NASA CONTRACTOR REPORT

NASA CR-2474

NASA CR-2474

COMPUTER PROGRAMS FOR CALCULATING THE STATIC LONGITUDINAL AERODYNAMIC CHARACTERISTICS OF WING-BODY-TAIL CONFIGURATIONS

by Michael R. Mendenhall, Frederick K. Goodwin, Marnix F. E. Dillenius, and David M. Kline

Prepared by
NIELSEN ENGINEERING & RESEARCH, INC.
Mountain View, Calif. 94043
for Ames Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • JANUARY 1975

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR 2474		
4. Title and Subtitle COMPUTER PROGRAMS FOR CALCULATING THE STATIC LONGITUDINAL AERODYNAMIC CHARACTERISTICS OF WING-BODY-TAIL CONFIGURATIONS		5. Report Date January 1975 6. Performing Organization Code
7. Author(s) Michael R. Mendenhall, Frederick K. Goodwin, Marnix F.E. Dillenius, and David M. Kline		8. Performing Organization Report No.
9. Performing Organization Name and Address Nielsen Engineering and Research, Inc. 510 Clyde Avenue Mountain View, CA 94043		10. Work Unit No. 11. Contract or Grant No. NAS 2-7347
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		13. Type of Report and Period Covered Contractor Report 14. Sponsoring Agency Code
15. Supplementary Notes		

16. Abstract

This document is a user's manual for four computer programs developed to calculate the longitudinal aerodynamic characteristics of wing-body and wing-body-tail combinations. The R1307 program is based on a linear method and is limited to the small range of angles of attack for which the lift and moment characteristics of wings and bodies are linear with angle of attack. The CRSFLW program is based on a crossflow method of predicting the forces and moments on bodies alone or wing-body combinations over a large angle of attack range. The method states that the normal-force distribution on a body is made up of a potential term given by slender-body theory and a viscous crossflow term modified by Newtonian theory. The SUBSON program predicts the longitudinal aerodynamic characteristics of wing-body-tail combinations at subsonic speeds and at angles of attack for which symmetrical pairs of vortices are shed from the body nose and the leading and side edges of the lifting surfaces. Program SUPSON predicts the longitudinal aerodynamic characteristics of wing-body-tail combinations at supersonic speeds in the same angle-of-attack range.

This program manual contains a description of the use of each program, instructions for preparation of input, a description of the output, program listings, and sample cases for each program.

17. Key Words (Suggested by Author(s))	18. Distribution Statement		
Computer programs to calculate longitudinal aerodynamic characteristics of wing-body-tail combinations	Unclassified - Unlimited CAT. 01		
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 219	22. Price* 5.75

*For sale by the National Technical Information Service, Springfield, Virginia 22151

TABLE OF CONTENTS

TABLE OF CONTENTS (CONCLUDED)

COMPUTER PROGRAMS FOR CALCULATING THE STATIC LONGITUDINAL AERODYNAMIC CHARACTERISTICS OF WING-BODY-TAIL CONFIGURATIONS

by Michael R. Mendenhall, Frederick K. Goodwin,
Marnix F. E. Dillenius, and David M. Kline
Nielsen Engineering & Research, Inc.

SUMMARY

This document is a user's manual for four computer programs developed to calculate the longitudinal aerodynamic characteristics of wing-body and wing-body-tail combinations. The R1307 program is based on a linear method and is limited to the small range of angles of attack for which the lift and moment characteristics of wings and bodies are linear with angle of attack. The CRSFLW program is based on a crossflow method of predicting the forces and moments on bodies alone or wing-body combinations over a large angle-of-attack range. The method states that the normal-force distribution on a body is made up of a potential term given by slender-body theory and a viscous crossflow term modified by Newtonian theory. The SUBSON program predicts the longitudinal aerodynamic characteristics of wing-body-tail combinations at subsonic speeds and at angles of attack for which symmetrical pairs of vortices are shed from the body nose and the leading and side edges of the lifting surfaces. Program SUPSON predicts the longitudinal aerodynamic characteristics of wing-body-tail combinations at supersonic speeds in the same angle-of-attack range.

This program manual contains a description of the use of each program, instructions for preparation of input, a description of the output, program listings, and sample cases for each program.

INTRODUCTION

An engineering prediction method for determining the longitudinal aerodynamic characteristics of wing-body-tail configurations including nonlinear aerodynamics of components and interference between them is presented in reference 1. Specifically, particular attention is paid to the nonlinearities associated with symmetrical vortex shedding from the nose of the fuselage and with leading-edge and side-edge separation vortices from the lifting surfaces. Four computer programs were developed to calculate the longitudinal aerodynamic characteristics of wing-body and wing-body-tail combinations. This document is a user's manual for these four computer programs. Principal reliance is made herein to reference 1 for a description of the methods and calculation procedures. Reference 1 also contains calculated results and comparisons with data for various types of configurations.

The first program, called R1307, is based on the method of reference 2. Use of this program is limited to the small range of angles of attack for which the lift and moment characteristics of wings and bodies are linear with angle of attack. In most cases, the upper angle limit of usefulness of R1307 is approximately 10°.

The second program, called CRSFLW, is a crossflow method of predicting the forces and moments on bodies-alone or wing-body combinations over a large angle-of-attack range. This program is based on the method described in references 3 and 4 which states that the normal-force distribution on a body is made up of a potential term given by slender-body theory and a viscous crossflow term modified by Newtonian theory. Nonlinear effects due to vortex shedding from the nose of the fuselage or the leading edge of the lifting surface are not included in this method.

The third program, called SUBSON, predicts the longitudinal aerodynamic characteristics of wing-body-tail combinations at subsonic speeds. It is based on the extension of the method of reference 2 to angles of attack for which symmetrical vortices are shed from the body nose and the leading and side edges of the lifting surfaces. The lifting surfaces are described by a vortex-lattice scheme.

The fourth program, called SUPSON, predicts the longitudinal aerodynamic characteristics of wing-body-tail combinations at supersonic speeds. It uses the same procedure as program SUBSON with the exception that a constant-pressure-panel lifting-surface theory describes the wing and tail surfaces.

Each of the above programs is presented in a separate self-contained section of this manual which contains a description of the use of the program, instructions for preparation of input, a description of the output, a complete program listing, and sample cases. A common list of references follows the fourth section.

PART I - R1307 COMPUTER PROGRAM

Introduction

This computer program automates the method presented in NACA Report 1307, reference 2, for determining the lift, pitching moment, and center of pressure of wing-body-tail combinations. The method is restricted to bodies of circular cross section with wings and tails which do not have swept-forward leading edges or swept-back trailing edges. It is further restricted to small angles of attack and small angles of wing and tail incidence in which the forces are linear with angle. For a complete description of the method the user of this program should consult reference 2.

The following sections of this write-up will present a description of the program, a description of the input, a description of the output, a program listing, and sample cases. The notation used is that of reference 2. The list of symbols from reference 2 is included herein for reference purposes.

The program is written in Fortran IV for the IBM 360 series machines. No tapes, drums, or disks other than the standard input/output units are required. The running time for a typical case on the IBM 360/67 is two to three seconds. To run the program on other machines such as the CDC 6600 minor changes are required since the inverse sine and cosine routines are used. On the IBM 360 these are called by ARSIN and ARCOS while on the CDC 6600 they are ASIN and ACOS. One or both of these routines are used in the following subprograms:

EQ19	EQ24
EQ24L	EQ30
EQ30L	VOLOG

List of Symbols

A_T tail-alone aspect ratio

A_W wing-alone aspect ratio

c_r chord at wing-body juncture or tail-body juncture

c_t	tip chord of wing or tail
C_L	lift coefficient based on wing-alone area except tail-alone lift coefficient based on tail-alone area
C_{L_α}	lift-curve slope for angle of attack, per radian
C_{L_δ}	lift-curve slope for wing or tail incidence, per radian
C_m	pitching-moment coefficient based on wing-alone area
C_{m_α}	pitching-moment-curve slope for angle of attack
C_{m_δ}	pitching-moment-curve slope for wing-incidence angle
d	body diameter
f	wing vortex semispan
k	ratio of lift component to lift of wing alone or tail alone for variable wing or tail incidence
K	ratio of lift component to lift of wing alone or tail alone for variable angle of attack
K_N	ratio of lift of body nose to lift of wing alone
ℓ	length of wing-body-tail combination
ℓ_W	distance from most forward point of body to intersection of wing leading edge and body
ℓ_M	distance from most forward point of body to center of moments
ℓ_R	reference length
ℓ_S	distance from most forward point of body to shoulder of body nose
ℓ_T	distance from most forward point of body to intersection of tail leading edge and body
$\bar{\ell}$	distance from most forward point of body to center of pressure position
L	lift force
m	cotangent of leading-edge sweep angle

M_∞	free-stream Mach number
r	body radius
r_N	body radius at shoulder of nose
r_W	body radius at wing
r_T	body radius at tail
s	maximum semispan of wing or tail in combination with body
S_N	cross-sectional area of nose at maximum section
S_R	reference area
S_T	tail-alone area
S_W	wing-alone area
V_s	volume of body nose up to shoulder
x,y,z	streamwise, spanwise, and vertical coordinates, respectively
$\bar{x}$	distance to center of pressure measured from intersection of wing leading edge and body for wing quantities and from intersection of tail leading edge and body for tail quantities
x_h	distance from intersection of wing leading edge and body to wing hinge line
α	angle of attack of body centerline
β	$\sqrt{\lvert M_\infty^2 - 1\rvert}$
βA	wing-alone or tail-alone effective aspect ratio
δ	wing-or tail-incidence angle, degrees
λ	taper ratio, $\left(\frac{c_t}{c_r}\right)$
Λ_{LE}	sweep angle of leading edge, degrees

Subscripts

N	body nose
T	tail

W	wing
B(T)	body in presence of tail
B(W)	body in presence of wing
T(B)	tail in presence of body
W(B)	wing in presence of body

Description of Program

The R1307 computer program consists of a main program, thirteen function subprograms, and four subroutine subprograms. The main program performs the calculations as shown on the calculating form presented in Table I of reference 2. All input and output takes place in the main program. The function and subroutine subprograms provide the quantities which are obtained from the charts and equations of reference 2 when following the procedure outlined in Table I of that reference. Unless otherwise noted, equations, figures, charts, and appendices referred to in this section are those in reference 2.

Function APENB calculates the tail interference factor using the equations of Appendix B. These values are plotted in Chart 7.

Function APENC calculates the tail interference factor using the equations of Appendix C. This function is used at supersonic speeds for rectangular tails when the wing vortex is inboard of the tip of the tail.

Function CHRT8 calculates the wing or tail lift-curve slope at supersonic speeds from the curves of Chart 8. Values of βC_{L_α} from the curves are tabulated in CHRT8 and linear interpolation in βA, λ, and zero sweep location is performed.

Function EQ14 calculates the value of $K_{W(B)}$ or $K_{T(B)}$ using equation (14). Values obtained from this equation are plotted in Chart 1.

Function EQ19 calculates the value of $k_{W(B)}$ or $k_{T(B)}$ using equation (19). Values obtained from this equation are also plotted in Chart 1.

Function EQ21 calculates the value of $K_{B(W)}$ or $K_{B(T)}$ using equation (21). Values obtained from this equation are plotted in Chart 1.

Function EQ24 calculates the value of $K_{B(W)}(\beta C_{L_\alpha})_W$ or $K_{B(T)}(\beta C_{L_\alpha})_T$ using equation (24). This function is used for the high-aspect-ratio range at supersonic speeds when there is an afterbody behind the wing or tail and βm_W or βm_T is greater than one. Values are plotted in Chart 4(a).

Function EQ24L calculates the same quantity as EQ24 for the case where there is no leading-edge sweep, $\beta m_W = \infty$ or $\beta m_T = \infty$. The limiting form of equation (24) for this case is

$$K_{B(W)} (\beta C_{L_\alpha})_W = \frac{8}{\pi(1+\lambda)\left(\frac{\beta d}{c_r}\right)\left(\frac{s}{r}-1\right)} \left\{ \left(1 + \frac{\beta d}{c_r}\right)^2 \cos^{-1}\left(\frac{\frac{\beta d}{c_r}}{1 + \frac{\beta d}{c_r}}\right) + \sqrt{1 + 2\frac{\beta d}{c_r}} - 1 - \left(\frac{\beta d}{c_r}\right)^2 \cosh^{-1}\left(1 + \frac{c_r}{\beta d}\right) - \frac{\pi}{2} \right\}$$

Function EQ26 also calculates the same quantity of EQ24 but for values of $\beta m_W \leq 1.0$ or $\beta m_T \leq 1.0$. Equation (26) is used for this calculation and values are plotted in Chart 4(a).

Functions EQ30, EQ30L, and EQ31 are analogous EQ24, EQ24L, and EQ26, respectively, for the case where there is no afterbody behind the wing or tail. Equations (30) and (31) are used for the calculations and values are plotted in Chart 4(b). The limiting form of equation (30) when $\beta m_W = \infty$ or $\beta m_T = \infty$ is

$$K_{B(W)} (\beta C_{L_\alpha})_W = \frac{8}{\pi(1+\lambda)\left(\frac{\beta d}{c_r}\right)\left(\frac{s}{r}-1\right)} \left\{ 2\cos^{-1}\left(\frac{\beta d}{c_r}\right) - \frac{\beta d}{c_r}\cosh^{-1}\left(\frac{c_r}{\beta d}\right) + \frac{\beta d}{c_r} - \left[1 - \left(\frac{\beta d}{c_r}\right)^2\right]\left[\frac{1}{2}\frac{\beta d}{c_r} + \frac{1\cdot 3}{2\cdot 4}\left(\frac{\beta d}{c_r}\right)^3 + \frac{1\cdot 3\cdot 5}{2\cdot 4\cdot 6}\left(\frac{\beta d}{c_r}\right)^5 + \ldots\right] \right\}$$

Function VOLOG calculates the volume of an ogive nose using the equations in the appendix of reference 3. In the notation of reference 2, the volume is

$$V_S = 8\pi r_N^3 \left[\frac{2}{3}\left(\frac{\ell_S}{2r_N}\right)^3 - \left(A_1 - \frac{1}{2}\right) A_2 \right]$$

where

$$A_1 = \left(\frac{\ell_S}{2r_N}\right)^2 + \frac{1}{4}$$

and

$$A_2 = \frac{\ell_S}{2r_N}\sqrt{A_1^2 - \left(\frac{\ell_S}{2r_N}\right)^2} + A_1^2 \sin^{-1}\left(\frac{\ell_S}{2r_N A_1}\right) - 2\left(\frac{\ell_S}{2r_N}\right)\left(A_1 - \frac{1}{2}\right)$$

Subroutine CH1011 calculates the wing-or tail-alone center of pressure at supersonic or subsonic speed. Curves for determining this quantity are given in Chart 10 and Chart 11 of reference 2. Values obtained from these curves are tabulated in the subroutine and linear interpolation in βA, λ, and zero sweep location is performed.

Subroutine CH1416 determines the center of pressure of the lift transferred to the body by the wing or tail. This quantity is presented in Charts 14, 15, and 16 of reference 2. For large aspect ratios at supersonic speeds, Chart 14 is used. For all other supersonic cases, Chart 15 is used. For all subsonic cases, Chart 16 is used. Values have been read from these charts and are tabulated in the subroutine. The subroutine selects the chart to be used on the basis of the Mach number and aspect-ratio parameter and then determines the value of $(\bar{x}/c_r)_{B(W)}$ or $(\bar{x}/c_r)_{B(T)}$ by linear interpolation in the tables.

Subroutine CH56 calculates the lateral position of the wing vortex $(f - r)_W/(s - r)_W$ at the wing location using Chart 5 for subsonic speeds and Chart 6 for supersonic speeds. Values have been read from these charts and are tabulated in the subroutine. Linear interpolation in βA, λ, and zero sweep location is performed.

Subroutine KFACT calls the appropriate function subprograms used to calculate $K_{W(B)}$, $K_{B(W)}$, and $k_{W(B)}$ or $K_{T(B)}$, $K_{B(T)}$, and $k_{T(B)}$. After these are determined, equation (33) of reference 2 is used to calculate $k_{B(W)}$ or $k_{B(T)}$.

Description of Input

Variable definitions.- The format of the input cards for the R1307 program is shown in figure 1. In this figure the program variable name is shown as well as the card columns in which the value is punched and the format in which it is punched. The remainder of this section consists of a table listing these program input variables along with the algebraic symbol used in reference 2, if applicable, and the input variable definition. The algebraic notation used in defining the configuration is shown in figure 2. A discussion of the preparation of the input is presented in the section following the table. All input length and area quantities are dimensional and should have consistent dimensions.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
Item 1		
NHEAD		Number of cards of information which identify run.
Item 2		
HEAD		Identifying information.
Item 3		
NTAIL		Is a tail present? NTAIL = 0; no NTAIL = 1; yes
NOSE		Ogive nose? NOSE = 0; no NOSE = 1; yes
NAFTBW		Afterbody behind wing trailing edge? NAFTBW = 0; no NAFTBW = 1; yes
NAFTBT		Afterbody behind tail trailing edge? NAFTBT = 0; no NAFTBT = 1; yes
Item 4		
FMACH	M_∞	Free-stream Mach number.
SLM	ℓ_M	Distance from most forward point of body to center of moments.
REFS	S_R	Reference area to be used in calculated lift and moment coefficients.
REFL	ℓ_R	Reference length to be used in calculated moment coefficients.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
Item 5		
RW	r_W	Average body radius at wing location.
SW	s_W	Maximum semispan of wing in combination with body.
CTW	c_t	Tip chord of wing.
CRW	c_r	Chord at wing-body juncture.
WLESWP	Λ_{LE}	Sweep angle of wing leading edge, degrees.
SLW	ℓ_W	Distance from most forward point of body to intersection of wing leading edge and body
XHW	x_h	Distance from intersection of wing leading edge and body to wing-hinge line
Item 6		
RT	r_T	Average body radius at tail location.
ST	s_T	Maximum semispan of tail in combination with body.
CTT	c_t	Tip chord of tail.
CRT	c_r	Chord at tail-body juncture.
TLESWP	Λ_{LE}	Sweep angle of tail leading edge, degrees.
SLT	ℓ_T	Distance from most forward point of body to intersection of tail leading edge and body.
Item 7		
RN	r_N	Body radius at shoulder of nose.
SLS	ℓ_S	Distance from most forward point of body to shoulder of body nose.
VS	V_S	Volume of body nose up to shoulder.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
Item 8		
CLAW	$(C_{L_\alpha})_W$	Wing lift-curve slope based on exposed wing area, per degree.
CLAT	$(C_{L_\alpha})_T$	Tail lift-curve slope based on exposed tail area, per degree.
CLAN	$(C_{L_\alpha})_N$	Nose lift-curve slope based on body cross-sectional area at shoulder of body nose, per degree.
Item 9		
ALFI	α_i	Initial body angle of attack for which calculation is to be performed, degrees.
ALFF	α_f	Final body angle of attack for which calculation is to be performed, degrees.
DALF	$\Delta\alpha$	Angle-of-attack increment to be used between α_i and α_f, degrees.
Item 10		
DWI	δ_{W_i}	Initial wing incidence angle for which calculation is to be performed, degrees.
DWF	δ_{W_f}	Final wing incidence angle for which calculation is to be performed, degrees.
DDW	$\Delta\delta_W$	Wing incidence angle increment to be used between δ_{W_i} and δ_{W_f}, degrees.
Item 11		
DTI	δ_{T_i}	Initial tail incidence angle for which calculation is to be performed, degrees.
DTF	δ_{T_f}	Final tail incidence angle for which calculation is to be performed, degrees.
DDT	$\Delta\delta_T$	Tail incidence angle increment to be used between δ_{T_i} and δ_{T_f}, degrees.

Input preparation.- A discussion of the input variables will be presented in this section as an aid in the preparation of the data deck. Before beginning this discussion a few words need to be said as to what the computer program treats as the wing and what it treats as the tail. If a configuration has one set of lifting surfaces, this is the wing regardless of its axial location on the body, and data describing the set are input as wing data. If there are two sets of lifting surfaces, the set closest to the nose is the wing and the aft set the tail. For example, if the configuration has a set of canards near the nose and a wing further aft on the body, the canard data are input as wing data and the wing data input as tail data.

Item number 1 of the input data is an index NHEAD which indicates how many cards of information, item number 2, are to follow to identify the run. The value of NHEAD must be one or greater. Item number 2 is a set of NHEAD cards containing hollerith information which the user wishes to use to identify the run. This information can be punched anywhere in the cards and is reproduced in the output just as it is read in.

Item number 3 contains four indices. The first, NTAIL, specifies whether a tail is (NTAIL = 1) or is not (NTAIL = 0) present. The second index, NOSE, specifies whether the nose is (NOSE = 1) or is not (NOSE = 0) an ogive. The purpose of this index is to provide a computation within the program of the nose volume for an ogive. For non-ogive noses, the volume must be input in Item 7. The third index, NAFTBW, specifies whether the body extends (NAFTBW = 1) or does not extend (NAFTBW = 0) behind the wing trailing edge. The last index, NAFTBT, specifies the same thing with respect to the tail. If NTAIL = 0 then NAFTBT should be input as zero. These last two indices are only used at supersonic Mach numbers. Thus, their values are immaterial at subsonic speeds.

Item numbers 4 through 7 are self explanatory if the table in the preceeding section and figure 2 are referred to. Item number 6, which contains the tail data, is omitted from the input deck if there is no tail, NTAIL = 0, in item number 3. If the nose is an ogive, NOSE = 1 in item number 3, then the nose volume, VS of item number 7, is input as zero and the program calculates the volume.

Item number 8 contains the lift-curve slopes, per degree, of the wing alone, CLAW, tail alone, CLAT, and nose, CLAN. The first two are determined by joining the exposed panels together. If experimental values of these quantities are known, they should be used. At subsonic Mach numbers CLAW and CLAT must be input. They can be obtained from, for example, reference 5. At supersonic speeds they can be input as zero and the program will determine them using Chart 8 of reference 2. If there is no tail, NTAIL = 0, CLAT is input as zero. If the nose lift-curve slope is not known, it is input as zero and the program uses the slender-body value of 0.0349 per degree.

The last three items of input, item numbers 9, 10, and 11, specify the ranges of angle of attack, wing incidence angle, and tail incidence angle for which calculations are to be performed. The first number on each card is the initial value of the angle, the second number is the final value, and the third number is the increment to be used in going from the initial to the final value. Calculations are performed for all combinations of these angles. If there is no tail, NTAIL = 0, zeros should be input for all three numbers in item 11.

Sample cases.- Listings of the input data decks for two sample cases are presented in figure 3 and sketches of the two configurations are shown in figures 4 and 5. Sample case 1 is the example used in the computing form presented as Table I in reference 2. Sample case 2 is the configuration used in the tests of reference 6.

Sample case 1 is a wing-body-tail combination. The nose is not an ogive, NOSE = 0, so that the nose volume, VS, is input. The Mach number, FMACH, is 1.99. Thus, the lift-curve slopes, CLAW, CLAT, and CLAN, are input as zero and the program is allowed to calculate them.

Sample case 2 is a canard-body-wing combination. The nose is an ogive, NOSE = 1, so that the nose volume, VS, is input as zero and the program calculates this quantity. The Mach number for this case is subsonic, FMACH = 0.13, so that the wing (canard) and tail (wing) lift-curve slope values, CLAW and CLAT, are input. In this example the nose lift-curve slope value, CLAN, is also input.

Description of Output

The output produced by the R1307 computer program for sample case 1 is shown in figure 6. The first output produced by the program, figure 6(a), is a tabulation of most of the input data, Items 1, 2, and 4 through 7 of figure 1. The next output, figure 6(b), lists quantities which are calculated by the program for the wing, tail, and nose and the lift-curve slopes which were either read in or calculated. The wing quantities which are tabulated are on the following page.

OUTPUT NOTATION	ALGEBRAIC NOTATION
EXPOSED AREA	S_W
BETA*AR	βA_W
BETA*D/CR	$\beta d/c_r$
AR PARAM	$\beta A_W(1 + 1/m\beta)(1 + \lambda)$
TAPER RATIO	λ
R/S	$(r/s)_W$
SM*BETA	$m\beta$
CKWB	$K_{W(B)}$
CKBW	$K_{B(W)}$
SKBW	$k_{W(B)}$
SKBW	$k_{B(W)}$
(XBAR/CR)W(B)	$(\bar{x}/c_r)_{W(B)}$
(XBAR/CR)B(W)	$(\bar{x}/c_r)_{B(W)}$
(F - R)W/(S - R)W	$(f - r)_W/(s - r)_W$

Except for the last quantity in the above list, the same quantities are tabulated for the tail. The listed nose quantities are:

OUTPUT NOTATION	ALGEBRAIC NOTATION
BASE AREA	$S_N = \pi r_N^2$
CKN	K_N
NOSE CENTER OF PRESSURE	$\bar{\ell}_N$

The definitions of the three lift-curve slopes are:

$$(CLA)W = \frac{d}{d\alpha}(C_L)_W = \frac{d}{d\alpha}\left(\frac{L_W}{q_\infty S_W}\right)$$

$$(CLA)T = \frac{d}{d\alpha}(C_L)_T = \frac{d}{d\alpha}\left(\frac{L_T}{q_\infty S_T}\right)$$

$$(CLA)N = \frac{d}{d\alpha}(C_L)_N = \frac{d}{d\alpha}\left(\frac{L_N}{q_\infty S_N}\right)$$

The first output on figure 6(c) is a series of lift and moment curve slopes for the complete configuration without including wing-tail interference. These six quantities appear in boxes 88 through 93 of Table I of reference 2. All of these slopes are evaluated at $\alpha = \delta_W = \delta_T = 0$ (A = DW = DT = 0 in the output notation) and the coefficients are formed using the input reference area S_R and reference length ℓ_R. For example

$$D(CL)/D(DW) = \frac{d}{d\delta_W}(C_{L_C}) = \frac{d}{d\delta_W}\left(\frac{L_C}{q_\infty S_R}\right)$$

and

$$D(CM)/D(DW) = \frac{d}{d\delta_W}(C_{m_C}) = \frac{d}{d\delta_W}\left(\frac{M_C}{q_\infty S_R \ell_R}\right)$$

The next quantities tabulated are lift and moment curve slopes for configuration components and the complete configuration including interference of the wing vortices on the tail. These are also evaluated at $\alpha = \delta_W = \delta_T = 0$. The column identified "BODY" is the nose component and that identified "WING-BODY" is the wing-body combination including the nose and wing-body interference. The column identified "TAIL-BODY MINUS NOSE" includes only the tail lift and that produced by tail-body interference. The last column pertains to the complete configuration including wing-tail interference. The quantities tabulated in this block of output appear in boxes 124 through 127 of Table I of reference 2.

The last output listed in figure 6(c) gives the lift and pitching-moment coefficients and the center of pressure of the complete configuration, both with and without wing-tail interference, as a function of angle of attack, A, wing incidence angle, DW, and tail incidence angle, DT. The quantity denoted as LM - L/LR is the center of pressure measured from the center of moments and made dimensionless by the reference length. It is calculated as CMC/CLC. The ranges of these three angles for which calculations were to be performed were read in as Items 9, 10, and 11 of the input data, see figure 1. The nine quantities in this table appear in boxes 94, 95, 96, 100, 104, 105, 120, 122, and 123 of Table I of reference 2.

The last page of output, figure 6(d), tabulates for the same angle ranges the lift and pitching-moment coefficient components including those due to wing interference on the tail. The columns headed "NOSE" are nose-alone quantities due to angle of attack. The columns headed "WING A+DW" are wing quantities, including wing-body interference, due to angle of attack and wing incidence. The columns headed "TAIL A+DT" are tail quantities, including tail-body interference, due to angle of attack and tail incidence. The remaining two columns headed "TAIL WING INT" are the coefficients produced by the tail due to wing vortex interference. The quantities tabulated on this page are not included in Table I of reference 2.

Figure 7 contains the output for sample case 2. The format is identical to that which has just been described. Since the configuration for sample case 2 is a canard-body-wing combination, "WING" in the output refers to the canard and "TAIL" refers to the wing.

