.00246	
835	0.00308	0.00400	0.00545	0.00794	0.01073	
840	0.01314	0.01634	0.01871	0.02034	0.02011	
845	0.01869	0.01575	0.01160	0.00656	0.0	
1270	20.0	41.0	12.0	10.0	37.0	
1275	12.0	3.0	10.0	10.0	19.0	
1280	6.0	0.0	10.0			
1660	2.0	3.0	5.0	7.0	9.0	
1665	11.0	13.0	15.0	17.0	19.0	

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

VL/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

EPA - 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)

** SURSON 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	1.185700
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	0.003000	0.007500	0.014000
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	27.000000	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	0.000000
640	.225000	.310000	.410000	.520000	.170000
645	.695000	.760000	.830000	1.020000	.615000
650	1.420000	1.575000	1.660000	1.690000	1.245000
655	1.690000	1.690000	1.645000	1.495000	1.690000
660	1.085000	.805000	.505000	.355000	1.275000
665	.155000	.105000	.060000	0.000000	.186000
705	20.000000	0.000000	.000300	.001000	0.000000
710	.003500	.005000	.007500	.012500	.002000
715	.050000	.100000	.200000	.300000	.025000
720	.500000	.600000	.700000	.800000	.400000
725	1.000000	0.000000	0.000000	0.000000	.900000
730	5.000000	0.000000	.326200	.598800	0.000000
735	1.000000	0.000000	0.000000	0.000000	.749500
750	0.000000	.000283	.000730	.001095	0.000000

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	19.980000	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

95

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
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	.047900
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
96 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 1 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	-.36450	-.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

97

** SUBSONIC INTERFERENCE PRESSI PROGRAM ** FUSFLAGE OUTPUT **

NO.	XQ	YQ	ZQ	VT/VFS	VM/VFS	CP
1	.01416	.01075	.02101	.02704	-.28092	.99913
2	.18386	.13806	.27105	.03243	-.00043	1.09191
3	.55616	.40163	.80035	.05369	.37004	.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	1.27224	-.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	1.08492	-.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	1.23727	-.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	1.00055	-.04294
39	64.71447	.14199	.37427	-.30926	1.00025	-.11108
40	64.95188	.02429	.35678	-.82505	1.09899	-.81970

** SUBSONIC INTERFERENCE PRESS PROGRAM ** FUSFLAGE 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

** SURSONIC INTERFERENCE PRESS PROGRAM ** FUSELAGE OUTPUT **

NO.	XQ	YQ	ZQ	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	.79054	.28307
6	2.51773	3.28195	.90150	.25138	.85464	.21027
7	3.60346	3.91171	1.18757	.20968	.93068	.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	1.15602	-.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

100

** SUBSONIC INTERFERENCE PRESSI

PROGRAM ** FUSELAGE OUTPUT **

NO.	X0	Y0	Z0	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

101

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

** SURSONIC 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 ARI

677.89A22

* CL=	.01568	* CD=	.00028	* CMXY=	-.09697	* CMYZ=	.00284	* CM=	-.09413
X	CLW/WA	CDW/WA	X	CLW/WA	CDW/WA				
.00050	.00626	.00931	.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	.03244				
.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							

1001

** WING I EAR PPESSURE 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

.149	.8746 .1372	.6018 .0781	.3983	.2474	.1962	.1937	.2169	.2443	.2528	.2210
.226	.7850 .1659	.5528 .1166	.3996	.3181	.3014	.2989	.2966	.2878	.2674	.2298
.303	.7732 .2007	.5590 .1481	.4236	.3598	.3514	.3506	.3446	.3295	.3035	.2639
.381	.7427 .2373	.5796 .1843	.4729	.4073	.3906	.3874	.3813	.3645	.3353	.2955
.458	.7662 .2923	.6133 .2404	.5093	.4363	.4154	.4117	.4058	.3890	.3629	.3338
.536	.7879 .3539	.6457 .3033	.5437	.4609	.4346	.4307	.4260	.4109	.3904	.3755
.613	.8030 .4139	.6660 .3636	.5644	.4757	.4466	.4429	.4398	.4280	.4154	.4163
.690	.8196 .4486	.6444 .3899	.5305	.4628	.4479	.4452	.4402	.4330	.4334	.4488
.768	.7960 .4543	.6004 .3912	.4831	.4358	.4337	.4331	.4272	.4230	.4314	.4547
.845	.6826 .3754	.5348 .3186	.4441	.3997	.3947	.3959	.3935	.3881	.3863	.3910
.923	.5042 .2598	.4236 .2160	.3705	.3702	.3187	.3190	.3187	.3120	.3003	.2870
.971	.3360 .1427	.2861 .1195	.2446	.2012	.1877	.1906	.1948	.1907	.1779	.1619

105

**WING NO LINEAR VELOCITY AND PRESSURE COEFFICIENT

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	.52001	.33793
.05450	1.19383	.99272	-.03918	.04933	-.41062	.01209	.42271	.24008
.14645	1.18525	1.06164	-.04301	.03210	-.34201	-.12664	.26537	.22008
.27300	1.16435	1.06088	-.05989	.03105	-.34783	-.12501	.22282	.18900
.42178	1.15484	1.05104	-.08830	.02906	-.33109	-.10452	.22650	.19188
.57822	1.14139	1.03414	-.11492	.03021	-.30709	-.06992	.27717	.14247
.72700	1.10036	.99368	-.12888	.04215	-.22278	.01084	.23362	.17359
.85355	1.04569	.94670	-.13100	.06145	-.10953	.10088	.21041	.15304
.94550	.98065	.89450	-.12304	.08743	.02323	.19629	.17306	.12245
.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	-.44706	-.08162	.36634	.31929
.57822	1.18865	1.02180	-.10304	.03211	-.40761	-.04492	.36268	.31406
.72700	1.13892	.97423	-.10164	.05001	-.29905	.04858	.34763	.29588
.85355	1.07537	.91905	-.09610	.06987	-.16320	.15252	.31572	.27690
.94550	1.00111	.86835	-.08624	.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	.04246	-.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

106

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	.08020	-.63581	.33196	.96777	.80345
.05450	1.25524	.91171	-.08512	.05249	-.55293	.16852	.72145	.52182
.14645	1.25081	1.00190	-.08897	.01695	-.54355	-.00408	.53946	.44878
.27300	1.25940	1.02617	-.09724	.00844	-.56431	-.05284	.51146	.41335
.42178	1.25014	1.02462	-.09816	.00041	-.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	.44336	.40373
.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	-.08300	.05713	-.57145	.18593	.75738	.62742
.14645	1.26691	.99645	-.09062	.01959	-.58016	.00670	.52686	.48725
.27300	1.27319	1.03076	-.09795	.00566	-.59563	-.06215	.53348	.44971
.42178	1.26091	1.02974	-.09750	.00364	-.56777	-.06005	.50771	.44654
.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	.04680	-.23704	.23429	.47134	.45853
.94550	1.04933	.83597	-.03681	.05365	-.10150	.30637	.40787	.39927
.99384	.91703	.80982	-.00950	.05788	.16125	.35142	.19017	.16501

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	.80816
.05450	1.26452	.89637	-.08344	.06208	-.57364	.19602	.76966	.65229
.14645	1.27794	.99361	-.09135	.02152	-.50530	.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	-.04097	.03609	-.21344	.21908	.43252	.42026
.94550	1.04511	.84706	-.02472	.04120	-.09209	.28796	.38005	.37188
.99384	.92108	.82132	-.00174	.04475	.15368	.33295	.17927	.15715

107

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	-.08810	.08766	-.63365	.30076	.93441	.79333
.05450	1.26208	.90336	-.08151	.06098	-.56783	.18316	.75099	.63156
.14645	1.27304	.99885	-.08640	.01914	-.59340	.00193	.59533	.49600
.27300	1.27985	1.04368	-.08855	-.00043	-.60918	-.08855	.52063	.43873
.42178	1.26976	1.04622	-.08430	-.00355	-.58565	-.09378	.49188	.43074
.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	.12871

ETA = .4583

108

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	.46608
.27300	1.27315	1.04765	-.08224	-.00159	-.59302	-.09671	.49631	.41829
.42178	1.26563	1.05168	-.07743	-.00515	-.57529	-.10506	.47074	.41112
.57822	1.20289	1.00185	-.05669	.00748	-.43222	-.00376	.42846	.39361
.72700	1.14278	.95803	-.03876	.01554	-.29903	.08254	.38157	.35514
.853	1.07673	.91186	-.01960	.02226	-.15745	.17058	.3280	.31579
.945...	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	.84201	-.08439	.08925	-.61340	.28875	.90215	.74887
.05450	1.25070	.92271	-.07640	.05683	-.54144	.14729	.68873	.56376
.14645	1.25206	1.01287	-.07577	.01621	-.54441	-.02610	.51831	.43172
.27300	1.26355	1.05152	-.07656	-.00175	-.57046	-.10469	.46577	.39280
.42178	1.25838	1.05687	-.07033	-.00597	-.55795	-.11578	.44217	.38670
.57822	1.20077	1.01140	-.04935	.00517	-.42680	-.02291	.40389	.36322
.72700	1.13603	.96520	-.02934	.01385	-.28387	.06861	.35246	.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	-.04250	.46138	.37993
.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	.29868
.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

109

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	.25197	.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
.37003

CD
.00295

CM
-8.70490

X/C
22.44391

** INDUCED DRAG = .01174

SAMPLE OUTPUT
(CASE II)

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

1	5.974400	55.920000	9.918330	.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

113

111

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	1.185700
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	.720000	.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

115

755	.001957	.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	.000972	.001481	.002680	.005110
800	.008720	.013990	.017290	.019590	.021500
805	.020810	.019010	.015370	.011160	0.000000
810	0.000000	.000195	.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	.002450
835	.003080	.004000	.005450	.007940	.010730
840	.013140	.016340	.018710	.020340	.0110
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	.613050	.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	.004000
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	.095400
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	3.773073
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	24.850472	25.686935	26.528146	27.373518
4790	28.222462	29.074388	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	51.741587
4820	52.426264	53.097055	53.753491	54.395115	55.021479
4825	55.632149	56.226697	56.804710	57.365784	57.909559
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	63.759326
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	-.03541
-.03904	-.04025	-.04039	-.03230	-.04362	-.04843	-.04764	-.04893	-.04854	-.04426
-.02262	-.02516	-.04276	-.03910	-.02976	-.02460	-.02731	-.06400	-.10873	-.24989
.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	-.82998
.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	-.22579
-.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	.18593
.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

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	1.08491	-.17989
20	30.35675	1.57743	3.09507	.02936	1.04219	-.08634
21	32.92495	1.57743	2.88399	-.08907	1.02316	-.05452
22	35.49847	1.57743	2.66640	-.10340	1.01022	-.04928
23	38.04018	1.57396	2.44962	-.10743	1.01631	-.04426
24	40.56510	1.56742	2.24717	-.10930	1.02214	-.05642
25	43.03046	1.55952	2.05033	-.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	60.57151	.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.	X0	Y0	Z0	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	.93082	.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	1.01810	-.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	61.77494	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

121

** SUBSONIC INTERFERENCE PRESSURE PROGRAM ** FUSELAGE OUTPUT **

NO.	X0	Y0	Z0	VT/VFS	VM/VFS	CP
1	.01416	.04011	.00660	.11173	-.09461	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.	X0	Y0	Z0	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 **

XQ	YQ	ZQ	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	1.04416	-.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	.81416	.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.

677.89R22

* 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	.75496	.54482	-.11262	.00632
.00453	-.01251	.66531	.56712	-.11027	.01710
.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 LIFT 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

.149	1.5100 .1963	1.0413 .1209	.7272	.5460	.4878	.4582	.4367	.4132	.3755	.3080
.226	1.6777 .2263	1.1549 .1574	.8095	.6202	.5603	.5201	.4793	.4347	.3839	.3195
.303	1.8259 .2643	1.2568 .1915	.8783	.6675	.6012	.5585	.5159	.4696	.4181	.3557
.381	1.8125 .2947	1.3249 .2237	.9849	.7490	.6526	.5982	.5540	.5056	.4485	.3819
.458	1.8805 .3491	1.3906 .2794	1.0434	.7908	.6850	.6270	.5809	.5311	.4761	.4197
.536	1.9372 .4104	1.4473 .3421	1.0941	.8252	.7104	.6498	.6033	.5542	.5040	.4612
.613	1.9720 .4634	1.4861 .4016	1.1269	.8468	.7264	.6642	.6179	.5711	.5282	.5011
.690	2.0136 .5019	1.4765 .4262	1.1004	.8271	.7283	.6654	.6161	.5732	.5431	.5306
.768	1.9948 .5027	1.4347 .4242	1.0526	.8064	.7084	.6460	.5950	.5550	.5333	.5298
.845	1.8605 .4150	1.3519 .3455	.9974	.7525	.6497	.5885	.5413	.5017	.4723	.4533
.923	1.5947 .2859	1.1723 .2338	.8643	.6252	.5179	.4607	.4227	.3896	.3579	.3280
.971	1.1677 .1564	.7938 .1272	.5231	.3422	.2837	.2615	.2467	.2287	.2066	.1839

**WING NO .LINEAR VELOCITY AND PRESSURE COEFFICIENTS

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	1.40235	.72910	-.10845	.13436	-.89520	.46892	1.76412	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	.08162	-.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	1.12963
.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	.35270	.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	.04433	-.72170	.07863	.80033	.67142
.42178	1.29152	.98168	-.13112	.04364	-.64399	.03450	.67849	.57746
.57822	1.25059	.97982	-.12674	.05102	-.55059	.03748	.58806	.50960
.72700	1.18461	.93749	-.11608	.07032	-.40138	.11730	.51877	.46544
.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	.11743	-1.10943	.47203	1.58146	1.37860
.14645	1.38472	.88790	-.14174	.05773	-.86108	.21224	1.07331	.90242
.27300	1.34825	.94890	-.13540	.03021	-.77507	.09891	.87394	.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
.99384	.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	.15547	.57408	.53817
.85355	1.12953	.86275	-.07164	.06265	-.27395	.25749	.53143	.52337
.94550	1.05789	.82878	-.05481	.05861	-.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	.75053
.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	.04009	-.25056	.24417	.49473	.48902
.94550	1.05407	.83951	-.04053	.05290	-.11156	.30021	.41177	.41170
.99384	.92386	.81908	-.01286	.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	.16201	-1.46084	.66559	2.12544	1.95527
.05450	1.48317	.72948	-.16831	.11406	-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	.01520	-.73348	.01894	.75242	.63939
.57822	1.24425	.95599	-.08539	.02540	-.52822	.08784	.61606	.56524
.72700	1.17723	.92673	-.06595	.03055	-.37671	.14201	.51873	.49147
.85355	1.10608	.88340	-.04543	.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	-.21342	.15880	-1.42317	.65323	2.07640	1.89842
.05450	1.47072	.74320	-.16345	.10952	-1.06775	.45301	1.52076	1.34009
.14645	1.39008	.89050	-.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	.61575
.57822	1.24380	.96493	-.07805	.01912	-.52613	.06897	.59510	.54253
.72700	1.16994	.93394	-.05641	.02438	-.35969	.12862	.48831	.46096
.853	1.09360	.89724	-.03517	.02916	-.19373	.19753	.3912	.34662
.94550	1.02527	.87459	-.01674	.02953	-.05123	.23921	.29044	.28808
.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	.58942
.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	.84941	.01384	.02989	.18473	.28426	.09953	.08509

130

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	-.15349	.09045	-.95425	.42903	1.38328	1.20003
.14645	1.33789	.92284	-.11575	.02666	-.74604	.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	.06850

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	-.11523	.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	.15470
.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	-.62655	-.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	.36087
.85355	1.07190	.94282	-.05641	-.02406	-.15008	.11161	.26169	.25436
.94550	1.02281	.95564	-.03684	-.02089	-.04729	.08698	.13427	.12831
.99384	.90453	.88232	-.03878	-.03319	.18326	.22481	.04155	.03410

131

	** 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 PARAMETER **

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

** INDUCED DRAG = .02620

LISTING OF PROGRAM

```

A04NZXX,5,250,70000,482511,RCST5786P FANPOD          00000010
*NOTAPES,PSS                                           00000020
UPDATE(N)                                              00000030
RUN76(SA,I=COMPILE,NL50000)                          00000040
REWIND(NEWPL,LGO)                                    00000050
COPYPSS.                                              00000060
CXIT.                                                  00000070
REWIND(NEWPL,LGO)                                    00000080
COPYPSS.                                              00000090
.                                                       00000100
*DECK MAIN                                             00000110
PROGRAM FANPOD(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE9,TAPE10,TAPE11,TAPE12,TAPE16,TAPE24) 00000120
1TAPE19,TAPE20,TAPE21,TAPE22,TAPE23,TAPE11,TAPE12,TAPE16,TAPE24) 00000130
C               ** MAIN **                             00000140
C*             HIGH-SPEED INTERFERENCE EFFECTS ON V/STOL AIRCRAFT 00000150
C*             (SUBSONIC INTERFERENCE PRESSURE PROGRAM). 00000160
C                                                       00000170
COMMON/DAT/ DA(5000)                                  00000180
COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP               00000190
COMMON/CFG/ DG(15)                                   00000200
COMMON/CPB/ DB(13)                                   00000210
COMMON/CPF/ DF(13)                                   00000220
COMMON/CPW/ DW(18)                                   00000230
COMMON/CPD/ DP(12)                                   00000240
COMMON/CPN/ DN(21)                                   00000250
COMMON/CFK/ DK(14)                                   00000260
COMMON/CSF/ DS(90)                                   00000270
C                                                       00000280
LARGE          DM1(91812)                             00000290
LARGE          DM2(103466)                            00000300
C                                                       00000310
C*             INITIAL                                  00000320
PI=3.141593                                           00000330
PI4=12.56637                                          00000340
RC=57.29578                                           00000350
DO 10 I=1,5000                                        00000360
10 DA(I)=0.0                                          00000370
DO 12 I=3380,3387                                     00000380
12 .....I)=1000000.                                  00000390
C*             READ,PRINT AND TEST INPUT DATA        00000400
20 CALL RDATA                                         00000410
C*             CALC GEOMETRY                          00000420
CALL GEOM                                             00000430

```

135

C*	CALL MATA	CALC INFLUENCE MATRIX A	00000440
C*	CALL MATB	CALC BOUNDARY CONDITION MATRIX B	00000450
C*	CALL SOLU	SOLVE LINEAR SIMULTANEOUS EQUATIONS (AX=B)	00000460
C*	CALL PLCAL	CALC PRESSURE COEF. , INTEGRATED LOADS AND INDUCED DRAG	00000470
C*	CALL SKINF	CALC SKIN FRICTION	00000480
C*	GO TO NEXT CASE		00000490
	GO TO 20		00000500
	END		00000510
*DECK	CODIM		00000520
	SUBROUTINE CODIM (XI,YI,NI,T,ANS,NA)		00000530
C			00000540
C****		A CONTROLLED DEVIATION ITERPOLATION METHOD	00000550
C			00000560
	LARGE	XI(1) ,YI(1) ,T(1) ,ANS(1)	00000570
C			00000580
	DATA XK/0.5/		00000590
C			00000600
	N=NI		00000610
	DO 910 IE=1,NA		00000620
	X=T(IE)		00000630
100	IF(N-2)110,120,200		00000640
110	Y = YI(N)		00000650
	GO TO 900		00000660
120	Y = (YI(2)-YI(1))/(XI(2)-XI(1))* (X-XI(1)) +YI(1)		00000670
	GO TO 900		00000680
200	J = 1		00000690
210	IF(XI(J)-X)230,220,250		00000700
220	Y =YI(J)		00000710
	GO TO 900		00000720
230	J = J+1		00000730
	IF(J-N)210,210,250		00000740
250	IF(J-2)120,155,260		00000750
155	J = 3		00000760
	JJ = 1		00000770
	GO TO 285		00000780
260	IF(J-N)280,265,270		00000790
265	J = N-1		00000800
	JJ = 2		00000810
			00000820
			00000830
			00000840
			00000850
			00000860

	GO TO 285	00000870
270	Y= (YI(N)-YI(N-1))/(XI(N)-XI(N-1))* (X-XI(N-1))+YI(N-1)	00000880
	GO TO 900	00000890
280	JJ = 3	00000900
285	IF(N-3)290,290,295	00000910
290	J = 3	00000920
295	K = J-1	00000930
	M = K-1	00000940
	L = J+1	00000950
	A1 = X-XI(M)	00000960
	A2 = X-XI(K)	00000970
	A3 = X-XI(J)	00000980
	AL = (X-XI(K))/(XI(J)-XI(K))	00000990
	S = AL*YI(J)+(1.0-AL)*YI(K)	00001000
	C1= A3*A2/((XI(M)-XI(K))*(XI(M)-XI(J)))	00001010
	C2= A1*A3/((XI(K)-XI(M))*(XI(K)-XI(J)))	00001020
	C3= A2*A1/((XI(J)-XI(M))*(XI(J)-XI(K)))	00001030
	P1 = C1*YI(M)+C2*YI(K)+C3*YI(J)	00001040
	IF(N-3)305,305,310	00001050
305	P2 = P1	00001060
	GO TO 315	00001070
310	A4 = X-XI(L)	00001080
	C4= A4*A3/((XI(K)-XI(J))*(XI(K)-XI(L)))	00001090
	C5= A2*A4/((XI(J)-XI(K))*(XI(J)-XI(L)))	00001100
	C6= A3*A2/((XI(L)-XI(K))*(XI(L)-XI(J)))	00001110
	P2 = C4*YI(K)+C5*YI(J)+C6*YI(L)	00001120
315	GO TO (320,330,350),JJ	00001130
320	P2 = P1	00001140
	AL = (X-XI(1))/(XI(2)-XI(1))	00001150
	S = AL*YI(2)+ (1.0-AL)*YI(1)	00001160
	P1= S + XK*(P2-S)	00001170
	GO TO 350	00001180
330	P1 = P2	00001190
	AL = (X-XI(N-1))/(XI(N)-XI(N-1))	00001200
	S = AL*YI(N) +(1.0-AL)*YI(N-1)	00001210
	P2 = S+ XK*(P1-S)	00001220
350	E1 = ABS(P1-S)	00001230
	E2 = ABS(P2-S)	00001240
	IF(E1+E2)400,400,410	00001250
400	Y = S	00001260
	GO TO 900	00001270
410	BT = (E1*AL)/(E1*AL+(1.0-AL)*E2)	00001280
	Y = BT*P2+(1.0-BT)*P1	00001290

900	ANS(IE)=Y	00001300
910	CONTINUE	00001310
	RETURN	00001320
	END	00001330
*DECK	WINGD	00001340
	SUBROUTINE WINGD (NETA,ETA,XL,XTXL,CR)	00001350
C		00001360
C*	TABLE LOOKUP OF PLANAR PLANFORM FOR X LE AND CHORD	00001370
C		00001380
	LARGE ETA(1),XL(1),XTXL(1),CR(1)	00001390
C		00001400
	DO 40 I=1,NETA	00001410
	L=1	00001420
	ET=ETA(I)	00001430
	IF(ET.LT.0.0) ET=ABS(ET)	00001440
	IF(ET.LE.CR(L)) GO TO 38	00001450
32	IF(ET-CR(L)) 37,38,34	00001460
34	L=L+3	00001470
	IF(L-60) 32,42,42	00001480
37	RATIO=(CR(L)-ET)/(CR(L)-CR(L-3))	00001490
	XL(I)=CR(L+1)-(CR(L+1)-CR(L-2))*RATIO	00001500
	XTXL(I)=CR(L+2)-(CR(L+2)-CR(L-1))*RATIO-XL(I)	00001510
	GO TO 40	00001520
38	XL(I)=CR(L+1)	00001530
	XTXL(I)=CR(L+2)-XL(I)	00001540
40	CONTINUE	00001550
	RETURN	00001560
42	WRITE (6,1000)(ETA(K),K=1,NETA)	00001570
	CALL EXIT	00001580
	STOP	00001590
1000	FORMAT(23H1 ** WINGD * ETA VALUES/(1H010F10.3))	00001600
	END	00001610
*DECK	PFUNC	00001620
	SUBROUTINE PFUNC (IW,NEV,EV,W,S)	00001630
C		00001640
C*	CALC. MATRIX S OF P FUNCTION	00001650
C		00001660
	COMMON/DAT/ DA(5000)	00001670
C		00001680
	DIMENSION W(1)	00001690
C		00001700
	LARGE EV(1),S(51,1)	00001710
C		00001720

	COMMON/CRG/ PI,PI4,RC,BETA	00001730
C	IC=W(IW)+.5	00001740
	IND=DA(IC)	00001750
	IF(IND) 20,30,20	00001760
20	EL=DA(IC+2)	00001770
	ER=DA(IC+3)	00001780
	GO TO 201	00001790
C	P-FUNCTION TYPE LOAD SHAPE	00001800
30	ETAS0 =DA(IC+1)	00001810
	RL=DA(IC+2)	00001820
	RR=DA(IC+3)	00001830
	ETAS=ETAS0	00001840
33	IF(ETAS0-RL) 40,41,41	00001850
41	IF(1.0-RR-ETAS0) 42,56,56	00001860
40	A10=1.0-ETAS0/RL	00001870
	B10=1.0/RL	00001880
	C10=-(RL+RR)*(1.0-ETAS0)/(RR*RL)	00001890
	E10=((RL+RR)/(RR*RL)-1.0/RL)*(1.0-(ETAS0+RR))	00001900
	D10=0.0	00001910
	GO TO 43	00001920
56	D10=(1.0-(ETAS0-RL))/RL	00001930
	C10=-(1.0/RR+1.0/RL)*(1.0-ETAS0)	00001940
	E10=(1.0-(ETAS0+RR))/RR	00001950
	A10=0.0	00001960
	B10=0.0	00001970
	GO TO 43	00001980
42	D10=(1.0-(ETAS0-RL))/RL	00001990
	C10=-(1.0/(1.0-ETAS0)+1.0/RL)*(1.0-ETAS0)	00002000
	A10=0.0	00002010
	B10=0.0	00002020
	E10=0.0	00002030
43	DO 200 IE=1,NEV	00002040
	ETA=EV(IE)	00002050
	IF(ETA,NE,0.0) GO TO 27	00002060
10	ETA=ETA-0.0000001	00002070
27	D=SQRT(1.0-ETA**2)	00002080
	IF(D.EQ.1.) D=.9999999	00002090
28	ETAS = ETAS0	00002100
	IF(ETAS0-RL) 44,45,45	00002110
44	ETAS=0.0	00002120
	C = SQRT(1.0-ETAS**2)	00002130
	E =(ETA*C -ETAS*D)/(ETA*C +ETAS*D)	00002140
		00002150

G	= (1.0+ETA*ETAS+D *C) / (1.0-ETA*ETAS-D *C)	00002160
IF(E)	9,10,11	00002170
9 E	= ABS(E)	00002180
11 IF(G)	12,10,14	00002190
12 G	= ABS(G)	00002200
14 EMTPH	= ((ETAS-ETA)*ALOG(E)-ETAS*ALOG(G)+2.*(ACOS(ETAS))*D)/PI	00002210
45 L	= 0	00002220
49 K	= 0	00002230
5 K	= K+1	00002240
C	= SQRT(1.0-ETAS**2)	00002250
H	= SQRT(ABS((1.0-ETA*ETAS-D *C) / (1.0-ETA*ETAS+D *C)))	00002260
O	= SQRT(ABS((1.0+ETA*ETAS-D *C) / (1.0+ETA*ETAS+D *C)))	00002270
IF(H)	16,10,16	00002280
16 IF(O)	18,10,18	00002290
18 PS	= -(((ETAS-ETA)**2)* ALOG(H) + ((ETAS+ETA)**2)* ALOG(O)	00002300
1	+ (4.*ETAS*(ACOS(ETAS)) - 2.*C)*D) / (6.28318*(1.-ETAS))	00002310
39 IF(ETASO-RL)	46,47,47	00002320
46 IF(L)	48,48,47	00002330
48 PSZERO	= PS	00002340
ETAS	= ETASO	00002350
L	= L+1	00002360
GO TO	49	00002370
47 IF(K-2)	2,3,4	00002380
2 PSO	= PS	00002390
IF(ETASO-RL)	50,51,51	00002400
51 IF(1.0-RR-ETASO)	52,50,50	00002410
50 ETAS	= ETASO+RR	00002420
GO TO	5	00002430
52 ETAS	= ETASO-RL	00002440
GO TO	5	00002450
3 IF(ETASO-RL)	53,55,55	00002460
55 IF(1.0-RR-ETASO)	57,58,58	00002470
58 PSRR	= PS	00002480
ETAS	= ETASO-RL	00002490
GO TO	5	00002500
53 PSRR	= PS	00002510
PSRL	= 0.0	00002520
GO TO	59	00002530
57 PSRL	= PS	00002540
PSRR	= 0.0	00002550
EMTPH	= 0.0	00002560
PSZERO	= 0.0	00002570
GO TO	59	00002580

```

4 PSRL =PS 00002590
  EMTPH =0.0 00002600
  PSZERO =0.0 00002610
59 PS =A10*EMTPH +B10*PSZERO +C10*PS0 +D10*PSRL +E10*PSRR 00002620
200 S(IE,IW)=PS 00002630
  GO TO 300 00002640
C FLAP OR KRUEGER TYPE LOAD SHAPE 00002650
201 PHII= ACOS(EL) 00002660
  PHIO= ACOS(ER) 00002670
  DO 250 IE=1,NEV 00002680
  ETA=EV(IE) 00002690
  PHI= ACOS(ETA) 00002700
  P1=ABS(PHI-PHII)/2.0 00002710
  P2=(PHI+PHII)/2.0 00002720
  IF(P1.NE.0.0) GO TO 202 00002730
  C1=0.0 00002740
  GO TO 204 00002750
202 C1=(EL-ETA)*ALOG(SIN(P1)/SIN(P2)) 00002760
204 C2=(EL+ETA)*ALOG(COS(P2)/COS(P1)) 00002770
  P1=ABS(PHI-PHIO)/2.0 00002780
  P2=(PHI+PHIO)/2.0 00002790
  IF(P1.NE.0.0) GO TO 206 00002800
  C3=0.0 00002810
  GO TO 208 00002820
206 C3=(ER-ETA)*ALOG(SIN(P1)/SIN(P2)) 00002830
208 C4=(ER+ETA)*ALOG(COS(P2)/COS(P1)) 00002840
210 S(IE,IW)=(C1+C2-C3-C4+2.*SIN(PHI)*(PHII-PHIO))/PI 00002850
250 CONTINUE 00002860
300 RETURN 00002870
  END 00002880
*DECK MSOLX 00002890
  SUBROUTINE MSOLX (NKT,NQT,NCB,AA,B,ID) 00002900
C 00002910
C* HOUSEHOLDER METHOD FOR SOLVING SET OF LINEAR 00002920
C* SIMULTANEOUS EQUATIONS 00002930
C 00002940
C LARGE B(ID,1),AA(ID,1) 00002950
C 00002960
C LARGE DUM(91812),A(102,103),AR(204) ,IL(102) 00002970
C 00002980
C NKTP=NKT+NCB 00002990
  DO 10 J=1,NKT 00003000
  DO 10 K=J,NKTP 00003010

```

141

10	A(J,K) = 0.0	00003020
	DO 60 K=1,NKT	00003030
20	DO 22 I=1,NKT	00003040
22	AR(I)=AA(K,I)	00003050
	DO 24 J=1,NCB	00003060
	NKTJ=NKT+J	00003070
24	AR(NKTJ)=B(K,J)	00003080
		00003090
C		00003100
30	DO 50 I=1,NKT	00003110
	R = SQRT(A(I,I) ** 2 + AR(I) **2)	00003120
	IF(R .EQ. 0.0) GO TO 50	00003130
	C = A(I,I) / R	00003140
	S = AR(I) / R	00003150
	DO 40 J=I,NKTP	00003160
	T2 = C * A(I,J) + S * AR(J)	00003170
	AR(J) = -S * A(I,J) + C * AR(J)	00003180
40	A(I,J) = T2	00003190
50	CONTINUE	00003200
60	CONTINUE	00003210
	II = 1	00003220
	DO 80 I=1,NKT	00003230
	IF(A(I,I) .LE. 0.0000001) GO TO 70	00003240
	IL(I) = II	00003250
	II = II + 1	00003260
	GO TO 80	00003270
70	IL(I) = 0	00003280
80	CONTINUE	00003290
	DO 250 J=1,NCB	00003300
	NKTJ=NKT+J	00003310
	DO 90 I=1,NKT	00003320
90	AR(I)=0.0	00003330
	II=NKT	00003340
	DO 210 I=1,NKT	00003350
	IF(IL(II) .LE. 0) GO TO 210	00003360
	J1 = IL(II)	00003370
	IF(II - NKT) 170, 200, 220	00003380
170	IK = II + 1	00003390
	DO 180 K=IK,NKT	00003400
180	AR(II) = AR(II) - A(J1,K) * AR(K)	00003410
200	AR(II)=(AR(II)+A(J1,NKTJ))/A(J1,II)	00003420
210	II = II - 1	00003430
220	CONTINUE	00003440
	DO 240 I=1,NKT	

```

240 AA(I,J)=AR(I)                                00003450
250 CONTINUE                                       00003460
      RETURN                                       00003470
      END                                           00003480
*DECK RDATA                                        00003490
      SUBROUTINE RDATA                              00003500
C                                                    00003510
C*          READ,PRINT AND TEST INPUT DATA        00003520
C                                                    00003530
      COMMON/DAT/ DA(5000)                          00003540
      DIMENSION WJS(1)                              00003550
      EQUIVALENCE (DA(4),XMACH) ,(DA(12),PRII) ,(DA(1205),WNVSO) 00003560
2,(DA(1206),WNVSI),(DA(1690),WJS) ,(DA(1270),WNVC),(DA(1271),WNVS) 00003570
3,(DA(1272),WNPC),(DA(1273),WNPS) ,(DA(1274),FNVX),(DA(1275),FNVY) 00003580
4,(DA(1276),FNDV),(DA(1277),WNJC) ,(DA(1278),WNJS),(DA(1279),FNJX) 00003590
5,(DA(1280),WNU) ,(DA(1281),WNW) ,(DA(1282),FNF) ,(DA(2492),PDA) 00003600
6,(DA(2507),PNVC),(DA(2508),PNVS) ,(DA(2509),PNU) ,(DA(2511),PNJC) 00003610
7,(DA(3390),XNB) ,(DA(3394),XNP1) ,(DA(3395),XNP2),(DA(4705),BNVX) 00003620
8,(DA(4706),BNVY),(DA(4707),BNDV) ,(DA(4708),BNJX),(DA(4709),BNF) 00003630
      00003640
      COMMON/CFG/ IDB,IDF,IDW,IDP,IDN,NQB,NQF,NQW,NQP,NQN 00003650
I      ,NKB,NKF,NKW,NKP,NKN 00003660
      COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP 00003670
      COMMON/CPB/ NVXB,NVXMB,NVYB,NDVB,NVBB,NVBPB,NFB,NJXB 00003680
      COMMON/CPF/ NVXF,NVXMF,NVYF,NDVF,NVBF,NVBPB,NFF,NJXF 00003690
      COMMON/CPW/ DW1(9),NU,NW,NVC,NVS,NJC,NJS,NVSO,NVSI 00003700
      COMMON/ CPP/ DP1(7),NVCP,NVSP,NJCP,NUP 00003710
      COMMON/CPN/ NB,NP1,NP2,NOBP,NT 00003720
C                                                    00003730
C*          READ INPUT DATA                        00003740
      CALL DECRD (DA(1)) 00003750
C*          PRINT INPUT DATA                       00003760
      J=1 00003770
      WRITE (6,10) J,(DA(I),I=1,4) 00003780
10 FORMAT (59H1 ** SUBSONIC INTERFERENCE PRESSURE PROGRAM * INPUT DAT 00003790
1A **//1H 16,4F19.6) 00003800
      DO 20 I=5,4996,5 00003810
      IF(DA(I) .NE.0.0) GO TO 12 00003820
      IF(DA(I+1).NE.0.0) GO TO 12 00003830
      IF(DA(I+2).NE.0.0) GO TO 12 00003840
      IF(DA(I+3).NE.0.0) GO TO 12 00003850
      IF(DA(I+4).NE.0.0) GO TO 12 00003860
      GO TO 20 00003870

```

143

```
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
```

```
00003880
00003890
00003900
00003910
00003920
00003930
00003940
00003950
00003960
00003970
00003980
00003990
00004000
00004010
00004020
00004030
00004040
00004050
00004060
00004070
00004080
00004090
00004100
00004110
00004120
00004130
00004140
00004150
00004160
00004170
00004180
00004190
00004200
00004210
00004220
00004230
00004240
00004250
00004260
00004270
00004280
00004290
00004300
```

```
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
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

```

```

00004310
00004320 ← 4325
00004330
00004340
00004350
00004360
00004370
00004380
00004390
00004400
00004410
00004420
00004430
00004440
00004450
00004460
00004470
00004480
00004490
00004500
00004510
00004520
00004530
00004540
00004550
00004560
00004570
00004580
00004590
00004600
00004610
00004620
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
00004730

```

	NQP=NJCP*NVSP	00004740
C*	INITIAL NACELLE	00004750
70	IF(IDN.EQ.0) GO TO 100	00004760
	REWIND 9	00004770
	REWIND 10	00004780
	REWIND 11	00004790
	REWIND 16	00004800
	NB=XNB	00004810
	NP1=XNP1	00004820
	NT=NP1-1	00004830
	IF(NB.EQ.1) GO TO 71	00004840
	NP2=XNP2	00004850
	NT=NT+NP2-1	00004860
71	NKN=NT	00004870
	NQN=NT	00004880
	IF(NT.LE.140) GO TO 100	00004890
	WRITE (6,72)	00004900
72	FORMAT (30H0 ** NACELLE * TOO MANY POINTS)	00004910
	GO TO 200	00004920
C*	TEST TOTAL POINTS	00004930
100	NKT=NKB+NKF+NKW+NKP	00004940
	IF(NKT.LE.450) GO TO 110	00004950
	WRITE (6,102)	00004960
102	FORMAT (30H0 ** RDATA * TOO MANY UNKNOWNNS)	00004970
	GO TO 200	00004980
110	NOBP=NQB+NQF+NQW+NQP	00004990
	IF(NOBP.LE.860) GO TO 120	00005000
	WRITE (6,112)	00005010
112	FORMAT (30H0 ** RDATA * TOO MANY EQUATIONS)	00005020
	GO TO 200	00005030
120	IF(XMACH.LT.1.) GO TO 210	00005040
	WRITE (6,130)	00005050
130	FORMAT (33H0 ** MACH GREATER OR EQUAL TO 1.0)	00005060
200	CALL EXIT	00005070
210	BETA=SQRT(1.-XMACH**2)	00005080
C	INITIAL PYLON DIHEDRAL ANGLE	00005090
	SINDP=0.0	00005100
	COSDP=1.0	00005110
	IF(IDP.EQ.0) GO TO 220	00005120
	PDAR=PDA/RC	00005130
	SINDP=SIN(PDAR)	00005140
	COSDP=COS(PDAR)	00005150
220	RETURN	00005160

```

END 00005170
*DECK DECRD 00005180
SUBROUTINE DECRD (DATA) 00005190
C 00005200
C THE FUNCTION OF THIS ROUTINE IS TO READ SINGLE-PRECISION REAL NUMBERS 00005210
C FROM CARD COLUMNS 13-72 WITH A 5E12.0 FORMAT. THE DATA IS STORED INTO 00005220
C THE ADDRESS OF THE FIRST ARGUMENT MINUS ONE PLUS THE CONTENTS OF CARD 00005230
C COLUMNS 2-12 AND THE FOLLOWING FOUR LOCATIONS. IF ANY DATA ITEM 00005240
C CONTAINS 12 BLANKS, NO DATA IS STORED. CARDS ARE READ AND DATA IS 00005250
C STORED UPTO AND INCLUDING A CARD WITH A MINUS IN COLUMN ONE. 00005260
C 00005270
C 00005280
C 00005290
C DIMENSION DATA(1),ADATA(5),IDATA(17),IIDATA(8) 00005300
C DATA IBLANK/10H / 00005310
C READ A CARD AND TEST IADD. ***** 00005320
15 READ(5,16)IIDATA 00005330
16 FORMAT(8A10) 00005340
IF(EOF(5)) 60,19,60 00005350
19 DECODE(72,17,IIDATA)IADD,ADATA 00005360
17 FORMAT(I12,5G12.0) 00005370
DECODE(80,18,IIDATA)IDATA 00005380
18 FORMAT(12X,17A4) 00005390
J=IADD 00005400
IF(IADD) 22,40,24 00005410
22 J=-J 00005420
C TEST FOR BLANK FIELDS AND STORE NON-BLANK FIELDS. ***** 00005430
24 DO 30 I=1,5 00005440
L=3*I 00005450
K=L-2 00005460
DO 26 M=K,L 00005470
IF(IDATA(M)-IBLANK)28,26,28 00005480
26 CONTINUE 00005490
GO TO 30 00005500
28 DATA(J)=ADATA(I) 00005510
30 J=J+1 00005520
IF(IADD)100,40,15 00005530
C ERROR PRINT OF BAD CARD IMAGE. ***** 00005540
40 WRITE(6,50)IADD,IDATA 00005550
50 FORMAT(17H0DECRD ER. CARD=(,I112,17A4,2H).) 00005560
60 CALL EXIT 00005570
100 RETURN 00005580
END 00005590
*DECK GEOM

```

147

SUBROUTINE GEOM

C		00005600
C*	CONTROL PROGRAM FOR GEOMETRY ROUTINES	00005610
C		00005620
	COMMON/DAT/ DA(5000)	00005630
	DIMENSION WJC(1),WJS(1)	00005640
	EQUIVALENC (DA(1660),WJC) ,(DA(1690),WJS)	00005650
	1,(DA(87),FOX) ,(DA(1460),FVX) ,(DA(2490),ATP)	00005660
	2,(DA(4015),BOX) ,(DA(4735),BVX)	00005670
C		00005680
	COMMON/CFG/ IDB, IDF, IDW, IDP, IDN	00005690
	COMMON/CPB/ NVXB, NVXMB, NVYB, NDVB, NVBB, NVBPB, NB	00005700
	COMMON/CPF/ NVXF, NVXMF, NVYF, NDVF, NVBF, NVBPF, NF	00005710
	COMMON/CPW/ DW1(13), NJC, NJS	00005720
C		00005730
	LARGE BOU(3000) ,DB1(13000),XBB(4000) ,YBB(4000) ,ZBB(4000)	00005740
*	,DF1(11700),XBF(4000) ,YBF(4000) ,ZBF(4000)	00005750
1	,EV(52) ,XLV(52) ,XTLV(52)	00005760
2	,DM1(1550) ,EP(30) ,XTLP(30) ,XOCP(30)	00005770
3	,DM2(10850),CFJ(50)	00005780
C		00005790
C*	START	00005800
	IF(IDB.NE.1) GO TO 10	00005810
	CALL BXYZ	00005820
	CALL BQPTS	00005830
	IF(NB.GT.0) CALL BCLS	00005840
C		00005850
	10 IF(IDF.NE.1) GO TO 20	00005860
	CALL FXYZ	00005870
	CALL FQPTS	00005880
	IF(NF.GT.0) CALL FCLS	00005890
C		00005900
	20 IF(IDW.NE.1) GO TO 30	00005910
	CALL WXYZ	00005920
	CALL RTWI	00005930
	CALL MATCS	00005940
	DO 22 J=1,NJS	00005950
	IE=WJS(J)	00005960
	EP(J)=EV(IE)	00005970
	22 XTLP(J)=XTLV(IE)	00005980
	DO 24 J=1,NJC	00005990
	IJ=WJC(J)	00006000
	24 XOCP(J)=CFJ(IJ+1)	00006010
		00006020

```

IF(DA(2000).NE.0.0) CALL WTZCS                                00006030
C
30 IF(IDP.NE.1) GO TO 40                                     00006040
   IF(ATP.GT.1.0) GO TO 34                                  00006050
   CALL PXYZ (NVXB,NDVB,NVBPB,BOX,BVX,XBB,YBB,ZBB)          00006060
   GO TO 36                                                  00006070
34 CALL PXYZ (NVXF,NDVF,NVBPF,FOX,FVX,XBF,YBF,ZBF)          00006080
36 IF(DA(3000).NE.0.0) CALL PTZCS                            00006090
C
40 IF(IDN.NE.1) GO TO 50                                     00006100
   CALL NXYZ                                                00006110
C
50 RETURN                                                    00006120
   END                                                       00006130
*DECK BXYZ                                                  00006140
SUBROUTINE BXYZ                                             00006150
C
C*          CALC. COORDINATE OF BODY VORTICES (XB,YB,ZB)    00006160
C
COMMON/DAT/ DA(5000)                                       00006170
C
DIMENSION XS(1),TYS(1),RZS(1),FVX(1),FVT(1),FJX(1),YYM(1) 00006180
1          ,XMC(1),ZZM(1),FZC(1)                          00006190
EQUIVALENCE (DA(4005),FCD),(DA(4007),FMF1),(DA(4008),FCPI) 00006200
1,(DA(4009),FLOI),(DA(4010),FTHI),(DA(4015),FOX),(DA(4016),FOY) 00006210
2,(DA(4017),FOZI),(DA(4020),FNXS),(DA(4021),XS),(DA(4050),FNTY) 00006220
3,(DA(4051),TYS),(DA(4090),RZS),(DA(4565),FNXM),(DA(4566),XMC) 00006230
4,(DA(4600),YYM),(DA(4635),ZZM),(DA(4670),FZC),(DA(4705),FNVX) 00006240
5,(DA(4707),FNDV),(DA(4735),FVX),(DA(4885),FVT),(DA(4905),FJX) 00006250
C
COMMON/CRG/ PI,PI4,RC,BETA                                  00006260
COMMON/CPB/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX,IFBE,SINA,COSA,FAREA 00006270
*          ,SA                                              00006280
C
LARGE      BOU(3000),DM1(11700),AXY(650),AYZ(650),XB(4000) 00006290
*,YB(4000),ZB(4000),FAN(23700),WIN(23430),PYL(6061),CEL(3700) 00006300
1,STH(400),CTH(400),TH(400)                                00006310
2,STC(150),CTC(150),VXI(150),VX2(150),ZC1(150),ZC2(150) 00006320
3,RS(30),STS(30),ZM(150),YM(150),ZC(150)                 00006330
4,YMAX(150),FVXL(150),XMCL(34),YMCL(34),ZZML(34),FZCL(34) 00006340
C
C          INITIAL                                          00006350
C
NXS=FNXS                                                    00006360

```


	NTY=FNTY	00006460
	NXM=FNXM	00006470
	FNVXM=FNVX-1.	00006480
	FNVB=NVB	00006490
C*	CALC LONGITUDINAL PARAMETER X	00006500
	FVX(1)=0.0	00006510
	FVX(NVX)=FCD	00006520
	IF(FLOI) 30,10,50	00006530
C	EVEN DELTA PHI	00006540
10	DPHI=PI/(FNVXM-1.)	00006550
	PHX=DPHI/2.	00006560
	FCD2=FCD/2.	00006570
	DO 20 I=2,NVXM	00006580
	FVX(I)=FCD2-FCD2*COS(PHX)	00006590
20	PHX=PHX+DPHI	00006600
	GO TO 50	00006610
C	EVEN DELTA X	00006620
30	DX=FCD/(FNVXM-1.)	00006630
	FVX(2)=DX/2.	00006640
	DO 40 I=3,NVXM	00006650
40	FVX(I)=FVX(I-1)+DX	00006660
C*	CALC LATERAL PARAMETERS	00006670
50	IF(FTHI) 120,120,60	00006680
C	INPUT THETA LIST	00006690
60	IF(FNDV.GT.1.) GO TO 80	00006700
	DO 70 I=1,NVY	00006710
70	TH(I)=FVT(I)	00006720
	TH(NVY+1)=180.	00006730
	GO TO 140	00006740
80	IC=0	00006750
	FVT(NVY+1)=180.	00006760
	DO 100 I=1,NVY	00006770
	DTH=(FVT(I+1)-FVT(I))/FNDV	00006780
	IC=IC+1	00006790
	TH(IC)=FVT(I)	00006800
	DO 100 J=2,NDV	00006810
	IC=IC+1	00006820
100	TH(IC)=TH(IC-1)+DTH	00006830
	TH(IC+1)=180.	00006840
	GO TO 140	00006850
C	EVEN DELTA THETA	00006860
120	DTH=180./FNVB	00006870
	TH(1)=0.	00006880

	DO 130 I=2,NVBP	00006890
130	TH(I)=TH(I-1)+DTH	00006900
C	SIN-COS THETA	00006910
140	DO 150 I=1,NVBP	00006920
	THR=TH(I)/RC	00006930
	STH(I)=SIN(THR)	00006940
150	CTH(I)=COS(THR)	00006950
C*	SHIFT ARRAYS TO LCM FOR CODIM	00006960
	DO 151 I=1,NVX	00006970
151	FVXL(I)=FVX(I)	00006980
	DO 152 I=1,NXM	00006990
	XMCL(I)=XMC(I)	00007000
	YMCL(I)=YYM(I)	00007010
	ZZML(I)=ZZM(I)	00007020
152	FZCL(I)=FZC(I)	00007030
C*	CALC CAMBER AND MUTIPLICATION FACTORS	00007040
	IF(NXM.NE.0) GO TO 154	00007050
	DO 153 I=1,NVX	00007060
	YM(I)=1.	00007070
	ZM(I)=1.	00007080
153	ZC(I)=0.0	00007090
	GO TO 192	00007100
154	CALL CODIM (XMCL,YMCL,NXM,FVXL,YM,NVX)	00007110
	IF(ABS(FMFI).EQ.1.0) GO TO 180	00007120
160	DO 170 I=1,NVX	00007130
170	ZM(I)=YM(I)	00007140
	GO TO 190	00007150
180	CALL CODIM (XMCL,ZZML,NXM,FVXL,ZM,NVX)	00007160
190	CALL CODIM (XMCL,FZCL,NXM,FVXL,ZC,NVX)	00007170
192	IF(FCPI) 220,220,200	00007180
C	PERPENDICULAR TO X-AXIS	00007190
200	DO 210 I=2,NVX	00007200
	STC(I)=0.0	00007210
210	CTC(I)=1.	00007220
	GO TO 250	00007230
C	PERPENDICULAR TO CAMBER LINE	00007240
220	DX=FCD/1000.	00007250
	DXH=DX/2.	00007260
	DO 230 I=2,NVX	00007270
	VX1(I-1)=FVX(I)+DXH	00007280
230	VX2(I-1)=FVX(I)-DXH	00007290
	CALL CODIM (XMCL,FZCL,NXM,VX1,ZC1,NVXM)	00007300
	CALL CODIM (XMCL,FZCL,NXM,VX2,ZC2,NVXM)	00007310

	DX2=DX*DX	00007320
	DO 240 I=2,NVX	00007330
	DZ=ZC2(I-1)-ZC1(I-1)	00007340
	DH=SQRT(DX2+DZ**2)	00007350
	STC(I)=DZ/DH	00007360
240	CTC(I)=DX/DH	00007370
C*	TABLE LOOKUP FOR RADIUS AT EACH POINT	00007380
C	INPUT (R VS THETA) VS X	00007390
250	CALL CODMT (NXS,NTY,NVB,TYS,RZS,TH,XB)	00007400
	L=1	00007410
	DO 350 I=2,NVX	00007420
	DO 310 J=2,NXS	00007430
	IF(FVX(I)-XS(J)) 320,320,310	00007440
310	CONTINUE	00007450
	RAT=0.0	00007460
	GO TO 330	00007470
320	RAT=(XS(J)-FVX(I))/(XS(J)-XS(J-1))	00007480
330	K=J	00007490
	DO 340 IT=1,NVB	00007500
	L=L+1	00007510
	ZB(L)=XB(K)-RAT*(XB(K)-XB(K-1))	00007520
340	K=K+NXS	00007530
	L=L+1	00007540
	L1=L-NVB	00007550
	ZB(L)=ZB(L1)	00007560
350	CONTINUE	00007570
C*	CALC XB,YB,ZB AND INTEGRATED LOAD PARAMETERS	00007580
	XB(1)=0.0	00007590
	YB(1)=0.0	00007600
	ZB(1)=0.0	00007610
	YMAX(1)=0.0	00007620
	FAREA=0.0	00007630
	L=1	00007640
	DO 390 IX=2,NVX	00007650
	C1=ZM(IX)*CTC(IX)	00007660
	IF(FMFI.LT.0.0) C1=YM(IX)*CTC(IX)	00007670
	YMAX(IX)=0.0	00007680
	DO 360 IY=1,NVBP	00007690
	L=L+1	00007700
	C2=ZB(L)*CTH(IY)	00007710
	IF(FMFI.GE.0.0) GO TO 352	00007720
	YB(L)=ZC(IX)+C1*ZB(L)*STH(IY)	00007730
	ZB(L)=C2*ZM(IX)	00007740

	GO TO 354	00007750
	352 YB(L)=YM(IX)*ZB(L)*STH(IY)	00007760
	ZB(L)=ZC(IX)+C1*C2	00007770
	354 XB(L)=FVX(IX)-C2*STC(IX)	00007780
	356 IF(YB(L).LE.YMAX(IX)) GO TO 360	00007790
	YMAX(IX)=YB(L)	00007800
	360 CONTINUE	00007810
	FAREA=(YMAX(IX)+YMAX(IX-1))*(FVX(IX)-FVX(IX-1))+FAREA	00007820
	390 CONTINUE	00007830
	IFBE=0	00007840
	IF(YMAX(NVX).GT.0.001) IFBE=1	00007850
C*	SETUP FOR COEFFICIENTS	00007860
C*	AXY AND AYZ=2*PROJECTED AREA	00007870
	IJ=1	00007880
	JX=FJX(1)	00007890
	NC=0	00007900
	SA=0.0	00007910
	M2=2	00007920
C	DO 460 IX=1,NVXM	00007930
	IF(IX.LE.JX) GO TO 410	00007940
	IF(IJ.GE.NJX) GO TO 410	00007950
	IJ=IJ+1	00007960
	JX=FJX(IJ)	00007970
		00007980
		00007990
C	410 DO 450 IY=1,NVY	00008000
	IF(IX.NE.JX) GO TO 420	00008010
	NC=NC+1	00008020
	AXY(NC)=0.0	00008030
	AYZ(NC)=0.0	00008040
		00008050
C	420 DO 450 I=1,NDV	00008060
	IF(IX.GT.1) GO TO 430	00008070
	AX=YB(M2)*XB(M2+1)-YB(M2+1)*XB(M2)	00008080
	AY=YB(M2)*ZB(M2+1)-YB(M2+1)*ZB(M2)	00008090
	AZ=XB(M2)*ZB(M2+1)-XB(M2+1)*ZB(M2)	00008100
	GO TO 440	00008110
		00008120
C	430 M1=M2-NVBP	00008130
	X4=XB(M2+1)-XB(M1)	00008140
	X3=XB(M1+1)-XB(M2)	00008150
	Y4=YB(M2+1)-YB(M1)	00008160
	Y3=YB(M1+1)-YB(M2)	00008170

	IC=0	00008610
	K1=NVB*P+NDVH-1	00008620
	DO 60 IY=1,NVY	00008630
	K2=IY*NDV-K1	00008640
	DO 50 IX=1,NJX	00008650
	JX=FJX(IX)	00008660
	J2=JX*NVB*P+K2	00008670
	J4=J2+1	00008680
	IF(JX-1) 10,10,20	00008690
10	J3=1	00008700
	J1=1	00008710
	GO TO 30	00008720
20	J1=J2-NVB*P	00008730
	J3=J1+1	00008740
30	IC=IC+1	00008750
	XQ(IC)=(XB(J1)+XB(J2)+XB(J3)+XB(J4))/4.	00008760
	YQ(IC)=(YB(J1)+YB(J2)+YB(J3)+YB(J4))/4.	00008770
	ZQ(IC)=(ZB(J1)+ZB(J2)+ZB(J3)+ZB(J4))/4.	00008780
C		00008790
	TM1=XB(J2)-XB(J1)+XB(J4)-XB(J3)	00008800
	TM2=YB(J2)-YB(J1)+YB(J4)-YB(J3)	00008810
	TM3=ZB(J2)-ZB(J1)+ZB(J4)-ZB(J3)	00008820
	TT1=XB(J3)-XB(J1)+XB(J4)-XB(J2)	00008830
	TT3=ZB(J3)-ZB(J1)+ZB(J4)-ZB(J2)	00008840
	TT2=YB(J3)-YB(J1)+YB(J4)-YB(J2)	00008850
C		00008860
	S2143 =SQRT(TM1**2+TM2**2+TM3**2)	00008870
	S3142=SQRT(TT1**2+TT2**2+TT3**2)	00008880
C		00008890
	TMX(IC)=TM1/S2143	00008900
	TMY(IC)=TM2/S2143	00008910
	TMZ(IC)=TM3/S2143	00008920
	TTX(IC)=TT1/S3142	00008930
	TTY(IC)=TT2/S3142	00008940
	TTZ(IC)=TT3/S3142	00008950
C		00008960
	TX=TMY(IC)*TTZ(IC)-TMZ(IC)*TTY(IC)	00008970
	TY=TMZ(IC)*TTX(IC)-TMX(IC)*TTZ(IC)	00008980
	TZ=TMX(IC)*TTY(IC)-TMY(IC)*TTX(IC)	00008990
	SRXYZ=SQRT(TX**2+TY**2+TZ**2)	00009000
	XN(IC)=TX/SRXYZ	00009010
	YN(IC)=TY/SRXYZ	00009020
50	ZN(IC)=TZ/SRXYZ	00009030