Program Listing

The R1307 computer program consists of the main program, thirteen function subprograms, and four subroutine subprograms. Each source deck is identified in columns 73 through 80 by a four-character identification and a three-digit number sequencing the cards within that deck. The program listing is given on the following pages. The table below will act as a table of contents for the listing.

```
C     LIFT AND CENTER OF PRESSURE OF WING-BODY-TAIL COMBINATIONS
C          METHOD OF NACA REPORT 1307, 1957
C
C     DIMENSION STATEMENT
C
      DIMENSION HEAD(20)
C
C     FORMAT STATEMENTS
C
  701 FORMAT(10I5)
  702 FORMAT(20A4)
  703 FORMAT (1X,20A4)
  704 FORMAT(1H1,11X,58HLIFT AND CENTER OF PRESSURE OF WING-BODY-TAIL CO
     1MBINATIONS/24X,32HMETHOD OF NACA REPORT 1307, 1957///)
  705 FORMAT(4F10.5)
  706 FORMAT(// 2X,17HFLIGHT CONDITIONS/5X,13HMACH NUMBER =,F7.3
     1                              //2X,20HREFERENCE QUANTITIES/5X,6HAREA
     2=,0PF11.5/5X,8HLENGTH =,F9.5//2X,43HMOMENT CENTER RELATIVE TO TIP
     3OF NOSE IS AT,F9.5)
  707 FORMAT(8F10.5)
  708 FORMAT(// 2X,15HWING INPUT DATA)
  709 FORMAT(5X,23HBODY RADIUS IN REGION =,F10.5/5X,10HSEMISPAN =,F10.5/
     15X,11HTIP CHORD =,F10.5/5X,20HEXPOSED ROOT CHORD =,F10.5/5X,
     226HLEADING EDGE SWEEP ANGLE =,F10.5/5X,34HDISTANCE FROM TIP OF NOS
     3E TO BODY,,1X ,27HLEADING EDGE INTERSECTION =,F10.5)
  710 FORMAT(5X,33HHINGE LINE LOCATION MEASURED FROM,1X ,33HBODY, LEADIN
     1G EDGE INTERSECTION =,F10.5)
  711 FORMAT(// 2X,15HTAIL INPUT DATA)
  712 FORMAT(// 2X,15HNOSE INPUT DATA/5X,20HRADIUS AT SHOULDER =,F10.5/
     15X,39HDISTANCE FROM TIP OF NOSE TO SHOULDER =,F10.5/5X,28HVOLUME O
     2F NOSE TO SHOULDER =,F10.5)
  713 FORMAT(//2X,29HLIFT-CURVE SLOPES, PER DEGREE/8X,6H(CLA)W,6X,
     16H(CLA)T,6X,6H(CLA)N/3X,3F12.5)
  714 FORMAT(  2X,26HCALCULATED WING QUANTITIES)
  715 FORMAT(5X,14HEXPOSED AREA =,F9.4/5X,9HBETA*AR =,F9.5/5X,11HBETA*D/
     1CR =,F9.5/5X,10HAR PARAM =,F9.5/5X,13HTAPER RATIO =,F9.5/5X,5HR/S
     2=,F9.5/5X,9HSM*BETA =,F9.5)
  716 FORMAT(5X,6HCKWB =,F9.5/5X,6HCKBW =,F9.5/5X,6HSKWB =,F9.5/5X,6HSKB
     1W =,F9.5)
  717 FORMAT(5X,15H(XBAR/CR)W(B) =,F9.5)
  718 FORMAT(5X,15H(XBAR/CR)B(W) =,F9.5)
  719 FORMAT(5X,15H(F-R)W/(S-R)W =,F9.5)
  720 FORMAT(//2X,26HCALCULATED TAIL QUANTITIES)
  721 FORMAT(5X,6HCKTB =,F9.5/5X,6HCKBT =,F9.5/5X,6HSKTB =,F9.5/5X,6HSKB
     1T =,F9.5)
  722 FORMAT(5X,15H(XBAR/CR)T(B) =,F9.5)
  723 FORMAT(5X,15H(XBAR/CR)B(T) =,F9.5)
  724 FORMAT(//2X,26HCALCULATED NOSE QUANTITIES)
  725 FORMAT(5X,11HBASE AREA =,F9.5/5X,5HCKN =,F9.5/5X,25HNOSE CENTER OF
     1 PRESSURE =,F9.5)
  726 FORMAT(   13X,54HLIFT AND MOMENT CURVE SLOPES OF COMPLETE CONFIGUR
     1ATION/27X,25HNO WING-TAIL INTERFERENCE/35X,10HPER DEGREE/2X,10HD(C
     2L)/D(A),3X,11HD(CL)/D(DW),2X,11HD(CL)/D(DT),2X,10HD(CM)/D(A),3X,11
     3HD(CM)/D(DW),2X,11HD(CM)/D(DT)/F11.5,5F13.5)
  727 FORMAT(// 12X,56HLIFT AND MOMENT CURVE SLOPES OF CONFIGURATION COM
     1PONENTS/26X,27HWITH WING-TAIL INTERFERENCE/35X,10HPER DEGREE/50X,
     29HTAIL-BODY/28X,4HBODY,6X,9HWING-BODY,2X,10HMINUS NOSE,3X,14HWING-
     3BODY-TAIL)
  728 FORMAT(2X,19HD(CL)/D(A),  A=DW=0,1X,3F12.5,2X,F12.5)
  729 FORMAT(2X,19HD(CM)/D(A),  A=DW=0,1X,3F12.5,2X,F12.5)
  730 FORMAT(2X,19HD(CL)/D(DW), A=DW=0,1X,3F12.5,2X,F12.5)
  731 FORMAT(2X,19HD(CM)/D(DW), A=DW=0,1X,3F12.5,2X,F12.5)
  732 FORMAT(//  15X,51HLIFT AND MOMENT COEFFICIENTS AND CENTER OF PRESS
     1URE/25X,30HFOR THE COMPLETE CONFIGURATION//24X,25HNO WING-TAIL INT
     2ERFERENCE,3X,27HWITH WING-TAIL INTERFERENCE/4X,1HA,6X,2HDW,5X,2HDT
     3,6X,3HCLC,6X,3HCMC,4X,7HLM=L/LR,6X,3HCLC,6X,3HCMC,4X,7HLM=L/LR)
  733 FORMAT(3F7.2,1X,3F9.4,2X,3F9.4)
  734 FORMAT(1H )
  735 FORMAT(1H1)
  736 FORMAT(15X,66HLIFT AND MOMENT COEFFICIENT COMPONENTS WITH WING-TAIL
     1L INTERFERENCE//31X,17HLIFT COEFFICIENTS,20X,19HMOMENT COEFFICIENT
     2S/25X,4HNOSE,5X,4HWING,5X,4HTAIL,5X,4HTAIL,7X,4HNOSE,5X,4HWING,5X,
     34HTAIL,5X,4HTAIL/4X,1HA,6X,2HDW,5X,2HDT,7X,1HA,6X,4HA+DW,5X,4HA+DT
     4,3X,8HWING INT,7X,1HA,6X,4HA+DW,5X,4HA+DT,3X,8HWING INT)
  737 FORMAT(3F7.2,1X,4F9.4,2X,4F9.4)
  750 FORMAT(///2X,25HMACH NUMBER CANNOT BE 1.0)
  751 FORMAT(///2X,26HSW MUST BE GREATER THAN RW)
  752 FORMAT(///2X,30HCTW CANNOT BE GREATER THAN CRW)
  753 FORMAT(///2X,41HWING LEADING EDGE CANNOT BE SWEPT FORWARD)
  754 FORMAT(///2X,39HWING TRAILING EDGE CANNOT BE SWEPT BACK)
  755 FORMAT(///2X,26HST MUST BE GREATER THAN RT)
  756 FORMAT(///2X,30HCTT CANNOT BE GREATER THAN CRT)
  757 FORMAT(///2X,41HTAIL LEADING EDGE CANNOT BE SWEPT FORWARD)
  758 FORMAT(///2X,39HTAIL TRAILING EDGE CANNOT BE SWEPT BACK)
  759 FORMAT(///2X,44HIF NOSE IS NOT AN OGIVE VOLUME MUST BE INPUT)
C
C     READ AND WRITE HEADING INFORMATION
C
      RAD=57.29578
 1000 READ (5,701) NHEAD
      WRITE (6,704)
      DO 1 N=1,NHEAD
      READ (5,702) HEAD
    1 WRITE (6,703) HEAD
C
C     READ CONTROL INDICES
C
      READ (5,701) NTAIL,NOSE,NAFTBW,NAFTWT
C
C     READ AND WRITE FLIGHT CONDITIONS AND REFERENCE CONDITIONS
C
      READ (5,705) FMACH,     SLM,REFS,REFL
      WRITE (6,706) FMACH,     REFS,REFL,SLM
C
C     READ AND WRITE WING INPUT DATA
C
      READ (5,707) RW,SW,CTW,CRW,WLESWP,SLW,XWH
      WRITE (6,708)
      WRITE (6,709) RW,SW,CTW,CRW,WLESWP,SLW
      WRITE (6,710) XWH
C
C     READ AND WRITE TAIL INPUT DATA IF TAIL PRESENT
C
      IF (NTAIL.EQ.0) GO TO 10
      READ (5,707) RT,ST,CTT,CRT,TLESWP,SLT
      WRITE (6,711)
      WRITE (6,709) RT,ST,CTT,CRT,TLESWP,SLT
C
C     INPUT NOSE DATA
C
   10 READ (5,707) RN,SLS,VS
      WRITE (6,712) RN,SLS,VS
      WRITE (6,735)
C
C     CHECK INPUT DATA FOR CASES NOT HANDLED BY PROGRAM
C
      NERR=0
      IF (FMACH.NE.1.0) GO TO 200
      NERR=1
      WRITE (6,750)
  200 IF ((SW-RW).GT.0.0) GO TO 201
      NERR=1
      WRITE (6,751)
  201 IF ((CTW/CRW).LE.1.0) GO TO 202
      NERR=1
      WRITE (6,752)
  202 IF (WLESWP.GE.0.0) GO TO 203
      NERR=1
      WRITE (6,753)
  203 DUM=(SW-RW)*TAN(WLESWP/RAD)
      WTESWP=RAD*ATAN((DUM+CTW-CRW)/(SW-RW))
      IF (WTESWP.LT.2.0) GO TO 204
      NERR=1
      WRITE (6,754)
```

```
204 IF (NTAIL.EQ.0) GO TO 206
    IF ((ST-RT).GT.0.0) GO TO 205
    NERR=1
    WRITE (6,755)
205 IF ((CTT/CRT).LE.1.0) GO TO 206
    NERR=1
    WRITE (6,756)
206 IF (TLESWP.GE.0.0) GO TO 207
    NERR=1
    WRITE (6,757)
207 DUM=(ST-RT)*TAN(TLESWP/RAD)
    TTESWP=RAD*ATAN((DUM+CTT-CRT)/(ST-RT))
    IF (TTESWP.LT.2.0) GO TO 208
    NERR=1
    WRITE (6,758)
208 IF (NOSE.EQ.1) GO TO 209
    IF (VS.GT.0.0) GO TO 209
    NERR=1
    WRITE (6,759)
209 IF (NERR.EQ.0) GO TO 212
    KM=6
210 DO 211 J=1,KM
211 READ (5,707) A
    GO TO 1000
212 CONTINUE
C
C     INPUT WING, TAIL, AND NOSE ALONE LIFT CURVE SLOPES
C           INPUT AS 0.0 IF PROGRAM IS TO CALCULATE THEM
C
    READ (5,707) CLAW,CLAT,CLAN
C
C     CALCULATE WING QUANTITIES AS REQUIRED BY BOXES 6-21 AND 47-62 OF
C           TABLE I OF REPORT 1307
C
    BETA=SQRT(ABS(FMACH**2-1.0))
    TAPERW=CTW/CRW
    ROSW=RW/SW
    IF (WLESWP.LE.0.0) GO TO 20
    SMBW=BETA/TAN(WLESWP/57.29578)
    GO TO 21
 20 SMBW=1.0E+08
 21 CSW=(SW-RW)*(CTW+CRW)
    BETAAW=4.0*BETA*((SW-RW)**2)/CSW
    IF (CLAW.GT.0.0) GO TO 28
    IF (FMACH.GT.1.0) GO TO 25
 23 WRITE (6,799)
799 FORMAT(///2X,70HFOR SUBSONIC MACH NUMBERS WING AND TAIL LIFT-CURVE
   1 SLOPE MUST BE INPUT)
    KM=3
    GO TO 210
 25 CLAW=CHRT8(BETAAW,SW,RW,CTW,CRW,WLESWP)/(BETA*RAD)
 28 CONTINUE
    BDOCRW=BETA*2.0*RW/CRW
    PARAMW=BETAAW*(1.0+1.0/SMBW)*(1.0+TAPERW)
    CRMXHW=(CRW-XHW)/57.3
    WRITE (6,714)
    WRITE (6,715) CSW,BETAAW,BDOCRW,PARAMW,TAPERW,ROSW,SMBW
    CLAR=CLAW*RAD
    CALL KFACT (ROSW,CKWB,FMACH,PARAMW,CKBW,NAFTBW,SMBW,TAPERW,BDOCRW,
   1CLAR,BETA,WLESWP,BETAAW,SKWB,SKBW)
    WRITE (6,716) CKWB,CKBW,SKWB,SKBW
    CALL CH1011(SW,RW,WLESWP,CTW,CRW,FMACH,TAPERW,BETAAW,XBCRW)
    WRITE (6,717) XBCRW
    BLWBA=SLW+XBCRW*CRW
    SLMMLW=SLM-BLWBA
    CALL CH1416(SW,RW,WLESWP,CTW,CRW,FMACH,PARAMW,NAFTBW,BDOCRW,XBCRBW
   1,SMBW,TAPERW,ROSW,BETAAW)
    WRITE (6,718) XBCRBW
    BLBW=SLW+XBCRBW*CRW
    SLMLBW=SLM-BLBW
    CALL CH56(SW,RW,WLESWP,CTW,CRW,FMACH,TAPERW,BETAAW,FRSRW)
    WRITE (6,719) FRSRW
    SMRW=SW-RW
    CLAWSR=CSW*CLAW/REFS
C
C     CALCULATE TAIL QUANTITIES AS REQUIRED BY BOXES 23-36 AND 63-78 OF
C           TABLE I OF REPORT 1307 IF TAIL IS PRESENT
C
    IF (NTAIL.EQ.0) GO TO 50
    TAPERT=CTT/CRT
    ROST=RT/ST
    IF (TLESWP.LE.0.0) GO TO 30
    SMBT=BETA/TAN(TLESWP/57.29578)
    GO TO 31
 30 SMBT=1.0E+08
 31 CST=(ST-RT)*(CTT+CRT)
    BETAAT=4.0*BETA*((ST-RT)**2)/CST
    IF (CLAT.GT.0.0) GO TO 35
    IF (FMACH.LT.1.0) GO TO 23
    CLAT=CHRT8(BETAAT,ST,RT,CTT,CRT,TLESWP)/(BETA*RAD)
 35 CONTINUE
    BDOCRT=2.0*BETA*RT/CRT
    PARAMT=BETAAT*(1.0+1.0/SMBT)*(1.0+TAPERT)
    WRITE (6,720)
    WRITE (6,715) CST,BETAAT,BDOCRT,PARAMT,TAPERT,ROST,SMBT
    CLAR=CLAT*RAD
    CALL KFACT (ROST,CKTB,FMACH,PARAMT,CKBT,NAFTBT,SMBT,TAPERT,BDOCRT,
   1CLAR,BETA,TLESWP,BETAAT,SKTB,SKBT)
    WRITE (6,721) CKTB,CKBT,SKTB,SKBT
    CALL CH1011(ST,RT,TLESWP,CTT,CRT,FMACH,TAPERT,BETAAT,XBCRT)
    WRITE (6,722) XBCRT
    BLTBA=SLT+XBCRT*CRT
    SLMMLT=SLM-BLTBA
    CALL CH1416(ST,RT,TLESWP,CTT,CRT,FMACH,PARAMT,NAFTBT,BDOCRT,XBCRBT
   1,SMBT,TAPERT,ROST,BETAAT)
    WRITE (6,723) XBCRBT
    BLBT=SLT+XBCRBT*CRT
    SLMLBT=SLM-BLBT
    SMRT=ST-RT
    CLATSR=CST*CLAT/REFS
C
C     CALCULATE NOSE QUANTITIES AS REQUIRED BY BOXES 39-42 AND 79-85 OF
C           TABLE I OF REPORT 1307
C
 50 CSN=3.14159*RN*RN
    CSNSR=CSN/REFS
    IF (CLAN.LE.0.0) CLAN=2.0/RAD
    CKN=CLAN*CSNSR/CLAWSR
    IF (NOSE.GT.0) VS=VOLOG(RN,BLS)
    VSOSN=VS/CSN
    BLN=BLS-VSOSN
    BLMMLN=SLM-BLN
    WRITE (6,724)
    WRITE (6,725)  CSN,CKN,BLN
C
C     OUTPUT LIFT-CURVE SLOPES
C
    WRITE (6,713) CLAW,CLAT,CLAN
    WRITE (6,735)
C
C     CALCULATE FORCE AND MOMENT DERIVATIVES FOR NO WING-TAIL
C           INTERFERENCE, BOXES 88-93 OF TABLE I OF REPORT 1307
C
    CLAC=(CKN+CKWB+CKBW)*CLAWSR
    IF (NTAIL.GT.0) CLAC=CLAC+(CKTB+CKBT)*CLATSR
    CLDWC=(SKWB+SKBW)*CLAWSR
    CLDTC=0.0
    IF (NTAIL.GT.0) CLDTC=(SKTB+SKBT)*CLATSR
    CMAC=(CKN*BLMMLN+CKWB*SLMMLW+CKBW*SLMLBW)*CLAWSR/REFL
    IF (NTAIL.GT.0) CMAC=CMAC+(CKTB*SLMMLT+CKBT*SLMLBT)*CLATSR/REFL
    CMDWC=(SKWB*SLMMLW+SKBW*SLMLBW)*CLAWSR/REFL
    CMDTC=0.0
    IF (NTAIL.GT.0) CMDTC=(SKTB*SLMMLT+SKBT*SLMLBT)*CLATSR/REFL
    WRITE (6,726) CLAC,CLDWC,CLDTC,CMAC,CMDWC,CMDTC
```

```
C
C     CALCULATE WING-TAIL INTERFERENCE QUANTITIES, BOXES 106-109
C          OF TABLE I OF REPORT 1307
C
      IF (NTAIL.EQ.0) GO TO 90
      B106=RAD*CLAWSR*CLATSR*BETA*SWHT*REFS/(6.28319*BETAAT*FKSRW*SMRW*
     1CBT)
      B107=(BLTBA-SLW-CRW)/RAD
      FOST=(FRSRW*SMRW*RT)/ST
      IF (TAPERT.GT.0.99.AND.FOST.LT.1.0.AND.FMACH.GT.1.0) GO TO 80
      SIHZ=APENB(TAPERT,HOST,FOST,0,0)
      GO TO 90
   80 SIHZ=APENC(HOST,CRT,ST,FOST,0,0)
C
C     CALCULATE LIFT AND MOMENT CURVE SLOPES OF CONFIGURATION COMPONENTS
C          WITH WING-TAIL INTERFERENCE, BOXES 124-127 OF TABLE I OF
C          REPORT 1307
C
   90 WRITE (6,727)
      CLA1=CKN*CLAWSR
      CLA2=(CKN+CKWB+CKBW)*CLAWSR
      CLA3=0.0
      CLA4=CLAC
      IF (NTAIL.EQ.0) GO TO 100
      CLA3=(CKTB+CKBT)*CLATSR
      CLA4=CLA4+B106*CKWB*SIHZ
  100 WRITE (6,728) CLA1,CLA2,CLA3,CLA4
      CMA1=CLA1*BLMMLN/REFL
      CMA2=(CKN*BLMMLN+CKWB*SLMMLW+CKBW*SLMLBW)*CLAWSR/REFL
      CMA3=0.0
      CMA4=CMAC
      IF (NTAIL.EQ.0) GO TO 110
      CMA3=(CKTB*SLMMLT+CKBT*SLMLBT)*CLATSR/REFL
      CMA4=CMA4+B106*CKWB*SIHZ*SLMMLT/REFL
  110 WRITE (6,729) CMA1,CMA2,CMA3,CMA4
      CLD1=0.0
      CLD2=CLDWC
      CLD3=0.0
      CLD4=CLDWC
      IF (NTAIL.GT.0) CLD4=CLD4+B106*SKWB*SIHZ
      WRITE (6,730) CLD1,CLD2,CLD3,CLD4
      CMD1=0.0
      CMD2=CMDWC
      CMD3=0.0
      CMD4=CMDWC
      IF (NTAIL.GT.0) CMD4=CMD4+B106*SKWB*SIHZ*SLMMLT/REFL
      WRITE (6,731) CMD1,CMD2,CMD3,CMD4
C
C     CALCULATE LIFT AND MOMENT COEFFICIENTS AND CENTER OF PRESSURE
C          FOR THE COMPLETE CONFIGURATION, BOXES 94-105 AND 112-123
C          OF TABLE I OF REPORT 1307
C
      READ (5,707) ALFI,ALFF,DALF
      READ (5,707) DWI,DWF,DDW
      READ (5,707) DTI,DTF,DDT
      IF (DALF.LE.0.0) DALF=1.0
      IF (DDW.LE.0.0) DDW=1.0
      IF (DDT.LE.0.0) DDT=1.0
      WRITE (6,732)
      DELTAT=DTI
  120 DELTAW=DWI
  130 ALPHA=ALFI
  140 B97=CLAC*ALPHA
      B98=CLDWC*DELTAW
      B99=CLDTC*DELTAT
      CLCNI=B97+B98+B99
      B101=CMAC*ALPHA
      B102=CMDWC*DELTAW
      B103=CMDTC*DELTAT
      CMCNI=B101+B102+B103
      XCPLNI=99.99999
      IF (CLCNI.NE.0.0) XCPLNI=CMCNI/CLCNI
      CLCWI=CLCNI
      CMCWI=CMCNI
      XCPLWI=XCPLNI
      IF (NTAIL.EQ.0) GO TO 170
      B112=B107*ALPHA
      B113=DELTAW*CRMXMW
      HOST=(B112-B113)/ST
      IF (TAPERT.GT.0.99.AND.FOST.LT.1.0.AND.FMACH.GT.1.0) GO TO 150
      SI=APENB(TAPERT,HOST,FOST,HOST)
      GO TO 160
  150 SI=APENC(ROST,CRT,ST,FOST,HOST)
  160 B116=CKWB*ALPHA
      B117=SKWB*DELTAW
      B118=B116+B117
      B119=B106*SI*B118
      CLCWI=CLCNI+B119
      B121=B119*SLMMLT/REFL
      CMCWI=CMCNI+B121
      XCPLWI=99.99999
      IF (CLCWI.NE.0.0) XCPLWI=CMCWI/CLCWI
  170 WRITE (6,733) ALPHA,DELTAW,DELTAT,CLCNI,CMCNI,XCPLNI,CLCWI,CMCWI,
     1XCPLWI
      ALPHA=ALPHA+DALF
      IF (ALPHA.LE.ALFF) GO TO 140
      WRITE (6,734)
      DELTAW=DELTAW+DDW
      IF (DELTAW.LE.DWF) GO TO 130
      DELTAT=DELTAT+DDT
      IF (DELTAT.LE.DTF) GO TO 120
C
C     CALCULATE LIFT AND MOMENT COEFFICIENT COMPONENTS INCLUDING TAIL
C          COMPONENT DUE TO WING INTERFERENCE
C
      CLTADT=0.0
      CLTINT=0.0
      CMTADT=0.0
      CMTINT=0.0
      WRITE (6,735)
      WRITE (6,736)
      DELTAT=DTI
  300 DELTAW=DWI
  310 ALPHA=ALFI
  315 CLNOSE=CKN*CLAWSR*ALPHA
      CLWADW=((CKWB+CKBW)*ALPHA+(SKWB+SKBW)*DELTAW)*CLAWSR
      IF (NTAIL.EQ.0) GO TO 340
      CLTADT=((CKTB+CKBT)*ALPHA+(SKTB+SKBT)*DELTAT)*CLATSR
      B112=B107*ALPHA
      B113=DELTAW*CRMXMW
      HOST=(B112-B113)/ST
      IF (TAPERT.GT.0.99.AND.FOST.LT.1.0.AND.FMACH.GT.1.0) GO TO 320
      SI=APENB(TAPERT,ROST,FOST,HOST)
      GO TO 330
  320 SI=APENC(ROST,CRT,ST,FOST,HOST)
  330 B116=CKWB*ALPHA
      B117=SKWB*DELTAW
      B118=B116+B117
      CLTINT=B106*SI*B118
  340 CMNOSE=CKN*BLMMLN*CLAWSR*ALPHA/REFL
      CMWADW=((CKWB*SLMMLW+CKBW*SLMLBW)*ALPHA+(SKWB*SLMMLW+SKBW*SLMLBW)*
     1DELTAW)*CLAWSR/REFL
      IF (NTAIL.EQ.0) GO TO 350
      CMTADT=((CKTB*SLMMLT+CKBT*SLMLBT)*ALPHA+(SKTB*SLMMLT+SKBT*SLMLBT)*
     1DELTAT)*CLATSR/REFL
      CMTINT=CLTINT*SLMMLT/REFL
  350 WRITE (6,737) ALPHA,DELTAW,DELTAT,CLNOSE,CLWADW,CLTADT,CLTINT,
     1CMNOSE,CMWADW,CMTADT,CMTINT
      ALPHA=ALPHA+DALF
      IF (ALPHA.LE.ALFF) GO TO 315
      WRITE (6,734)
      DELTAW=DELTAW+DDW
      IF (DELTAW.LE.DWF) GO TO 310
      DELTAT=DELTAT+DDT
      IF (DELTAT.LE.DTF) GO TO 300
      GO TO 1000
      END
```

```
      FUNCTION APENB (TAPER,ROS,FOS,HOS)
C
C     CALCULATION OF TAIL INTERFERENCE FACTOR BY THE METHOD OF APPENDIX
C         B OF REPORT 1307, VALUES ARE PLOTTED IN CHART 7 OF THAT
C         REPORT
C
      DIMENSION F(4),H(4),CL(4)
      R=ROS
      F(1)=FOS
      H(1)=HOS
      F(2)=-FOS
      H(2)=HOS
      TA=FOS**2+HOS**2
      F(3)=FOS*R*R/TA
      H(3)=HOS*R*R/TA
      F(4)=-F(3)
      H(4)=H(3)
      DO 10 J=1,4
      TA=(1,0-R*TAPER-F(J)*(1,0-TAPER))/(2,0*(1,0-R))
      TB=ALOG((H(J)**2+(F(J)-1,0)**2)/(H(J)**2+(F(J)-R)**2))
      TC=1,0-R
      IF (H(J),EQ,0,0) GO TO 20
      TC=TC+H(J)*(ATAN((F(J)-1,0)/H(J))-ATAN((F(J)-R)/H(J)))
   20 CL(J)=TA*TB-(1,0-TAPER)*TC/(1,0-R)
   10 CONTINUE
      APENB=2,0*(CL(1)-CL(2)-CL(3)+CL(4))/(1,0+TAPER)
      RETURN
      END
```

```
      FUNCTION APENC (ROB,C,B,FOS,HOB)
C
C     CALCULATION OF TAIL INTERFERENCE FACTOR BY THE METHOD OF APPENDIX
C         C OF REPORT 1307
C
      DIMENSION CHI(2)
      DATA RTWO/1,41421/
      BOC=B/C
      ROC=ROB*BOC
      FOC=FOB*BOC
      HOC=HOB*BOC
      DO 10 J=1,2
      DA=2,0*(FOC-BOC)+1,0
      DB=-2,0*(FOC+BOC)+1,0
      THCS=(2,0*HOC)**2
      AA=DA*DA-1,0-THCS
      AB=DB*DB-1,0-THCS
      GA=SQRT(AA**2+4,0*DA*DA*THCS)
      GB=SQRT(AB**2+4,0*DB*DB*THCS)
      THC=2,0*HOC
      ABDA=ABS(DA)
      ABDB=ABS(DB)
      TA=0,5*ALOG(((DB+1,0)**2+THCS)/((DA+1,0)**2+THCS))
      TB=DB-DA
      RGAPAA=SQRT(GA+AA)
      RGAMAA=SQRT(GA-AA)
      RGBPAB=SQRT(GB+AB)
      RGBMAB=SQRT(GB-AB)
      TCN=DA**2+THCS+GA+RTWO*ABDA*RGAPAA+RTWO*THC*RGAMAA
      TCD=DB**2+THCS+GB+RTWO*ABDB*RGBPAB+RTWO*THC*RGBMAB
      TC=0,5*ALOG(TCN/TCD)
      TD=RTWO*DA*THC*RGAMAA/GA
      TE=AA*DA*RGAPAA/(RTWO*ABDA*GA)
      TF=RTWO*DB*THC*RGBMAB/GB
      TG=AB*DB*RGBPAB/(RTWO*ABDB*GB)
      THN=(HOC**2+(FOC-SOC+1,0)**2)*(HOC**2+(FOC+HOC)**2)
      THD=(HOC**2+(FOC-HOC)**2)*(HOC**2+(FOC+SOC-1,0)**2)
      TH=0,5*ALOG(THN/THD)
      CHI(J)=2,0*(TA+TB+TC+TD+TE-TF-TG+TH)/SOC
      IF (J,EQ,2) GO TO 10
      DUM=(ROC**2)/(FOC**2+HOC**2)
      FOC=DUM*FOC
      HOC=DUM*HOC
   10 CONTINUE
      APENC=2,0*(1,0-ROS)*SOC*(CHI(1)-CHI(2))/(4,0*(1,0-ROS)-1,0/SOC)
      RETURN
      END
```

```
      FUNCTION CHRT8 (BAR,S,R,CT,CR,SWPLE)
C
C     CALCULATE BETA*D(CL)/D(ALPHA) FOR SUPERSONIC MACH NUMBERS, CHART 8
C         OF REPORT 1307
C
      DIMENSION BA(11),CL(3),CH8(11,4)
      DATA RAD/57,29578/
      DATA BA/0,0,0,5,1,0,1,5,2,0,2,5,3,0,4,0,5,0,6,0,10,0/
      DATA CH8/
     1 0,0, 0,75, 1,45, 2,07, 2,59, 3,02, 3,41, 4,00, 4,00, 4,00, 4,00,
     2 0,0, 0,86, 2,09, 2,73, 3,41, 3,55, 3,65, 3,76, 3,82, 3,86, 4,00,
     3 0,0, 0,80, 1,81, 2,65, 3,12, 3,36, 3,52, 3,70, 3,79, 3,82, 4,00,
     4 0,0, 0,86, 2,02, 2,67, 3,00, 3,20, 3,34, 3,51, 3,60, 3,65, 4,00/
      IF (BAR,LE,10,0) GO TO 10
      CHRT8=4,0
      RETURN
   10 I=0
   11 I=I+1
      IF (BAR,GE,BA(I)) GO TO 11
      IM=I-1
      IP=I
      DXBA=(BAR-BA(IM))/(BA(IP)-BA(IM))
      TAPER=CT/CR
      IF (TAPER,LT,0,5) GO TO 20
      DXSL=(TAPER-0,5)/0,5
      TA=CH8(IM,3)+DXBA*(CH8(IP,3)-CH8(IM,3))
      TB=CH8(IM,4)+DXBA*(CH8(IP,4)-CH8(IM,4))
      CHRT8=TA+DXSL*(TB-TA)
      RETURN
   20 CTLE=(S-R)*TAN(SWPLE/RAD)
      SWPMC=RAD*ATAN((CTLE+0,5*(CT-CR))/(S-R))
      SWPTE=RAD*ATAN((CTLE+CT-CR)/(S-R))
      DXSL=TAPER/0,5
      IF (SWPMC,LE,0,0) GO TO 30
      JM=2
      JP=1
      DXCL=SWPMC/(SWPMC-SWPTE)
      GO TO 40
   30 JM=1
      JP=2
      DXCL=SWPLE/(SWPLE-SWPMC)
   40 DO 50 K=1,3
   50 CL(K)=CH8(IM,K)+DXBA*(CH8(IP,K)-CH8(IM,K))
      TA=CL(JM)+DXCL*(CL(JP)-CL(JM))
      TB=CL(3)
      CHRT8=TA+DXSL*(TB-TA)
      RETURN
      END
```

```
      FUNCTION EQ14(X)                                                  LN05 001
C                                                                       LN05 002
C     CALCULATE KW(B) OR KT(B) FROM EQ. 14 OF REPORT 1307               LN05 003
C                                                                       LN05 004
      IF (X.LE.0.0) GO TO 10                                            LN05 005
      IF (X.GE.1.0) GO TO 20                                            LN05 006
      X2=X*X                                                            LN05 007
      TA=1+X2*X2                                                        LN05 008
      TB=0.5*ATAN(0.5*(1.0/X-X))+3.14159/4.0                            LN05 009
      TC=X2*(1.0/X-X+2.0*ATAN(X))                                       LN05 010
      EQ14=2.0*(TA*TB-TC)/(3.14159*(1.0-X)**2)                          LN05 011
      RETURN                                                            LN05 012
   10 EQ14=1.0                                                          LN05 013
      RETURN                                                            LN05 014
   20 EQ14=2.0                                                          LN05 015
      RETURN                                                            LN05 016
      END                                                               LN05 017
```

```
      FUNCTION EQ19 (ROS)                                               LN06 001
C                                                                       LN06 002
C     CALCULATE SKW(B) OR SKT(B) FROM EQ 19 OF REPORT 1307              LN06 003
C                                                                       LN06 004
      IF (ROS.LE.0.0.OR.ROS.GE.1.0) GO TO 10                            LN06 005
      PI=3.14159                                                        LN06 006
      PIS=PI*PI                                                         LN06 007
      X=1.0/ROS                                                         LN06 008
      TA=X*X                                                            LN06 009
      TB=X+1.0                                                          LN06 010
      TC=X-1.0                                                          LN06 011
      TD=X*X+1.0                                                        LN06 012
      TE=X*X-1.0                                                        LN06 013
      TF=TC*TC                                                          LN06 014
      TG=ARSIN(TE/TD)                                                   LN06 015
      A=0.25*PIS*TB*TB/TA                                               LN06 016
      B=PI*TD*TD*TG/(TA*TF)                                             LN06 017
      C=2.0*PI*TB/(X*TC)                                                LN06 018
      D=TD*TD*TG*TG/(TA*TF)                                             LN06 019
      E=4.0*TB*TG/(X*TC)                                                LN06 020
      F=8.0*ALOG(TD/(2.0*X))/TF                                         LN06 021
      EQ19=(A+B-C+D-E+F)/PIS                                            LN06 022
      RETURN                                                            LN06 023
   10 EQ19=1.0                                                          LN06 024
      RETURN                                                            LN06 025
      END                                                               LN06 026
```

```
      FUNCTION EQ21(X)                                                  LN07 001
C                                                                       LN07 002
C     CALCULATE KB(W) OR KB(T) FROM EQ 21 OF REPORT 1307                LN07 003
C                                                                       LN07 004
      IF (X.LE.0.0) GO TO 10                                            LN07 005
      IF (X.GE.1.0) GO TO 20                                            LN07 006
      EQ21=(1.0-X*X)**2/(1.0-X)**2-EQ14(X)                              LN07 007
      RETURN                                                            LN07 008
   10 EQ21=0.0                                                          LN07 009
      RETURN                                                            LN07 010
   20 EQ21=2.0                                                          LN07 011
      RETURN                                                            LN07 012
      END                                                               LN07 013
```

```
      FUNCTION EQ24(BM,TAPER,BDCR,ROS)                                  LN08 001
C                                                                       LN08 002
C     CALCULATE KB(W)*BCLAW OR KB(T)*BCLAT FROM EQ 24 FOR M GREATER THANLN08 003
C          1.0, VALUE OF THE ASPECT RATIO PARAMETER GREATER THAN 4, AND LN08 004
C          MB GREATER THAN 1                                            LN08 005
C                                                                       LN08 006
      IF (ROS.LE.0.0) GO TO 10                                          LN08 007
      IF (ROS.GE.1.0) GO TO 20                                          LN08 008
      TA=1.0+BM                                                         LN08 009
      TB=TA*BDCR                                                        LN08 010
      TC=SQRT(BM*BM-1.0)                                                LN08 011
      TD=BM/TA                                                          LN08 012
      TE=TD*(((TB+BM)/BM)**2)*ARCOS((1.0+TB)/(BM+TB))                   LN08 013
      TF=TC*(SQRT(1.0+2.0*BDCR)-1.0)/TA                                 LN08 014
      TG=1.0+1.0/BDCR                                                   LN08 015
      TH=TC*BDCR*BDCR*ALOG(TG+SQRT(TG*TG-1.0))/BM                       LN08 016
      TI=BM*ARCOS(1.0/BM)/TA                                            LN08 017
      EQ24=8.0*BM*(TE+TF-TH-TI)/(3.14159*(1.0+TAPER)*BDCR*(1.0/ROS-1.0))LN08 018
      EQ24=EQ24/TC                                                      LN08 019
      RETURN                                                            LN08 020
   10 EQ24=0.0                                                          LN08 021
      RETURN                                                            LN08 022
   20 EQ24=2.0                                                          LN08 023
      RETURN                                                            LN08 024
      END                                                               LN08 025
```

```
      FUNCTION EQ24L(TAPER,BDCR,ROS)                                    LN09 001
C                                                                       LN09 002
C     LIMITING FORM OF EQ 24 WHEN BM EQUAL INFINITY, NO L.E. SWEEP      LN09 003
C                                                                       LN09 004
      IF (ROS.LE.0.0) GO TO 10                                          LN09 005
      IF (ROS.GE.1.0) GO TO 20                                          LN09 006
      TA=1.0+BDCR                                                       LN09 007
      TB=1.0+1.0/BDCR                                                   LN09 008
      TC=TA*TA*ARCOS(BDCR/TA)+SQRT(1.0+2.0*BDCR)-1.0-BDCR*BDCR*ALOG(TB+ LN09 009
     1SQRT(TB*TB-1.0))-1.5708                                           LN09 010
      EQ24L=8.0*TC/(3.14159*(1.0+TAPER)*BDCR*(1.0/ROS-1.0))             LN09 011
      RETURN                                                            LN09 012
   10 EQ24L=0.0                                                         LN09 013
      RETURN                                                            LN09 014
   20 EQ24L=2.0                                                         LN09 015
      RETURN                                                            LN09 016
      END                                                               LN09 017
```

```
      FUNCTION EQ26(BM,TAPER,BDCR,ROS)                                  LN10 001
C                                                                       LN10 002
C     CALCULATE KB(W)*BCLAW OR KB(T)*BCLAT FROM EQ 26 FOR M GREATER THANLN10 003
C          1.0, VALUE OF ASPECT RATIO PARAMETER GREATER THAN 4, AND     LN10 004
C          MB LESS THAN OR EQUAL TO 1                                   LN10 005
C                                                                       LN10 006
      IF (ROS.LE.0.0) GO TO 10                                          LN10 007
      IF (ROS.GE.1.0) GO TO 20                                          LN10 008
      TA=BM/(1.0+BM)                                                    LN10 009
      TB=BDCR/TA                                                        LN10 010
      TC=1.0+TB                                                         LN10 011
      TD=SQRT(1.0/TC)                                                   LN10 012
      TE=SQRT(TC)*(TC+1.0)-2.0-TB*TB*(0.5*ALOG(1.0+TD)-0.5*ALOG(1.0-TD))LN10 013
      EQ26=16.0*TA*TA*TE/(3.14159*(1.0+TAPER)*BDCR*(1.0/ROS-1.0))       LN10 014
      RETURN                                                            LN10 015
   10 EQ26=0.0                                                          LN10 016
      RETURN                                                            LN10 017
   20 EQ26=2.0                                                          LN10 018
      RETURN                                                            LN10 019
      END                                                               LN10 020
```

```
      FUNCTION EQ30(BM,TAPER,BDCRA,ROS)
C
C     CALCULATE KB(W)*BCLAW OR KB(T)*BCLAT FROM EQ 30 FOR M GREATER THAN
C           1.0, VALUE OF THE ASPECT RATIO PARAMETER GREATER THAN 4, AND
C             MB GREATER THAN 1.  NO AFTERBODY.
C
      IF (ROS.LE.0.0) GO TO 10
      IF (ROS.GE.1.0) GO TO 20
      BDCR=BDCRA
      RRF=1.0
      IF (BDCR.LE.1.0) GO TO 1
      BDCR=1.0
      RRF=BDCRA
    1 TA=SQRT(BM*BM-1.0)
      TB=1.0+BM/BDCR
      TC=1.0/BDCR
      TD=TC*TC
      TE=TB*TB*ARCOS((BM+TC)/TB)-BM*BM*TD*ARCOS(1.0/BM)+BM*TD*TA*ARSIN(B
     1DCR)-TA*ALOG(TC+SQRT(TC*TC-1.0))
      EQ30=8.0*BDCR*TE/(3.14159*TA*(TAPER+1.0)*(1.0/ROS-1.0)*RRF)
      RETURN
   10 EQ30=0.0
      RETURN
   20 EQ30=2.0
      RETURN
      END
```

```
      FUNCTION EQ30L(TAPER,BDCRA,ROS)
C
C     LIMITING FORM OF EQUATION 30 WHEN BM EQUAL INFINITY, NO L.E. SWEEP
C
      IF (ROS.LE.0.0) GO TO 10
      IF (ROS.GE.1.0) GO TO 20
      BDCR=BDCRA
      RRF=1.0
      IF (BDCR.LT.1.0) GO TO 1
      TC=1.0
      RRF=BDCRA
    3 EQ30L=8.0*TC/(3.14159*(TAPER+1.0)*(1.0/ROS-1.0)*RRF)
      RETURN
    1 TA=1.0/BDCR
      TB=BDCR**2
      TC=2.0*ARCOS(BDCR)-BDCR*ALOG(TA+SQRT(TA*TA-1.0))+BDCR
      TD=1.0-TB
      TE=TD*BDCR/2.0
      SUM=TE
      FN=1.0
    2 FN=FN+2.0
      TE=TD*FN*TB*TE/(FN+1.0)
      SUM=SUM+TE
      IF (TE.GT.0.00001) GO TO 2
      TC=TC-SUM
      GO TO 3
   10 EQ30L=0.0
      RETURN
   20 EQ30L=2.0
      RETURN
      END
```

```
      FUNCTION EQ31(BM,TAPER,BDCRA,ROS)
C
C     CALCULATE KB(W)*BCLAW OR KB(T)*BCLAT FROM EQ 31 FOR M GREATER THAN
C           1.0, VALUE OF ASPECT RATIO PARAMETER GREATER THAN 4, AND
C             MB GREATER THAN 1.  NO AFTERBODY.
C
      IF (ROS.LE.0.0) GO TO 10
      IF (ROS.GE.1.0) GO TO 20
      BDCR=BDCRA
      RRF=1.0
      IF (BDCR.LE.1.0) GO TO 1
      BDCR=1.0
      RRF=BDCRA
    1 TA=SQRT(BM)
      TB=1.0+BM/BDCR
      TC=1.0/BDCR-1.0
      TD=1.0+BM
      TE=TB*SQRT(TC*TB)
      TF=((1.0/BDCR)**2)*BM*TA
      TG=BM*((1.0/BDCR)**2)*TD*(ATAN(SQRT(1.0/BM))-ATAN(SQRT(TC/TB)))
      TH=SQRT(BM*TC/TB)
      TI=TD*(0.5*ALOG(1.0+TH)-0.5*ALOG(1.0-TH))/TA
      EQ31=16.0*TA*BDCR*(TE-TF+TG-TI)/(3.14159*TD*(TAPER+1.0)*(1.0/ROS
     1-1.0)*RRF)
      RETURN
   10 EQ31=0.0
      RETURN
   20 EQ31=2.0
      RETURN
      END
```

```
      FUNCTION VOLOG(RB,SL)
C
C     CALCULATES VOLUME OF OGIVE NOSE USING EQUATIONS OF APPENDIX
C           OF TN D 6996 BY JORGENSEN
C
      D=2.0*RB
      SLDSQ=(SL/D)**2
      R=SLDSQ+0.25
      SLD=SL/D
      TC=SLD*SQRT(R*R-SLDSQ)+R*R*ARSIN(SLD/R)-2.0*SLD*(R-0.5)
      VDC=3.14159*(0.66667*SLDSQ*SLD-TC*(R-0.5))
      VOLOG=VDC*(D**3)
      RETURN
      END
```

```
      SUBROUTINE CH1011(S,R,SWPLE,CT,CR,FMACH,TAPER,BAR,XBCR)
C
C     SUBROUTINE TO CALCULATE WING OR TAIL CENTER OF PRESSURE, CHART 10
C          OR 11 OF REPORT 1307, IN TABLE I OF THAT REPORT THESE
C          QUANTITIES ARE IN BOXES 51 AND 52 OR 67 AND 68
C
      DIMENSION BA(11),C10(11,3,3),C11(11,3,3),V(3,3)
      DATA BA/0.0,0.5,1.0,1.5,2.0,2.5,3.0,4.0,6.0,8.0,1000.0/
      DATA C10/
     1     .000,.092,.174,.236,.274,.300,.318,.331,.335,.335,.335,
     2     .000,.131,.231,.301,.340,.363,.379,.390,.390,.390,.390,
     3     .000,.205,.338,.412,.448,.463,.473,.477,.484,.491,.491,
     4     .336,.386,.421,.443,.458,.468,.476,.484,.494,.498,.498,
     5     .172,.309,.386,.436,.465,.481,.486,.490,.496,.501,.501,
     6     .000,.205,.338,.412,.448,.463,.473,.477,.484,.491,.491,
     7     .667,.667,.667,.667,.667,.667,.667,.667,.667,.667,.667,
     8     .340,.455,.512,.545,.566,.576,.583,.595,.604,.609,.609,
     9     .000,.205,.338,.412,.448,.463,.473,.477,.484,.491,.491/
      DATA C11/
     1     .000,.095,.133,.146,.148,.149,.150,.153,.157,.162,.162,
     2     .000,.123,.178,.192,.195,.195,.196,.198,.200,.200,.200,
     3     .000,.164,.228,.245,.249,.250,.250,.250,.250,.250,.250,
     4     .333,.342,.350,.355,.356,.355,.354,.352,.351,.350,.350,
     5     .167,.248,.288,.306,.310,.310,.310,.310,.310,.310,.310,
     6     .000,.164,.228,.245,.249,.250,.250,.250,.250,.250,.250,
     7     .667,.600,.570,.560,.554,.551,.549,.543,.539,.542,.542,
     8     .333,.367,.392,.408,.410,.410,.410,.410,.410,.410,.410,
     9     .000,.164,.228,.245,.249,.250,.250,.250,.250,.250,.250/
      DATA RAD/57.29578/
      CTLE=(S-R)*TAN(SWPLE/RAD)
      SWPMC=RAD*ATAN((CTLE+0.5*(CT-CR))/(S-R))
      SWPTE=RAD*ATAN((CTLE+CT-CR)/(S-R))
      IF (SWPTE.LE.1.0) GO TO 10
      WRITE (6,701)
  701 FORMAT(//1X,49HPROGRAM WILL NOT HANDLE SWEPT-BACK TRAILING EDGES/
     11X,20HEXECUTION TERMINATED)
      STOP
   10 IF (SWPMC.LE.0.0) GO TO 20
      KM=2
      KP=3
      DXCL=SWPMC/(SWPMC-SWPTE)
      GO TO 30
   20 KM=1
      KP=2
      DXCL=1.0
      IF (TAPER.NE.1.0) DXCL=SWPLE/(SWPLE-SWPMC)
   30 IF (TAPER.LE.0.5) GO TO 40
      JM=2
      JP=3
      DXSL=(TAPER-0.5)/0.5
      GO TO 50
   40 JM=1
      JP=2
      DXSL=TAPER/0.5
   50 I=0
   51 I=I+1
      IF (BAR.GE.BA(I)) GO TO 51
      IM=I-1
      IP=I
      DXBA=(BAR-BA(IM))/(BA(IP)-BA(IM))
      IF (FMACH.LE.1.0) GO TO 65
      DO 60 K=KM,KP
      DO 60 J=JM,JP
   60 V(K,J)=C10(IM,J,K)+DXBA*(C10(IP,J,K)-C10(IM,J,K))
      GO TO 80
   65 DO 70 K=KM,KP
      DO 70 J=JM,JP
   70 V(K,J)=C11(IM,J,K)+DXBA*(C11(IP,J,K)-C11(IM,J,K))
   80 TA=V(KM,JM)+DXSL*(V(KM,JP)-V(KM,JM))
      TB=V(KP,JM)+DXSL*(V(KP,JP)-V(KP,JM))
      XBCR=TA+DXCL*(TB-TA)
      RETURN
      END
```

```
      SUBROUTINE CH1416(S,R,SWPLE,CT,CR,FMACH,ARPARM,NAFTER,BDCR,XBCR,
     1SWB,TAPER,RUS,BAR)
C
C     SUBROUTINE TO CALCULATE BODY CENTER OF PRESSURE DUE TO WING OR
C          TAIL, CHART 14, 15, OR 16 OF REPORT 1307.  IN TABLE I OF THAT
C          REPORT THIS QUANTITY IS IN BOX 57 OR 73.
C
      DIMENSION BD14B(6),XB14B(6,2),BA15(7),                TA15(7,4),
     1TB15(7,4),TC15(7,4),TD15(7,4),TE15(7,4),TF15(7,4),TG15(7,4),TH15(7
     2,4),TI15(7,4),BA16(8),                TA16(8,4),TB16(8,4),TC16(8,4)
     3,TD16(8,4),TE16(8,4),TF16(8,4),TG16(8,4),TH16(8,4),TI16(8,4)
      DIMENSION XB15A(7,4,3),XB15B(7,4,3),XB15C(7,4,3),
     1XB16A(8,4,3),XB16B(8,4,3),XB16C(8,4,3)
      DIMENSION VA(4,3,3),V(3,3)
      EQUIVALENCE (XB15A(1,1,1),TA15(1,1)),(XB15A(1,1,2),TB15(1,1)),
     1(XB15A(1,1,3),TC15(1,1)),(XB15B(1,1,1),TD15(1,1)),
     2(XB15B(1,1,2),TE15(1,1)),(XB15B(1,1,3),TF15(1,1)),
     3(XB15C(1,1,1),TG15(1,1)),(XB15C(1,1,2),TH15(1,1)),
     4(XB15C(1,1,3),TI15(1,1))
      EQUIVALENCE (XB16A(1,1,1),TA16(1,1)),(XB16A(1,1,2),TB16(1,1)),
     1(XB16A(1,1,3),TC16(1,1)),(XB16B(1,1,1),TD16(1,1)),
     2(XB16B(1,1,2),TE16(1,1)),(XB16B(1,1,3),TF16(1,1)),
     3(XB16C(1,1,1),TG16(1,1)),(XB16C(1,1,2),TH16(1,1)),
     4(XB16C(1,1,3),TI16(1,1))
      DATA RAD/57.29578/
      DATA BD14B/0.0, 0.2, 0.4, 0.6, 0.8, 1.0/
      DATA XB14B/0.500, 0.590, 0.637, 0.658, 0.665, 0.667,
     1           0.500, 0.575, 0.622, 0.646, 0.660, 0.667/
      DATA BA15/ 0.0, 0.5, 1.0, 1.5, 2.0, 3.0, 4.0/
      DATA BA16/ 0.0, 0.5, 1.0, 1.5, 2.0, 3.0, 4.0, 1000.0/
      DATA TA15/ 0.00, 0.13, 0.25, 0.34, 0.40, 0.48, 0.50,
     1           0.00, 0.15, 0.30, 0.41, 0.50, 0.63, 0.72,
     2           0.00, 0.25, 0.45, 0.60, 0.75, 0.95, 1.13,
     3           0.00, 0.37, 0.65, 0.90, 1.12, 1.50, 1.80/
      DATA TB15/ 0.00, 0.14, 0.27, 0.38, 0.46, 0.50, 0.50,
     1           0.00, 0.21, 0.40, 0.57, 0.67, 0.82, 0.90,
     2           0.00, 0.32, 0.58, 0.79, 0.95, 1.20, 1.42,
     3           0.00, 0.60, 0.97, 1.24, 1.48, 1.90, 2.31/
      DATA TC15/ 0.00, 0.09, 0.34, 0.46, 0.50, 0.50, 0.50,
     1           0.00, 0.26, 0.55, 0.69, 0.77, 0.89, 1.00,
     2           0.00, 0.48, 0.78, 0.98, 1.13, 1.42, 1.68,
     3           0.00, 0.77, 1.17, 1.50, 1.80, 2.39, 2.98/
      DATA TD15/ 0.26, 0.35, 0.43, 0.47, 0.50, 0.50, 0.50,
     1           0.26, 0.40, 0.51, 0.59, 0.65, 0.73, 0.80,
     2           0.26, 0.47, 0.63, 0.75, 0.85, 1.02, 1.15,
     3           0.26, 0.60, 0.84, 1.05, 1.21, 1.53, 1.85/
      DATA TE15/ 0.14, 0.30, 0.42, 0.47, 0.50, 0.50, 0.50,
     1           0.14, 0.38, 0.54, 0.64, 0.71, 0.81, 0.90,
     2           0.14, 0.48, 0.70, 0.87, 1.00, 1.23, 1.43,
     3           0.14, 0.67, 1.00, 1.27, 1.52, 1.96, 2.43/
      DATA TG15/ 0.50, 0.50, 0.50, 0.50, 0.50, 0.50, 0.50,
     1           0.50, 0.54, 0.58, 0.61, 0.65, 0.71, 0.77,
     2           0.50, 0.59, 0.68, 0.76, 0.84, 1.01, 1.16,
     3           0.50, 0.70, 0.88, 1.05, 1.21, 1.52, 1.83/
      DATA TH15/ 0.27, 0.41, 0.48, 0.50, 0.50, 0.50, 0.50,
     1           0.27, 0.46, 0.58, 0.66, 0.71, 0.81, 0.90,
     2           0.27, 0.58, 0.76, 0.88, 1.00, 1.20, 1.41,
     3           0.27, 0.74, 1.04, 1.28, 1.50, 1.96, 2.40/
      DATA TA16/ .000, .108, .162, .195, .215, .238, .250, .250,
     1           .000, .085, .128, .152, .168, .187, .194, .194,
     2           .000, .070, .110, .132, .147, .165, .170, .170,
     3           .000, .062, .099, .120, .135, .153, .160, .160/
      DATA TB16/ .000, .095, .155, .197, .223, .244, .250, .250,
     1           .000, .085, .140, .173, .194, .215, .221, .221,
     2           .000, .075, .125, .162, .183, .205, .210, .210,
     3           .000, .068, .115, .154, .175, .197, .204, .204/
      DATA TC16/ .000, .100, .158, .190, .210, .230, .250, .250,
     1           .000, .100, .158, .190, .210, .230, .250, .250,
     2           .000, .100, .158, .190, .210, .230, .250, .250,
     3           .000, .100, .158, .190, .210, .230, .250, .250/
      DATA TD16/ .250, .250, .250, .250, .250, .250, .250, .250,
     1           .250, .263, .275, .284, .290, .301, .305, .305,
     2           .250, .270, .286, .300, .309, .321, .325, .325,
```

```
     3           .250, .275, .294, .308, .317, .329, .335, .335/
      DATA TE16/ .140, .163, .190, .210, .225, .242, .250, .250,
     1           .140, .177, .210, .233, .250, .269, .275, .275,
     2           .140, .185, .220, .243, .260, .279, .285, .285,
     3           .140, .195, .230, .253, .268, .287, .291, .291/
      DATA TG16/ .500, .465, .432, .400, .370, .320, .286, .286,
     1           .500, .483, .467, .457, .440, .417, .409, .409,
     2           .500, .496, .492, .487, .483, .476, .473, .473,
     3           .500, .504, .507, .510, .511, .512, .514, .514/
      DATA TM16/ .250, .250, .250, .250, .250, .250, .250, .250,
     1           .250, .275, .292, .305, .315, .327, .330, .330,
     2           .250, .280, .304, .321, .336, .353, .360, .360,
     3           .250, .290, .315, .332, .353, .372, .380, .380/
      DO 1 J=1,7
      DO 1 K=1,4
      TF15(J,K)=TC15(J,K)
    1 TI15(J,K)=TC15(J,K)
      DO 2 J=1,8
      DO 2 K=1,4
      TF16(J,K)=TC16(J,K)
    2 TI16(J,K)=TC16(J,K)
      CTLE=(S-R)*TAN(SWPLE/RAD)
      SWPMC=RAD*ATAN((CTLE+0.5*(CT-CR))/(S-R))
      SWPTE=RAD*ATAN((CTLE+CT-CR)/(S-R))
      IF (FMACH.LE.1.0) GO TO 100
      IF (ARPARM.LT.4.0) GO TO 100
C
C     CHART 14
C
      IF (NAFTER.EQ.0) GO TU 50
C
C     CHART 14(A)
C
      IF (BDCR.GT.1.0) GO TO 25
      IF (BDCR.GT.0.4) GO TO 10
      XBCR=0.5+0.55*BDCR
      RETURN
   10 IF (BDCR.GT.0.8) GO TO 15
      XBCR=0.72+0.5*(BDCR-0.4)
      RETURN
   15 XBCR=0.92+0.4*(BDCR-0.8)
      RETURN
   25 IF (SMB.GT.0.2) GU TO 30
      XBCR=1.0+0.378*(BDCR-1.0)
      RETURN
   30 IF (SMB.GT.2.0) GO TO 35
      XBCR=1.0+0.4*(BDCR-1.0)
      RETURN
   35 XBCR=1.0+0.411*(BDCR-1.0)
      RETURN
C
C     CHART 14(B)
C
   50 IF (BDCR.LT.1.0) GO TO 55
      XBCR=0.667
      RETURN
   55 L=1
      IF (SMB.GT.1.0) L=2
      J=0
   60 J=J+1
      IF (BDCR.GT.BD14B(J)) GO TO 60
      JM=J-1
      XBCR=XB14B(JM,L)+(XB14B(J,L)-XB14B(JM,L))*(BDCR-BD14B(JM))/
     1(BD14B(J)-BD14B(JM))
      RETURN
C
C     CHART 15 AND 16
C
  100 IF (SWPMC.LE.0.0) GO TO 105
      LM=2
      LP=3
      DXCL=SWPMC/(SWPMC-SWPTE)
      GO TU 110
  105 LM=1
      LP=2
      DXCL=1.0
      IF (TAPER.NE.1.0) DXCL=SWPLE/(SWPLE-SWPMC)
  110 IF (TAPER.LE.0.5) GO TO 115
      KM=2
      KP=3
      DXSL=(TAPER-0.5)/0.5
      GO TO 120
  115 KM=1
      KP=2
      DXSL=TAPER/0.5
  120 IF (ROB.GT.0.2) GO TO 125
      JM=1
      JP=2
      DXRB=ROB/0.2
      GO TO 135
  125 IF (ROB.GT.0.4) GO TO 130
      JM=2
      JP=3
      DXRB=(ROB-0.2)/0.2
      GO TO 135
  130 JM=3
      JP=4
      DXRB=(ROB-0.4)/0.2
  135 IF (FMACH.LE.1.0) GO TO 200
C
C     CHART 15
C
      I=0
  140 I=I+1
      IF (BAR.GE.BA15(I)) GO TU 140
      IM=I-1
      IP=I
      DXBA=(BAR-BA15(IM))/(BA15(IP)-BA15(IM))
      DO 145 K=KM,KP
      DO 145 J=JM,JP
      VA(J,K,1)=XB15A(IM,J,K)+DXBA*(XB15A(IP,J,K)-XB15A(IM,J,K))
      VA(J,K,2)=XB15B(IM,J,K)+DXBA*(XB15B(IP,J,K)-XB15B(IM,J,K))
  145 VA(J,K,3)=XB15C(IM,J,K)+DXBA*(XB15C(IP,J,K)-XB15C(IM,J,K))
      DO 150 L=LM,LP
      DO 150 K=KM,KP
  150 V(K,L)=VA(JM,K,L)+DXRB*(VA(JP,K,L)-VA(JM,K,L))
      VALA=V(KM,LM)+DXSL*(V(KP,LM)-V(KM,LM))
      VALB=V(KM,LP)+DXSL*(V(KP,LP)-V(KP,LP))
      XBCR=VALA+DXCL*(VALB-VALA)
      RETURN
C
C     CHART 16
C
  200 I=0
  210 I=I+1
      IF (BAR.GE.BA16(I)) GU TO 210
      IM=I-1
      IP=I
      DXBA=(BAR-BA16(IM))/(BA16(IP)-BA16(IM))
      DO 220 K=KM,KP
      DO 220 J=JM,JP
      VA(J,K,1)=XB16A(IM,J,K)+DXBA*(XB16A(IP,J,K)-XB16A(IM,J,K))
      VA(J,K,2)=XB16B(IM,J,K)+DXBA*(XB16B(IP,J,K)-XB16B(IM,J,K))
  220 VA(J,K,3)=XB16C(IM,J,K)+DXBA*(XB16C(IP,J,K)-XB16C(IM,J,K))
      DO 230 L=LM,LP
      DO 230 K=KM,KP
  230 V(K,L)=VA(JM,K,L)+DXRB*(VA(JP,K,L)-VA(JM,K,L))
      VALA=V(KM,LM)+DXSL*(V(KP,LM)-V(KM,LM))
      VALB=V(KM,LP)+DXSL*(V(KP,LP)-V(KM,LP))
      XBCR=VALA+DXCL*(VALB-VALA)
      RETURN
      END
```

```
      SUBROUTINE CH56(S,R,SWPLE,CT,CR,FMACH,TAPER,BAR,FWBR)
C
C     SUBROUTINE TO CALCULATE LATERAL POSITION OF WING VORTEX, CHART 5
C         OR CHART 6 OF REPORT 1307, IN TABLE I OF THAT REPORT THIS
C         QUANTITY IS IN BOX 60.
C
      DIMENSION BA(9),              CH5(9,3,3),CH6(9,3,3),V(3,3)
      DATA BA/ 0.0, 1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0/
      DATA CH5/ .783, .729, .695, .675, .650, .638, .629, .623, .616,
     1          .783, .766, .760, .758, .758, .758, .758, .755, .750,
     2          .783, .784, .799, .812, .825, .833, .841, .850, .858,
     3          .783, .770, .748, .720, .700, .684, .666, .650, .641,
     4          .783, .783, .782, .779, .777, .775, .775, .770, .768,
     5          .783, .787, .800, .811, .825, .835, .843, .850, .853,
     6          .783, .781, .766, .750, .736, .720, .708, .692, .675,
     7          .783, .784, .792, .792, .792, .792, .792, .792, .792,
     8          .783, .784, .799, .812, .821, .831, .840, .845, .853/
       DATA CH6/ .783, .741, .670, .548, .500, .500, .500, .500, .500,
     1          .783, .758, .708, .660, .700, .714, .717, .720, .720,
     2          .783, .790, .753, .832, .882, .908, .923, .929, .931,
     3          .783, .735, .667, .630, .608, .592, .576, .566, .559,
     4          .783, .775, .729, .741, .753, .758, .766, .763, .758,
     5          .783, .787, .754, .831, .880, .908, .925, .933, .940,
     6          .783, .783, .787, .790, .792, .737, .702, .676, .657,
     7          .783, .781, .800, .833, .841, .833, .817, .813, .814,
     8          .783, .783, .751, .836, .883, .912, .929, .937, .944/
      DATA RAD/57.29578/
      CTLE=(S-R)*TAN(SWPLE/RAD)
      SWPMC=RAD*ATAN((CTLE+0.5*(CT-CR))/(S-R))
      SWPTE=RAD*ATAN((CTLE+CT-CR)/(S-R))
      IF (SWPMC.LE.0.0) GO TO 10
      KM=2
      KP=3
      DXCL=SWPMC/(SWPMC-SWPTE)
      GO TO 20
   10 KM=1
      KP=2
      DXCL=1.0
      IF (TAPER.NE.1.0) DXCL=SWPLE/(SWPLE-SWPMC)
   20 IF (TAPER.LE.0.5) GO TO 30
      JM=2
      JP=3
      DXSL=(TAPER-0.5)/0.5
      GO TO 40
   30 JM=1
      JP=2
      DXSL=TAPER/0.5
   40 I=0
   50 I=I+1
      IF (I.EQ.9) GO TO 60
      IF (BAR.GE.BA(I)) GO TO 50
   60 IM=I-1
      IP=I
      DXBA=(BAR-BA(IM))/(BA(IP)-BA(IM))
      IF (FMACH.GT.1.0) GO TO 75
      DO 70 K=KM,KP
      DO 70 J=JM,JP
   70 V(J,K)=CH5(IM,J,K)+DXBA*(CH5(IP,J,K)-CH5(IM,J,K))
      GO TO 85
   75 DO 80 K=KM,KP
      DO 80 J=JM,JP
   80 V(J,K)=CH6(IM,J,K)+DXBA*(CH6(IP,J,K)-CH6(IM,J,K))
   85 VA=V(JM,KM)+DXSL*(V(JP,KM)-V(JM,KM))
      VB=V(JM,KP)+DXSL*(V(JP,KP)-V(JM,KP))
      FRSR=VA+DXCL*(VB-VA)
      RETURN
      END
```

```
      SUBROUTINE KFACT (ROS,CKWB,FMACH,PARAM,CKBW,NAFTER,SMB,TAPER,BDCR,
     1CLA,BETA,SWPLL,BETAA,SKWB,SKBW)
C
C     SUBROUTINE TO CALCULATE INTERFERENCE FACTORS, BOXES 47-50 OR
C         65-66 OF TABLE I OF REPORT 1307
C
      CKWB=EQ14(ROS)
      IF (FMACH.GT.1.0.AND.PARAM.GT.4.0) GO TO 10
      CKBW=EQ21(ROS)
      GO TO 100
   10 IF (NAFTER.EQ.0) GO TO 30
      IF (SMB.GT.1.0) GO TO 15
      CKBW=EQ26(SMB,TAPER,BDCR,ROS)/(BETA*CLA)
      GO TO 100
   15 IF (SWPLE.GT.0.0) GO TO 20
      CKBW=EQ24L(TAPER,BDCR,ROS)/(BETA*CLA)
      GO TO 100
   20 CKBW=EQ24(SMB,TAPER,BDCR,ROS)/(BETA*CLA)
      GO TO 100
   30 IF (SMB.GT.1.0) GO TO 35
      CKBW=EQ31(SMB,TAPER,BDCR,ROS)/(BETA*CLA)
      GO TO 100
   35 IF (SWPLE.GT.0.0) GO TO 40
      CKBW=EQ30L(TAPER,BDCR,ROS)/(BETA*CLA)
      GO TO 100
   40 CKBW=EQ30(SMB,TAPER,BDCR,ROS)/(BETA*CLA)
  100 CONTINUE
      SKWB=EQ19(ROS)
      SKBW=CKWB-SKWB
      RETURN
      END
```