60	CONTINUE	00009040
	RETURN	00009050
	END	00009060
*DECK	BCLS	00009070
	SUBROUTINE BCLS	00009080
C		00009090
C*	CALC CHORDWISE LOAD SHAPES FOR THE FUSELAGE	00009100
C		00009110
	COMMON/DAT/ DA(5000)	00009120
	DIMENSION FVX(1),F(1)	00009130
	EQUIVALENCE (DA(4710),F) ,(DA(47351),FVX) ,(DA(40051),FCD)	00009140
C		00009150
	COMMON/CPB/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF	00009160
C		00009170
	LARGE BOU(3000) ,DM1(7800) ,PM(150,25)	00009180
1	,DM2(70341),PHI(150) ,CPH(150) ,XOC(150)	00009190
C*	START	00009200
C	CALC X/C , COS(PHI) AND PHI	00009210
	DO 10 I=2,NVXM	00009220
	XOC(I)=FVX(I)/FCD	00009230
	CPH(I)=1.-2.*XOC(I)	00009240
10	PHI(I)= ACOS(CPH(I))	00009250
	IF(XOC(NVXM).LT.1.0) GO TO 30	00009260
	WRITE (6,20)	00009270
20	FORMAT (32H1 ** BCLS * X/C GREATER THAN 1.0)	00009280
	CALL EXIT	00009290
C*	NO. OF LOAD SHAPE LOOP	00009300
30	DO 200 J=1,NF	00009310
	DO 90 I=2,NVXM	00009320
	IF(F(I).GE.1.0) GO TO 100	00009330
C	LINEAR LOAD SHAPES	00009340
	DXC=XOC(I)-F(J)	00009350
	IF(DXC.GT.0.0) GO TO 70	00009360
	PM(I,J)=0.0	00009370
	GO TO 90	00009380
70	PM(I,J)=DXC/(1.-F(J))	00009390
90	CONTINUE	00009400
	GO TO 200	00009410
C	TRIG. LOAD SHAPE	00009420
100	DO 190 I=2,NVXM	00009430
	IF(F(J)-2.) 110,120,130	00009440
110	PM(I,J)=(1.+CPH(I))/SIN(PHI(I))	00009450
	GO TO 190	00009460

```

120 PM(I,J)=(1.-CPH(I))/SIN(PHI(I))                                00009470
    GO TO 190                                                       00009480
130 IF(AMOD(F(J),2.)) 150,150,140                                00009490
140 PM(I,J)=SIN((F(J)-1.)/2.*PHI(I))                            00009500
    GO TO 190                                                       00009510
150 PM(I,J)=COS((F(J)-2.)/2.*PHI(I))                            00009520
190 CONTINUE                                                       00009530
200 CONTINUE                                                       00009540
    RETURN                                                           00009550
    END                                                             00009560
*DECK FXYZ                                                         00009570
    SUBROUTINE FXYZ                                               00009580
C                                                                    00009590
C*      CALC. COORDINATE OF BODY VORTICES (XB,YB,ZB)           00009600
C                                                                    00009610
    COMMON/DAT/ DA(5000)                                          00009620
C                                                                    00009630
    DIMENSION XS(29) ,TYS(36) ,RZS(475) ,FVX(150) ,FVT(50)     00009640
    1 ,FJX(149) ,YYM(17)                                         00009650
    1 ,XMC(1),ZZM(1),FZC(1)                                       00009660
C                                                                    00009670
    EQUIVALENCE (DA(5),FCD) , (DA(17),FLOI) , (DA(18),FTHI)     00009680
    1,(DA(19),FCTI) , (DA(21),XS) , (DA(51),TYS) , (DA(90),RZS) 00009690
    2,(DA(566),XMC) , (DA(600),YYM) , (DA(635),ZZM) , (DA(670),FZC) 00009700
    3,(DA(1276),FNDV) ,(DA(1460),FVX) , (DA(1610),FVT) , (DA(15),FMFI) 00009710
    4,(DA(16),FCPI) , (DA(1720),FJX)                               00009720
    5,(DA(87),FOX) , (DA(88),FOY) , (DA(89),FOZ)                 00009730
    6,(DA(20),FNXS) , (DA(50),FNTY) , (DA(565),FNXM) , (DA(1274),FNVX) 00009740
C                                                                    00009750
    COMMON/CRG/ PI,PI4,RC,BETA                                     00009760
    COMMON/CPF/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX,IFBE,SINA,COSA,FAREA 00009770
    * ,SA                                                           00009780
C                                                                    00009790
    LARGE BOU(3000) ,FUS(25000),DM1(10400),AXY(650) ,AYZ(650) 00009800
    *,XB(4000) ,YB(4000) ,ZB(4000) ,WIN(23430),PYL(6061) ,CEL(3700) 00009810
    1,STH(400) ,CTH(400) ,TH(400)                                00009820
    2,STC(150) ,CTC(150) ,VX1(150) ,VX2(150) ,ZC1(150) ,ZC2(150) 00009830
    3,RS(30) ,STS(30) ,ZM(150) ,YM(150) ,ZC(150) ,YMIN(150)    00009840
    4,YMAX(150) ,FVXL(150) ,XMCL(34) ,YMCL(34) ,ZZML(34) ,FZCL(34) 00009850
C                                                                    00009860
C                                                                    00009870
    INITIAL                                                         00009880
    NX5=FNXS                                                       00009880
    NTY=FNTY                                                       00009890

```


	NXM=FNXM	00009900
	FNVXM=FNVX-1.	00009910
	FNVB=NVB	00009920
C*	CALC LONGITUDINAL PARAMETER X	00009930
	FVX(1)=0.0	00009940
	FVX(NVX)=FCD	00009950
	IF(FLOI) 30,10,50	00009960
C	EVEN DELTA PHI	00009970
10	DPHI=PI/(FNVXM-1.)	00009980
	PHX=DPHI/2.	00009990
	FCD2=FCD/2.	00010000
	DO 20 I=2,NVXM	00010010
	FVX(I)=FCD2-FCD2*COS(PHX)	00010020
20	PHX=PHX+DPHI	00010030
	GO TO 50	00010040
C	EVEN DELTA X	00010050
30	DX=FCD/(FNVXM-1.)	00010060
	FVX(2)=DX/2.	00010070
	DO 40 I=3,NVXM	00010080
40	FVX(I)=FVX(I-1)+DX	00010090
C*	CALC LATERAL PARAMETERS	00010100
50	IF(FTHI) 120,120,60	00010110
C	INPUT THETA LIST	00010120
60	IF(FNDV.GT.1.) GO TO 80	00010130
	DO 70 I=1,NVY	00010140
70	TH(I)=FVT(I)	00010150
	TH(NVY+1)=360.	00010160
	GO TO 140	00010170
80	IC=0	00010180
	FVT(NVY+1)=360.	00010190
	DO 100 I=1,NVY	00010200
	DTH=(FVT(I+1)-FVT(I))/FNDV	00010210
	IC=IC+1	00010220
	TH(IC)=FVT(I)	00010230
	DO 100 J=2,NDV	00010240
	IC=IC+1	00010250
100	TH(IC)=TH(IC-1)+DTH	00010260
	TH(IC+1)=360.	00010270
	GO TO 140	00010280
C	EVEN DELTA THETA	00010290
120	DTH=360./FNVB	00010300
	TH(1)=0.	00010310
	DO 130 I=2,NVBP	00010320

	130 TH(I)=TH(I-1)+DTH	00010330
C	SIN-COS THETA	00010340
	140 DO 150 I=1,NVBP	00010350
	THR=TH(I)/RC	00010360
	STH(I)=SIN(THR)	00010370
	150 CTH(I)=COS(THR)	00010380
C*	SHIFT ARRAYS TO LCM FOR CODIM	00010390
	DO 151 I=1,NVX	00010400
	151 FVXL(I)=FVX(I)	00010410
	DO 152 I=1,NXM	00010420
	XMCL(I)=XMC(I)	00010430
	YMCL(I)=YYM(I)	00010440
	ZZML(I)=ZZM(I)	00010450
	152 FZCL(I)=FZC(I)	00010460
C*	CALC CAMBER AND MUTIPLICATION FACTORS	00010470
	IF(NXM.NE.0) GO TO 154	00010480
	DO 153 I=1,NVX	00010490
	YM(I)=1.	00010500
	ZM(I)=1.	00010510
	153 ZC(I)=0.0	00010520
	GO TO 192	00010530
	154 CALL CODIM (XMCL,YMCL,NXM,FVXL,YM,NVX)	00010540
	IF(ABS(FMFI).EQ.1.0) GO TO 180	00010550
	160 DO 170 I=1,NVX	00010560
	170 ZM(I)=YM(I)	00010570
	GO TO 190	00010580
	180 CALL CODIM (XMCL,ZZML,NXM,FVXL,ZM,NVX)	00010590
	190 CALL CODIM (XMCL,FZCL,NXM,FVXL,ZC,NVX)	00010600
	192 IF(FCPI) 220,220,200	00010610
C	PERPENDICULAR TO X-AXIS	00010620
	200 DO 210 I=2,NVX	00010630
	STC(I)=0.0	00010640
	210 CTC(I)=1.	00010650
	GO TO 250	00010660
C	PERPENDICULAR TO CAMBER LINE	00010670
	220 DX=FCD/1000.	00010680
	DXH=DX/2.	00010690
	DO 230 I=2,NVX	00010700
	VX1(I-1)=FVX(I)+DXH	00010710
	230 VX2(I-1)=FVX(I)-DXH	00010720
	CALL CODIM (XMCL,FZCL,NXM,VX1,ZC1,NVXM)	00010730
	CALL CODIM (XMCL,FZCL,NXM,VX2,ZC2,NVXM)	00010740
	DX2=DX*DX	00010750

DO 240 I=2,NVX	00010760
DZ=ZC2(I-1)-ZC1(I-1)	00010770
DH=SQRT(DX2+DZ**2)	00010780
STC(I)=DZ/DH	00010790
240 CTC(I)=DX/DH	00010800
C* TABLE LOOKUP FOR RADIUS AT EACH POINT	00010810
C INPUT (R VS THETA) VS X	00010820
250 CALL CODMT (NXS,NTY,NVB,TYS,RZS,TH,XB)	00010830
L=1	00010840
DO 350 I=2,NVX	00010850
DO 310 J=2,NXS	00010860
IF(FVX(I)-XS(J)) 320,320,310	00010870
310 CONTINUE	00010880
RAT=0.0	00010890
GO TO 330	00010900
320 RAT=(XS(J)-FVX(I))/(XS(J)-XS(J-1))	00010910
330 K=J	00010920
DO 340 IT=1,NVB	00010930
L=L+1	00010940
ZB(L)=XB(K)-RAT*(XB(K)-XB(K-1))	00010950
340 K=K+NXS	00010960
L=L+1	00010970
L1=L-NVB	00010980
ZB(L)=ZB(L1)	00010990
350 CONTINUE	00011000
C* CALC XB,YB,ZB AND INTEGRATED LOAD PARAMETERS	00011010
XB(1)=0.0	00011020
YB(1)=0.0	00011030
ZB(1)=0.0	00011040
YMIN(1)=0.0	00011050
YMAX(1)=0.0	00011060
FAREA=0.0	00011070
L=1	00011080
DO 390 IX=2,NVX	00011090
C1=ZM(IX)*CTC(IX)	00011100
IF(FMFI.LT.0.0) C1=YM(IX)*CTC(IX)	00011110
YMIN(IX)=1000000.	00011120
YMAX(IX)=0.0	00011130
DO 360 IY=1,NVBP	00011140
L=L+1	00011150
C2=ZB(L)*CTH(IY)	00011160
IF(FMFI.GE.0.0) GO TO 352	00011170
YB(L)=ZC(IX)+C1*ZB(L)*STH(IY)	00011180

	ZB(L)=C2*ZM(IX)	00011190
	GO TO 354	00011200
352	YB(L)=YM(IX)*ZB(L)*STH(IY)	00011210
	ZB(L)=ZC(IX)+C1*C2	00011220
354	XB(L)=FVX(IX)-C2*STC(IX)	00011230
	IF(YB(L).GT.YMAX(IX)) YMAX(IX)=YB(L)	00011240
	IF(YB(L).LT.YMIN(IX)) YMIN(IX)=YB(L)	00011250
360	CONTINUE	00011260
	DY1=YMAX(IX)-YMIN(IX)	00011270
	DY2=YMAX(IX-1)-YMIN(IX-1)	00011280
	FAREA=(DY1+DY2) *(FVX(IX)-FVX(IX-1))+FAREA	00011290
390	CONTINUE	00011300
	IFBE=0	00011310
	IF(DY1.GT.0.001) IFBE=1	00011320
C*	SETUP FOR COEFFICIENTS	00011330
C*	AXY AND AYZ = PROJECTED AREA	00011340
	IJ=1	00011350
	JX=FJX(1)	00011360
	NC=0	00011370
	SA=0.0	00011380
	M2=2	00011390
C		00011400
	DO 460 IX=1,NVXM	00011410
	IF(IX.LE.JX) GO TO 410	00011420
	IF(IJ.GE.NJX) GO TO 410	00011430
	IJ=IJ+1	00011440
	JX=FJX(IJ)	00011450
C		00011460
410	DO 450 IY=1,NVY	00011470
	IF(IX.NE.JX) GO TO 420	00011480
	NC=NC+1	00011490
	AXY(NC)=0.0	00011500
	AYZ(NC)=0.0	00011510
C		00011520
420	DO 450 I=1,NDV	00011530
	IF(IX.GT.1) GO TO 430	00011540
	AX=YB(M2)*XB(M2+1)-YB(M2+1)*XB(M2)	00011550
	AY=YB(M2)*ZB(M2+1)-YB(M2+1)*ZB(M2)	00011560
	AZ=XB(M2)*ZB(M2+1)-XB(M2+1)*ZB(M2)	00011570
	GO TO 440	00011580
C		00011590
430	M1=M2-NVBP	00011600
	X4=XB(M2+1)-XB(M1)	00011610

	X3=XB(M1+1)-XB(M2)	00011620
	Y4=YB(M2+1)-YB(M1)	00011630
	Y3=YB(M1+1)-YB(M2)	00011640
	Z4=ZB(M2+1)-ZB(M1)	00011650
	Z3=ZB(M1+1)-ZB(M2)	00011660
	AX=Y4*X3-Y3*X4	00011670
	AY=Y4*Z3-Y3*Z4	00011680
	AZ=X4*Z3-X3*Z4	00011690
C		00011700
	440 IF(IX.NE.JX) GO TO 445	00011710
	AXY(NC)=AXY(NC)+AX	00011720
	AYZ(NC)=AYZ(NC)+AY	00011730
	445 SA=SQRT(AX**2+AY**2+AZ**2)+SA	00011740
C		00011750
	M2=M2+1	00011760
	450 CONTINUE	00011770
	460 M2=M2+1	00011780
C*	COMPRESS. EFFECT AND TRANSLATE COORDINATES	00011790
	500 DO 510 I=1,L	00011800
	XB(I)=XB(I)+FOX	00011810
	YB(I)=BETA*(YB(I)+FOY)	00011820
	510 ZB(I)=BETA*(ZB(I)+FOZ)	00011830
	RETURN	00011840
	END	00011850
*DECK	FQPTS	00011860
	SUBROUTINE FQPTS	00011870
C	FANPOD	00011880
C*	CALC DIRECTION MATRICES AND COORDINATE OF CONTROL	00011890
C*	POINTS Q	00011900
C		00011910
	COMMON/DAT/ DA(5000)	00011920
	DIMENSION FJX(1)	00011930
	EQUIVALENCE (DA(1720),FJX)	00011940
C		00011950
	COMMON/CPF/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX	00011960
C		00011970
	LARGE BOU(3000) ,FUS(25000)	00011980
*	,XQ(650) ,YQ(650) ,ZQ(650)	00011990
1	,TMX(650) ,TMY(650) ,TMZ(650)	0002000
2	,TTX(650) ,TTY(650) ,TTZ(650)	0002010
3	,XN(650) ,YN(650) ,ZN(650)	0002020
4	,DMI(3900) ,XB(4000) ,YB(4000) ,ZB(4000)	0002030
C		0002040

C*	START	00012050
	NDVH=NDV/2	00012060
	IC=0	00012070
	K1=NVBP+NDVH-1	00012080
	DO 60 IY=1,NVY	00012090
	K2=IY*NDV-K1	00012100
	DO 50 IX=1,NJX	00012110
	JX=FJX(IX)	00012120
	J2=JX*NVBP+K2	00012130
	J4=J2+1	00012140
	IF(JX-1) 10,10,20	00012150
10	J3=1	00012160
	J1=1	00012170
	GO TO 30	00012180
20	J1=J2-NVBP	00012190
	J3=J1+1	00012200
30	IC=IC+1	00012210
	XQ(IC)=(XB(J1)+XB(J2)+XB(J3)+XB(J4))/4.	00012220
	YQ(IC)=(YB(J1)+YB(J2)+YB(J3)+YB(J4))/4.	00012230
	ZQ(IC)=(ZB(J1)+ZB(J2)+ZB(J3)+ZB(J4))/4.	00012240
C		00012250
	TM1=XB(J2)-XB(J1)+XB(J4)-XB(J3)	00012260
	TM2=YB(J2)-YB(J1)+YB(J4)-YB(J3)	00012270
	TM3=ZB(J2)-ZB(J1)+ZB(J4)-ZB(J3)	00012280
	TT1=XB(J3)-XB(J1)+XB(J4)-XB(J2)	00012290
	TT3=ZB(J3)-ZB(J1)+ZB(J4)-ZB(J2)	00012300
	TT2=YB(J3)-YB(J1)+YB(J4)-YB(J2)	00012310
C		00012320
	S2143 =SQRT(TM1**2+TM2**2+TM3**2)	00012330
	S3142=SQRT(TT1**2+TT2**2+TT3**2)	00012340
C		00012350
	TMX(IC)=TM1/S2143	00012360
	TMY(IC)=TM2/S2143	00012370
	TMZ(IC)=TM3/S2143	00012380
	TTX(IC)=TT1/S3142	00012390
	TTY(IC)=TT2/S3142	00012400
	TTZ(IC)=TT3/S3142	00012410
C		00012420
	TX=TMY(IC)*TTZ(IC)-TMZ(IC)*TTY(IC)	00012430
	TY=TMZ(IC)*TTX(IC)-TMX(IC)*TTZ(IC)	00012440
	TZ=TMX(IC)*TTY(IC)-TMY(IC)*TTX(IC)	00012450
	SRXYZ=SQRT(TX**2+TY**2+TZ**2)	00012460
	XN(IC)=TX/SRXYZ	00012470

	YN(IC)=TY/SRXYZ	00012480
50	ZN(IC)=TZ/SRXYZ	00012490
60	CONTINUE	00012500
	RETURN	00012510
	END	00012520
*DECK	FCLS	00012530
	SUBROUTINE FCLS	00012540
C		00012550
C	CALC CHORDWISE LOAD SHAPES FOR THE FANPOD	00012560
C		00012570
	COMMON/DAT/ DA(5000)	00012580
	DIMENSION FVX(1),F(1)	00012590
	EQUIVALENCE (DA(1295),F) ,(DA(1460),FVX),(DA(5),FCD)	00012600
C		00012610
C	COMMON/CPF/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF	00012620
		00012630
	LARGE BOU(3000) ,FUS(25000),DM1(7800) ,PM(100,25)	00012640
1	,DM2(46591),PHI(150) ,CPH(150) ,XOC(150)	00012650
C*	START	00012660
C	CALC X/C , COS(PHI) AND PHI	00012670
	DO 10 I=2,NVXM	00012680
	XOC(I)=FVX(I)/FCD	00012690
	CPH(I)=1.-2.*XOC(I)	00012700
10	PHI(I)= ACOS(CPH(I))	00012710
	IF(XOC(NVXM).LT.1.0) GO TO 30	00012720
	WRITE (6,20)	00012730
20	FORMAT (32H1 ** FCLS * X/C GREATER THAN 1.0)	00012740
	CALL EXIT	00012750
C*	NO. OF LOAD SHAPE LOOP	00012760
30	DO 200 J=1,NF	00012770
	IF(F(1).GE.1.0) GO TO 100	00012780
C	LINEAR LOAD SHAPES	00012790
	DO 90 I=2,NVXM	00012800
	DXC=XOC(I)-F(J)	00012810
	IF(DXC.GT.0.0) GO TO 70	00012820
	PM(I,J)=0.0	00012830
	GO TO 90	00012840
70	PM(I,J)=DXC/(1.-F(J))	00012850
90	CONTINUE	00012860
	GO TO 200	00012870
C	TRIG. LOAD SHAPE	00012880
100	DO 190 I=2,NVXM	00012890
	IF(F(J)-2.) 110,120,130	00012900

110	PM(I,J)=(1.+CPH(I))/SIN(PHI(I))	00012910
	GO TO 190	00012920
120	PM(I,J)=(1.-CPH(I))/SIN(PHI(I))	00012930
	GO TO 190	00012940
130	IF(AMOD(F(J),2.)) 150,150,140	00012950
140	PM(I,J)=SIN((F(J)-1.)/2.*PHI(I))	00012960
	GO TO 190	00012970
150	PM(I,J)=COS((F(J)-2.)/2.*PHI(I))	00012980
190	CONTINUE	00012990
200	CONTINUE	00013000
	RETURN	00013010
	END	00013020
*DECK	WXYZ	00013030
	SUBROUTINE WXYZ	00013040
C		00013050
C*	CALC. COORDINATE OF WING VORTICES AND CONTROL POINTS	00013060
C		00013070
	COMMON/DAT/ DA(5000)	00013080
	DIMENSION WPI(1),WPO(1)	00013090
C		00013100
	EQUIVALENCE (DA(2),WSPAN) ,(DA(1203),WITO),(DA(1204),WITI)	00013110
	1,(DA(1210),WPO) ,(DA(1240),WPI) ,(DA(1200),WOX) ,(DA(1202),WOZ)	00013120
	2,(DA(1208),XNEO),(DA(1209),XNEI) ,(DA(1270),WNVC),(DA(1281),WNW)	00013130
C		00013140
	COMMON/CRG/ PI,PI4,RC,BETA	00013150
	COMMON/CFG/ IDB,IDF	00013160
	COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3	00013170
	1 ,NE5,NTB,NU,NW,NVC,NVS,NJC,NJS,NVSO,NVSI,IWX	00013180
C		00013190
	LARGE BOU(3000) ,FUS(25000),FAN(23700)	00013200
	1 ,EV(52) ,XLV(52) ,XTLV(52)	00013210
	2 ,DM1(810) ,WPE(30) ,WPT(60) ,DM2(11540)	00013220
	3 ,CFI(50) ,CFJ(50) ,STH(50) ,ACT(50)	00013230
	4 ,ET(52) ,XLT(52) ,XTLT(52)	00013240
C		00013250
C*	START	00013260
	DTH=PI/WNVC	00013270
	DTH2=DTH/2.	00013280
	TH=DTH2	00013290
	DO 2 I=1,NVC	00013300
	CFI(I)=.5*(1.-COS(TH))	00013310
	STH(I)=SIN(TH)	00013320
	ACT(I)=TH-DTH2	00013330

	CFJ(I)=.5*(1.-COS(ACT(I)))	00013340
2	TH=TH+DTH	00013350
	HWS=WSPAN/2.	00013360
	IWO=3.*XNEO-2.	00013370
C	SETUP FOR WING ONLY	00013380
	IF(IDB.NE.0) GO TO 18	00013390
	IF(IDF.NE.0) GO TO 18	00013400
	DO 10 I=1,IWO,3	00013410
	WPT(I)=WPO(I)	00013420
	WPT(I+1)=WPO(I+1)	00013430
	WPT(I+2)=WPO(I+2)	00013440
10	CONTINUE	00013450
	NPT=I	00013460
	NVSH=NVS/2	00013470
	WNVSH=NVSH	00013480
	DL=I./WNVSH	00013490
	NE2=NVSH+1	00013500
	EV(2)=1.-DL	00013510
	EV(1)=EV(2)+.625*DL	00013520
	ET(1)=1.-DL/2.	00013530
	ET(2)=ET(1)-DL	00013540
	DO 14 I=3,NE2	00013550
	ET(I)=ET(I-1)-DL	00013560
14	EV(I)=EV(I-1)-DL	00013570
	NE5=100	00013580
	NE3=100	00013590
	DLO=DL	00013600
	DL2=DL	00013610
	GO TO 82	00013620
18	NVSH=NVS0/2	00013630
	NE3=NVSH-3	00013640
C	SETUP WING PLANFORM TABLE	00013650
	IC=1	00013660
	IF(IDF.EQ.0) GO TO 44	00013670
	IWX=3.*XNEI-2.	00013680
	IC=IWX	00013690
	DO 40 I=1,IWX,3	00013700
	WPE(IC)=WPI(I)	00013710
	WPE(IC+1)=WPI(I+1)+WOX	00013720
	WPE(IC+2)=WPI(I+2)+WOX	00013730
	WPT(I)=WPE(IC)	00013740
	WPT(I+1)=WPE(IC+1)	00013750
	WPT(I+2)=WPE(IC+2)	00013760

40	IC=IC-3	00013770
	WPE(6)=WPE(6)-WOX	00013780
	WPE(9)=WPE(9)-WOX	00013790
	WPT(IWX-4)=WPT(IWX-4)-WOX	00013800
	WPT(IWX-1)=WPT(IWX-1)-WOX	00013810
	CALL FRXYZ (WITI,NTI,1,IWX)	00013820
	IF(IDB.EQ.0) GO TO 42	00013830
	NE2=NVS/2	00013840
	NE5=NE2-3	00013850
	WPT(6)=WPT(6)-WOX	00013860
	WPT(9)=WPT(9)-WOX	00013870
	CALL BRXYZ	00013880
42	IC=IWX+3	00013890
44	DO 50 I=1,IWO,3	00013900
	WPE(I)=WPO(I)	00013910
	WPE(I+1)=WPO(I+1)+WOX	00013920
	WPE(I+2)=WPO(I+2)+WOX	00013930
	IF(I.EQ.4) WPE(I+2)=WPO(I+2)	00013940
	IF(I.EQ.7) WPE(I+2)=WPO(I+2)	00013950
	WPT(IC)=WPE(I)	00013960
	WPT(IC+1)=WPE(I+1)	00013970
	WPT(IC+2)=WPE(I+2)	00013980
	IC=IC+3	00013990
50	CONTINUE	00014000
60	NPT=IC-3	00014010
C*	CALC VORTEX LATTICE SPACING SPANWISE	00014020
	WNVSH=NVSH	00014030
	DLO=(1.-WPO(10))/(WNVSH-3.5)	00014040
	EV(NE3+1)=(WPO(10)+WPO(7))/2.	00014050
	ET(NE3+1)=WPO(7)	00014060
	EV(NE3+2)=(WPO(7)+WPO(4))/2.	00014070
	ET(NE3+2)=WPO(4)	00014080
	EV(NE3+3)=(WPO(4)+WPO(1))/2.	00014090
	ET(NE3+3)=WPO(1)	00014100
	EV(2)=1.-DLO	00014110
	EV(1)=EV(2)+.625*DLO	00014120
	ET(1)=1.-DLO/2.	00014130
	ET(2)=ET(1)-DLO	00014140
	DO 70 I=3,NE3	00014150
	ET(I)=ET(I-1)-DLO	00014160
70	EV(I)=EV(I-1)-DLO	00014170
	IF(IDF.NE.0) GO TO 72	00014180
	NE5=NE3	00014190

NE2=NE5+3	00014200
NE3=100	00014210
CALL BRXYZ	00014220
ET(NE2+1)=1.	00014230
GO TO 84	00014240
72 CALL FRXYZ (WITO,NT0,0,IWX)	00014250
INNER WING	00014260
EV(NE3+4)=(WPI(IWX-3)+WPI(IWX))/2.	00014270
ET(NE3+4)=WPI(IWX)	00014280
EV(NE3+5)=(WPI(IWX-6)+WPI(IWX-3))/2.	00014290
ET(NE3+5)=WPI(IWX-3)	00014300
EV(NE3+6)=(WPI(IWX-9)+WPI(IWX-6))/2.	00014310
ET(NE3+6)=WPI(IWX-6)	00014320
NE4=NE3+7	00014330
NE5=100	00014340
IF(IDB.EQ.0) GO TO 76	00014350
NVSH=NVSI/2	00014360
WNVSH=NVSH	00014370
DLI=(WPI(IWX)-WPI(1))/WNVSH	00014380
NE2=NVS/2	00014390
NE5=NE2-3	00014400
EV(NE5+1)=(WPI(10)+WPI(7))/2.	00014410
ET(NE5+1)=WPI(10)	00014420
EV(NE5+2)=(WPI(7)+WPI(4))/2.	00014430
ET(NE5+2)=WPI(7)	00014440
EV(NE2)=(WPI(4)+WPI(1))/2.	00014450
ET(NE2)=WPI(4)	00014460
ET(NE2+1)=WPI(1)	00014470
IF(NE4.GT.NE5) GO TO 84	00014480
DO 74 I=NE4,NE5	00014490
ET(I)=ET(I-1)-DLI	00014500
74 EV(I)=EV(I-1)-DLI	00014510
GO TO 84	00014520
76 NVSH=NVSI/2+1	00014530
WNVSH=NVSH	00014540
DLI=WPI(IWX-9)/(WNVSH-3.5)	00014550
NE2=NVS/2+1	00014560
DL2=DLI/2.	00014570
DO 80 I=NE4,NE2	00014580
ET(I)=EV(I-1)-DL2	00014590
80 EV(I)=EV(I-1)-DLI	00014600
82 EV(NE2)=0.	00014610
ET(NE2+1)=-DL2	00014620

```

84 NE1=NE2+1                                00014630
   EV(NE1)=1.                                00014640
C       FIND X LEADING EDGE AND CHORD AT EACH ETA STATIONS 00014650
   CALL WINGD (NE1,ET,XLT,XLT,WP)            00014660
   CALL WINGD (NE1,EV,XLV,XTLV,WP)          00014670
C       TEST WING PLANFORM TABLE FOR NACELLE TYPE BREAK 00014680
   IF(WNW,NE,0.0) GO TO 100                 00014690
   DO 90 I=1,NPT,3                          00014700
   IF(WPT(I+3).GE.1.0) GO TO 100            00014710
   IF(ABS(WPT(I)-WPT(I+3)).GT.0.01) GO TO 90 00014720
   DO 85 J=1,NE2                             00014730
   IF(ABS(WPT(I)-ET(J)).GT.0.005) GO TO 85 00014740
   ET(J)=-ET(J)                              00014750
   XLT(J)=WPT(I+1)                          00014760
   XTLT(J)=WPT(I+2)-WPT(I+1)               00014770
85 CONTINUE                                  00014780
90 CONTINUE                                  00014790
100 HWS=HWS*BETA                             00014800
   RETURN                                    00014810
   END                                       00014820
*DECK BRXYZ                                  00014830
   SUBROUTINE BRXYZ                          00014840
C                                             00014850
C*      CALC. COORDINATES OF ROOT SECTION 00014860
C                                             00014870
   COMMON/DAT/ DA(5000)                     00014880
   DIMENSION FVX(150) ,WJC(30) ,WJS(30)    00014890
C                                             00014900
   EQUIVALENCE (DA(1202),ZCON) ,(DA(1207),WITH) ,(DA(1660),WJC) 00014910
1 ,(DA(1690),WJS) ,(DA(4015),FOX) ,(DA(4735),FVX) ,(DA(1270),WNVC) 00014920
C                                             00014930
   COMMON/CRG/ PI,PI4,RC,BETA               00014940
   COMMON/CPB/ NVX,NVXM,NVY,NDV,NVB,NVBP    00014950
   COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3 00014960
1 ,NE5,NVT,NU,NW,NVC,NVS,NJC,NJS          00014970
C                                             00014980
   LARGE BOU(3000) ,FUS(13000)             00014990
1 ,XB(4000) ,YB(4000) ,ZB(4000) ,FAN(23700) 00015000
2 ,EV(52) ,XLV(52) ,XTLV(52)              00015010
3 ,XD(270) ,YD(270) ,ZD(270)              00015020
4 ,DM1(30) ,WPE(60) ,DM2(6140)            00015030
5 ,XR(1800) ,YR(1800) ,ZR(1800)           00015040
6 ,CFI(50) ,DM3(10784) ,PYL(6061) ,CEL(3700) 00015050

```

	7	,XRF(50)	,YRF(50)	,ZRF(50)	,TAN1(150)	,TAN2(150)	00015060
C							00015070
C*		SEARCH LIST OF FUSELAGE X FOR XLE AND XTE OF WING ROOT					00015080
		DO 50 I=1,NVX					00015090
		FVXT=FVX(I)+FOX					00015100
		IF(WPE(2).LT.FVXT) GO TO 60					00015110
	50	CONTINUE					00015120
	60	IS=I					00015130
		DO 70 I=IS,NVX					00015140
		FVXT=FVX(I)+FOX					00015150
		IF(WPE(3).LT.FVXT) GO TO 90					00015160
	70	CONTINUE					00015170
C							00015180
C*		SHIFT VORTEX COORD. OF FUSELAGE TO THE MERGING ROOT					00015190
	90	IE=I					00015200
		ITV=WITH-1.					00015210
		IC=ITV*NDV+2+NVBP*(IS-2)					00015220
		IC1=IC-NVBP					00015230
		IF(IC1.LE.0) IC1=IC					00015240
		YCON=HWS*WPE(10)					00015250
		NPT=IE-IS+2					00015260
		DO 92 I=1,NPT					00015270
		XD(I)=XB(IC1)					00015280
		YD(I)=YB(IC1)/BETA					00015290
		ZD(I)=ZB(IC1)/BETA					00015300
	92	IC1=IC1+NVBP					00015310
		CD=WPE(3)-WPE(2)					00015320
		DO 94 I=1,NVC					00015330
	94	XRF(I)=WPE(2)+CD*CFI(I)					00015340
		CALL CODIM (XD,YD,NPT,XRF,YRF,NVC)					00015350
		CALL CODIM (XD,ZD,NPT,XRF,ZRF,NVC)					00015360
		IS=0					00015370
		DO 96 I=1,NVC					00015380
		IS=IS+4					00015390
		XR(IS)=XRF(I)					00015400
		YR(IS)=YRF(I)					00015410
	96	ZR(IS)=ZRF(I)					00015420
		XRS=XR(IS)					00015430
		NVT=NVX-IE+1					00015440
		IC=IC1-NVBP					00015450
		DO 100 I=1,NVT					00015460
		IS=IS+4					00015470
		XR(IS)=XB(IC)					00015480

	YR(IS)=YB(IC)/BETA	00015490
	ZR(IS)=ZB(IC)/BETA	00015500
100	IC=IC+NVBP	00015510
C		00015520
C*	CALC VORTEX COORD. OF MERGING ROOT TO PLANAR WING	00015530
	XRE=XR(IS)	00015540
	WNVT=NVT	00015550
	CD=WPE(12)-WPE(11)	00015560
	IS=-3	00015570
	DO 110 I=1,NVC	00015580
	IS=IS+4	00015590
	XR(IS)=WPE(11)+CD*CFI(I)	00015600
	YR(IS)=YCON	00015610
110	ZR(IS)=ZCON	00015620
	XRR=XR(IS)	00015630
	DXF=XRE-XRS	00015640
	DXR=XRE-XRR	00015650
	DO 120 I=1,NVT	00015660
	IS=IS+4	00015670
	XR(IS)=(XR(IS+3)-XRS)/DXF*DXR+XRR	00015680
	YR(IS)=YCON	00015690
120	ZR(IS)=ZCON	00015700
C		00015710
C*	CALC COORD. OF THE TWO CENTER ROOT VORTEX LINES	00015720
	TTH=.6666667	00015730
	IX=4*NVC	00015740
	X1=WPE(3)	00015750
	Y1=(YR(IX)+YR(IX+4))/2.	00015760
	X2=WPE(12)	00015770
	Y2=YCON	00015780
	DX=X2-X1	00015790
	DY=Y2-Y1	00015800
	DZ=ZCON-(ZR(IX)+ZR(IX+4))/2.	00015810
	SQD=SQRT(DX**2+DY**2+DZ**2)	00015820
	XP1=X1+DX/3.-WPE(6)*DY/SQD	00015830
	YP1=Y1+DY/3.+WPE(6)*DX/SQD	00015840
	XP2=X1+TTH*DX-WPE(9)*DY/SQD	00015850
	YP2=Y1+TTH*DY+WPE(9)*DX/SQD	00015860
	TT1=(XP1-X1)/(YP1-Y1)	00015870
	TT2=(X2-XP2)/(Y2-YP2)	00015880
	DY=(WPE(4)-WPE(1))*HWS	00015890
	TL1=(WPE(5)-WPE(2))/DY	00015900
	TL2=(WPE(11)-WPE(8))/DY	00015910

C	DT1=(TT1-TL1)/WNVC	00015920
	DT2=(TT2-TL2)/WNVC	00015930
	TAN1(1)=TL1+DT1/2.	00015940
	TAN2(1)=TL2+DT2/2.	00015950
	DO 130 I=2,NVC	00015960
	TAN1(I)=TAN1(I-1)+DT1	00015970
130	TAN2(I)=TAN2(I-1)+DT2	00015980
C		00015990
	IS=1	00016000
	DO 140 I=1,NVC	00016010
	DX=XR(IS)-XR(IS+3)	00016020
	DY=YR(IS)-YR(IS+3)	00016030
	DZ=ZR(IS)-ZR(IS+3)	00016040
	DXDY=DX/DY	00016050
	TC1=(TAN1(I)-DXDY)/(1.+DXDY*TAN1(I))/3.	00016060
	TC2=(TAN2(I)-DXDY)/(1.+DXDY*TAN2(I))/3.	00016070
	XR(IS+2)=XR(IS+3)+DX/3.+DY*TC1	00016080
	YR(IS+2)=YR(IS+3)+DY/3.-DX*TC1	00016090
	ZR(IS+2)=ZR(IS+3)+DZ/3.	00016100
	XR(IS+1)=XR(IS+3)+TTH*DX-DY*TC2	00016110
	YR(IS+1)=YR(IS+3)+TTH*DY+DX*TC2	00016120
	ZR(IS+1)=ZR(IS+3)+TTH*DZ	00016130
140	IS=IS+4	00016140
C*	CALC XR,YR,ZR FOR TRAILING VORTICES	00016150
	DT1=TAN1(NVC)/WNVT	00016160
	DT2=TAN2(NVC)/WNVT	00016170
	TAN1(1)=TAN1(NVC)-DT1	00016180
	TAN2(1)=TAN2(NVC)-DT2	00016190
	DO 150 I=2,NVT	00016200
	TAN1(I)=TAN1(I-1)-DT1	00016210
150	TAN2(I)=TAN2(I-1)-DT2	00016220
C		00016230
		00016240
	DO 160 I=1,NVT	00016250
	DX=XR(IS)-XR(IS+3)	00016260
	DY=YR(IS)-YR(IS+3)	00016270
	DZ=ZR(IS)-ZR(IS+3)	00016280
	DXDY=DX/DY	00016290
	TC1=(TAN1(I)-DXDY)/(1.+DXDY*TAN1(I))/3.	00016300
	TC2=(TAN2(I)-DXDY)/(1.+DXDY*TAN2(I))/3.	00016310
	XR(IS+2)=XR(IS+3)+DX/3.+DY*TC1	00016320
	YR(IS+2)=YR(IS+3)+DY/3.-DX*TC1	00016330
	ZR(IS+2)=ZR(IS+3)+DZ/3.	00016340

	XR(IS+1)=XR(IS+3)+TTH*DX-DY*TC2	00016350
	YR(IS+1)=YR(IS+3)+TTH*DY+DX*TC2	00016360
	ZR(IS+1)=ZR(IS+3)+TTH*DZ	00016370
160	IS=IS+4	00016380
C*	CALC CHORDS FOR WING ROOT SECTION	00016390
	WPE(3)=SQRT((WPE(2)-XR(4))**2+(WPE(1)*HWS-YR(4))**2)+WPE(2)	00016400
	WPE(6)=SQRT((WPE(5)-XR(3))**2+(WPE(4)*HWS-YR(3))**2)+WPE(5)	00016410
	WPE(9)=SQRT((WPE(8)-XR(2))**2+(WPE(7)*HWS-YR(2))**2)+WPE(8)	00016420
	IS=4	00016430
	DO 190 I=2,NVC	00016440
	IS=IS+4	00016450
	WPE(3)=SQRT((XR(IS-4)-XR(IS))**2+(YR(IS-4)-YR(IS))**2)+WPE(3)	00016460
	WPE(6)=SQRT((XR(IS-5)-XR(IS-1))**2+(YR(IS-5)-YR(IS-1))**2)+WPE(6)	00016470
190	WPE(9)=SQRT((XR(IS-6)-XR(IS-2))**2+(YR(IS-6)-YR(IS-2))**2)+WPE(9)	00016480
	WPE(3)=SQRT((XR(IS)-X1)**2+(YR(IS)-Y1)**2)+WPE(3)	00016490
	WPE(6)=SQRT((XR(IS-1)-XP1)**2+(YR(IS-1)-YP1)**2)+WPE(6)	00016500
	WPE(9)=SQRT((XR(IS-2)-XP2)**2+(YR(IS-2)-YP2)**2)+WPE(9)	00016510
C*	COMPRESS. EFFECT ON WING ROOT PARAMETERS	00016520
	IS=4*(NVC+NVT)	00016530
	IF(BETA.EQ.1.) GO TO 196	00016540
	DO 192 I=1,IS	00016550
	YR(I)=YR(I)*BETA	00016560
192	ZR(I)=ZR(I)*BETA	00016570
C	SHIFT ANS.	00016580
196	IC=1200	00016590
	DO 200 I=1,IS	00016600
	IC=IC+1	00016610
	XR(IC)=XR(I)	00016620
	YR(IC)=YR(I)	00016630
200	ZR(IC)=ZR(I)	00016640
C*	CALC CONTROL POINTS AT SPECIFIED WJC AND WJS	00016650
	IS=180	00016660
	DO 230 IJS=1,NJS	00016670
	IY=WJS(IJS)	00016680
	IE=IY-NE5	00016690
	IF(IE.LE.0) GO TO 230	00016700
	IF(IE.GT.3) GO TO 230	00016710
	DO 225 IJC=1,NJC	00016720
	IX=WJC(IJC)	00016730
	IC=4*IX+IE+1200	00016740
	IS=IS+1	00016750
	XD(IS)=(XR(IC)+XR(IC+1)+XR(IC-4)+XR(IC-3))/4.	00016760
	YD(IS)=(YR(IC)+YR(IC+1)+YR(IC-4)+YR(IC-3))/4.	00016770


```

225 ZD(IS)=(ZR(IC)+ZR(IC+1)+ZR(IC-4)+ZR(IC-3))/4.          00016780
230 CONTINUE                                                00016790
    RETURN                                                  00016800
    END                                                      00016810
*DECK FRXYZ                                                00016820
    SUBROUTINE FRXYZ (WITH,NVT,IST,IXX)                    00016830
C                                                            00016840
C*          CALC. COORDINATES OF ROOT SECTION            00016850
C                                                            00016860
    COMMON/DAT/ DA(5000)                                    00016870
    DIMENSION  FVX(150) ,WJC(30) ,WJS(30)                00016880
C                                                            00016890
    EQUIVALENCE (DA(1202),ZCON) ,(DA(1460),FVX) ,(DA(1660),WJC) 00016900
1,(DA(1690),WJS) ,(DA(87),FOX) ,(DA(1270),WNVC)          00016910
C                                                            00016920
    COMMON/CRG/ PI,PI4,RC,BETA                             00016930
    COMMON/CPF/ NVX,NVXM,NVY,NDV,NVB,NVBP                00016940
    COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3             00016950
1 ,NE5,NDD,NU,NW,NVC,NVS,NJC,NJS                        00016960
C                                                            00016970
    LARGE          BOU(3000) ,FUS(25000),DM1(11700)      00016980
1 ,XB(4000) ,YB(4000) ,ZB(4000)                        00016990
2 ,EV(52) ,XLV(52) ,XTLV(52)                            00017000
3 ,XD(270) ,YD(270) ,ZD(270)                            00017010
4 ,WPE(30) ,WPT(60) ,DM2(6140)                          00017020
5 ,XR(1800) ,YR(1800) ,ZR(1800)                         00017030
6 ,CFI(50) ,DM3(10784),PYL(6061) ,CEL(3700)            00017040
7 ,XRF(50) ,YRF(50) ,ZRF(50) ,TAN1(150) ,TAN2(150)    00017050
C                                                            00017060
C*          SEARCH LIST OF FUSELAGE X FOR XLE AND XTE OF WING ROOT 00017070
    DO 50 I=1,NVX                                          00017080
    FVXT=FVX(I)+FOX                                        00017090
    IF(WPE(2).LT.FVXT) GO TO 60                            00017100
50 CONTINUE                                               00017110
60 IS=I                                                  00017120
    DO 70 I=IS,NVX                                        00017130
    FVXT=FVX(I)+FOX                                        00017140
    IF(WPE(3).LT.FVXT) GO TO 90                            00017150
70 CONTINUE                                               00017160
C                                                            00017170
C*          SHIFT VORTEX COORD. OF FUSELAGE TO THE MERGING ROOT 00017180
90 IE=I                                                  00017190
    ITV=WITH-1.                                           00017200

```

	IC=ITV*NDV+2+NVBP*(IS-2)	00017210
	IC1=IC-NVBP	00017220
	IF(IC1.LE.0) IC1=IC	00017230
	YCON=HWS*WPE(10)	00017240
	NPT=IE-IS+2	00017250
	DO 92 I=1,NPT	00017260
	XD(I)=XB(IC1)	00017270
	YD(I)=YB(IC1)/BETA	00017280
	ZD(I)=ZB(IC1)/BETA	00017290
92	IC1=IC1+NVBP	00017300
	CD=WPE(3)-WPE(2)	00017310
	DO 94 I=1,NVC	00017320
94	XRF(I)=WPE(2)+CD*CFI(I)	00017330
	CALL CODIM (XD,YD,NPT,XRF,YRF,NVC)	00017340
	CALL CODIM (XD,ZD,NPT,XRF,ZRF,NVC)	00017350
	IS=0	00017360
	DO 96 I=1,NVC	00017370
	IS=IS+4	00017380
	XR(IS)=XRF(I)	00017390
	YR(IS)=YRF(I)	00017400
96	ZR(IS)=ZRF(I)	00017410
	XRS=XR(IS)	00017420
	NVT=NVX-IE+1	00017430
	IC=IC1-NVBP	00017440
	DO 100 I=1,NVT	00017450
	IS=IS+4	00017460
	XR(IS)=XB(IC)	00017470
	YR(IS)=YB(IC)/BETA	00017480
	ZR(IS)=ZB(IC)/BETA	00017490
100	IC=IC+NVBP	00017500
C		00017510
C*	CALC VORTEX COORD. OF MERGING ROOT TO PLANAR WING	00017520
	XRE=XR(IS)	00017530
	WNVT=NVT	00017540
	CD=WPE(12)-WPE(11)	00017550
	IS=-3	00017560
	DO 110 I=1,NVC	00017570
	IS=IS+4	00017580
	XR(IS)=WPE(11)+CD*CFI(I)	00017590
	YR(IS)=YCON	00017600
110	ZR(IS)=ZCON	00017610
	XRR=XR(IS)	00017620
	DXF=XRE-XRS	00017630

```

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.