ITEM 1 FORMAT (I5); 1 card

1
NHEAD

ITEM 2 FORMAT (20A4); NHEAD card

1
HEAD

ITEM 3 FORMAT (4I5); 1 card

1	6	11	16
NTAIL	NOSE	NAFTBW	NAFTBT

ITEM 4 FORMAT (4F10.5); 1 card

1	11	21	31
FMACH	SLM	REFS	REFL

ITEM 5 FORMAT (7F10.5); 1 card

1	11	21	31	41	51	61
RW	SW	CTW	CRW	WLESWP	SLW	XHW

ITEM 6 FORMAT (6F10.5); 1 card, omit if NTAIL = 0

1	11	21	31	41	51
RT	ST	CTT	CRT	TLESWP	SLT

ITEM 7 FORMAT (3F10.5); 1 card

1	11	21
RN	SLS	VS

ITEM 8 FORMAT (3F10.5); 1 card

1	11	21
CLAW	CLAT	CLAN

ITEM 9 FORMAT (3F10.5); 1 card

1	11	21
ALFI	ALFF	DALF

ITEM 10 FORMAT (3F10.5); 1 card

1	11	21
DWI	DWF	DDW

ITEM 11 FORMAT (3F10.5); 1 card

1	11	21
DTI	DTF	DDT

Figure 1.- Input format for R1307 program.

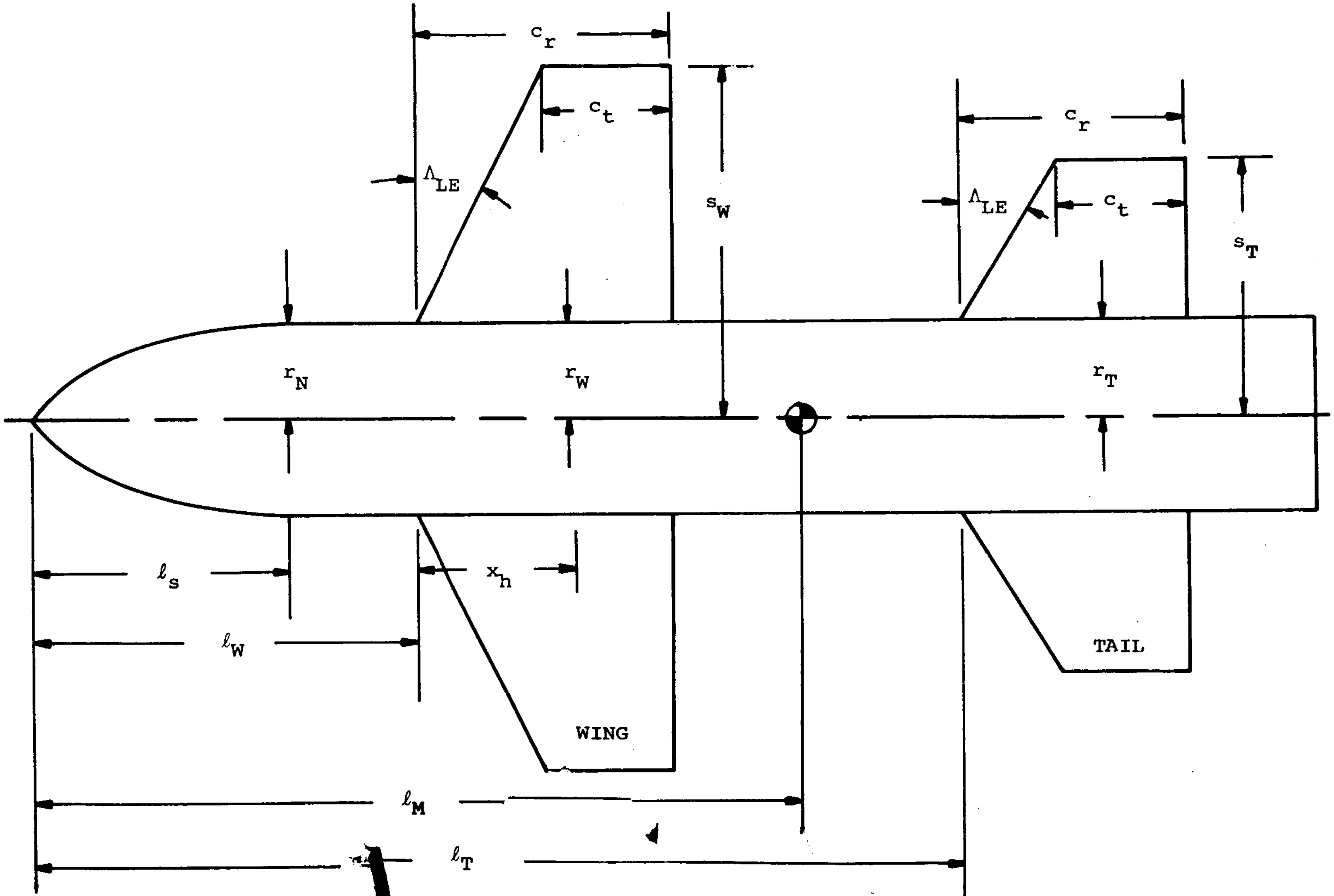


Figure 2.- Algebraic notation defining configuration.

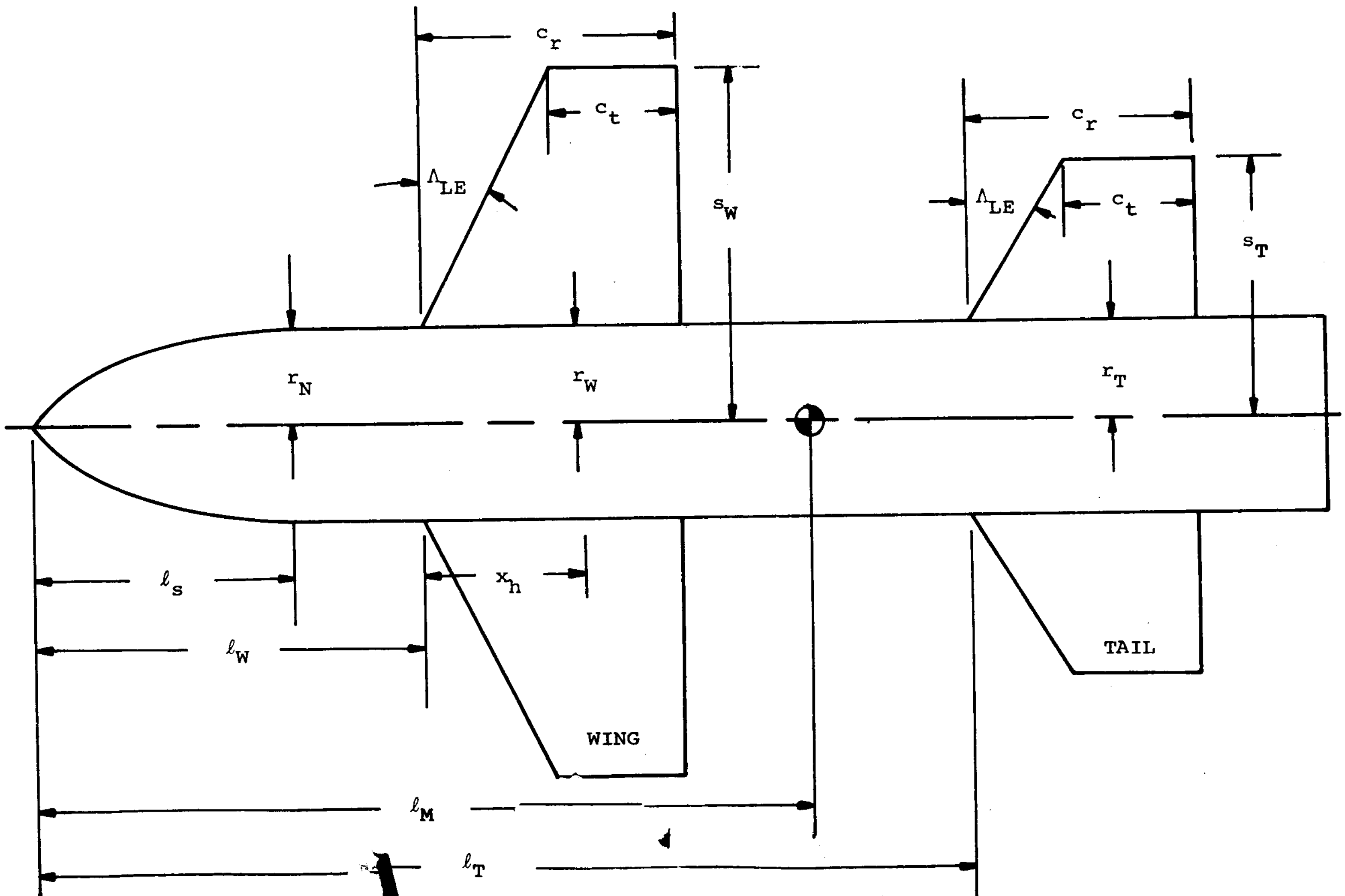


Figure 2.- Algebraic notation defining configuration.

```
   3
SAMPLE CASE 1   WING - BODY - TAIL CONFIGURATION
 SAMPLE CALCULATION OF TABLE I OF REPORT 1307
  THIS IS NO. 101 OF TABLE III OF THAT REPORT
    1    0    1    0
   1.99      5.25      5.062     10.5
   0.562     2.812     0.0        2.25     45.0      3.75      1.375
   0.562     1.812     0.0        1.25     45.0      9.16
   0.562     3.19      1.56
    0.0       0.0       0.0
    0.0      15.0       5.0
    0.0       4.9       4.9
    0.0       4.9       4.9
```

(a) Sample case 1.

```
   2
SAMPLE CASE 2   CANARD - BODY - WING  CONFIGURATION
  REF. NASA TM X-643   JAN. 1962   BRADY, PAGE, KUENIG
 1    1    1    0
.13       39.3      446.      19.11
1.03      5.03      0.        8.05      63.43     7.5      3.5
2.        15.55     0.        25.       59.       29.
.84       7.5       0.
.0385     .0418     .0341
0.        24.       4.
0.        15.       5.
0.        0.        0.
```

(b) Sample case 2.

Figure 3.- Sample input decks for R1307 program.

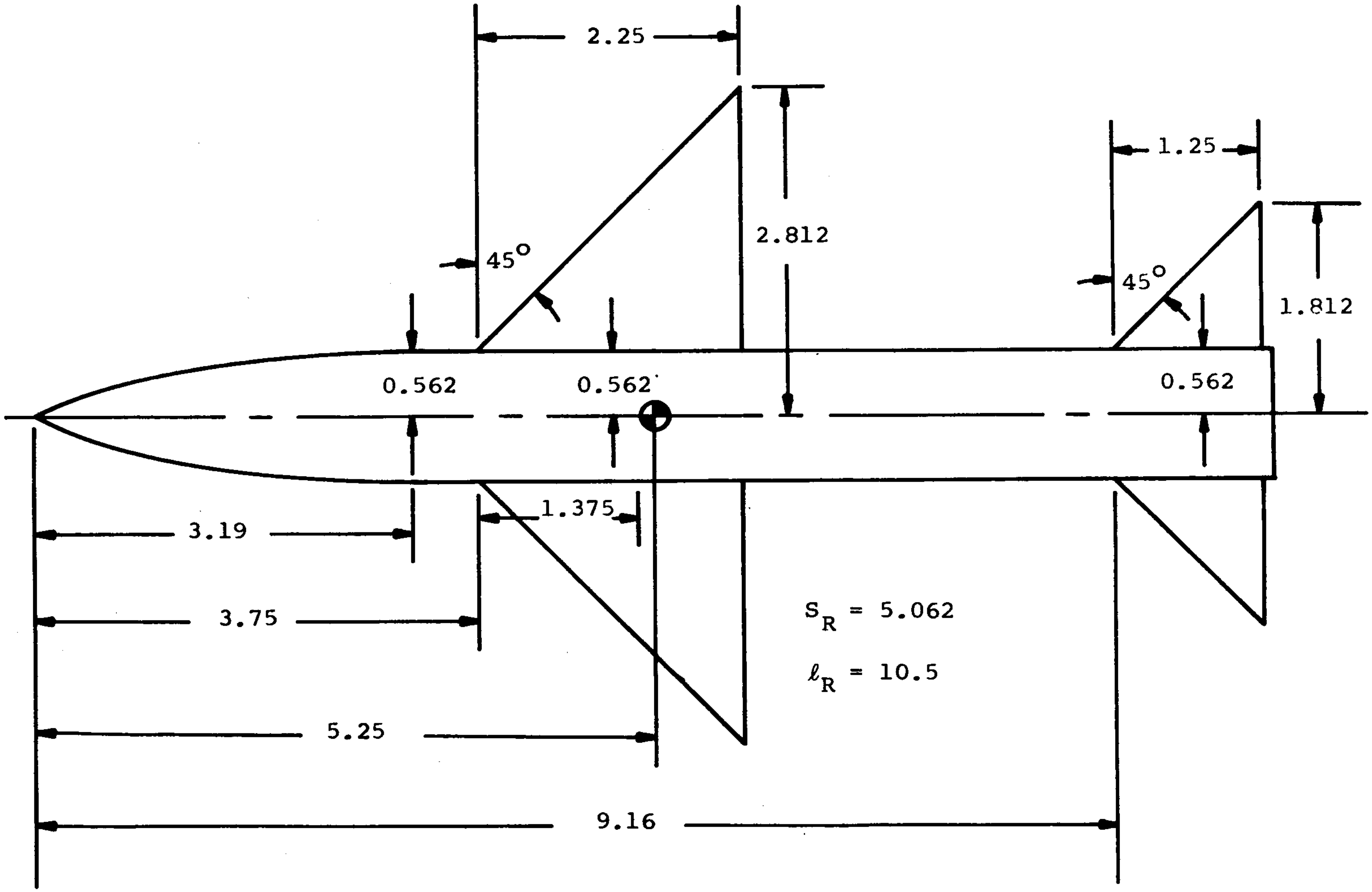


Figure 4.- Sample case 1 configuration.

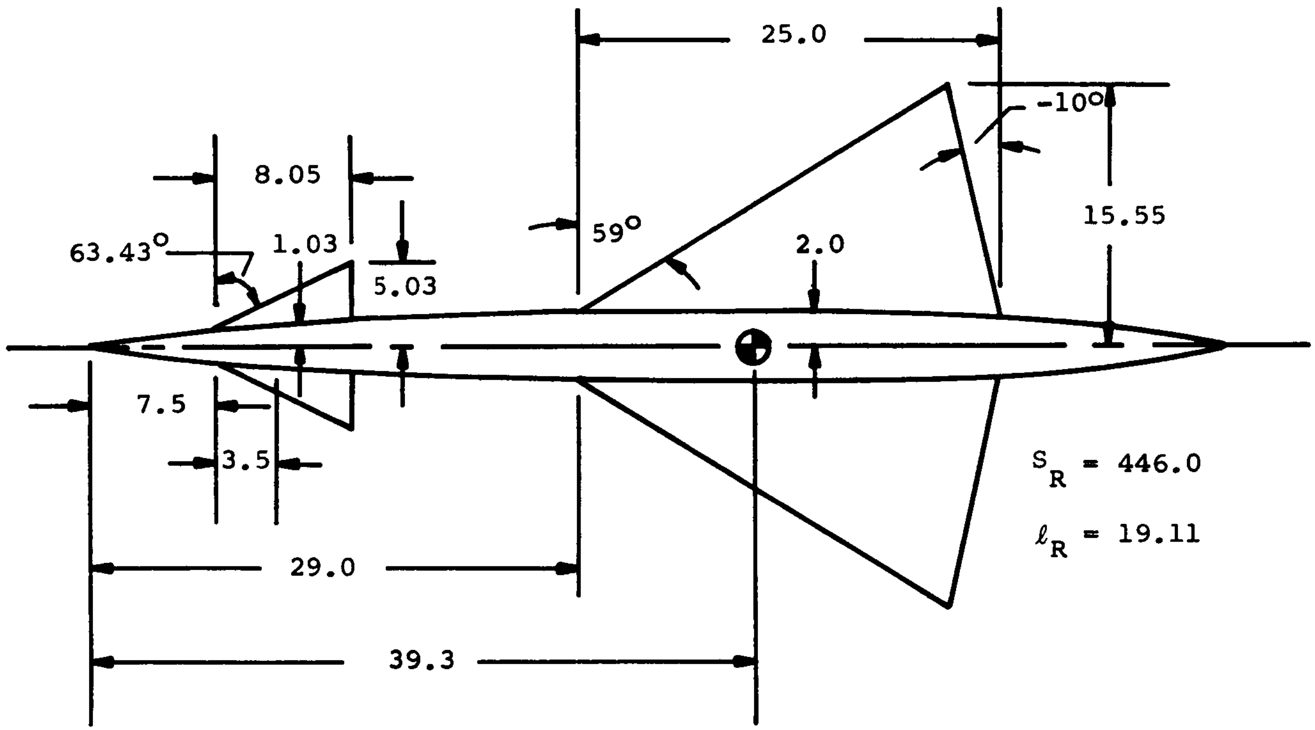


Figure 5.- Sample case 2 configuration.

```
            LIFT AND CENTER OF PRESSURE OF WING-BODY-TAIL COMBINATIONS
                        METHOD OF NACA REPORT 1307, 1957


SAMPLE CASE 1   WING - BODY - TAIL CONFIGURATION
 SAMPLE CALCULATION OF TABLE I OF REPORT 1307
  THIS IS NO. 101 OF TABLE III OF THAT REPORT


 FLIGHT CONDITIONS
    MACH NUMBER =  1.990

 REFERENCE QUANTITIES
    AREA =     5.06200
    LENGTH = 10.50000

 MOMENT CENTER RELATIVE TO TIP OF NOSE IS AT  5.25000


 WING INPUT DATA
    BODY RADIUS IN REGION =   0.56200
    SEMISPAN =   2.81200
    TIP CHORD =   0.00000
    EXPOSED ROOT CHORD =   2.25000
    LEADING EDGE SWEEP ANGLE =  45.00000
    DISTANCE FROM TIP OF NOSE TO BODY, LEADING EDGE INTERSECTION =   3.75000
    HINGE LINE LOCATION MEASURED FROM BODY, LEADING EDGE INTERSECTION =   1.37500


 TAIL INPUT DATA
    BODY RADIUS IN REGION =   0.56200
    SEMISPAN =   1.81200
    TIP CHORD =   0.00000
    EXPOSED ROOT CHORD =   1.25000
    LEADING EDGE SWEEP ANGLE =  45.00000
    DISTANCE FROM TIP OF NOSE TO BODY, LEADING EDGE INTERSECTION =   9.16000


 NOSE INPUT DATA
    RADIUS AT SHOULDER =   0.56200
    DISTANCE FROM TIP OF NOSE TO SHOULDER =   3.19000
    VOLUME OF NOSE TO SHOULDER =   1.56000
```

(a) Page 1.

Figure 6.- Output from R1307 program for sample case 1.

```
CALCULATED WING QUANTITIES
   EXPOSED AREA =    5.0625
   BETA*AR =  6.38197
   BETA*D/CR =  0.85948
   AR PARAM = 10.88196
   TAPER RATIO =  0.00000
   R/S =  0.19986
   SM*BETA =  1.72049
   CKWB =  1.16151
   CKBW =  0.23115
   SKWB =  0.94389
   SKBW =  0.21762
   (XBAR/CR)W(B) =  0.66700
   (XBAR/CR)B(W) =  0.94379
   (F-R)W/(S-R)W =  0.67907


CALCULATED TAIL QUANTITIES
   EXPOSED AREA =    1.5625
   BETA*AR =  6.38197
   BETA*D/CR =  1.54707
   AR PARAM = 10.88196
   TAPER RATIO =  0.00000
   R/S =  0.31015
   SM*BETA =  1.72049
   CKTB =  1.25233
   CKBT =  0.12817
   SKTB =  0.93548
   SKBT =  0.32686
   (XBAR/CR)T(B) =  0.66700
   (XBAR/CR)B(T) =  0.66700


CALCULATED NOSE QUANTITIES
   BASE AREA =  0.99225
   CKN =  0.16861
   NOSE CENTER OF PRESSURE =  1.61782


LIFT-CURVE SLOPES, PER DEGREE
      (CLA)W        (CLA)T        (CLA)N
      0.04058       0.04058       0.03491
```

(b) Page 2.

Figure 6.- Continued.

LIFT AND MOMENT CURVE SLOPES OF COMPLETE CONFIGURATION
NO WING-TAIL INTERFERENCE
PER DEGREE

D(CL)/D(A)	D(CL)/D(DW)	D(CL)/D(DT)	D(CM)/D(A)	D(CM)/D(DW)	D(CM)/D(DT)
0.08077	0.04714	0.01581	-0.00606	-0.00053	-0.00714

LIFT AND MOMENT CURVE SLOPES OF CONFIGURATION COMPONENTS
WITH WING-TAIL INTERFERENCE
PER DEGREE

	BODY	WING-BODY	TAIL-BODY MINUS NOSE	WING-BODY-TAIL
D(CL)/D(A), A=DW=0	0.00684	0.06336	0.01742	0.07298
D(CM)/D(A), A=DW=0	0.00237	0.00181	-0.00787	-0.00254
D(CL)/D(DW), A=DW=0	0.00000	0.04714	0.00000	0.04080
D(CM)/D(DW), A=DW=0	0.00000	-0.00053	0.00000	0.00233

LIFT AND MOMENT COEFFICIENTS AND CENTER OF PRESSURE
FOR THE COMPLETE CONFIGURATION

			NO WING-TAIL INTERFERENCE			WITH WING-TAIL INTERFERENCE		
A	DW	DT	CLC	CMC	LM-L/LR	CLC	CMC	LM-L/LR
0.00	0.00	0.00	0.0000	0.0000	100.0000	0.0000	0.0000	100.0000
5.00	0.00	0.00	0.4039	-0.0303	-0.0750	0.3685	-0.0143	-0.0389
10.00	0.00	0.00	0.8077	-0.0606	-0.0750	0.7489	-0.0340	-0.0455
15.00	0.00	0.00	1.2116	-0.0909	-0.0750	1.1395	-0.0583	-0.0512
0.00	4.90	0.00	0.2310	-0.0026	-0.0112	0.2001	0.0114	0.0568
5.00	4.90	0.00	0.6348	-0.0329	-0.0518	0.5691	-0.0032	-0.0056
10.00	4.90	0.00	1.0387	-0.0632	-0.0608	0.9529	-0.0244	-0.0256
15.00	4.90	0.00	1.4426	-0.0935	-0.0648	1.3473	-0.0504	-0.0374
0.00	0.00	4.90	0.0775	-0.0350	-0.4518	0.0775	-0.0350	-0.4518
5.00	0.00	4.90	0.4813	-0.0653	-0.1357	0.4459	-0.0493	-0.1106
10.00	0.00	4.90	0.8852	-0.0956	-0.1080	0.8264	-0.0690	-0.0836
15.00	0.00	4.90	1.2891	-0.1259	-0.0977	1.2169	-0.0933	-0.0767
0.00	4.90	4.90	0.3084	-0.0376	-0.1219	0.2776	-0.0236	-0.0852
5.00	4.90	4.90	0.7123	-0.0679	-0.0953	0.6466	-0.0382	-0.0591
10.00	4.90	4.90	1.1162	-0.0982	-0.0880	1.0304	-0.0594	-0.0577
15.00	4.90	4.90	1.5201	-0.1285	-0.0845	1.4247	-0.0854	-0.0600

(c) Page 3.

Figure 6.- Continued.

LIFT AND MOMENT COEFFICIENT COMPONENTS WITH WING-TAIL INTERFERENCE

			LIFT COEFFICIENTS				MOMENT COEFFICIENTS			
			NOSE	WING	TAIL	TAIL	NOSE	WING	TAIL	TAIL
A	DW	DT	A	A+DW	A+DT	WING INT	A	A+DW	A+DT	WING INT
0.00	0.00	0.00	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
5.00	0.00	0.00	0.0342	0.2826	0.0871	-0.0354	0.0118	-0.0028	-0.0393	0.0160
10.00	0.00	0.00	0.0684	0.5652	0.1742	-0.0588	0.0237	-0.0056	-0.0787	0.0266
15.00	0.00	0.00	0.1026	0.8477	0.2612	-0.0721	0.0355	-0.0084	-0.1180	0.0326
0.00	4.90	0.00	0.0000	0.2310	0.0000	-0.0309	0.0000	-0.0026	0.0000	0.0139
5.00	4.90	0.00	0.0342	0.5135	0.0871	-0.0657	0.0118	-0.0054	-0.0393	0.0297
10.00	4.90	0.00	0.0684	0.7961	0.1742	-0.0858	0.0237	-0.0082	-0.0787	0.0388
15.00	4.90	0.00	0.1026	1.0787	0.2612	-0.0953	0.0355	-0.0110	-0.1180	0.0431
0.00	0.00	4.90	0.0000	0.0000	0.0775	0.0000	0.0000	0.0000	-0.0350	0.0000
5.00	0.00	4.90	0.0342	0.2826	0.1646	-0.0354	0.0118	-0.0028	-0.0743	0.0160
10.00	0.00	4.90	0.0684	0.5652	0.2516	-0.0588	0.0237	-0.0056	-0.1137	0.0266
15.00	0.00	4.90	0.1026	0.8477	0.3387	-0.0721	0.0355	-0.0084	-0.1530	0.0326
0.00	4.90	4.90	0.0000	0.2310	0.0775	-0.0309	0.0000	-0.0026	-0.0350	0.0139
5.00	4.90	4.90	0.0342	0.5135	0.1646	-0.0657	0.0118	-0.0054	-0.0743	0.0297
10.00	4.90	4.90	0.0684	0.7961	0.2516	-0.0858	0.0237	-0.0082	-0.1137	0.0388
15.00	4.90	4.90	0.1026	1.0787	0.3387	-0.0953	0.0355	-0.0110	-0.1530	0.0431

(d) Page 4.

Figure 6.- Concluded.

```
LIFT AND CENTER OF PRESSURE OF WING-BODY-TAIL COMBINATIONS
            METHOD OF NACA REPORT 1307, 1957


  SAMPLE CASE 2   CANARD - BODY - WING  CONFIGURATION
    REF.  NASA TM X-643   JAN. 1962   BRADY, PAGE, KOENIG

FLIGHT CONDITIONS
   MACH NUMBER =  0.130

REFERENCE QUANTITIES
   AREA =  446.00000
   LENGTH = 19.10999

MOMENT CENTER RELATIVE TO TIP OF NOSE IS AT 39.29999


WING INPUT DATA
   BODY RADIUS IN REGION =   1.03000
   SEMISPAN =   5.03000
   TIP CHORD =   0.00000
   EXPOSED ROOT CHORD =   8.05000
   LEADING EDGE SWEEP ANGLE =  63.42999
   DISTANCE FROM TIP OF NOSE TO BODY, LEADING EDGE INTERSECTION =   7.50000
   HINGE LINE LOCATION MEASURED FROM BODY, LEADING EDGE INTERSECTION =   3.50000


TAIL INPUT DATA
   BODY RADIUS IN REGION =   2.00000
   SEMISPAN =  15.55000
   TIP CHORD =   0.00000
   EXPOSED ROOT CHORD =  25.00000
   LEADING EDGE SWEEP ANGLE =  59.00000
   DISTANCE FROM TIP OF NOSE TO BODY, LEADING EDGE INTERSECTION =  29.00000


NOSE INPUT DATA
   RADIUS AT SHOULDER =   0.84000
   DISTANCE FROM TIP OF NOSE TO SHOULDER =   7.50000
   VOLUME OF NOSE TO SHOULDER =   0.00000
```

(a) Page 1.

Figure 7.- Output from R1307 program for sample case 2.

```
CALCULATED WING QUANTITIES
   EXPOSED AREA =  32.2000
   BETA*AR =  1.97071
   BETA*D/CR =  0.25373
   AR PARAM =  5.94499
   TAPER RATIO =  0.00000
   P/S =  0.20477
   SM*BETA =  0.49587
   CKWB =  1.16585
   CKBW =  0.28563
   SKWB =  0.94328
   SKBW =  0.22257
   (XBAR/CR)W(B) =  0.55112
   (XBAR/CR)B(W) =  0.43253
   (F-R)W/(S-R)W =  0.76615

CALCULATED TAIL QUANTITIES
   EXPOSED AREA = 338.7498
   BETA*AR =  2.14960
   BETA*D/CR =  0.15864
   AR PARAM =  5.75776
   TAPER RATIO =  0.00000
   R/S =  0.12862
   SM*BETA =  0.59576
   CKTB =  1.10047
   CKBT =  0.17331
   SKTB =  0.95595
   SKBT =  0.14452
   (XBAR/CR)T(B) =  0.50990
   (XBAR/CR)B(T) =  0.38095

CALCULATED NOSE QUANTITIES
   BASE AREA =  2.21670
   CKN =  0.06097
   NOSE CENTER OF PRESSURE =  3.46735

LIFT-CURVE SLOPES, PER DEGREE
      (CLA)W        (CLA)T        (CLA)N
      0.03850       0.04180       0.03410
```

(b) Page 2.

Figure 7.- Continued.

LIFT AND MOMENT CURVE SLOPES OF COMPLETE CONFIGURATION
NO WING-TAIL INTERFERENCE
PER DEGREE

D(CL)/D(A)	D(CL)/D(DW)	D(CL)/D(DT)	D(CM)/D(A)	D(CM)/D(DW)	D(CM)/D(DT)
0.04464	0.00324	0.03494	0.00188	0.00467	-0.00370

LIFT AND MOMENT CURVE SLOPES OF CONFIGURATION COMPONENTS
WITH WING-TAIL INTERFERENCE
PER DEGREE

		BODY	WING-BODY	TAIL-BODY MINUS NOSE	WING-BODY-TAIL
D(CL)/D(A),	A=DW=0	0.00017	0.00420	0.04044	0.03824
D(CM)/D(A),	A=DW=0	0.00032	0.00613	-0.00425	0.00270
D(CL)/D(DW),	A=DW=0	0.00000	0.00324	0.00000	-0.00194
D(CM)/D(DW),	A=DW=0	0.00000	0.00467	0.00000	0.00533

LIFT AND MOMENT COEFFICIENTS AND CENTER OF PRESSURE
FOR THE COMPLETE CONFIGURATION

			NO WING-TAIL INTERFERENCE			WITH WING-TAIL INTERFERENCE		
A	DW	DT	CLC	CMC	LM-L/LR	CLC	CMC	LM-L/LR
0.00	0.00	0.00	0.0000	0.0000	100.0000	0.0000	0.0000	100.0000
4.00	0.00	0.00	0.1786	0.0075	0.0421	0.1597	0.0099	0.0623
8.00	0.00	0.00	0.3572	0.0150	0.0421	0.3276	0.0188	0.0575
12.00	0.00	0.00	0.5357	0.0226	0.0421	0.5000	0.0271	0.0543
16.00	0.00	0.00	0.7143	0.0301	0.0421	0.6753	0.0351	0.0520
20.00	0.00	0.00	0.8929	0.0376	0.0421	0.8527	0.0428	0.0502
24.00	0.00	0.00	1.0715	0.0451	0.0421	1.0313	0.0503	0.0488
0.00	5.00	0.00	0.0162	0.0233	1.4409	-0.0079	0.0264	-3.3435
4.00	5.00	0.00	0.1948	0.0309	0.1585	0.1544	0.0360	0.2334
8.00	5.00	0.00	0.3734	0.0384	0.1028	0.3266	0.0444	0.1359
12.00	5.00	0.00	0.5519	0.0459	0.0832	0.5019	0.0523	0.1042
16.00	5.00	0.00	0.7305	0.0534	0.0732	0.6796	0.0600	0.0882
20.00	5.00	0.00	0.9091	0.0610	0.0671	0.8588	0.0674	0.0785
24.00	5.00	0.00	1.0877	0.0685	0.0630	1.0389	0.0747	0.0719
0.00	10.00	0.00	0.0324	0.0467	1.4409	-0.0126	0.0525	-4.1598
4.00	10.00	0.00	0.2110	0.0542	0.2570	0.1464	0.0625	0.4270
8.00	10.00	0.00	0.3896	0.0617	0.1585	0.3238	0.0702	0.2167
12.00	10.00	0.00	0.5681	0.0693	0.1219	0.5026	0.0777	0.1545
16.00	10.00	0.00	0.7467	0.0768	0.1028	0.6828	0.0850	0.1245
20.00	10.00	0.00	0.9253	0.0843	0.0911	0.8640	0.0922	0.1067
24.00	10.00	0.00	1.1039	0.0918	0.0832	1.0458	0.0993	0.0949
0.00	15.00	0.00	0.0486	0.0700	1.4409	-0.0146	0.0781	-5.3420
4.00	15.00	0.00	0.2272	0.0776	0.3414	0.1350	0.0894	0.6621
8.00	15.00	0.00	0.4058	0.0851	0.2097	0.3190	0.0962	0.3016
12.00	15.00	0.00	0.5843	0.0926	0.1585	0.5018	0.1032	0.2056
16.00	15.00	0.00	0.7629	0.1001	0.1313	0.6849	0.1101	0.1608
20.00	15.00	0.00	0.9415	0.1077	0.1143	0.8684	0.1170	0.1348
24.00	15.00	0.00	1.1201	0.1152	0.1028	1.0520	0.1239	0.1178

(c) Page 3.

Figure 7.- Continued.

LIFT AND MOMENT COEFFICIENT COMPONENTS WITH WING-TAIL INTERFERENCE

			LIFT COEFFICIENTS				MOMENT COEFFICIENTS			
			NOSE	WING	TAIL	TAIL	NOSE	WING	TAIL	TAIL
A	DW	DT	A	A+DW	A+DT	WING INT	A	A+DW	A+DT	WING INT
0.00	0.00	0.00	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4.00	0.00	0.00	0.0007	0.0161	0.1618	-0.0189	0.0013	0.0233	-0.0170	0.0024
8.00	0.00	0.00	0.0014	0.0323	0.3235	-0.0295	0.0025	0.0465	-0.0340	0.0038
12.00	0.00	0.00	0.0020	0.0484	0.4853	-0.0357	0.0038	0.0698	-0.0510	0.0046
16.00	0.00	0.00	0.0027	0.0646	0.6470	-0.0390	0.0051	0.0930	-0.0680	0.0050
20.00	0.00	0.00	0.0034	0.0807	0.8088	-0.0402	0.0064	0.1163	-0.0850	0.0051
24.00	0.00	0.00	0.0041	0.0968	0.9706	-0.0401	0.0076	0.1395	-0.1020	0.0051
0.00	5.00	0.00	0.0000	0.0162	0.0000	-0.0241	0.0000	0.0233	0.0000	0.0031
4.00	5.00	0.00	0.0007	0.0323	0.1618	-0.0403	0.0013	0.0466	-0.0170	0.0052
8.00	5.00	0.00	0.0014	0.0485	0.3235	-0.0467	0.0025	0.0699	-0.0340	0.0060
12.00	5.00	0.00	0.0020	0.0646	0.4853	-0.0500	0.0038	0.0931	-0.0510	0.0064
16.00	5.00	0.00	0.0027	0.0808	0.6470	-0.0510	0.0051	0.1164	-0.0680	0.0065
20.00	5.00	0.00	0.0034	0.0969	0.8088	-0.0503	0.0064	0.1396	-0.0850	0.0064
24.00	5.00	0.00	0.0041	0.1130	0.9706	-0.0488	0.0076	0.1629	-0.1020	0.0062
0.00	10.00	0.00	0.0000	0.0324	0.0000	-0.0450	0.0000	0.0467	0.0000	0.0058
4.00	10.00	0.00	0.0007	0.0485	0.1618	-0.0646	0.0013	0.0699	-0.0170	0.0083
8.00	10.00	0.00	0.0014	0.0647	0.3235	-0.0657	0.0025	0.0932	-0.0340	0.0084
12.00	10.00	0.00	0.0020	0.0808	0.4853	-0.0655	0.0038	0.1165	-0.0510	0.0084
16.00	10.00	0.00	0.0027	0.0970	0.6470	-0.0640	0.0051	0.1397	-0.0680	0.0082
20.00	10.00	0.00	0.0034	0.1131	0.8088	-0.0613	0.0064	0.1630	-0.0850	0.0078
24.00	10.00	0.00	0.0041	0.1292	0.9706	-0.0581	0.0076	0.1862	-0.1020	0.0074
0.00	15.00	0.00	0.0000	0.0486	0.0000	-0.0632	0.0000	0.0700	0.0000	0.0081
4.00	15.00	0.00	0.0007	0.0647	0.1618	-0.0922	0.0013	0.0933	-0.0170	0.0118
8.00	15.00	0.00	0.0014	0.0809	0.3235	-0.0868	0.0025	0.1166	-0.0340	0.0111
12.00	15.00	0.00	0.0020	0.0970	0.4853	-0.0825	0.0038	0.1398	-0.0510	0.0106
16.00	15.00	0.00	0.0027	0.1132	0.6470	-0.0780	0.0051	0.1631	-0.0680	0.0100
20.00	15.00	0.00	0.0034	0.1293	0.8088	-0.0731	0.0064	0.1863	-0.0850	0.0094
24.00	15.00	0.00	0.0041	0.1454	0.9706	-0.0680	0.0076	0.2096	-0.1020	0.0087

(d) Page 4.

Figure 7.- Concluded.

PART II - CRSFLW COMPUTER PROGRAM

Introduction

This computer program is based on the method presented in NASA TN D-6996 and TN D-7228 (refs. 3 and 4) to calculate the normal force, axial force, and pitching moment of a body alone or wing-body combination. The bodies must be slender but may have circular or noncircular cross sections, and the method is not restricted to small angles of attack. The method of calculation of normal-force and pitching-moment coefficients is based on the concept from reference 7 that the normal-force distribution over a body is made up of a potential term given by slender-body theory and a viscous crossflow term modified by Newtonian theory. Empirical information on crossflow drag coefficients as a function of Mach number and Reynolds number is incorporated into the program. This allows the procedure to be applied over a wide range of angles of attack, Mach numbers, and Reynolds numbers.

The following sections present a description of the program, a description of the input, a description of the output, a program listing, and sample cases. A list of symbols from references 3 and 4 is included for reference.

The program is written in FORTRAN IV for the IBM 360 series machines. No tapes, drums, or disks other than the standard input/output units are required. The running time for a typical case on the IBM 360/67 is three seconds. The program will run on other machines such as the CDC 6600 with no modifications.

List of Symbols

A_b body base area (at $x = \ell$)

A_p planform area

A_r reference area

a,b semimajor and semiminor axes of elliptic cross section

C_A axial-force coefficient, $\frac{F_a}{q_\infty A_r}$

c_{d_n} crossflow drag coefficient of circular cylinder section, $\frac{F_n}{q_n(\Delta \ell_{cy}) d_{cy}}$

C_D drag coefficient, $\frac{\text{drag}}{q_\infty A_r}$

C_L lift coefficient, $\frac{\text{lift}}{q_\infty A_r}$

C_m pitching-moment coefficient about station x_m from nose, $\frac{\text{pitching moment}}{q_\infty A_r X}$

C_N normal-force coefficient, $\frac{F_n}{q_\infty A_r}$

c_n local normal-force coefficient per unit length

d body cross-section diameter

D drag

k ratio of corner radius to body width for bodies of square cross section

ℓ body length

L lift

M_∞ free-stream Mach number

q_∞ free-stream dynamic pressure, $\frac{1}{2} \rho V_\infty^2$

r body cross-section radius

Re free-stream Reynolds number, $\frac{\rho V_\infty d}{\mu}$

s wing semispan

V body volume

V_∞ free-stream velocity

w body width

X reference length

x axial distance from body nose

x_{ac} distance from nose to aerodynamic force center

x_c distance from nose to centroid of body planform area

x_m distance from nose to pitching-moment reference center

α angle of attack

γ ratio of specific heats (taken as 1.4 for air)

ϵ wing planform semiapex angle

η crossflow drag proportionality factor

μ viscosity coefficient of air

ρ density of air

Subscripts

cy cylinder

LE leading edge

Newt Newtonian theory

SB slender-body theory

SF skin friction

TE trailing edge

W wave or pressure

Description of Program

The CRSFLW computer program consists of a main program, eight function subprograms, and two subroutine subprograms. All input and output takes place in the main program. The function and subroutine subprograms provide quantities from the curves and equations of references 3 and 4 and other specific services to the main program.

Function subprogram CNSB computes the ratio $(C_n/C_{n_o})_{SB}$ for winged circular cross-section bodies and winged elliptic cross-section bodies with the major axis parallel to the crossflow velocity or normal to the crossflow using equations (13), (14), and (15) of reference 4, respectively. This same routine is used for winged-bodies with varying cross sections in which the above ratio changes with x-distance.

Function CNNT computes the ratio $(C_n/C_{n_o})_{Newt}$ for the same wing-body configurations considered above using equations (16), (17), and (18) of reference 4.

Function CNRSB computes $(C_n/C_{n_o})_{SB}$ for bodies alone with similar cross sections over their length. The result is identically 1.0 for circular bodies, and results for elliptic bodies with their major axis perpendicular to the crossflow velocity, or normal to it, are computed using equations (21) and (22) of reference 3, respectively. For bodies with square cross sections with rounded corners, $(C_n/C_{n_o})_{SB} = 1.19$ at $k = 0$ (no corner radius) and $(C_n/C_{n_o})_{SB} = 1.0$ at $k = 0.5$ (circular cross section); therefore, linear interpolation between these two end points is used for intermediate cases.

Function CNRNT computes $(C_n/C_{n_o})_{Newt}$ for bodies alone with similar cross section over their length. The result is identically 1.0 for circular bodies, and results for elliptic bodies with their major axis perpendicular to the crossflow, or normal to it, are computed using equations (21) and (22) of reference 3, respectively. Bodies with square cross sections with rounded corners are considered using equation (23) of the same reference.

Function CAW computes the wave or pressure contribution to the axial force, C_{A_W}, for various nose shapes and body combinations at $M > 1$. Forward facing conical-nosed bodies are considered using equation (10) of reference 3. C_{A_W} for tangent ogive noses and for Newtonian minimum drag noses are obtained from correlation curves in figure 6 of the above reference. For circular bodies with flat noses, it is assumed that C_{A_W} is equal to the stagnation pressure coefficient and figure 7 of reference 3 is used. C_{A_W} for conical-nosed bodies of elliptic cross section is computed using equation (28) of the same reference.

Function CDN computes the crossflow drag coefficient, C_{d_n}, of a circular body as a function of crossflow Mach number and crossflow Reynolds number from figures 1, 2, and 3 of reference 3.

Function ETA computes the ratio of the crossflow drag coefficient for a finite length cylinder to that for an infinite length cylinder using figure 4 in reference 3.

Function FUN calculates the integrand in equations (7) and (8) of reference 4 for C_N and C_m for the cases with variable cross-sectional shapes over the length of the body or wing-body combinations.

Subroutine SIMP is a Simpson's Rule integration package used to evaluate the integrals for C_N and C_m when the cross-sectional shape is variable over the length of the body.

Subroutine CEL2 computes the complete elliptic integral of the second kind, E. This is used in equation (10) of reference 4 to compute a modification factor, λ, defined as the ratio of the lift of the triangular wing alone by linearized theory to that by slender-body theory.

Description of Input

Variable definitions.- The format of the input cards for the CRSFLW program is shown in figure 1. The variable names are shown as well as the card columns in which the value is punched and the format in which it is punched. The remainder of this section consists of a table listing the program input variable along with the appropriate algebraic symbol, and the variable definition. A discussion of the preparation of the input is presented following the table. The algebraic notation used to define the configuration and described in the input table is shown in figure 2. All input length and area quantities are dimensional and should have consistent units. Also shown in figure 2 are sketches of a body alone with the positive sense of the forces and moments illustrated.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
Item 1		
LHEAD		80 columns of alphanumeric information used for identification.
Item 2		
RL	ℓ	Length of the body.
XM	x_m	Position about which the pitching moments are to be taken.
AB	A_b	Base area.
AR	A_r	Reference area.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
AP	A_p	Planform area.
V	V	Volume of body.
XC	x_c	x-coordinate of body centroid.
XX	X	Reference length.

Item 3

NSHP		Body shape index: NSHP = 1 Circular body. = 2 Elliptical body with major axis horizontal. = 3 Elliptical body with major axis vertical. = 4 Square body with rounded corners.
NVAR		NVAR = 1 Body shape similar over entire length. = 2 Body shape varies with length. (This option used only when NSHP = 2 or 3.)
NWING		NWING = 0 No wing present. $\neq$ 0 Wing present.
NOSE		NOSE = 1 Conical nose. = 2 Ogive nose. = 3 Newtonian minimum drag nose. = 4 Flat nose.

NOTE: If NSHP = 4, then NOSE = 4 and NWING = 0 is the only acceptable combination of variables.

Item 4

NXS		Number of x-stations along the body where body shape information is input. $2 \leq$ NXS ≤ 50

Item 5

X(I),I=1,NXS	x	Values of x at each station at which body information is to be input. (If there are more than 8, continue on following cards.) X(1) = 0.0 and X(NXS) = RL.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
Item 6		Item 6 consists of parts (a), (b), or (c) depending on body shape.
(a) Circular Body (NSHP = 1)		
R(I),I=1,NXS	r(x)	The radius of the body at each x-station.
	--- or ---	
(b) Elliptical Body (NSHP = 2 or 3)		
(1) A(I),I=1, NXS	a(x)	The value of the semimajor axis at x-station; use as many cards as necessary, 8 values per card.
(2) B(I),I=1, NXS	b(x)	The value of the semiminor axis at each x-station.
	--- or ---	
(c) Square Body (NSHP = 4)		
(1) RK	k	Ratio of the corner radius to the body width. Only one value allowed for entire body. ($0 \leq k \leq 0.5$)
(2) SSQ(I), I=1,NXS	w	Length of square side at each x-station.
Item 7		Item 7 consists of one card plus parts (a) or (b) depending on type of body. Omit Item 7 for blunt-nosed body. (NOSE = 4)
SNOSE	ℓ_s	Length of nose.
(a) Circular Body (NSHP = 1)		
RNOSE	r	Radius of body at base of nose.
	--- or ---	
(b) Elliptical Body (NSHP = 2 or 3)		
ANOSE	a	Semimajor axis at base of nose.
BNOSE	b	Semiminor axis at base of nose.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
Item 8		Omit Item 8 if no wing is present. (NWING = 0)
XLE	x_{LE}	x-coordinate of the wing leading edge at the wing-body juncture.
XLET	$x_{LE_{tip}}$	x-coordinate of leading edge of wing tip. $x_{LE_{tip}} \geq x_{LE}$
XTET	$x_{TE_{tip}}$	x-coordinate of trailing edge of wing tip. $x_{TE_{tip}} \geq x_{LE_{tip}}$
XTE	x_{TE}	x-coordinate of wing trailing edge at the wing-body juncture. $x_{TE} \geq x_{TE_{tip}}$
SSPAN	s	Wing semispan.
EPS	ϵ	Wing planform semiapex angle, radians.
Item 9		Item 9 consists of (a) or (b) depending on body shape. Omit if no wing is present. (NWING = 0)
(a) Circular Body (NSHP = 1)		
RLE	r	Radius of the body at the juncture of the wing leading edge and the body.
	--- or ---	
(b) Elliptical Body (NSHP = 2 or 3)		
ALE	a	Semimajor axis at juncture of wing leading edge and body.
BLE	b	Semiminor axis at juncture of wing leading edge and body.
Item 10		
ALFI	α_i	Initial value of angle of attack, degrees.
ALFF	α_f	Final value of angle of attack, degrees.
DALF	$\Delta\alpha$	Increment in angle of attack.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
CASF	$C_{A_{SF}}$	Axial-force coefficient due to skin friction.
GAM	γ	Ratio of specific heats, typically 1.4 for air.
RE	Re	Reynolds number based on body diameter and free-stream properties.
FMACH	M_∞	Mach number.

NOTE: The angle-of-attack range for bodies alone is $0 \leq \alpha \leq 180^\circ$.
The angle-of-attack range for wing-body combinations is $0 \leq \alpha \leq 90^\circ$.

Input preparation.- A discussion of the input variables is presented in this section as an aid in the preparation of the data deck.

Item number 1 is a single card containing any information which the user wishes to use to identify the run. This information is reproduced in the output exactly as it is punched on this card.

Item 2 is geometry and reference information and all the variables are explained in the previous table or in figure 2.

Item 3 contains four indices controlling the type of body to be considered. The first, NSHP, specifies the body cross-sectional shape to be circular (NSHP = 1), elliptical (NSHP = 2 or 3), or square with rounded corners (NSHP = 4). The second index, NVAR, specifies whether the body shape is similar (NVAR = 1) or not (NVAR = 2) over the length. This index is concerned only with elliptical cross sections (NSHP = 2 or 3). The third index, NWING, specifies whether a wing is (NWING > 0) or is not (NWING = 0) present. The last index, NOSE, specifies the nose shape to be a cone (NOSE = 1), an ogive (NOSE = 2), a Newtonian minimum drag shape (NOSE = 3), or a flat nose (NOSE = 4).

Item 4 is a single index, NXS, specifying the number of stations along the body at which shape information is to be input. This number must be equal to or less than 50.

Item 5 contains the x-stations at which the body information is to be input. The first value should be the tip of the nose and the last should be the base of the body. There are eight values per card in increasing order of x distance up to the total of NXS stations.

Item 6 contains the variables defining the body shape at the x-stations in Item 5. For a circular cross section (NSHP = 1), Item 6 is the radius of each station. For an elliptic station, (NSHP = 2 or 3), Item 6 is made up of two cards. Part (1) contains lengths of the semimajor axis at the x-stations, and part (2) contains the lengths of the semiminor axis at the same stations. If the body is square with rounded corners, the first part of Item 6 is a single card containing the ratio of the corner radius to the body width. Part (2) of Item 6 is the length of the side at each x-station.

Item 7 is made up of two cards, the first of which contains the length of the nose, SNOSE. If the body cross section is circular, (NSHP = 1), the second card of Item 7 contains the radius of the nose at the base, (RNOSE). If the body cross section is elliptic (NSHP = 2 or 3), the second card of Item 7 contains the length of the semimajor axis, ANOSE, and the length of the semiminor axis, BNOSE. Item 7 is omitted if NOSE = 4.

Item 8 contains the variables describing the geometry of the wing. If no wing is present (NWING = 0), Item 8 is omitted. The six variables in Item 8 are self explanatory.

The variables in Item 9 are a function of the body shape in the vicinity of the wing. If no wing is present, Item 9 is omitted. For a circular body (NSHP = 1), Item 9 contains the radius of the body (RLE) at the juncture of the wing leading edge and the body. For an elliptic body (NSHP = 2 or 3), Item 9 contains the lengths of the semimajor and semiminor axes, ALE and BLE, respectively, at the juncture of the wing leading edge and the body.

Item 10 is the last card making up a particular run. The first three variables are the initial angle of attack, ALFI, the final angle of attack, ALFF, and the increment in angle of attack DALF. The next quantity is the axial-force coefficient due to skin friction, CASF. GAM is the ratio of specific heats. RE is the free-stream Reynolds number based on body diameter and FMACH is the free-stream Mach number. The angle-of-attack range for bodies alone is 0° to 180° and that for wing-body combinations is 0° to 90°.

Input decks may be stacked for multiple runs. A second case starting with Item 1 can be placed directly after Item 10.

Sample cases.- Listings of the input decks for five sample cases are presented in figure 3 and sketches of the configurations are shown in figure 4. Sample cases 1 and 2 are body-alone configurations taken from figure 9 of reference 3.

Sample case 1 is body number 2 of that reference and is a flat-nosed cylinder. Sample case 2 is body number 5 of the same reference and is the same cylindrical body with a conical nose attached. Sample cases 3, 4, and 5 are wing-body configurations taken from figure 6 of reference 4. These latter configurations all have the same body length, base area, and aspect ratio.

Description of Output

The output produced by the CRSFLW computer program for sample case 4 is shown in figure 5. The first page of output is a summary of input quantities. Various notes are printed describing the specified components of the configuration. The last items printed on this first page are the flow conditions: Re, γ and M_∞.

The next page is headed by the identification information on the first card of the input deck. Following this are the calculated results printed on two lines. As indicated by the heading, the first line contains α, C_N, C_A, C_m, x_{ac}, C_L, C_D, and L/D. The second line contains components of C_N, C_A, and C_m. CN1 and CM1 are the potential portions of the normal force and pitching moments, and CN2 and CM2 are the viscous crossflow portions of the lift and pitching moments. The axial-force components are defined in the following table. The last variable is the crossflow drag coefficient, C_{d_c}. The output variables are defined as follows.