```

```

00017640
00017650
00017660
00017670
00017680
00017690
00017700
00017710
00017720
00017730
00017740
00017750
00017760
00017770
00017780
00017790
00017800
00017810
00017820
00017830
00017840
00017850
00017860
00017870
00017880
00017890
00017900
00017910
00017920
00017930
00017940
00017950
00017960
00017970
00017980
00017990
00018000
00018010
00018020
00018030
00018040
00018050
00018060

```

	TC2=(TAN2(I)-DXDY)/(1.+DXDY*TAN2(I))/3.	00018070
	XR(IS+2)=XR(IS+3)+DX/3.+DY*TC1	00018080
	YR(IS+2)=YR(IS+3)+DY/3.-DX*TC1	00018090
	ZR(IS+2)=ZR(IS+3)+DZ/3.	00018100
	XR(IS+1)=XR(IS+3)+TTH*DX-DY*TC2	00018110
	YR(IS+1)=YR(IS+3)+TTH*DY+DX*TC2	00018120
	ZR(IS+1)=ZR(IS+3)+TTH*DZ	00018130
140	IS=IS+4	00018140
C*	CALC XR,YR,ZR FOR TRAILING VORTICES	00018150
	DT1=TAN1(NVC)/WNVT	00018160
	DT2=TAN2(NVC)/WNVT	00018170
	TAN1(I)=TAN1(NVC)-DT1	00018180
	TAN2(I)=TAN2(NVC)-DT2	00018190
	DO 150 I=2,NVT	00018200
	TAN1(I)=TAN1(I-1)-DT1	00018210
150	TAN2(I)=TAN2(I-1)-DT2	00018220
C		00018230
	DO 160 I=1,NVT	00018240
	DX=XR(IS)-XR(IS+3)	00018250
	DY=YR(IS)-YR(IS+3)	00018260
	DZ=ZR(IS)-ZR(IS+3)	00018270
	DXDY=DX/DY	00018280
	TC1=(TAN1(I)-DXDY)/(1.+DXDY*TAN1(I))/3.	00018290
	TC2=(TAN2(I)-DXDY)/(1.+DXDY*TAN2(I))/3.	00018300
	XR(IS+2)=XR(IS+3)+DX/3.+DY*TC1	00018310
	YR(IS+2)=YR(IS+3)+DY/3.-DX*TC1	00018320
	ZR(IS+2)=ZR(IS+3)+DZ/3.	00018330
	XR(IS+1)=XR(IS+3)+TTH*DX-DY*TC2	00018340
	YR(IS+1)=YR(IS+3)+TTH*DY+DX*TC2	00018350
	ZR(IS+1)=ZR(IS+3)+TTH*DZ	00018360
160	IS=IS+4	00018370
C*	CALC CHORDS FOR WING ROOT SECTION	00018380
	WPE(3)=SQRT((WPE(2)-XR(4))**2+(WPE(1)*HWS-YR(4))**2)+WPE(2)	00018390
	WPE(6)=SQRT((WPE(5)-XR(3))**2+(WPE(4)*HWS-YR(3))**2)+WPE(5)	00018400
	WPE(9)=SQRT((WPE(8)-XR(2))**2+(WPE(7)*HWS-YR(2))**2)+WPE(8)	00018410
	IS=4	00018420
	DO 190 I=2,NVC	00018430
	IS=IS+4	00018440
	WPE(3)=SQRT((XR(IS-4)-XR(IS))**2+(YR(IS-4)-YR(IS))**2)+WPE(3)	00018450
	WPE(6)=SQRT((XR(IS-5)-XR(IS-1))**2+(YR(IS-5)-YR(IS-1))**2)+WPE(6)	00018460
190	WPE(9)=SQRT((XR(IS-6)-XR(IS-2))**2+(YR(IS-6)-YR(IS-2))**2)+WPE(9)	00018470
	WPE(3)=SQRT((XR(IS)-X1)**2+(YR(IS)-Y1)**2)+WPE(3)	00018480
	WPE(6)=SQRT((XR(IS-1)-XP1)**2+(YR(IS-1)-YP1)**2)+WPE(6)	00018490

	WPE(9)=SQRT((XR(IS-2)-XP2)**2+(YR(IS-2)-YP2)**2)+WPE(9)	00018500
C*	COMPRESS. EFFECT ON WING ROOT PARAMETERS	00018510
	IS=4*(NVC+NVT)	00018520
	IF(BETA.EQ.1.) GO TO 196	00018530
	DO 192 I=1,IS	00018540
	YR(I)=YR(I)*BETA	00018550
192	ZR(I)=ZR(I)*BETA	00018560
C	SHIFT ANS.	00018570
196	IF(IST.EQ.0) GO TO 220	00018580
	IX=0	00018590
	DO 200 I=1,IS,4	00018600
	IC=I+603	00018610
	DO 200 J=1,4	00018620
	IX=IX+1	00018630
	XR(IC)=XR(IX)	00018640
	YR(IC)=YR(IX)	00018650
	ZR(IC)=ZR(IX)	00018660
200	IC=IC-I	00018670
C*	CALC CONTROL POINTS AT SPECIFIED WJC AND WJS	00018680
	IS=90	00018690
	DO 210 IJS=1,NJS	00018700
	IY=WJS(IJS)	00018710
	IE=IY-NE3	00018720
	IF(IE.LE.3) GO TO 210	00018730
	IF(IE.GT.6) GO TO 210	00018740
	DO 205 IJC=1,NJC	00018750
	IX=WJC(IJC)	00018760
	IC=4*IX+IE+597	00018770
	IS=IS+1	00018780
	XD(IS)=(XR(IC)+XR(IC+1)+XR(IC-4)+XR(IC-3))/4.	00018790
	YD(IS)=(YR(IC)+YR(IC+1)+YR(IC-4)+YR(IC-3))/4.	00018800
205	ZD(IS)=(ZR(IC)+ZR(IC+1)+ZR(IC-4)+ZR(IC-3))/4.	00018810
210	CONTINUE	00018820
	WPT(IXX+2)=WPE(3)	00018830
	WPT(IXX-1)=WPE(6)	00018840
	WPT(IXX-4)=WPE(9)	00018850
	GO TO 300	00018860
220	IS=0	00018870
	DO 230 IJS=1,NJS	00018880
	IY=WJS(IJS)	00018890
	IE=IY-NE3	00018900
	IF(IE.LE.0) GO TO 230	00018910
	IF(IE.GT.3) GO TO 230	00018920

```

DO 225 IJC=1,NJC                                00018930
IX=WJC(IJC)                                     00018940
IC=4*IX+IE                                       00018950
IS=IS+1                                           00018960
XD(IS)=(XR(IC)+XR(IC+1)+XR(IC-4)+XR(IC-3))/4.    00018970
YD(IS)=(YR(IC)+YR(IC+1)+YR(IC-4)+YR(IC-3))/4.    00018980
225 ZD(IS)=(ZR(IC)+ZR(IC+1)+ZR(IC-4)+ZR(IC-3))/4. 00018990
230 CONTINUE                                       00019000
WPT(IXX+5)=WPE(3)                                00019010
WPT(IXX+8)=WPE(6)                                00019020
WPT(IXX+11)=WPE(9)                               00019030
300 RETURN                                        00019040
END                                                00019045
*DECK RTWI                                         00019050
SUBROUTINE RTWI                                    00019060
C                                                  00019070
C*          ROTATE WING ROOT COORDINATES TO THE ACTUAL CHORD LINES. 00019080
C                                                  00019090
COMMON/DAT/ DA(5000)                              00019100
DIMENSION WET(1),TWI(1),TDZL(1),DZLE(52),EX(52)  00019110
EQUIVALENCE (DA(731),WET) ,(DA(1970),TWI) ,(DA(1985),TDZL) 00019120
C                                                  00019130
COMMON/CRG/ PI,PI4,RC,BETA                        00019140
COMMON/CFG/ IDB,IDF                               00019150
COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3,NE5,NTB,NU,NW,NVC 00019160
C                                                  00019170
LARGE      DM1(51700),EV(52) ,DM2(7144)          00019180
1          ,XB1(600) ,XB2(600) ,XB3(600) ,DM3(1800) 00019190
2          ,ZB1(600) ,ZB2(600) ,ZB3(600) ,DM4(200) 00019200
3          ,ET(52) ,X ,DM5(6722)                 00019210
4          ,COSTT(52) ,SINTT(52) ,COSTV(52) ,SINTV(52) 00019220
5          ,XB(600) ,XB1(600) ,XBR(600)          00019230
6          ,ZB(600) ,ZB1(600) ,ZBR(600)          00019240
C                                                  00019250
C          LOOKUP FOR TWIST                          00019260
NEW=DA(730)                                       00019270
NVSH=NE2+1                                        00019280
CALL TWIST (NEW,WET,TWI,NVSH,EV,COSTV,SINTV)     00019290
CALL TWIST (NEW,WET,TWI,NVSH,ET,COSTT,SINTT)     00019300
C          LOOKUP FOR DZLE AT TRAIL VORTICES       00019310
IF(NEW.LT.1) NEW=1                               00019320
DO 5 I=1,NVSH                                     00019330
5 EX(I)=ET(I)                                     00019340

```

CALL COD (NEW,WET,TDZL,NVSH,EX,DZLE)	00019350
DO 10 I=1,NVSH	00019360
IF(DA(730).LT.1.0) DZLE(I)=0.0	00019370
10 DZLE(I)=DZLE(I)*BETA	00019380
C ROTATE XB AND ZB	00019390
NES=NE5	00019400
NEE=NE2	00019410
IF(IDF.EQ.0) GO TO 40	00019420
NE4=NE3+7	00019430
NE6=NE3+3	00019440
DZLE(NE6)=0.0	00019450
COSTT(NE6)=1.0	00019460
SINTT(NE6)=0.0	00019470
NVO=NVC+NT0	00019480
NVI=NVC+NTI	00019490
IS=0	00019500
DO 20 IX=1,NVO	00019510
DO 20 IE=NE3,NE6	00019520
IS=IS+1	00019530
DXX=XB1(IS)-XLT(IE)	00019540
XB(IS)=XLT(IE)+DXX*COSTT(IE)	00019550
20 ZB(IS)=ZB1(IS)-DXX*SINTT(IE)+DZLE(IE)	00019560
C	00019570
NEX=NE3+4	00019580
DZLE(NEX)=0.0	00019590
COSTT(NEX)=1.0	00019600
SINTT(NEX)=0.0	00019610
IS=0	00019620
DO 30 IX=1,NVI	00019630
DO 30 IE=NEX,NE4	00019640
IS=IS+1	00019650
DXX=XB2(IS)-XLT(IE)	00019660
XB1(IS)=XLT(IE)+DXX*COSTT(IE)	00019670
30 ZB1(IS)=ZB2(IS)-DXX*SINTT(IE)+DZLE(IE)	00019680
C	00019690
NES=NES+1	00019700
NEE=NEE+1	00019710
40 IF(IDB.EQ.0) GO TO 80	00019720
NVO=NVC+NTB	00019730
DZLE(NEE)=0.0	00019740
COSTT(NEE)=1.0	00019750
SINTT(NEE)=0.0	00019760
IS=0	00019770

	DO 60 IX=1,NVO	00019780
	DO 60 IE=NES,NEE	00019790
	IS=IS+1	00019800
	DXX=XB3(IS)-XLT(IE)	00019810
	XBR(IS)=XLT(IE)+DXX*COSTT(IE)	00019820
	60 ZBR(IS)=ZB3(IS)-DXX*SINTT(IE)+DZLE(IE)	00019830
C		00019840
	80 CONTINUE	00019850
C	LOOKUP FOR DZLE AT CENTER OF VORTICES	00019860
	DO 85 I=1,NVSH	00019870
	85 EX(I)=EV(I)	00019880
	CALL COD (NEW,WET,TDZL,NVSH,EX,DZLE)	00019890
	DO 90 I=1,NVSH	00019900
	IF(DA(730).LT.1.0) DZLE(I)=0.0	00019910
	90 SINTT(I)=DZLE(I)	00019920
	RETURN	00019930
	END	00019940
	*DECK TWIST	00019950
	SUBROUTINE TWIST (NEI,EI,II,NEO,EO,CT,ST)	00019960
C		00019970
C*	LOOKUP TWIST ANGLE ANDCALC COS AND SIN OF TWIST	00019980
C		00019990
	LARGE EO(1),CT(1),ST(1)	00020000
	DIMENSION EI(1),II(1)	00020010
C		00020020
C		00020030
	DO 60 I=1,NEO	00020040
	IF(NEI.NE.0) GO TO 5	00020050
	CT(I)=1.0	00020060
	ST(I)=0.0	00020070
	GO TO 60	00020080
	5 ETA=ABS(EO(I))	00020090
	IF(ETA.GT.EI(1)) GO TO 10	00020100
	TW=II(1)	00020110
	GO TO 50	00020120
	10 DO 20 J=2,NEI	00020130
	IF(ETA-EI(J)) 40,30,20	00020140
	20 CONTINUE	00020150
	TW=II(NEI)	00020160
	GO TO 50	00020170
	30 TW=II(J)	00020180
	GO TO 50	00020190
	40 RATY=(EI(J)-ETA)/(EI(J)-EI(J-1))	00020200

	TW=TI(J)-RATY*(TI(J)-TI(J-1))	00020210
50	CT(I)=COS(TW)	00020220
	ST(I)=SIN(TW)	00020230
60	CONTINUE	00020240
	RETURN	00020250
	END	00020260
*DECK	MATCS	00020270
	SUBROUTINE MATCS	00020280
C		00020290
C*	CALC MATRICES C AND S	00020300
C		00020310
	COMMON/DAT/ DA(5000)	00020320
C		00020330
	DIMENSION W(10)	00020340
C		00020350
	EQUIVALENCE (DA(1285),W) ,(NI,NVC) ,(DA(1281),WNW)	00020360
	1,(DA(1960),DAF) ,(DA(1961),PXCF) ,(DA(1965),DAK) ,(DA(1966),PXCK)	00020370
	2,(DA(2),WS)	00020380
C		00020390
	COMMON/CRG/ PI,PI4,RC,BETA	00020400
	COMMON/CPW/ DW1(9),NU,NW,NVC,NVS	00020410
	COMMON/CFK/ NDA,NUT,THEF,SINTF,SINTF2,COSTF,THEK,SINTK,SINTK2	00020420
	1 ,COSTK,SINDF,COSDF,SINDK,COSDK	00020430
C		00020440
	LARGE BOU(3000) ,FUS(25000),FAN(23700),EV(52) ,DM1(12544)	00020450
	1 ,CFI(50) ,CFJ(50) ,STH(50) ,ACT(50) ,DM2(156)	00020460
	2 ,C(40,40) ,S(51,10) ,CSEC(40,10),DM3(7968)	00020470
	3 ,PYL(6061) ,CEL(3700) ,BC(40,10)	00020480
C*	CALC MATRIX C	00020490
C	FLAP SETUP	00020500
	NDA=0	00020510
	IF(DAF.NE.0.0) GO TO 4	00020520
	THEF=0.0	00020530
	SINTF=0.0	00020540
	SINTF2=0.0	00020550
	GO TO 10	00020560
4	NDA=1	00020570
	COSTF=1.-2.*PXCF	00020580
	THEF= ACOS(COSTF)	00020590
	SINTF=SIN(THEF)	00020600
	SINTF2=SIN(2.*THEF)	00020610
	SINDF=SIN(DAF)	00020620
	COSDF=COS(DAF)	00020630

	NUS=NU-1	00020640
	IF(DAK.EQ.0.0) NUS=NU	00020650
	BC(1,NUS)=SINTF/2.	00020660
	PIT=THEF-PI	00020670
	DO 5 I=2,NI	00020680
	IF(THEF.LE.ACT(I)) GO TO 6	00020690
5	BC(I,NUS)=PIT	00020700
6	KF=I	00020710
	DO 8 J=KF,NI	00020720
8	BC(J,NUS)=THEF	00020730
C	KRUEGER SETUP	00020740
10	IF(DAK.NE.0.0) GO TO 12	00020750
	THEK=0.0	00020760
	SINTK=0.0	00020770
	SINTK2=0.0	00020780
	GO TO 20	00020790
12	NDA=NDA+1	00020800
	COSTK=1.-2.*PXCK	00020810
	THEK= ACOS(COSTK)	00020820
	SINTK=SIN(THEK)	00020830
	SINTK2=SIN(2.*THEK)	00020840
	SINDK=SIN(DAK)	00020850
	COSDK=COS(DAK)	00020860
	BC(1,NU)=SINTK/2.	00020870
	PIT=THEK-PI	00020880
	DO 14 I=2,NI	00020890
	IF(THEK.LT.ACT(I)) GO TO 16	00020900
14	BC(I,NU)=PIT	00020910
16	KK=I	00020920
	DO 18 J=KK,NI	00020930
18	BC(J,NU)=THEK	00020940
20	NUT=NU-NDA	00020950
C*		00020960
	XI=0.0	00020970
	DO 22 I=2,NUT	00020980
	XI=XI+1.	00020990
	BC(1,I)=0.0	00021000
	DO 22 J=2,NI	00021010
22	BC(J,I)=-COS(XI*ACT(J))	00021020
	BC(1,2)=.25	00021030
	DO 24 I=1,NI	00021040
	C(1,I)=1.	00021050
	BC(I,1)=1.	00021060

```

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(I,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)

```

```

00021070
00021080
00021090
00021100
00021110
00021120
00021130
00021140
00021150
00021160
00021170
00021180
00021190
00021200
00021210
00021220
00021230
00021240
00021250
00021260
00021270
00021280
00021290
00021300
00021310
00021320
00021330
00021340
00021350
00021360
00021370
00021380
00021390
00021400
00021410
00021420
00021430
00021440
00021450
00021460
00021470
00021480
00021490

```

C							00021500
C	COMMON/CPW/	HWS,NTI,NTO,DLI,DLO,NE2,NE3,DW1(4),NVC,NVS,NJC,NJS					00021510
							00021520
	LARGE	BOU(3000),FUS(25000),FAN(23700)					00021530
	1, EV(52)	,XLV(52),XTLV(52),DPI(840),WPT(60),BS(650)					00021540
	2	,EP(30),XTLP(30),XOCP(30)					00021550
	3	,CSJ(30,30),TSJ(30,30),DM1(9000)					00021560
	4	,CFI(50),CFJ(50),DM2(2766)					00021570
	5	,TSI(52,40),TZI(52,40),DM3(3808),PYL(6061),CEL(3700)					00021580
	6	,CHD(24),ZD(24),ZCJ(50),ZCI(50),ZJ(50,20)					00021590
	7	,ZI(50,20),EL(20),XOCL(24)					00021600
C							00021610
C*		INITIAL					00021620
	LD=0						00021630
	NDE=DA(2025)						00021640
	NXD=DA(2000)						00021650
C*		SHIFT TO LCM FOR WINGD AND CODIM					00021660
	DO 10 I=1,NDE						00021670
	10 EL(I)=ED(I)						00021680
	DO 20 I=1,NXD						00021690
	20 XOCL(I)=XOCD(I)						00021700
	CALL WINGD (NDE,EL,ZD,CHD,WPT)						00021710
	NII=NI+1						00021720
	CFJ(NII)=1.						00021730
	DO 100 I=1,NDE						00021740
	30 DO 40 IX=1,NXD						00021750
	LD=LD+1						00021760
	40 ZD(IX)=TZC(LD)*CHD(I)						00021770
	CALL CODIM (XOCL,ZD,NXD,CFJ,ZCJ,NII)						00021780
	CALL CODIM (XOCL,ZD,NXD,CFI,ZCI,NI)						00021790
	DO 60 K=1,NI						00021800
	ZJ(K,I)=ZCJ(K)						00021810
	60 ZI(K,I)=ZCI(K)						00021820
	ZJ(NII,I)=ZCJ(NII)						00021830
	100 CONTINUE						00021840
C							00021850
C		LINEAR INTERPOLATION FOR Z/C SPANWISE AND CALC SLOPE					00021860
	DO 200 IE=1,NEP						00021870
	IF(EP(IE).GT.ED(1)) GO TO 120						00021880
	II=1						00021890
	GO TO 140						00021900
	120 DO 130 IS=2,NDE						00021910
	II=IS						00021920

IF(EP(IE)-ED(IS)) 150,140,130	00021930
130 CONTINUE	00021940
140 DO 145 J=1,NJ	00021950
IC=WJC(J)+1.	00021960
145 TSJ(IE,J)=(ZI(IC,II)-ZI(IC-1,II))/(CFI(IC)-CFI(IC-1))/XTLP(IE)	00021970
GO TO 200	00021980
150 RATY=(ED(II)-EP(IE))/(ED(II)-ED(II-1))	00021990
160 DO 170 J=1,NJ	00022000
IC=WJC(J)+1.	00022010
Z1=ZI(IC-1,II)-RATY*(ZI(IC-1,II)-ZI(IC-1,II-1))	00022020
Z2=ZI(IC,II)-RATY*(ZI(IC,II)-ZI(IC,II-1))	00022030
170 TSJ(IE,J)=(Z2-Z1)/(CFI(IC)-CFI(IC-1))/XTLP(IE)	00022040
200 CONTINUE	00022050
DO 300 IE=1,NE2	00022060
IF(EV(IE).GT.ED(1)) GO TO 220	00022070
II=1	00022080
GO TO 240	00022090
220 DO 230 IS=2,NDE	00022100
II=IS	00022110
IF(EV(IE)-ED(IS)) 250,240,230	00022120
230 CONTINUE	00022130
240 DO 245 I=1,NI	00022140
TZI(IE,I)=ZI(I,II)	00022150
245 TSI(IE,I)=(ZJ(I+1,II)-ZJ(I,II))/(CFJ(I+1)-CFJ(I))	00022160
GO TO 300	00022170
250 RATY=(ED(II)-EV(IE))/(ED(II)-ED(II-1))	00022180
260 DO 270 I=1,NI	00022190
Z1=ZJ(I,II)-RATY*(ZJ(I,II)-ZJ(I,II-1))	00022200
Z2=ZJ(I+1,II)-RATY*(ZJ(I+1,II)-ZJ(I+1,II-1))	00022210
TZI(IE,I)=ZI(I,II)-RATY*(ZI(I,II)-ZI(I,II-1))	00022220
270 TSI(IE,I)=(Z2-Z1)/(CFJ(I+1)-CFJ(I))	00022230
300 CONTINUE	00022240
LT=NE2+1	00022250
DO 320 I=1,NI	00022260
TZI(LT,I)=ZI(I,NDE)	00022270
320 TSI(LT,I)=(ZJ(I+1,NDE)-ZJ(I,NDE))/(CFJ(I+1)-CFJ(I))	00022280
RETURN	00022290
END	00022300
*DECK PXYZ	00022310
SUBROUTINE PXYZ (INVX,NDV,NVBP,FOX,FVX,XB,YB,ZB)	00022320
C	00022330
C*	00022340
C	00022350

CALC. COORDINATES OF PYLON VORTICES AND CONTROL POINTS

```

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),POL)
2,(DA(2507),PNVC),(DA(2508),PNVS) ,(DA(2512),PJC) ,(DA(2960),PPT)
C
COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP
COMMON/PPP/ HPS,NVSH,NVT,DL,NEVP,ZCON,YCOF,NVC,NVS,NJC,NUP
C
LARGE DMI(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.

```

```

00022360
00022370
00022380
00022390
00022400
00022410
00022420
00022430
00022440
00022450
00022460
00022470
00022480
00022490
00022500
00022510
00022520
00022530
00022540
00022550
00022560
00022570
00022580
00022590
00022600
00022610
00022620
00022630
00022640
00022650
00022660
00022670
00022680
00022690
00022700
00022710
00022720
00022730
00022740
00022750
00022760
00022770
00022780

```

```

DO 6 J=2,NVC                                00022790
6 C(J,I)=1./(CFJ(J)-CFI(I))                 00022800
BC(1,1)=.5                                  00022810
CALL MSOLX (NVC,NVC,NUP,C,BC,20)           00022820
C                                             00022830
SPAN=PHT+PHT                                00022840
DO 8 I=1,NUP                                 00022850
CSEC(1,I)=C(1,I)*SPAN                       00022860
DO 8 K=2,NVC                                00022870
8 CSEC(K,I)=C(K,I)*SPAN+CSEC(K-1,I)         00022880
C                                             00022890
SHIFT PYLON PLANFORM TABLE                00022900
MPS=PHT                                      00022910
DO 10 I=1,37,3                              00022920
WPE(I)=PPT(I)                               00022930
WPE(I+1)=PPT(I+1)                           00022940
WPE(I+2)=PPT(I+2)                           00022950
IF(PPT(I).GE.1.) GO TO 12                   00022960
10 CONTINUE                                  00022970
12 NPP=I-2                                    00022980
C#                                           00022990
CALC VORTEX LATTICE SPACE SPANWISE         00023000
DL=(1.-WPE(10))/(PNVS-3.5)                 00023010
NEVP=NVS-3                                  00023020
EV(NEVP+1)=(WPE(10)+WPE(7))/2.             00023030
EV(NEVP+2)=(WPE( 7)+WPE(4))/2.             00023040
EV(NEVP+3)=(WPE( 4)+WPE(1))/2.             00023050
30 EV(2)=1.-DL                               00023060
EV(1)=EV(2)+.625*DL                         00023070
ET(1)=1.-DL/2.                              00023080
ET(2)=ET(1)-DL                              00023090
DO 40 I=3,NEVP                               00023100
ET(I)=ET(I-1)-DL                           00023110
40 EV(I)=EV(I-1)-DL                         00023120
NVSH=NVS+1                                  00023130
EV(NVSH)=1.0                                00023140
CALL WINGD (NEVP,ET,XLT,XLT,WPE)           00023150
C                                             00023160
TEST WING PLANFORM FOR NACELLE TYPE BREAK 00023170
DO 44 I=1,NPP,3                              00023180
IF(ABS(WPE(I)-WPE(I+3)).GT.0.01) GO TO 44  00023190
DO 42 J=1,NEVP                               00023200
IF(ABS(WPE(I)-ET(J)).GT.0.005) GO TO 42    00023210
ET(J)=-ET(J)                                00023220
XLT(J)=DA(I+1)                              00023230
XLT(J)=DA(I+2)-DA(I+1)                     00023240

```

42	CONTINUE	00023220
44	CONTINUE	00023230
C		00023240
	IF(ATP.EQ.3.0) GO TO 102	00023250
C*	SEARCH LIST OF FUSELAGE X FOR XLE AND XTE OF WING ROOT	00023260
	DO 50 I=1,NVX	00023270
	FVXT=FVX(I)+FOX-POX	00023280
	IF(WPE(2).LT.FVXT) GO TO 60	00023290
50	CONTINUE	00023300
60	IS=I	00023310
	DO 70 I=IS,NVX	00023320
	FVXT=FVX(I)+FOX-POX	00023330
	IF(WPE(3).LT.FVXT) GO TO 90	00023340
70	CONTINUE	00023350
C		00023360
C*	SHIFT VORTEX COORD. OF FUSELAGE TO THE MERGING ROOT	00023370
90	IE=I	00023380
	ITV=PITH-1.0	00023390
	IC=(IS-2)*NVBP+2+ITV*NDV	00023400
	IC1=IC-NVBP	00023410
	IF(IC1.LE.0) IC1=IC	00023420
	NPT=IE-IS+2	00023430
	DO 92 I=1,NPT	00023440
	XD(I)=XB(IC1)-POX	00023450
	YP=YB(IC1)/BETA-POY	00023460
	ZP=ZB(IC1)/BETA-POZ	00023470
	YD(I)=YP*SINDP+ZP*COSDP	00023480
	ZD(I)=ZP*SINDP-YP*COSDP	00023490
92	IC1=IC1+NVBP	00023500
	CD=WPE(3)-WPE(2)	00023510
	DO 94 I=1,NVC	00023520
94	XRF(I)=WPE(2)+CD*CFI(I)	00023530
	CALL CODIM (XD,YD,NPT,XRF,YRF,NVC)	00023540
	CALL CODIM (XD,ZD,NPT,XRF,ZRF,NVC)	00023550
	IS=0	00023560
	DO 96 I=1,NVC	00023570
	IS=IS+4	00023580
	XR(IS)=XRF(I)	00023590
	YR(IS)=YRF(I)	00023600
96	ZR(IS)=ZRF(I)	00023610
	XRS=XR(IS)	00023620
	NVT=NVX-IE+1	00023630
	IC=IC1-NVBP	00023640

	DO 100 I=1,NVT	00023650
	IS=IS+4	00023660
	XR(IS)=XB(IC)-POX	00023670
	YP=YB(IC)/BETA-POY	00023680
	ZP=ZB(IC)/BETA-POZ	00023690
	YR(IS)=YP*SINDP+ZP*COSDP	00023700
	ZR(IS)=ZP*SINDP-YP*COSDP	00023710
100	IC=IC+NVBP	00023720
	GO TO 108	00023730
C	PYLON ATTACH TO WING	00023740
102	CD=WPE(3)-WPE(2)	00023750
	YP=WPE(1)*HPS	00023760
	IS=0	00023770
	DO 104 I=1,NVC	00023780
	IS=IS+4	00023790
	XR(IS)=WPE(2)+CD*CFI(I)	00023800
	YR(IS)=YP	00023810
104	ZR(IS)=0.0	00023820
	XRS=XR(IS)	00023830
	DX=XRS-XR(IS-4)	00023840
	NVT=2	00023850
	DO 106 I=1,NVT	00023860
	IS=IS+4	00023870
	XR(IS)=XR(IS-4)+DX	00023880
	YR(IS)=YP	00023890
106	ZR(IS)=0.0	00023900
C		00023910
C*	CALC VORTEX COORD. OF MERGING ROOT TO PLANAR WING	00023920
108	ZCON=0.0	00023930
	YCON=HPS*WPE(10)	00023940
	XRE=XR(IS)	00023950
	WNVT=NVT	00023960
	CD=WPE(12)-WPE(11)	00023970
	IS=-3	00023980
	DO 110 I=1,NVC	00023990
	IS=IS+4	00024000
	XR(IS)=WPE(11)+CD*CFI(I)	00024010
	YR(IS)=YCON	00024020
110	ZR(IS)=ZCON	00024030
	XRR=XR(IS)	00024040
	DXF=XRE-XRS	00024050
	DXR=XRE-XRR	00024060
	DO 120 I=1,NVT	00024070

	IS=IS+4	00024080
	XR(IS)=(XR(IS+3)-XRS)/DXF*DXR+XRR	00024090
	YR(IS)=YCON	00024100
120	ZR(IS)=ZCON	00024110
C		00024120
C*	CALC COORD. OF THE TWO CENTER ROOT VORTEX LINES	00024130
	TTH=.6666667	00024140
	IX=4*NVC	00024150
	X1=WPE(3)	00024160
	Y1=(YR(IX)+YR(IX+4))/2.	00024170
	X2=WPE(12)	00024180
	Y2=YCON	00024190
	DX=X2-X1	00024200
	DY=Y2-Y1	00024210
	DZ=ZCON-(ZR(IX)+ZR(IX+4))/2.	00024220
	SQD=SQRT(DX**2+DY**2+DZ**2)	00024230
	XP1=X1+DX/3.-WPE(6)*DY/SQD	00024240
	YP1=Y1+DY/3.+WPE(6)*DX/SQD	00024250
	XP2=X1+TTH*DX-WPE(9)*DY/SQD	00024260
	YP2=Y1+TTH*DY+WPE(9)*DX/SQD	00024270
	TT1=(XP1-X1)/(YP1-Y1)	00024280
	TT2=(X2-XP2)/(Y2-YP2)	00024290
	DY=(WPE(4)-WPE(1))*HPS	00024300
	TL1=(WPE(5)-WPE(2))/DY	00024310
	TL2=(WPE(11)-WPE(8))/DY	00024320
C		00024330
	DT1=(TT1-TL1)/PNVC	00024340
	DT2=(TT2-TL2)/PNVC	00024350
	TAN1(1)=TL1+DT1/2.	00024360
	TAN2(1)=TL2+DT2/2.	00024370
	DO 130 I=2,NVC	00024380
	TAN1(I)=TAN1(I-1)+DT1	00024390
130	TAN2(I)=TAN2(I-1)+DT2	00024400
C		00024410
	IS=1	00024420
	DO 140 I=1,NVC	00024430
	DX=XR(IS)-XR(IS+3)	00024440
	DY=YR(IS)-YR(IS+3)	00024450
	DZ=ZR(IS)-ZR(IS+3)	00024460
	DXDY=DX/DY	00024470
	TC1=(TAN1(I)-DXDY)/(1.+DXDY*TAN1(I))/3.	00024480
	TC2=(TAN2(I)-DXDY)/(1.+DXDY*TAN2(I))/3.	00024490
	XR(IS+2)=XR(IS+3)+DX/3.+DY*TC1	00024500

	YR(IS+2)=YR(IS+3)+DY/3.-DX*TC1	00024510
	ZR(IS+2)=ZR(IS+3)+DZ/3.	00024520
	XR(IS+1)=XR(IS+3)+TTH*DX-DY*TC2	00024530
	YR(IS+1)=YR(IS+3)+TTH*DY+DX*TC2	00024540
	ZR(IS+1)=ZR(IS+3)+TTH*DZ	00024550
140	IS=IS+4	00024560
C*	CALC XR,YR,ZR FOR TRAILING VORTICES	00024570
	DT1=TAN1(NVC)/WNVT	00024580
	DT2=TAN2(NVC)/WNVT	00024590
	TAN1(1)=TAN1(NVC)-DT1	00024600
	TAN2(1)=TAN2(NVC)-DT2	00024610
	DO 150 I=2,NVT	00024620
	TAN1(I)=TAN1(I-1)-DT1	00024630
150	TAN2(I)=TAN2(I-1)-DT2	00024640
C		00024650
	DO 160 I=1,NVT	00024660
	DX=XR(IS)-XR(IS+3)	00024670
	DY=YR(IS)-YR(IS+3)	00024680
	DZ=ZR(IS)-ZR(IS+3)	00024690
	DXDY=DX/DY	00024700
	TC1=(TAN1(1)-DXDY)/(1.+DXDY*TAN1(1))/3.	00024710
	TC2=(TAN2(1)-DXDY)/(1.+DXDY*TAN2(1))/3.	00024720
	XR(IS+2)=XR(IS+3)+DX/3.+DY*TC1	00024730
	YR(IS+2)=YR(IS+3)+DY/3.-DX*TC1	00024740
	ZR(IS+2)=ZR(IS+3)+DZ/3.	00024750
	XR(IS+1)=XR(IS+3)+TTH*DX-DY*TC2	00024760
	YR(IS+1)=YR(IS+3)+TTH*DY+DX*TC2	00024770
	ZR(IS+1)=ZR(IS+3)+TTH*DZ	00024780
160	IS=IS+4	00024790
C		00024800
C*	CALC CONTROL POINTS AT SPECIFIED WJC AND WJS	00024810
	IS=0	00024820
	DO 180 IY=1,NVS	00024830
	IE=IY-NEVP	00024840
	IF(IE.LE.0) GO TO 180	00024850
	DO 170 IJC=1,NJC	00024860
	IX=PJC(IJC)	00024870
	IC=4*IX+IE	00024880
	IS=IS+1	00024890
	XD(IS)=(XR(IC)+XR(IC+1)+XR(IC-4)+XR(IC-3))/4.	00024900
	YD(IS)=(YR(IC)+YR(IC+1)+YR(IC-4)+YR(IC-3))/4.	00024910
170	ZD(IS)=(ZR(IC)+ZR(IC+1)+ZR(IC-4)+ZR(IC-3))/4.	00024920
180	CONTINUE	00024930

```

      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 SLOPES FROM 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/PPP/ HPS,NVSH,DP1(5),NVC,NVS,NJC

```

```

00024940
00024950
00024960
00024970
00024980
00024990
00025000
00025010
00025020
00025030
00025040
00025050
00025060
00025070
00025080
00025090
00025100
00025110
00025120
00025130
00025140
00025150
00025160
00025170
00025180
00025190
00025200
00025210
00025220
00025230
00025240
00025250
00025260
00025270
00025280
00025290
00025300
00025310
00025320
00025330
00025340
00025350
00025360

```

C					00025370	
	LARGE	DM1(75130)			00025380	
	1	◦EV(22)	◦XLV(22)	◦XTLV(22)	◦WPE(39)	00025390
	2	◦DP2(1380)	◦CFI(20)	◦CFJ(20)		00025400
	3	◦DP3(506)	◦CSJ(20,20)	◦TSJ(20,20)		00025410
	4	◦DP4(2390)	◦TSI(21,20)	◦TZI(21,20)	◦CEL(3700)	00025420
	5	◦CHD(20)	◦ZD(20)	◦ZCJ(20)		00025430
	6	◦ZCI(20)	◦ZJ(20,20)	◦ZI(20,20)		00025440
	7	◦EL(20)	◦XOCL(20)			00025450
C					00025460	
C		INITIAL			00025470	
		LD=0			00025480	
		NDE=DA(3020)			00025490	
		NXD=DA(3000)			00025500	
C			SHIFT TO LCM FOR WINGD AND CODIM		00025510	
		DO 10 I=1,NDE			00025520	
	10	EL(I)=ED(I)			00025530	
		DO 20 I=1,NXD			00025540	
	20	XOCL(I)=XOCD(I)			00025550	
C					00025560	
		CALL WINGD (NDE,EL,ZD,CHD,WPE)			00025570	
		NI1=NVC+1			00025580	
		CFJ(NI1)=1.			00025590	
		DO 100 I=1,NDE			00025600	
	30	DO 40 IX=1,NXD			00025610	
		LD=LD+1			00025620	
	40	ZD(IX)=TZC(LD)*CHD(I)			00025630	
		CALL CODIM (XOCL,ZD,NXD,CFJ,ZCJ,NI1)			00025640	
		CALL CODIM (XOCL,ZD,NXD,CFI,ZCI,NVC)			00025650	
		DO 60 K=1,NVC			00025660	
		ZJ(K,I)=ZCJ(K)			00025670	
	60	ZI(K,I)=ZCI(K)			00025680	
		ZJ(NI1,I)=ZCJ(NI1)			00025690	
	100	CONTINUE			00025700	
C					00025710	
C			LINEAR INTERPOLATION FOR Z/C SPANWISE AND CALC. SLOPE		00025720	
		DO 200 IE=1,NVS			00025730	
		IF(EV(IE).GT.ED(1)) GO TO 120			00025740	
		II=1			00025750	
		GO TO 140			00025760	
	120	DO 130 IS=2,NDE			00025770	
		II=IS			00025780	
		IF(EV(IE)-ED(IS)) 150,140,130			00025790	

130	CONTINUE	00025800
140	DO 145 J=1,NJC	00025810
	IC=PJC(J)+1.	00025820
145	TSJ(IE,J)=(ZI(IC,II)-ZI(IC-1,II))/(CFI(IC)-CFI(IC-1))/XTLV(IE)	00025830
	GO TO 200	00025840
150	RATY=(ED(II)-EV(IE))/(ED(II)-ED(II-1))	00025850
160	DO 170 J=1,NJC	00025860
	IC=PJC(J)+1.	00025870
	Z1=ZI(IC-1,II)-RATY*(ZI(IC-1,II)-ZI(IC-1,II-1))	00025880
	Z2=ZI(IC,II)-RATY*(ZI(IC,II)-ZI(IC,II-1))	00025890
170	TSJ(IE,J)=(Z2-Z1)/(CFI(IC)-CFI(IC-1))/XTLV(IE)	00025900
200	CONTINUE	00025910
	NEVH=NVSH-1	00025920
	DO 300 IE=1,NEVH	00025930
	IF(EV(IE).GT.ED(1)) GO TO 220	00025940
	II=1	00025950
	GO TO 240	00025960
220	DO 230 IS=2,NDE	00025970
	II=IS	00025980
	IF(EV(IE)-ED(IS)) 250,240,230	00025990
230	CONTINUE	00026000
240	DO 245 I=1,NVC	00026010
	TZI(IE,I)=ZI(I,II)	00026020
245	TSI(IE,I)=(ZJ(I+1,II)-ZJ(I,II))/(CFJ(I+1)-CFJ(I))	00026030
	GO TO 300	00026040
250	RATY=(ED(II)-EV(IE))/(ED(II)-ED(II-1))	00026050
260	DO 270 I=1,NVC	00026060
	Z1=ZJ(I,II)-RATY*(ZJ(I,II)-ZJ(I,II-1))	00026070
	Z2=ZJ(I+1,II)-RATY*(ZJ(I+1,II)-ZJ(I+1,II-1))	00026080
	TZI(IE,I)=ZI(I,II)-RATY*(ZI(I,II)-ZI(I,II-1))	00026090
270	TSI(IE,I)=(Z2-Z1)/(CFJ(I+1)-CFJ(I))	00026100
300	CONTINUE	00026110
	LT=NVSH	00026120
	DO 320 I=1,NVC	00026130
	TZI(LT,I)=ZI(I,NDE)	00026140
320	TSI(LT,I)=(ZJ(I+1,NDE)-ZJ(I,NDE))/(CFJ(I+1)-CFJ(I))	00026150
	RETURN	00026160
	END	00026170
*DECK	NXYZ	00026180
	SUBROUTINE NXYZ	00026190
C		00026200
C*	SETUP NACELLE COORDINATES AND PARAMETERS	00026210
C		00026220

	COMMON/DAT/ DA(5000)	00026230
C		00026240
	DIMENSION XB1(1),YB1(1),XB2(1),YB2(1)	00026250
C		00026260
	EQUIVALENCE (DA(3400),XB1) ,(DA(3550),YB1) ,(DA(3700),XB2)	00026270
	1,(DA(3850),YB2) ,(ND(1),NP1) ,(ND(2),NP2)	00026280
C		00026290
	COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP	00026300
	COMMON/CFG/ ID1(5),NQB,NQF,NQW,NQP,NQN	00026310
	COMMON/CPN/ NB,ND(3),NT	00026320
C		00026330
	LARGE DM1(81191),DM2(3000)	00026340
	1 ,X1(140) ,Y1(140) ,SINAN(140),COSAN(140),DELS(140)	00026350
C		00026360
C		00026370
	ND(NB+1)=NQB+NQF+NQW+NQP	00026380
	DO 10 I=1,NP1	00026390
	X1(I)=XB1(I)	00026400
	10 Y1(I)=YB1(I)*BETA	00026410
C		00026420
	IF(NB.EQ.1) GO TO 30	00026430
	IC=NP1	00026440
	DO 20 I=1,NP2	00026450
	IC=IC+1	00026460
	X1(IC)=XB2(I)	00026470
	20 Y1(IC)=YB2(I)*BETA	00026480
C		00026490
	30 N1=0	00026500
	J1=0	00026510
	DO 50 K=1,NB	00026520
	M1=N1+1	00026530
	N1=N1+ND(K)-1	00026540
	DO 40 J=M1,N1	00026550
	J1=J1+1	00026560
	T1=X1(J1+1)-X1(J1)	00026570
	T2=Y1(J1+1)-Y1(J1)	00026580
	DELS(J)=SQRT(T1*T1+T2*T2)	00026590
	COSAN(J)=T1/DELS(J)	00026600
	40 SINAN(J)=T2/DELS(J)	00026610
	50 J1=J1+1	00026620
C		00026630
	RETURN	00026640
	END	00026650

*DECK	CODMT	00026660
	SUBROUTINE CODMT (NXS,NTY,NVB,XI,YI,TH,RZ)	00026670
C		00026680
C*	A CONTROLLED DEVIATION ITERPOLATION METHOD	00026690
C*	WITH MULTIPLE TABLES	00026700
C		00026710
	DIMENSION XI(1),YI(1)	00026720
C		00026730
	LARGE TH(1),RZ(1)	00026740
C		00026750
	DATA XK/0.5/	00026760
C*	START	00026770
	N=NTY	00026780
	NM3=N-3	00026790
	IC=0	00026800
	DO 400 IT=1,NVB	00026810
	X=TH(IT)	00026820
	J=1	00026830
	IF(N-2) 10,30,50	00026840
10	DO 20 IX=1,NXS	00026850
	IC=IC+1	00026860
	RZ(IC)=YI(J)	00026870
20	J=J+N	00026880
	GO TO 400	00026890
30	RXI=(X-XI(1))/(XI(2)-XI(1))	00026900
	DO 40 IX=1,NXS	00026910
	IC=IC+1	00026920
	RZ(IC)=(YI(J+1)-YI(J))*RXI+YI(J)	00026930
40	J=J+N	00026940
	GO TO 400	00026950
50	IF(XI(J)-X) 60,10,70	00026960
60	J=J+1	00026970
	IF(J-N) 50,50,70	00026980
70	IF(J-2) 10,80,90	00026990
80	J=3	00027000
	JJ=1	00027010
	GO TO 120	00027020
90	IF(J-N) 100,110,10	00027030
100	JJ=3	00027040
	GO TO 120	00027050
110	J=N-1	00027060
	JJ=2	00027070
120	IF(NM3) 130,130,140	00027080

130	J=3	00027090
140	K=J-1	00027100
	M=K-1	00027110
	L=J+1	00027120
	A1=X-XI(M)	00027130
	A2=X-XI(K)	00027140
	A3=X-XI(J)	00027150
	C1=A3*A2/((XI(M)-XI(K))*(XI(M)-XI(J)))	00027160
	C2=A1*A3/((XI(K)-XI(M))*(XI(K)-XI(J)))	00027170
	C3=A2*A1/((XI(J)-XI(M))*(XI(J)-XI(K)))	00027180
	IF(NM3) 160,160,150	00027190
150	A4=X-XI(L)	00027200
	C4=A4*A3/((XI(K)-XI(J))*(XI(K)-XI(L)))	00027210
	C5=A2*A4/((XI(J)-XI(K))*(XI(J)-XI(L)))	00027220
	C6=A3*A2/((XI(L)-XI(K))*(XI(L)-XI(J)))	00027230
160	IF(JJ-2) 170,180,190	00027240
170	AL=(X-XI(1))/(XI(2)-XI(1))	00027250
	GO TO 200	00027260
180	AL=(X-XI(N-1))/(XI(N)-XI(N-1))	00027270
	GO TO 200	00027280
190	AL=(X-XI(K))/(XI(J)-XI(K))	00027290
200	DO 300 IX=1,NXS	00027300
	P1=C1*YI(M)+C2*YI(K)+C3*YI(J)	00027310
	IF(NM3) 210,210,220	00027320
210	P2=P1	00027330
	GO TO 230	00027340
220	P2=C4*YI(K)+C5*YI(J)+C6*YI(L)	00027350
230	IF(JJ-2) 240,250,260	00027360
240	P2=P1	00027370
	IA=(IX-1)*N+1	00027380
	S=AL*YI(IA+1)+(1.-AL)*YI(IA)	00027390
	P1=S+XK*(P2-S)	00027400
	GO TO 270	00027410
250	P1=P2	00027420
	IA=IX*N	00027430
	S=AL*YI(IA)+(1.-AL)*YI(IA-1)	00027440
	P2=S+XK*(P1-S)	00027450
	GO TO 270	00027460
260	S=AL*YI(J)+(1.-AL)*YI(K)	00027470
270	E1=ABS(P1-S)	00027480
	E2=ABS(P2-S)	00027490
	IC=IC+1	00027500
	IF(E1+E2) 280,280,290	00027510

```

280 RZ( IC )=S                                00027520
GO TO 295                                      00027530
290 BT=(E1*AL)/(E1*AL+(1.-AL)*E2)            00027540
RZ( IC )=BT*P2+(1.-BT)*P1                    00027550
295 J=J+N.                                     00027560
K=K+N.                                         00027570
L=L+N.                                         00027580
M=M+N.                                         00027590
300 CONTINUE                                  00027600
400 CONTINUE                                  00027610
RETURN                                         00027620
END                                             00027630
*DECK MATA                                     00027640
SUBROUTINE MATA                               00027650
C                                               00027660
C*          CONTROL ROUTINE FOR THE CALC OF THE INFLUENCE MATRIX-A 00027670
C                                               00027680
COMMON/DAT/ DA(5000)                          00027690
C                                               00027700
DIMENSION WJS(1),PJC(1),XB1(1),YB1(1),XB2(1),YB2(1) 00027710
C                                               00027720
EQUIVALENCE (DA(1202),WOZ) ,(DA(1690),WJS) ,(DA(2501),POX) 00027730
1,(DA(2502),POY) ,(DA(2503),POZ) ,(DA(2517),PJC) ,(DA(3391),XNO) 00027740
2,(DA(3392),YNO) ,(DA(3393),ZNO) ,(DA(3400),XB1) ,(DA(3550),YB1) 00027750
3,(DA(3700),XB2) ,(DA(3850),YB2)              00027760
C                                               00027770
COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP        00027780
COMMON/CFG/ IDB,IDF,IDW,IDP,IDN,NQB,NQF,NQW,NQP,NQN 00027790
COMMON/CPW/ HWS,IW1(5),NE3,NE5,IW2(5),NJC,NJS 00027800
COMMON/CPV/ HPS,IP1(3),NEVP,IP2(3),NVSP,NJCP 00027810
COMMON/CPN/ NB,NP1,NP2                        00027820
C                                               00027830
LARGE DM1(3000) ,XQB(650) ,YQB(650) ,ZQB(650) ,DM2(23050) 00027840
1 ,XQF(650) ,YQF(650) ,ZQF(650) ,DM3(21750) 00027850
2 ,EV(52) ,XLV(52) ,XTLV(52) 00027860
3 ,XD(270) ,YD(270) ,ZD(270) ,DM4(740) 00027870
4 ,EP(30) ,XTLP(30) ,XOCP(30) ,DM5(21634) 00027880
5 ,EVP(22) ,XLVP(22) ,XTLVP(22) ,WPE(39) 00027890
6 ,XDP(60) ,YDP(60) ,ZDP(60) ,DM6(1220) ,CFJP(20) 00027900
7 ,DM7(4536) ,XOB(1000) ,YOB(1000) ,ZOB(1000) ,DM8(700) 00027910
8 ,XA(650) ,YA(650) ,ZA(650) 00027920
C                                               00027930
C*          INITIAL                            00027940

```

	ICN=0	00027950
	I19=2	00027960
	I20=2	00027970
	I21=2	00027980
	I22=2	00027990
C		00028000
C*	CONTROL POINTS Q ON FUSELAGE	00028010
	IF(IDB.EQ.0) GO TO 100	00028020
	DO 30 I=1,NQB	00028030
	XA(I)=XQB(I)	00028040
	YA(I)=YQB(I)	00028050
	30 ZA(I)=ZQB(I)	00028060
C	CALC INFL OF FUSELAGE	00028070
	IF(IDB.EQ.1) GO TO 44	00028080
	DO 42 I=1,NQB	00028090
	42 READ (19)	00028100
	GO TO 50	00028110
	44 I19=1	00028120
	CALL AQXVB (NQB,1)	00028130
C	CALC INFL OF FANPOD	00028140
	50 IF(IDF.EQ.0) GO TO 60	00028150
	IDX=IDB+IDF	00028160
	IF(IDX.NE.4) GO TO 54	00028170
	DO 52 I=1,NQB	00028180
	52 READ (20)	00028190
	GO TO 60	00028200
	54 I20=1	00028210
	CALL AQXVF(NQB,2)	00028220
C	CALC INFL OF WING	00028230
	60 IF(IDW.EQ.0) GO TO 70	00028240
	IDX=IDB+IDW	00028250
	IF(IDX.NE.4) GO TO 64	00028260
	DO 62 I=1,NQB	00028270
	62 READ (21)	00028280
	GO TO 70	00028290
	64 I21=1	00028300
	CALL AQXVW(NQB,3)	00028310
C	CALC INFL OF PYLON	00028320
	70 IF(IDP.EQ.0) GO TO 80	00028330
	IDX=IDB+IDP	00028340
	IF(IDX.NE.4) GO TO 74	00028350
	DO 72 I=1,NQB	00028360
	72 READ (22)	00028370

	GO TO 80	00028380
74	I22=1	00028390
	CALL AQXVP(NQB,4)	00028400
C	SETUP NACELLE OFF BODY POINTS	00028410
80	IF(IDN.EQ.0) GO TO 100	00028420
	DO 82 I=1,NQB	00028430
	ICN=ICN+1	00028440
	XOB(ICN)=XA(I)	00028450
	YOB(ICN)=YA(I)	00028460
82	ZOB(ICN)=ZA(I)	00028470
C		00028480
C*	CONTROL POINTS Q ON FANPOD	00028490
100	IF(IDF.EQ.0) GO TO 200	00028500
	DO 130 I=1,NQF	00028510
	XA(I)=XQF(I)	00028520
	YA(I)=YQF(I)	00028530
130	ZA(I)=ZQF(I)	00028540
C	CALC INFL OF FUSELAGE	00028550
	IF(IDB.EQ.0) GO TO 150	00028560
	IDX=IDF+I19+IDB	00028570
	IF(IDX.NE.6) GO TO 144	00028580
	DO 142 I=1,NQF	00028590
142	READ (19)	00028600
	GO TO 150	00028610
144	I19=1	00028620
	CALL AQXVB(NQF,6)	00028630
C	CALC INFL OF FANPOD	00028640
150	IDX=IDF+120	00028650
	IF(IDX.NE.4) GO TO 154	00028660
	DO 152 I=1,NQF	00028670
152	READ (20)	00028680
	GO TO 160	00028690
154	I20=1	00028700
	CALL AQXVF(NQF,7)	00028710
C	CALC INFL OF WING	00028720
160	IF(IDW.EQ.0) GO TO 170	00028730
	IDX=IDF+I21+IDW	00028740
	IF(IDX.NE.6) GO TO 164	00028750
	DO 162 I=1,NQF	00028760
162	READ (21)	00028770
	GO TO 170	00028780
164	I21=1	00028790
	CALL AQXVW(NQF,8)	00028800

C	CALC INFL OF PYLON	00028810
170	IF(IDP.EQ.0) GO TO 180	00028820
	IDX=IDF+I22+IDP	00028830
	IF(IDX.NE.6) GO TO 174	00028840
	DO 172 I=1,NQF	00028850
172	READ (22)	00028860
	GO TO 180	00028870
174	I22=1	00028880
	CALL AQXVP(NQF,9)	00028890
C	SETUP NACELLE OFF BODY POINTS	00028900
180	IF(IDN.EQ.0) GO TO 200	00028910
	DO 182 I=1,NQF	00028920
	ICN=ICN+1	00028930
	XOB(ICN)=XA(I)	00028940
	YOB(ICN)=YA(I)	00028950
182	ZOB(ICN)=ZA(I)	00028960
C		00028970
C*	CONTROL POINTS Q ON WING	00028980
200	IF(IDW.EQ.0) GO TO 300	00028990
	IR=0	00029000
	IQ=0	00029010
	NE4=NE3+7	00029020
	NET=NE3+4	00029030
	NE6=NE5+1	00029040
	DO 220 I=1,NJS	00029050
	IE=WJS(I)	00029060
	Y=HWS*EP(I)	00029070
	DO 220 J=1,NJC	00029080
	IQ=IQ+1	00029090
	IF(IDF.EQ.0) GO TO 210	00029100
	IF(IE.LE.NE3) GO TO 214	00029110
	IF(IE.GE.NE4) GO TO 210	00029120
	IF(IE.EQ.NET.AND.J.EQ.1) IR=90	00029130
	GO TO 212	00029140
210	IF(IE.LE.NE5) GO TO 214	00029150
	IF(IE.EQ.NE6.AND.J.EQ.1) IR=180	00029160
212	IR=IR+1	00029170
	XA(IQ)=XD(IR)	00029180
	YA(IQ)=YD(IR)	00029190
	ZA(IQ)=ZD(IR)	00029200
	GO TO 220	00029210
214	XA(IQ)=XLV(IE)+XTLV(IE)*XOCP(J)	00029220
	YA(IQ)=Y	00029230

	ZA(IQ)=WOZ*BETA	00029240
220	CONTINUE	00029250
C	CALC INFL OF FUSELAGE	00029260
	IF(IDB.EQ.0) GO TO 250	00029270
	IDX=IDW+I19+IDB	00029280
	IF(IDX.NE.6) GO TO 244	00029290
	DO 242 I=1,NQW	00029300
242	READ (19)	00029310
	GO TO 250	00029320
244	I19=1	00029330
	CALL AQXVB(NQW,11)	00029340
C	CALC INFL OF FANPOD	00029350
250	IF(IDF.EQ.0) GO TO 260	00029360
	IDX=IDW+I20+IDF	00029370
	IF(IDX.NE.6) GO TO 254	00029380
	DO 252 I=1,NQW	00029390
252	READ (20)	00029400
	GO TO 260	00029410
254	I20=1	00029420
	CALL AQXVF(NQW,12)	00029430
C	CALC INFL OF WING	00029440
260	IDX=IDW+I21	00029450
	IF(IDX.NE.4) GO TO 264	00029460
	DO 262 I=1,NQW	00029470
262	READ (21)	00029480
	GO TO 270	00029490
264	I21=1	00029500
	CALL AQWVW(13)	00029510
C	CALC INFL OF PYLON	00029520
270	IF(IDP.EQ.0) GO TO 280	00029530
	IDX=IDW+I22+IDP	00029540
	IF(IDX.NE.6) GO TO 274	00029550
	DO 272 I=1,NQW	00029560
272	READ (22)	00029570
	GO TO 280	00029580
274	I22=1	00029590
	CALL AQXVP(NQW,14)	00029600
C	SETUP NACELLE OFF BODY POINTS	00029610
280	IF(IDN.EQ.0) GO TO 300	00029620
	DO 282 I=1,NQW	00029630
	ICN=ICN+1	00029640
	XOB(ICN)=XA(I)	00029650
	YOB(ICN)=YA(I)	00029660

282	ZOB(ICN)=ZA(I)	00029670
C		00029680
C*	CONTROL POINTS Q ON PYLON	00029690
300	IF(IDP.EQ.0) GO TO 400	00029700
	IR=0	00029710
	IQ=0	00029720
	PYB=POY*BETA	00029730
	PZB=POZ*BETA	00029740
	DO 320 I=1,NVSP	00029750
	Y=HPS*EVP(I)	00029760
	YC=PYB+Y*SINDP	00029770
	ZC=PZB+Y*COSDP	00029780
	DO 320 J=1,NJCP	00029790
	IJ=PJC(J)	00029800
	IQ=IQ+1	00029810
	IF(I.LE.NEVP) GO TO 310	00029820
	IR=IR+1	00029830
	XA(IQ)=XDP(IR)+POX	00029840
	YA(IQ)=(YDP(IR)*SINDP-ZDP(IR)*COSDP)+PYB	00029850
	ZA(IQ)=(YDP(IR)*COSDP+ZDP(IR)*SINDP)+PZB	00029860
	GO TO 320	00029870
310	XA(IQ)=XLVP(I)+XTLVP(I)*CFJP(IJ+1)+POX	00029880
	YA(IQ)=YC	00029890
	ZA(IQ)=ZC	00029900
320	CONTINUE	00029910
C	CALC INFL OF FUSELAGE	00029920
	IF(IDB.EQ.0) GO TO 350	00029930
	IDX=IDP+I19+IDB	00029940
	IF(IDX.NE.6) GO TO 344	00029950
	DO 342 I=1,NQP	00029960
342	READ (19)	00029970
	GO TO 350	00029980
344	I19=1	00029990
	CALL AQXVB(NQP,16)	00030000
C	CALC INFL OF FANPOD	00030010
350	IF(IDF.EQ.0) GO TO 360	00030020
	IDX=IDP+I20+IDF	00030030
	IF(IDX.NE.6) GO TO 354	00030040
	DO 352 I=1,NQP	00030050
352	READ (20)	00030060
	GO TO 360	00030070
354	I20=1	00030080
	CALL AQXVF(NQP,17)	00030090

C	CALC INFL OF WING	00030100
360	IF(IDW.EQ.0) GO TO 370	00030110
	IDX=IDP+I21+IDW	00030120
	IF(IDX.NE.6) GO TO 364	00030130
	DO 362 I=1,NQP	00030140
362	READ (21)	00030150
	GO TO 370	00030160
364	I21=1	00030170
	CALL AQXVW(NQP,18)	00030180
C	SETUP NACELLE OFF BODY POINTS	00030190
370	IF(IDN.EQ.0) GO TO 380	00030200
	DO 372 I=1,NQP	00030210
	ICN=ICN+1	00030220
	XOB(ICN)=XA(I)	00030230
	YOB(ICN)=YA(I)	00030240
372	ZOB(ICN)=ZA(I)	00030250
C	CALC INFL OF PYLON	00030260
380	IDX=IDP+I22	00030270
	IF(IDX.NE.4) GO TO 384	00030280
	DO 382 I=1,NQP	00030290
382	READ (22)	00030300
	GO TO 400	00030310
384	I22=1	00030320
	CALL AQPVP	00030330
	CALL AQXVP(NQP,19)	00030340
C		00030350
C*	CONTROL POINTS Q ON NACELLE	00030360
400	IF(IDN.EQ.0) GO TO 500	00030370
	YNOB=YNO*BETA	00030380
	ZNOB=ZNO*BETA	00030390
	DO 410 I=2,NP1	00030400
	XA(I-1)=XNO+(XB1(I)+XB1(I-1))/2.	00030410
	YN=(YB1(I)+YB1(I-1))/2.*BETA	00030420
	YA(I-1)=YNOB-YN*SINDP	00030430
410	ZA(I-1)=ZNOB-YN*COSDP	00030440
	IF(NB.EQ.1) GO TO 430	00030450
	IC=NP1-1	00030460
	DO 420 I=2,NP2	00030470
	IC=IC+1	00030480
	XA(IC)=XNO+(XB2(I)+XB2(I-1))/2.	00030490
	YN=(YB2(I)+YB2(I-1))/2.	00030500
	YA(IC)=YNOB-YN*SINDP	00030510
420	ZA(IC)=ZNOB-YN*COSDP	00030520

C	SETUP NACELLE OFF BODY POINTS	00030530
430	DO 440 I=1,NQN	00030540
	ICN=ICN+1	00030550
	XOB(ICN)=XA(I)	00030560
	YOB(ICN)=YA(I)	00030570
440	ZOB(ICN)=ZA(I)	00030580
C	CALC INFL OF FUSELAGE	00030590
	IF(IDB.EQ.0) GO TO 450	00030600
	IDX=IDN+I19+IDB	00030610
	IF(IDX.NE.6) GO TO 444	00030620
	DO 442 I=1,NQN	00030630
442	READ (19)	00030640
	GO TO 450	00030650
444	I19=1	00030660
	CALL AQXVB(NQN,21)	00030670
C	CALC INFL OF FANPOD	00030680
450	IF(IDF.EQ.0) GO TO 460	00030690
	IDX=IDN+I20+IDF	00030700
	IF(IDX.NE.6) GO TO 454	00030710
	DO 452 I=1,NQN	00030720
452	READ (20)	00030730
	GO TO 460	00030740
454	I20=1	00030750
	CALL AQXVF(NQN,22)	00030760
C	CALC INFL OF WING	00030770
460	IF(IDW.EQ.0) GO TO 470	00030780
	IDX=IDN+I21+IDW	00030790
	IF(IDX.NE.6) GO TO 464	00030800
	DO 462 I=1,NQN	00030810
462	READ (21)	00030820
	GO TO 470	00030830
464	I21=1	00030840
	CALL AQXVW(NQN,23)	00030850
C	CALC INFL OF PYLON	00030860
470	IF(IDP.EQ.0) GO TO 500	00030870
	IDX=IDN+I22+IDP	00030880
	IF(IDX.NE.6) GO TO 474	00030890
	DO 472 I=1,NQN	00030900
472	READ (22)	00030910
	GO TO 500	00030920
474	I22=1	00030930
	CALL AQXVP(NQN,24)	00030940
C		00030950

```

C*          REIND I/O UNITS                                00030960
  500 IF(IDB.NE.0) REWIND 19                               00030970
      IF(IDF.NE.0) REWIND 20                               00030980
      IF(IDW.NE.0) REWIND 21                               00030990
      IF(IDP.NE.0) REWIND 22                               00031000
C                                                    00031010
C*          CALC INFL OF NACELLE ON OTHER COMPONENTS      00031020
      IF(IDN.NE.0) CALL NACELL                             00031030
C                                                    00031040
C*          COMBINE INFLUENCE MATRICES AND SETUP FOR MATRIX SOLU. 00031050
      CALL MATAPE                                         00031060
C                                                    00031070
      RETURN                                              00031080
      END                                                  00031090
*DECK AQXVB                                              00031100
      SUBROUTINE AQXVB(NQ,ITYP)                             00031110
C                                                    00031120
C*          CALC MATRIX-A (Q ON X-COMPONENT AND VORTICES ON FUSE.) 00031130
C                                                    00031140
      COMMON/DAT/ DA(5000)                                00031150
C                                                    00031160
      EQUIVALENCE      (DA(3386),XLFX),(DA(3387),XLFY),(DA(4707),FNDV) 00031170
C                                                    00031180
      COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP              00031190
      COMMON/CFG/ IG1(10),NKB                             00031200
      COMMON/CPB/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX,IFBE  00031210
C                                                    00031220
      LARGE          DM1(10800)                            00031230
*          ,PM(150,25),DFM(150) ,DM2(1300)              00031240
*          ,XB(4000) ,YB(4000) ,ZB(4000) ,DM3(56891)    00031250
1          ,XQ(650) ,YQ(650) ,ZQ(650)                  00031260
2          ,AX(250) ,AY(250) ,AZ(250)                   00031270
3          ,QBX(51) ,QBY(51) ,QBZ(51)                   00031280
4          ,QBXP(51) ,QBYP(51) ,QBZP(51)                00031290
5          ,QTX(51) ,QTY(51) ,QTZ(51)                   00031300
6          ,DL(51) ,DLP(51) ,SL(51) ,PLM(51,25)         00031310
C                                                    00031320
C*          START                                          00031330
      XLFY2=XLFY*XLFY                                     00031340
C          NO. OF CONTROL POINTS Q LOOP                  00031350
      DO 400 IQ=1,NQ                                     00031360
C          CLEAR SUMMATION MATRICES                      00031370
      IF(NF.NE.0) GO TO 11                               00031380

```

	DO 10 I=1,NVY	00031390
	QBXP(I)=0.0	00031400
	QBYP(I)=0.0	00031410
10	QBZP(I)=0.0	00031420
	GO TO 14	00031430
11	DO 12 I=1,NKB	00031440
	AX(I)=0.0	00031450
	AY(I)=0.0	00031460
12	AZ(I)=0.0	00031470
C	NVX AND NVY LOOPS	00031480
14	IPAS=0	00031490
	DO 300 IX=1,NVXM	00031500
	LI=(IX-1)*NVBP+2	00031510
	LF=LI+1	00031520
	NVYX=NVY+NVY	00031530
C	TEST TO BYPASS X-SEGMENT	00031540
	IF(IQ.EQ.1) GO TO 15	00031550
	IF(IX.GE.(NVXM-1)) GO TO 15	00031560
	TT=ABS(XQ(IQ)-XB(LI))/DFM(IX)	00031570
	IF(TT.GT.XLFX) GO TO 270	00031580
15	DZ2=(ZB(LI)-ZQ(IQ))**2	00031590
	DYB=YB(LI)-YB(LI+1)	00031600
	IF(DYB.EQ.0.0) GO TO 16	00031610
	DF2=(FNDV*DYB)**2	00031620
16	TT=(((-YB(LI)-YQ(IQ))**2+DZ2)/DF2	00031630
	IF(TT.GT.XLFY2) NVYX=NVY	00031640
	DO 180 IS=1,NVYX	00031650
	K=NDV	00031660
	IF(IX.NE.NVXM) GO TO 18	00031670
	K=0	00031680
	LF=LI+NDV	00031690
	LI=LF-NVBP	00031700
18	ISS=IS-NVY	00031710
	IF(ISS) 20,20,32	00031720
20	JS=IS	00031730
	QBX(JS)=0.0	00031740
	QBY(JS)=0.0	00031750
	QBZ(JS)=0.0	00031760
30	TY1=YB(LF)-YB(LI)	00031770
	TY2=YB(LF)-YQ(IQ)	00031780
	TY3=YB(LI)-YQ(IQ)	00031790
	GO TO 50	00031800
32	JS=ISS	00031810

40	TY1=YB(LI)-YB(LF)	00031820
	TY2=-YB(LF)-YQ(IQ)	00031830
	TY3=-YB(LI)-YQ(IQ)	00031840
50	TX1=XB(LF)-XB(LI)	00031850
	TX2=XB(LF)-XQ(IQ)	00031860
	TX3=XB(LI)-XQ(IQ)	00031870
	TZ1=ZB(LF)-ZB(LI)	00031880
	TZ2=ZB(LF)-ZQ(IQ)	00031890
	TZ3=ZB(LI)-ZQ(IQ)	00031900
	HN=TX2*TX1+TY2*TY1+TZ2*TZ1	00031910
	HD=TX1**2+TY1**2+TZ1**2	00031920
	SQHD=SQRT(HD)	00031930
	IF(SQHD.GT.0.00001) GO TO 52	00031940
51	QSR=0.0	00031950
	GO TO 54	00031960
52	HT=HN/HD	00031970
	HX=TX2-HT*TX1	00031980
	HY=TY2-HT*TY1	00031990
	HZ=TZ2-HT*TZ1	00032000
	SHP=SQRT(HX**2+HY**2+HZ**2)	00032010
	IF(SHP.LT.0.00001) GO TO 51	00032020
	SRF=HN/SQRT(TX2**2+TY2**2+TZ2**2)	00032030
	HI=TX1*TX3+TY1*TY3+TZ1*TZ3	00032040
	SRI=HI/SQRT(TX3**2+TY3**2+TZ3**2)	00032050
	QT=(SRF-SRI)/(SQHD*SHP*PI4)	00032060
	TX=TY2*TZ1-TZ2*TY1	00032070
	TY=TZ2*TX1-TX2*TZ1	00032080
	TZ=TX2*TY1-TY2*TX1	00032090
	QSR=QT/SQRT(TX**2+TY**2+TZ**2)	00032100
	IF(ISS.GT.0) QSR=-QSR	00032110
54	K=K-1	00032120
	IF(K) 140,60,60	00032130
C*	LATERAL VORTICES	00032140
60	QBX(JS)=QSR*TX+QBX(JS)	00032150
	QBY(JS)=QSR*TY+QBY(JS)	00032160
	QBZ(JS)=QSR*TZ+QBZ(JS)	00032170
	IF(K) 100,100,70	00032180
C	SETUP FOR ADDITIONAL LATERAL	00032190
70	LI=LF	00032200
	LF=LF+1	00032210
	IF(ISS) 30,30,40	00032220
C	SETUP FOR LONGITUDINAL	00032230
100	LI=LF-NVBP	00032240

	IF(IX.EQ.1) LI=1	00032250
	IF(ISS) 30,30,40	00032260
C*	LONGITUDINAL VORTICES	00032270
140	IF(ISS) 160,155,150	00032280
150	QTX(JS)=QSR*TX+QTX(JS)	00032290
	QTY(JS)=QSR*TY+QTY(JS)	00032300
	QTZ(JS)=QSR*TZ+QTZ(JS)	00032310
	GO TO 170	00032320
155	LF=LI+1	00032330
160	QTX(JS)=QSR*TX	00032340
	QTY(JS)=QSR*TY	00032350
	QTZ(JS)=QSR*TZ	00032360
	DL(JS)=SQMD	00032370
170	LI=LF	00032380
	LF=LF+1	00032390
180	CONTINUE	00032400
	IF(IPAS.EQ.0) GO TO 190	00032410
	IPAS=0	00032420
	READ (23)	00032430
	GO TO 255	00032440
190	IF(IX.NE.NVXM) GO TO 198	00032450
	IF(IFBE.EQ.0) GO TO 198	00032460
C	* COMPUTE TRAIL VORTICES	00032470
191	DO 196 I=1,NVY	00032480
	PASS=1.	00032490
	HY=YB(LI)-YQ(IQ)	00032500
	HX=XB(LI)-XQ(IQ)	00032510
	HZ=ZB(LI)-ZQ(IQ)	00032520
192	DD1=HY*HY+HZ*HZ	00032530
	IF(DD1.LT.0.00001) GO TO 193	00032540
	DD2=SQRT(HX*HX+DD1)	00032550
	IF(DD2.LT.0.00001) GO TO 193	00032560
	QSR=(1.-HX/DD2)/DD1*PASS/PI4	00032570
	QTY(JS)=QTY(JS)+QSR*HZ	00032580
	QTZ(JS)=QTZ(JS)-QSR*HY	00032590
193	IF(PASS) 195,195,194	00032600
194	PASS=-1.	00032610
	HY=-YB(LI)-YQ(IQ)	00032620
	GO TO 192	00032630
195	LI=LI-NDV	00032640
196	JS=JS-1	00032650
C*	CALC MATRIX L	00032660
198	IF(NF.LE.0) GO TO 210	00032670

IF(ITYP.NE.1) GO TO 200	00032680
IF(IQ.EQ.1) GO TO 201	00032690
200 IF(IX.EQ.1) GO TO 255	00032700
READ (23) ((PLM(J,I),J=1,NVY),I=1,NF)	00032710
GO TO 210	00032720
201 DFM(IX)=DL(1)	00032730
IF(IX.NE.1) GO TO 204	00032740
DLP(1)=DL(1)	00032750
DO 202 J=2,NVY	00032760
202 DLP(J)=(DL(J-1)+DL(J))/2.	00032770
DO 203 J=1,NVY	00032780
DO 203 I=1,NF	00032790
203 PLM(J,I)=0.0	00032800
GO TO 255	00032810
204 DLX=DL(1)/2.	00032820
SL(1)=DLP(1)+DLX	00032830
DLP(1)=DLX	00032840
DO 205 J=2,NVY	00032850
DLX=(DL(J-1)+DL(J))/4.	00032860
SL(J)=DLP(J)+DLX	00032870
205 DLP(J)=DLX	00032880
IF(IX.NE.NVXM) GO TO 207	00032890
DO 206 J=1,NVY	00032900
206 SL(J)=SL(J)+DLP(J)	00032910
207 DO 208 J=1,NVY	00032920
DO 208 I=1,NF	00032930
208 PLM(J,I)=PM(IX,I)*SL(J)+PLM(J,I)	00032940
WRITE (23) ((PLM(J,I),J=1,NVY),I=1,NF)	00032950
C* SUM FOR AX,AY AND AZ	00032960
?10 TX=QBXP(1)-QBX(1)+QTX(1)-QTX(NVY)	00032970
TY=QBYP(1)-QBY(1)+QTY(1)-QTY(NVY)	00032980
TZ=QBZP(1)-QBZ(1)+QTZ(1)-QTZ(NVY)	00032990
IF(NF) 212,212,214	00033000
212 AX(IX)=TX	00033010
AY(IX)=TY	00033020
AZ(IX)=TZ	00033030
GO TO 230	00033040
214 DO 220 I=1,NF	00033050
AX(I)=TX*PLM(1,I)+AX(I)	00033060
AY(I)=TY*PLM(1,I)+AY(I)	00033070
220 AZ(I)=TZ*PLM(1,I)+AZ(I)	00033080
IC=NF	00033090
230 DO 250 J=2,NVY	00033100

	TX=QBXP(J)-QBX(J)+QTX(J)-QTX(J-1)	00033110
	TY=QBYP(J)-QBY(J)+QTY(J)-QTY(J-1)	00033120
	TZ=QBZP(J)-QBZ(J)+QTZ(J)-QTZ(J-1)	00033130
	IF(NF) 232,232,234	00033140
232	IC=(J-1)*NVXM+IX	00033150
	AX(IC)=TX	00033160
	AY(IC)=TY	00033170
	AZ(IC)=TZ	00033180
	GO TO 250	00033190
234	DO 240 I=1,NF	00033200
	IC=IC+1	00033210
	AX(IC)=TX*PLM(J,I)+AX(IC)	00033220
	AY(IC)=TY*PLM(J,I)+AY(IC)	00033230
240	AZ(IC)=TZ*PLM(J,I)+AZ(IC)	00033240
250	CONTINUE	00033250
255	DO 260 I=1,NVY	00033260
	QBXP(I)=QBX(I)	00033270
	QBYP(I)=QBY(I)	00033280
260	QBZP(I)=QBZ(I)	00033290
	GO TO 300	00033300
270	IPAS=1	00033310
	IF(IX.EQ.1) GO TO 300	00033320
	READ(23)	00033330
300	CONTINUE	00033340
	REWIND 23	00033350
C		00033360
	IF(ITYP.LT.16) GO TO 320	00033370
	DO 310 I=1,NKB	00033380
	AYT=AY(I)	00033390
	AY(I)=AYT*SINDP+AZ(I)*COSDP	00033400
310	AZ(I)=-AYT*COSDP+AZ(I)*SINDP	00033410
C		00033420
	320 WRITE(19)(AX(IC),AY(IC),AZ(IC),IC=1,NKB)	00033430
	400 CONTINUE	00033440
	WRITE(6,450) ITYP	00033450
	450 FORMAT(21H0 ** COMPLETED MATRIX,14)	00033460
	RETURN	00033470
	END	00033480
*DECK	AQXVF	00033490
	SUBROUTINE AQXVF(NQ,ITYP)	00033500
C		00033510
C*	CALC MATRIX-A (Q ON X-COMPONENT AND VORTICES ON FANP.)	00033520
C		00033530

```

COMMON/DAT/ DA(5000)                                00033540
EQUIVALENCE      (DA(3380),XLFX) ,(DA(3381),XLFY) ,(DA(1276),FNDV) 00033550
C
COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP             00033560
COMMON/CFG/ IGI(11),NKF                            00033570
COMMON/CPF/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX,IFBE  00033580
C
LARGE          DMI(35800)                            00033590
*              ,PM(100,25),DFM(100) ,DM2(1300)     00033600
*              ,XB(4000) ,YB(4000) ,ZB(4000) ,DM3(33191) 00033610
1              ,XQ(650) ,YQ(650) ,ZQ(650)         00033620
2              ,AX(250) ,AY(250) ,AZ(250)         00033630
3              ,QBX(51) ,QBY(51) ,QBZ(51)         00033640
4              ,QBXP(51) ,QBYP(51) ,QBZP(51)      00033650
5              ,QTX(51) ,QTY(51) ,QTZ(51)         00033660
6              ,DL(51) ,DLP(51) ,SL(51) ,PLM(51,25) 00033670
C
C*              START                                00033680
PI4=4.*PI                                           00033690
XLFY2=XLFY*XLFY                                     00033700
C              NO. OF CONTROL POINTS Q LOOP        00033710
DO 400 IQ=1,NQ                                       00033720
C              CLEAR SUMMATION MATRICES            00033730
IF(NF.NE.0) GO TO 11                                00033740
DO 10 I=1,NVY                                       00033750
QBXP(I)=0.0                                         00033760
QBYP(I)=0.0                                         00033770
10 QBZP(I)=0.0                                       00033780
GO TO 14                                             00033790
11 DO 12 I=1,NKF                                     00033800
AX(I)=0.0                                           00033810
AY(I)=0.0                                           00033820
12 AZ(I)=0.0                                         00033830
C              NVX AND NVY LOOPS                    00033840
14 IPAS=0                                           00033850
DO 300 IX=1,NVXM                                    00033860
LI=(IX-1)*NVBP+2                                    00033870
LF=LI+1                                             00033880
NVYX=NVY+NVY                                        00033890
C              TEST TO BYPASS X-SEGMENT            00033900
IF(IQ.EQ.1) GO TO 15                                00033910
IF(IX.GE.(NVXM-1)) GO TO 15                        00033920
TT=ABS(XQ(IQ)-XB(LI))/DFM(IX)                       00033930

```


	IF(TT.GT.XLFX) GO TO 270	00033970
15	DZ2=(ZB(LI)-ZQ(IQ))**2	00033980
	DYB=YB(LI)-YB(LI+1)	00033990
	IF(DYB.EQ.0.0) GO TO 16	00034000
	DF2=(FNDV*DYB)**2	00034010
16	TT=((-YB(LI)-YQ(IQ))**2+DZ2)/DF2	00034020
	IF(TT.GT.XLFY2) NVYX=NVY	00034030
	DO 180 IS=1,NVYX	00034040
	K=NDV	00034050
	IF(IX.NE.NVXM) GO TO 18	00034060
	K=0	00034070
	LF=LI+NDV	00034080
	LI=LF-NVBP	00034090
18	ISS=IS-NVY	00034100
	IF(ISS) 20,20,32	00034110
20	JS=IS	00034120
	QBX(JS)=0.0	00034130
	QBY(JS)=0.0	00034140
	QBZ(JS)=0.0	00034150
30	TY1=YB(LF)-YB(LI)	00034160
	TY2=YB(LF)-YQ(IQ)	00034170
	TY3=YB(LI)-YQ(IQ)	00034180
	GO TO 50	00034190
32	JS=ISS	00034200
40	TY1=YB(LI)-YB(LF)	00034210
	TY2=-YB(LF)-YQ(IQ)	00034220
	TY3=-YB(LI)-YQ(IQ)	00034230
50	TX1=XB(LF)-XB(LI)	00034240
	TX2=XB(LF)-XQ(IQ)	00034250
	TX3=XB(LI)-XQ(IQ)	00034260
	TZ1=ZB(LF)-ZB(LI)	00034270
	TZ2=ZB(LF)-ZQ(IQ)	00034280
	TZ3=ZB(LI)-ZQ(IQ)	00034290
	HN=TX2*TX1+TY2*TY1+TZ2*TZ1	00034300
	HD=TX1**2+TY1**2+TZ1**2	00034310
	SQHD=SQRT(HD)	00034320
	IF(SQHD.GT.0.00001) GO TO 52	00034330
51	QSR=0.0	00034340
	GO TO 54	00034350
52	HT=HN/HD	00034360
	HX=TX2-HT*TX1	00034370
	HY=TY2-HT*TY1	00034380
	HZ=TZ2-HT*TZ1	00034390

	SHP=SQRT(HX**2+HY**2+HZ**2)	00034400
	IF(SHP.LT.0.00001) GO TO 51	00034410
	SRF=HN/SQRT(TX2**2+TY2**2+TZ2**2)	00034420
	HI=TX1*TX3+TY1*TY3+TZ1*TZ3	00034430
	SRI=HI/SQRT(TX3**2+TY3**2+TZ3**2)	00034440
	QT=(SRF-SRI)/(SQHD*SHP*PI4)	00034450
	TX=TY2*TZ1-TZ2*TY1	00034460
	TY=TZ2*TX1-TX2*TZ1	00034470
	TZ=TX2*TY1-TY2*TX1	00034480
	QSR=QT/SQRT(TX**2+TY**2+TZ**2)	00034490
	IF(ISS.GT.0) QSR=-QSR	00034500
54	K=K-1	00034510
	IF(K) 140,60,60	00034520
C*	LATERAL VORTICES	00034530
60	QBX(JS)=QSR*TX+QBX(JS)	00034540
	QBY(JS)=QSR*TY+QBY(JS)	00034550
	QBZ(JS)=QSR*TZ+QBZ(JS)	00034560
	IF(K) 100,100,70	00034570
C	SETUP FOR ADDITIONAL LATERAL	00034580
70	LI=LF	00034590
	LF=LF+1	00034600
	IF(ISS) 30,30,40	00034610
C	SETUP FOR LONGITUDINAL	00034620
100	LI=LF-NVBP	00034630
	IF(IX.EQ.1) LI=1	00034640
	IF(ISS) 30,30,40	00034650
C*	LONGITUDINAL VORTICES	00034660
140	IF(ISS) 160,155,150	00034670
150	QTX(JS)=QSR*TX+QTX(JS)	00034680
	QTY(JS)=QSR*TY+QTY(JS)	00034690
	QTZ(JS)=QSR*TZ+QTZ(JS)	00034700
	GO TO 170	00034710
155	LF=LI+1	00034720
160	QTX(JS)=QSR*TX	00034730
	QTY(JS)=QSR*TY	00034740
	QTZ(JS)=QSR*TZ	00034750
	DL(JS)=SQHD	00034760
170	LI=LF	00034770
	LF=LF+1	00034780
180	CONTINUE	00034790
	IF(IPAS.EQ.0) GO TO 190	00034800
	IPAS=0	00034810
	READ (24)	00034820

	GO TO 255	00034830
190	IF(IX.NE.NVXM) GO TO 198	00034840
	IF(IFBE.EQ.0) GO TO 198	00034850
C	* COMPUTE TRAIL VORTICES	00034860
191	DO 196 I=1,NVY	00034870
	PASS=1.	00034880
	HY=YB(LI)-YQ(IQ)	00034890
	HX=XB(LI)-XQ(IQ)	00034900
	HZ=ZB(LI)-ZQ(IQ)	00034910
192	DD1=HY*HY+HZ*HZ	00034920
	IF(DD1.LT.0.00001) GO TO 193	00034930
	DD2=SQRT(HX*HX+DD1)	00034940
	IF(DD2.LT.0.00001) GO TO 193	00034950
	QSR=(1.-HX/DD2)/DD1*PASS/PI4	00034960
	QTY(JS)=QTY(JS)+QSR*HZ	00034970
	QTZ(JS)=QTZ(JS)-QSR*HY	00034980
193	IF(PASS) 195,195,194	00034990
194	PASS=-1.	00035000
	HY=-YB(LI)-YQ(IQ)	00035010
	GO TO 192	00035020
195	LI=LI-NDV	00035030
196	JS=JS-1	00035040
C*	CALC MATRIX L	00035050
198	IF(NF.LE.0) GO TO 210	00035060
	IF(ITYP.GT.7) GO TO 200	00035070
	IF(IQ.EQ.1) GO TO 201	00035080
200	IF(IX.EQ.1) GO TO 255	00035090
	READ (24) ((PLM(J,I),J=1,NVY),I=1,NF)	00035100
	GO TO 210	00035110
201	DFM(IX)=DL(1)	00035120
	IF(IX.NE.1) GO TO 204	00035130
	DLP(1)=(DL(1)+DL(NVY))/2.	00035140
	DO 202 J=2,NVY	00035150
202	DLP(J)=(DL(J-1)+DL(J))/2.	00035160
	DO 203 J=1,NVY	00035170
	DO 203 I=1,NF	00035180
203	PLM(J,I)=0.0	00035190
	GO TO 255	00035200
204	DLX=(DL(1)+DL(NVY))/4.	00035210
	SL(1)=DLP(1)+DLX	00035220
	DLP(1)=DLX	00035230
	DO 205 J=2,NVY	00035240
	DLX=(DL(J-1)+DL(J))/4.	00035250

	SL(J)=DLP(J)+DLX	00035260
205	DLP(J)=DLX	00035270
	IF(IX.NE.NVXM) GO TO 207	00035280
	DO 206 J=1,NVY	00035290
206	SL(J)=SL(J)+DLP(J)	00035300
207	DO 208 J=1,NVY	00035310
	DO 208 I=1,NF	00035320
208	PLM(J,I)=PM(IX,I)*SL(J)+PLM(J,I)	00035330
	WRITE (24) ((PLM(J,I),J=1,NVY),I=1,NF)	00035340
C*	SUM FOR AX,AY AND AZ	00035350
210	TX=QBXP(1)-QBX(1)+QTX(1)-QTX(NVY)	00035360
	TY=QBYP(1)-QBY(1)+QTY(1)-QTY(NVY)	00035370
	TZ=QBZP(1)-QBZ(1)+QTZ(1)-QTZ(NVY)	00035380
	IF(NF) 212,212,214	00035390
212	AX(IX)=TX	00035400
	AY(IX)=TY	00035410
	AZ(IX)=TZ	00035420
	GO TO 230	00035430
214	DO 220 I=1,NF	00035440
	AX(I)=TX*PLM(1,I)+AX(I)	00035450
	AY(I)=TY*PLM(1,I)+AY(I)	00035460
220	AZ(I)=TZ*PLM(1,I)+AZ(I)	00035470
	IC=NF	00035480
230	DO 250 J=2,NVY	00035490
	TX=QBXP(J)-QBX(J)+QTX(J)-QTX(J-1)	00035500
	TY=QBYP(J)-QBY(J)+QTY(J)-QTY(J-1)	00035510
	TZ=QBZP(J)-QBZ(J)+QTZ(J)-QTZ(J-1)	00035520
	IF(NF) 232,232,234	00035530
232	IC=(J-1)*NVXM+IX	00035540
	AX(IC)=TX	00035550
	AY(IC)=TY	00035560
	AZ(IC)=TZ	00035570
	GO TO 250	00035580
234	DO 240 I=1,NF	00035590
	IC=IC+1	00035600
	AX(IC)=TX*PLM(J,I)+AX(IC)	00035610
	AY(IC)=TY*PLM(J,I)+AY(IC)	00035620
240	AZ(IC)=TZ*PLM(J,I)+AZ(IC)	00035630
250	CONTINUE	00035640
255	DO 260 I=1,NVY	00035650
	QBXP(I)=QBX(I)	00035660
	QBYP(I)=QBY(I)	00035670
260	QBZP(I)=QBZ(I)	00035680

```

GO TO 300 00035690
270 IPAS=1 00035700
    IF(IX.EQ.1) GO TO 300 00035710
    READ (24) 00035720
300 CONTINUE 00035730
    REWIND 24 00035740
C 00035750
    IF(ITYP.LT.17) GO TO 320 00035760
    DO 310 I=1,NKF 00035770
    AYT=AY(I) 00035780
    AY(I)=AYT*SINDP+AZ(I)*COSDP 00035790
310 AZ(I)=-AYT*COSDP+AZ(I)*SINDP 00035800
C 00035810
320 WRITE (20) (AX(IC),AY(IC),AZ(IC),IC=1,NKF) 00035820
400 CONTINUE 00035830
    WRITE (6,450) ITYP 00035840
450 FORMAT (21H0 ** COMPLETED MATRIX,I4) 00035850
    RETURN 00035860
    END 00035870
*DECK AQXVW 00035880
    SUBROUTINE AQXVW(NQ,ITYP) 00035890
C 00035900
C*          CALC MATRIX-A (Q ON X-COMPONENT AND VORTICES ON WING) 00035910
C 00035920
    COMMON/DAT/ DA(5000) 00035930
C 00035940
    EQUIVALENCE (DA(1281),WNW) ,(DA(1270),WNVC) 00035950
1,(DA(1202),WOZ) ,(DA(3382),XLW) 00035960
C 00035970
    COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP 00035980
    COMMON/CFG/ IDB,IDF,IDD(10),NKW 00035990
    COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3 00036000
1          ,NE5,NTB,NU,NW,NVC,NVS,NJC,NJS 00036010
C 00036020
    LARGE      DM1(51700) 00036030
1          ,EV(52) ,XLV(52) ,XTLV(52) ,DM2(8840) 00036040
2          ,YB(600) ,YBI(600) ,YBR(600) ,DM3(1800) 00036050
3          ,CFI(50) ,CFJ(50) ,STH(50) ,ACT(50) 00036060
4          ,ET(52) ,XLT(52) ,XTLT(52) 00036070
5          ,C(40,40) ,S(51,10) ,DM4(400) ,TSI(52,40),DM5(2080) 00036080
6          ,COSTT(52) ,DZLE(52) ,COSTV(52) ,SINTV(52) 00036090
7          ,XB(600) ,XBI(600) ,XBR(600) 00036100
8          ,ZB(600) ,ZBI(600) ,ZBR(600) ,DM6(9761) 00036110

```

9	•XQ(650)	•YQ(650)	•ZQ(650)	00036120
*	•AX(250)	•AY(250)	•AZ(250)	00036130
*	•SEU(30)	•SEV(30)	•SEW(30)	00036140
C				00036150
C*	INITIAL AND CALC MATRIX A			00036160
	XLW2=4.*XLW*XLW			00036170
	NE1=NE2			00036180
	NVSH=NE2+1			00036190
	NE4=NE3+7			00036200
	NE6=NE3+3			00036210
	INDT=DA(2000)			00036220
	IF(INDT.NE.0) NE1=NVSH			00036230
	DTH=PI/WNVC			00036240
	DO 300 IQ=1,NQ			00036250
	DO 12 I=1,NKW			00036260
	AX(I)=0.0			00036270
	AY(I)=0.0			00036280
12	AZ(I)=0.0			00036290
	IF(IDF.EQ.0) GO TO 14			00036300
	CALL AQXVWR(IQ,NT0,NE3,XB,YB,ZB)			00036310
	CALL AQXVWR(IQ,NTI,NE6,XBI,YBI,ZBI)			00036320
14	IF(IDB.NE.0) CALL AQXVWR(IQ,NTB,NE5,XBR,YBR,ZBR)			00036330
C*	CHORDWISE LOOP			00036340
	TUV=0.			00036350
	TVV=0.			00036360
	TWV=0.			00036370
	DO 250 I=1,NVC			00036380
	IR=4*I-3			00036390
	IF(IDF.EQ.0) IR=IR+1200			00036400
	DO 15 IW=1,NW			00036410
	SEU(IW)=0.0			00036420
	SEV(IW)=0.0			00036430
15	SEW(IW)=0.0			00036440
C*	SPANWISE LOOP			00036450
	XV3=XLV(1)+XTLV(1)*CFI(1)*COSTT(1)			00036460
	XV2=XLV(1)+XTLV(1)*CFI(1)*COSTV(1)			00036470
	XVT=XLV(NVSH)+XTLV(NVSH)*CFI(1)*COSTV(NVSH)			00036480
	DO 200 IE=1,NE1			00036490
	IF(IE.NE.1) GO TO 22			00036500
	TL=2./DLO			00036510
	TPT=(XVT-XV3)*TL/HWS			00036520
	GO TO 24			00036530
22	IF(IE.NE.(NE3+4)) GO TO 26			00036540

	TL=2./DLI	00036550
24	DL4=2.*TL/BETA	00036560
	TDY=HWS/TL	00036570
	SC=TDY	00036580
	DTX=DTH/SC	00036590
26	KT=IE	00036600
	WZV=WOZ-XTLV(IE)*CFI(I)*SINTV(IE)+DZLE(IE)	00036610
	ZVW=(ZQ(IQ)-WZV*BETA)/SC	00036620
	IF(INDT.NE.0) GO TO 40	00036630
	IF(IE.GT.NE5) GO TO 200	00036640
	IF(IE.LE.NE3) GO TO 48	00036650
	IF(IE.GE.NE4) GO TO 48	00036660
	GO TO 200	00036670
40	IF(IE.NE.NE1) GO TO 42	00036680
	TPI=TPT	00036690
	TPO=-TPT	00036700
	TL=2./DLO	00036710
	DL4=2.*TL/BETA	00036720
	TDY=HWS/TL	00036730
	SC=TDY	00036740
	DTX=DTH/SC	00036750
	YV=HWS	00036760
	XVW=(XQ(IQ)-XVT)/SC	00036770
	GO TO 60	00036780
42	IF(IE.GT.NE5) GO TO 43	00036790
	IF(IE.LE.NE3) GO TO 48	00036800
	IF(IE.GE.NE4) GO TO 48	00036810
43	TPO=(XB(IR)-XB(IR+1))/(YB(IR)-YB(IR+1))	00036820
	TPI=TPO	00036830
46	YV=.5*(YB(IR)+YB(IR+1))	00036840
	XVW=(XQ(IQ)-(XB(IR)+XB(IR+1))/2.)/SC	00036850
	ZVW=(ZQ(IQ)-(ZB(IR)+ZB(IR+1))/2.)/SC	00036860
	IR=IR+1	00036870
	IF(IE.EQ.NE6) IR=IR+597	00036880
	IF(IE.EQ.(NE3+6)) IR=IR+597	00036890
	GO TO 60	00036900
48	YV=HWS*EV(IE)	00036910
	IF(IE.NE.1) GO TO 50	00036920
	TPI=TPT	00036930
	TPO=TPT	00036940
	XVW=4.*(XQ(IQ)-XV2)/SC	00036950
	ZVW=4.*ZVW	00036960
	GO TO 60	00036970

50	IF(IE-NE4) 58,52,54	00036980
52	XV1=XLT(IE)+XTLT(IE)*CFI(I)*COSTT(IE)	00036990
54	KT=KT+1	00037000
58	XV2=XLV(IE)+XTLV(IE)*CFI(I)*COSTV(IE)	00037010
	XV3=XLT(KT)+XTLT(KT)*CFI(I)*COSTT(KT)	00037020
	XVW=(XQ(IQ)-XV2)/SC	00037030
	TPI=(XV2-XV3)/TDY	00037040
	TPO=(XV1-XV2)/TDY	00037050
	IF(IE.EQ.NE2) GO TO 60	00037060
	IF(ET(KT).LT.0.0) TPI=TPO	00037070
60	ZV2=ZVW**2	00037080
	XZ2=XVW**2+ZV2	00037090
	ZL4=ZVW*DL4	00037100
	IPASS=1	00037110
70	YVW=(YQ(IQ)-YV)/SC	00037120
	EVI=0.	00037130
	EVO=0.	00037140
	IF((YVW**2+ZVW**2).GT.XLW2) GO TO 120	00037150
	TI2=TPI**2+1.	00037160
	TO2=TPO**2+1.	00037170
	XTI=XVW+TPI	00037180
	XTO=XVW-TPO	00037190
	IF(IE.EQ.1) YVW=4.*YVW	00037200
	YCP=YVW+1.	00037210
	YP2=YCP**2	00037220
	YCM=YVW-1.	00037230
	YM2=YCM**2	00037240
	SQ0=SQRT(XTO**2+YM2+ZV2)	00037250
	SQ1=SQRT(XTI**2+YP2+ZV2)	00037260
	SQR=SQRT(XZ2+YVW**2)	00037270
	XYTI=XVW-YVW*TPI	00037280
	XYTO=XVW-YVW*TPO	00037290
	R12=(XTI/SQ1+1.)/(YP2+ZV2)	00037300
	R22=XYTI**2+ZV2*TI2	00037310
	R32=XYTO**2+ZV2*TO2	00037320
	R42=(XTO/SQ0+1.)/(YM2+ZV2)	00037330
	EX=0.0	00037340
	EY=R12-R42	00037350
	EZ=YCM*R42-YCP*R12	00037360
	IF(R22/TI2.LE.1.E-10) GO TO 80	00037370
	YXTI=YVW+XVW*TPI	00037380
	EX=((YXTI+TI2)/SQ1-YXTI/SQR)/R22	00037390
	EY=EY-EX*TPI	00037400

	EVI=EX*XYTI	00037410
	EZ=EZ-EVI	00037420
80	IF(R32/T02.LE.1.E-10) GO TO 90	00037430
	YXTO=YVW+XVW*TPO	00037440
	EXX=(YXTO/SQR+(T02-YXTO)/SQ0)/R32	00037450
	EX=EX+EXX	00037460
	EY=EY-EXX*TPO	00037470
	EVO=-EXX*XYTO	00037480
	EZ=EZ+EVO	00037490
90	IF(INDT.EQ.0) GO TO 99	00037500
	IF(IE.EQ.1) GO TO 98	00037510
	STS=1./SQR	00037520
	EUI=1./SQI-STS	00037530
	EUO=1./SQO-STS	00037540
	EWI=0.	00037550
	IF(R22/TI2.LE.1.E-10) GO TO 92	00037560
	PII=SQRT(TI2)	00037570
	EWI=(YXTI/SQR-(YXTI+TI2)/SQI)*ZVW*PII/R22	00037580
92	EWO=0.	00037590
	IF(R32/T02.LE.1.E-10) GO TO 94	00037600
	POI=SQRT(T02)	00037610
	EWO=(YXTO/SQR-(YXTO-T02)/SQ0)*ZVW*POI/R32	00037620
94	ST=TSI(IE.I)*STH(I)*DTX	00037630
	IF(IE.NE.NE1) GO TO 96	00037640
	IF(IPASS.EQ.0.0) GO TO 97	00037650
96	C4=ST/TI2	00037660
	TUV=TUV-(EUI*TPI-EVI)*C4	00037670
	TVV=TVV-(EVI*TPI+EUI)*C4	00037680
	TWV=TWV-EWI*PII*C4	00037690
	IF(IE.EQ.NE1) GO TO 130	00037700
97	C1=ST/T02	00037710
	TUV=TUV+(EUO*TPO-EVO)*C1	00037720
	TVV=TVV+(EVO*TPO+EUO)*C1	00037730
	TWV=TWV+EWO*POI*C1	00037740
	IF(IE.EQ.NE1) GO TO 200	00037750
98	IF(IE.GT.NE5) GO TO 120	00037760
	IF(IE.GE.NE4) GO TO 99	00037770
	IF(IE.GT.NE3) GO TO 120	00037780
99	EX=EX*ZL4	00037790
	EY=EY*ZL4	00037800
	EZ=EZ*DL4	00037810
	IF(WNW.NE.0.0) GO TO 100	00037820
	SEU(IE)=EX+SEU(IE)	00037830

	SEV(IE)=EY+SEV(IE)	00037840
	SEW(IE)=EZ+SEW(IE)	00037850
	GO TO 120	00037860
100	DO 110 IW=1,NW	00037870
	SEU(IW)=SEU(IW)+EX*S(IE,IW)	00037880
	SEV(IW)=SEV(IW)+EY*S(IE,IW)	00037890
110	SEW(IW)=SEW(IW)+EZ*S(IE,IW)	00037900
120	IF(IPASS.EQ.0) GO TO 190	00037910
	IF(IDB.NE.0) GO TO 130	00037920
	IF(IE.EQ.NE2) GO TO 200	00037930
130	IPASS=0	00037940
	TPX=-TPI	00037950
	TPI=-TPO	00037960
	TPO=TPX	00037970
	YV=-YV	00037980
	GO TO 70	00037990
190	XV1=XV3	00038000
200	CONTINUE	00038010
	IF(WNW.NE.0.0) GO TO 204	00038020
	SEU(1)=4.*SEU(1)	00038030
	SEV(1)=4.*SEV(1)	00038040
	SEW(1)=4.*SEW(1)	00038050
204	MY=0	00038060
	DO 210 IU=1,NU	00038070
	DO 210 IW=1,NW	00038080
	MY=MY+1	00038090
	AX(MY)=AX(MY)+SEU(IW)*C(I,IU)	00038100
	AY(MY)=AY(MY)+SEV(IW)*C(I,IU)	00038110
210	AZ(MY)=AZ(MY)+SEW(IW)*C(I,IU)	00038120
250	CONTINUE	00038130
	TUV=TUV/PI4/BETA	00038140
	TVV=TVV/PI4	00038150
	TWV=TWV/PI4	00038160
		00038170
	IF(ITYP.LT.18) GO TO 290	00038180
	TVT=TVV	00038190
	TVV=TVT*SINDP+TWV*COSDP	00038200
	TWV=-TVT*COSDP+TWV*SINDP	00038210
	DO 260 I=1,NKW	00038220
	AYT=AY(I)	00038230
	AY(I)=AYT*SINDP+AZ(I)*COSDP	00038240
260	AZ(I)=-AYT*COSDP+AZ(I)*SINDP	00038250
		00038260

290	WRITE (21) (AX(IU),AY(IU),AZ(IU),IU=1,NKW),TUV,TVV,TWV	00038270
300	CONTINUE	00038280
	WRITE (6,450) ITYP	00038290
450	FORMAT (21H0 ** COMPLETED MATRIX,I4)	00038300
	RETURN	00038310
	END	00038320
*DECK	AQXVWR	00038330
	SUBROUTINE AQXVWR(IQ,NVT,NEVX,XB,YB,ZB)	00038340
C		00038350
C*	CALC MATRIX-A (Q ON X-COMPONENT AND VORTICES ON ROOT)	00038360
C		00038370
	COMMON/DAT/ DA(5000)	00038380
C		00038390
	LARGE XB(1),YB(1),ZB(1)	00038400
C		00038410
	EQUIVALENCE (DA(1281),WNW) ,(DA(3382),XLW)	00038420
C		00038430
	COMMON/CPW/ IDD(9),NU,NW,NVC	00038440
C		00038450
	LARGE DM1(51700),DM2(14552)	00038460
1	,S(51,10) ,CSEC(40,10),DM3(17729)	00038470
2	,XQ(650) ,YQ(650) ,ZQ(650)	00038480
3	,AX(250) ,AY(250) ,AZ(250)	00038490
4	,QBX(600) ,QBY(600) ,QBZ(600)	00038500
5	,QTX(600) ,QTY(600) ,QTZ(600)	00038510
C		00038520
C	INITIAL	00038530
	NVCT=NVC+NVT-1	00038540
	NPAS=0	00038550
	IF(ABS((-YB(4)-YQ(IQ))/(YB(1)-YB(2))),GT,XLW) NPAS=1	00038560
C	NVX AND NVY LOOPS	00038570
	LI=1	00038580
	LF=5	00038590
	JS=0	00038600
	DO 300 IX=1,NVCT	00038610
	K=0	00038620
	IF(IX.GT.NVC) JS=JS-4	00038630
	DO 200 IS=1,4	00038640
	JS=JS+1	00038650
20	IPASS=1	00038660
	IF(IX.GT.NVC) IPASS=3	00038670
30	TY1=YB(LF)-YB(LI)	00038680
	TY2=YB(LF)-YQ(IQ)	00038690

	TY3=YB(LI)-YQ(IQ)	00038700
	GO TO 50	00038710
40	IPASS=2	00038720
	IF(NPAS.EQ.0) GO TO 44	00038730
	IF(K.EQ.0) GO TO 160	00038740
	GO TO 70	00038750
44	TY1=YB(LI)-YB(LF)	00038760
	TY2=-YB(LF)-YQ(IQ)	00038770
	TY3=-YB(LI)-YQ(IQ)	00038780
50	TX1=XB(LF)-XB(LI)	00038790
	TX2=XB(LF)-XQ(IQ)	00038800
	TX3=XB(LI)-XQ(IQ)	00038810
	TZ1=ZB(LF)-ZB(LI)	00038820
	TZ2=ZB(LF)-ZQ(IQ)	00038830
	TZ3=ZB(LI)-ZQ(IQ)	00038840
	HN=TX2*TX1+TY2*TY1+TZ2*TZ1	00038850
	HD=TX1**2+TY1**2+TZ1**2	00038860
	SQHD=SQRT(HD)	00038870
	IF(SQHD.GT.0.00001) GO TO 52	00038880
	QSR=0.0	00038890
	GO TO 54	00038900
52	HT=HN/HD	00038910
	HX=TX2-HT*TX1	00038920
	HY=TY2-HT*TY1	00038930
	HZ=TZ2-HT*TZ1	00038940
	SHP=SQRT(HX**2+HY**2+HZ**2)	00038950
	SRF=HN/SQRT(TX2**2+TY2**2+TZ2**2)	00038960
	HI=TX1*TX3+TY1*TY3+TZ1*TZ3	00038970
	SRI=HI/SQRT(TX3**2+TY3**2+TZ3**2)	00038980
	QT=(SRF-SRI)/SQHD/SHP	00038990
	TX=TY2*TZ1-TZ2*TY1	00039000
	TY=TZ2*TX1-TX2*TZ1	00039010
	TZ=TX2*TY1-TY2*TX1	00039020
	QSR=QT/SQRT(TX**2+TY**2+TZ**2)	00039030
54	IF(K.EQ.0) GO TO 140	00039040
C*	LATERAL VORTICES	00039050
	IF(IPASS.EQ.2) GO TO 60	00039060
	QBX(JS)=QSR*TX	00039070
	QBY(JS)=QSR*TY	00039080
	QBZ(JS)=QSR*TZ	00039090
	LF=LI	00039100
	LI=LF+1	00039110
	GO TO 40	00039120

60	QBX(JS)=QSR*TX+QBX(JS)	00039130
	QBY(JS)=QSR*TY+QBY(JS)	00039140
	QBZ(JS)=QSR*TZ+QBZ(JS)	00039150
70	LF=LI+4	00039160
	K=0	00039170
	GO TO 20	00039180
C*	LONGITUDINAL VORTICES	00039190
140	IF(IPASS.GE.2) GO TO 150	00039200
	QTX(JS)=QSR*TX	00039210
	QTY(JS)=QSR*TY	00039220
	QTZ(JS)=QSR*TZ	00039230
145	LF=LI	00039240
	LI=LF+4	00039250
	GO TO 40	00039260
150	QTX(JS)=QSR*TX+QTX(JS)	00039270
	QTY(JS)=QSR*TY+QTY(JS)	00039280
	QTZ(JS)=QSR*TZ+QTZ(JS)	00039290
	IF(IPASS.EQ.3) GO TO 145	00039300
160	IF(IX.LE.NVC) GO TO 190	00039310
	LI=LF+1	00039320
	LF=LI+4	00039330
	GO TO 200	00039340
190	LI=LF	00039350
	LF=LI+1	00039360
	K=1	00039370
200	CONTINUE	00039380
	IF(IX.GT.NVC) GO TO 300	00039390
	LI=LI+1	00039400
	LF=LI+4	00039410
300	CONTINUE	00039420
C		0003
C*	CALC CONTRIBUTION FROM TRAIL VORTICES	00039430
	JS=JS-4	00039440
	DO 320 IS=1,4	00039450
	JS=JS+1	00039460
	PASS=1.	00039470
	HY=YB(LI)-YQ(IQ)	00039480
	HX=XB(LI)-XQ(IQ)	00039490
	HZ=ZB(LI)-ZQ(IQ)	00039500
310	DD1=HY*HY+HZ*HZ	00039510
	IF(DD1.LT.0.00001) GO TO 315	00039520
	DD2=SQRT(HX*HX+DD1)	00039530
	IF(DD2.LT.0.00001) GO TO 315	00039540

	QSR=(1.-HX/DD2)/DD1*PASS	00039550
	QTY(JS)=QTY(JS)+QSR*HZ	00039560
	QTZ(JS)=QTZ(JS)-QSR*HY	00039570
315	IF(NPAS.EQ.1) GO TO 320	00039580
	IF(PASS.LT.0.) GO TO 320	00039590
	PASS=-1.	00039600
	HY=-YB(LI)-YQ(IQ)	00039610
	GO TO 310	00039620
320	LI=LI+1	00039630
C		00039640
C*	SUM FOR AX,AY AND AZ	00039650
	N=4*NVC+1	00039660
	DO 340 I=1,3	00039670
	N=N+1	00039680
	QBX(N)=0.0	00039690
	QBY(N)=0.0	00039700
340	QBZ(N)=0.0	00039710
	N=0	00039720
	DO 360 J=1,NVC	00039730
	N=N+1	00039740
	DO 360 I=1,3	00039750
	N=N+1	00039760
	EX=QTX(N-1)-QTX(N)+QBX(N+4)-QBX(N)	00039770
	EY=QTY(N-1)-QTY(N)+QBY(N+4)-QBY(N)	00039780
	EZ=QTZ(N-1)-QTZ(N)+QBZ(N+4)-QBZ(N)	00039790
	IE=NEVX+I	00039800
	IF(WNW.NE.0.0) GO TO 348	00039810
	K=IE	00039820
	DO 342 IU=1,NU	00039830
	AX(K)=EX*CSEC(J,IU)+AX(K)	00039840
	AY(K)=EY*CSEC(J,IU)+AY(K)	00039850
	AZ(K)=EZ*CSEC(J,IU)+AZ(K)	00039860
342	K=K+NW	00039870
	GO TO 360	00039880
348	K=0	00039890
	DO 350 IU=1,NU	00039900
	DO 350 IW=1,NW	00039910
	K=K+1	00039920
	SCS=S(IE,IW)*CSEC(J,IU)	00039930
	AX(K)=EX*SCS+AX(K)	00039940
	AY(K)=EY*SCS+AY(K)	00039950
350	AZ(K)=EZ*SCS+AZ(K)	00039960
360	CONTINUE	00039970

```

RETURN                                00039980
END                                    00039990
*DECK AQWVW                            00040000
SUBROUTINE AQWVW (ITYP)                00040010
C                                        00040020
C*          CALC MATRIX-A (Q ON WING AND VORTICES ON WING) 00040030
C                                        00040040
COMMON/DAT/ DA(5000)                   00040050
C                                        00040060
EQUIVALENCE      (DA(1270),WNVC) ,(DA(1281),WNW) ,(DA(3382),XLW) 00040070
C                                        00040080
COMMON/CRG/ PI,PI4,RC,BETA             00040090
COMMON/CFG/ IDB,IDF,IDD(10),NKW        00040100
COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3 00040110
1      ,NE5,NTB,NU,NW,NVC,NVS,NJC,NJS   00040120
LARGE      DMI(51700)                   00040130
1      ,EV(52)      ,XLV(52)      ,XTLV(52)      ,DM2(3440) 00040140
2      ,UV(30,30)  ,UVP(30,30) ,VV(30,30) ,VVP(30,30) 00040150
3      ,XB(600)   ,XBI(600)   ,XBR(600)   00040160
4      ,YB(600)   ,YBI(600)   ,YBR(600)   00040170
5      ,ZB(600)   ,ZBI(600)   ,ZBR(600)   00040180
6      ,CFI(50)   ,CFJ(50)   ,STH(50)   ,ACT(50)   00040190
7      ,ET(52)   ,XLT(52)   ,XTLT(52)   00040200
8      ,C(40,40) ,S(51,10) ,CSEC(40,10) 00040210
9      ,TSI(52,40),TZI(52,40),DM3(13569) 00040220
*      ,XQ(650)  ,YQ(650)  ,ZQ(650)  00040230
*      ,AX(250)  ,AY(250)  ,AZ(250)  ,SUMES(30) 00040240
C                                        00040250
C                                        00040260
NE1=NE2+1                               00040270
NVSH=NE1                                00040280
NE4=NE3+7                               00040290
NE6=NE3+3                               00040300
INDT=DA(2000)                            00040310
IF(INDT.EQ.0) NE1=NE2                    00040320
DTH=PI/WNVC*.5                           00040330
C*          Q ON WING LOOP                00040340
IC=0                                       00040350
DO 350 KP=1,NJS                           00040360
DO 340 J=1,NJC                             00040370
PUV=0.0                                    00040380
PVV=0.0                                    00040390
TUV=0.0                                    00040400

```

	TVV=0.	00040410
	DO 12 I=1,NKW	00040420
	AX(I)=0.0	00040430
	AY(I)=0.0	00040440
12	AZ(I)=0.0	00040450
	IC=IC+1	00040460
	XP=XQ(IC)	00040470
	EVP=YQ(IC)/HWS	00040480
	IF(IDF.EQ.0.AND.IDB.EQ.0) GO TO 40	00040490
	IF(IDF.EQ.0) GO TO 14	00040500
	CALL AQXVWR(IC,NT0,NE3,XB,YB,ZB)	00040510
	CALL AQXVWR(IC,NTI,NE6,XBI,YBI,ZBI)	00040520
14	IF(IDB.NE.0) CALL AQXVWR(IC,NTB,NE5,XBR,YBR,ZBR)	00040530
	DO 32 I=1,NKW	00040540
32	AZ(I)=-AZ(I)	00040550
C	I LOOP	00040560
40	DO 330 I=1,NVC	00040570
	IR=4*I-3	00040580
	IF(IDF.EQ.0) IR=IR+1200	00040590
	DO 100 NS=1,NW	00040600
100	SUMES(NS)=0.0	00040610
C	ETAV LOOP	00040620
	XV3=XLT(1)+XTLT(1)*CFI(I)	00040630
	XV2=XLV(1)+XTLV(1)*CFI(I)	00040640
	XVT=XLV(NVSH)+XTLV(NVSH)*CFI(I)	00040650
	DO 310 KV=1,NE1	00040660
	IF(KV.NE.1) GO TO 102	00040670
	TL=2./DLO	00040680
	TPT=(XVT-XV3)*TL/HWS	00040690
	GO TO 104	00040700
102	IF(KV.NE.(NE3+4)) GO TO 106	00040710
	TL=2./DLI	00040720
104	DL4=2.*TL/BETA	00040730
	X2=TL/HWS	00040740
	DTX=X2*DTH	00040750
	TDY=HWS/TL	00040760
106	KT=KV	00040770
	IPASS=1	00040780
	EVY=EV(KV)	00040790
	IF(INDT.NE.0) GO TO 110	00040800
	IF(KV.GT.NE5) GO TO 310	00040810
	IF(KV.LE.NE3) GO TO 140	00040820
	IF(KV.GE.NE4) GO TO 140	00040830

GO TO 310	00040840
110 IF(KV.NE.NE1) GO TO 120	00040850
TPI=TPT	00040860
TPO=-TPT	00040870
TL=2./DLO	00040880
DL4=2.*TL/BETA	00040890
X2=TL/HWS	00040900
DTX=X2*DTH	00040910
TDY=HWS/TL	00040920
XC=X2*(XP-XVT)	00040930
GO TO 190	00040940
120 IF(KV.GT.NE5) GO TO 121	00040950
IF(KV.LE.NE3) GO TO 140	00040960
IF(KV.GE.NE4) GO TO 140	00040970
121 TPO=(XB(IR)-XB(IR+1))/(YB(IR)-YB(IR+1))	00040980
130 EVY=(YB(IR)+YB(IR+1))/2./HWS	00040990
XC=X2*(XP-(XB(IR)+XB(IR+1))/2.)	00041000
TPI=TPO	00041010
IR=IR+1	00041020
IF(KV.EQ.NE6) IR=IR+597	00041030
IF(KV.EQ.(NE3+6)) IR=IR+597	00041040
GO TO 190	00041050
140 IF(KV.NE.1) GO TO 150	00041060
TPI=TPT	00041070
TPO=TPT	00041080
XC=4.*X2*(XP-XV2)	00041090
GO TO 190	00041100
150 IF(KV.NE4) 180,160,170	00041110
160 XV1=XLT(KV)+XTLT(KV)*CFI(I)	00041120
170 KT=KT+1	00041130
180 XV2=XLV(KV)+XTLV(KV)*CFI(I)	00041140
XV3=XLT(KT)+XTLT(KT)*CFI(I)	00041150
TPI=(XV2-XV3)/TDY	00041160
TPO=(XV1-XV2)/TDY	00041170
IF(KV.EQ.NE2) GO TO 182	00041180
IF(ET(KT).LT.0.0) TPI=TPO	00041190
182 XC=X2*(XP-XV2)	00041200
190 YC=TL*(EVP-EVY)	00041210
EVI=0.	00041220
EVO=0.	00041230
IF(ABS(YC/2.).GT.XLW) GO TO 302	00041240
IF(KV.EQ.1) YC=4.*YC	00041250
YCP=YC+1.	00041260

	YCM=YC-1.	00041270
	XTO=XC-TPO	00041280
	XTI=XC+TPI	00041290
	SQO=SQRT(XTO**2+YCM**2)	00041300
	SQI=SQRT(XTI**2+YCP**2)	00041310
	EC=(XTI/SQI+1.)/YCP-(XTO/SQO+1.)/YCM	00041320
	SQC=SQRT(XC**2+YC**2)	00041330
	C1=TPI**2+1.	00041340
	C4=C1	00041350
	C2=XC-YC*TPI	00041360
	IF(C2**2/C1.LE.1.E-10) GO TO 230	00041370
	C3=YC+XC*TPI	00041380
	EVI=((C3+C1)/SQI-C3/SQC)/C2	00041390
230	C1=TPO**2+1.	00041400
	C2=XC-YC*TPO	00041410
	IF(C2**2/C1.LE.1.E-10) GO TO 250	00041420
	C3=YC+XC*TPO	00041430
240	EVO=((C3-C1)/SQO-C3/SQC)/C2	00041440
250	IF(INDT.EQ.0) GO TO 290	00041450
	IF(KV.EQ.1) GO TO 290	00041460
	STS=1./SQC	00041470
	EUI=1./SQI-STS	00041480
	EUO=1./SQO-STS	00041490
	AYC=ABS(YC/X2)	00041500
	ST=TSI(KV*I)*STH(I)*DTX	00041510
	SP=TZI(KV*I)/STH(I)*DTX*2.0	00041520
	TVI=ST*EVI	00041530
	TVO=ST*EVO	00041540
	KX=KV*IPASS	00041550
260	IF(KV.NE.NE1) GO TO 280	00041560
	IF(KX.NE.NE1) GO TO 270	00041570
262	PUV=PUV-SP*(EUI*TPI-EVI)/C4	00041580
	PVV=PVV-SP*(EVI*TPI+EUI)/C4	00041590
	EUI=ST*EUI	00041600
	TUV=TUV-(EUI*TPI-TVI)/C4	00041610
	TVV=TVV-(TVI*TPI+EUI)/C4	00041620
	GO TO 305	00041630
270	PUV=PUV+SP*(EUO*TPO-EVO)/C1	00041640
	PVV=PVV+SP*(EVO*TPO+EUO)/C1	00041650
	EUO=ST*EUO	00041660
	TUV=TUV+(EUO*TPO-TVO)/C1	00041670
	TVV=TVV+(TVO*TPO+EUO)/C1	00041680
	GO TO 310	00041690