OUTPUT NOTATION	ALBEGRAIC NOTATION
ALPHA	α, degrees
CN	$C_N = \frac{N}{qA_r}$
CA	$C_A = \frac{A}{qA_r}$
CM	$C_m = \frac{M}{qA_rX}$

OUTPUT NOTATION	ALGEBRAIC NOTATION
XAC	$x_{ac} = \left(\frac{x_m}{X} - \frac{C_m}{C_N}\right) X$
CL	$C_L = C_N \cos\alpha - C_A \sin\alpha$
CD	$C_D = C_N \sin\alpha + C_A \cos\alpha$
L/D	C_L/C_D
CDC	C_{d_n}

I. For nonvarying cross-sectional shape and no wing present.

OUTPUT NOTATION	ALGEBRAIC NOTATION
CN1	$\frac{A_b}{A_r} \sin 2\alpha \cos\frac{\alpha}{2} \left(\frac{C_n}{C_{n_o}}\right)_{SB}$
CN2	$\eta C_{d_n} \sin^2\alpha \left(\frac{C_n}{C_{n_o}}\right)_{Newt}$
CM1	$\left\{\left[\frac{V - A_b(\ell - x_m)}{A_r x}\right] \sin 2\alpha \cos\frac{\alpha}{2}\right\}\left(\frac{C_m}{C_{m_o}}\right)_{SB}$
CM2	$\left[\eta C_{d_n} \frac{A_p}{A_r}\left(\frac{x_m - x_c}{x}\right) \sin^2\alpha\right]\left(\frac{C_m}{C_{m_o}}\right)_{Newt}$
CA1	C_{A_W}
CA2	$C_{A_{SF}}$
CA3	C_{A_B}

II. For a varying cross-sectional shape or a body with lifting surfaces.

OUTPUT NOTATION	ALGEBRAIC NOTATION
CN1	$\left(\frac{A_b}{A_r} \sin 2\alpha \cos \frac{\alpha}{2}\right) \frac{1}{\ell} \int_0^{\ell} \lambda \left(\frac{C_n}{C_{n_o}}\right)_{SB} dx$
CN2	$\frac{2\eta C_{d_n} \sin^2\alpha}{A_r} \int_0^{\ell} \left(\frac{C_n}{C_{n_o}}\right)_{Newt} r \, dx$
CM1	$\left(\frac{A_b}{A_r} \sin 2\alpha \cos \frac{\alpha}{2}\right) \frac{1}{\ell X} \int_0^{\ell} \lambda \left(\frac{C_n}{C_{n_o}}\right)_{SB} (x_m - x) \, dx$
CM2	$\frac{2\eta C_{d_n} \sin^2\alpha}{A_r X} \int_0^{\ell} \left(\frac{C_n}{C_{n_o}}\right)_{Newt} r (x_m - x) \, dx$

Program Listing

The CRSFLW computer program consists of the main program, eight function subprograms and two subroutine subprograms. Each source deck is identified in columns 73 through 80 by a four-character identification and a three-digit number sequencing the cards within that deck. The program listing is given on the following pages. The table below will act as a table of contents for the listing.

```
C     PROGRAM CHSFLW  -  CROSSFLOW PREDICTION METHOD
C
      COMMON RL,XM,NSHP,NVAR,NWING,NOSE,NXS,X(50),A(50),B(50),R(50),
     1 SSQ(50),RK,SNOSE,RNOSE,XLE,XLET,XTET,XTE,SSPAN,ALE,BLE,RLE
      COMMON /INDEX/INT
      COMMON /FLOW/FMACH,GAM,ANOSE,BNOSE
      COMMON /FP/RLAM
      EXTERNAL FUN
      DIMENSION LHEAD(20)
      DATA PI/3.14159265/,DTOR/.01745/
      DATA NIT/12/,TOL/.001/
    1 FORMAT(8F10.5)
    2 FORMAT(10I5)
  701 FORMAT(1H1,10X,20A4/20X,
     115HBODY INPUT DATA//5X,5HRL = ,F10.5,3X,                 5HXM = ,
     2F10.5/5X,5HAB = ,F10.5,3X,5HAR = ,F10.5,3X,5HAP = ,F10.5/5X,
     3 5H V = ,F10.5,3X,                     5HXC = ,F10.5,3X,5HXR = ,F10.5)
  702 FORMAT(//20X,13HCIRCULAR BODY)
  703 FORMAT(//20X,15HELLIPTICAL BODY)
  704 FORMAT(//20X,16HRECTANGULAR BODY)
  705 FORMAT(/10X,41HWING IS PRESENT WITH LEADING EDGE AT X = ,F10.5,2X,
     125HAND TRAILING EDGE AT X = ,F10.5,2X,12H, EPSILON = ,F10.5)
  706 FORMAT(/10X,15HNO WING PRESENT)
  707 FORMAT(/10X,12HCONICAL NOSE)
  708 FORMAT(/10X,10HOGIVE NOSE)
  709 FORMAT(/10X,28HNEWTONIAN MINIMUM DRAG SHAPE )
  710 FORMAT(/10X,9HFLAT NOSE)
  711 FORMAT(/10X,14HNOSE LENGTH = ,F10.5,3X,24H   RADIUS AFT OF NOSE =
     1,F10.5)
  712 FORMAT(1H1,10X,20A4//)
  713 FORMAT (4X5HALPHA,6X2HCN,8X2HCA,8X2HCM,8X3HXAC,7X2HCL,8X2HCD,
     1          8X3HL/D / 12X3HCN1,7X3HCN2,7X3HCA1,7X3HCA2,7X3HCA3,
     2          7X3HCM1,7X3HCM2,7X3HCDC)
  714 FORMAT (/F10.3,7F10.4 / 10X,8(1PE10.3))
  715 FORMAT(/5X,19HERROR IN INTEGRAL ,I1,8H, ANS = ,E12.5)
  716 FORMAT(//10X,15HFLOW CONDITIONS/5X,18HREYNOLDS NUMBER = ,E12.5,5X,
     18HGAMMA = ,F6.3 ,5X,14HMACH NUMBER = ,F6.3 )
  717 FORMAT(1H+,40X,30HBODY SHAPE SIMILAR OVER LENGTH)
  718 FORMAT(1H+,40X,29HBODY SHAPE VARIES WITH LENGTH)
  719 FORMAT(5X,20HTABULATION OF RADIUS/3X,1HI,11X,4HX(I),11X,4HR(I)/)
  720 FORMAT(1X,I3,5X,F10.5,5X,F10.5)
  721 FORMAT(5X,21HTABULATION OF A AND B/3X,1HI,11X,4HX(I),11X,4HA(I),1
     1X,4HB(I)/)
  722 FORMAT(1X,I3,5X,F10.3,5X,F10.3,5X,F10.3)
  723 FORMAT(5X,27HTABULATION OF SQUARE-SIDE,S/5X,16HCORNER RADIUS = ,F1
     10.3,4H * S //3X,1HI,11X,4HX(I),11X,4HS(I))
  724 FORMAT(20A4)
  725 FORMAT(1X,20A4)
C
C     READ AND WRITE HEADING FOR THIS CASE
C
  999 READ(5,724) LHEAD
      EPS=0.0
C     READ AND WRITE INPUT DATA FOR BODY
C
      READ(5,1) RL,XM,AB,AR,AP,V,XC,XX
      WRITE(6,701) LHEAD,RL,XM,AB,AR,AP,V,XC,XX
C        RL   =   BODY LENGTH
C        XM   =   MOMENT CENTER
C        AB   =   A-SUB-B
C        AR   =   A-SUB-R
C        AP   =   A-SUB-P
C        V    =   V
C        XC   =   X-SUB-C
C         XX = X
C
      READ(5,2)  NSHP,NVAR,NWING,NOSE
C
C        NSHP = 1  --  CIRCULAR BODY
C        NSHP = 2  --  ELLIPTICAL WITH HORIZONTAL AXIS LONGER
C        NSHP = 3  --  ELLIPTICAL WITH VERTICAL AXIS LONGER
C        NSHP = 4  --  SQUARE WITH ROUNDED CORNERS
C
C         NVAR = 1  --  SIMILAR SHAPE OVER LENGTH
C        NVAR = 2  --  ECCENTRICITY OF ELLIPSE VARIES WITH X
C
C       NWING = 0 -- NO WING PRESENT
C       NWING = 1 -- WING PRESENT
C
C
C        NOSE = 1 -- CONICAL NOSE
C        NOSE = 2 -- OGIVE NOSE
C        NOSE = 3 -- NEWTONIAN MINIMUM DRAG SHAPE
C        NOSE = 4 -- FLAT NOSE
C
      GO TO (51,52,52,54),NSHP
   51 WRITE(6,702)
      WRITE(6,717)
      READ(5,2) NXS
      READ(5,1) (X(I),I=1,NXS)
      READ(5,1) (R(I),I=1,NXS)
      WRITE(6,719)
      WRITE(6,720) (I,X(I),R(I),I=1,NXS)
      GO TO 60
   52 READ(5,2)  NXS
      READ(5,1) (X(I),I=1,NXS)
      READ(5,1) (A(I),I=1,NXS)
      READ(5,1) (B(I),I=1,NXS)
      WRITE(6,703)
      GO TO (521,522), NVAR
  521 WRITE(6,717)
      GO TO 523
  522 WRITE(6,718)
  523 WRITE(6,721)
      WRITE(6,722) (I,X(I),A(I),B(I),I=1,NXS)
      GO TO 60
   54 READ(5,2) NXS
      READ(5,1) RK
      READ(5,1) (X(I),I=1,NXS)
      READ(5,1) (SSQ(I),I=1,NXS)
      WRITE(6,704)
      GO TO (541,542), NVAR
  541 WRITE(6,717)
      GO TO 543
  542 WRITE(6,718)
  543 WRITE(6,723) RK
      WRITE(6,720) (I,X(I),SSQ(I),I=1,NXS)
C
C     NOSE INPUT DATA
C
C     SNOSE = LENGTH OF NOSE
C     RNOSE = RADIUS AT AFT END OF NOSE SECTION
C
   60 IF(NOSE.EQ.4) GO TO 64
      READ(5,1) SNOSE
      GO TO (602,601,601,603), NSHP
  601 READ(5,1) ANOSE,BNOSE
      RNOSE=SQRT(ANOSE*BNOSE)
      GO TO (61,62,63,64) , NOSE
  602 READ(5,1) RNOSE
  603 GO TO (61,62,63,64), NOSE
   61 WRITE(6,707)
      GO TO 65
   62 WRITE(6,708)
      GO TO 65
   63 WRITE(6,709)
      GO TO 65
   64 WRITE(6,710)
      GO TO 66
   65 WRITE(6,711) SNOSE,RNOSE
```

```
   66 IF(NWING) 67,70,67                                                CF01 141
C                                                                       CF01 142
C     IF WING PRESENT, READ AND WRITE WING DATA                         CF01 143
C                                                                       CF01 144
C     XLE = LEADING EDGE X-COORDINATE                                   CF01 145
C     XTE = TRAILING EDGE X-COORDINATE                                  CF01 146
C     XLET = COORDINATE OF LEADING EDGE OF WINGTIP                      CF01 147
C     XTET = COORDINATE OF TRAILING EDGE OF WINGTIP                     CF01 148
C     SSPAN = SEMISPAN OF WING                                          CF01 149
C       EPS = WING SEMIAPEX ANGLE, RADIANS                              CF01 150
C                                                                       CF01 151
   67 READ(5,1) XLE,XLET,XTET,XTE,SSPAN,EPS                             CF01 152
      IF(XLE.GE.XLET) XLET=XLE+ .01*(XTE-XLE)                           CF01A152
      IF(XTET.LT.XLET) XTET=XLET                                        CF01B152
      IF(XTET.GT.XTE) XTE=XTET                                          CF01C152
      IF (XTET.GE.XTE .AND. RL.GT.XTE) XTE=XTE+0.01*(XTE-XLE)           CF01D152
      GO TO (68,69,69), NSHP                                            CF01 153
   68 READ(5,1) RLE                                                     CF01 154
      GO TO 71                                                          CF01 155
   69 READ(5,1) ALE,BLE                                                 CF01 156
   71 WRITE(6,705) XLE,XTE,EPS                                          CF01 157
      GO TO 72                                                          CF01 158
C                                                                       CF01 159
C     READ AND WRITE FLOW CONDITIONS                                    CF01 160
C                                                                       CF01 161
C     FMACH = M-INFINITY                                                CF01 162
C     RE    = FREE-STREAM REYNOLDS NUMBER BASED ON BODY DIAMETER        CF01 163
C     GAM   = GAMMA                                                     CF01 164
C     ALFI  = INITIAL ALPHA                                             CF01 165
C     ALFF = FINAL ALPHA                                                CF01 166
C     DALF = INCREMENT TO BE USED FOR ALPHA                             CF01 167
C     CASF  = SKIN FRICTION COEFFICIENT                                 CF01 168
C                                                                       CF01 169
C                                                                       CF01 170
   70 WRITE(6,706)                                                      CF01 171
      SSPAN=0.0                                                         CF01A171
      ALE=0.0                                                           CF01B171
      BLE=0.0                                                           CF01C171
   72 READ(5,1) ALFI,ALFF,DALF,CASF,GAM,RE,FMACH                        CF01 172
      WRITE(6,716) RE,GAM,FMACH                                         CF01 173
      IF(NWING.NE.0) GO TO 2000                                         CF01 174
      GO TO (1000,2000),NVAR                                            CF01 175
C                                                                       CF01 176
C     EQUATIONS FOR BODY WITH SIMILAR CROSS-SECTION AND NO WING         CF01 177
C                                                                       CF01 178
 1000 CNRSBN=CNRSB(NSHP)                                                CF01 179
      CMRSBN=CNRSBN                                                     CF01 180
      CNRNTN=CNRNT(NSHP)                                                CF01 181
      CMRNTN=CNRNTN                                                     CF01 182
      WRITE (6,712) LHEAD                                               CF01 183
      WRITE (6,713)                                                     CF01 184
      ALF=ALFI                                                          CF01 185
      IF(ALF-90.) 1010,1010,1020                                        CF01 186
C                                                                       CF01 187
C     EQUATIONS OF 0.LE.ALF.LE.90                                       CF01 188
C                                                                       CF01 189
 1010 ISA=1                                                             CF01 190
      CA1=CAN(NOSE,RNOSE,SNOSE)                                         CF01 191
      CA2=CASF                                                          CF01 192
      CA3=(-2./(GAM*FMACH*FMACH))*((2./(GAM+1.))**1.4*(1./FMACH)**2.5   CF01 193
     1 *(2.*GAM*FMACH*FMACH-(GAM-1.))/(GAM+1.)-1.)                      CF01 194
      IF (FMACH.LE.1.0) CA3=0.0                                         CF01A194
      CA0=CA1+CA2+CA3                                                   CF01 195
      ETAC=ETA(FMACH)                                                   CF01 196
 1012 ALFR=ABS(ALF)*DTOR                                                CF01 197
      SINAL=SIN(ALFR)                                                   CF01 198
      COSAL=COS(ALFR)                                                   CF01 199
      SIN2A=SIN(2.*ALFR)                                                CF01 200
      COSA2=COS(ALFR/2.)                                                CF01 201
      FMACHN=FMACH*SINAL                                                CF01 202
      REN=RE*SINAL                                                      CF01 203
      CONC=CON(FMACHN,REN)                                              CF01 204
      CN1=(AB/AR)*SIN2A*COSA2*CNRSBN                                    CF01 205
      CN2=ETAC*CONC*(AP/AR)*SINAL*SINAL*CNRNTN                          CF01 206
      CN=CN1+CN2                                                        CF01 207
      CM1=( (V-AB*(RL-XM))/(AR*XX) )*SIN2A*COSA2*CMRSBN                 CF01 208
      CM2=ETAC*CONC*(AP/AR)*((XM-XC)/XX)*SINAL*SINAL*CMRNTN             CF01 209
      CM=CM1+CM2                                                        CF01 210
      CA=CA0*COSAL*COSAL                                                CF01 211
      IF (ALF.GT.0.0) GO TO 1013                                        CF01 212
      CM=-CM                                                            CF01 213
      CN=-CN                                                            CF01 214
 1023 IF (ALF.NE.0.0 .AND. ALF.NE.180.) GO TO 1013                      CF01 215
      XAC=0.0                                                           CF01 216
      GO TO 1025                                                        CF01 217
 1013 XAC=(XM/XX-CM/CN)*XX                                              CF01 218
 1025 ALFR=ALF*DTOR                                                     CF01 219
      COSAL=COS(ALFR)                                                   CF01 220
      SINAL=SIN(ALFR)                                                   CF01 221
      CL=CN*COSAL - CA*SINAL                                            CF01 222
      CD=CN*SINAL + CA*COSAL                                            CF01 223
      CLD=CL/CD                                                         CF01 224
      WRITE (6,714) ALF,CN,CA,CM,XAC,CL,CD,CLD,CN1,CN2,CA1,CA2,CA3,     CF01 225
     1                   CM1,CM2,CONC                                   CF01 226
      ALF=ALF+DALF                                                      CF01 227
      IF(ALF-ALFF) 1014,1014,999                                        CF01 228
 1014 IF(ALF.GT.90..AND.ISA.EQ.1) ISA=2                                 CF01 229
      GO TO (1010,1020,1022),ISA                                        CF01 230
C                                                                       CF01 231
C     EQUATIONS FOR ALF.GT.90                                           CF01 232
C                                                                       CF01 233
 1020 CA1=CAN(4,0.,0.)                                                  CF01 234
      CA0=-(CA1+CA2+CA3)                                                CF01 235
      ISA=3                                                             CF01 236
 1022 ALFP=180.-ALF                                                     CF01 237
      ALFR=ALFP*DTOR                                                    CF01 238
      SINAL=SIN(ALFR)                                                   CF01 239
      COSAL=COS(ALFR)                                                   CF01 240
      SIN2A=SIN(2.*ALFR)                                                CF01 241
      COSA2=COS(ALFR/2.)                                                CF01 242
      FMACHN=FMACH*SINAL                                                CF01 243
      REN=RE*SINAL                                                      CF01 244
      CONC=CON(FMACHN,REN)                                              CF01 245
      CA=CA0*COSAL*COSAL                                                CF01 246
      CN1=(AB/AR)*SIN2A*COSA2*CNRSBN                                    CF01 247
      CN2=ETAC*CONC*(AP/AR)*SINAL*SINAL*CNRNTN                          CF01 248
      CN=CN1+CN2                                                        CF01 249
      CM1=-(V-AB*XM)*SIN2A*COSA2*CMRSBN/(AR*XX)                         CF01 250
      CM2=ETAC*CONC*AP*(XM-XC)*SINAL*SINAL*CMRNTN/(AR*XX)               CF01 251
      CM=CM1+CM2                                                        CF01 252
      GO TO 1023                                                        CF01 253
C                                                                       CF01 254
C     EQUATIONS FOR NON-SIMILAR CROSS SECTION OR WING PRESENT, OR BOTH  CF01 255
C                                                                       CF01 256
 2000 WRITE(6,712)  LHEAD                                               CF01 257
      WRITE (6,713)                                                     CF01 258
      IF(NWING.EQ.0) GO TO 2100                                         CF01A258
      IF (ALFF.GT.90.0) ALFF=90.0                                       CF01 259
 2100 CONTINUE                                                          CF01A259
      IF(FMACH-1.)2001,2002,2002                                        CF01 260
 2001 BETA=SQRT(1.-FMACH*FMACH)                                         CF01 261
      GO TO 2003                                                        CF01 262
 2002 BETA=SQRT(FMACH*FMACH-1.)                                         CF01 263
 2003 IF(BETA*TAN(EPS)-1.) 2006,2006,2004                               CF01 264
 2006 AK=SQRT(1.-BETA*BETA*TAN(EPS)*TAN(EPS))                           CF01 265
      AA=1.0                                                            CF01 266
      BB=1.0-AK*AK                                                      CF01 267
      RLAM=1.0                                                          CF01A267
      IF(NWING.GT.0) CALL CEL2(RLAM,AK,AA,BB,IDUM)                      CF01 268
      RLAM=1./RLAM                                                      CF01 269
      GO TO 2005                                                        CF01 270
 2004 RLAM=2./(PI*TAN(EPS)*BETA)                                        CF01 271
 2005 INT=1                                                             CF01 272
      CALL SIMP(0.,RL,NIT,IER,TOL,FUN,ANS1)                             CF01 273
      IF(IER.EQ.0) WRITE(6,715) INT,ANS1                                CF01 274
      INT=2                                                             CF01 275
      CALL SIMP(0.,RL,NIT,IER,TOL,FUN,ANS2)                             CF01 276
      IF(IER.EQ.0) WRITE(6,715) INT,ANS2                                CF01 277
      INT=3                                                             CF01 278
      CALL SIMP(0.,RL,NIT,IER,TOL,FUN,ANS3)                             CF01 279
      IF(IER.EQ.0) WRITE(6,715) INT,ANS3                                CF01 280
```

```
      INT=4
      CALL SIMP(0.,RL,NIT,IER,TOL,FUN,ANS4)
      IF(IER.EQ.0) WRITE(6,715) INT,ANS4
      ISA=1
      ALF=ALFI
      CAB=(-2./(GAM*FMACH*FMACH))*((2./(GAM+1.))**1.4*(1./FMACH)**2.8
     1*(2.*GAM*FMACH*FMACH-GAM+1.)/(GAM+1.)-1.)
      IF (FMACH.LE.1.0) CAB=0.0
      CA1=CAN(NOSE,RNOSE,SNOSE)
      CA0=CA1 + CAB + CASF
      ETAC=ETA(FMACH)
 2010 ALFR=ABS(ALF)*DTOR
 2023 SINAL=SIN(ALFR)
      COSAL=COS(ALFR)
      COSA2=COS(ALFR/2.)
      SIN2A=SIN(ALFR*2.)
      CN1=(AB/AR)*SIN2A*COSA2*ANS1/RL
      FMACHN=FMACH*SINAL
      RENRE=SINAL
      CDNC=CDN(FMACHN,REN)
      CN2=2.*ETAC*CDNC*SINAL*SINAL*ANS2/AR
      CN=CN1+CN2
      CM1=(AB/AR)*SIN2A*COSA2*ANS3/(RL*XX)
      CM2=2.*ETAC*CDNC*SINAL*SINAL*ANS4/(AR*XX)
      CM=CM1+CM2
      CA= CA0*COSAL*COSAL
      IF (ALF.GT.0.0) GO TO 2015
      CN=-CN
      CM=-CM
 2015 XAC=0.0
      IF(ALF.NE.0.0.AND.ALF.NE.180.0) XAC=XM-CM*XX/CN
C
C
 2016 ALFR=ALF*DTOR
      COSAL=COS(ALFR)
      SINAL=SIN(ALFR)
      CL=CN*COSAL - CA*SINAL
      CD=CN*SINAL + CA*COSAL
      CLD=CL/CD
      WRITE (6,714) ALF,CN,CA,CM,XAC,CL,CD,CLD,CN1,CN2,CA1,CASF,CAB,
     1              CM1,CM2,CDNC
      ALF=ALF+DALF
      IF(ALF-ALFF) 2011,2011,999
 2011 IF(ALF.GT.90.0.AND.ISA.EQ.1) ISA=2
      GO TO (2010,2020,2022), ISA
C
C       EQUATIONS FOR ALF.GT.90
 2020 CA1=CAN(4,0.,0.)
      CA0=- (CA1+CAB+CASF)
      ISA=3
 2022 ALFP=180.-ALF
      ALFR=ALFP*DTOR
      GO TO 2023
      END


        FUNCTION CNSB(XI)
C
C     THIS ROUTINE COMPUTES INTEGRAND VALUES OF (CN/CN0)SB FOR
C     CROSS-SECTIONAL SHAPES WHICH VARY WITH LENGTH, OR BODIES
C     WITH WING ATTACHED
C
      COMMON RL,XM,NSHP,NVAR,NWING,NOSE,NXS,X(50),A(50),B(50),R(50),
     1 SSO(50),RK,SNOSE,RNOSE,XLE,XLET,XTET,XTE,SSPAN,ALE,BLE,RLE
      COMMON /FSB/RR
      REAL INTRP
      INTRP(XL,XU,YL,YU,X)=YL+(X-XL)*(YU-YL)/(XU-XL)
C
C     CHECK TO SEE IF WE ARE AT THE POINT OF A POINTED NOSE
C
      XX=XI
      IF(XI.EQ.0..AND.NOSE.NE.4) XX=RL*1.E-5
C
C
C
C     COMPUTE WHERE XX IS ALONG THE WING, IF PRESENT, AND ALONG
C     THE BODY
C
      IF(NWING) 100,101,100
  101 KW=0
      GO TO 125
  100 IF(XX-XTE) 102,101,101
  102 IF(XX-XTET) 103,104,105
  103 IF(XX-XLET) 106,104,104
  106 IF(XX-XLE) 101,101,107
  104 KW=2
      GO TO 125
  105 KW=3
      GO TO 125
  107 KW=1
  125 DO 128 J=1,NXS
      IF(XX-X(J)) 131,130,128
  128 CONTINUE
  130 KX=-J
      GO TO 150
  131 KX=J
  150 GO TO (1100,1200,1300), NSHP
C
C     CIRCULAR BODY WITH WING PRESENT
C
 1100 IF(KX) 1112,1112,1114
 1112 KX=-KX
      RR=R(KX)
      GO TO 1120
 1114 KXM=KX-1
      RR=INTRP(X(KXM),X(KX),R(KXM),R(KX),XX)
 1120 SSPI=SSPAN-RLE
      IF(KW.NE.0) GO TO (1102,1104,1106), KW
      CNSB=1.
      RETURN
 1102 S=INTRP(XLE,XLET,0.,SSPI,XX) + RLE
      GO TO 1110
 1104 S=SSPAN
      GO TO 1110
 1106 S=INTRP(XTET,XTE,SSPI,0.,XX) + RLE
 1110 CNSB=S*S/(RR*RR) + RR*RR/(S*S) - 1.
      RETURN
C
C     ELLIPTICAL BODY WITH MAJOR AXIS HORIZONTAL
C     WITH OR WITHOUT WING
C
 1200 IF(KW.EQ.0) GO TO 1210
      SSPI=SSPAN-ALE
      GO TO (1202,1204,1206),KW
 1202 S=INTRP(XLE,XLET,0.,SSPI,XX) + ALE
      GO TO 1210
 1204 S=SSPAN
      GO TO 1210
 1206 S=INTRP(XTET,XTE,SSPI,0.,XX) + ALE
 1210 IF(KX) 1212,1212,1214
 1212 KX=-KX
      AA=A(KX)
      BB=B(KX)
      GO TO 1225
C
 1214 KXM=KX-1
      AA=INTRP(X(KXM),X(KX),A(KXM),A(KX),XX)
      BB=INTRP(X(KXM),X(KX),B(KXM),B(KX),XX)
 1225 RR=SQRT(AA*BB)
      IF(AA-BB) 1226,1227,1228
 1227 IF(KW) 1229,1229,1110
 1229 CNSB=1.0
      RETURN
 1226 T=AA
      AA=BB
      BB=T
 1228 CONTINUE
      IF(KW) 1220,1222,1220
 1220 SIG=0.5*(S+SQRT(S*S+BB*BB-AA*AA))
      CH=SIG+(AA+BB)*(AA+BB)/(4.*SIG)
      CNSB=(CH*CH+AA*AA)/(AA*BB)
      RETURN
 1222 CNSB=AA/BB
      RETURN
```

```
C                                                                         CF02 091
C     ELLIPTICAL BODY WITH MAJOR AXIS VERTICAL                            CF02 092
C     WITH OR WITHOUT WING ATTACHED                                       CF02 093
C                                                                         CF02 094
 1300 IF(KW.EQ.0) GO TO 1310                                              CF02 095
      SSPI=SSPAN-BLE                                                      CF02 096
      GO TO (1302,1304,1306),KW                                           CF02 097
 1302 S=INTRP(XLE,XLET,0.,SSPI,XX) + BLE                                  CF02 098
      GO TO 1310                                                          CF02 099
 1304 S=SSPAN                                                             CF02 100
      GO TO 1310                                                          CF02 101
 1306 S=INTRP(XTET,XTE,SSPI,0.,XX) + BLE                                  CF02 102
 1310 IF(KX) 1312,1312,1314                                               CF02 103
 1312 KX=-KX                                                              CF02 104
      AA=A(KX)                                                            CF02 105
      BB=B(KX)                                                            CF02 106
      GO TO 1325                                                          CF02 107
C                                                                         CF02 108
 1314 KXM=KX-1                                                            CF02 109
      AA=INTRP(X(KXM),X(KX),A(KXM),A(KX),XX)                              CF02 110
      BB=INTRP(X(KXM),X(KX),B(KXM),B(KX),XX)                              CF02 111
 1325 RR=SQRT(AA*BB)                                                      CF02 112
      IF(AA-BB) 1326,1327,1328                                            CF02A112
 1327 IF(KW) 1229,1229,1110                                               CF02B112
 1326 T=AA                                                                CF02C112
      AA=BB                                                               CF02D112
      BB=T                                                                CF02E112
      GO TO 1228                                                          CF02F112
 1328 CONTINUE                                                            CF02G112
      IF(KW) 1320,1322,1320                                               CF02 113
 1320 SIG=0.5*(S+SQRT(S*S+AA*AA-BB*BB))                                   CF02 114
      CH=SIG-(AA+BB)*(AA+BB)/(4.*SIG)                                     CF02 115
      CNSB=(CH*CH+BB*BB)/(AA*BB)                                          CF02 116
      RETURN                                                              CF02 117
 1322 CNSB=BB/AA                                                          CF02 118
      RETURN                                                              CF02 119
      END                                                                 CF02 120
```

```
      FUNCTION CNNT(XI)                                                   CF03 001
C                                                                         CF03 002
C     THIS SUBROUTINE CALCULATES THE VALUE OF (CN/CNO)NENT FOR BODIES WICF03 003
C     VARYING CROSS-SECTION OR WITH A WING                                CF03 004
C                                                                         CF03 005
      REAL INTRP                                                          CF03 006
      COMMON RL,XM,NSHP,NVAR,NWING,NOSE,NXS,X(50),A(50),B(50),R(50),      CF03 007
     1 SSO(50),NK,SNOSE,RNOSE,XLE,XLET,XTET,XTE,SSPAN,ALE,BLE,RLE         CF03 008
      COMMON /FSB/RR                                                      CF03 009
      INTRP(XL,XU,YL,YU,X)=YL+(X-XL)*(YU-YL)/(XU-XL)                      CF03 010
C                                                                         CF03 011
C     CHECK IF WE ARE ON THE POINT OF A NOSE                              CF03 012
C                                                                         CF03 013
      XX=XI                                                               CF03 014
      IF(XI.EQ.0..AND.NOSE.NE.4) XX=RL*1.E-5                              CF03 015
C                                                                         CF03 016
C     FIND LOCATION ALONG BODY AND WING (IF PRESENT)                      CF03 017
C                                                                         CF03 018
      IF(NWING) 2,1,2                                                     CF03 019
    1 KW=0                                                                CF03 020
      GO TO 10                                                            CF03 021
    2 IF(XX-XLE) 1,1,3                                                    CF03 022
    3 IF(XX-XLET) 7,6,4                                                   CF03 023
    4 IF(XX-XTET) 6,6,5                                                   CF03 024
    5 IF(XX-XTE) 8,1,1                                                    CF03 025
    6 KW=2                                                                CF03 026
      GO TO 10                                                            CF03 027
    7 KW=1                                                                CF03 028
      GO TO 10                                                            CF03 029
    8 KW=3                                                                CF03 030
   10 DO 12 I=1,NXS                                                       CF03 031
      IF(XX-X(I)) 14,13,12                                                CF03 032
   12 CONTINUE                                                            CF03 033
   13 KX=-I                                                               CF03 034
      GO TO (100,200,300), NSHP                                           CF03 035
   14 KX=I                                                                CF03 036
      GO TO (100,200,300), NSHP                                           CF03 037
C                                                                         CF03 038
C     CIRCULAR BODY WITH WING PRESENT                                     CF03 039
C                                                                         CF03 040
  100 IF(KX) 112,112,114                                                  CF03 041
  112 KX=-KX                                                              CF03 042
      RR=R(KX)                                                            CF03 043
      GO TO 115                                                           CF03 044
  114 KXM=KX-1                                                            CF03 045
      RR=INTRP(X(KXM),X(KX),R(KXM),R(KX),XX)                              CF03 046
  115 IF(KW.NE.0) GO TO (102,104,106), KW                                 CF03 047
      CNNT=1.                                                             CF03 048
      RETURN                                                              CF03 049
  102 SSPI=SSPAN-RLE                                                      CF03 050
      S=INTRP(XLE,XLET,0.,SSPI,XX) + RLE                                  CF03 051
      GO TO 110                                                           CF03 052
  104 S=SSPAN                                                             CF03 053
      GO TO 110                                                           CF03 054
  106 SSPI=SSPAN-RLE                                                      CF03 055
      S=INTRP(XTET,XTE,SSPI,0.,XX) + RLE                                  CF03 056
  110 CNNT=1.5*(S/RR-.333333333333)                                       CF03 057
      RETURN                                                              CF03 058
C                                                                         CF03 059
C     ELLIPTIC BODY WITH MAJOR AXIS HORIZONTAL                            CF03 060
C                                                                         CF03 061
  200 IF(KX) 201,201,202                                                  CF03 062
  201 KX=-KX                                                              CF03 063
      AA=A(KX)                                                            CF03 064
      BB=B(KX)                                                            CF03 065
      RR=SQRT(AA*BB)                                                      CF03 066
      GO TO 205                                                           CF03 067
  202 KXM=KX-1                                                            CF03 068
      AA=INTRP(X(KXM),X(KX),A(KXM),A(KX),XX)                              CF03 069
      BB=INTRP(X(KXM),X(KX),B(KXM),B(KX),XX)                              CF03 070
      RR=SQRT(AA*BB)                                                      CF03 071
  205 S=AA                                                                CF03 072
      SSPI=SSPAN-ALE                                                      CF03 073
      IF(KW.EQ.0) GO TO 215                                               CF03 074
      GO TO (211,212,213),KW                                              CF03 075
  211 S=INTRP(XLE,XLET,0.,SSPI,XX) + ALE                                  CF03 076
      GO TO 215                                                           CF03 077
  212 S=SSPAN                                                             CF03 078
      GO TO 215                                                           CF03 079
  213 S=INTRP(XTET,XTE,SSPI,0.,XX) + ALE                                  CF03 080
  215 IF (AA-BB) 216,110,218                                              CF03 081
  216 T=AA                                                                CF03A081
      AA=BB                                                               CF03B081
      BB=T                                                                CF03C081
      GO TO 318                                                           CF03D081
  218 AB=AA/BB                                                            CF03E081
      IF(AB.LT.1.001) AB=1.001                                            CF03 082
      BA=1./AB                                                            CF03 083
      RR=SQRT(AA*BB)                                                      CF03 084
      CNNT=1.5*SQRT(AB)*(-BA*BA*ALOG(AB*(1.+SQRT(1.-BA*BA) )  )           CF03 085
     1  /(1.-BA*BA)**1.5 + 1./(1.-BA*BA) + S/AA - 1.  )                   CF03 086
      RETURN                                                              CF03 087
C                                                                         CF03 088
C     ELLIPTIC BODY WITH MAJOR AXIS VERTICAL                              CF03 089
C                                                                         CF03 090
  300 IF(KX) 301,301,302                                                  CF03 091
  301 KX=-KX                                                              CF03 092
      AA=A(KX)                                                            CF03 093
      BB=B(KX)                                                            CF03 094
      RR=SQRT(AA*BB)                                                      CF03 095
      GO TO 305                                                           CF03 096
  302 KXM=KX-1                                                            CF03 097
      AA=INTRP(X(KXM),X(KX),A(KXM),A(KX),XX)                              CF03 098
      BB=INTRP(X(KXM),X(KX),B(KXM),B(KX),XX)                              CF03 099
      RR=SQRT(AA*BB)                                                      CF03 100
  305 S=BB                                                                CF03 101
      SSPI=SSPAN-BLE                                                      CF03 102
      IF(KW.EQ.0) GO TO 315                                               CF03 103
      GO TO (311,312,313),KW                                              CF03 104
  311 S=INTRP(XLE,XLET,0.,SSPI,XX) + BLE                                  CF03 105
      GO TO 315                                                           CF03 106
```

```
  312 S=SSPAN
      GO TO 315
  313 S=INTRP(XTET,XTE,SSPI,0.,XX) + BLE
  315 IF(AA-BB) 316,110,318
  316 T=AA
      AA=BB
      BB=T
      GO TO 218
  318 AB=AA/BB
      IF(AB.LT.1.001) AB=1.001
      BA=1./AB
      DUM=1.0
      CNNT=1.5*SQRT(BA)*(AB*AB*ATAN2(SQRT(AB*AB-1.),DUM)/(AB*AB-1.)**1.5
     1  -1./(AB*AB-1.) + S/BB - 1.  )
      RETURN
      END
```

```
      FUNCTION CNRSB(NSHP)
C
C     THIS ROUTINE CALCULATES (CN/CNO)SB FOR THE CASE OF
C     BODY WITH SIMILAR SHAPE OVER LENGTH AND NO WING ATTACHED
C     THESE VALUES ARE ALSO THOSE FOR (CM/CMO)SB FOR THIS CASE
C
      COMMON RL,XM,NDUM,NVAR,NWING,NOSE,NXS,X(50),A(50),B(50),R(50),
     1 SSG(50),RK,SNOSE,RNOSE,XLE,XLET,XTET,XTE,SSPAN,ALE,BLE,RLE
      GO TO (100,200,300,400), NSHP
  100 CNRSB=1.
      RETURN
  200 CNRSB=A(2)/B(2)
      RETURN
  300 CNRSB=B(2)/A(2)
      RETURN
  400 CNRSB=1.19 - 0.38*RK
      RETURN
      END
```

```
      FUNCTION CNRNT(NSHP)
      COMMON RL,XM,NDUM,NVAR,NWING,NOSE,NXS,X(50),A(50),B(50),R(50),
     1 SSG(50),RK,SNOSE,RNOSE,XLE,XLET,XTET,XTE,SSPAN,ALE,BLE,RLE
      DATA PI/3.14159265/,DUM/1.0/
C
C     THIS ROUTINE CALCULATES (CN/CNO)NEWT FOR BODIES OF SIMILAR CROSS
C     SECTION OVER LENGTH AND WITHOUT WING
C     ALSO USED FOR THE VALUE OF (CM/CMO)NEWT FOR THE SAME CASE
C
      GO TO (1100,1200,1300,1400), NSHP
 1100 CNRNT=1.
      RETURN
 1200 AB=A(2)/B(2)
      BA=1./AB
      CNRNT=1.5*SQRT(AB)*( -BA*BA*ALOG(AB*(1.+SQRT(1.-BA*BA)))
     1 /(1.-BA*BA)**1.5 + 1./(1.-BA*BA)  )
      RETURN
 1300 AB=A(2)/B(2)
      BA=1./AB
      CNRNT=1.5*SQRT(BA)*(AB*AB*ATAN2(SQRT(AB*AB-1.),DUM)/
     1         ((AB*AB-1.0)**1.5) - 1.0/(AB*AB-1.0))
      RETURN
 1400 CNRNT=0.5*(1.5-RK)*SQRT(PI/(1.-(4.-PI)*RK*RK) )
      RETURN
      END
```

```
      FUNCTION CAW(NOSE,RNOSE,SNOSE)
      REAL INTRP
      DIMENSION X1(20),TAB1(20)
      DIMENSION X2(20),TAB2(20)
      COMMON RL,DUM,NSHP
      COMMON /FLOW/FMACH,GAM,ANOSE,BNOSE
C
C        DATA IN TAB1 FROM FIG. 6 NASA TN D-6996
C
C        DATA IN TAB2 FROM FIG. 7 NASA TN D-6996
C
      DATA X1/0.,.3,.4,.5,.6,.7,.8,.9,1.,1.1,1.2,1.3,1.4,1.5,1.6,1.7,
     1  1.8,3*0./,NTAB1/17/
      DATA TAB1/0.,.09,.13,.18,.23,.3,.38,.47,.56,.66,.78,.90,1.04,1.18,
     1 1.32,1.42,1.63,3*0./
      DATA NTAB2/16/,X2/0.,1.,1.5,  2.,2.5,3.,3.5,4.,4.5,5.,5.5,6.,7.,
     1  8.,9.,10.,4*0./
      DATA TAB2/1.3,1.54,1.62,1.66,1.72,1.758,1.779,1.79,1.8,1.808,
     1  1.815,1.819,1.825,1.828,1.83,1.831,4*0./
      INTRP(XL,XU,YL,YU,X) = YL + (X-XL)*(YU-YL)/(XU-XL)
      IF(FMACH.GT.1.) GO TO (100,200,200,40), NSHP
      CAW=0.
      RETURN
  100 GO TO (10,20,30,40), NOSE
   10 SNTHT=RNOSE/SQRT(RNOSE*RNOSE+SNOSE*SNOSE)
      BETA=SQRT(FMACH*FMACH-1.)
C      CONICAL-NOSED BODIES  -  CIRCULAR CROSS SECTION
C        EQN. (10) , NASA TN D-6996
      CAW=4.*SNTHT*SNTHT*(2.5+8.*BETA*SNTHT)/(1.+16.*SNTHT*BETA)
      RETURN
   20 X=2.*RNOSE*FMACH/RL
C      OGIVE NOSE  -  FIGURE 6 , NASA TN D-6996
      DO 21 I=1,NTAB1
      IF(X-X1(I))23,22,21
   21 CONTINUE
   22 WDP=TAB1(I)
      GO TO 24
   23 IM=I-1
      WDP=INTRP(X1(IM),X1(I),TAB1(IM),TAB1(I),X)
   24 CAW=2.*WDP/(GAM*FMACH*FMACH)
      RETURN
   30 X=2.*RNOSE*FMACH/RL
C      NEWTONIAN MINIMUM DRAG NOSE  -  FIGURE 6 , NASA TN D-6996
      CAW=0.442*(X**1.683)*2.0/(GAM*FMACH*FMACH)
      RETURN
   40 CONTINUE
C      FLAT NOSE - CIRCULAR BODY        FIGURE 7 , NASA TN D-6996
      DO 41 I=1,NTAB2
      IF(FMACH-X2(I)) 43,42,41
   41 CONTINUE
   42 CAW=TAB2(I)
      RETURN
   43 IM=I-1
      CAW=INTRP(X2(IM),X2(I),TAB2(IM),TAB2(I),FMACH)
      RETURN
  200 GO TO (210,20 ,30 ,40), NOSE
C
C
  210 C=ANOSE/SNOSE
      D=BNOSE/SNOSE
      BET=SQRT(FMACH*FMACH-1.)
      HLM=ALOG(4./(BET*(C+D)))-1.
C      CONICAL NOSED BODY  -  ELLIPTIC CROSS SECTION
C      EQN. (28)  ,   NASA TN D-6996
      CAW=C*D*(2.*HLM+1.) + BET*BET*C*D*(3.*C*D*HLM*HLM+1.5*(C*C+D*D)
     1  *HLM - 0.5*(C-D)*(C-D) + 0.5*C*D  )
     2  +C*C*D*D*((GAM+1.)*FMACH**4/(BET*BET)-(2.+FMACH*FMACH)*HLM+
     3  (FMACH*FMACH*.25-1.)*(C*C+D*D)/(2.*C*D) )
     4 + FMACH*FMACH*C*C*D*D* (.375*(C*C+D*D)/(C*D)-HLM-1.
     5   + 1.5*C*D*ALOG(C/D)/(C*C-D*D)  )
      RETURN
      END
```