280	PUV=PUV+SP*((EUO*TPO-EVO)/C1-(EUI*TPI-EVI)/C4)	00041700
	PVV=PVV+SP*((EVO*TPO+EUO)/C1-(EVI*TPI+EUI)/C4)	00041710
	EUI=ST*EUI	00041720
	EUO=ST*EUO	00041730
	TUV=TUV+(EUO*TPO-TV0)/C1-(EUI*TPI-TVI)/C4	00041740
	TVV=TVV+(TV0*TPO+EU0)/C1-(TVI*TPI+EUI)/C4	00041750
290	IF(KV.GT.NE5) GO TO 302	00041760
	IF(KV.GE.NE4) GO TO 291	00041770
	IF(KV.GT.NE3) GO TO 302	00041780
291	EC=(EC+EVI-EVO)*DL4	00041790
	IF(WNW.NE.0.0) GO TO 292	00041800
	SUMES(KV)=SUMES(KV)+EC	00041810
	GO TO 302	00041820
292	DO 300 NS=1,NW	00041830
300	SUMES(NS)=SUMES(NS)+EC*S(KV,NS)	00041840
302	IF(IPASS.EQ.0) GO TO 308	00041850
	IF(IDB.NE.0) GO TO 305	00041860
	IF(KV.EQ.NE2) GO TO 310	00041870
305	IPASS=0	00041880
	EVY=-EVY	00041890
	TPX=-TPI	00041900
	TPI=-TPO	00041910
	TPO=TPX	00041920
	GO TO 190	00041930
308	XV1=XV3	00041940
310	CONTINUE	00041950
	IF(WNW.EQ.0.0) SUMES(1)=4.*SUMES(1)	00041960
	MY=0	00041970
312	DO 314 NC=1,NU	00041980
	DO 314 NS=1,NW	00041990
	MY=MY+1	00042000
314	AZ(MY)=AZ(MY)+SUMES(NS)*C(I,NC)	00042010
330	CONTINUE	00042020
	WRITE (21) (AZ(MY),MY=1,NKW)	00042030
	UVP(KP,J)=PUV/PI4/BETA	00042040
	VVP(KP,J)=PVV/PI4	00042050
	UV(KP,J)=TUV/PI4/BETA	00042060
340	VV(KP,J)=TVV/PI4	00042070
350	CONTINUE	00042080
	IF(INDT.NE.0) CALL AQWVWT	00042090
		00042100
	WRITE (6,450) ITYP	00042110
450	FORMAT (21H0 ** COMPLETED MATRIX,14)	00042120

```

RETURN                                00042130
END                                    00042140
*DECK AQVWWT                          00042150
SUBROUTINE AQVWWT                      00042160
C                                       00042170
C*          CALC ADDITIONAL INFLUENCE OF WING THICKNESS ON 00042180
C          VELOCITIES DUE TO BREAKS,ETC. 00042190
C                                       00042200
COMMON/DAT/ DA(5000)                  00042210
C                                       00042220
EQUIVALENCE      (DA(1270),WNVC) ,(DA(3382),XLW) 00042230
C                                       00042240
COMMON/CRG/ PI,PI4,RC,BETA           00042250
COMMON/CFG/ IDB,IDF                   00042260
COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3 00042270
1      ,NE5,NTB,NU,NW,NVC,NVS,NJC,NJS 00042280
C                                       00042290
LARGE      DM1(51700)                  00042300
1      ,EV(52)      ,XLV(52)      ,XTLV(52)      ,DM2(3440) 00042310
2      ,UV(30,30)   ,UVP(30,30)   ,VV(30,30)   ,VVP(30,30) 00042320
3      ,XB(1800)   ,YB(1800)   ,ZB(1800) 00042330
4      ,CFI(50)   ,CFJ(50)   ,STH(50)   ,ACT(50) 00042340
5      ,ET(52)   ,XLT(52)   ,XTLT(52) 00042350
6      ,DM3(2510) ,TSI(52,40) ,TZI(52,40) 00042360
7      ,DM4(13569),XQ(650)   ,YQ(650)   ,ZQ(650) 00042370
C                                       00042380
C                                       00042390
NE1=NE2+1                             00042400
NVSH=NE1                              00042410
NE4=NE3+7                             00042420
NE6=NE3+3                             00042430
DTH=PI/WNVC*.5                        00042440
C*          Q ON WING LOOP              00042450
IC=0                                    00042460
DO 350 KP=1,NJS                        00042470
DO 340 J=1,NJC                          00042480
PUV=0.                                  00042490
PVV=0.                                  00042500
TUV=0.                                  00042510
TVV=0.                                  00042520
IC=IC+1                                 00042530
XP=XQ(IC)                               00042540
EVP=YQ(IC)/HWS                          00042550

```

C	I LOOP	00042560
40	DO 330 I=2,NVC	00042570
	STHJ=SIN(ACT(I))	00042580
	IR=4*I-7	00042590
	IF(IDF.EQ.0) IR=IR+1200	00042600
C	ETAV LOOP	00042610
	XV3=XLT(1)+XTLT(1)*CFJ(I)	00042620
	XV2=XLV(1)+XTLV(1)*CFJ(I)	00042630
	XVT=XLV(NVSH)+XTLV(NVSH)*CFJ(I)	00042640
	DL 310 KV=1,NE1	00042650
	IF(KV.NE.1) GO TO 102	00042660
	TL=2./DLO	00042670
	TPT=(XVT-XV3)*TL/HWS	00042680
	GO TO 104	00042690
102	IF(KV.NE.(NE3+4)) GO TO 106	00042700
	TL=2./DLI	00042710
104	DL4=2.*TL/BETA	00042720
	X2=TL/HWS	00042730
	DTX=X2*DTH	00042740
	TDY=HWS/TL	00042750
106	KT=KV	00042760
	IPASS=1	00042770
	EVY=EV(KV)	00042780
110	IF(KV.NE.NE1) GO TO 120	00042790
	TPI=TPT	00042800
	TPO=-TPT	00042810
	TL=2./DLO	00042820
	DL4=2.*TL/BETA	00042830
	X2=TL/HWS	00042840
	DTX=X2*DTH	00042850
	TDY=HWS/TL	00042860
	XC=X2*(XP-XVT)	00042870
	GO TO 190	00042880
120	IF(KV.GT.NE5) GO TO 130	00042890
	IF(KV.LE.NE3) GO TO 140	00042900
	IF(KV.GE.NE4) GO TO 140	00042910
130	XB1=(XB(IR)+XB(IR+4))/2.0	00042920
	XB2=(XB(IR+1)+XB(IR+5))/2.0	00042930
	YB1=(YB(IR)+YB(IR+4))/2.0	00042940
	YB2=(YB(IR+1)+YB(IR+5))/2.0	00042950
	TPO=(XB1-XB2)/(YB1-YB2)	00042960
	EVY=(YB1+YB2)/2.0/HWS	00042970
	XC=X2*(XP-(XB1+XB2)/2.0)	00042980

	TPI=TPO	00042990
	IR=IR+1	00043000
	IF(KV.EQ.NE6) IR=IR+597	00043010
	IF(KV.EQ.(NE3+6)) IR=IR+597	00043020
	GO TO 190	00043030
140	IF(KV.NE.1) GO TO 150	00043040
	TPI=TPT	00043050
	TPO=TPT	00043060
	XC=4.*X2*(XP-XV2)	00043070
	GO TO 190	00043080
150	IF(KV-NE4) 180,160,170	00043090
160	XV1=XLT(KV)+XTLT(KV)*CFJ(I)	00043100
170	KT=KT+1	00043110
180	XV2=XLV(KV)+XTLV(KV)*CFJ(I)	00043120
	XV3=XLT(KT)+XTLT(KT)*CFJ(I)	00043130
	TPI=(XV2-XV3)/TDY	00043140
	TPO=(XV1-XV2)/TDY	00043150
	IF(KV.EQ.NE2) GO TO 182	00043160
	IF(ET(KT).LT.0.0) TPI=TPO	00043170
182	XC=X2*(XP-XV2)	00043180
190	YC=TL*(EVP-EVY)	00043190
	EVI=0.	00043200
	EVO=0.	00043210
	IF(ABS(YC/2.).GT.XLW) GO TO 290	00043220
	IF(KV.EQ.1) YC=4.*YC	00043230
	YCP=YC+1.	00043240
	YCM=YC-1.	00043250
	XTO=XC-TPO	00043260
	XTI=XC+TPI	00043270
	SQO=SQRT(XTO**2+YCM**2)	00043280
	SQI=SQRT(XTI**2+YCP**2)	00043290
	SQC=SQRT(XC**2+YC**2)	00043300
	C1=TPI**2+1.	00043310
	C4=C1	00043320
	C2=XC-YC*TPI	00043330
	IF(C2**2/C1.LE.1.E-10) GO TO 230	00043340
	C3=YC+XC*TPI	00043350
	EVI=((C3+C1)/SQI-C3/SQC)/C2	00043360
230	C1=TPO**2+1.	00043370
	C2=XC-YC*TPO	00043380
	IF(C2**2/C1.LE.1.E-10) GO TO 250	00043390
	C3=YC+XC*TPO	00043400
240	EVO=((C3-C1)/SQO-C3/SQC)/C2	00043410

250	IF(KV.EQ.1) GO TO 290	00043420
	IF(SQC.GT.1.E-10) GO TO 254	00043430
	STS=0.0	00043440
	GO TO 256	00043450
254	STS=1./SQC	00043460
256	EUI=1./SQI-STS	00043470
	EUO=1./SQO-STS	00043480
	AYC=ABS(YC/X2)	00043490
	SLOPE=(TSI(KV,I)+TSI(KV,I-1))/2.0	00043500
	DEFL=(TZI(KV,I)+TZI(KV,I-1))/2.0	00043510
	ST=SLOPE*STHJ*DTX	00043520
	SP=DEFL/STHJ*DTX*2.0	00043530
	TVI=ST*EVI	00043540
	TVO=ST*EVO	00043550
	KX=KV*IPASS	00043560
260	IF(KV.NE.NE1) GO TO 280	00043570
	IF(KX.NE.NE1) GO TO 270	00043580
262	PUV=PUV-SP*(EUI*TPI-EVI)/C4	00043590
	PVV=PVV-SP*(EVI*TPI+EUI)/C4	00043600
	EUI=ST*EUI	00043610
	TUV=TUV-(EUI*TPI-TVI)/C4	00043620
	TVV=TVV-(TVI*TPI+EUI)/C4	00043630
	GO TO 305	00043640
270	PUV=PUV+SP*(EUO*TPO-EVO)/C1	00043650
	PVV=PVV+SP*(EVO*TPO+EUO)/C1	00043660
	EUO=ST*EUO	00043670
	TUV=TUV+(EUO*TPO-TVO)/C1	00043680
	TVV=TVV+(TVO*TPO+EUO)/C1	00043690
	GO TO 310	00043700
280	PUV=PUV+SP*((EUO*TPO-EVO)/C1-(EUI*TPI-EVI)/C4)	00043710
	PVV=PVV+SP*((EVO*TPO+EUO)/C1-(EVI*TPI+EUI)/C4)	00043720
	EUI=ST*EUI	00043730
	EUO=ST*EUO	00043740
	TUV=TUV+(EUO*TPO-TVO)/C1-(EUI*TPI-TVI)/C4	00043750
	TVV=TVV+(TVO*TPO+EUO)/C1-(TVI*TPI+EUI)/C4	00043760
290	IF(IPASS.EQ.0) GO TO 308	00043770
	IF(IDB.NE.0) GO TO 305	00043780
	IF(KV.EQ.NE2) GO TO 310	00043790
305	IPASS=0	00043800
	EVY=-EVY	00043810
	TPX=-TPI	00043820
	TPI=-TPO	00043830
	TPO=TPX	00043840

```

GO TO 190
308 XVI=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/PPP/ 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*          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

```

```

00043850
00043860
00043870
00043880
00043890
00043900
00043910
00043920
00043930
00043940
00043950
00043960
00043970
00043980
00043990
00044000
00044010
00044020
00044030
00044040
00044050
00044060
00044070
00044080
00044090
00044100
00044110
00044120
00044130
00044140
00044150
00044160
00044170
00044180
00044190
00044200
00044210
00044220
00044230
00044240
00044250
00044260
00044270

```


DL4=4./DL/BETA	00044280
TDY=DL*HPS/2.	00044290
SC=TDY	00044300
NE1=NEVP	00044310
NE2=NVSH-1	00044320
INDT=DA(3000)	00044330
IF(INDT.NE.0) NE1=NVSH	00044340
DTH=PI/PNVC	00044350
DTX=DTH/SC	00044360
DO 300 IQ=1,NQ	00044370
C* CHORDWISE LOOP	00044380
TVT=0.0	00044390
TWT=0.0	00044400
TUV=0.	00044410
YQS=YQ(IQ)	00044420
SDS=SINDP	00044430
IPASS=1	00044440
XQ(IQ)=XQ(IQ)-POX	00044450
YQT=YQS-PYB	00044460
ZQT=ZQ(IQ)-PZB	00044470
20 YQ(IQ)=YQT*SDS+ZQT*COSSDP	00044480
ZQ(IQ)=ZQT*SDS-YQT*COSSDP	00044490
CALL AQXVPR (IQ,IPASS,SDS)	00044500
TVV=0.	00044510
TWV=0.	00044520
IF(IPASS.EQ.0) GO TO 30	00044530
IF(ITYP.EQ.19) GO TO 251	00044540
30 IR=1	00044550
DO 250 I=1,NVC	00044560
DO 40 IW=1,NVS	00044570
SEU(IW)=0.0	00044580
SEV(IW)=0.0	00044590
40 SEW(IW)=0.0	00044600
C* SPANWISE LOOP	00044610
XV3=XLT(1)+XLT(1)*CFI(I)	00044620
XV2=XLV(1)+XLT(1)*CFI(I)	00044630
XVT=XLV(NVSH)+XLT(NVSH)*CFI(I)	00044640
TPT=(XVT-XV3)/TDY	00044650
DO 200 IE=1,NE1	00044660
ZVW=(ZQ(IQ)-ZCON)/SC	00044670
IF(INDT.EQ.0.0) GO TO 48	00044680
EVI=0.	00044690
EVO=0.	00044700

	IF(IE.NE.NE1) GO TO 42	00044710
	TPI=TPT	00044720
	TPO=-TPT	00044730
	YV=HPS	00044740
	XVW=(XQ(IQ)-XVT)/SC	00044750
	GO TO 60	00044760
42	IF(IE.LE.NEVP) GO TO 48	00044770
	TPO=(XB(IR)-XB(IR+1))/(YB(IR)-YB(IR+1))	00044780
44	TPI=TPO	00044790
	YV=.5*(YB(IR)+YB(IR+1))	00044800
	XVW=(XQ(IQ)-(XB(IR)+XB(IR+1))/2.)/SC	00044810
	ZVW=(ZQ(IQ)-(ZB(IR)+ZB(IR+1))/2.)/SC	00044820
	IR=IR+1	00044830
	IF(IE.EQ.NE2) IR=IR+1	00044840
	GO TO 60	00044850
48	YV=HPS*EV(IE)	00044860
	IF(IE.NE.1) GO TO 50	00044870
	TPI=TPT	00044880
	TPO=TPT	00044890
	XVW=4.*(XQ(IQ)-XV2)/SC	00044900
	ZVW=4.*ZVW	00044910
	GO TO 60	00044920
50	XV2=XLV(IE)+XTLV(IE)*CFI(I)	00044930
	XV3=XLT(IE)+XTLT(IE)*CFI(I)	00044940
	XVW=(XQ(IQ)-XV2)/SC	00044950
	TPI=(XV2-XV3)/TDY	00044960
	TPO=(XV1-XV2)/TDY	00044970
	IF(IE.EQ.NE2) GO TO 60	00044980
	IF(ET(IE).LT.0.0) TPI=TPO	00044990
60	IF(ITYP.EQ.24) ZVW=2.	00045000
	YVW=(YQ(IQ)-YV)/SC	00045010
	T12=TPI**2+1.	00045020
	T02=TPO**2+1.	00045030
	XTI=XVW+TPI	00045040
	XTO=XVW-TPO	00045050
	IF(IE.EQ.1) YVW=4.*YVW	00045060
	YCP=YVW+1.	00045070
	YP2=YCP**2	00045080
	YCM=YVW-1.	00045090
	YM2=YCM**2	00045100
70	ZV2=ZVW**2	00045110
	IF((YVW**2+ZV2).GT.XLP2) GO TO 190	00045120
	XZ2=XVW**2+ZV2	00045130

ZL4=ZVW*DL4	00045140
SQO=SQRT(XTO**2+YM2+ZV2)	00045150
SQI=SQRT(XTI**2+YP2+ZV2)	00045160
SQR=SQRT(XZ2+YVW**2)	00045170
XYTI=XVW-YVW*TPI	00045180
XYTO=XVW-YVW*TPO	00045190
R12=(XTI/SQI+1.)/(YP2+ZV2)	00045200
R22=XYTI**2+ZV2*TI2	00045210
R32=XYTO**2+ZV2*TO2	00045220
R42=(XTO/SQO+1.)/(YM2+ZV2)	00045230
EX=0.0	00045240
EY=R12-R42	00045250
EZ=YCM*R42-YCP*R12	00045260
IF(R22/TI2.LE.1.E-10) GO TO 80	00045270
YXTI=YVW+XVW*TPI	00045280
EX=((YXTI+TI2)/SQI-YXTI/SQR)/R22	00045290
EY=EY-EX*TPI	00045300
EVI=EX*XYTI	00045310
EZ=EZ-EVI	00045320
80 IF(R32/TO2.LE.1.E-10) GO TO 90	00045330
YXTO=YVW+XVW*TPO	00045340
EXX=(YXTO/SQR+(TO2-YXTO)/SQO)/R32	00045350
EX=EX+EXX	00045360
EY=EY-EXX*TPO	00045370
EVO=-EXX*XYTO	00045380
EZ=EZ+EVO	00045390
90 IF(INDT.EQ.0) GO TO 98	00045400
IF(IE.EQ.1) GO TO 98	00045410
STS=1./SQR	00045420
EUI=1./SQI-STS	00045430
EUO=1./SQO-STS	00045440
EWI=0.	00045450
IF(R22/TI2.LE.1.E-10) GO TO 92	00045460
PII=SQRT(TI2)	00045470
EWI=(YXTI/SQR-(YXTI+TI2)/SQI)*ZVW*PII/R22	00045480
92 EWO=0.	00045490
IF(R32/TO2.LE.1.E-10) GO TO 94	00045500
POI=SQRT(TO2)	00045510
EWO=(YXTO/SQR-(YXTO-TO2)/SQO)*ZVW*POI/R32	00045520
94 ST=TSI(IE,I)*STH(I)*DTX	00045530
IF(IE.NE.NE1) GO TO 96	00045540
IF(IPASS.EQ.0.0) GO TO 97	00045550
96 C4=ST/TI2	00045560

	TUV=TUV-(EUI*TPI-EVI)*C4	00045570
	TVV=TVV-(EVI*TPI+EUI)*C4	00045580
	TWV=TWV-EWI*PII*C4	00045590
	IF(IE.EQ.NE1) GO TO 190	00045600
97	C1=ST/TO2	00045610
	TUV=TUV+(EU0*TPO-EV0)*C1	00045620
	TVV=TVV+(EVO*TPO+EU0)*C1	00045630
	TWV=TWV+EWO*POI*C1	00045640
	IF(IE.EQ.NE1) GO TO 200	00045650
98	IF(IE.GT.NEVP) GO TO 190	00045660
	IF(ITYP.NE.24) GO TO 100	00045670
	IF(IPASS.EQ.1) GO TO 190	00045680
100	EX=EX*ZL4	00045690
	EY=EY*ZL4	00045700
	EZ=EZ*DL4	00045710
	SEU(IE)=EX+SEU(IE)	00045720
	SEV(IE)=EY+SEV(IE)	00045730
	SEW(IE)=EZ+SEW(IE)	00045740
190	XV1=XV3	00045750
200	CONTINUE	00045760
	SEU(1)=4.*SEU(1)	00045770
	SEV(1)=4.*SEV(1)	00045780
	SEW(1)=4.*SEW(1)	00045790
	MY=0	00045800
	DO 210 IU=1,NUP	00045810
	DO 210 IW=1,NVS	00045820
	MY=MY+1	00045830
	AX(MY)=AX(MY)+SEU(IW)*C(I,IU)	00045840
	SEVT=SEV(IW)*SDS-SEW(IW)*COSDP	00045850
	IF(IPASS.EQ.0) SEVT=-SEVT	00045860
	AY(MY)=AY(MY)+SEVT *C(I,IU)	00045870
	SEWT=SEV(IW)*COSDP+SEW(IW)*SDS	00045880
210	AZ(MY)=AZ(MY)+SEWT *C(I,IU)	00045890
250	CONTINUE	00045900
		00045910
	IF(IPASS.EQ.0) GO TO 252	00045920
251	IPASS=0	00045930
	IF(IMAGE.EQ.0) GO TO 252	00045940
	TVT=TVV*SDS-TWV*COSDP	00045950
	TWT=TVV*COSDP+TWV*SDS	00045960
	YQT=-YQS-PYB	00045970
	GO TO 20	00045980
		00045990

252	TUV=TUV/PI4/BETA	00046000
	TVY=TVV	00046010
	TVVT=TVY*SDS-TWV*COSDP	00046020
	IF (IMAGE.NE.0) TVVT=-TVVT	00046030
	TVV=(TVVT+TVT)/PI4	00046040
	TWV=(TVY*COSDP+TWV*SDS+TWT)/PI4	00046050
C		00046060
	IF (ITYP.LT.19) GO TO 264	00046070
	TVY=TVV	00046080
	TVV=TVY*SINDP+TWV*COSDP	00046090
	TWV=-TVY*COSDP+TWV*SINDP	00046100
	DO 260 I=1,NKP	00046110
	AYT=AY(I)	00046120
	AY(I)=AYT*SINDP+AZ(I)*COSDP	00046130
260	AZ(I)=-AYT*COSDP+AZ(I)*SINDP	00046140
C		00046150
264	IF (ITYP.NE.19) GO TO 280	00046160
	READ (9) (AZU(IU),IU=1,NKP)	00046170
	DO 270 IU=1,NKP	00046180
270	AZ(IU)=AZU(IU)-AZ(IU)	00046190
280	WRITE (22) (AX(IU),AY(IU),AZ(IU),IU=1,NKP),TUV,TVV,TWV	00046200
C	RESET Q COORDINATES	00046210
	XQ(IQ)=XQ(IQ)+POX	00046220
	YQ(IQ)=YQS	00046230
	ZQ(IQ)=ZQT+PZB	00046240
300	CONTINUE	00046250
C		00046260
	REWIND 9	00046270
	WRITE (6,350) ITYP	00046280
350	FORMAT (21H0 ** COMPLETED MATRIX,14)	00046290
	RETURN	00046300
	END	00046310
*DECK	AQXVPR	00046320
	SUBROUTINE AQXVPR (IQ,IPASS,SDS)	00046330
C		00046340
C*	CALC MATRIX-A *Q ON COMPONENTS AND VORTICES ON PYLON ROOT	00046350
C		00046360
	COMMON/DAT/ DA(5000)	00046370
C		00046380
	EQUIVALENCE (DA(3383),XLP)	00046390
C		00046400
	COMMON/CRG/ RDG(5),COSDP	00046410
	COMMON/CFG/ IDG(13),NKP	00046420

	COMMON/CPP/ HPS,NVSH,NVT,DL,NEVP,ZCON,YCON,NVC,NVS,NJC,NUP	00046430
	LARGE DM1(75130)	00046440
1	,EV(22) ,XLV(22) ,XTLV(22) ,WPE(39)	00046450
2	,XD(60) ,YD(60) ,ZD(60)	00046460
3	,XB(400) ,YB(400) ,ZB(400)	00046470
4	,DM2(3526) ,CSEC(21,10),DM3(840) ,CEL(3700)	00046480
5	,XQ(650) ,YQ(650) ,ZQ(650)	00046490
6	,AX(250) ,AY(250) ,AZ(250) ,A(250)	00046500
7	,QBX(600) ,QBY(600) ,QBZ(600)	00046510
8	,QTX(600) ,QTY(600) ,QTZ(600)	00046520
C		00046530
C*	CONTROL POINT INDEX	00046540
	NVCT=NVC+NVT-1	00046550
	IF(IPASS.EQ.1) GO TO 10	00046560
	IF(ABS((YB(4)-YQ(IQ))/(YB(1)-YB(2))).GT.XLP) GO TO 400	00046570
	GO TO 14	00046580
10	DO 12 I=1,NKP	00046590
	AX(I)=0.0	00046600
	AY(I)=0.0	00046610
12	AZ(I)=0.0	00046620
C	NVX AND NVY LOOPS	00046630
14	LI=1	00046640
	LF=5	00046650
	JS=0	00046660
	DO 300 IX=1,NVCT	00046670
	K=0	00046680
	IF(IX.LE.NVC) GO TO 20	00046690
	JS=JS-4	00046700
	K=-1	00046710
20	DO 200 IS=1,4	00046720
	JS=JS+1	00046730
30	TZ1=ZB(LF)-ZB(LI)	00046740
	TZ2=ZB(LF)-ZQ(IQ)	00046750
	TZ3=ZB(LI)-ZQ(IQ)	00046760
50	TX1=XB(LF)-XB(LI)	00046770
	TX2=XB(LF)-XQ(IQ)	00046780
	TX3=XB(LI)-XQ(IQ)	00046790
	TY1=YB(LF)-YB(LI)	00046800
	TY2=YB(LF)-YQ(IQ)	00046810
	TY3=YB(LI)-YQ(IQ)	00046820
	HN=TX2*TX1+TY2*TY1+TZ2*TZ1	00046830
	HD=TX1**2+TY1**2+TZ1**2	00046840
	SQHD=SQRT(HD)	00046850

	IF(SQHD.GT.0.00001) GO TO 52	00046860
	QSR=0.0	00046870
	GO TO 54	00046880
52	HT=HN/HD	00046890
	HX=TX2-HT*TX1	00046900
	HY=TY2-HT*TY1	00046910
	HZ=TZ2-HT*TZ1	00046920
	SHP=SQRT(HX**2+HY**2+HZ**2)	00046930
	SRF=HN/SQRT(TX2**2+TY2**2+TZ2**2)	00046940
	HI=TX1*TX3+TY1*TY3+TZ1*TZ3	00046950
	SRI=HI/SQRT(TX3**2+TY3**2+TZ3**2)	00046960
	QT=(SRF-SRI)/SQHD/SHP	00046970
	TX=TY2*TZ1-TZ2*TY1	00046980
	TY=TZ2*TX1-TX2*TZ1	00046990
	TZ=TX2*TY1-TY2*TX1	00047000
	QSR=QT/SQRT(TX**2+TY**2+TZ**2)	00047010
54	IF(K) 150,140,70	00047020
C*	LATERAL VORTICES	00047030
70	QBX(JS)=QSR*TX	00047040
	QBY(JS)=QSR*TY	00047050
	QBZ(JS)=QSR*TZ	00047060
	LI=LF	00047070
	LF=LI+4	00047080
	K=0	00047090
	GO TO 30	00047100
C*	LONGITUDINAL VORTICES	00047110
140	QTX(JS)=QSR*TX	00047120
	QTY(JS)=QSR*TY	00047130
	QTZ(JS)=QSR*TZ	00047140
	LF=LI+1	00047150
	K=1	00047160
	GO TO 200	00047170
150	QTX(JS)=QSR*TX+QTX(JS)	00047180
	QTY(JS)=QSR*TY+QTY(JS)	00047190
	QTZ(JS)=QSR*TZ+QTZ(JS)	00047200
	LI=LI+1	00047210
	LF=LI+4	00047220
200	CONTINUE	00047230
	IF(IX.GT.NVC) GO TO 300	00047240
	LI=LI+1	00047250
	LF=LI+4	00047260
300	CONTINUE	00047270
C		00047280

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
QTZ(JS)=QTZ(JS)-QSR*HY
320 LI=LI+1

```

00047290
00047300
00047310
00047320
00047330 ✓
00047340
00047350
00047360
00047370
00047380
00047390
00047400
00047410
00047420
00047430
00047440
00047450
00047460
00047470
00047480
00047490
00047500
00047510
00047520
00047530
00047540
00047550
00047560
00047570
00047580
00047590
00047600
00047610
00047620
00047630
00047640
00047650
00047660
00047670
00047680
00047690
00047700
00047710

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

```

245


```

END 00047720
*DECK AQPVP 00047730
SUBROUTINE AQPVP 00047740
C 00047750
C* CALC MATRIX-A Q ON PYLON AND VERTICES ON PYLON 00047760
C 00047770
COMMON/DAT/ DA(5000) 00047780
C 00047790
DIMENSION PJC(20) 00047800
C 00047810
EQUIVALENCE (DA(2507),PNVC) ,(DA(3383),XLP) ,(DA(2512),PJC) 00047820
C 00047830
COMMON/CRG/ PI,PI4,RC,BETA 00047840
COMMON/CFG/ IDG(13),NKP 00047850
COMMON/PPP/ HPS,NVSH,NVT,DL,NEVP,ZCON,YCON,NVC,NVS,NJC,NUP 00047860
C 00047870
LARGE DM1(75130) 00047880
1 ,EV(22) ,XLV(22) ,XTLV(22) ,WPE(39) 00047890
2 ,XD(60) ,YD(60) ,ZD(60) 00047900
3 ,XB(400) ,YB(400) ,ZB(400) 00047910
4 ,CFI(20) ,CFJ(20) ,STH(20) ,ACT(20) 00047920
5 ,ET(22) ,XLT(22) ,XTLT(22) ,C(20,20) 00047930
6 ,CSJ(20,20) ,TSJ(20,20) ,UV(20,20) 00047940
7 ,UVP(20,20) ,VV(20,20) ,VVP(20,20) 00047950
8 ,DM2(790) ,TSI(21,20) ,TZI(21,20) ,CEL(3700) 00047960
9 ,XQ(650) ,YQ(650) ,ZQ(650) ,AZ(250) ,SUMES(30) 00047970
C 00047980
C 00047990
REWIND 9 00048000
NE1=NEVP 00048010
NE2=NVSH-1 00048020
INDT=DA(3000) 00048030
IF(INDT.NE.0) NE1=NVSH 00048040
10 DL4=4./DL/BETA 00048050
TL=2./DL 00048060
X2=TL/HPS 00048070
DTH=PI/PNVC*.5 00048080
DTX=X2*DTH 00048090
TDY=HPS/TL 00048100
C* Q ON PYLON LOOP 00048110
IP=0 00048120
IC=0 00048130
DO 350 KP=1,NE2 00048140

```

	DO 340 J=1,NJC	00048150
	JX=PJC(J)	00048160
	PUV=0.	00048170
	PVV=0.	00048180
	TUV=0.	00048190
	TVV=0.	00048200
	IC=IC+1	00048210
	IF(KP.LE.NEVP) GO TO 20	00048220
	IP=IP+1	00048230
	XP=XD(IP)	00048240
	EVP=YD(IP)/HPS	00048250
	GO TO 30	00048260
20	XP=XLV(KP)+XTLV(KP)*CFJ(JX+1)	00048270
	EVP=EV(KP)	00048280
30	DO 32 I=1,NKP	00048290
32	AZ(I)=0.0	00048300
C	I LOOP	00048310
40	IR=1	00048320
	DO 330 I=1,NVC	00048330
	DO 100 NS=1,NVS	00048340
100	SUMES(NS)=0.0	00048350
C	ETAV LOOP	00048360
	XV3=XLT(1)+XTLT(1)*CFI(I)	00048370
	XV2=XLV(1)+XTLV(1)*CFI(I)	00048380
	XVT=XLV(NVSH)+XTLV(NVSH)*CFI(I)	00048390
	TPT=(XVT-XV3)/TDY	00048400
	DO 310 KV=1,NE1	00048410
	EVI=0.	00048420
	EVO=0.	00048430
	EVY=EV(KV)	00048440
	IF(INDT.EQ.0) GO TO 140	00048450
	IF(KV.NE.NE1) GO TO 120	00048460
	TPI=TPT	00048470
	TPO=-TPT	00048480
	XC=X2*(XP-XVT)	00048490
	GO TO 190	00048500
120	IF(KV.LE.NEVP) GO TO 140	00048510
	TPO=(XB(IR)-XB(IR+1))/(YB(IR)-YB(IR+1))	00048520
130	EVY=(YB(IR)+YB(IR+1))/2./HPS	00048530
	XC=X2*(XP-(XB(IR)+XB(IR+1))/2.)	00048540
	TPI=TPO	00048550
	IR=IR+1	00048560
	IF(KV.EQ.NE2) IR=IR+1	00048570

	GO TO 190	00048580
140	IF(KV.NE.1) GO TO 180	00048590
	TPI=TPT	00048600
	TPO=TPT	00048610
	XC=4.*X2*(XP-XV2)	00048620
	GO TO 190	00048630
180	XV2=XLV(KV)+XTLV(KV)*CFI(I)	00048640
	XV3=XLT(KV)+XILT(KV)*CFI(I)	00048650
	TPI=(XV2-XV3)/TDY	00048660
	TPO=(XV1-XV2)/TDY	00048670
	IF(KV.EQ.NE2) GO TO 182	00048680
	IF(ET(KV).LT.0.0) TPI=TPO	00048690
182	XC=X2*(XP-XV2)	00048700
190	YC=TL*(EVP-EVY)	00048710
	IF(ABS(YC/2.).GT.XLP) GO TO 300	00048720
	IF(KV.EQ.1) YC=4.*YC	00048730
	YCP=YC+1.	00048740
	YCM=YC-1.	00048750
	XTO=XC-TPO	00048760
	XTI=XC+TPI	00048770
	SQO=SQRT(XTO**2+YCM**2)	00048780
	SQI=SQRT(XTI**2+YCP**2)	00048790
	EC=(XTI/SQI+1.)/YCP-(XTO/SQO+1.)/YCM	00048800
	SQC=SQRT(XC**2+YC**2)	00048810
	C1=TPI**2+1.	00048820
	C4=C1	00048830
	C2=XC-YC*TPI	00048840
	IF(C2**2/C1.LE.1.E-10) GO TO 230	00048850
	C3=YC+XC*TPI	0004
	EVI=((C3+C1)/SQI-C3/SQC)/C2	00048860
230	C1=TPO**2+1.	00048870
	C2=XC-YC*TPO	00048880
	IF(C2**2/C1.LE.1.E-10) GO TO 250	00048890
	C3=YC+XC*TPO	00048900
240	EVO=((C3-C1)/SQO-C3/SQC)/C2	00048910
250	IF(INDT.EQ.0) GO TO 290	00048920
	IF(KV.EQ.1) GO TO 290	00048930
	STS=1./SQC	00048940
	EUI=1./SQI-STS	00048950
	EUO=1./SQO-STS	00048960
	AYC=ABS(YC/X2)	00048970
	ST=TSI(KV,I)*STH(I)*DTX	00048980
	SP=TZI(KV,I)/STH(I)*DTX*2.0	00048990

	TVI=ST*EVI	00049000
	TVO=ST*EVO	00049010
260	IF(KV.NE.NE1) GO TO 280	00049020
	PUV=PUV-SP*(EUI*TPI-EVI)/C4	00049030
	PVV=PVV-SP*(EVI*TPI+EUI)/C4	00049040
	EUI=ST*EUI	00049050
	TUV=TUV-(EUI*TPI-TVI)/C4	00049060
	TVV=TVV-(TVI*TPI+EUI)/C4	00049070
	GO TO 300	00049080
280	PUV=PUV+SP*((EVO*TPO-EVO)/C1-(EUI*TPI-EVI)/C4)	00049090
	PVV=PVV+SP*((EVO*TPO+EVO)/C1-(EVI*TPI+EUI)/C4)	00049100
	EUI=ST*EUI	00049110
	EVO=ST*EVO	00049120
	TUV=TUV+(EVO*TPO-TVO)/C1-(EUI*TPI-TVI)/C4	00049130
	TVV=TVV+(TVO*TPO+EVO)/C1-(TVI*TPI+EUI)/C4	00049140
290	IF(KV.GT.NEVP) GO TO 300	00049150
	EC=(EC+EVI-EVO)*DL4	00049160
	SUMES(KV)=SUMES(KV)+EC	00049170
300	XV1=XV3	00049180
310	CONTINUE	00049190
	SUMES(1)=4.*SUMES(1)	00049200
	MY=0	00049210
312	DO 314 NC=1,NUP	00049220
	DO 314 NS=1,NVS	00049230
	MY=MY+1	00049240
314	AZ(MY)=AZ(MY)+SUMES(NS)*C(I,NC)	00049250
330	CONTINUE	00049260
	WRITE (9) (AZ(MY),MY=1,NKP)	00049270
	UVP(KP,J)=PUV/PI4/BETA	00049280
	VVP(KP,J)=PVV/PI4	00049290
	UV(KP,J)=TUV/PI4/BETA	00049300
340	VV(KP,J)=TVV/PI4	00049310
350	CONTINUE	00049320
	REWIND 9	00049330
	IF(INDT.NE.0) CALL AQPVP	00049340
	RETURN	00049350
	END	00049360
*DECK	AQPVP	00049370
	SUBROUTINE AQPVP	00049380
C		00049390
C*	CALC ADDITIONAL INFLUENCE OF PYLON THICKNESS	00049400
C	DUE TO BREAKS ETC.	00049410
C		00049420

219

	COMMON/DAT/ DA(5000)	00049430
C		00049440
	DIMENSION PJC(20)	00049450
C		00049460
	EQUIVALENCE (DA(2507),PNVC) ,(DA(3383),XLP) ,(DA(2512),PJC)	00049470
C		00049480
	COMMON/CRG/ PI,PI4,RC,BETA	00049490
	COMMON/PPP/ HPS,NVSH,NVT,DL,NEVP,ZCON,YCON,NVC,NVS,NJC,NUP	00049500
C		00049510
	LARGE DM1(75130)	00049520
1	,EV(22) ,XLV(22) ,XTLV(22) ,WPE(39)	00049530
2	,XD(60) ,YD(60) ,ZD(60)	00049540
3	,XB(400) ,YB(400) ,ZB(400)	00049550
4	,CFI(20) ,CFJ(20) ,STH(20) ,ACT(20)	00049560
5	,ET(22) ,XLT(22) ,XTLI(22) ,C(20,20)	00049570
6	,CSJ(20,20),TSJ(20,20),UV(20,20)	00049580
7	,UVP(20,20),VV(20,20),VVP(20,20)	00049590
8	,DM2(790) ,TSI(21,20),TZI(21,20),CEL(3700)	00049600
9	,XQ(650) ,YQ(650) ,ZQ(650) ,AZ(250) ,SUMES(30)	00049610
C		00049620
C		00049630
	NE1=NVSH	00049640
	NE2=NVSH-1	00049650
10	DL4=4./DL/BETA	00049660
	TL=2./DL	00049670
	X2=TL/HPS	00049680
	DTH=PI/PNVC*.5	00049690
	DTX=X2*DTH	00049700
	TDY=HPS/TL	00049710
C*	Q ON PYLON LOOP	00049720
	IP=0	00049730
	IC=0	00049740
	DO 350 KP=1,NE2	00049750
	DO 340 J=1,NJC	00049760
	JX=PJC(J)	00049770
	PUV=0.	00049780
	PVV=0.	00049790
	TUV=0.	00049800
	TVV=0.	00049810
	IC=IC+1	00049820
	IF(KP.LE.NEVP) GO TO 20	00049830
	IP=IP+1	00049840
	XP=XD(IP)	00049850

	EVP=YD(IP)/HPS	00049860
	GO TO 40	00049870
20	XP=XLV(KP)+XTLV(KP)*CFJ(JX+1)	00049880
	EVP=EV(KP)	00049890
C	I LOOP	00049900
40	IR=1	00049910
	DO 330 I=2,NVC	00049920
C	ETAV LOOP	00049930
	STHJ=SIN(ACT(I))	00049940
	XV3=XLT(1)+XTLT(1)*CFJ(I)	00049950
	XV2=XLV(1)+XTLV(1)*CFJ(I)	00049960
	XVT=XLV(NVSH)+XTLV(NVSH)*CFJ(I)	00049970
	TPT=(XV1-XV3)/TDY	00049980
	DO 310 KV=1,NE1	00049990
	EVI=0.	00050000
	EVO=0.	00050010
	EVY=EV(KV)	00050020
	IF(KV.NE.NE1) GO TO 120	00050030
	TPI=TPT	00050040
	TPO=-TPT	00050050
	XC=X2*(XP-XVT)	00050060
	GO TO 190	00050070
120	IF(KV.LE.NEVP) GO TO 140	00050080
	XB1=(XB(IR)+XB(IR+4))/2.	00050090
	XB2=(XB(IR+1)+XB(IR+5))/2.	00050100
	YB1=(YB(IR)+YB(IR+4))/2.	00050110
	YB2=(YB(IR+1)+YB(IR+5))/2.	00050120
	TPO=(XB1-XB2)/(YB1-YB2)	00050130
	EVY=(YB1+YB2)/2./HPS	00050140
	XC=X2*(XP-(XB1+XB2)/2.)	00050150
	TPI=TPO	00050160
	IR=IR+1	00050170
	IF(KV.EQ.NE2) IR=IR+1	00050180
	GO TO 190	00050190
140	IF(KV.NE.1) GO TO 180	00050200
	TPI=TPT	00050210
	TPO=TPT	00050220
	XC=4.*X2*(XP-XV2)	00050230
	GO TO 190	00050240
180	XV2=XLV(KV)+XTLV(KV)*CFJ(I)	00050250
	XV3=XLT(KV)+XTLT(KV)*CFJ(I)	00050260
	TPI=(XV2-XV3)/TDY	00050270
	TPO=(XV1-XV2)/TDY	00050280

	IF(KV.EQ.NE2) GO TO 182	00050290
	IF(ET(KV).LT.0.0) TPI=TPO	00050300
182	XC=X2*(XP-XV2)	00050310
190	YC=TL*(EVP-EVY)	00050320
	IF(ABS(YC/2.).GT.XLP) GO TO 300	00050330
	IF(KV.EQ.1) YC=4.*YC	00050340
	YCP=YC+1.	00050350
	YCM=YC-1.	00050360
	XTO=XC-TPO	00050370
	XTI=XC+TPI	00050380
	SQO=SQRT(XTO**2+YCM**2)	00050390
	SQI=SQRT(XTI**2+YCP**2)	00050400
	SQC=SQRT(XC**2+YC**2)	00050410
	C1=TPI**2+1.	00050420
	C4=C1	00050430
	C2=XC-YC*TPI	00050440
	IF(C2**2/C1.LE.1.E-10) GO TO 230	00050450
	C3=YC+XC*TPI	00050460
	EVI=((C3+C1)/SQI-C3/SQC)/C2	00050470
230	C1=TPO**2+1.	00050480
	C2=XC-YC*TPO	00050490
	IF(C2**2/C1.LE.1.E-10) GO TO 250	00050500
	C3=YC+XC*TPO	00050510
240	EVO=((C3-C1)/SQO-C3/SQC)/C2	00050520
250	IF(KV.EQ.1) GO TO 300	00050530
	IF(SQC.GT.1.E-10) GO TO 254	00050540
	STS=0.0	00050550
	GO TO 256	00050560
254	STS=1./SQC	00050570
256	EUI=1./SQI-ST5	00050580
	EUO=1./SQO-ST5	00050590
	AYC=ABS(YC/X2)	00050600
	SLOPE=(TSI(KV,I)+TSI(KV,I-1))/2.0	00050610
	DEFL=(TZI(KV,I)+TZI(KV,I-1))/2.0	00050620
	ST=SLOPE*STHJ*DTX	00050630
	SP=DEFL/STHJ*DTX*2.0	00050640
	TVI=ST*EVI	00050650
	TVO=ST*EVO	00050660
260	IF(KV.NE.NE1) GO TO 280	00050670
	PUV=PUV-SP*(EUI*TPI-EVI)/C4	00050680
	PVV=PVV-SP*(EVI*TPI+EUI)/C4	00050690
	EUI=ST*EUI	00050700
	TUV=TUV-(EUI*TPI-TV1)/C4	00050710

	TVV=TVV-(TVI*TPI+EUI)/C4	00050720
	GO TO 300	00050730
280	PUV=PUV+SP*((EUO*TPO-EVO)/C1-(EUI*TPI-EVI)/C4)	00050740
	PVV=PVV+SP*((EVO*TPO+EUO)/C1-(EVI*TPI+EUI)/C4)	00050750
	EUI=ST*EUI	00050760
	EUO=ST*EUO	00050770
	TUV=TUV+(EUO*TPO-TVO)/C1-(EUI*TPI-TVI)/C4	00050780
	TVV=TVV+(TVO*TPO+EUO)/C1-(TVI*TPI+EUI)/C4	00050790
300	XV1=XV3	00050800
310	CONTINUE	00050810
330	CONTINUE	00050820
	UVP(KP,J)=PUV/PI4/BETA+UVP(KP,J)	00050830
	VVP(KP,J)=PVV/PI4+VVP(KP,J)	00050840
	UV(KP,J)=TUV/PI4/BETA+UV(KP,J)	00050850
340	VV(KP,J)=TVV/PI4+VV(KP,J)	00050860
350	CONTINUE	00050870
	RETURN	00050880
	END	00050890
*DECK	NACELL	00050900
	SUBROUTINE NACELL	00050910
C		00050920
C*	CONTROL ROUTINE TO CALC. INFLUENCE MATRICES OF NACELLE	00050930
C	ON OTHER COMPONENTS AND NACELLE ON NACELLE	00050940
C		00050950
C	COMMON/DAT/ DA(5000)	00050960
C		00050970
C	EQUIVALENCE (DA(2492),PDA) ,(DA(3391),XNO) ,(DA(3392),YNO)	00050980
	1,(DA(3393),ZNO)	00050990
C		00051000
C	COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP	00051010
	COMMON/CFG/ ID1(5),NQB,NQF,NQW,NQP,NQN	00051020
C		00051030
C	LARGE DM1(81191),XOB(1000),YOB(1000),ZOB(1000),DM2(700)	00051040
	1,X2(1000),Y2(1000),SDT(1000),CDT(1000)	00051050
C		00051060
C	INITIAL	00051070
	PDAR=PDA/RC	00051080
	NQT=NQB+NQF+NQW+NQP	00051090
	YNB=YNO*BETA	00051100
	ZNB=ZNO*BETA	00051110
C	INFLUENCE OF RIGHT SIDE NACELLE	00051120
	L1=NQT	00051130
	DO 10 I=1,L1	00051140

	X2(I)=XOB(I)-XNO	00051150
	TY=YOB(I)-YNB	00051160
	TZ=ZOB(I)-ZNB	00051170
	YNQ=-TY*SINDP-TZ*COSDP	00051180
	ZNQ=TY*COSDP-TZ*SINDP	00051190
	Y2(I)=SQRT(YNQ**2+ZNQ**2)	00051200
	THE=ATAN2(ZNQ,YNQ)	00051210
	ANG=PDAR-THE	00051220
	SDT(I)=SIN(ANG)	00051230
10	CDT(I)=COS(ANG)	00051240
	CALL AQXVN(1,L1)	00051250
C	TEST FOR IMAGE	0005
	SYM=YNO+PDA	00051260
	IF(SYM.EQ.0.0) GO TO 30	00051270
C	INFLUENCE OF LEFT SIDE NACELLE	00051280
	L1=NQT+NQN	00051290
	DO 20 I=1,L1	00051300
	X2(I)=XOB(I)-XNO	00051310
	TY=YOB(I)+YNB	00051320
	TZ=ZOB(I)-ZNB	00051330
	YNQ=TY*SINDP-TZ*COSDP	00051340
	ZNQ=TY*COSDP+TZ*SINDP	00051350
	Y2(I)=SQRT(YNQ**2+ZNQ**2)	00051360
	THE=ATAN2(ZNQ,YNQ)	00051370
	ANG=-PDAR-THE	00051380
	SDT(I)=SIN(ANG)	00051390
20	CDT(I)=COS(ANG)	00051400
	CALL AQXVN(2,L1)	00051410
C		00051420
30	REWIND 9	00051430
	WRITE(6,50)	00051440
50	FORMAT(44H0 ** COMPLETED INFLUENCE MATRICES OF NACELLE)	00051450
C		00051460
	RETURN	00051470
	END	00051480
*DECK	AQXVN	00051490
	SUBROUTINE AQXVN(IPASS,L1)	00051500
C		00051510
C*	CALCULATE INFLUENCE MATRICES OF NACELLE	00051520
C		00051530
	COMMON/DAT/ DA(5000)	00051540
C		00051550
	EQUIVALENCE (DA(2492),PDA) ,(DA(3392),YNO) ,(DA(3385),XLNR)	00051560

```

C      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

```

```

00051570
00051580
00051590
00051600
00051610
00051620
00051630
00051640
00051650
00051660
00051670
00051680
00051690
00051700
00051710
00051720
00051730
00051740
00051750
00051760
00051770
00051780
00051790
00051800
00051810
00051820
00051830
00051840
00051850
00051860
00051870
00051880
00051890
00051900
00051910
00051920
00051930
00051940
00051950
00051960
00051970
00051980
00051990

```

255

100	CONTINUE	00052000
110	J1=J1+1	00052010
	NT=N1	00052020
C	SAVE A,B,C MATRICES ONTAPE	00052030
	IF(IPASS.EQ.2) GO TO 140	00052040
	DO 130 IK=1,NT	00052050
	A(IK)=AX(IK)	00052060
	B(IK)=-AY(IK)*SDT(I)	00052070
130	C(IK)=-AY(IK)*CDT(I)	00052080
	IF(SYM.EQ.0.0) GO TO 160	00052090
	WRITE (11) (A(IK),B(IK),C(IK),IK=1,NT)	00052100
	GO TO 400	00052110
C	140 READ (11) (A(IK),B(IK),C(IK),IK=1,NT)	00052120
	DO 150 IK=1,NT	00052130
	A(IK)=A(IK)+AX(IK)	00052140
	B(IK)=B(IK)-AY(IK)*SDT(I)	00052150
150	C(IK)=C(IK)-AY(IK)*CDT(I)	00052160
C	160 IF(BON.EQ.0.0) GO TO 180	00052170
	IF(I.LE.IS) GO TO 200	00052180
	IF(I.GT.IE) GO TO 180	00052190
	DO 170 IK=1,NT	00052200
	YT=B(IK)	00052210
	B(IK)=YT*SINDP+C(IK)*COSDP	00052220
170	C(IK)=-YT*COSDP+C(IK)*SINDP	00052230
	GO TO 200	00052240
C	180 DO 190 IK=1,NT	00052250
	YT=B(IK)	00052260
	B(IK)=-YT*SINDP-C(IK)*COSDP	00052270
190	C(IK)=YT*COSDP-C(IK)*SINDP	00052280
C	200 WRITE (9) (A(IK),B(IK),C(IK),IK=1,NT)	00052290
C	400 CONTINUE	00052300
	IF(IPASS.EQ.2) GO TO 500	00052310
	* TEST IF OFF BODY COMPLETED	00052320
410	IF(BON) 420,500,420	00052330
420	BON=0.0	00052340
	L1=NT	00052350
	N1=0	00052360
	J1=0	00052370
		00052380
		00052390
		00052400
		00052410
		00052420

	DO 450 K=1,NB	00052430
	M1=N1+1	00052440
	N1=N1+ND(K)-1	00052450
	DO 440 J=M1,N1	00052460
	J1=J1+1	00052470
	X2(J)=(X1(J1+1)+X1(J1))/2.	00052480
	Y2(J)=(Y1(J1+1)+Y1(J1))/2.	00052490
440	CONTINUE	00052500
450	CONTINUE	00052510
	GO TO 10	00052520
500	CONTINUE	00052530
	REWIND 11	00052540
	RETURN	00052550
	END	00052560
*DECK	XYZ	00052570
	SUBROUTINE XYZ	00052580
C		00052590
C	* CONTROL FOR X,Y,Z MATRICES COMPUTATION	00052600
C		00052610
C		00052620
C	COMMON/CPN/ NB,ND(3),NT,BON,YZERO,IAC,I,J,J1	00052630
1	,SJ,DS,DX,DY,NI,XJ,YJ,XK,EEK,EKK	00052640
C		00052650
	LARGE DM1(81191),XOB(1000),YOB(1000),ZOB(1000)	00052660
1	,X1(140),Y1(140),SINA(140),COSA(140),DELS(140)	00052670
2	,X2(1000),Y2(1000),SDT(1000),CDT(1000)	00052680
3	,A(140),B(140),C(140),AX(140),AY(140)	00052690
C		00052700
C	* START	00052710
	IF (BON) 100,10,100	00052720
10	IF (J-1) 110,20,110	00052730
C	* J EQUAL I PATH	00052740
20	T1=.5*DELS(J)	00052750
	SJ=T1/Y2(J)	00052760
	IF (SJ-.08) 30,30,40	00052770
30	CALL XYZ1	00052780
	GO TO 1000	00052790
40	SJ=.08	00052800
	CALL XYZ1	00052810
	NI=33	00052820
	T2=.08*Y2(J)	00052830
	DS=(T1-T2)/32.	00052840
	DX=DS*COSA(J)	00052850

	DY=DS*SINA(J)	00052860
	XJ=X2(J)+T2*COXA(J)-DX	00052870
	YJ=Y2(J)+T2*SINA(J)-DY	00052880
	CALL XYZ2	00052890
	GO TO 300	00052900
C	* INITIAL Y COORDINATE MID-POINT FOR ZERO TEST	00052910
100	YZERO=Y2(I)-.000001	00052920
C	* J NOT EQUAL I PATH	00052930
C	* COMPUTE MINIMUM DISTANCE TO I MIDPOINT	00052940
110	D1=(X2(I)-X1(J1))**2+(Y2(I)-Y1(J1))**2	00052950
	X22=(X1(J1+1)+X1(J1))*0.5	00052960
	Y22=(Y1(J1+1)+Y1(J1))*0.5	00052970
	D2=(X2(I)-X22)**2+(Y2(I)-Y22)**2	00052980
	D3=(X2(I)-X1(J1+1))**2+(Y2(I)-Y1(J1+1))**2	00052990
	IF (D1-D2) 130,130,120	00053000
120	IF (D2-D3) 150,150,140	00053010
130	IF (D1-D3) 160,160,140	00053020
140	DM=SQRT(D3)	00053030
	GO TO 170	00053040
150	DM=SQRT(D2)	00053050
	GO TO 170	00053060
160	DM=SQRT(D1)	00053070
C	* COMPUTE NO. OF INTERVALS(NI) AND DELTA S (DS)	00053080
C	FOR SIMPSON RULE INTEGRATION	00053090
170	IF (DM.EQ.0.0) GO TO 200	00053100
	NI=8.*DELS(J)/DM+0.9	00053110
	IF (NI) 180,180,190	00053120
180	NI=3	00053130
	DS=DELS(J)/2.	00053140
	GO TO 220	00053150
190	NI=NI+NI	00053160
	IF (NI-128) 210,200,200	00053170
200	NI=129	00053180
	DS=DELS(J)/128.	00053190
	GO TO 220	00053200
210	XNI=NI	00053210
	DS=DELS(J)/XNI	00053220
	NI=NI+1	00053230
220	DX=DS*COXA(J)	00053240
	DY=DS*SINA(J)	00053250
300	XJ=X1(J1)-DX	00053260
	YJ=Y1(J1)-DY	00053270
	CALL XYZ2	00053280

```

1000 RETURN                                00053290
      END                                  00053300
*DECK XYZ1                                00053310
      SUBROUTINE XYZ1                     00053320
C                                           00053330
C                                           00053340
C           * COMPUTE X,Y,Z MATRICES FOR SJ LESS THAN OR EQUAL .08 00053350
C                                           00053360
C                                           00053370
      COMMON/CPN/ NB,ND(3),NT,BON,YZERO,IAC,I,J,J1
1      ,SJ,DS,DX,DY,NI,XJ,YJ,XK,EEK,EKK  00053380
C                                           00053390
C
      LARGE      DM1(81191),XOB(1000) ,YOB(1000) ,ZOB(1000) 00053400
1      ,X1(140)  ,Y1(140)  ,SINA(140) ,COSA(140) ,DELS(140) 00053410
2      ,X2(1000) ,Y2(1000) ,SDT(1000) ,CDT(1000) 00053420
3      ,A(140)  ,B(140)  ,C(140)  ,AX(140)  ,AY(140) 00053430
C                                           00053440
C           * START                                  00053450
C           * INITIALIZE                             00053460
C
      T1=SJ*SJ                                00053470
      T2=ALOG(SJ/8.)                          00053480
      T3=SINA(J)*SINA(J)                      00053490
      T4=T2+T3                                00053500
      T5=.666666667*T3                        00053510
      T6=T5*T3                                00053520
      T7=SJ+SJ                                00053530
      T8=T7+T7                                00053540
      T9=6.2831853*COSA(J)                   00053550
      T10=6.2831853*SINA(J)                  00053560
C           * AXIS FLOW                             00053570
C
20     T11=T1*SJ                              00053580
      AX(J)=T10+SINA(J)*COSA(J)*T7+(T4+2.16666667)*T11/12.) 00053590
      AY(J)=T7*T4-T9-(1.+T2-T3-T6)*T11/8.    00053600
100    RETURN                                00053610
      END                                  00053620
*DECK XYZ2                                00053630
      SUBROUTINE XYZ2                     00053640
C                                           00053650
C                                           00053660
C           * COMPUTE X,Y,Z MATRICES USING SIMPSON RULE INTEGRATION 00053670
C                                           00053680
C                                           00053690
      COMMON/CPN/ NB,ND(3),NT,BON,YZERO,IAC,I,J,J1
1      ,SJ,DS,DX,DY,NI,XJ,YJ,XK,EEK,EKK  00053700
C                                           00053710

```

259

```

LARGE      DM1(81191),XOB(1000) ,YOB(1000) ,ZOB(1000)      00053720
1          ,X1(140)   ,Y1(140)   ,SINA(140) ,COSA(140) ,DELS(140) 00053730
2          ,X2(1000) ,Y2(1000) ,SDT(1000) ,CDT(1000)      00053740
3          ,A(140)   ,B(140)   ,C(140)   ,AX(140)  ,AY(140) 00053750
C
C          * START      00053760
C          * INITIALIZE 00053770
C          * INITIALIZE 00053780
S2=.6666666667*DS      00053790
T1=Y2(I)*Y2(I)        00053800
C          * NO. OF INTERVAL LOOP 00053810
DO 1000 IS=1,N1      00053820
XJ=XJ+DX              00053830
YJ=YJ+DY              00053840
T2=YJ*YJ              00053850
T3=X2(I)-XJ           00053860
T4=T3*T3              00053870
T5=(Y2(I)+YJ)**2      00053880
T6=T4+T5              00053890
T7=SQRT(T6)          00053900
                  00053910
T8=T2+T4              00053920
T9=(Y2(I)-YJ)**2      00053930
T10=T9+T4             00053940
C          * COMPUTE ELLIPIC INTEGRAL 00053950
XK=(T4+T9)/T6         00053960
CALL ELIP              00053970
C          * AXIS FLOW  00053980
20 T11=YJ/T7           00053990
F1=T11*T3*EEK/T10     00054000
IF (Y2(I).NE.0.0) GO TO 25 00054010
F2=0.0                00054020
GO TO 26               00054030
25 F2=T11*(EKK+EEK*(T1-T8)/T10)/Y2(I) 00054040
26 F3=T11*EEK          00054050
C          * SIMPSON RULE INTEGRATION 00054060
IF (IS-1) 30,30,40   00054070
C          * FIRST PASS  00054080
30 AXS=F1              00054090
AYS=F2                 00054100
IA=0                   00054110
GO TO 1000             00054120
40 IF (IS-N1) 50,80,50 00054130
50 IF (IA) 70,60,70    00054140

```

```

C          * EVEN PASS                                00054150
60 AXS=AXS+4.*F1                                     00054160
   AYS=AYS+4.*F2                                     00054170
   IA=1                                               00054180
   GO TO 1000                                         00054190
C          * ODD PASS                                00054200
70 AXS=AXS+F1+F1                                     00054210
   AYS=AYS+F2+F2                                     00054220
   IA=0                                               00054230
   GO TO 1000                                         00054240
C          * LAST PASS                                00054250
80 S4=S2+S2                                           00054260
   IF (J-I) 100,90,100                               00054270
90 IF (BON.NE.0.0) GO TO 100                          00054280
   AX(J)=AX(J)-S4*(AXS+F1)                           00054290
   AY(J)=AY(J)-S2*(AYS+F2)                           00054300
   GO TO 1000                                         00054310
100 AX(J)=-S4*(AXS+F1)                               00054320
    AY(J)=-S2*(AYS+F2)                               00054330
1000 CONTINUE                                         00054340
    RETURN                                           00054350
    END                                             00054360
*DECK ELIP                                           00054370
    SUBROUTINE ELIP                                   00054380
C                                                     00054390
C          * HASTINGS APPROXIMATION FOR ELLIPTIC INTERGALS 00054400
C                                                     00054410
COMMON/CPN/ NB,ND(3),NT,BON,YZERO,IAC,I,J,J1       00054420
1          ,SJ,DS,DX,DY,NI,XJ,YJ,XK,EEK,EKK        00054430
C                                                     00054440
C          * START                                    00054450
10 ETA=XK                                             00054460
   IF (ETA) 20,20,40                                 00054470
20 WRITE (6,30) ETA,I,J                             00054480
30 FORMAT (1H0 36H *** ERROR IN SUBROUTINE ELIP * ETA= F15.8 / 00054490
1          6H * I=I6,5X,5H * J=J16,5X,27H * ALOG(ETA) SET TO -88.028) 00054500
   ELN=-88.028                                       00054510
   ETA=0.0                                           00054520
   GO TO 50                                           00054530
40 ELN=ALOG(ETA)                                       00054540
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      0.004417870121)))
      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
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
C          COMBINE INFL OF FANPOD

```

```

00054580
00054590
00054600
00054610
00054620
00054630
00054640
00054650
00054660
00054670
00054680
00054690
00054700
00054710
00054720
00054730
00054740
00054750
00054760
00054770
00054780
00054790
00054800
00054810
00054820
00054830
00054840
00054850
00054860
00054870
00054880
00054890
00054900
00054910
00054920
00054930
00054940
00054950
00054960
00054970
00054980
00054990
00055000

```

	IF(IDF.EQ.0) GO TO 40	00055010
	READ (20) (AX(I),AY(I),AZ(I),I=1,NKF)	00055020
	DO 30 I=1,NKF	00055030
	IC=IC+1	00055040
	30 A(IC)=AX(I)*BXN(IQ)+AY(I)*BYN(IQ)+AZ(I)*BZN(IQ)	00055050
C	COMBINE INFL OF WING	00055060
	40 IX=IX+1	00055070
	IF(IDW.EQ.0) GO TO 60	00055080
	READ (21) (AX(I),AY(I),AZ(I),I=1,NKW),TUV,TVV,TWV	00055090
	BW(IX)=BETA*(BETA*TUV*BXN(IQ)+TVV*BYN(IQ)+TWV*BZN(IQ))	00055100
	DO 50 I=1,NKW	00055110
	IC=IC+1	00055120
	50 A(IC)=AX(I)*BXN(IQ)+AY(I)*BYN(IQ)+AZ(I)*BZN(IQ)	00055130
C	COMBINE INFL OF PYLON	00055140
	60 IF(IDP.EQ.0) GO TO 80	00055150
	READ (22) (AX(I),AY(I),AZ(I),I=1,NKP),TUV,TVV,TWV	00055160
	BP(IX)=BETA*(BETA*TUV*BXN(IQ)+TVV*BYN(IQ)+TWV*BZN(IQ))	00055170
	DO 70 I=1,NKP	00055180
	IC=IC+1	00055190
	70 A(IC)=AX(I)*BXN(IQ)+AY(I)*BYN(IQ)+AZ(I)*BZN(IQ)	00055200
C	WRITE INFL MATRIX ON UNIT 12	00055210
	80 WRITE (12) (A(I),I=1,IC)	00055220
	90 CONTINUE	00055230
C		00055240
C*	CONTROL POINTS Q ON FANPOD	00055250
	100 IF(IDF.EQ.0) GO TO 200	00055260
	DO 190 IQ=1,NQF	00055270
	IC=0	00055280
C	COMBINE INFL OF FUSELAGE	00055290
	IF(IDB.EQ.0) GO TO 130	00055300
	READ (19) (AX(I),AY(I),AZ(I),I=1,NKB)	00055310
	DO 120 I=1,NKB	00055320
	IC=IC+1	00055330
	120 A(IC)=AX(I)*FXN(IQ)+AY(I)*FYN(IQ)+AZ(I)*FZN(IQ)	00055340
	130 CONTINUE	00055350
C	COMBINE INFL OF FANPOD	00055360
	READ (20) (AX(I),AY(I),AZ(I),I=1,NKF)	00055370
	DO 140 I=1,NKF	00055380
	IC=IC+1	00055390
	140 A(IC)=AX(I)*FXN(IQ)+AY(I)*FYN(IQ)+AZ(I)*FZN(IQ)	00055400
	IX=IX+1	00055410
C	COMBINE INFL OF WING	00055420
	IF(IDW.EQ.0) GO TO 160	00055430

	READ (21) (AX(I),AY(I),AZ(I),I=1,NKW),TUV,TVV,TWV	00055440
	BW(IX)=BETA*(BETA*TUV*FXN(IQ)+TVV*FYN(IQ)+TWV*FZN(IQ))	00055450
	DO 150 I=1,NKW	00055460
	IC=IC+1	00055470
	150 A(IC)=AX(I)*FXN(IQ)+AY(I)*FYN(IQ)+AZ(I)*FZN(IQ)	00055480
C	COMBINE INFL OF PYLON	00055490
	160 IF(IDP.EQ.0) GO TO 180	00055500
	READ (22) (AX(I),AY(I),AZ(I),I=1,NKP),TUV,TVV,TWV	00055510
	BP(IX)=BETA*(BETA*TUV*FXN(IQ)+TVV*FYN(IQ)+TWV*FZN(IQ))	00055520
	DO 170 I=1,NKP	00055530
	IC=IC+1	00055540
	170 A(IC)=AX(I)*FXN(IQ)+AY(I)*FYN(IQ)+AZ(I)*FZN(IQ)	00055550
C	WRITE INFL MATRIX ON UNIT 12	00055560
	180 WRITE (12) (A(I),I=1,IC)	00055570
	190 CONTINUE	00055580
C		00055590
C*	CONTROL POINTS Q ON WING	00055600
	200 IF(IDW.EQ.0) GO TO 300	00055610
	DO 290 IQ=1,NQW	00055620
	IC=0	00055630
C	COMBINE INFL OF FUSELAGE	00055640
	IF(IDB.EQ.0) GO TO 230	00055650
	READ (19) (AX(I),AY(I),AZ(I),I=1,NKB)	00055660
	DO 220 I=1,NKB	00055670
	IC=IC+1	00055680
	220 A(IC)=-AZ(I)	00055690
C	COMBINE INFL OF FANPOD	00055700
	230 IF(IDF.EQ.0) GO TO 250	00055710
	READ (20) (AX(I),AY(I),AZ(I),I=1,NKF)	00055720
	DO 240 I=1,NKF	00055730
	IC=IC+1	00055740
	240 A(IC)=-AZ(I)	00055750
	250 IX=IX+1	00055760
C	COMBINE INFL OF WING	00055770
	READ (21) (AZ(I),I=1,NKW)	00055780
	DO 260 I=1,NKW	00055790
	IC=IC+1	00055800
	260 A(IC)=AZ(I)	00055810
C	COMBINE INFL OF PYLON	00055820
	IF(IDP.EQ.0) GO TO 280	00055830
	READ (22) (AX(I),AY(I),AZ(I),I=1,NKP),TUV,TVV,TWV	00055840
	BP(IX)=-TWV*BETA	00055850
	DO 270 I=1,NKP	00055860

	IC=IC+1	00055870
	270 A(IC)=-AZ(I)	00055880
C	WRITE INFL MATRIX ON UNIT 12	00055890
	280 WRITE (12) (A(I),I=1,IC)	00055900
	290 CONTINUE	00055910
C		00055920
C*	CONTROL POINTS Q ON PYLON	00055930
	300 IF(IDP.EQ.0) GO TO 400	00055940
	DO 390 IQ=1,NQP	00055950
	IC=0	00055960
C	COMBINE INFL OF FUSELAGE	00055970
	IF(IDB.EQ.0) GO TO 330	00055980
	READ (19) (AX(I),AY(I),AZ(I),I=1,NKB)	00055990
	DO 320 I=1,NKB	00056000
	IC=IC+1	00056010
	320 A(IC)=-AZ(I)	00056020
C	COMBINE INFL OF FANPOD	00056030
	330 IF(IDF.EQ.0) GO TO 350	00056040
	READ (20) (AX(I),AY(I),AZ(I),I=1,NKF)	00056050
	DO 340 I=1,NKF	00056060
	IC=IC+1	00056070
	340 A(IC)=-AZ(I)	00056080
	350 IX=IX+1	00056090
C	COMBINE INFL OF WING	00056100
	IF(IDW.EQ.0) GO TO 370	00056110
	READ (21) (AX(I),AY(I),AZ(I),I=1,NKW),TUV,TVV,TWV	00056120
	BW(IX)=-BETA*TWV	00056130
	DO 360 I=1,NKW	00056140
	IC=IC+1	00056150
	360 A(IC)=-AZ(I)	00056160
	370 CONTINUE	00056170
C	COMBINE INFL OF PYLON	00056180
	READ (22) (AX(I),AY(I),AZ(I),I=1,NKP),TUV,TVV,TWV	00056190
	DO 380 I=1,NKP BP(IX)=-BETA*TWV	00056200
	IC=IC+1	00056210
	380 A(IC)=AZ(I)	00056220
C	WRITE INFL MATRIX ON UNIT 12	00056230
	WRITE (12) (A(I),I=1,IC)	00056240
	390 CONTINUE	00056250
C		00056260
C*	CONTROL POINTS Q ON NACELLE	00056270
	400 IF(IDN.EQ.0) GO TO 500	00056280
C*	THE MINUS SIGN BEFORE COSAN IS DUE TO AY BEING IN THE	00056290

```

C*          PYLON COORD. SYS.. AY IS EQUAL TO -AY IN NACELLE SYS. 00056300
DO 490 IQ=1,NQN 00056310
IC=0 00056320
C          COMBINE INFL OF FUSELAGE 00056330
IF(IDB.EQ.0) GO TO 420 00056340
READ (19) (AX(I),AY(I),AZ(I),I=1,NKB) 00056350
DO 410 I=1,NKB 00056360
IC=IC+1 00056370
410 A(IC)=-SINAN(IQ)*AX(I)-COSAN(IQ)*AY(I) 00056380
C          COMBINE INFL OF FANPOD 00056390
420 IF(IDF.EQ.0) GO TO 440 00056400
READ (20) (AX(I),AY(I),AZ(I),I=1,NKF) 00056410
DO 430 I=1,NKF 00056420
IC=IC+1 00056430
430 A(IC)=-SINAN(IQ)*AX(I)-COSAN(IQ)*AY(I) 00056440
440 IX=IX+1 00056450
C          COMBINE INFL OF WING 00056460
IF(IDW.EQ.0) GO TO 460 00056470
READ (21) (AX(I),AY(I),AZ(I),I=1,NKW),TUV,TVV,TWV 00056480
BW(IX)=BETA*(-SINAN(IQ)*TUV*BETA-COSAN(IQ)*TVV) 00056490
DO 450 I=1,NKW 00056500
IC=IC+1 00056510
450 A(IC)=-SINAN(IQ)*AX(I)-COSAN(IQ)*AY(I) 00056520
C          COMBINE INFL OF PYLON 00056530
460 IF(IDP.EQ.0) GO TO 480 00056540
READ (22) (AX(I),AY(I),AZ(I),I=1,NKP),TUV,TVV,TWV 00056550
BP(IX)=BETA*(-SINAN(IQ)*TUV*BETA-COSAN(IQ)*TVV) 00056560
DO 470 I=1,NKP 00056570
IC=IC+1 00056580
470 A(IC)=-SINAN(IQ)*AX(I)-COSAN(IQ)*AY(I) 00056590
C          WRITE INFL MATRIX ON UNIT 12 00056600
480 WRITE (12) (A(I),I=1,IC) 00056610
490 CONTINUE 00056620
C 00056630
C*          REWIND I/O UNITS 00056640
500 REWIND 12 00056650
IF(IDB.NE.0) REWIND 19 00056660
IF(IDF.NE.0) REWIND 20 00056670
IF(IDW.NE.0) REWIND 21 00056680
IF(IDP.NE.0) REWIND 22 00056690
C 00056700
C*          SETUP ON UNIT 10 THE INFLUENCE MATRICES OF NACELLE 00056710
IF(IDN.EQ.0) GO TO 700 00056720

```

C	Q ON FUSELAGE	00056730
	IF (IDB.EQ.0) GO TO 530	00056740
	DO 520 IQ=1,NQB	00056750
	READ (9) (AX(I),AY(I),AZ(I),I=1,NKN)	00056760
	DO 510 I=1,NKN	00056770
	510 A(I)=AX(I)*BXN(IQ)+AY(I)*BYN(IQ)+AZ(I)*BZN(IQ)	00056780
	520 WRITE (10) (A(I),I=1,NKN)	00056790
C	Q ON FANPOD	00056800
	530 IF (IDF.EQ.0) GO TO 560	00056810
	DO 550 IQ=1,NQF	00056820
	READ (9) (AX(I),AY(I),AZ(I),I=1,NKN)	00056830
	DO 540 I=1,NKN	00056840
	540 A(I)=AX(I)*FXN(IQ)+AY(I)*FYN(IQ)+AZ(I)*FZN(IQ)	00056850
	550 WRITE (10) (A(I),I=1,NKN)	00056860
C	Q ON WING	00056870
	560 IF (IDW.EQ.0) GO TO 590	00056880
	DO 580 IQ=1,NQW	00056890
	READ (9) (AX(I),AY(I),AZ(I),I=1,NKN)	00056900
	DO 570 I=1,NKN	00056910
	570 A(I)=-AZ(I)	00056920
	580 WRITE (10) (A(I),I=1,NKN)	00056930
	590 IF (IDP.EQ.0) GO TO 620	00056940
	DO 610 IQ=1,NQP	00056950
C	Q ON PYLON	00056960
	READ (9) (AX(I),AY(I),AZ(I),I=1,NKN)	00056970
	DO 600 I=1,NKN	00056980
	600 A(I)=-AZ(I)	00056990
	610 WRITE (10) (A(I),I=1,NKN)	00057000
C	Q ON NACELLE	00057010
	620 DO 640 IQ=1,NQN	00057020
	READ (9) (AX(I),AY(I),AZ(I),I=1,NKN)	00057030
	DO 630 I=1,NKN	00057040
	630 A(I)=-SINAN(IQ)*AX(I)+COSAN(IQ)*AY(I)	00057050
	640 WRITE (10) (A(I),I=1,NKN)	00057060
C	REWIND 9	00057070
	REWIND 10	00057080
C	700 RETURN	00057090
	END	00057100
*DECK	MATB	00057110
	SUBROUTINE MATB	00057120
C		00057130
		00057140
		00057150

```

C*          CALC MATRIX-B                                00057160
C                                                    00057170
COMMON/DAT/ DA(5000)                                00057180
C                                                    00057190
EQUIVALENCE      (DA(6),FAA)      ,(DA(8),WAA)      ,(DA(10),WADI)    00057200
1,(DA(9),WAAI)    ,(DA(4006),BAA)                                     00057210
2,(DA(1960),DAF)  ,(DA(1961),PXCF) ,(DA(1962),ETF1) ,(DA(1963),ETF0) 00057220
3,(DA(1965),DAK)  ,(DA(1966),PXCK) ,(DA(1967),ETKI) ,(DA(1968),ETKO) 00057230
4,(DA(2504),PAA)  ,(DA(2505),PAAI) ,(DA(2506),PADI)                                     00057240
C                                                    00057250
COMMON/CRG/ PI,PI4,RC,BETA,SINDP,COSDP              00057260
COMMON/CFG/ IDB,IDF,IDW,IDP,IDN,NQB,NQF,NQW,NQP,NQN 00057270
COMMON/CPB/ IB1(9),SIBA,COBA                        00057280
COMMON/CPF/ IF1(9),SIFA,COFA                        00057290
COMMON/CPW/ IW1(13),NJC,NJS                         00057300
COMMON/CFK/ NDA                                      00057310
C                                                    00057320
LARGE            BT(1000)  ,BW(1000)  ,BP(1000)  ,DM1(5850)          00057330
1                ,BXN(650)  ,BYN(650)  ,BZN(650)  ,DM2(23050)        00057340
2                ,FXN(650)  ,FYN(650)  ,FZN(650)  ,DM3(16956)        00057350
3                ,BS(650)   ,EP(30)   ,XTLP(30)   ,XOCP(30)  ,DM4(26285) 00057360
4                ,BSP(360)  ,DM5(4330) ,SINAN(140)                                     00057370
C                                                    00057380
C*              START                                  00057390
C              SETUP SB DUE TO THICKNESS              00057400
C              NQT=NQB+NQF+NQW+NQP+NQN                00057410
C              DO 10 I=1,NQT                           00057420
10 BT(I)=BW(I)+BP(I)                                  00057430
C*              CALC MATRIX-B OF THE BODY              00057440
C              IF(IDB.EQ.0) GO TO 25                   00057450
C              BAAB=BAA/RC*BETA                        00057460
C              COBA=COS(BAAB)                          00057470
C              SIBA=SIN(BAAB)                          00057480
C              DO 20 I=1,NQB                            00057490
20 BT(I)=-COBA*BXN(I)-SIBA*BZN(I)-BT(I)              00057500
25 IS=NQB                                              00057510
C*              CALC MATRIX-B OF THE FANPOD           00057520
C              IF(IDF.EQ.0) GO TO 40                   00057530
C              FAAB=FAA/RC*BETA                        00057540
C              COFA=COS(FAAB)                          00057550
C              SIFA=SIN(FAAB)                          00057560
C              DO 30 I=1,NQF                            00057570
C              IS=IS+1                                  00057580

```

30	BT(IS)=-COFA*FXN(I)-SIFA*FZN(I)-BT(IS)	00057590
C*	CALC MATRIX-B OF THE WING	00057600
40	IF(IDW.EQ.0) GO TO 150	00057610
	WAAR=WAA*BETA/RC	00057620
	IF(WAAI) 70,70,90	00057630
70	DO 80 I=1,NQW	00057640
	IS=IS+1	00057650
	BS(I)=WAAR/BETA	00057660
80	BT(IS)=WAAR-BT(IS)	00057670
	GO TO 140	00057680
90	IF(WADI) 100,100,110	00057690
100	CALL WCAS	00057700
	GO TO 120	00057710
110	CALL WCZCS	00057720
120	DO 130 I=1,NQW	00057730
	IS=IS+1	00057740
	BS(I)=BS(I)+WAAR/BETA	00057750
130	BT(IS)=BS(I) *BETA -BT(IS)	00057760
140	CONTINUE	00057770
C*	ADD FLAP AND/OR KRUEGER DEFLECTION	00057780
	IF(NDA.EQ.0) GO TO 150	00057790
	IB=NQB+NQF	00057800
	AF=ATAN(BETA*TAN(DAF))	00057810
	BCDF=COS(AF)	00057820
	BSDF=SIN(AF)	00057830
	AK=ATAN(BETA*TAN(DAK))	00057840
	BCDK=COS(AK)	00057850
	BSDK=SIN(AK)	00057860
	DO 146 IS=1,NJS	00057870
	DO 144 IC=1,NJC	00057880
	XC=XOCP(IC)	00057890
	IB=IB+1	00057900
	IF(DAF.EQ.0.0) GO TO 142	00057910
	IF(EP(IS).LT.ETFI) GO TO 142	00057920
	IF(EP(IS).GT.ETFO) GO TO 142	00057930
	IF(XC.LE.PXCF) GO TO 142	00057940
	BT(IB)=BT(IB)*BCDF+BSDF	00057950
142	IF(DAK.EQ.0.0) GO TO 144	00057960
	IF(EP(IS).LT.ETKI) GO TO 144	00057970
	IF(EP(IS).GT.ETKO) GO TO 144	00057980
	IF(XC.GE.PXCK) GO TO 144	00057990
	BT(IB)=BT(IB)*BCDK-BSDK	00058000
144	CONTINUE	00058010


```

146 CONTINUE                                00058020
150 IF(IDP.EQ.0) GO TO 240                  00058030
C*          CALC MATRIX-B OF THE PYLON      00058040
  PAAR=PAA*BETA/RC*SINDP                    00058050
  IF(PAAI) 170,170,190                      00058060
170 DO 180 I=1,NQP                          00058070
  IS=IS+1                                    00058080
  BSP(I)=PAAR/BETA                          00058090
180 BT(IS)=PAAR-BT(IS)                      00058100
  GO TO 240                                  00058110
190 IF(PADI) 200,200,210                   00058120
200 CALL PCAS                                00058130
  GO TO 220                                  00058140
210 CALL PCZCS                              00058150
220 DO 230 I=1,NQP                          00058160
  IS=IS+1                                    00058170
  BSP(I)=BSP(I)+PAAR/BETA                   00058180
230 BT(IS)=BSP(I) *BETA -BT(IS)            00058190
240 IF(IDN.EQ.0) GO TO 280                  00058200
C*          CALC MATRIX-B OF THE NACELLE   00058210
  DO 250 I=1,NQN                            00058220
  IS=IS+1                                    00058230
250 BT(IS)=SINAN(I)-BT(IS)                 00058240
C*          PRINT MATRIX-B                 00058250
280 IF(DA(12).LT.0.0) GO TO 300             00058260
  WRITE (6,290) (BT(IS),IS=1,NQT)          00058270
290 FORMAT (12H1 * MATRIX-B/(1H 10F10.5))  00058280
300 RETURN                                   00058290
  END                                         00058300
*DECK WCAS                                  00058310
  SUBROUTINE WCAS                            00058320
C                                             00058330
C  SUBR. TLU TABLE LOOKUP FOR THE LOCAL ANGLE OF ATTACK AT SPECIFIED 00058340
C  ETA AND X/C.                              00058350
C                                             00058360
  COMMON/DAT/ DA(5000)                      00058370
C                                             00058380
  DIMENSION BI(1) ,WXC(24) ,WET(19) ,WJC(30) ,WJS(30) 00058390
  1 ,TWI(19)                                00058400
C                                             00058410
  EQUIVALENCE (DA(706),WXC) ,(DA(731),WET) ,(DA(1660),WJC) 00058420
  1,(DA(1690),WJS) ,(DA(1970),TWI) ,(DA(1270),WNVC) 00058430
  2,(DA(750),BI)                            00058440

```

270

C			00058450
	COMMON/CPW/	DW1(13),NJC,NJS	00058460
C			00058470
	LARGE	BOU(3000) ,FUS(25000),FAN(23700)	00058480
	1	,EV(52) ,DMI(1004) ,RS(650) ,DM2(60) ,XOCP(30)	00058490
C			00058500
C		START	00058510
		NXW=DA(705)	00058520
		NEW=DA(730)	00058530
		IB=0	00058540
		DO 620 IJS=1,NJS	00058550
		IE=WJS(IJS)	00058560
		IF(EV(IE)-WET(1)) 510,510,515	00058570
510		J=1	00058580
		GO TO 530	00058590
515		DO 520 NC=2,NEW	00058600
		J=NC	00058610
		IF(EV(IE)-WET(J)) 540,530,520	00058620
520		CONTINUE	00058630
530		RATY=0.0	00058640
		GO TO 550	00058650
540		RATY=(WET(J)-EV(IE))/(WET(J)-WET(J-1))	00058660
550		TWB=TWI(J)-RATY*(TWI(J)-TWI(J-1))	00058670
		DO 610 IJC=1,NJC	00058680
		XC=XOCP(IJC)	00058690
		IF(XC-WXC(1)) 560,560,565	00058700
560		I=1	00058710
		GO TO 580	00058720
565		DO 570 NC=2,NXW	00058730
		I=NC	00058740
		IF(XC-WXC(I)) 590,580,570	00058750
570		CONTINUE	00058760
580		RATX=0.0	00058770
		GO TO 600	00058780
590		RATX=(WXC(I)-XC)/(WXC(I)-WXC(I-1))	00058790
600		NPX=I+NXW*(J-1)	00058800
		AX=BI(NPX)-RATX*(BI(NPX)-BI(NPX-1))	00058810
		NPX=NPX-NXW	00058820
		BX=BI(NPX)-RATX*(BI(NPX)-BI(NPX-1))	00058830
		IB=IB+1	00058840
610		BS(IB)=AX-RATY*(AX-BX)+TWB	00058850
620		CONTINUE	00058860
		RETURN	00058870

271

```

END 00058880
*DECK WCZCS 00058890
SUBROUTINE WCZCS 00058900
C 00058910
C* CALC. LOCAL ANGLES OF ATTACK FROM DEFLECTION INPUT(L/C) 00058920
C 00058930
COMMON/DAT/ DA(5000) 00058940
C 00058950
DIMENSION WXC(24) ,WET(19) ,WAD(456) ,TWI(19) ,WJC(30) 00058960
1 ,WJS(30) 00058970
C 00058980
EQUIVALENCE (DA(706),WXC) ,(DA(731),WET) ,(DA(750),WAD) 00058990
1,(DA(1970),TWI) ,(DA(1660),WJC) ,(DA(1690),WJS) ,(DA(1270),WNVC) 00059000
C 00059010
COMMON/CPW/ DW1(13),NJC,NJS 00059020
C 00059030
LARGE BOU(3000) ,FUS(25000) ,FAN(23700) 00059040
1 ,DMI(996) ,WPT(60) ,BS(650) 00059050
2 ,EP(30) ,XTLP(30) ,XOCP(30) 00059060
3 ,DM2(21634) ,PYL(6061) ,CEL(3700) 00059070
4 ,XLP(19) ,CHD(19) ,ZI(24) 00059080
5 ,XI(24) ,XO(48) ,ZO(48) ,Z(48,19) 00059090
6 ,WEL(20) 00059100
C 00059110
C SET FOR CODIM AND CURVE FIT FOR DEFLECTIONS (Z) 00059120
C 00059130
NXW=DA(705) 00059140
NEW=DA(730) 00059150
C SHIFT TO LCM FOR WINGD 00059160
DO 10 I=1,NEW 00059170
10 WEL(I)=WET(I) 00059180
CALL WINGD (NEW,WEL,XLP,CHD,WPT) 00059190
LD=0 00059200
DXI=.5*(1.-COS(1.5708/WNVC)) 00059210
DO 100 IE=1,NEW 00059220
DO 20 IX=1,NXW 00059230
LD=LD+1 00059240
ZI(IX)=WAD(LD)*CHD(IE) 00059250
20 XI(IX)=WXC(IX) 00059260
IC=0 00059270
DO 40 IJC=1,NJC 00059280
IC=IC+2 00059290
XO(IC-1)=XOCP(IJC)-DXI 00059300
40 XO(IC)=XOCP(IJC)+DXI

```

	CALL CODIM (XI,ZI,NXW,XO,ZO,IC)	00059310
	DO 60 IX=1,IC	00059320
	60 Z(IX,IE)=ZO(IX)	00059330
	100 CONTINUE	00059340
C	LINEAR INTERPOLATE FOR Z SPANWISE AND CALC SLOPE	00059350
	KK=0	00059360
	DXT=DXI+DXI	00059370
	DO 200 IE=1,NJS	00059380
	DX=DXT*XTLP(IE)	00059390
	IF(EP(IE).GT.WET(1)) GO TO 120	00059400
	II=1	00059410
	GO TO 140	00059420
120	DO 130 IS=2,NEW	00059430
	II=IS	00059440
	IF(EP(IE)-WET(IS)) 150,140,130	00059450
130	CONTINUE	00059460
140	RATY=0.0	00059470
	GO TO 160	00059480
150	RATY=(WET(II)-EP(IE))/(WET(II)-WET(II-1))	00059490
160	TWB=TWI(II)-RATY*(TWI(II)-TWI(II-1))	00059500
	IC=0	00059510
	DO 170 IJC=1,NJC	00059520
	IC=IC+1	00059530
	Z1=Z(IC,II)-RATY*(Z(IC,II)-Z(IC,II-1))	00059540
	IC=IC+1	00059550
	Z2=Z(IC,II)-RATY*(Z(IC,II)-Z(IC,II-1))	00059560
	KK=KK+1	00059570
170	BS(KK)=(Z1-Z2)/DX+TWB	00059580
200	CONTINUE	00059590
	RETURN	00059600
	END	00059610
*DECK	PCAS	00059620
	SUBROUTINE PCAS	00059630
C		00059640
C	SUBR. TLU TABLE LOOKUP FOR THE LOCAL ANGLE OF ATTACK AT SPECIFIED	00059650
C	ETA AND X/C.	00059660
C		00059670
C	COMMON/DAT/ DA(5000)	00059680
C		00059690
C	DIMENSION BI(361) ,PXC(19) ,PET(19) ,PJC(18) ,TWI(19)	00059700
C		00059710
C	EQUIVALENCE (DA(2531),PXC) ,(DA(2551),PET) ,(DA(2512),PJC)	00059720
1	,(DA(2940),TWI) ,(DA(2507),PNVC),(DA(2570),BI)	00059730

C				00059740
	COMMON/CPP/	DP1(8) ,NVS ,NJC		00059750
C				00059760
	LARGE	DM1(75130) ,EV(22) ,DM2(1483) ,CFJ(20)		00059770
	I	,DM3(3126) ,BS(360)		00059780
C				00059790
C		START		00059800
		NXW=DA(2530)		00059810
		NEW=DA(2550)		00059820
		IB=0		00059830
		DO 620 IE=1 ,NVS		00059840
		IF(EV(IE)-PET(1)) 510,510,515		00059850
510		J=1		00059860
		GO TO 530		00059870
515		DO 520 NC=2 ,NEW		00059880
		J=NC		00059890
		IF(EV(IE)-PET(J)) 540,530,520		00059900
520		CONTINUE		00059910
530		RATY=0.0		00059920
		GO TO 550		00059930
540		RATY=(PET(J)-EV(IE))/(PET(J)-PET(J-1))		00059940
550		TWB=TWI(J)-RATY*(TWI(J)-TWI(J-1))		00059950
		DO 610 IJC=1 ,NJC		00059960
		IJ=PJC(IJC)+1.0		00059970
		XC=CFJ(IJ)		00059980
		IF(XC-PXC(1)) 560,560,565		00059990
560		I=1		00060000
		GO TO 580		00060010
565		DO 570 NC=2 ,NXW		00060020
		I=NC		00060030
		IF(XC-PXC(I)) 590,580,570		00060040
570		CONTINUE		00060050
580		RATX=0.0		00060060
		GO TO 600		00060070
590		RATX=(PXC(I)-XC)/(PXC(I)-PXC(I-1))		00060080
600		NPX=I+NXW*(J-1)		00060090
		AX=BI(NPX)-RATX*(BI(NPX)-BI(NPX-1))		00060100
		NPX=NPX-NXW		00060110
		BX=BI(NPX)-RATX*(BI(NPX)-BI(NPX-1))		00060120
		IB=IB+1		00060130
610		BS(IB)=AX-RATY*(AX-BX)+TWB		00060140
620		CONTINUE		00060150
		RETURN		00060160

```

END 00060170
*DECK PCZCS 00060180
SUBROUTINE PCZCS 00060190
C 00060200
C* CALC. LOCAL ANGLES OF ATTACK FROM DEFLECTION INPUT(Z/C) 00060210
C 00060220
COMMON/DAT/ DA(5000) 00060230
C 00060240
DIMENSION PXC(19) ,PET(19) ,PAD(361) ,TWI(19) ,PJC(18) 00060250
C 00060260
EQUIVALENCE (DA(2531),PXC) ,(DA(2551),PET) ,(DA(2570),PAD) 00060270
1 ,(DA(2940),TWI) ,(DA(2507),PNVC) ,(DA(2512),PJC) 00060280
C 00060290
COMMON/ CPP/ DP1(8),NVS,NJC 00060300
C 00060310
LARGE DM1(75130) 00060320
1 ,EV(22) ,XLV(22) ,XTLV(22) ,WPE(39) 00060330
2 ,DP2(1400) ,CFJ(20) ,DP3(3126) ,BS(360) 00060340
3 ,DM2(4750) ,XLP(19) ,CHD(19) ,ZI(24) 00060350
4 ,XI(24) ,XO(48) ,ZO(48) ,Z(48,19) 00060360
5 ,PEL(20) 00060370
C 00060380
C SET FOR CODIM AND CURVE FIT FOR DEFLECTIONS 00060390
LD=0 00060400
NXW=DA(2530) 00060410
NEW=DA(2550) 00060420
C SHIFT TO LCM FOR WINGD 00060430
DO 10 I=1,NEW 00060440
10 PEL(I)=PET(I) 00060450
C 00060460
CALL WINGD (NEW,PEL,XLP,CHD,WPE) 00060470
DXI=.5*(1.-COS(1.5708/PNVC)) 00060480
DO 100 IE=1,NEW 00060490
DO 20 IX=1,NXW 00060500
LD=LD+1 00060510
ZI(IX)=PAD(LD)*CHD(IE) 00060520
20 XI(IX)=PXC(IX) 00060530
IC=0 00060540
DO 40 IJC=1,NJC 00060550
IC=IC+2 00060560
IJ=PJC(IJC)+1.0 00060570
XO(IC-1)=CFJ(IJ)-DXI 00060580
40 XO(IC)=CFJ(IJ)+DXI 00060590

```

	CALL CODIM (XI,ZI,NXW,XO,ZO,IC)	00060600
	DO 60 IX=1,IC	00060610
	60 Z(IX,IE)=ZO(IX)	00060620
	100 CONTINUE	00060630
C	LINEAR INTERPOLATE FOR Z SPANWISE AND CALC SLOPE	00060640
	KK=0	00060650
	DXT=DXI+DXI	00060660
	DO 200 IE=1,NVS	00060670
	DX=DXT*XTLV(IE)	00060680
	IF(EV(IE).GT.PET(1)) GO TO 120	00060690
	II=1	00060700
	GO TO 140	00060710
120	DO 130 IS=2,NEW	00060720
	II=IS	00060730
	IF(EV(IE)-PET(IS)) 150,140,130	00060740
130	CONTINUE	00060750
140	RATY=0.0	00060760
	GO TO 160	00060770
150	RATY=(PET(II)-EV(IE))/(PET(II)-PET(II-1))	00060780
160	TWB=TWI(II)-RATY*(TWI(II)-TWI(II-1))	00060790
	IC=0	00060800
	DO 170 IJC=1,NJC	00060810
	IC=IC+1	00060820
	Z1=Z(IC,II)-RATY*(Z(IC,II)-Z(IC,II-1))	00060830
	IC=IC+1	00060840
	Z2=Z(IC,II)-RATY*(Z(IC,II)-Z(IC,II-1))	00060850
	KK=KK+1	00060860
170	BS(KK)=(Z1-Z2)/DX+TWB	00060870
200	CONTINUE	00060880
	RETURN	00060890
	END	00060900
*DECK	SOLU	00060910
	SUBROUTINE SOLU	00060920
C		00060930
C*	SOLVE LINEAR SIMULTANEOUS EQUATIONS (AX=B)	00060940
C		00060950
	COMMON/DAT/ DA(5000)	00060960
C		00060970
	COMMON/CRG/ PI,PI4,RC,BETA	00060980
	COMMON/CFG/ IDB,IDF,IDW,IDP,IDN,NQB,NQF,NQW,NQP,NQN	00060990
1	,NKB,NKF,NKW,NKP,NKN	00061000
C		00061010
	LARGE B(1000) ,DM1(90812),A(1000) ,AR(590)	00061020

C		00061030
C		00061040
	L=NKN	00061050
	M=NQB+NQF+NQW+NQP	00061060
	N=NKB+NKF+NKW+NKP	00061070
	NQT=M+L	00061080
	NKT=N+L	00061090
	IF(L.NE.0) GO TO 10	00061100
C*	NO NACELLE	00061110
	CALL MSOL (N,M,12)	00061120
	GO TO 30	00061130
10	IF(N.NE.0) GO TO 20	00061140
C*	NACELLE ONLY	00061150
	CALL MSOL (L,L,10)	00061160
	GO TO 30	00061170
C*	WITH NACELLE	00061180
20	CALL PARTM (L,M,N)	00061190
C*	PRINT LEAST SQUARE MATRIX-B	00061200
30	IF(DA(12).EQ.0.0) GO TO 110	00061210
	L1=N+1	00061220
	DO 50 K=1,NQT	00061230
	A(K)=0.0	00061240
	IF(N.NE.0) READ (12) (AR(I),I=1,N)	00061250
	IF(L.NE.0) READ (10) (AR(I),I=L1,NKT)	00061260
	DO 50 I=1,NKT	00061270
	50 A(K)=A(K)+B(I)*AR(I)	00061280
C		00061290
	IF(L.NE.0) REWIND 10	00061300
	IF(N.NE.0) REWIND 12	00061310
C		00061320
	WRITE (6,100) (A(K),K=1,NQT)	00061330
100	FORMAT (26H1 ** LEAST SQUARE MATRIX-B/(1H 10F10.5))	00061340
C		00061350
C*	COMPRESS. EFFECT ON COEFFICIENTS	00061360
110	IF(BETA.EQ.1.) GO TO 140	00061370
	DO 120 I=1,NKT	00061380
120	B(I)=B(I)/BETA	00061390
C		00061400
140	RETURN	00061410
	END	00061420
*DECK	MSOL	00061430
	SUBROUTINE MSOL (NKX,NQX,MT)	00061440
C		00061450

C*	SOLVES AX=B FOR X, GIVEN A AND B	00061460
C		00061470
	COMMON/DAT/ DA(5000)	00061480
	DIMENSION WJC(1) ,PJC(1)	00061490
C		00061500
	EQUIVALENCE (DA(14),WSJC) ,(DA(1277),WNJC),(DA(1660),WJC)	00061510
1	,(DA(2510),PSJC) ,(DA(2511),PNJC),(DA(2512),PJC)	00061520
C		00061530
	COMMON/CFG/ IDB,IDF,IDW,IDP,IDN,NQB,NQF,NQW,NQP,NQN	00061540
C		00061550
	LARGE B(1000) ,DM1(90812)	00061560
	LARGE A(101975) ,AR(591) ,IXC(450) ,IL(450)	00061570
C		00061580
C		00061590
	NJC=WNJC	00061600
	NJCP=PNJC	00061610
	NKTP=NKX+1	00061620
	N=(NKX*(NKX+3))/2	00061630
	DO 10 I=1,N	00061640
10	A(I)=0.0	00061650
	NKT2=NKX+2	00061660
	NK1=NQB+NQF+1	00061670
	NK2=NK1+NQW	00061680
	NK3=NK2+NQP	00061690
	JC=0	00061700
	JCP=0	00061710
	DO 60 K=1,NQX	00061720
	READ (MT) (AR(L),L=1,NKX)	00061730
	IF(IDW.EQ.0) GO TO 32	00061740
	IF(K.LT.NK1) GO TO 32	00061750
	IF(K.GE.NK2) GO TO 32	00061760
	IF(JC.GE.NJC) JC=0	00061770
	JC=JC+1	00061780
	IF(WSJC.GT.WJC(JC)) GO TO 60	00061790
	GO TO 36	00061800
32	IF(IDP.EQ.0) GO TO 36	00061810
	IF(K.LT.NK2) GO TO 36	00061820
	IF(K.GE.NK3) GO TO 36	00061830
	IF(JCP.GE.NJCP) JCP=0	00061840
	JCP=JCP+1	00061850
	IF(PSJC.GT.PJC(JCP)) GO TO 60	00061860
36	AR(NKX+1)=B(K)	00061870
	IXI=1	00061880

DO 50 I=1,NKX	00061890
R=SQRT(A(IXI)**2+AR(I)**2)	00061900
IF(R.EQ.0.0) GO TO 50	00061910
C=A(IXI)/R	00061920
S=AR(I)/R	00061930
IXJ=IXI	00061940
DO 40 J=I,NKTP	00061950
T2=C*A(IXJ)+S*AR(J)	00061960
AR(J)=-S*A(IXJ)+C*AR(J)	00061970
A(IXJ)=T2	00061980
40 IXJ=IXJ+1	00061990
50 IXI=IXI+NKT2-I	00062000
60 CONTINUE	00062010
REWIND MT	00062020
II=1	00062030
IXI=1	00062040
DO 80 I=1,NKX	00062050
IXC(II)=IXI	00062060
B(II)=0.0	00062070
IL(II)=0	00062080
IF(A(IXI).LE.0.0000001) GO TO 80	00062090
IL(II)=II	00062100
II=II+1	00062110
80 IXI=IXI+NKT2-I	00062120
II=NKX	00062130
DO 210 J=1,NKX	00062140
IF(IL(II).LE.0) GO TO 210	00062150
JI=IL(II)	00062160
JS=IXC(JI)-JI	00062170
JXI=JS+II	00062180
JXN=JS+NKTP	00062190
IF(II-NKX) 170,200,220	00062200
170 IK=II+1	00062210
JXK=JXI	00062220
DO 180 K=IK,NKX	00062230
JXK=JXK+1	00062240
180 B(II)=B(II)-A(JXK)*B(K)	00062250
200 B(II)=(B(II)+A(JXN))/A(JXI)	00062260
210 II=II-1	00062270
220 CONTINUE	00062280
RETURN	00062290
END	00062300
*DECK MSOLP	00062310

	SUBROUTINE MSOLP(NKT,NQT,NCB,M)	00062320
C		00062330
C*	HOUSEHOLDER METHOD FOR SOLVING SET OF LINEAR	00062340
C*	SIMULTANEOUS EQUATIONS	00062350
C		00062360
	LARGE B(1000) ,DMI(90812)	00062370
	LARGE A(101975) ,AR(591) ,KRR(450) ,IL(450)	00062380
C		00062390
C		00062400
	N1=NCB-1	00062410
	KR=0	00062420
	NKTP=NKT+NCB	00062430
	DO 10 J=1,NKT	00062440
	KRR(J)=KR+J-1	00062450
	KR=KRR(J)	00062460
	DO 10 K=J,NKTP	00062470
	LL=(J-1)*NKTP+K-KRR(J)	00062480
10	A(LL)=0.0	00062490
	DO 60 K=1,NQT	00062500
	READ (10) (AR(I),I=1,NKT)	00062510
	KR=NKT+1	00062520
	LL=NKT+N1	00062530
	READ (12) (AR(I),I=KR,LL)	00062540
	AR(NKT+NCB)=B(M+K)	00062550
C		00062560
30	DO 50 I=1,NKT	00062570
	LL=(I-1)*NKTP+I-KRR(I)	00062580
	R=SQRT(A(LL)**2+AR(I)**2)	00062590
	IF(R .EQ. 0.0) GO TO 50	00062600
	C=A(LL)/R	00062610
	S = AR(I) / R	00062620
	DO 40 J=I,NKTP	00062630
	LL=(I-1)*NKTP+J-KRR(I)	00062640
	T2=C*A(LL)+S*AR(J)	00062650
	AR(J)=-S*A(LL)+C*AR(J)	00062660
40	A(LL)=T2	00062670
50	CONTINUE	00062680
60	CONTINUE	00062690
	REWIND 10	00062700
	REWIND 12	00062710
	II = 1	00062720
	DO 80 I=1,NKT	00062730
	LL=(I-1)*NKTP+I-KRR(I)	00062740

	IF(A(LL)-0.0000001)70,92,92	00062750
92	CONTINUE	00062760
	IL(I) = II	00062770
	II = II + 1	00062780
	GO TO 80	00062790
70	IL(I) = 0	00062800
80	CONTINUE	00062810
	DO 250 J=1,NCB	00062820
	NKTJ=NKT+J	00062830
	DO 90 I=1,NKT	00062840
90	AR(I)=0.0	00062850
	II=NKT	00062860
	DO 210 I=1,NKT	00062870
	IF(IL(II) .LE. 0) GO TO 210	00062880
	J1 = IL(II)	00062890
	LX=(J1-1)*NKTP+NKTJ-KRR(J1)	00062900
	LM=(J1-1)*NKTP+II-KRR(J1)	00062910
	IF(II - NKT) 170, 200, 220	00062920
170	IK = II + 1	00062930
	DO 180 K=IK,NKT	00062940
	LL=(J1-1)*NKTP+K-KRR(J1)	00062950
180	AR(II)=AR(II)-A(LL)*AR(K)	00062960
200	AR(II)=(AR(II)+A(LX))/A(LM)	00062970
210	II = II - 1	00062980
220	CONTINUE	00062990
	WRITE (16) (AR(I),I=1,NKT)	00063000
250	CONTINUE	00063010
	REWIND 16	00063020
	RETURN	00063030
	END	00063040
*DECK	PARTM	00063050
	SUBROUTINE PARTM(L,M,N)	00063060
C		00063070
C*	PARTISION-HOUSEHOLDER MATRIX SOLUTION	00063080
C		00063090
	LARGE B(1000) ,DM1(90812)	00063100
	LARGE A(38835),SAVE(140,451),AR(591) ,A11(450) ,IL(450)	00063110
C	SIZE OF SAVE=(L,N+1)	00063120
C		00063130
	NCB=N+1	00063140
	DO 10 I=1,M	00063150
	READ (10)	00063160
10	READ (12)	00063170

CALL MSOLP (L,L,NCB,M)	00063180
DO 20 I=1,NCB	00063190
20 READ (16) (SAVE(J,I),J=1,L)	00063200
REWIND 16	00063210
DO 70 I=1,M	00063220
READ (10) (A11(LL),LL=1,L)	00063230
DO 50 J=1,NCB	00063240
AR(J)=0.0	00063250
DO 40 K=1,L	00063260
40 AR(J)=AR(J)+A11(K)*SAVE(K,J)	00063270
50 CONTINUE	00063280
READ (12) (A11(LL),LL=1,N)	00063290
DO 60 J=1,N	00063300
60 A11(J)=AR(J)-A11(J)	00063310
WRITE (11) (A11(J),J=1,N)	00063320
70 B(I)=AR(N+1)-B(I)	00063330
REWIND 10	00063340
REWIND 11	00063350
REWIND 12	00063360
CALL MSOL (N,M,11)	00063370
DO 80 I=1,NCB	00063380
80 READ (16) (SAVE(J,I),J=1,L)	00063390
REWIND 16	00063400
DO 100 K=1,L	00063410
KK=N+K	00063420
TEMP=0.0	00063430
DO 90 J=1,N	00063440
90 TEMP=TEMP+B(J)*SAVE(K,J)	00063450
100 B(KK)=SAVE(K,NCB)-TEMP	00063460
RETURN	00063470
END	00063480
*DECK PLCAL	00063490
SUBROUTINE PLCAL	00063500
C	00063510
C* CALC PRESSURE COEFFICIENTS AND INTEGRATED LOADS	00063520
C	00063530
COMMON/CFG/ IDB, IDF, IDW, IDP	00063540
C	00063550
C INITIAL	00063560
BCL=0.0	00063570
BCD=0.0	00063580
BCM=0.0	00063590
FCL=0.0	00063600

	FCD=0.0	00063610
	FCM=0.0	00063620
	WCL=0.0	00063630
	WCD=0.0	00063640
	WCM=0.0	00063650
	PCL=0.0	00063660
	PCD=0.0	00063670
	PCM=0.0	00063680
C	FUSELAGE	00063690
	IF(IDB.EQ.0) GO TO 10	00063700
	CALL BVRCP	00063710
	CALL BLOAD (BCL,BCD,BCM)	00063720
C	FANPOD	00063730
	10 IF(IDF.EQ.0) GO TO 20	00063740
	CALL FVRCP	00063750
	CALL FLOAD (FCL,FCD,FCM)	00063760
C	WING	00063770
	20 IF(IDW.EQ.0) GO TO 30	00063780
	CALL WDPQ	00063790
	CALL WVAP	00063800
	CALL WLOAD (WCL,WCD,WCM)	00063810
C	PYLON	00063820
	30 IF(IDP.EQ.0) GO TO 40	00063830
	CALL PDPQ	00063840
	CALL PVAP	00063850
	CALL PLOAD (PCL,PCD,PCM)	00063860
C	SUM AND PRINT	00063870
	40 TCL=BCL+FCL+WCL+PCL	00063880
	TCD=BCD+FCD+WCD+PCD	00063890
	TCM=BCM+FCM+WCM+PCM	00063910
	TXC=-TCM/TCL	00063920
C	WRITE (6,50) TCL,TCD,TCM,TXC	00063930
	50 FORMAT (1H1,20X,36H ** TOTAL CONFIGURATION PARAMETERS */1H0,40X,	00063940
	1 ,2HCL,13X,2HCD,13X,2HCM,13X,3HX/C/1H ,30X,4F15.5)	00063950
C	IF(IDW.EQ.0) GO TO 70	00063960
	CALL IDRAG (BCL,FCL,CDI)	00063970
	WRITE (6,60) CDI	00063980
	60 FORMAT (1H0,20X,18H ** INDUCED DRAG =,1F10.5)	00063990
C	REWIND I/O	00064000
	70 IF(IDB.NE.0) REWIND 19	00064010
		00064020
		00064030

```

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(4731),FVX)    ,(DA(4905),FJX)
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/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              ,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
I=NE5+3
XF(I,NUI+1)=0.0

```

```

00064040
00064050
00064060
00064070
00064080
00064090
00064100
00064110
00064120
00064130
00064140
00064150
00064160
00064170
00064180
00064190
00064200
00064210
00064220
00064230
00064240
00064250
00064260
00064270
00064280
00064290
00064300
00064310
00064320
00064330
00064340
00064350
00064360
00064370
00064380
00064390
00064400
00064410
00064420
00064430
00064440
00064450
00064460

```

	CALL SFC (I,EV)	00064470
	XKV=WP2*(XF(I,1)-XF(I,2)/2.+XF(I,NUT+1)*SINTF+XF(I,NU)*SINTK)	00064480
C*	CALC CONTRIBUTION TO FLOW FROM ADJACENT VORTICES	00064490
30	NC=0	00064500
	DO 36 IY=1,NVY	00064510
	DO 36 IJ=1,NJX	00064520
	J=FJX(IJ)	00064530
	K=J+1	00064540
	IF(J.EQ.NVXM) K=J	00064550
	NC=NC+1	00064560
	KC=(IY-1)*NF	00064570
	VMV(NC)=0.0	00064580
	DO 35 I=1,NF	00064590
	KC=KC+1	00064600
35	VMV(NC)=CKV(KC)*(PM(J,I)+PM(K,I))+VMV(NC)	00064610
36	VMV(NC)=VMV(NC)/4.	00064620
	IX=1	00064630
	DO 40 IJ=1,NJX	00064640
	JX=FJX(IJ)	00064650
37	IX=IX+1	00064660
	IF(IX.GE.JX) GO TO 38	00064670
	READ (23)	00064680
	GO TO 37	00064690
38	READ (23) ((PLM(J,I),J=1,NVY),I=1,NF)	00064700
	NC=IJ	00064710
	KC=0	00064720
	DO 40 J=1,NVY	00064730
	CK(NC)=0.0	00064740
	DO 39 I=1,NF	00064750
	KC=KC+1	00064760
39	CK(NC)=CKV(KC)*PLM(J,I)+CK(NC)	00064770
40	NC=NC+NJX	00064780
	REWIND 23	00064790
	DO 50 I=1,NJX	00064800
50	XL(I)=YQ(I)+YQ(I)	00064810
	N1=NJX	00064820
	DO 53 J=2,NVY	00064830
	DO 53 I=1,NJX	00064840
	N1=N1+1	00064850
	N2=N1-NJX	00064860
53	XL(N1)=SQRT((XQ(N2)-XQ(N1))**2+(YQ(N2)-YQ(N1))**2+(ZQ(N2)-ZQ(N1))	00064870
	1 **2)	00064880
	NC=0	00064890


```

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 VIV(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)+(ITX(J)*COSA*BETA+ITZ(J)*SINA)
VMV(J)=VMV(J)+(IMX(J)*COSA*BETA+IMZ(J)*SINA)
DO 60 I=1,NKB
VTV(J)=(ITX(J)*AX(I)+ITY(J)*AY(I)+ITZ(J)*AZ(I))*CKV(I)+VTV(J)
60 VMV(J)=(IMX(J)*AX(I)+IMY(J)*AY(I)+IMZ(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)=(ITX(J)*AX(I)+ITY(J)*AY(I)+ITZ(J)*AZ(I))*CKV(II)+VTV(J)

```

```

00064900
00064910
00064920
00064930
00064940
00064950
00064960
00064970
00064980
00064990
00065000
00065010
00065020
00065030
00065040
00065050
00065060
00065070
00065080
00065090
00065100
00065110
00065120
00065130
00065140
00065150
00065160
00065170
00065180
00065190
00065200
00065210
00065220
00065230
00065240
00065250
00065260
00065270
00065280
00065290
00065300
00065310
00065320