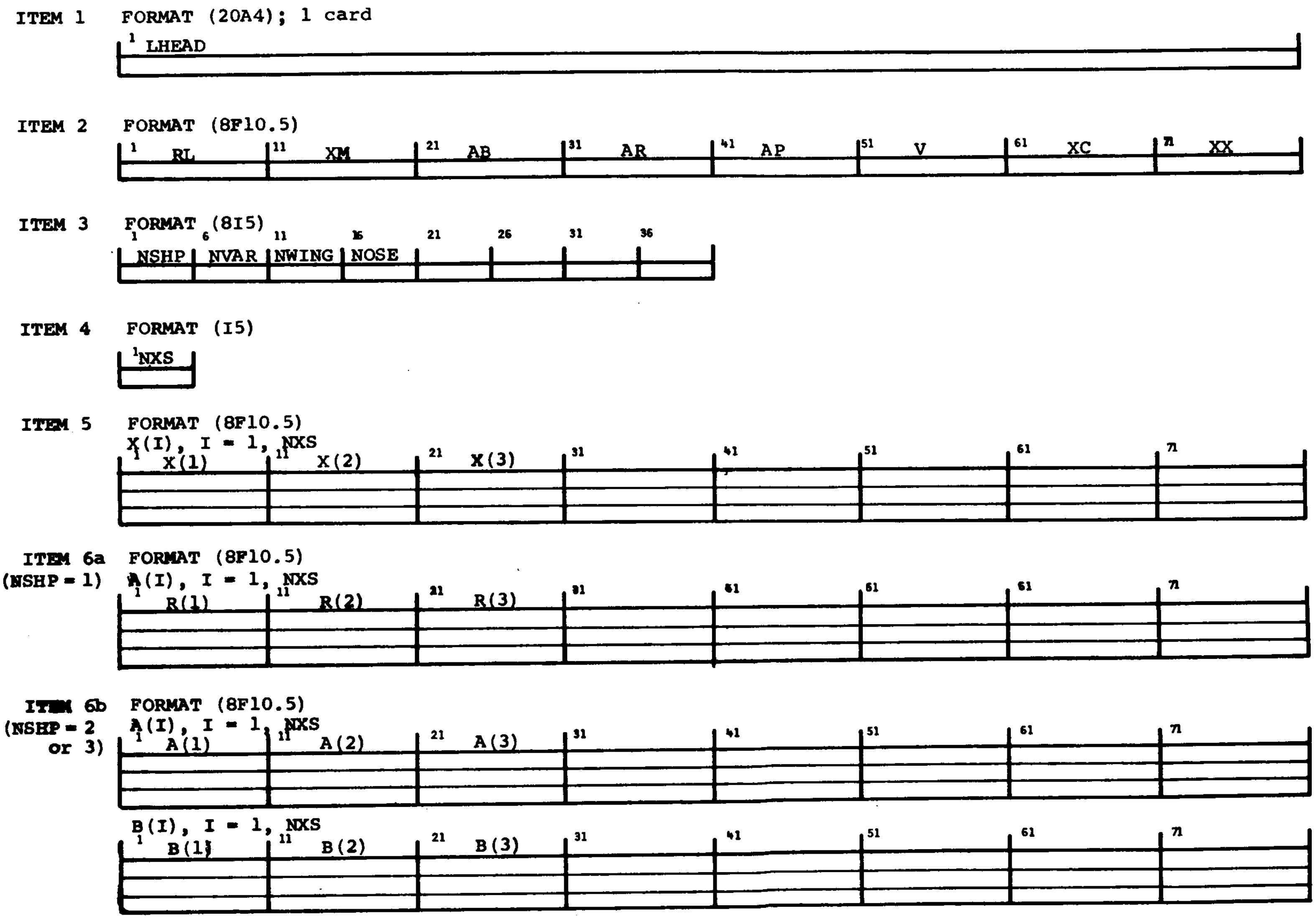
```
      FUNCTION CDN(FMACHN,REN)
C
C     THIS SUBROUTINE CALCULATES, BY LOOKING IN TABLES AND
C     INTERPOLATING, THE CROSS-FLOW DRAG COEFFICIENT CDN
C
      DIMENSION  X1(20),TAB1(20),X2(15),TAB2(15),TMACH(6),TAMS(10,6)
     1,X3(10)
      DIMENSION X4(10),TAB4(10)
      REAL INTRP
      DATA X1/.4,.5,.6,.7,.8,.9,.96,1.,1.1,1.2,1.3,1.4,1.5,1.6,1.8,2.0,
     12.2,2.8,4.0,4.8 /
      DATA TAB1/1.29,1.36,1.50,1.62,1.66,1.8,2.1,2.04,1.85,1.71,1.64,
     11.56,1.52,1.5,1.45,1.4,1.38,1.33,1.28,1.27/
      DATA NTAB1/20/,NTAB2/11/
      DATA X2/4.0,4.5441,4.9542,5.1761,5.5563,5.6021,5.699,5.7782,
     15.8451,5.9542,6.0,4*0.0/
      DATA TAB2/1.2,1.2,1.22,1.22,.72,.32,.26,.25,.26,.28,.29,4*0.0/
      DATA TMACH/.25,.3,.35,.4,.45,.5/
      DATA NTMACH/6/,NTAB3/10/
      DATA X3/6.0,6.301,6.4771,6.6021,6.699,6.7782,6.8451,6.9031,
     16.9542,7.0/
      DATA TAB3/.2,.45,.54,.59,.6,.59,.58,.55,.53,.52,
     1 .3,.48,.56,.6,.62,.61,.605,.6,.59,.58,
     2 .3,.45,.53,.6,.64,.69,.7,.64,.62,.6,
     3 .5,.53,.57,.6,.64,.69,.76,.88,.67,.63,
     4 20*0.0/
      DATA X4/0.,.1,.2,.3,.4,.5,4*0./,NTAB4/6/
      DATA TAB4/1.2,1.2,.92,.28,.28,1.36,4*0.0/
C
C     STATEMENT FUNCTION USED FOR INTERPOLATION
C
      INTRP(XL,XU,YL,YU,X) = YL + (X-XL)*(YU-YL)/(XU-XL)
C
C     CURVE SELECTION LOGIC
C
      IF(FMACHN-0.4) 2,2,5
    2 IF(REN-1.E+4) 55,55,3
    3 IF(REN-1.E+6) 65,65,4
    4 IF(REN-1.E+8) 70,70,8
    5 IF(FMACHN-0.5) 40,40,8
C
C     USE THE DATA FROM FIGURE 1 NASA TN D-6996
C
    8 DO 10 I=1,NTAB1
      IF(FMACHN-X1(I))11,12,10
   10 CONTINUE
   11 XL=X1(I-1)
      XU=X1(I)
      YL=TAB1(I-1)
      YU=TAB1(I)
      X=FMACHN
  100 CDN=INTRP(XL,XU,YL,YU,X)
      RETURN
   12 CDN=TAB1(I)
      RETURN
C
C     USE DATA ON LOWER CURVE, FIG. 1 TN D-6996
C
   40 DO 60 I=1,NTAB4
      IF(FMACHN-X4(I))50,50,60
   60 CONTINUE
   50 XL=X4(I-1)
      XU=X4(I)
      YL=TAB4(I-1)
      YU=TAB4(I)
      X=FMACHN
      GO TO 100
C
C     THIS PATH SHORT CIRCUITS FIGURE 2 FOR REN .LE. 1.E+4
C
   55 CDN=1.2
      RETURN
C
C     USE DATA FROM FIGURE 2 , NASA TN D-6996
C
   65 ALGR=ALOG10(REN)
      DO 66 I=1,NTAB2
      IF(ALGR-X2(I)) 67,68,66
   66 CONTINUE
   67 XL=X2(I-1)
      XU=X2(I)
      YL=TAB2(I-1)
      YU=TAB2(I)
      X=ALGR
      GO TO 100
   68 CDN=TAB2(I)
      RETURN
C
C     USE THE DATA FROM FIGURE 3, NASA TN D-6996 (DOUBLE INTERP.)
C
   70 X=ALOG10(REN)
      DO 72 I=1,NTMACH
      IF(FMACHN-TMACH(I)) 74,73,72
   72 CONTINUE
   73 DO 75 K=1,NTAB3
      IF(X-X3(K)) 83,82,75
   75 CONTINUE
   82 CDN=TAB3(K,I)
      RETURN
   83 XXL=X3(K-1)
      XXU=X3(K)
      YYL=TAB3(K-1,I)
      YYU=TAB3(K,I)
  110 CDN=INTRP(XXL,XXU,YYL,YYU,X)
      RETURN
   74 TML=TMACH(I-1)
      TMU=TMACH(I)
      DO 78 K=1,NTAB3
      IF(X-X3(K)) 86,85,78
   78 CONTINUE
   85 V1=TAB3(K,I-1)
      V2=TAB3(K,I)
      GO TO 90
   86 TXL=X3(K-1)
      TXU=X3(K)
      TYL=TAB3(K-1,I)
      TYU=TAB3(K,I)
      V1=INTRP(TXL,TXU,TYL,TYU,X)
      TYL=TAB3(K-1,I-1)
      TYU=TAB3(K,I-1)
      V2=INTRP(TXL,TXU,TYL,TYU,X)
   90 CDN=INTRP(TML,TMU,V1,V2,FMACHN)
      RETURN
      END
```

```
      FUNCTION ETA(FMACH)
C       RATIO OF CROSSFLOW DRAG COEFFICIENT FOR A FINITE LENGTH CYLINDER
C        TO THAT FOR AN INFINITE LENGTH CYLINDER - FIG. 4 NASA TN D-6996
      COMMON /FLOW/DUM,GAM,ANOSE,BNOSE
      COMMON RL,XM,NSHP,NVAR,NWING,NOSE,NXS,X(50),A(50),B(50),R(50),
     1 SBG(50),RK,SNOSE,RNOSE,XLE,XLET,XTET,XTE,SSPAN,ALE,BLE,RLE
      DIMENSION X1(10),TAB1(10)
      REAL INTRP
      DATA NTAB1/6/,X1/0.,4.,12.,20.,32.,40.,4*0./,TAB1/.5,.61,.68,.75,
     1  .8,.82,4*0./
      INTRP(XL,XU,YL,YU,X)= YL + (X-XL)*(YU-YL)/(XU-XL)
C
C
      IF(FMACH-1.) 20,10,10
   10 ETA=1.
      RETURN
   20 GO TO (21,22,22,21), NSHP
   21 XX=RL/RNOSE
      GO TO 25
   22 XX=RL/SQRT(ANOSE*BNOSE)
   25 CONTINUE
      DO 30 I=1,NTAB1
      IF(XX-X1(I)) 32,31,30
   30 CONTINUE
   31 ETA=TAB1(I)
      RETURN
   32 IM=I-1
      ETA=INTRP(X1(IM),X1(I),TAB1(IM),TAB1(I),XX)
      RETURN
      END
```

```
      FUNCTION FUN(X)
C
C     THIS FUNCTION CALCULATES ALL THE INTEGRANDS IN THE EQUATION
C     FOR CN AND FOR CM FOR THE VARYING-SHAPE CASES
C
      COMMON /FP/RLAM/FSB/R/INDEX/INT
      COMMON RDUM,XM
C
C
      GO TO (1,2,1,2), INT
    1 F=CNSB(X)
      GO TO (10,20,30,40), INT
    2 F=CNNT(X)
      GO TO (10,20,30,40), INT
   10 FUN=F*RLAM
      RETURN
   20 FUN=F*R
      RETURN
   30 FUN=F*RLAM*(XM-X)
      RETURN
   40 FUN=F*R*(XM-X)
      RETURN
      END
```

```
      SUBROUTINE SIMP(DXL,DXU,NIT,NIT1,DTOL,FUN,DANS)
      NIT1=10
      DH=(DXU-DXL)/2.
      DSUM1=FUN(DXL)+FUN(DXU)
      DSUM2=FUN(DXL+DH)
      DANS=DH*(DSUM1+4.00*DSUM2)/3.00
      N=2
      DO 1 I=1,NIT
      DANS1=DANS
      N=N*2
      DH=DH/2.
      DSUM3=0.
      NLIM=N-1
      DO 2 K=1,NLIM,2
      DK=K
    2 DSUM3=DSUM3 + FUN(DXL + DH*DK)
      DANS=DH*(DSUM1+2.00*DSUM2+4.00*DSUM3)/3.
      IF(ABS(DANS).LE.1.E-8) GO TO 5
      DERR=(DANS-DANS1)/DANS
      IF(ABS(DERR).LE.DTOL) RETURN
      GO TO 1
    5 CONTINUE
      DERR=ABS(DANS-DANS1)
      IF(DERR.LE.1.E-8) RETURN
    1 DSUM2=DSUM2+DSUM3
      NIT1=0
      RETURN
      END
```

```
      SUBROUTINE CEL2(RES,AK,A,B,IER)                                   CF11 001
C                                                                       CF11 002
C     ..................................................................CF11 003
C                                                                       CF11 004
C        SUBROUTINE CEL2                                                CF11 005
C                                                                       CF11 006
C        PURPOSE                                                        CF11 007
C           COMPUTES THE GENERALIZED COMPLETE ELLIPTIC INTEGRAL OF      CF11 008
C           SECOND KIND.                                                CF11 009
C                                                                       CF11 010
C        USAGE                                                          CF11 011
C           CALL CEL2(RES,AK,A,B,IER)                                   CF11 012
C                                                                       CF11 013
C        DESCRIPTION OF PARAMETERS                                      CF11 014
C           RES    - RESULT VALUE                                       CF11 015
C           AK     - MODULUS (INPUT)                                    CF11 016
C           A      - CONSTANT TERM IN NUMERATOR                         CF11 017
C           B      - FACTOR OF QUADRATIC TERM IN NUMERATOR              CF11 018
C           IER    - RESULTANT ERROR CODE WHERE                         CF11 019
C                    IER=0  NO ERROR                                    CF11 020
C                    IER=1  AK NOT IN RANGE -1 TO +1                    CF11 021
C                                                                       CF11 022
C        REMARKS                                                        CF11 023
C           FOR AK = +1,-1 THE RESULT VALUE IS SET TO 1.E30 IF B IS     CF11 024
C           POSITIVE, TO -1.E30 IF B IS NEGATIVE.                       CF11 025
C           SPECIAL CASES ARE                                           CF11 026
C           K(K) OBTAINED WITH A = 1, B = 1                             CF11 027
C           E(K) OBTAINED WITH A = 1, B = CK*CK WHERE CK IS             CF11 028
C           COMPLEMENTARY MODULUS.                                      CF11 029
C           B(K) OBTAINED WITH A = 1, B = 0                             CF11 030
C           D(K) OBTAINED WITH A = 0, B = 1                             CF11 031
C           WHERE K, E, B, D DEFINE SPECIAL CASES OF THE GENERALIZED    CF11 032
C           COMPLETE ELLIPTIC INTEGRAL OF SECOND KIND IN THE USUAL      CF11 033
C           NOTATION, AND THE ARGUMENT K OF THESE FUNCTIONS MEANS       CF11 034
C           THE MODULUS.                                                CF11 035
C                                                                       CF11 036
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                  CF11 037
C           NONE                                                        CF11 038
C                                                                       CF11 039
C        METHOD                                                         CF11 040
C           DEFINITION                                                  CF11 041
C           RES=INTEGRAL((A+B*T*T)/(SQRT((1+T*T)*(1+(CK*T)**2))*(1+T*T))CF11 042
C           SUMMED OVER T FROM 0 TO INFINITY).                          CF11 043
C           EVALUATION                                                  CF11 044
C           LANDENS TRANSFORMATION IS USED FOR CALCULATION.             CF11 045
C           REFERENCE                                                   CF11 046
C           R.BULIRSCH, 'NUMERICAL CALCULATION OF ELLIPTIC INTEGRALS    CF11 047
C           AND ELLIPTIC FUNCTIONS', HANDBOOK SERIES SPECIAL FUNCTIONS, CF11 048
C           NUMERISCHE MATHEMATIK VOL. 7, 1965, PP. 78-90.              CF11 049
C                                                                       CF11 050
C     ..................................................................CF11 051
C                                                                       CF11 052
C                                                                       CF11 053
      IER=0                                                             CF11 054
C                                                                       CF11 055
C     TEST MODULUS                                                      CF11 056
C                                                                       CF11 057
      GEO=1.-AK*AK                                                      CF11 058
      IF(GEO)1,2,6                                                      CF11 059
    1 IER=1                                                             CF11 060
      RETURN                                                            CF11 061
C                                                                       CF11 062
C     SET RESULT VALUE = OVERFLOW                                       CF11 063
C                                                                       CF11 064
    2 IF(B)3,5,4                                                        CF11 065
    3 RES=-1.E30                                                        CF11 066
      RETURN                                                            CF11 067
    4 RES=1.E30                                                         CF11 068
      RETURN                                                            CF11 069
    5 RES=A                                                             CF11 070
      RETURN                                                            CF11 071
C                                                                       CF11 072
C     COMPUTE INTEGRAL                                                  CF11 073
C                                                                       CF11 074
    6 GEO=SQRT(GEO)                                                     CF11 075
      ARI=1.                                                            CF11 076
      AA=A                                                              CF11 077
      AN=A+B                                                            CF11 078
      W=B                                                               CF11 079
    7 W=W+AA*GEO                                                        CF11 080
      W=W+W                                                             CF11 081
      AA=AN                                                             CF11 082
      AARI=ARI                                                          CF11 083
      ARI=GEO+ARI                                                       CF11 084
      AN=W/ARI+AN                                                       CF11 085
C                                                                       CF11 086
C     TEST OF ACCURACY                                                  CF11 087
C                                                                       CF11 088
      IF(AARI-GEO-1.E-4*AARI)9,9,8                                      CF11 089
    8 GEO=SQRT(GEO*AARI)                                                CF11 090
      GEO=GEO+GEO                                                       CF11 091
      GO TO 7                                                           CF11 092
    9 RES=.78539816*AN/ARI                                              CF11 093
      RETURN                                                            CF11 094
      END                                                               CF11 095
```



(a) Page 1.

Figure 1.- Input format for CRSFLW program.

ITEM 6c (NSHP = 4) FORMAT (8F10.5)

1
RK

SSQ(I), I = 1, NXS

1	11	21	31	41	51	61	71
SSQ(1)	SSQ(2)	SSQ(3)					

ITEM 7 FORMAT (2F10.5) Omit Item 7 if NOSE = 4.

1
SNOSE

ITEM 7a (NSHP = 1)

1
RNOSE

ITEM 7b (NSHP = 2 or 3)

1	11
ANOSE	BNOSE

ITEM 8 FORMAT (8F10.5) Omit Item 8 if NWING = 0.

1	11	21	31	41	51	61	71
XLE	XLET	XTET	XTE	SSPAN	EPS		

ITEM 9a FORMAT (8F10.5) Omit Item 9 if NWING = 0.