```

236

```

61 VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(II)+VMV(J)      00065330
C
62 IF(IDW.EQ.0) GO TO 65                                                    00065340
   DO 64 J=1,NQB                                                            00065350
   READ(21) (AX(IS),AY(IS),AZ(IS),IS=1,NKW),UV,VV,WV                       00065360
   IF(DA(2000).EQ.0.0) GO TO 63                                            00065370
   UB=UV*BETA                                                                00065380
   VTV(J)=TTX(J)*UB+TTY(J)*VV+TTZ(J)*WV+VTV(J)                          00065390
   VMV(J)=TMX(J)*UB+TMY(J)*VV+TMZ(J)*WV+VMV(J)                          00065400
63 II=NKB+NKF                                                                00065410
   DO 64 I=1,NKW                                                            00065420
   II=II+1                                                                    00065430
   VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(II)+VTV(J)      00065440
64 VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(II)+VMV(J)      00065450
65 IF(IDP.EQ.0) GO TO 72                                                    00065460
   DO 70 J=1,NQB                                                            00065470
   READ (22) (AX(IS),AY(IS),AZ(IS),IS=1,NKP),UV,VV,WV                     00065480
   IF(DA(3000).EQ.0.0) GO TO 66                                            00065490
   UB=UV*BETA                                                                00065500
   VTV(J)=TTX(J)*UB+TTY(J)*VV+TTZ(J)*WV+VTV(J)                          00065510
   VMV(J)=TMX(J)*UB+TMY(J)*VV+TMZ(J)*WV+VMV(J)                          00065520
66 II=NKB+NKF+NKW                                                            00065530
   DO 70 I=1,NKP                                                            00065540
   II=II+1                                                                    00065550
   VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(II)+VTV(J)      00065560
70 VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(II)+VMV(J)      00065570
72 IF(IDN.EQ.0) GO TO 80                                                    00065580
   DO 74 J=1,NQB                                                            00065590
   READ (9) (AX(IS),AY(IS),AZ(IS),IS=1,NKN)                                00065600
   II=NKB+NKF+NKW+NKP                                                       00065610
   DO 74 I=1,NKN                                                            00065620
   II=II+1                                                                    00065630
   VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(II)+VTV(J)      00065640
74 VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(II)+VMV(J)      00065650
C
80 DO 89 J=1,NQB                                                            00065660
   IF(XMACH.LE.0.0) GO TO 88                                                00065670
   VTV(J)=VTV(J)/SQRT((TTX(J)*BETA)**2+TTY(J)**2+TTZ(J)**2)              00065680
   VMV(J)=VMV(J)/SQRT((TMX(J)*BETA)**2+TMY(J)**2+TMZ(J)**2)              00065690
   XMS=XMACH**2                                                              00065700
   XM1=1.42857/XMS                                                           00065710
   CMN=1.+0.2*XMS*(1.-VTV(J)**2-VMV(J)**2)                                00065720
   IF(CMN.GT.0.0) GO TO 87                                                  00065730

```

```

CP(J)=-XM1                                00065760
GO TO 89                                   00065770
87 CP(J)=XM1*(CMN**3.5-1.)                 00065780
GO TO 89                                   00065790
88 CP(J)=1.-VTV(J)**2-VMV(J)**2           00065800
89 CONTINUE                                00065810
C*          PRINT OUTPUT                    00065820
  NC=0                                       00065830
  DO 110 J=1,NVY                             00065840
  WRITE (6,90)                                00065850
90 FORMAT (1H1,20X,65H ** SUBSONIC INTERFERENCE PRESSURE PROGRAM ** 00065860
1FUSELAGE OUTPUT **/6HU NO.,7X,2HXQ,1UX,2HYQ,1UX,2HZQ,8X,6HVT/VFS 00065870
2,6X,6HVM/VFS,8X,2HCP/)                    00065880
  DO 110 I=1,NJX                             00065890
  NC=NC+1                                     00065900
  ZP=ZQ(NC)/BETA                             00065910
  YP=YQ(NC)/BETA                             00065920
  WRITE (6,100) I,XQ(NC),YP,ZP,VTV(NC),VMV(NC),CP(NC) 00065930
100 FORMAT (1H ,I5,6F12.5)                  00065940
110 CONTINUE                                00065950
  RETURN                                       00065960
  END                                          00065970
*DECK BLOAD                                 00065980
  SUBROUTINE BLOAD (FCL,CDF,FCM)             00065990
C
C*          CALCULATE FUSELAGE LOCAL INTEGRATED LOADS 00066000
C
COMMON/DAT/ DA(5000)                        00066010
C
DIMENSION FVX(1),FJX(1)                    00066020
EQUIVALENCE (DA(4005),FCD) ,(DA(4735),FVX) ,(DA(4905),FJX) 00066030
1,(DA(1),AR) ,(DA(2),WSPAN) ,(DA(3),XMAC) ,(DA(4006),FAA) 00066040
2,(DA(7),XCG)                                00066050
C
COMMON/CRG/ PI,PI4,RC,BETA                 00066060
COMMON/CFG/ IDB,IDF,IDW,IDP,IDN            00066070
COMMON/CPB/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX,IFBE,SINA,COSA,FAREA 00066080
C
LARGE          BOU(3000) ,XQ(650) ,YQ(650) ,ZQ(650) 00066090
1              ,DM1(9750) ,AXY(650) ,AYZ(650) ,DM2(68891),DM3(2471) 00066100
2              ,CP(650) ,FDX(150) ,CLW(150) ,CDW(150) ,CMX(150) 00066110
3              ,FX(71) ,CLWA(71) ,CMXA(71) ,CMZ(150) ,CMZA(71) 00066120
4              ,CDWA(71)                                00066130

```

C	START	00066190
C*	CALC C(L)*W/W(AVG) AT DOWNWASH X STATION	00066200
	IY=1	00066210
	DO 40 I=1,NJX	00066220
	JX=FJX(I)	00066230
	DX=FVX(JX+1)-FVX(JX)	00066240
	FDX(I)=(FVX(JX)+FVX(JX+1))/2.	00066250
	IX=I	00066260
	CLW(I)=0.0	00066270
	CMX(I)=0.0	00066280
	CMZ(I)=0.0	00066290
	CDW(I)=0.0	00066300
	DO 30 J=1,NVY	00066310
	CLW(I)=CP(IX)*(AXY(IY)*COSA +AYZ(IY)*SINA)+CLW(I)	00066320
	CMX(I)=CP(IX)*(XQ(IX)-XCG)*AXY(IY)+CMX(I)	00066330
	CMZ(I)=CP(IX)*ZQ(IX)*AYZ(IY)+CMZ(I)	00066340
	CDW(I)=CP(IX)*(AXY(IY)*SINA -AYZ(IY)*COSA)+CDW(I)	00066350
	IX=IX+NJX	00066360
30	IY=IY+1	00066370
	CAX=FAREA*DX	00066380
	CSX=FCD/CAX	00066390
	CLW(I)=CLW(I)*CSX	00066400
	CMX(I)=-CMX(I)/CAX	00066410
	CMZ(I)=-CMZ(I)/CAX	00066420
40	CDW(I)=CDW(I)*CSX	00066430
C	* CALC C*W/W(AVG) AT EQUAL INTERVAL OF PHI	00066440
	DX=PI/70.	00066450
	FX(I)=0.0	00066460
	SPHI=0.0	00066470
	FCD2=FCD/2.	00066480
	DO 50 I=2,70	00066490
	SPHI=SPHI+DX	00066500
50	FX(I)=FCD2*(1.-COS(SPHI))	00066510
	CALL CODIM (FDX,CLW,NJX,FX,CLWA,70)	00066520
	CALL CODIM (FDX,CMX,NJX,FX,CMXA,70)	00066530
	CALL CODIM (FDX,CMZ,NJX,FX,CMZA,70)	00066540
	CALL CODIM (FDX,CDW,NJX,FX,CDWA,70)	00066550
	SPHI=0.0	00066560
	DO 52 I=2,70	00066570
	FX(I) = FX(I) / FCD	00066580
	SPHI=SPHI+DX	00066590
	HSP=SIN(SPHI)/2.	00066600
	CLW(I)=HSP*CLWA(I)	00066610

CF

```

CMX(I)=HSP*CMXA(I) 00066620
CMZ(I)=HSP*CMZA(I) 00066630
52 CDW(I)=HSP*CDWA(I) 00066640
C* SIMPSON RULE INTEGRATION. 00066650
SL1=CLW(70) 00066660
SL2=0.0 00066670
XM1=CMX(70) 00066680
XM2=0.0 00066690
ZM1=CMZ(70) 00066700
ZM2=0.0 00066710
SD1=CDW(70) 00066720
SD2=0.0 00066730
DO 60 I=2,69,2 00066740
SL1= CLW(I)+SL1 00066750
SL2= CLW(I+1)+SL2 00066760
XM1= CMX(I)+XM1 00066770
XM2= CMX(I+1)+XM2 00066780
ZM1= CMZ(I)+ZM1 00066790
ZM2= CMZ(I+1)+ZM2 00066800
SD1= CDW(I)+SD1 00066810
60 SD2= CDW(I+1)+SD2 00066820
DX3=DX/3. 00066830
IDT=IDF+IDW+IDP+IDN 00066840
IF(IDT.NE.0) GO TO 70 00066850
SR=1. 00066860
CR=1. 00066870
GO TO 80 00066880
70 SR=FAREA*AR/WSPAN**2 00066890
CR=FCD/XMAC 00066900
80 FCMXY=DX3*(4.*XM1+2.*XM2)*SR*CR 00066910
FCL=DX3*(4.*SL1+2.*SL2)*SR 00066920
FCMYZ=DX3*(4.*ZM1+2.*ZM2)*SR*CR 00066930
CDF=DX3*(4.*SD1+2.*SD2)*SR 00066940
FCM=FCMXY+FCMYZ 00066950
C* PRINT OUTPUT 00066960
WRITE (6,100) FAREA,FCL,CDF,FCMXY,FCMYZ,FCM,(FX(I),CLWA(I),CDWA(I)) 00066970
1 ,FX(I+35),CLWA(I+35),CDWA(I+35),I=2,35) 00066980
2 ,FX(36),CLWA(36),CDWA(36) 00066990
100 FORMAT (1H1,10X,45H ** FUSELAGE LOADS ** REF. FUSELAGE AREA=, 00067000
1 F14.5/1H0,10X, 5H* CL=F10.5,5X, 5H* CD=F10.5,3X, 00067010
2 7H* CMXY=F11.5,3X,7H* CMYZ=F11.5,5X, 5H* CM=F11.5/ 00067020
3 1H0,12X,1HX,15X,6HCLW/WA,11X,6HCDW/WA,12X,1HX,15X,6HCLW/WA 00067030
4 ,11X,6HCDW/WA/ 1H0,6F17.5 /(1H ,6F17.5)) 00067040

```

```

RETURN 00067050
END 00067060
*DECK FVRCP 00067070
SUBROUTINE FVRCP 00067080
C 00067090
C* CALC. VELOCITY RATIO AND PRESSURE COEFFICIENT AT 00067100
C* CONTROL POINTS ON THE FANPOD 00067110
C 00067120
COMMON/DAT/ DA(5000) 00067130
DIMENSION FJX(1),FVX(1) 00067140
EQUIVALENC (DA(4),XMACH) ,(DA(11),FWCI) ,(DA(1720),FJX) 00067150
1 ,(DA(2),WSPAN) ,(DA(1281),WNW) ,(DA(1460),FVX) 00067160
2 ,(DA(87),FOX) ,(DA(1203),WITO),(DA(1204),WITI) 00067170
C 00067180
COMMON/CRG/ PI,PI4,RC,BETA 00067190
COMMON/CFG/ IDB,IDF,IDW,IDP,IDN,NQB,NQF,NQW,NQP,NQN 00067200
1 ,NKB,NKF,NKW,NKP,NKN 00067210
COMMON/CFK/ NDA,NUT,THEF,SINTF,SINTF2,COSTF,THEK,SINTK,SINTK2 00067220
COMMON/CPF/ NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX,IFBE,SINA,COSA 00067230
COMMON/CPW/ DWI(6),NE3,NE5,NTB,NU,DW2(5),NVSO,NVSI,IWX 00067240
C 00067250
LARGE CKV(1000) ,DM1(2000) ,FUS(25000) 00067260
* ,XQ(650) ,YQ(650) ,ZQ(650) 00067270
1 ,TMX(650) ,TMY(650) ,TMZ(650) 00067280
2 ,TTX(650) ,TTY(650) ,TTZ(650) 00067290
3 ,XN(650) ,YN(650) ,ZN(650) 00067300
4 ,PM(100,25),DFI(13400),EV(52) ,DM2(944) 00067310
5 ,WPT(60) ,DM3(22374),PYL(6061) ,CEL(3700) 00067320
6 ,XF(51,10) ,DM4(561) 00067330
7 ,AX(250) ,AY(250) ,AZ(250) 00067340
8 ,CK(650) ,CP(650) ,XL(650) ,VMV(650) ,VTV(650) 00067350
9 ,ETA(2) ,PLM(51,25) 00067360
C 00067370
C* START 00067380
C* CALC. XKV IF WING-BODY 00067390
IF(IDW.EQ.0) GO TO 30 00067400
WP2=2.*WSPAN*PI*BETA 00067410
J=NE3+3 00067420
I=J+1 00067430
XF(I,NUT+1)=0.0 00067440
XF(J,NUT+1)=0.0 00067450
CALL SFC (I,EV) 00067460
XKI=WP2*(XF(I,1)-XF(I,2))/2.+XF(I,NUT+1)*SINTF+XF(I,NU)*SINTK) 00067470

```

	XKO=WP2*(XF(J,1)-XF(J,2)/2.+XF(J,NUT+1)*SINTF+XF(J,NU)*SINTK)	00067480
C*	CALC CONTRIBUTION TO FLOW FROM ADJACENT VORTICES	00067490
30	NC=0	00067500
	DO 36 IY=1,NVY	00067510
	DO 36 IJ=1,NJX	00067520
	J=FJX(IJ)	00067530
	K=J+1	00067540
	IF(J.EQ.NVXM) K=J	00067550
	NC=NC+1	00067560
	KC=(IY-1)*NF+NKB	00067570
	VMV(NC)=0.0	00067580
	DO 35 I=1,NF	00067590
	KC=KC+1	00067600
35	VMV(NC)=CKV(KC)*(PM(J,I)+PM(K,I))+VMV(NC)	00067610
36	VMV(NC)=VMV(NC)/4.	00067620
	IX=1	00067630
	DO 40 IJ=1,NJX	00067640
	JX=FJX(IJ)	00067650
37	IX=IX+1	00067660
	IF(IX.GE.JX) GO TO 38	00067670
	READ (24)	00067680
	GO TO 37	00067690
38	READ (24) ((PLM(J,I),J=1,NVY),I=1,NF)	00067700
	NC=IJ	00067710
	KC=NKB	00067720
	DO 40 J=1,NVY	00067730
	CK(NC)=0.0	00067740
	DO 39 I=1,NF	00067750
	KC=KC+1	00067760
39	CK(NC)=CKV(KC)*PLM(J,I)+CK(NC)	00067770
40	NC=NC+NJX	00067780
	REWIND 24	00067790
49	NC=0	00067800
	N1=0	00067810
	NCC=NJX*(NVY-1)	00067820
	DO 53 J=1,NVY	00067830
	DO 53 I=1,NJX	00067840
	N1=N1+1	00067850
	N2=N1-NJX	00067860
	IF(J.EQ.1) N2=N1+NCC	00067870
53	XL(N1)=SQRT((XQ(N2)-XQ(N1))**2+(YQ(N2)-YQ(N1))**2+(ZQ(N2)-ZQ(N1))	00067880
	I **2)	00067890
	NVYH=NVY/2+1	00067900

	DO 60 I=1,NKF	00068340
	N=N+1	00068350
	VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(N)+VTV(J)	00068360
60	VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(N)+VMV(J)	00068370
	IF(IDB.EQ.0) GO TO 62	00068380
	DO 61 J=1,NQF	00068390
	READ(19) (AX(IS),AY(IS),AZ(IS),IS=1,NKB)	00068400
	DO 61 I=1,NKB	00068410
	VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(I)+VTV(J)	00068420
61	VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(I)+VMV(J)	00068430
		00068440
C	62 IF(IDW.EQ.0) GO TO 65	00068450
	DO 64 J=1,NQF	00068460
	READ(21) (AX(IS),AY(IS),AZ(IS),IS=1,NKW),(V,VV,WV	00068470
	IF(DA(2000).EQ.0.0) GO TO 63	00068480
	UB=UV*BETA	00068490
	VTV(J)=TTX(J)*UB+TTY(J)*VV+TTZ(J)*WV+VTV(J)	00068500
	VMV(J)=TMX(J)*UB+TMY(J)*VV+TMZ(J)*WV+VMV(J)	00068510
63	II=NKB+NKF	00068520
	DO 64 I=1,NKW	00068530
	II=II+1	00068540
	VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(II)+VTV(J)	00068550
64	VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(II)+VMV(J)	00068560
65	IF(IDP.EQ.0) GO TO 72	00068570
	DO 70 J=1,NQF	00068580
	READ (22) (AX(IS),AY(IS),AZ(IS),IS=1,NKP),UV,VV,WV	00068590
	IF(DA(3000).EQ.0.0) GO TO 66	00068600
	UB=UV*BETA	00068610
	VTV(J)=TTX(J)*UB+TTY(J)*VV+TTZ(J)*WV+VTV(J)	00068620
	VMV(J)=TMX(J)*UB+TMY(J)*VV+TMZ(J)*WV+VMV(J)	00068630
66	II=NKB+NKF+NKW	00068640
	DO 70 I=1,NKP	00068650
	II=II+1	00068660
	VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(II)+VTV(J)	00068670
70	VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(II)+VMV(J)	00068680
72	IF(IDN.EQ.0) GO TO 80	00068690
	DO 74 J=1,NQF	00068700
	READ (9) (AX(IS),AY(IS),AZ(IS),IS=1,NKN)	00068710
	II=NKB+NKF+NKW+NKP	00068720
	DO 74 I=1,NKN	00068730
	II=II+1	00068740
	VTV(J)=(TTX(J)*AX(I)+TTY(J)*AY(I)+TTZ(J)*AZ(I))*CKV(II)+VTV(J)	00068750
74	VMV(J)=(TMX(J)*AX(I)+TMY(J)*AY(I)+TMZ(J)*AZ(I))*CKV(II)+VMV(J)	00068760

```

C
80 DO 89 J=1,NQF
   IF(XMACH.LE.0.0) GO TO 88
   VTV(J)=VTV(J)/SQRT((ITX(J)*BETA)**2+ITY(J)**2+ITZ(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,15,6F12.5)
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(146()),FVX) ,(DA(1720),FJX)
1,(DA(1),AR)      ,(DA(2),WSPAN)  ,(DA(3),XMAC)  ,(DA(6),FAA)
2,(DA(7),XCG)

```

```

00068770
00068780
00068790
00068800
00068810
00068820
00068830
00068840
00068850
00068860
00068870
00068880
00068890
00068900
00068910
00068920
00068930
00068940
00068950
00068960
00068970
00068980
00068990
00069000
00069010
00069020
00069030
00069040
00069050
00069060
00069070
00069080
00069090
00069100
00069110
00069120
00069130
00069140
00069150
00069160
00069170
00069180
00069190

```

						00069200
C	COMMON/CRG/	PI,PI4,RC,BETA				00069210
	COMMON/CFG/	IDB,IDF,IDW,IDP,IDN				00069220
	COMMON/CPF/	NVX,NVXM,NVY,NDV,NVB,NVBP,NF,NJX,IFBE,SINA,COSA,FAREA				00069230
C						00069240
	LARGE	ROU(3000),FUS(25000),XQ(650)	,YQ(650)	,ZQ(650)		00069250
1		,DM1(8450),AXY(650)	,AYZ(650)	,DM2(45191),DM3(2471)		00069260
2		,CP(650),FDX(150)	,CLW(150)	,CDW(150),CMX(150)		00069270
3		,FX(71),CLWA(71)	,CMXA(71)	,CMZ(150),CMZA(71)		00069280
4		,CDWA(71)				00069290
C		START				00069300
C*		CALC C(L)*W/W(AVG) AT DOWNWASH X STATION				00069310
	IY=1					00069320
	DO 40 I=1,NJX					00069330
	JX=FJX(I)					00069340
	DX=FVX(JX+1)-FVX(JX)					00069350
	FDX(I)=(FVX(JX)+FVX(JX+1))/2.					00069360
	IX=I					00069370
	CLW(I)=0.0					00069380
	CMX(I)=0.0					00069390
	CMZ(I)=0.0					00069400
	CDW(I)=0.0					00069410
	DO 30 J=1,NVY					00069420
	CLW(I)=CP(IX)*(AXY(IY)*COSA +AYZ(IY)*SINA)+CLW(I)					00069430
	CMX(I)=CP(IX)*(XQ(IX)-XCG)*AXY(IY)+CMX(I)					00069440
	CMZ(I)=CP(IX)*ZQ(IX)*AYZ(IY)+CMZ(I)					00069450
	CDW(I)=CP(IX)*(AXY(IY)*SINA -AYZ(IY)*COSA)+CDW(I)					00069460
	IX=IX+NJX					00069470
30	IY=IY+1					00069480
	CAX=FAREA*DX					00069490
	CSX=FCD/CAX					00069500
	CLW(I)=CLW(I)*CSX					00069510
	CMX(I)=-CMX(I)/CAX					00069520
	CMZ(I)=-CMZ(I)/CAX					00069530
40	CDW(I)=CDW(I)*CSX					00069540
C	*	CALC C*W/W(AVG) AT EQUAL INTERVAL OF PHI				00069550
	DX=PI/70.					00069560
	FX(1)=0.0					00069570
	SPHI=0.0					00069580
	FCD2=FCD/2.					00069590
	DO 50 I=2,70					00069600
	SPHI=SPHI+DX					00069610
50	FX(I)=FCD2*(1.-COS(SPHI))					00069620

CALL CODIM (FDX,CLW,NJX,FX,CLWA,70)	00069630
CALL CODIM (FDX,CMX,NJX,FX,CMXA,70)	00069640
CALL CODIM (FDX,CMZ,NJX,FX,CMZA,70)	00069650
CALL CODIM (FDX,CDW,NJX,FX,CDWA,70)	00069660
SPHI=0.0	00069670
DO 52 I=2,70	00069680
FX(I) = FX(I) / FCD	00069690
SPHI=SPHI+DX	00069700
HSP=SIN(SPHI)/2.	00069710
CLW(I)=HSP*CLWA(I)	00069720
CMX(I)=HSP*CMXA(I)	00069730
CMZ(I)=HSP*CMZA(I)	00069740
52 CDW(I)=HSP*CDWA(I)	00069750
C* SIMPSON RULE INTEGRATION	00069760
SL1=CLW(70)	00069770
SL2=0.0	00069780
XM1=CMX(70)	00069790
XM2=0.0	00069800
ZM1=CMZ(70)	00069810
ZM2=0.0	00069820
SD1=CDW(70)	00069830
SD2=0.0	00069840
DO 60 I=2,69,2	00069850
SL1= CLW(I)+SL1	00069860
SL2= CLW(I+1)+SL2	00069870
XM1= CMX(I)+XM1	00069880
XM2= CMX(I+1)+XM2	00069890
ZM1= CMZ(I)+ZM1	00069900
ZM2= CMZ(I+1)+ZM2	00069910
SD1= CDW(I)+SD1	00069920
60 SD2= CDW(I+1)+SD2	00069930
DX3=DX/3.	00069940
IDT=IDB+IDW+IDP+IDN	00069950
IF(IDT.NE.0) GO TO 70	00069960
SR=1.	00069970
CR=1.	00069980
GO TO 80	00069990
70 SR=FAREA*AR/WSPAN**2	00070000
CR=FCD/XMAC	00070010
80 FCMXY=DX3*(4.*XM1+2.*XM2)*SR*CR	00070020
FCL=DX3*(4.*SL1+2.*SL2)*SR	00070030
FCMYZ=DX3*(4.*ZM1+2.*ZM2)*SR*CR	00070040
CDF=DX3*(4.*SD1+2.*SD2)*SR	00070050

```

FCM=FCMXY+FCMYZ                                00070060
C*          PRINT OUTPUT                          00070070
WRITE (6,100) FAREA,FCL,CDF,FCMXY,FCMYZ,FCM,(FX(I),CLWA(I),CDWA(I) 00070080
1          ,FX(I+35),CLWA(I+35),CDWA(I+35),I=2,35) 00070090
2          ,FX(36),CLWA(36),CDWA(36)              00070100
100 FORMAT (1H1,10X,45H ** FANPOD LOADS          ** REF. FANPOD AREA=, 00070110
1          F14.5/1H0,10X, 5H* CL=F10.5,5X, 5H* CD=F10.5,3X, 00070120
2          7H* CMXY=F11.5,3X,7H* CMYZ=F11.5,5X, 5H* CM=F11.5/ 00070130
3          1H0,12X,1HX,15X,6HCLW/WA,11X,6HCDW/WA,12X,1HX,15X,6HCLW/WA 00070140
4          ,11X,6HCDW/WA/ 1H0,6F17.5/(1H,6F17.5)) 00070150
RETURN                                           00070160
END                                              00070170
*DECK WDPQ                                       00070180
SUBROUTINE WDPQ                                  00070190
C                                                00070200
C*          CALC WING PRESSURE COEFFICIENTS (DP/Q) 00070210
C                                                00070220
COMMON/DAT/ DA(5000)                             00070230
C                                                00070240
DIMENSION WPE(60) ,WPC(30) ,WPS(30)             00070250
C                                                00070260
EQUIVALENCE (DA(2),WSPAN) ,(DA(4),XMACH) ,(DA(11),FWCI) 00070270
1          ,(DA(1210),WPE) ,(DA(1870),WPC) ,(DA(1900),WPS) 00070280
2          ,(DA(1960),DAF) ,(DA(1965),DAK) ,(DA(1270),WNVC) 00070290
3          ,(DA(1281),WNW)                       00070300
C                                                00070310
COMMON/CRG/ PI,PI4,RC,BETA                       00070320
COMMON/CPW/ DW1(9),NU,NW                         00070330
COMMON/CFG/ DF1(10),NKB,NKF                     00070340
COMMON/CFK/ NDA,NUT,THEF,SINTF,SINTF2,COSTF,THEK,SINTK,SINTK2 00070350
C                                                00070360
LARGE      CKV(1000) ,BOU(2000) ,FUS(25000),FAN(23700) 00070370
1          ,EV(52) ,DM1(944) ,WPT(60)           00070380
2          ,DM2(22374),PYL(6061),CEL(3700)      00070390
3          ,XF(51,10) ,DM3(561) ,TA(30,10)     00070400
4          ,DPQR(30) ,XLE(30) ,XTLE(30) ,THE(30) 00070410
5          ,EPS(30)                               00070420
C                                                00070430
C                                                00070440
NPC=DA(1272)                                     00070450
NPS=DA(1273)                                     00070460
C          SHIFT TO LCM FOR SFC AND WINGD        00070470
DO 10 I=1,NPS                                    00070480

```

10	EPS(I)=WPS(I)	00070490
C	CALC MATRIX-F	00070500
	IF(WNW.NE.0.0) GO TO 570	00070510
	NPS=NW	00070520
	IC=NW	00070530
	DO 400 IE=1,NPS	00070540
	WPS(IC)=EV(IE)	00070550
	EPS(IC)=EV(IE)	00070560
400	IC=IC-1	00070570
	IC=NKB+NKF	00070580
	DO 410 IU=1,NU	00070590
	IX=NPS	00070600
	DO 410 IE=1,NPS	00070610
	IC=IC+1	00070620
	XF(IX,IU)=CKV(IC)/BETA	00070630
410	IX=IX-1	00070640
	GO TO 580	00070650
570	CALL SFC (NPS,EPS)	00070660
C	CALC CHORD LOAD SHAPES FUNCTION	00070670
580	DO 594 I=1,NPC	00070680
	THE(I)= ACOS(1.-2.*WPC(I))	00070690
	THED2=THE(I)/2.	00070700
	TA(I,1)= COS(THED2)/ SIN(THED2)	00070710
	ANC=0.0	00070720
	DO 590 J=2,NUT	00070730
	ANC=ANC+1.	00070740
590	TA(I,J) = SIN(ANC * THE(I))	00070750
	IF(DAF.EQ.0.0) GO TO 592	00070760
	SINTT=SIN(.5*(THE(I)-THEF))	00070770
	IF(ABS(SINTT).LT.0.001) SINTT=0.001	00070780
	TA(I,NUT+1)=ALOG(ABS(SIN(.5*(THE(I)+THEF))/SINTT))	00070790
592	IF(DAK.EQ.0.0) GO TO 594	00070800
	SINTT=SIN(.5*(THE(I)-THEK))	00070810
	IF(ABS(SINTT).LT.0.001) SINTT=0.001	00070820
	TA(I,NU)=ALOG(ABS(SIN(.5*(THE(I)+THEK))/SINTT))	00070830
594	CONTINUE	00070840
	X3=8.*WSPAN	00070850
	CALL WINGD (NPS,EPS,XLE,XTLE,WPT)	00070860
	WRITE (6,1000) (WPC(I),I=1,NPC)	00070870
	WRITE (6,2000)	00070880
	DO 620 NE=1,NPS	00070890
595	EBC=X3/XTLE(NE)	00070900
	DO 610 I=1,NPC	00070910

```

DPQR(1)=0.0                                00070920
DO 600 J=1,NU                                00070930
600 DPQR(1)=DPQR(1)+XF(NE,J)*TA(I,J)        00070940
610 DPQR(1)=EBC*DPQR(1)                     00070950
WRITE (6,3000)WPS(NE),(DPQR(1),I=1,NPC)    00070960
620 CONTINUE                                  00070970
640 RETURN                                    00070980
1000 FORMAT (31H1 ** WING LINEAR PRESSURE COEF./1HU,7X,14H** LIST OF X/00070990
1C/1H0,6X,10F10.4/(1H ,6X,10F10.4))        00071000
2000 FORMAT(35H0* ETA ** LISTS OF CP AT ABOVE X/C) 00071010
3000 FORMAT(1H0F6.3,10F10.4/(1H 6X,10F10.4)) 00071020
END                                           00071030
*DECK WVAP                                    00071040
SUBROUTINE WVAP                              00071050
C                                              00071060
C *VAP* CALC. VELOCITIES AND PRESSURES WITH THICKNESS 00071070
C                                              00071080
COMMON/DAT/ DA(5000)                        00071090
C                                              00071100
DIMENSION AJ(30)                            00071110
C                                              00071120
EQUIVALENCE (DA(2),WS) ,(DA(4),XMACH) ,(DA(1270),XNI) 00071130
1,(DA(1660),AJ) ,(DA(1960),DAF) ,(DA(1965),DAK) 00071140
2,(NJC,NJ) ,(NJS,NEP)                      00071150
C                                              00071160
COMMON/CRG/ PI,PI4,RC,BETA                 00071170
COMMON/CFG/ IDB,IDF,IDW,IDP,IDN,NQB,NQF,NQW,NQP,NQN 00071180
1 ,NKB,NKF,NKW,NKP,NKN                    00071190
COMMON/CPW/ DW1(9),NU,NW,NVC,NVS,NJC,NJS 00071200
COMMON/CFK/ NDA,NUT,THEF,SINTF,SINTF2,COSTF,THEK,SINTK,SINTK2 00071210
1 ,COSTK,SINDF,COSDF,SINDK,COSDK          00071220
C                                              00071230
LARGE CKV(1000) ,BOU(2000) ,FUS(25000),FAN(23700) 00071240
* ,DM1(996) ,WPT(60) ,BS(650)            00071250
1 ,EP(30) ,XTLP(30) ,XOCP(30)            00071260
2 ,CSJ(30,30),TSJ(30,30),UV(30,30)      00071270
3 ,UVP(30,30),VV(30,30) ,VVP(30,30)     00071280
4 ,DM2(25995)                             00071290
5 ,XF(51,10) ,SRE(51) ,S(51,10)         00071300
6 ,BE(30,30) ,CF(30,30) ,CA(30,10) ,TA(30,11) ,TB(30,11) 00071310
7 ,TPTE(30) ,TPLE(30) ,PF(30,10)        00071320
8 ,UVU(30) ,UVL(30,30),VVU(30) ,VVL(30,30) 00071330
9 ,CPU(30) ,CPL(30) ,CPN(30) ,CPO(30)   00071340

```

	*	AX(250)	AY(250)	AZ(250)	00071350
C					00071360
C*		INITIAL THETA AND TRIG. FUNC.			00071370
		DTH=PI/XNI			00071380
		DO 40 J=1,NJ			00071390
		THE=DTH*AJ(J)			00071400
		THE2=THE/2.			00071410
		TA(J,1)=COS(THE2)/SIN(THE2)			00071420
		CF(J,1)=1.			00071430
		C=0.			00071440
		DO 10 N=1,NUT			00071450
		C=C+1.			00071460
		CA(J,N)=COS(C*THE)			00071470
		CF(J,N+1)=CA(J,N)			00071480
10		TA(J,N+1)=SIN(C*THE)			00071490
		TB(J,1)=TA(J,2)+THE			00071500
		TB(J,2)=(THE2-TA(J,3))/4.			00071510
		DO 20 N=3,NUT			00071520
		D2=N+N			00071530
		D1=D2-4.			00071540
20		TB(J,N)=TA(J,N-1)/D1-TA(J,N+1)/D2			00071550
		IF(DAF.EQ.0.0) GO TO 30			00071560
		SINTT=SIN(.5*(THE-THEF))			00071570
		IF(ABS(SINTT).LT.0.001) SINTT=.001			00071580
		AST=ABS(SIN(.5*(THE+THEF)))/SINTT			00071590
		TA(J,NUT+1)=ALOG(AST)			00071600
		TB(J,NUT+1)=(COS(THE)-COSTF)*ALOG(1./AST)+THE*SINTF			00071610
30		IF(DAK.EQ.0.0) GO TO 40			00071620
		SINTT=SIN(.5*(THE-THEK))			00071630
		IF(ABS(SINTT).LT.0.001) SINTT=.001			00071640
		AST=ABS(SIN(.5*(THE+THEK)))/SINTT			00071650
		TA(J,NU)=ALOG(AST)			00071660
		TB(J,NU)=(COS(THE)-COSTK)*ALOG(1./AST)+THE*SINTK			00071670
40		CONTINUE			00071680
		IF(DA(2000).EQ.0.0) GO TO 200			00071690
C*		CALC SLOPE OF CAMBER SURFACE			00071700
		NA=0			00071710
		DO 41 I=1,NEP			00071720
		DO 41 J=1,NJ			00071730
		NA=NA+1			00071740
41		BE(J,I)=BS(NA)			00071750
		CALL MSOLX(NUT,NJ,NEP,CF,BE,30)			00071760
		DO 44 IE=1,NEP			00071770

	DO 44 J=1,NJ	00071780
	SUMA=0.0	00071790
	DO 42 N=2,NUT	00071800
42	SUMA=CF(N,IE)*CA(J,N-1)+SUMA	00071810
	SUMB=0.0	00071820
	C=0.0	00071830
	DO 43 N=3,NUT,2	00071840
	C=C+1.	00071850
43	SUMB=CF(N,IE)/(4.*C**2-1.)+SUMB	00071860
44	CSJ(IE,J)=-SUMA-SUMB	00071870
C*	TPS FOR TAN PHI STAR	00071880
	HWS=WS/2.	00071890
	CALL TPS (HWS,WPT,NEP,EP,TPTE,TPLE)	00071900
C*	SPF FOR DERIVATIVE OF F MATRIX	00071910
	CALL SFC (NEP,EP)	00071920
	CALL SPF	00071930
C*	CALC. VELOCITIES AND PRESSURES	00071940
	DO 45 IE=1,NEP	00071950
	DO 45 J=1,NJ	00071960
	UVL(IE,J)=0.0	00071970
45	VVL(IE,J)=0.0	00071980
	IF(IDB.EQ.0) GO TO 47	00071990
C*	CALC INFLUENCE OF OTHER COMPONENTS ON WING	00072000
	DO 46 IE=1,NEP	00072010
	DO 46 J=1,NJ	00072020
	READ (19) (AX(IS),AY(IS),AZ(IS),IS=1,NKB)	00072030
	DO 46 IS=1,NKB	00072040
	UVL(IE,J)=UVL(IE,J)+AX(IS)*CKV(IS)	00072050
46	VVL(IE,J)=VVL(IE,J)+AY(IS)*CKV(IS)	00072060
47	IF(IDF.EQ.0) GO TO 49	00072070
	DO 48 IE=1,NEP	00072080
	DO 48 J=1,NJ	00072090
	READ (20) (AX(IS),AY(IS),AZ(IS),IS=1,NKF)	00072100
	IC=NKB	00072110
	DO 48 IS=1,NKF	00072120
	IC=IC+1	00072130
	UVL(IE,J)=UVL(IE,J)+AX(IS)*CKV(IC)	00072140
48	VVL(IE,J)=VVL(IE,J)+AY(IS)*CKV(IC)	00072150
49	IF(IDP.EQ.0) GO TO 51	00072160
	DO 50 IE=1,NEP	00072170
	DO 50 J=1,NJ	00072180
	READ (22) (AX(IS),AY(IS),AZ(IS),IS=1,NKP),PUV,PVV	00072190
	IC=NKB+NKF+NKW	00072200

	CP0(J)=C1*(CP1+CPS)	00072640
	UVS=C2*(CP1*TPU+CPS)	00072650
	UVU(J)=(UV(IE,J)+UVS+(1.+TPX*DSU)/TP1)/DSU+UVB/DSU	00072660
	UVL(IE,J)=(UV(IE,J)-UVS+(1.+TPX*DSL)/TP1)/DSL+UVB/DSL	00072670
	VV1=2.*PF(IE,1)*TB(J,1)-C2*CP1*TP	00072680
	VVS=VV1*TPV+2.*VVS-C2*CPS*TP	00072690
	TTP=TP/TP1	00072700
	VVU(J)=(VV(IE,J)+VVS-TTP*(1.-DSU))/DSU*BETA+VVB/DSU	00072710
	VVL(IE,J)=(VV(IE,J)-VVS-TTP*(1.-DSL))/DSL*BETA+VVB/DSL	00072720
	CPU(J)=1.-UVU(J)**2-VVU(J)**2	00072730
	CPL(J)=1.-UVL(IE,J)**2-VVL(IE,J)**2	00072740
C	COMPRESSIBILITY EFFECTS	00072750
	IF(XMACH.LE.0.0) GO TO 78	00072760
	XMS=XMACH**2	00072770
	XM1=1.42857/XMS	00072780
	XM2=0.2*XMS	00072790
	IF(XMACH.GT.0.01) GO TO 62	00072800
	CPU(J)=CPU(J)+XMS*CPU(J)/4.0	00072810
	CPL(J)=CPL(J)+XMS*CPL(J)/4.0	00072820
	GO TO 70	00072830
62	CPU(J)=XM1*(1.+XM2*CPU(J))**3.5-XM1	00072840
	CPL(J)=XM1*(1.+XM2*CPL(J))**3.5-XM1	00072850
C		00072860
70	SINP=TP/TP2	00072870
	COSP=1./TP2	00072880
	CP2=COSP**2	00072890
	SP2=SINP**2	00072900
	SCP=SINP*COSP	00072910
	CPMA=XMACH*COSP	00072920
	CPM2=CPMA**2	00072930
	SRCM=SQRT(1.-CPM2)	00072940
	CSR=.5*(1.-SRCM)	00072950
	CPU(J)=CPU(J)*SRCM/(1.-CPU(J)*CSR)	00072960
	CPL(J)=CPL(J)*SRCM/(1.-CPL(J)*CSR)	00072970
	BWU=SQRT(1.-CPM2*(1.-CPU(J)*CPMA))	00072980
	BWL=SQRT(1.-CPM2*(1.-CPL(J)*CPMA))	00072990
	DVMU=UV(IE,J)+UVS+UVB	00073000
	DVML=UV(IE,J)-UVS+UVB	00073010
	DVTU=(VV(IE,J)+VVS)*BETA+VVB	00073020
	DVTL=(VV(IE,J)-VVS)*BETA+VVB	00073030
	SBDU=SQRT(BWU**2+DZU)	00073040
	SBDL=SQRT(BWL**2+DZL)	00073050
	UVU(J)=(BWU*CP2+(BSQ*DVMU*CP2-BETA*DVTU*SCP)/SRCM)/SBDU	00073060

	* +(1.+DVMU)*SP2+DVTU*SCP	00073070
	UVL(IE,J)=(BWL*CP2+(BSQ*DVML*CP2-BETA*DVTL*SCP)/SRCM)/SBDL	00073080
	* +(1.+DVML)*SP2+DVTL*SCP	00073090
	VVU(J)=- (BWU*SCP+(BSQ*DVMU*SCP-BETA*DVTU*SP2)/SRCM)/SBDU	00073100
	* +(1.+DVMU)*SCP+DVTU*CP2	00073110
	VVL(IE,J)=- (BWL*SCP+(BSQ*DVML*SCP-BETA*DVTL*SP2)/SRCM)/SBDL	00073120
	* +(1.+DVML)*SCP+DVTL*CP2	00073130
	CPU(J)=1.-UVU(J)**2-VVU(J)**2	00073140
	CPL(J)=1.-UVL(IE,J)**2-VVL(IE,J)**2	00073150
	IF(XMACH.GT.0.01) GO TO 72	00073160
	CPU(J)=CPU(J)+XMS*CPU(J)/4.0	00073170
	CPL(J)=CPL(J)+XMS*CPL(J)/4.0	00073180
	GO TO 78	00073190
	72 CPU(J)=XM1*(1.+XM2*CPU(J))**3.5-XM1	00073200
	CPL(J)=XM1*(1.+XM2*CPL(J))**3.5-XM1	00073210
C	SAVE CP FOR LOAD CALC	00073220
	78 BE(IE,J)=CPU(J)	00073230
	CF(IE,J)=CPL(J)	00073240
	80 CPN(J)=CPL(J)-CPU(J)	00073250
	90 WRITE (6,120) EP(IE),(XOCP(J),UVU(J),UVL(IE,J),VVU(J),	00073260
	1 VVL(IE,J),CPU(J),CPL(J),CPN(J),CPO(J),J=1,NJ)	00073270
	100 FORMAT (54H1 **WING NON-LINEAR VELOCITY AND PRESSURE COEFFICIENTS)	00073280
	120 FORMAT (8H0 ETA =,F7.4/1H ,7X,3HX/C,7X,6HU/V UP,6X,7HU/V LOW,5X,	00073290
	1 6HV/V UP,6X,7HV/V LOW,6X,5HCP UP,7X,6HCP LOW,6X,6HCP L-U,	00073300
	2 5X,7HCP NETL/(1H ,9F12.5))	00073310
	GO TO 300	00073320
C	CALC LINEAR PRESSURE COEFFICIENTS AND SKIP RECORDS	00073330
	200 WS8=8.*WS	00073340
	CALL SFC (NEP,EP)	00073350
	DO 220 IE=1,NEP	00073360
	C1=WS8/XTLP(IE)	00073370
	DO 220 J=1,NJ	00073380
	READ (21)	00073390
	IF(IDB.NE.0) READ (19)	00073400
	IF(IDF.NE.0) READ (20)	00073410
	IF(IDP.NE.0) READ (22)	00073420
	IF(IDN.NE.0) READ (9)	00073430
		00073440
	CPS=0.0	00073450
	DO 210 N=1,NU	00073460
	210 CPS=CPS+XF(IE,N)*TA(J,N)	00073470
	220 BE(IE,J)=C1*CPS	00073480
C		00073490

```

300 RETURN                                00073500
END                                        00073510
*DECK WLOAD                               00073520
SUBROUTINE WLOAD(WCL,WCD,WCM)            00073530
C                                         00073540
C*          CALC. WING LOAD PARAMETERS    00073550
C                                         00073560
COMMON/DAT/ DA(5000)                    00073570
C                                         00073580
DIMENSION WJS(1),WJC(1)                 00073590
C                                         00073600
EQUIVALENCE      (DA(1),AR)      ,(DA(2),SPAN)      ,(DA(3),XMAC)      00073610
1,(DA(7),XCG)      ,(DA(1270),WNVC) ,(DA(1660),WJC) ,(DA(1690),WJS) 00073620
2,(DA(1960),DAF) ,(DA(1961),PXCF) ,(DA(1962),ETFI) ,(DA(1963),ETFO) 00073630
3,(DA(1965),DAK) ,(DA(1966),PXCK) ,(DA(1967),ETKI) ,(DA(1968),ETKO) 00073640
C                                         00073650
COMMON/CRG/ PI                                00073660
COMMON/CFG/ IDB,IDF                          00073670
COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3,NE5,IW1(3),NVC,NVS,NJC,NJS 00073680
COMMON/CFK/ IK1(10),SINDF,COSDF,SINDK,COSDK   00073690
C                                         00073700
LARGE      BOU(3000) ,FUS(25000),FAN(23700) 00073710
1          ,EV(52)      ,XLV(52)      ,XTLV(52)      ,DM1(900)      ,BS(650)      00073720
2          ,EP(30)      ,XTLP(30)      ,XOCP(30)      ,DM2(900)      ,TSJ(30,30) 00073730
3          ,DM3(3600) ,XR(1800)      ,YR(1800)      ,ZR(1800)      00073740
4          ,DM4(10834),PYL(6061) ,CEL(370)      ,DM5(1071)      00073750
5          ,CPU(30,30),CPL(30,30),DSS(30,30)    00073760
6          ,THI(32)      ,CNI(32)      ,CMI(32)      ,CDI(32)      ,STH(30)      00073770
7          ,THO(40)      ,CNO(40)      ,CMO(40)      ,CDO(40)      00073780
8          ,EPI(31)      ,CNCCI(31) ,CMCCI(31) ,CDCCI(31) ,XCCP(52) 00073790
9          ,EVO(52)      ,XLE(52)      ,CNCCO(52) ,CMCCO(52) ,CDCCO(52) 00073800
C                                         00073810
C*          CALC DS/DSLE OF ROOT SECTIONS 00073820
IF(IDB.NE.0) DSLEB=SQRT((YR(1201)-YR(1202))**2+(ZR(1201)-ZR(1202))**2) 00073830
1 **2)                                00073840
IF(IDF.EQ.0) GO TO 10                    00073850
DSLE1=SQRT((YR(1)-YR(2))**2+(ZR(1)-ZR(2))**2) 00073860
DSLE2=SQRT((YR(601)-YR(602))**2+(ZR(601)-ZR(602))**2) 00073870
C                                         00073880
SPANWISE LOOP                             00073880
10 DO 50 IJS=1,NJS                         00073890
IY=WJS(IJS)                               00073900
IEB=IY-NE5                                00073910
IEF=IY-NE3                                00073920

```

C	CHORDWISE LOOP	00073930
	70 50 IJC=1,NJC	00073940
	IX=WJC(IJC)	00073950
	DSS(IJS,IJC)=1.0	00073960
	IF(IDF.EQ.0) GO TO 30	00073970
	IF(IEF.LE.0) GO TO 50	00073980
	IF(IEF.GT.3) GO TO 20	00073990
	DSLE=DSLE1	00074000
	IC=4*IX+IEF	00074010
	GO TO 40	00074020
C	INBOARD ROOT OF FANPOD	00074030
	20 IF(IEF.GT.6) GO TO 30	00074040
	DSLE=DSLE2	00074050
	IC=4*IX+IEF+597	00074060
	GO TO 40	00074070
C	ROOT OF FUSELAGE	00074080
	30 IF(IDB.EQ.0) GO TO 50	00074090
	IF(IEB.LE.0) GO TO 50	00074100
	IF(IEB.GT.3) GO TO 50	00074110
	DSLE=DSLEB	00074120
	IC=4*IX+IEB+1200	00074130
C		00074140
	40 DY=(YR(IC)-YR(IC+1)+YR(IC-4)-YR(IC-3))/2.0	00074150
	DZ=(ZR(IC)-ZR(IC+1)+ZR(IC-4)-ZR(IC-3))/2.0	00074160
	DS=SQRT(DY**2+DZ**2)	00074170
	DSS(IJS,IJC)=DS/DSLE	00074180
	50 CONTINUE	00074190
C	OUTBOARD ROOT OF FANPOD	00074200
C	CALC CHORDWISE THETAS	00074210
	DTH=PI/WNVC	00074220
	CONS=.5*AR/SPAN*DTH	00074230
	THO(1)=DTH	00074240
	DO 60 I=2,NVC	00074250
	60 THO(I)=THO(I-1)+DTH	00074260
C		00074270
	THI(1)=0.0	00074280
	DO 70 IC=1,NJC	00074290
	THI(IC+1)=WJC(IC)*DTH	00074300
	70 STH(IC)=SIN(THI(IC+1))	00074310
	NJI=NJC+2	00074320
	THI(NJI)=PI	00074330
C		00074340
C*	CALC LIFT, DRAG AND MOMENT COEFFICIENTS	00074350

	IV=NJS	00074360
	NA=0	00074370
	CNI(1)=0.0	00074380
	CMI(1)=0.0	00074390
	CDI(1)=0.0	00074400
	CNI(NJI)=0.0	00074410
	CMI(NJI)=0.0	00074420
	CDI(NJI)=0.0	00074430
C		00074440
	DO 160 IE=1,NJS	00074450
	DO 140 IC=1,NJC	00074460
	TD=0.0	00074470
C	CALC TAN OF FLAP DEFLECTION	00074480
	IF(DAF.EQ.0.0) GO TO 100	00074490
	IF(EP(IE).LE.ETFI) GO TO 100	00074500
	IF(EP(IE).GE.ETFO) GO TO 100	00074510
	IF(XOCP(IC).LE.PXCF) GO TO 100	00074520
	TD=SINDF/COSDF	00074530
	GO TO 110	00074540
C	CALC TAN OF KRUGER DEFLECTION	00074550
100	IF(DAK.EQ.0) GO TO 110	00074560
	IF(EP(IE).LE.ETKI) GO TO 110	00074570
	IF(EP(IE).GE.ETKO) GO TO 110	00074580
	IF(XOCP(IC).LE.PXCK) GO TO 110	00074590
	TD=SINDK/COSDK	00074600
C	INCLUDE THICKNESS SLOPES,CAMBER AND TWIST	00074610
110	NA=NA+1	00074620
	IF(DA(2000).EQ.0.0) GO TO 120	00074630
	STCTU=-TSJ(IE,IC)+BS(NA)	00074640
	STCTL=TSJ(IE,IC)+BS(NA)	00074650
	TAU=(TD+STCTU)/(1.-TD*STCTU)	00074660
	TAL=(TD+STCTL)/(1.-TD*STCTL)	00074670
	CPN=CPL(IE,IC)-CPU(IE,IC)	00074680
	CPD=TAL*CPL(IE,IC)-TAU*CPU(IE,IC)	00074690
	GO TO 130	00074700
120	CPN=CPU(IE,IC)	00074710
	CPD=(TD+BS(NA))/(1.-TD*BS(NA))*CPU(IE,IC)	00074720
130	CPM=CPN*XOCP(IC)*XTLP(IE)	00074730
	DST=DSS(IE,IC)*STH(IC)	00074740
	CNI(IC+1)=DST*CPN	00074750
	CMI(IC+1)=DST*CPM	00074760
140	CDI(IC+1)=DST*CPD	00074770
C	CHORDWISE EXPANSION	00074780

```

CALL CODIM(THI,CNI,NJI,THO,CNO,NVC) 00074790
CALL CODIM(THI,CMI,NJI,THO,CMO,NVC) 00074800
CALL CODIM(THI,CDI,NJI,THO,CDO,NVC) 00074810
C      SUM CN,CM AND CD CHORDWISE 00074820
      CN=0.0 00074830
      CM=0.0 00074840
      CD=0.0 00074850
      DO 150 I=1,NVC 00074860
      CN=CN+CNO(I) 00074870
      CM=CM+CMO(I) 00074880
150    CD=CD+CDO(I) 00074890
C      SET PARAMETERS IN ASCENDING ORDER 00074900
      EPI(IV)=EP(IE) 00074910
      CC=CONS*XTLP(IE) 00074920
      CNCCI(IV)=CC*CN 00074930
      CMCCI(IV)=-CC*CM/XMAC 00074940
      CDCCI(IV)=CC*CD 00074950
160    IV=IV-1 00074960
C      00074970
C*      INTEGRATE SPANWISE 00074980
      NSI=NJS+1 00074990
      EPI(NSI)=1.0 00075000
      CNCCI(NSI)=0.0 00075010
      CMCCI(NSI)=0.0 00075020
      CDCCI(NSI)=0.0 00075030
      IV=NE2 00075040
      DO 170 I=1,NE2 00075050
      EVO(IV)=EV(I) 00075060
      XLE(IV)=XLV(I) 00075070
170    IV=IV-1 00075080
C      SPANWISE EXPANSION 00075090
      CALL CODIM(EPI,CNCCI,NSI,EVO,CNCCO,NE2) 00075100
      CALL CODIM(EPI,CMCCI,NSI,EVO,CMCCO,NE2) 00075110
      CALL CODIM(EPI,CDCCI,NSI,EVO,CDCCO,NE2) 00075120
C      CALC TOTAL WING LIFT,DRAG AND MOMENT 00075130
      WCL=0.0 00075140
      WCM=0.0 00075150
      WCD=0.0 00075160
      NEE=NE2-NE3-3 00075170
      DO 240 I=1,NE2 00075180
      IF(ABS(CNCCO(I)).LT.0.000001) CNCCO(I)=0.000001 00075190
      XCCP(I)=-CMCCO(I)/CNCCO(I) 00075200
      CMCG=CMCCO(I)-XLE(I)*CNCCO(I)+XCG*CNCCO(I) 00075210

```



```

IF(IDF.EQ.0) GO TO 200                                00075220
IF(I.GT.NEE) GO TO 200                                00075230
CNX=DLI*CNCCO(I)                                       00075240
CMX=DLI*CMCG                                           00075250
CDX=DLI*CDCCO(I)                                       00075260
GO TO 210                                             00075270
200 CNX=DLO*CNCCO(I)                                    00075280
    CMX=DLO*CMCG                                       00075290
    CDX=DLO*CDCCO(I)                                    00075300
210 IF(IDB.NE.0) GO TO 220                            00075310
    IF(I.NE.1) GO TO 220                              00075320
    CNX=CNX/2.                                         00075330
    CMX=CMX/2.                                         00075340
    CDX=CDX/2.                                         00075350
    GO TO 230                                           00075360
220 IF(I.NE.NE2) GO TO 230                            00075370
    CNX=CNX/3.                                         00075380
    CMX=CMX/3.                                         00075390
    CDX=CDX/3.                                         00075400
230 WCL=WCL+CNX                                       00075410
    WCM=WCM+CMX                                       00075420
240 WCD=WCD+CDX                                       00075430
    IF(ABS(WCL).LT.0.000001) WCL=0.000001           00075440
    WXC=-WCM/WCL                                       00075450
C                                                       00075460
C               PRINT OUTPUT                            00075470
C                                                       00075480
    WRITE (6,300) (I,EVO(I),CNCCO(I),CDCCO(I),CMCCO(I),XCCP(I),
1   I=1,NE2)                                           00075490
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 ,115,5F15.5)) 00075500
C                                                       00075510
C                                                       00075520
    WRITE (6,310) WCL,WCD,WCM,WXC                       00075530
310 FORMAT (1H0,20X,22H ** TOTAL WING LOADS */1H ,39X,3HWCL,12X,3HWCD,
1   12X,3HWCM,12X,3HX/C/1H ,30X,4F15.5)              00075540
C                                                       00075550
C                                                       00075560
    RETURN                                              00075570
    END                                                00075580
*DECK PDPQ                                           00075590
    SUBROUTINE PDPQ                                     00075600
C                                                       00075610
C*               CALC PYLON PRESSURE COEFFICIENTS (DP/Q) 00075620
C                                                       00075630
    COMMON/DAT/ DA(5000)                               00075640

```



```

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
C  COMMON/DAT/ DA(5000)
C
C  DIMENSION  PJC(18)
C
C  EQUIVALENCE      (DA(4),XMACH)      ,(DA(2500),PHT) ,(DA(2512),PJC)
00076080
00076090
00076100
00076110
00076120
00076130
00076140
00076150
00076160
00076170
00076180
00076190
00076200
00076210
00076220
00076230
00076240
00076250
00076260
00076270
00076280
00076290
00076300
00076310
00076320
00076330
00076340
00076350
00076360
00076370
00076380
00076390
00076400
00076410
00076420
00076430
00076440
00076450
00076460
00076470
00076480
00076490
00076500

```

```

1*(DA(19),PVPC) 00076510
C 00076520
COMMON/CRG/ PI,PI4,RC,BETA 00076530
COMMON/CFG/ IDB, IDF, IDW, IDP, IDN, IG1(5), NKB, NKF, NKW, NKP, NKN 00076540
COMMON/ CPP/ DPP(3), DL, DPP1(3), NVC, NVS, NJC, NUP 00076550
C 00076560
LARGE CKV(1000), DM1(74130) 00076570
1 ,EV(22) ,XLV(22) ,XTLV(22) ,WPE(39) ,DP1(1400) 00076580
2 ,CFJ(20) ,STH(20) ,ACT(20) ,DP2(466) 00076590
3 ,CSJ(20,20),TSJ(20,20),UV(20,20) ,UVP(20,20) 00076600
4 ,VV(20,20) ,VVP(20,20),DPP2(220) ,BS(360) 00076610
5 ,DM2(4750) ,XF(20,10) ,BE(20,20) ,CF(20,20) ,CA(20,10) 00076620
6 ,TA(20,11) ,TB(20,10) ,TPTE(20) ,TPLE(20) ,PF(20,10) 00076630
7 ,UVU(20,20),UVL(20,20),VVU(20,20),VVL(20,20) 00076640
8 ,CPU(20,20),CPL(20,20),CPN(20,20),CPO(20,20) 00076650
9 ,AX(250) ,AY(250) ,AZ(250) ,XOCP(20) 00076660
C 00076670
C* INITIAL THETA AND TRIG. FUNC. 00076680
DO 20 J=1,NJC 00076690
IJ=PJC(J)+1. 00076700
THE=ACT(IJ) 00076710
THE2=THE/2.0 00076720
XOCP(J)=CFJ(IJ) 00076730
TA(J,1)=COS(THE2)/SIN(THE2) 00076740
CF(J,1)=1.0 00076750
C=0. 00076760
DO 10 N=1,NUP 00076770
C=C+1. 00076780
TA(J,N+1)=SIN(C*THE) 00076790
CA(J,N)=COS(C*THE) 00076800
10 CF(J,N+1)=CA(J,N) 00076810
TB(J,1)=TA(J,2)+THE 00076820
TB(J,2)=THE2-TA(J,3)/4. 00076830
DO 20 N=3,NUP 00076840
D2=N+N 00076850
D1=D2-4. 00076860
20 TB(J,N)=TA(J,N-1)/D1-TA(J,N+1)/D2 00076870
IF(DA(3000).EQ.0.0) GO TO 200 00076880
C CALC SLOPE OF CAMBER SURFACE 00076890
NA=0 00076900
DO 30 I=1,NVS 00076910
DO 30 J=1,NJC 00076920
NA=NA+1 00076930

```

313

30	BE(J,I)=BS(NA)	00076940
	CALL MSOLX (NUP,NJC,NVS,CF,BE,20)	00076950
C*	CALC DERIVATIVES OF F VALUES	00076960
	IC=NKB+NKF+NKW	00076970
	DO 36 IU=1,NUP	00076980
	DO 36 IE=1,NVS	00076990
	IC=IC+1	00077000
36	XF(IE,IU)=CKV(IC)/BETA	00077010
	DLA=.625*DL	00077020
	DLB=DLA+DL	00077030
	DLC=DL+DL	00077040
	K=NKB+NKF+NKW	00077050
	DO 40 IU=1,NUP	00077060
	K=K+2	00077070
	PF(1,IU)=(CKV(K-1)-CKV(K))/DLA	00077080
	PF(2,IU)=(CKV(K-1)-CKV(K+1))/DLB	00077090
	DO 40 IE=3,NVS	00077100
	K=K+1	00077110
	IF(IE.NE.NVS) GO TO 38	00077120
	PF(IE,IU)=0.0	00077130
	GO TO 40	00077140
38	PF(IE,IU)=(CKV(K-1)-CKV(K+1))/DLC	00077150
40	CONTINUE	00077160
C*	CALC SLOPE OF CAMBER SURFACE	00077170
	DO 46 IE=1,NVS	00077180
	DO 46 J=1,NJC	00077190
	SUMA=0.0	00077200
	DO 42 N=2,NUP	00077210
42	SUMA=CF(N,IE)*CA(J,N-1)+SUMA	00077220
	SUMB=0.0	00077230
	C=0.0	00077240
	DO 44 N=3,NUP,2	00077250
	C=C+1.	00077260
44	SUMB=CF(N,IE)/(4.*C**2-1.)+SUMB	00077270
46	CSJ(IE,J)=-SUMA-SUMB	00077280
C*	TPS FOR TAN PHI STAR	00077290
	CALL TPS (PHT,WPE,NVS,EV,TPTE,TPLE)	00077300
C*	CALC INFLUENCE OF OTHER COMPONENTS ON PYLON	00077310
	DO 147 IE=1,NVS	00077320
	DO 147 J=1,NJC	00077330
	UVL(IE,J)=0.0	00077340
147	VVL(IE,J)=0.0	00077350
C		00077360

	IF(IDB.EQ.0) GO TO 148	00077370
	DO 47 IE=1,NVS	00077380
	DO 47 J=1,NJC	00077390
	UVL(IE,J)=0.0	00077400
	VVL(IE,J)=0.0	00077410
	READ (19) (AX(IS),AY(IS),AZ(IS),IS=1,NKB)	00077420
	DO 47 IS=1,NKB	00077430
	UVL(IE,J)=UVL(IE,J)+AX(IS)*CKV(IS)	00077440
	47 VVL(IE,J)=VVL(IE,J)+AY(IS)*CKV(IS)	00077450
C		00077460
148	IF(IDF.EQ.0) GO TO 150	00077470
	DO 48 IE=1,NVS	00077480
	DO 48 J=1,NJC	00077490
	READ (20) (AX(IS),AY(IS),AZ(IS),IS=1,NKF)	00077500
	IC=NKB	00077510
	DO 48 IS=1,NKF	00077520
	IC=IC+1	00077530
	UVL(IE,J)=UVL(IE,J)+AX(IS)*CKV(IC)	00077540
	48 VVL(IE,J)=VVL(IE,J)+AY(IS)*CKV(IC)	00077550
C		00077560
150	IF(IDW.EQ.0) GO TO 151	00077570
	DO 50 IE=1,NVS	00077580
	DO 50 J=1,NJC	00077590
	READ (21) (AX(IS),AY(IS),AZ(IS),IS=1,NKW),WUV,WVV	00077600
	IC=NKB+NKF	00077610
	UVL(IE,J)=UVL(IE,J)+WUV	00077620
	VVL(IE,J)=VVL(IE,J)+WVV	00077630
	DO 50 IS=1,NKW	00077640
	IC=IC+1	00077650
	UVL(IE,J)=UVL(IE,J)+AX(IS)*CKV(IC)	00077660
	50 VVL(IE,J)=VVL(IE,J)+AY(IS)*CKV(IC)	00077670
C		00077680
151	IF(IDP.EQ.0) GO TO 152	00077690
	DO 51 IE=1,NVS	00077700
	DO 51 J=1,NJC	00077710
	READ (22) (AX(IS),AY(IS),AZ(IS),IS=1,NKP),PUV,PVV	00077720
	IC=NKB+NKF+NKW	00077730
	UVL(IE,J)=UVL(IE,J)+PUV	00077740
	VVL(IE,J)=VVL(IE,J)+PVV	00077750
	DO 51 IS=1,NKP	00077760
	IC=IC+1	00077770
	UVL(IE,J)=UVL(IE,J)+AX(IS)*CKV(IC)	00077780
	51 VVL(IE,J)=VVL(IE,J)+AY(IS)*CKV(IC)	00077790

C		00077800
	152 IF(IDN.EQ.0) GO TO 55	00077810
	DO 52 IE=1,NVS	00077820
	DO 52 J=1,NJC	00077830
	READ (9) (AX(IS),AY(IS),AZ(IS),IS=1,NKN)	00077840
	IC=NKB+NKF+NKW+NKP	00077850
	DO 52 IS=1,NKN	00077860
	IC=IC+1	00077870
	UVL(IE,J)=UVL(IE,J)+AX(IS)*CKV(IC)	00077880
	52 VVL(IE,J)=VVL(IE,J)+AY(IS)*CKV(IC)	00077890
C*	CALC. VELOCITIES AND PRESSURES	00077900
	55 PS8=16.*PHT	00077910
	BSQ=BETA**2	00077920
	DO 80 IE=1,NVS	00077930
	C1=PS8/XTLV(IE)	00077940
	C2=C1/4.	00077950
	DTP=TPTE(IE)-TPLE(IE)	00077960
	DO 80 J=1,NJC	00077970
	UVB=UVL(IE,J)/BETA	00077980
	VVB=VVL(IE,J)	00077990
	TP=TPLE(IE)+DTP*XOCP(J)	00078000
	TPX=TP*TP	00078010
	TP1=1.+TPX	00078020
	TP2=SQRT(TP1)	00078030
	TPU=TP2*(UV(IE,J)-UVP(IE,J))+1.	00078040
	TPV=TP2*(VV(IE,J)-VVP(IE,J))+1.	00078050
	DZU=TP1*(TSJ(IE,J)-CSJ(IE,J))**2	00078060
	DZL=TP1*(TSJ(IE,J)+CSJ(IE,J))**2	00078070
	DSU=SQRT(1.+DZU)	00078080
	DSL=SQRT(1.+DZL)	00078090
	CPS=0.	00078100
	VVS=0.	00078110
	DO 58 N=2,NUP	00078120
	CPS=CPS+XF(IE,N)*TA(J,N)	00078130
	58 VVS=VVS+PF(IE,N)*TB(J,N)	00078140
	CP1=XF(IE,1)*TA(J,1)	00078150
	CPO(IE,J)=C1*(CP1+CPS)	00078160
	UVS=C2*(CP1*TPU+CPS)	00078170
	UVU(IE,J)=(UV(IE,J)+UVS+(1.+TPX*DSU)/TP1)/DSU+UVB/DSU	00078180
	UVL(IE,J)=(UV(IE,J)-UVS+(1.+TPX*DSL)/TP1)/DSL+UVB/DSL	00078190
	VV1=2.*PF(IE,1)*TB(J,1)-C2*CP1*TP	00078200
	VVS=VV1*TPV+2.*VVS-C2*CPS*TP	00078210
	TTP=TP/TP1	00078220

	VVU(IE,J)=(VV(IE,J)+VVS-TTP*(1.-DSU))/DSU*BETA+VVB/DSU	00078230
	VVL(IE,J)=(VV(IE,J)-VVS-TTP*(1.-DSL))/DSL*BETA+VVB/DSL	00078240
	CPU(IE,J)=1.-UVU(IE,J)**2-VVU(IE,J)**2	00078250
	CPL(IE,J)=1.-UVL(IE,J)**2-VVL(IE,J)**2	00078260
C	COMPRESSIBILITY EFFECTS	00078270
	IF(XMACH.LE.0.0) GO TO 78	00078280
	XMS=XMACH**2	00078290
	XM1=1.42857/XMS	00078300
	XM2=0.2*XMS	00078310
	IF(XMACH.GT.0.01) GO TO 62	00078320
	CPU(IE,J)=CPU(IE,J)+XMS*CPU(IE,J)/4.0	00078330
	CPL(IE,J)=CPL(IE,J)+XMS*CPL(IE,J)/4.0	00078340
	GO TO 70	00078350
62	CPU(IE,J)=XM1*(1.+XM2*CPU(IE,J))**3.5-XM1	00078360
	CPL(IE,J)=XM1*(1.+XM2*CPL(IE,J))**3.5-XM1	00078370
C		00078380
70	SINP=TP/TP2	00078390
	COSP=1./TP2	00078400
	CP2=COSP**2	00078410
	SP2=SINP**2	00078420
	SCP=SINP*COSP	00078430
	CPMA=XMACH*COSP	00078440
	CPM2=CPMA**2	00078450
	SRCM=SQRT(1.-CPM2)	00078460
	CSR=.5*(1.-SRCM)	00078470
	CPU(IE,J)=CPU(IE,J)*SRCM/(1.-CPU(IE,J)*CSR)	00078480
	CPL(IE,J)=CPL(IE,J)*SRCM/(1.-CPL(IE,J)*CSR)	00078490
	BWU=SQRT(1.-CPM2*(1.-CPU(IE,J)*CPMA))	00078500
	BWL=SQRT(1.-CPM2*(1.-CPL(IE,J)*CPMA))	00078510
	DVMU=UV(IE,J)+UVS+UVB	00078520
	DVML=UV(IE,J)-UVS+UVB	00078530
	DVTU=(VV(IE,J)+VVS)*BETA+VVB	00078540
	DVTL=(VV(IE,J)-VVS)*BETA+VVB	00078550
	SBDU=SQRT(BWU**2+DZU)	00078560
	SBDL=SQRT(BWL**2+DZL)	00078570
	UVU(IE,J)=(BWU*CP2+(BSQ*DVMU*CP2-BETA*DVTU*SCP)/SRCM)/SBDU	00078580
*	+ (1.+DVMU)*SP2+DVTU*SCP	00078590
	VVL(IE,J)=(BWL*CP2+(BSQ*DVML*CP2-BETA*DVTL*SCP)/SRCM)/SBDL	00078600
*	+ (1.+DVML)*SP2+DVTL*SCP	00078610
	VVU(IE,J)=-(BWU*SCP+(BSQ*DVMU*SCP-BETA*DVTU*SP2)/SRCM)/SBDU	00078620
*	+ (1.+DVMU)*SCP+DVTU*CP2	00078630
	VVL(IE,J)=-(BWL*SCP+(BSQ*DVML*SCP-BETA*DVTL*SP2)/SRCM)/SBDL	00078640
*	+ (1.+DVML)*SCP+DVTL*CP2	00078650

	CPL(IE,J)=1.-UVL(IE,J)**2-VVL(IE,J)**2	00078660
	CPU(IE,J)=1.-UVU(IE,J)**2-VVU(IE,J)**2	00078670
	IF(XMACH.GT.0.01) GO TO 72	00078680
	CPU(IE,J)=CPU(IE,J)+XMS*CPU(IE,J)/4.0	00078690
	CPL(IE,J)=CPL(IE,J)+XMS*CPL(IE,J)/4.0	00078700
	GO TO 78	00078710
	72 CPU(IE,J)=XM1*(1.+XM2*CPU(IE,J))**3.5-XM1	00078720
	CPL(IE,J)=XM1*(1.+XM2*CPL(IE,J))**3.5-XM1	00078730
	78 CPN(IE,J)=CPL(IE,J)-CPU(IE,J)	00078740
	80 CONTINUE	00078750
C	PRINT	00078760
	WRITE (6,100)	00078770
	IS=PVPC+1.0	00078780
	DO 90 IE=IS,NVS	00078790
	90 WRITE (6,120) EV(IE),(XOCP(J),UVU(IE,J),UVL(IE,J),VVU(IE,J),	00078800
	1 VVL(IE,J),CPU(IE,J),CPL(IE,J),CPN(IE,J),CPO(IE,J),J=1,NJC)	00078810
	100 FORMAT (54H1 *PYLON NON-LINEAR VELOCITY AND PRESSURE COEFFICIENTS)	00078820
	120 FORMAT (8H0 ETA =,F7.4/1H ,7X,3HX/C,7X,6HU/V UP,6X,7HU/V LOW,5X,	00078830
	1 6HV/V UP,6X,7HV/V LOW,6X,5HCP UP,7X,6HCP LOW,6X,6HCP L-U,	00078840
	2 5X,7HCP NETL/(1H ,9F12.5))	00078850
	GO TO 300	00078860
C	CALC LINEAR PRESSURE COEFFICIENTS AND SKIP RECORDS	00078870
	200 PSB=16.*PHT	00078880
	DO 220 IE=1,NVS	00078890
	C1=PSB/XTLV(IE)	00078900
	DO 220 J=1,NJC	00078910
		00078920
C	CPS=0.0	00078930
	DO 210 N=1,NUP	00078940
	210 CPS=CPS+XF(IE,N)*TA(J,N)	00078950
	220 CPU(IE,J)=C1*CPS	00078960
		00078970
C	300 RETURN	00078980
	END	00078990
*DECK	PLOAD	00079000
	SUBROUTINE PLOAD (PCL,PCD,PCM)	00079010
C		00079020
C*	CALC PYLON LOAD PARAMETERS	00079030
C		00079040
	COMMON/DAT/ DA(5000)	00079050
C		00079060
	DIMENSION PJC(1)	00079070
C		00079080

```

EQUIVALENCE      (DA(1),AR)      ,(DA(2),SPAN)      ,(DA(3),XMAC)      00079090
1,(DA(7),XCG)    ,(DA(19),PVPC)    ,(DA(2507),PNVC), (DA(2512),PJC) 00079100
2,(DA(2492),PDA) ,(DA(2501),POX)                                00079110
C                                                         00079120
COMMON/CRG/ PI,PI4,RC                                00079130
COMMON/PPP/ HPS,NVSH,NVT,DL,NEVP,ZCON,YCON,NVC,NVS,NJC,NUP 00079140
C                                                         00079150
LARGE            DM1(75130),EV(22)      ,XLV(22)      ,XILV(22)      ,DM2(219) 00079160
1                ,XR(400)      ,YR(400)      ,ZR(400)      ,CFI(20)      ,CFJ(20) 00079170
2                ,DM3(906)      ,TSJ(20,20)    ,DM4(1820)    ,BS(360)      ,DM5(4950) 00079180
3                ,DSS(20,20)    ,DM6(2860)    ,CPU(20,20)  ,CPL(20,20) 00079190
4                ,THI(22)      ,CNI(22)      ,CMI(22)      ,CDI(22)      ,STH(20) 00079200
5                ,THO(20)      ,CNO(20)      ,CMO(20)      ,CDO(20)      ,XCCP(20) 00079210
6                ,EVO(20)      ,XLE(20)      ,CNCC(20)     ,CMCC(20)     ,CDCC(20) 00079220
C                                                         00079230
C*                CALC DS/DSLE OF ROOT SECTION          00079240
DSLE=SQRT((YR(1)-YR(2))**2+(ZR(1)-ZR(2))**2)          00079250
DO 10 IJS=1,NVS                                        00079260
IEP=IJS-NEVP                                          00079270
DO 10 IJC=1,NJC                                        00079280
DSS(IJS,IJC)=1.0                                      00079290
IF(IEP.LE.0) GO TO 10                                 00079300
IX=PJC(IJC)                                           00079310
IC=4*IX+IEP                                           00079320
DY=(YR(IC)-YR(IC+1)+YR(IC-4)-YR(IC-3))/2.           00079330
DZ=(ZR(IC)-ZR(IC+1)+ZR(IC-4)-ZR(IC-3))/2.           00079340
DSS(IJS,IJC)=SQRT(DY**2+DZ**2)/DSLE                  00079350
10 CONTINUE                                           00079360
C                                                         00079370
C*                CALC CHORDWISE THETAS                  00079380
DTH=PI/PNVC                                           00079380
CONS=.5*AR/SPAN*DTH                                   00079390
THO(1)=DTH                                             00079400
DO 20 I=2,NVC                                         00079410
20 THO(I)=THO(I-1)+DTH                                00079420
C                                                         00079430
C                                                         00079440
THI(1)=0.0                                            00079440
DO 30 IC=1,NJC                                        00079450
THI(IC+1)=PJC(IC)*DTH                                00079460
30 STH(IC)=SIN(THI(IC+1))                             00079470
NJI=NJC+2                                             00079480
THI(NJI)=PI                                           00079490
C                                                         00079500
C*                CALC LIFT,DRAG AND MOMENT COEFFICIENTS 00079510

```

	IV=NVS-PVPC	00079520
	NA=0	00079530
	CNI(1)=0.0	00079540
	CMI(1)=0.0	00079550
	CDI(1)=0.0	00079560
	CNI(NJI)=0.0	00079570
	CMI(NJI)=0.0	00079580
	CDI(NJI)=0.0	00079590
C		00079600
	DO 90 IE=1,NVS	00079610
	IF(IE.GT.PVPC) GO TO 40	00079620
	NA=NA+NJC	00079630
	GO TO 90	00079640
40	DO 70 IC=1,NJC	00079650
	NA=NA+1	00079660
C		00079670
	CALC TAN ALPHA AND CP NET	00079680
	IF(DA(3000).EQ.0.0) GO TO 50	00079690
	TAU=-TSJ(IE,IC)+BS(NA)	00079700
	TAL=TSJ(IE,IC)+BS(NA)	00079710
	CPN=CPL(IE,IC)-CPU(IE,IC)	00079720
	CPD=TAL*CPL(IE,IC)-TAU*CPU(IE,IC)	00079730
	GO TO 60	00079740
C		00079750
50	CPN=CPU(IE,IC)	00079760
	CPD=BS(NA)*CPU(IE,IC)	00079770
60	IX=PJC(IC)	00079780
	CPM=CPN*CFJ(IX+1)*XTLV(IE)	00079790
	DST=DSS(IE,IC)*STH(IC)	00079800
	CNI(IC+1)=DST*CPN	00079810
	CMI(IC+1)=DST*CPM	00079820
70	CDI(IC+1)=DST*CPD	00079830
C		00079840
	CHORDWISE EXPANSION	00079850
	CALL CODIM (THI,CNI,NJI,THO,CNO,NVC)	00079860
	CALL CODIM (THI,CMI,NJI,THO,CMO,NVC)	00079870
	CALL CODIM (THI,CDI,NJI,THO,CDO,NVC)	00079880
C		00079890
	SUM,CN,CM AND CD CHORDWISE	00079900
	CN=0.0	00079910
	CM=0.0	00079920
	CD=0.0	00079930
	DO 80 I=1,NVC	00079940
	CN=CN+CNO(I)	
	CM=CM+CMO(I)	
80	CD=CD+CDO(I)	

C	SET PARAMETERS IN ASCENDING ORDER	00079950
	EVO(IV)=EV(IE)	00079960
	CC=CONS*XTLV(IE)	00079970
	CNCC(IV)=CC*CN	00079980
	CMCC(IV)=-CC*CM/XMAC	00079990
	CDCC(IV)=CC*CD	00080000
	XLE(IV)=XLV(IE)	00080010
	IV=IV-1	00080020
90	CONTINUE	00080030
C	CALC TOTAL PYLON LIFT ,DRAG AND MOMENT	00080040
	PCL=0.0	00080050
	PCM=0.0	00080060
	PCD=0.0	00080070
	NEE=NVS-PVPC	00080080
	CDA=COS(190.-PDA)/RC	00080090
C		00080100
	DO 110 I=1,NEE	00080110
	IF(ABS(CNCC(I)).LT.0.000001) CNCC(I)=0.000001	00080120
	XCCP(I)=-CMCC(I)/CNCC(I)	00080130
	XARM=XCG-POX-XLE(I)	00080140
	CMCG=CMCC(I)+XARM*CNCC(I)	00080150
	CNX=CNCC(I)*DL	00080160
	CMX=CMCG*DL	00080170
	CDX=CDCC(I)*DL	00080180
	IF(I.NE.NVS) GO TO 100	00080190
	CNX=CNX/3.	00080200
	CMX=CMX/3.	00080210
	CDX=CDX/3.	00080220
100	PCL=PCL+CNX*CDA	00080230
	PCM=PCM+CMX	00080240
110	PCD=PCD+CDX	00080250
	IF(ABS(PCL).LT.0.000001) PCL=0.000001	00080260
	PXC=-PCM/PCL	00080270
C*	PRINT OUTPUT	00080280
C		00080290
	WRITE (6,300) (I,EVO(I),CNCC(I),CDCC(I),CMCC(I),XCCP(I),I=1,NEE)	00080300
300	FORMAT (1H1,20X,18H ** PYLON LOADS **/1H ,24X,3HETA,12X,6HCNC/CA,	00080310
1	9X,6HCDC/CA,9X,6HCMC/CA,9X,6HX/C CP/(1H ,115,5F15.5))	00080320
C		00080330
	WRITE (6,310) PCL,PCD,PCM,PXC	00080340
310	FORMAT (1H0,20X,23H ** TOTAL PYLON LOADS **/1H ,39X,3HPCL,12X,3HPCD	00080350
1	,12X,3HPCM,12X,3HX/C/1H ,30X,4F15.5)	00080360
C		00080370

	RETURN	00080380
	END	00080390
*DECK	SFC	00080400
	SUBROUTINE SFC (NETA,ETA)	00080410
C		00080420
C*	CALC MATRIX F	00080430
C		00080440
	COMMON/DAT/ DA(5000)	00080450
C		00080460
	DIMENSION W(1)	00080470
C		00080480
	LARGE ETA(1)	00080490
C		00080500
	EQUIVALENCE (DA(1285),W) ,(DA(1281),WNW)	00080510
C		00080520
	COMMON/CRG/ PI,PI4,RC,BETA	00080530
	COMMON/CFG/ DF1(10),NKB,NKF	00080540
	COMMON/CPW/ DW1(9),NU,NW	00080550
	LARGE CKV(1000) ,BOU(2000) ,FUS(25000),FAN(23700)	00080560
1	,WIN(23430),PYL(6061) ,CEL(3700)	00080570
2	,XF(51,10) ,SRE(51) ,S(51,10)	00080580
C		00080590
	DO 10 J=1,NETA	00080600
10	SRE(J)=SQRT(1.-ETA(J)**2)	00080610
	IF(WNW.NE.0.0) GO TO 500	00080620
	K=NKB+NKF	00080630
	DO 20 J=1,NU	00080640
	DO 20 NE=1,NETA	00080650
	K=K+1	00080660
20	XF(NE,J)=CKV(K)/BETA	00080670
	GO TO 600	00080680
500	DO 550 I=1,NW	00080690
	L=W(I)	00080700
	IF(L.LT.1000) GO TO 510	00080710
	CALL PFUNC (I,NETA,ETA,W,S)	00080720
	GO TO 550	00080730
510	DO 540 J=1,NETA	00080740
	IF(ETA(J).GT.0.00001) GO TO 518	00080750
	S(J,I)=0.0	00080760
	IF(L.EQ.0) S(J,I)=1.0	00080770
	GO TO 540	00080780
518	S(J,I)=SRE(J)*ETA(J)**L	00080790
540	CONTINUE	00080800

	550 CONTINUE	00080810
C	CALC F MATRIX	00080820
	560 DO 575 NE=1,NETA	00080830
	K=NKB+NKF	00080840
	DO 572 J=1,NU	00080850
	XF(NE,J)=0.0	00080860
	DO 570 I=1,NW	00080870
	K=K+1	00080880
	570 XF(NE,J)=XF(NE,J)+CKV(K)*S(NE,I)	00080890
	572 XF(NE,J)=XF(NE,J)/BETA	00080900
	575 CONTINUE	00080910
	600 RETURN	00080920
	END	00080930
*DECK	TPS	00080940
	SUBROUTINE TPS (HWS,WT,NEP,EP,TPTE,TPLE)	00080950
C		00080960
C*	*TPS* CALC. LEADING AND TRAILING EDGE TAN PHI STAR	00080970
C		00080980
	LARGE WT(1),EP(1),TPTE(1),TPLE(1)	00080990
C		00081000
	LARGE DMI(90000)	00081010
I	,YB(30) ,CB(30) ,PBL(30) ,PBT(30)	00081020
C		00081030
C*	CALC. YB,CB AND PHI FROM PLANFORM	00081040
	IC=0	00081050
	DO 10 L=1,59,3	00081060
	IC=IC+1	00081070
	YB(IC)=HWS*WT(L)	00081080
	CB(IC)=WT(L+2)-WT(L+1)	00081090
	IF(WT(L).GE.1.0) GO TO 20	00081100
	DY=HWS*(WT(L+3)-WT(L))	00081110
	PBL(IC)=ATAN((WT(L+4)-WT(L+1))/DY)	00081120
10	PBT(IC)=ATAN((WT(L+5)-WT(L+2))/DY)	00081130
20	PBL(IC)=-PBL(IC-1)	00081140
	PBT(IC)=-PBT(IC-1)	00081150
	NB=IC	00081160
C*	CALC. TPLE AND TPTE	00081170
	DO 110 IP=1,NEP	00081180
	YP=HWS*EP(IP)	00081190
	DO 40 IB=2,NB	00081200
	IF(YP.LE.YB(IB)) GO TO 50	00081210
40	CONTINUE	00081220
50	IX=IB	00081230

IF(IX.GT.2) GO TO 60	00081240
APL=0.	00081250
APT=0.	00081260
GO TO 70	00081270
60 APL=(PBL(IX-1)+PBL(IX-2))/2.	00081280
APT=(PBT(IX-1)+PBT(IX-2))/2.	00081290
70 DY=YB(IX)-YB(IX-1)	00081300
IF(CB(IX-1).GE.DY) GO TO 90	00081310
ALL=PBL(IX-1)	00081320
ALT=PBT(IX-1)	00081330
YPC=YB(IX-1)+CB(IX-1)	00081340
IF(YP.GT.YPC) GO TO 80	00081350
C=CB(IX-1)	00081360
YBE=YB(IX-1)	00081370
GO TO 100	00081380
80 C=DY-CB(IX-1)	00081390
YBE=YB(IX)	00081400
APL=(PBL(IX-1)+PBL(IX))/2.	00081410
APT=(PBT(IX-1)+PBT(IX))/2.	00081420
GO TO 100	00081430
90 C=DY	00081440
YBE=YB(IX-1)	00081450
ALL=(PBL(IX-1)+PBL(IX))/2.	00081460
ALT=(PBT(IX-1)+PBT(IX))/2.	00081470
100 AYC=6.283185*ABS((YP-YBE)/C)	00081480
IF(ALL.NE.0.0) GO TO 102	00081490
TLY=AYC	00081500
GO TO 104	00081510
102 TLY=TAN(ALL)/ALL*AYC	00081520
104 SLB=SQRT(1.+TLY**2)-TLY	00081530
CL=COS(ALL)	00081540
CLP=COS(SLB*APL)	00081550
CL2=COS(SLB*ALL)**2	00081560
TPLE(IP)=SQRT(CL2-CL**2*CLP**2)/(CL*CLP)	00081570
IF(ALT.NE.0.0) GO TO 106	00081580
TLY=AYC	00081590
GO TO 108	00081600
106 TLY=TAN(ALT)/ALT*AYC	00081610
108 SLB=SQRT(1.+TLY**2)-TLY	00081620
CL=COS(ALT)	00081630
CLP=COS(SLB*APT)	00081640
CL2=COS(SLB*ALT)**2	00081650
110 TPTE(IP)=SQRT(CL2-CL**2*CLP**2)/(CL*CLP)	00081660

	RETURN	00081670
	END	00081680
*DECK	SPF	00081690
	SUBROUTINE SPF	00081700
C		00081710
C*	*SPF* CALC. THE DERIVATIVES OF F VALUES	00081720
C		00081730
	COMMON/DAT/ DA(5000)	00081740
	DIMENSION W(10) ,WD(25)	00081750
C		00081760
	EQUIVALENCE (DA(1285),W) ,(DA(1281),WNW) ,(DA(1930),WD)	00081770
	1,(NEP,NJS)	00081780
C		00081790
	COMMON/CRG/ PI,PI4,RC,BETA	00081800
	COMMON/CFG/ DF1(10),NKB,NKF	00081810
	COMMON/CPW/ HWS,NTI,NTO,DLI,DLO,NE2,NE3	00081820
	1 ,NE5,NTB,NU,NW,NVC,NVS,NJC,NJS	00081830
C		00081840
	LARGE CKV(1000) ,BOU(2000) ,FUS(25000),FAN(23700)	00081850
	1 ,DM1(1706) ,EP(30) ,DM2(21694),PYL(6061) ,CEL(3700)	00081860
	2 ,DM3(510) ,SRE(51)	00081870
	3 ,DM4(3330) ,PF(30,10) ,EPO(30),ES(4),CS(4),S(30,10)	00081880
C		00081890
C*	CALC SLOPES OF SPANWISE LOAD SHAPES	00081900
	IF(WNW.EQ.0.0) GO TO 170	00081910
	IC=1	00081920
	DO 150 IW=1,NW	00081930
	L=W(IW)+1.1	00081940
	IF(L.GT.1000) GO TO 30	00081950
C		00081960
	DO 20 IE=1,NEP	00081970
	EPL=EP(IE)**L	00081980
	S(IE,IW)=-EPL/SRE(IE)	00081990
	IF(IW.EQ.1) GO TO 20	00082000
	S(IE,IW)=S(IE,IW)+SRE(IE)*W(IW)*EPO(IE)	00082010
	20 EPO(IE)=EPL	00082020
	GO TO 150	00082030
C		00082040
	30 IF(WD(IC).EQ.0.0) GO TO 60	00082050
	ES(1)=WD(IC+2)	00082060
	ES(2)=WD(IC+3)	00082070
	CS(1)=1.	00082080
	CS(2)=-1.	00082090

GO TO 90	00082100
60 ETAB=WD(IC+1)	00082110
RI=WD(IC+2)	00082120
RO=WD(IC+3)	00082130
ES(1)=ETAB-RI	00082140
ES(2)=ETAB	00082150
ES(3)=ETAB+RO	00082160
EB1=1.-ETAB	00082170
ERO=EB1-RO	00082180
RR=(RI+RO)/(RI*RO)	00082190
IF(ES(1).GE.0.0) GO TO 70	00082200
ES(1)=0.0	00082210
ES(4)=0.0	00082220
CS(1)=1./RI	00082230
CS(2)=-RR*EB1	00082240
CS(3)=(RR-CS(1))*ERO	00082250
CS(4)=1.-ETAB/RI	00082260
NP=4	00082270
GO TO 100	00082280
70 IF(ERO.LT.0.0) GO TO 80	00082290
CS(1)=EB1/RI+1.	00082300
CS(2)=-RR*EB1	00082310
CS(3)=EB1/RO-1.	00082320
NP=3	00082330
GO TO 100	00082340
80 CS(1)=EB1/RI+1.	00082350
CS(2)=-CS(1)	00082360
90 NP=2	00082370
100 DO 145 IE=1,NP	00082380
S(IE,IW)=0.0	00082390
ETA=EP(IE)	00082400
IF(ETA.EQ.0.0) GO TO 145	00082410
ES2=2.*ETA/SRE(IE)	00082420
DO 140 K=1,NP	00082430
AES=ACOS(ES(K))	00082440
SES=SQRT(1.-ES(K)**2)	00082450
IF(SES.EQ.SRE(IE)) SES=SES+.000001	00082460
E1=ES(K)*SRE(IE)	00082470
E2=ETA *SES	00082480
E=ABS((E1-E2)/(E1+E2))	00082490
IF(K.EQ.4) GO TO 110	00082500
IF(WD(IC).EQ.0.0) GO TO 120	00082510
110 PS=(ALOG(1./E)-ES2*AES)/PI	00082520

	GO TO 140	00082530
120	IF(ABS(ES(K)-ETA).GT.0.000001) GO TO 130	00082540
	EX=-ES(K)*ALOG(ES(K)**2)	00082550
	GO TO 135	00082560
130	EX=ES(K)*ALOG(E)+ETA*ALOG(ABS((SES+SRE(IE))/(SES-SRE(IE))))	00082570
135	PS=(EX+ES2*(ES(K)*AES-SES))/(PI*(1.-ES(K)))	00082580
140	S(IE,IW)=S(IE,IW)+CS(K)*PS	00082590
145	CONTINUE	00082600
	IC=IC+5	00082610
150	CONTINUE	00082620
C*	CALC THE DERIVATIVES OF F VALUES	00082630
	DO 160 IE=1,NEP	00082640
	K=NKB+NKF	00082650
	DO 160 IU=1,NU	00082660
	PF(IE,IU)=0.0	00082670
	DO 160 IW=1,NW	00082680
	K=K+1	00082690
160	PF(IE,IU)=PF(IE,IU)+CKV(K)*S(IE,IW)	00082700
	GO TO 220	00082710
C	FIND SLOPE OF COEF. IF SPANWISE DISCRETE VORTEX	00082720
170	DLA=.625*DLO	00082730
	DLB=DLA+DLO	00082740
	NE7=NE3+4	00082750
	NE6=NE3+3	00082760
	K=NKB+NKF	00082770
	DO 214 IU=1,NU	00082780
	K=K+2	00082790
	PF(1,IU)=(CKV(K-1)-CKV(K))/DLA	00082800
	PF(2,IU)=(CKV(K-1)-CKV(K+1))/DLB	00082810
	DO 212 IE=3,NJS	00082820
	K=K+1	00082830
	IF(IE,NE,NJS) GO TO 200	00082840
	PF(IE,IU)=0.0	00082850
	GO TO 212	00082860
200	IF(IE-NE6) 210,208,202	00082870
202	IF(IE,NE,NE7) GO TO 204	00082880
	PF(IE,IU)=(CKV(K)-CKV(K+1))/DLI	00082890
	GO TO 212	00082900
204	PF(IE,IU)=(CKV(K-1)-CKV(K+1))/(DLI+DLI)	00082910
	GO TO 212	00082920
208	PF(IE,IU)=(CKV(K-1)-CKV(K))/DLO	00082930
	GO TO 212	00082940
210	PF(IE,IU)=(CKV(K-1)-CKV(K+1))/(DLO+DLO)	00082950

```

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

```

```

32=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 ABSCISSA 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)=CNCCO(I) * DLI * HRA
GO TO 50

```

```

C
IV=IV-1 ← 30 CLS(IV)=FCL * HRA

```

```

C
← 40 CLS(IV)=CNCCO(I) * DLO * HRA
IF(I.EQ.NE2) CLS(IV)=CLS(IV)/3.

```

```

00082960
00082970
00082980
00082990
00083000
00083010
00083020
00083030
00083040
00083050
00083060
00083070
00083080
00083090
00083100
00083110
00083120
00083130
00083140
00083150
00083160
00083170
00083180
00083190 83195
00083200
00083210
00083220
00083230
00083240
00083250
00083260
00083270
00083280
00083290 83295
00083300
00083310
00083320
00083330 83335
00083340
00083350 83355
00083360
00083370 83375
00083380

```

C		00083390
	50 IV=IV-1	00083400
	60 CONTINUE	00083410
C		00083420
	NVT=NVT-1	00083430
	DO 70 I=1,NVT	00083440
	70 EOC(I)=(1.-ET(I))/2.	00083450
C*	SETUP SECTION LOADS AND ABSCISSA ON RIGHT SIDE	00083460
	NN=NVT+NVT	00083470
	IV=NVT	00083480
	IS=NVT+1	00083490
	DO 80 I=IS,NN	00083500
	EOC(I)=1.-EOC(IV)	00083510
	CLS(I+1)=CLS(IV)	00083520
	80 IV=IV-1	00083530
C*	CALC EMINTON-LORD Q AND P MATRICES	00083540
	CALL EMLQP (NN)	00083550
C*	INTEGRATE SECTION LOADS CURVE	00083560
	NX=NN+1	00083570
	SS(1)=CLS(1)	00083580
	DO 90 I=2,NX	00083590
	90 SS(I)=SS(I-1)+CLS(I)	00083600
	DSS=SS(NX)	00083610
C*	CALC INDUCED DRAG COEF	00083620
	DO 100 N=1,NN	00083630
	100 C(N)=SS(N)-DSS*Q(N)	00083640
C		00083650
	SUMB=0.0	00083660
	DO 120 M=1,NN	00083670
	SUMC=0.0	00083680
	DO 110 N=1,NN	00083690
	110 SUMC=SUMC+P(M,N)*C(N)	00083700
	120 SUMB=SUMB+SUMC*C(M)	00083710
C		00083720
	CDI=(4.*DSS**2/PI+SUMB*PI)/B2*.25/RAREA	00083730
C		00083740
	RETURN	00083750
	END	00083760
*DECK	EMLQP	00083770
	SUBROUTINE EMLQP(NN)	00083780
C		00083790
C	*	00083800
C	COMPUTES COEFFICIENTS FOR EMLORD EQUATION	00083810

	COMMON/CRG/ PI	00083820
C		00083830
	LARGE DUM(102624)	00083840
	LARGE XOC(102) ,Q(102) ,P(102,102),B(102,102)	00083850
		00083860
C		00083870
C	* START	00083880
C	* SETUP MATRIES P AND B	00083890
	DO 100 N=1,NN	00083900
	X=XOC(N)	00083910
	XV=1.-2.*X	00083920
	ANG=ATAN2(SQRT(1.-XV**2),XV)	00083930
	Q(N)=(ANG-2.*XV*SQRT(X-X**2))/PI	00083940
	DO 70 M=N,NN	00083950
	Y=XOC(M)	00083960
	IF(M-N) 30,20,30	00083970
20	B(M,N)=1.0	00083980
	GO TO 40	00083990
30	B(M,N)=0.0	00084000
40	E=(X-Y)**2	00084010
	E1=X+Y-2.*X*Y	00084020
	E2=2.*SQRT(X*Y*(1.-X)*(1.-Y))	00084030
	IF(E) 60,50,60	00084040
50	P(M,N)=E1*E2	00084050
	GO TO 70	00084060
60	P(M,N)=E/2.*ALOG((E1-E2)/(E1+E2))+E1*E2	00084070
70	CONTINUE	00084080
	NK=N-1	00084090
	IF(NK) 100,100,80	00084100
80	DO 90 M=1,NK	00084110
	E=P(N,M)	00084120
	P(M,N)=E	00084130
90	B(M,N)=0.0	00084140
100	CONTINUE	00084150
C	* SOLVE SIMULTANEOUS EQUATIONS FOR COEFFICIENTS	00084160
	CALL MSOLX (NN,NN,NN,P,B,102)	00084170
	RETURN	00084180
	END	00084190
*DECK	SKINF	00084200
	SUBROUTINE SKINF	00084210
C		00084220
C*	CALC SKIN FRICTION DRAG	00084230
C		00084240
	COMMON/DAT/ DA(5000)	

	B=(1.+GM2)/TRTT-1.	00084680
	DENO=SQRT(B**2+4.*A2)	00084690
	ALP=(2.*A2-B)/DENO	00084700
	BET=B/DENO	00084710
	C1=.242*(ASIN(ALP)+ASIN(BET))/SQRT(GM2)	00084720
	C2=.76*ALOG10(TRTT)	00084730
C*	LAMINAR PARAMETERS	00084740
	TRTL=1.+0.851*GM1	00084750
	TST=1.+0.72*(TRTL-1.)	00084760
	TS=TST*TFS	00084770
	XMUS=2.270E-8*TS*SQRT(TS)/(TS+198.6)	00084780
	CS=XMUS/XMU/TST	00084790
	SRCS=SQRT(CS)	00084800
C*	FLAT PLATE NATURAL TRANSITION POINT	00084810
	IF(TRID.LT.0.0) GO TO 60	00084820
C	SMOOTH	00084830
	CALL COD (10,TAUI,DRTI,1,TAU,DRT)	00084840
	RTTR=163.+DRT	00084850
	RXTR=(RTTR/.664)**2/CS	00084860
	IF(XKS.EQ.0.0) GO TO 60	00084870
C	ROUGH	00084880
	RK=R*XKS	00084890
	CALL COD (10,RKI,RXTRI,1,RK,RXTX)	00084900
	RXTR=RXTX/670000.	00084910
	RTTR=.664*SQRT(RXTR)*SRCS	00084920
C		00084930
C*	NO. OF COMPONENTS LOOP	00084940
	60 SAT=0.0	00084950
	CFT=0.0	00084960
	DO 300 I=1,5	00084970
	IF(CID(I).EQ.0.0) GO TO 300	00084980
C*	CALC FORM CORRECTION AND AREA RATIO	00084990
	XK=1.0	00085000
	IF(CID(I)-2.) 80,70,75	00085010
	70 XK=1.+CKAF*TCM(I)+60.*TCM(I)**4	00085020
	GO TO 80	00085030
	75 XK=1.+1.5*TCM(I)*SQRT(TCM(I))+7.*TCM(I)**3	00085040
	80 SRK=SA(I)/SREF*XK	00085050
C		00085060
C*	NO. OF SEGMENTS PER COMPONENT	00085070
	IF(I.EQ.1) TL=TLB	00085080
	IF(I.EQ.2) TL=TLF	00085090
	IF(I.EQ.5) TL=TLN	00085100

	NSC=NS(I)	00085110
	DO 290 IS=1,NSC	00085120
	IF(I.EQ.3) TL=TLW(IS)	00085130
	IF(I.EQ.4) TL=TLP(IS)	00085140
	TL=TL/CONV	00085150
	IF(TRID.GE.0.0) GO TO 90	00085160
	RXTR=R*XTRL(I)*TL	00085170
	RTTR=.664*SQRT(RXTR)*SRCS	00085180
C		00085190
	90 XTR=RXTR/R	00085200
	CFRDX=2.*RTTR	00085210
C*	ITERATE FOR CFS (SMOOTH)	00085220
	FP=0.0	00085230
	FM=0.0	00085240
	IP=1	00085250
	CF=.001	00085260
	CO=100.	00085270
C		00085280
	SL=TL	00085290
	IF(CFRDX.LE.0.0) GO TO 95	00085300
	CFDO=(C1/(ALOG10(CFRDX)-C2))**2	00085310
	RDX=CFRDX/CFDO	00085320
	DX=RDX/R	00085330
	SL=TL-XTR+DX	00085340
	95 RLS=R*SL	00085350
	100 IF(ABS(1.-CF/CO).LT.0.001) GO TO 170	00085360
	F=ALOG10(CF*RLS)-C1/SQRT(CF)-C2	00085370
	IF(F) 110,170,120	00085380
	110 FM=F	00085390
	CM=CF	00085400
	GO TO 130	00085410
	120 FP=F	00085420
	CP=CF	00085430
	130 IF(IP.EQ.0) GO TO 140	00085440
	IP=0	00085450
	CF=.007	00085460
	GO TO 100	00085470
	140 IF((FP*FM).LT.0.0) GO TO 160	00085480
	WRITE (6,150) I	00085490
	150 FORMAT (51H0 ** SKINF * NO SOLUTION WITHIN CFS LIMITS * COMP.=,I5)	00085500
	CFCS=0.0	00085510
	GO TO 180	00085520
		00085530

160	CO=CF	00085540
	CF=(CM+CP)/2.	00085550
	GO TO 100	00085560
C		00085570
170	CFS=CF*SL/TL*SRK	00085580
	CFR=0.0	00085590
180	IF(XKS.EQ.0.0) GO TO 280	00085600
C*	ITERATE FOR CFR (ROUGH)	00085610
	DX=0.0	00085620
	IF((XTR/TL).LT.0.0001) GO TO 270	00085630
	IP=1	00085640
	DX=XTR/2.0	00085650
C		00085660
200	F1=1.89+1.62*ALOG10(DX/XKS)	00085670
	F2=SQRT(F1)	00085680
	CF=1./((TRIT*F2*F1**2)	00085690
	DXO=DX	00085700
	RDX=CFRDX/CF	00085710
	DX=RDX/R	00085720
	IF(ABS(1.-DX/DXO).LT.0.001) GO TO 270	00085730
	IP=IP+1	00085740
	IF(IP.LE.100) GO TO 200	00085750
	WRITE (6,250) I	00085760
250	FORMAT (51H0 ** SKINF * NO SOLUTION WITHIN CFR LIMITS * COMP.=,I5)	00085770
	CFR=0.0	00085780
	GO TO 280	00085790
C		00085800
270	SL=TL-XTR+DX	00085810
	F1=1.89+1.62*ALOG10(SL/XKS)	00085820
	F2=SQRT(F1)	00085830
	CF=1./((TRIT*F2*F1**2)	00085840
	CFR=CF*SL/TL*SRK	00085850
C*	SUM SKIN FRICTION DRAG	00085860
280	CFU=CFR	00085870
	IF(CFR.LT.CFS) CFU=CFS	00085880
	CFT=CFT+CFU	00085890
290	CONTINUE	00085900
	SAT=SAT+SA(I)	00085910
C		00085920
300	CONTINUE	00085930
C		00085940
	WRITE (6,310) XMACH,TFS,PFS,XKS,TAU,SREF,SAT,CFT	00085950
310	FORMAT (23H1 ** SKIN FRICTION DRAG/1H0,4X,5HMACH=,F7.3,4X,5HTEMP=,00085960	00085960

1	F9.3,4X,6HPRESS=,F10.3,4X,3HKS=,F9.6,4X,4HTAU=,F7.4/	00085970
2	19H0 * REFERENCE AREA=,F10.3/21H0 * WET SURFACE AREA=,F10.3/	00085980
3	29H0 * TOTAL SKIN FRICTION DRAG=,F10.5)	00085990
C		00086000
	400 RETURN	00086010
	END	00086020
	*DECK SETSF	00086030
	SUBROUTINE SETSF	00086040
C		00086050
C*	SETUP SKIN FRICTION DRAG INPUT DATA	00086060
C		00086070
	COMMON/DAT/ DA(5000)	00086080
C		00086090
	DIMENSION XB1(1),YB1(1)	00086100
C		00086110
	EQUIVALENCE (DA(2),WS) ,(DA(19),PVPC) ,(DA(2500),PHT)	00086120
	1,(DA(2502),POY) ,(DA(3392),YNO) ,(DA(3400),XB1) ,(DA(3550),YB1)	00086130
	2,(DA(2492),PDA)	00086140
C		00086150
	COMMON/CFG/ IDB,IDF,IDW,IDP,IDN	00086160
	COMMON/CPB/ DB1(12),SAB	00086170
	COMMON/CPF/ DF1(12),SAF	00086180
	COMMON/CPW/ DW1(3),DLI,DLO,NE2,NE3,DW2(5),NVSW	00086190
	COMMON/CPV/ DP1(3),DLP,DP2(4),NVSP	00086200
	COMMON/CPN/ DN1,NP1	00086210
	COMMON/CSF/ CID(5),NS(5),SA(5),TLW(52),TLP(22),TLN	00086220
C		00086230
	LARGE DM1(51804) ,XTLVW(52) ,DM2(12744),XTLTW(52)	00086240
1	DM3(10522),XTLVP(22)	00086250
C		00086260
C	FUSELAGE SETUP	00086270
	CID(1)=0.0	00086280
	IF(IDB.EQ.0) GO TO 10	00086290
	CID(1)=3.0	00086300
	NS(1)=1	00086310
	SA(1)=SAB	00086320
C		00086330
	FANPOD SETUP	00086340
10	CID(2)=0.0	00086350
	IF(IDF.EQ.0) GO TO 20	00086360
	CID(2)=3.0	00086370
	NS(2)=1	00086380
	SA(2)=SAF	00086390
C		00086390
	WING SETUP	

		00086400
20	CID(3)=0.0	00086410
	IF(IDW.EQ.0) GO TO 40	00086420
	CID(3)=2.0	00086430
	NS(3)=NE2	00086440
	NE6=NE3+3	00086450
	NE1=NE2+1	00086460
	NVSH=NVSX/2	00086470
	TWS=WS+WS	00086480
	DS4=TWS*DLO	00086490
	TLW(1)=(XTLVW(NE1)+XTLTW(1))/2.0	00086500
	SA(3)=TLW(1)*DS4/2.0	00086510
	DO 30 I=2,NVSH	00086520
	IF(I.GT.NE6) DS4=TWS*DLI	00086530
	TLW(I)=XTLVW(I)	00086540
	SA(3)=SA(3)+TLW(I)*DS4	00086550
30	CONTINUE	00086560
	IF(IDB.NE.0) GO TO 40	00086570
	TLW(NE2)=(XTLVW(NE2)+XTLTW(NE2))/2.0	00086580
	SA(3)=SA(3)+TLW(NE2)*DS4/2.0	00086590
C	PYLON SETUP	00086600
40	CID(4)=0.0	00086610
	IF(IDP.EQ.0) GO TO 60	00086620
	CID(4)=2.0	00086630
	NVPC=PVPC	00086640
	NS(4)=NVSP-NVPC	00086650
	DS4=4.*PHT*DLP	00086660
	NSX=NVPC+1	00086670
	IC=0	00086680
	SA(4)=0.0	00086690
	DO 50 I=NSX,NVSP	00086700
	IC=IC+1	00086710
	TLP(IC)=XTLVP(I)	00086720
	SA(4)=SA(4)+TLP(IC)*DS4	00086730
50	CONTINUE	00086740
	IF(POY.EQ.0.0.AND.PDA.EQ.0.0) SA(4)=SA(4)/2.0	00086750
C	NACELLE SETUP	00086760
60	CID(5)=0.0	00086770
	IF(IDN.EQ.0) GO TO 70	00086780
	CID(5)=2.0	00086790
	NS(5)=1	00086800
	TLN=XB1(1)	00086810
	IF(XB1(NP1).GT.XB1(1)) TLN=XB1(NP1)	00086820
	SA(5)=12.56637*YB1(1)*TLN	

	IF (YNO.EQ.0.0.AND.PDA.EQ.0.0) SA(5)=SA(5)/2.0	00086830
C		00086840
	70 RETURN	00086850
	END	00086860
*DECK	COD	00086870
	SUBROUTINE COD (NI,XI,YI,NA,XO,ANS)	00086880
C		00086890
C****	A CONTROLLED DEVIATION ITERPOLATION METHOD	00086900
C		00086910
	DIMENSION XI(1) ,YI(1) ,XO(1) ,ANS(1)	00086920
C		00086930
	DATA XK/0.5/	00086940
C		00086950
	N=NI	00086960
	DO 910 IE=1,NA	00086970
	X=XO(IE)	00086980
100	IF(N-2)110,120,200	00086990
110	Y = YI(N)	00087000
	GO TO 900	00087010
120	Y = (YI(2)-YI(1))/(XI(2)-XI(1))* (X-XI(1)) +YI(1)	00087020
	GO TO 900	00087030
200	J = 1	00087040
210	IF(XI(J)-X)230,220,250	00087050
220	Y =YI(J)	00087060
	GO TO 900	00087070
230	J = J+1	00087080
	IF(J-N)210,210,250	00087090
250	IF(J-2)220,155,260	00087100
155	J = 3	00087110
	JJ = 1	00087120
	GO TO 285	00087130
260	IF(J-N)280,265,110	00087140
265	J = N-1	00087150
	JJ = 2	00087160
	GO TO 285	00087170
280	JJ = 3	00087180
285	IF(N-3)290,290,295	00087190
290	J = 3	00087200
295	K = J-1	00087210
	M = K-1	00087220
	L = J+1	00087230
	A1 = X-XI(M)	00087240
	A2 = X-XI(K)	00087250

	A3 = X-XI(J)	00087260
	AL = (X-XI(K))/(XI(J)-XI(K))	00087270
	S = AL*YI(J)+(1.0-AL)*YI(K)	00087280
	C1= A3*A2/((XI(M)-XI(K))*(XI(M)-XI(J)))	00087290
	C2= A1*A3/((XI(K)-XI(M))*(XI(K)-XI(J)))	00087300
	C3= A2*A1/((XI(J)-XI(M))*(XI(J)-XI(K)))	00087310
	P1 = C1*YI(M)+C2*YI(K)+C3*YI(J)	00087320
	IF(N-3)305,305,310	00087330
305	P2 = P1	00087340
	GO TO 315	00087350
310	A4 = X-XI(L)	00087360
	C4= A4*A3/((XI(K)-XI(J))*(XI(K)-XI(L)))	00087370
	C5= A2*A4/((XI(J)-XI(K))*(XI(J)-XI(L)))	00087380
	C6= A3*A2/((XI(L)-XI(K))*(XI(L)-XI(J)))	00087390
	P2 = C4*YI(K)+C5*YI(J)+C6*YI(L)	00087400
315	GO TO (320,330,350),JJ	00087410
320	P2 = P1	00087420
	AL = (X-XI(1))/(XI(2)-XI(1))	00087430
	S = AL*YI(2)+ (1.0-AL)*YI(1)	00087440
	P1= S + XK*(P2-S)	00087450
	GO TO 350	00087460
330	P1 = P2	00087470
	AL = (X-XI(N-1))/(XI(N)-XI(N-1))	00087480
	S = AL*YI(N) + (1.0-AL)*YI(N-1)	00087490
	P2 = S+ XK*(P1-S)	00087500
350	E1 = ABS(P1-S)	00087510
	E2 = ABS(P2-S)	00087520
	IF(E1+E2)400,400,410	00087530
400	Y = S	00087540
	GO TO 900	00087550
410	BT = (E1*AL)/(E1*AL+(1.0-AL)*E2)	00087560
	Y = BT*P2+(1.0-BT)*P1	00087570
900	ANS(IE)=Y	00087580
910	CONTINUE	00087590
	RETURN	00087600
	END	00087610
		00087620
		00087630
		00087640
		00087650
		00087660
		00087670
		00087680
	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,
!

00087690
00087700
00087710
00087720
•