1
RLE

ITEM 9b (NSHP = 2 or 3)

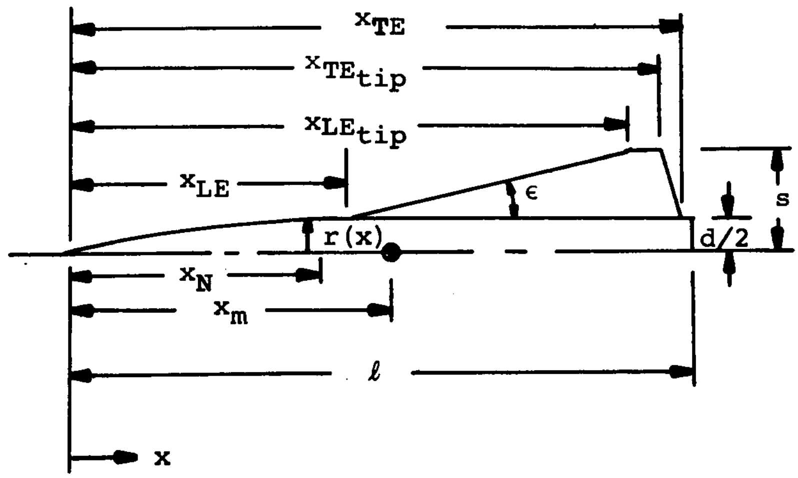
1	11
ALE	BLE

ITEM 10 FORMAT (8F10.5)

1	11	21	31	41	51	61	71
ALFI	ALFF	DALF	CASF	GAM	RE	FMACH	

(b) Page 2.

Figure 1.- Concluded.



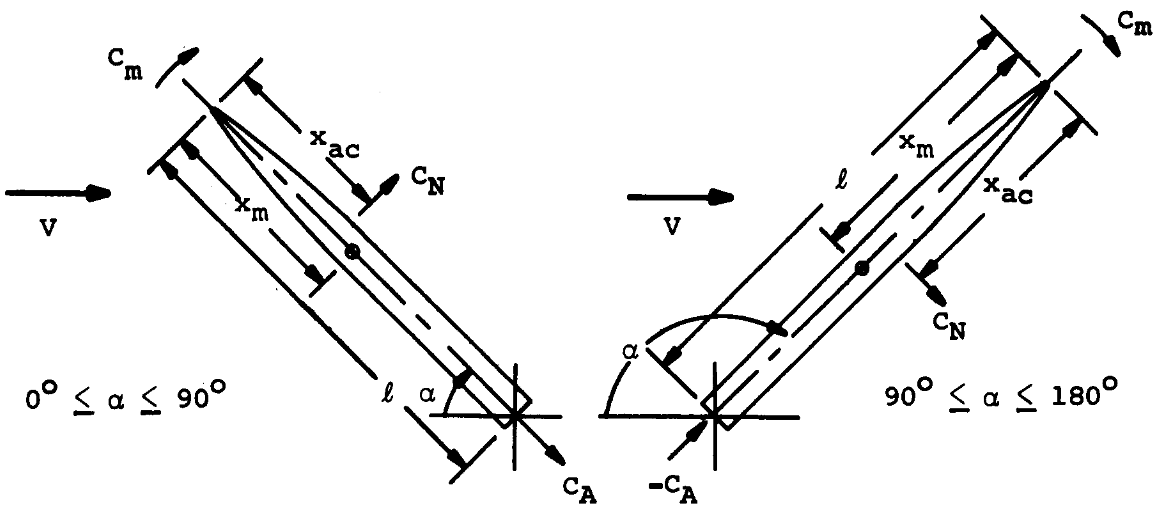


Figure 2.- Algebraic notation defining configuration.

```
SAMPLE CASE  1     NASA TN D=6996      BODY NO, 2      FLAT=NOSE CYLINDER
 30,48     15,24      11,40      11,40      116,13     347,32     15,24        3,81
  1    1    0    4
  2
 0,0       30,48
 1,905     1,905
  0,0      180,0      10,0       0,0        1,4       125000,     2,86
SAMPLE CASE  2     NASA TN D=6996      BODY NO, 5      CONE=CYLINDER
41,91      20,96      11,40      11,40      137,90     390,74     23,66        3,81
  1    1    0    1
  3
 0,0       11,43      41,91
0,0        1,905      1,905
 11,43
 1,905
  0,0      180,0      10,0       0,0        1,4       125000,     2,86
SAMPLE CASE  3   NASA TN D=7228  WING=CONICAL BODY         1,5 ASPECT RATIO
 25,18     25,18      36,96      36,96      237,95     310,22     18,88       25,18
  1    1    1    1
  2
0,0        25,18
0,0        3,43
 25,18
 3,43
 0,0       25,18      25,18      25,18       9,45       ,359
 0,0
 0,0       90,0       5,0        0,0        1,4       2180000,   1,97
SAMPLE CASE  4   NASA TN D=7228  WING=ELLIPTICAL BODY , MAJOR AXIS HORIZONTAL
 25,18     25,18      36,96      36,96      237,95     310,22     18,88       25,18
  2    1    1    1
  2
0,0        25,18
0,0        5,94
0,0        1,98
 25,18
 5,94      1,98
 0,0       25,18      25,18      25,18       9,45       ,359
 0,0       0,0
0,        90,        5,         0,0        1,4       2180000,   1,97
SAMPLE CASE  5   NASA TN D=7228  WING=ELLIPTICAL BODY , MAJOR AXIS VERTICAL
 25,18     25,18      36,96      36,96      237,95     310,22     18,88       25,18
  3    1    1    1
  2
0,0        25,18
0,0        5,94
0,0        1,98
 25,18
 5,94      1,98
 0,0       25,18      25,18      25,18       9,45       ,359
 0,0       0,0
 0,0       90,0       5,0        0,0        1,4       2180000,   1,97
```

Figure 3.- Sample input decks for CRSFLW program.

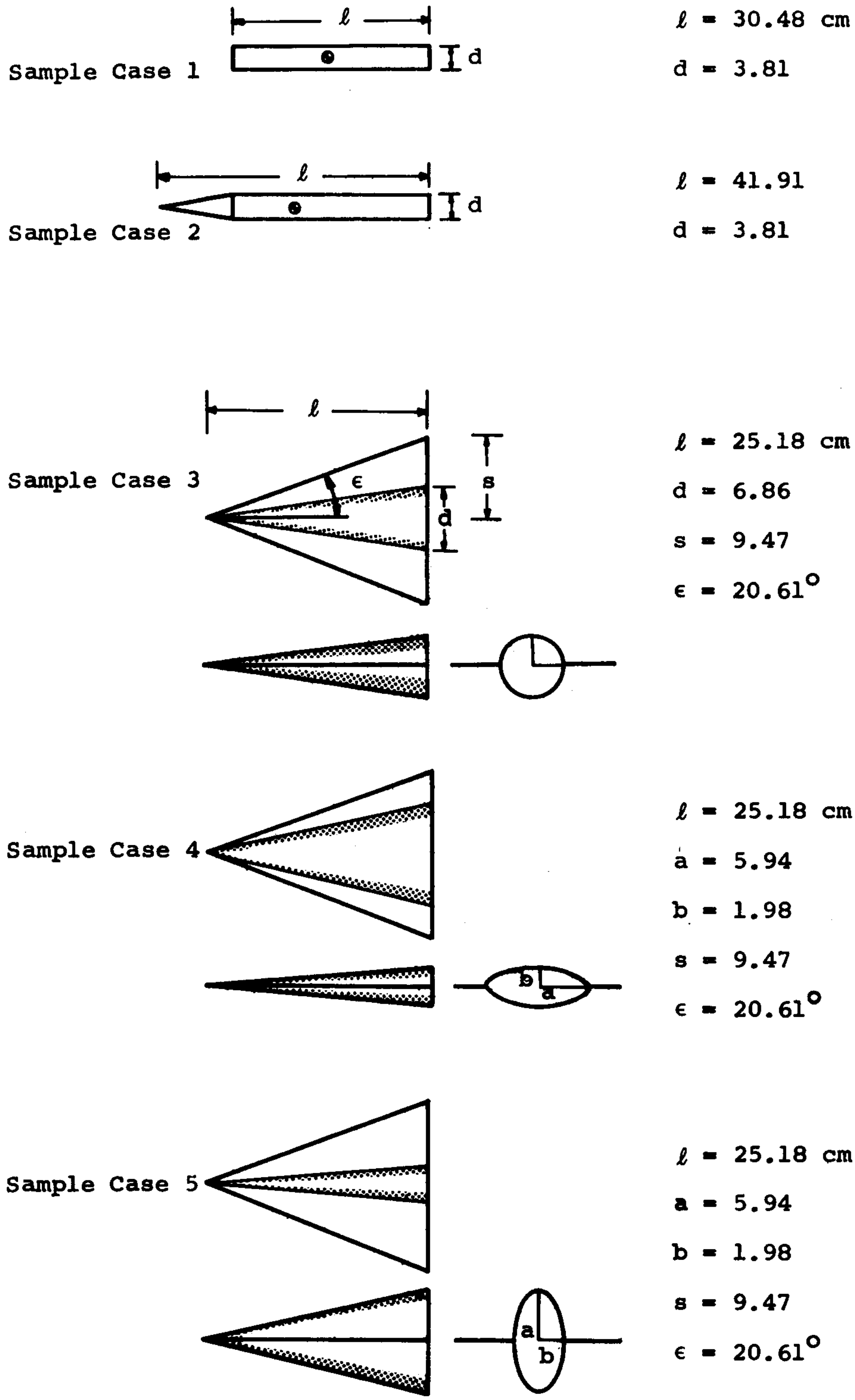


Figure 4.- Configurations used in sample cases
for CRSFLW program.

```
        SAMPLE CASE  4   NASA TN D-7228  WING-ELLIPTICAL BODY , MAJOR AXIS HORIZONTAL
                BODY INPUT DATA

  RL =   25.18000   XM =   25.18000
  AB =   36.96000   AR =   36.96000   AP =  237.95000
   V =  310.22000   XC =   18.88000   XR =   25.18000


      ELLIPTICAL BODY  (NSHP = 2)    BODY SHAPE SIMILAR OVER LENGTH
  TABULATION OF A AND B
I          X(I)           A(I)           B(I)

1          .000           .000           .000
2        25.180          5.940          1.980

      CONICAL NOSE  (NOSE = 1)

      NOSE LENGTH =   25.18000      RADIUS AFT OF NOSE =    3.42946

      WING IS PRESENT WITH LEADING EDGE AT X =     .00000  AND TRAILING EDGE AT X =   25.18000  , EPSILON =     .35900


      FLOW CONDITIONS
  REYNOLDS NUMBER =   .21800E+07     GAMMA =  1.400     MACH NUMBER =  1.970
```

(a) Page 1.

Figure 5.- Output from CRSFLW program.

SAMPLE CASE 4 NASA TN D-7228 WING-ELLIPTICAL BODY , MAJOR AXIS HORIZONTAL

ALPHA	CN / CN1	CA / CN2	CM / CA1	XAC / CA2	CL / CA3	CD / CM1	L/D / CM2	CDC
.000	-.0000	.2467	-.0000	.0000	.0000	.2467	.0000	
	.0	.0	6.491E-02	.0	1.818E-01	.0	.0	1.200E+00
5.000	.9808	.2448	.4784	12.8987	.9558	.3294	2.9017	
	9.067E-01	7.413E-02	6.491E-02	.0	1.818E-01	4.537E-01	2.471E-02	1.085E+00
10.000	1.9245	.2393	.9389	12.8950	1.8537	.5698	3.2535	
	1.781E+00	1.458E-01	6.491E-02	.0	1.818E-01	8.910E-01	4.792E-02	5.301E-01
15.000	3.4185	.2302	1.5723	13.5990	3.2425	1.1070	2.9292	
	2.591E+00	8.276E-01	6.491E-02	.0	1.818E-01	1.296E+00	2.759E-01	1.374E+00
20.000	4.9797	.2179	2.2125	13.9923	4.6050	1.9076	2.4140	
	3.309E+00	1.671E+00	6.491E-02	.0	1.818E-01	1.655E+00	5.570E-01	1.588E+00
25.000	6.6485	.2027	2.8691	14.3138	5.9402	2.9930	1.9847	
	3.909E+00	2.740E+00	6.491E-02	.0	1.818E-01	1.956E+00	9.132E-01	1.705E+00
30.000	9.0105	.1851	3.7338	14.7458	7.7113	4.6648	1.6531	
	4.372E+00	4.638E+00	6.491E-02	.0	1.818E-01	2.188E+00	1.546E+00	2.063E+00
35.000	10.0354	.1656	4.1276	14.8234	8.1262	5.8908	1.3795	
	4.684E+00	5.351E+00	6.491E-02	.0	1.818E-01	2.344E+00	1.784E+00	1.808E+00
40.000	11.0203	.1448	4.4814	14.9405	8.3499	7.1935	1.1607	
	4.837E+00	6.183E+00	6.491E-02	.0	1.818E-01	2.420E+00	2.061E+00	1.664E+00
45.000	11.6713	.1234	4.7638	15.0756	8.3083	8.4803	.9797	
	4.830E+00	7.042E+00	6.491E-02	.0	1.818E-01	2.417E+00	2.347E+00	1.566E+00
50.000	12.6798	.1020	5.0060	15.2389	8.0739	9.7775	.8258	
	4.666E+00	8.014E+00	6.491E-02	.0	1.818E-01	2.335E+00	2.671E+00	1.518E+00
55.000	13.3910	.0812	5.1916	15.4179	7.6163	11.0145	.6915	
	4.358E+00	9.033E+00	6.491E-02	.0	1.818E-01	2.181E+00	3.011E+00	1.497E+00
60.000	13.8627	.0617	5.2759	15.5968	6.8803	12.0349	.5717	
	3.922E+00	9.941E+00	6.491E-02	.0	1.818E-01	1.962E+00	3.314E+00	1.474E+00
65.000	14.1197	.0441	5.2709	15.7802	5.9300	12.8142	.4628	
	3.379E+00	1.074E+01	6.491E-02	.0	1.818E-01	1.691E+00	3.580E+00	1.454E+00
70.000	14.1708	.0289	5.1836	15.9692	4.8226	13.3249	.3619	
	2.754E+00	1.142E+01	6.491E-02	.0	1.818E-01	1.378E+00	3.806E+00	1.437E+00
75.000	14.0304	.0166	5.0235	16.1645	3.6187	13.5557	.2669	
	2.076E+00	1.195E+01	6.491E-02	.0	1.818E-01	1.039E+00	3.985E+00	1.424E+00
80.000	13.7178	.0075	4.8017	16.3661	2.3783	13.5101	.1760	
	1.372E+00	1.235E+01	6.491E-02	.0	1.818E-01	6.864E-01	4.115E+00	1.415E+00
85.000	13.2551	.0019	4.5305	16.5736	1.1571	13.2045	.0876	
	6.715E-01	1.258E+01	6.491E-02	.0	1.818E-01	3.360E-01	4.195E+00	1.409E+00
90.000	12.6658	.0000	4.2223	16.7859	.0038	12.6658	.0003	
	2.191E-03	1.266E+01	6.491E-02	.0	1.818E-01	1.096E-03	4.221E+00	1.408E+00

(b) Page 2.

Figure 5.- Concluded.

PART III - SUBSON COMPUTER PROGRAM

Introduction

This computer program predicts the static longitudinal aerodynamic characteristics of wing-body-tail combinations at subsonic speeds. It is an extension of the method of reference 2 to angles of attack for which symmetrical body vortices are shed from the nose of the configuration and leading-edge and side-edge separation vortices are shed from the wing and tail. The body is limited to circular cross-section shapes.

The program is written in FORTRAN IV for the IBM 360 series machines. No tapes, drums, or disks other than the standard input/output units are required. Minor changes are required to run the program on other machines such as the CDC 6600. Typical running time on the IBM 360/67 for a wing-body-tail configuration is the order of one to two minutes. Actual time is dependent on the type of vortex lattice used to represent the lifting surfaces and whether or not a trim condition is calculated. Some specific running times are noted in the discussion of the sample cases.

The following sections present a description of the program, a description of the input, a description of the output, a program listing, and sample cases. The algebraic notation used in this section is the same as that used in reference 1. A list of symbols from reference 1 is included for reference.

List of Symbols

$\mathcal{R}$	aspect ratio
a	local body radius
b	semispan
C_A	axial-force coefficient
C_{d_c}	crossflow drag coefficient
C_{D_i}	induced drag coefficient
C_L	lift coefficient, $\frac{L}{qS}$
C_m	pitching-moment coefficient, $\frac{M}{qS\ell}$

C_N	normal-force coefficient, $\frac{N}{qS}$
c	local chord
c_ℓ	section-lift coefficient
c_s	section leading-edge suction coefficient
K_v^*	vortex-lift ratio, figure 4
L	lift force
ℓ	reference length
M	pitching moment about center of moments, or free-stream Mach number
N	normal force
q	free-stream dynamic pressure
r	body radius
r_N	radius of base of nose
S	reference area
u,v,w	perturbation velocities along x,y,z directions, respectively
V	free-stream velocity
x,y,z	configuration coordinates with origin at body nose, figure 2
x_m	x location of center of moments
x_s	x position for onset of separation from body nose
$\bar{x}$	center-of-pressure location
z_m	z location of center of moments
α	body angle of attack
β	$\sqrt{1 - M^2}$
Γ_B	right body-vortex strength, positive counterclockwise when viewed from rear of configuration
Γ_n	n'th separation-vortex strength on right wing panel
Γ_t	trailing-vortex strength on right wing panel

δ	lifting-surface deflection angle, positive trailing edge down
θ_N	nose angle, degrees
Λ	sweep angle
ρ	density
σ	complex vortex position, y + iz
ϕ	dihedral angle

Subscripts

A	afterbody
avg	average
B	body
B(T)	body in presence of tail
B(W)	body in presence of wing
c	canard
CP	center of pressure
e	tail or empennage
HL	hinge line
LE	leading edge
N	nose
p	potential
root	root chord
SE	side edge
T(B)	tail in presence of body
TE	trailing edge
t	trailing vortex
tip	tip chord
v	vortex
W(B)	wing in presence of body
w	wing

Description of Method

A brief description of the method is presented herein. The user should consult reference 1 for a complete description and details of the theoretical approach.

An axisymmetric nose at some moderate angle of attack sheds a symmetric pair of body vortices. These shed body vortices, whose strength and position are determined from data correlations, are tracked downstream past the wing using slender-body techniques in the crossflow plane. The vortex-induced velocities are computed at the wing control points and combined with the Beskin upwash induced by the body to obtain the total upwash induced on each wing panel. This, added to the free-stream contribution, results in a total local incidence angle distribution over the wing.

The lifting surfaces (wing and horizontal tail or canard) are modeled by a vortex-lattice scheme (ref. 8) which has the capability to include velocity fields from external sources. The total upwash at the control points must be cancelled by the wing-circulation-induced velocity to satisfy the tangency boundary condition of the vortex-lattice method. The wing loading and trailing-vortex strength and position are obtained from this vortex-lattice calculation. The distribution of leading-edge suction and side-edge suction (if present) and their associated vortex positions and strengths are also obtained from the vortex-lattice calculations. The leading-edge separation vortex lift is obtained from the suction distribution with the help of the Polhamus vortex-lift analogy (ref. 9) and correlation curves.

The trajectories of the body vortices, the wing trailing vortex, and the wing leading-edge separation vortices are computed downstream past the afterbody and horizontal tail. These trajectories are computed in the crossflow plane considering mutual interference between the vortices and interference from their images in the body. The induced velocity field at the tail is computed, and the tail loading is obtained in a manner similar to that just described for the wing. The forces on the body due to the presence of the wing and tail are computed by the method of reference 2. The free vortex-induced forces on the body are computed using the method of Sacks (ref. 10).

The forces and moments on the entire configuration are obtained by summing the contributions of the various components. These forces are resolved into normal and axial force (excluding frictional drag), and lift and induced drag.

The subsonic prediction method includes an option to compute the trim conditions of a wing-body-tail configuration at some specified angle of attack. This is carried out by an iterative process in which the incidence of the tail or wing (canard) is varied until a zero pitching moment is achieved.

Description of Program

The SUBSON computer program consists of a main program and fifteen subroutines. The main program (SB01) accepts all the input, prints most of the output, and generally directs the flow of the calculation. The subroutines or groups of subroutines provide specific services to the main program during the calculation procedure. The following is a list of the subroutines and their general purpose.

Subroutine LATTUS sets up the horseshoe vortex-lattice arrangement for the lifting surfaces. It locates the coordinates of the control points, calculates the influence coefficient matrix, and computes any geometry-related parameters connected with the lifting surfaces.

Subroutine SHAPE does a table look-up for the body radius and slope and local lifting-surface semispan at any prescribed axial station.

Subroutine BDYVTX uses tables derived from data correlations to look up the strength and position of the pair of symmetric vortices shed from the body nose.

Subroutine CNVNZ computes the nose vortex-induced normal force and pitching moment on the nose of the body using the method of reference 10.

Subroutine TRJTRY computes the trajectories of the free vortices past the configuration using the subroutines FCT, OUTP, HPCG, and SHAPE.

Subroutine FCT computes the derivatives in the equations of motion for each free vortex.

Subroutine HPCG is a predictor-corrector integration package which uses a Runge-Kutta starting procedure.

Subroutine OUTP stores the vortex positions in a table at specified intervals in x. When necessary, some diagnostic information on the vortex trajectories is available as optional output.

Subroutine EXTVEL computes the vortex-induced velocity at wing or tail control points.

Subroutine VTXLAT computes the strengths of the bound vortices on the lifting surface. It also computes the leading-edge suction distribution, the strength and spanwise position of the associated separation vortex, and the strength and position of the trailing vortex. This subroutine calls subroutines LOAD1 and INVERS.

Subroutine LOAD1 uses the circulation distribution from the previous subroutine to compute the span-loading distribution and the forces and center of pressure on the lifting surface.

Subroutine INVERS solves a system of linear simultaneous equations for the circulation strengths.

Subroutine INFWW computes the influence function for a horseshoe vortex.

Subroutine ZVTX determines the vertical position of the leading-edge separation vortex using a table look-up of correlated data for delta wings.

Subroutine CNVTX computes the vortex-induced force and center of pressure on the afterbody using the method of reference 10.

Description of Input

Variable definitions.- The format of the input cards for the SUBSON program is shown in figure 1. In this figure the program variable name is shown as well as the card columns in which the value is punched and the format in which it is punched. The following is a table of the input variables along with the algebraic symbol where applicable. The input length and area quantities are dimensional and should have consistent dimensions. The variable is defined and its limits shown where necessary. The algebraic notation used in defining the configuration is shown in figure 2. A discussion of the preparation of the input is presented in the section following the table.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
Item 1		
NHEAD		Number of heading cards.
NTBL		Number of entries in table of body coordinates. $5 \leq$ NTBL $\leq$ (96 - MSWW - MSWT)
NPRINT		Output option: NPRINT $<$ 0 Minimum output, final aerodynamic characteristics only. = 0 Standard output. = 10 Optional additional output.
MPRINT		Output option in trajectory calculation: MPRINT = 0 No additional output. $>$ 0 Output vortex trajectories as calculated. (This option should be used as diagnostic only if program has prematurely terminated execution during a previous trajectory calculation.)
NOSEV		NOSEV = 0 No nose separation vortex pair. = 1 Nose separation vortex pair included.
NTRIM		NTRIM = 0 No trim calculation. $>$ 0 Trim condition calculation, as follows. = 1 Wing incidence variable. = 2 Tail incidence variable.
NCWW		Number of chordwise vortices on wing. $2 \leq$ NCWW ≤ 10
MSWW		Number of spanwise vortices on wing. $2 \leq$ MSWW ≤ 25

NOTE: (NCWW $\times$ MSWW) ≤ 100.

NCAMW		NCAMW = 0 No wing camberline slopes to be input. = 1 Camberline slope at each wing control points to be input.
NPSIW		NPSIW = 0 Unbroken wing leading edge and trailing edge. $>$ 0 Input leading- and trailing-edge sweep angles at each specified spanwise station.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
NSEPW		Number of leading-edge separation vortices shed from wing. $1 \leq$ NSEPW ≤ 2
NCWT		Number of chordwise vortices on tail. $2 \leq$ NCWT ≤ 10
MSWT		Number of spanwise vortices on tail. $2 \leq$ MSWT ≤ 25

NOTE: (NCWT × MSWT) ≤ 100.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
NCAMT		NCAMT = 0 No tail camberline slopes to be input. > 0 Camberline slopes at tail control points to be input.
NPSIT		NPSIT = 0 Straight tail leading edge and trailing edge. > 0 Input leading- and trailing-edge sweep angles at each specified tail spanwise station.
NSEPT		Number of leading-edge separation vortices shed from tail. $1 \leq$ NSEPT ≤ 2
NBODY		Index controlling body upwash. NBODY < 0 No body upwash included in wing and tail interference calculation. = 0 Body upwash included.

Item 2

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
TITLE		Any alphabetic or numeric identification information. Number of cards equal to NHEAD.

Item 3

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
XBDY(J)		x-station at which body coordinates are defined.
RBDY(J)		Body radius at above stations.

Item 4

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
XM	x_m	x-coordinate of moment center.
ZM	z_m	z-coordinate of moment center.
EMACH	M	Mach number.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
REFS	S	Reference area.
REFL	ℓ	Reference length.
THETAN	θ_N	Nose angle, degrees.
DXOUT		x-increment in output table of free vortex trajectories (DXOUT $\leq$ RAVGW), typically.
DXI		Maximum integration interval for vortex trajectory calculation. (DXI $\leq$ DXOUT)
Item 5		
XLEW	x_{LE_w}	x-coordinate of wing leading-edge intersection with body.
XTEW	x_{TE_w}	x-coordinate of wing trailing-edge intersection with body.
XHLW	x_{HL_w}	x-coordinate of wing hinge line at wing-body juncture.
ZHLW	z_{HL_w}	z-coordinate of wing hinge line at wing-body juncture.
XCPW	x_{CP_w}	x-coordinate of alternate wing center of pressure location.
BS2W	b/2	Wing semispan.
RAVGW	r_{avg_w}	Average body radius at wing.
YSEPW		Spanwise location of 2^{nd} leading-edge separation vortex. If NSEPW = 1, YSEPW = b/2.
Item 6		
PHIW	ϕ	Dihedral angle of wing, positive tip up, degrees.
PSILEW(1)	Λ_{LE_w}	Leading-edge sweep angle at first wing station adjacent to body, degrees. Sweepback is positive.
PSITEW(1)	Λ_{TE_w}	Trailing-edge sweep angle at first wing station adjacent to body, degrees. Sweepforward is negative.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
CMTEST		Tolerance for C_m in trim calculation if NTRIM $\neq$ 0, CMTEST > 0.0.
Item 7		
YW	y	y-coordinate of outboard side of lifting-surface panels on wing. Last value must equal b/2. MSWW values must be input.
Item 8		Delete if NPSIW = 0.
PSILEW	Λ_{LE_w}	Leading-edge sweep angle corresponding to values of y in Item 7, degrees. MSWW values must be input.
Item 9		Delete if NPSIW = 0.
PSITEW	Λ_{TE_w}	Trailing-edge sweep angle corresponding to values of y in Item 7, degrees. MSWW values must be input.
Item 10		
ALPHLW(J)	α_w	Tangent of local angle on wing due to camber and twist. (Values are input from leading edge to trailing edge, from root to tip.) MSWW $\times$ NCWW values are required.
Item 11		
XLET	x_{LE_e}	x-coordinate of tail leading-edge intersection with body.
XTET	x_{TE_e}	x-coordinate of tail trailing-edge intersection with body.
XHLT	x_{HL_e}	x-coordinate of tail hinge line.
ZHLT	z_{HL_e}	z-coordinate of tail hinge line.
XCPT	x_{CP_e}	x-coordinate of alternate tail center of pressure location.
BS2T	b/2	Semispan of tail.
RAVGT	r_{avg_e}	Average body radius at tail.
YSEPT		Spanwise location of second leading-edge separation vortex. If NSEPT = 1, YSEPT = b/2.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
Item 12		
PHIT	ϕ_e	Tail dihedral angle, degrees.
PSILET(1)	Λ_{LE_e}	Leading-edge sweep angle of tail at first station adjacent to body, degrees.
PSITET(1)	Λ_{TE_e}	Trailing-edge sweep angle of tail at first station adjacent to body, degrees.
Item 13		
YT	y	y-coordinate of outboard side of lifting-surface panels on tail. Last value must equal b/2.
Item 14		Delete if MPSIT = 0.
PSILET(J)		Leading-edge sweep angle corresponding to values of y in Item 13, degrees.
Item 15		Delete if MPSIT = 0.
PSITET(J)		Trailing-edge sweep angle corresponding to values of y in Item 13, degrees.
Item 16		
ALPHT(J)	α_e	Tangent of local angle on tail due to camber and twist. (Values are input from leading edge to trailing edge, from root to tip.) MSWT × NCWT values are required.
Item 17		
NDEX		Index controlling next case of input. NDEX = 1 Execute program using variables on this card. = 0 Ignore this card and return to beginning for new case.
ALPHAD	α	Angle of attack of configuration, degrees.
ALPIW	δ_w	Incidence angle of wing relative to body axis, degrees.
AKVLWI	$K^*_{v_{LE_w}}$	Fraction of leading-edge suction converted to lift in inboard wing region. $(0 \leq K^*_{v_{LE}} \leq 1.0)$

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
AKVLW2	$K^*_{v_{LE_w}}$	Fraction of leading-edge suction converted to lift in outboard wing region. ($0 \leq K^*_{v_{LE}} \leq 1.0$)
AKVSW	$K^*_{v_{SE_w}}$	Fraction of side-edge suction converted to lift on wing. ($0 \leq K^*_{v_{SE}} \leq 1.0$)
ALPIT	δ_e	Incidence angle of tail relative to body axis, degrees.
AKVLT1	$K^*_{v_{LE_e}}$	Fraction of leading-edge suction converted to lift in inboard tail region. ($0 \leq K^*_{v_{LE}} \leq 1.0$)
AKVLT2	$K^*_{v_{LE_e}}$	Fraction of leading-edge suction converted to lift in outboard tail region. ($0 \leq K^*_{v_{LE}} \leq 1.0$)
AKVST	$K^*_{v_{SE_e}}$	Fraction of side-edge suction converted to lift on tail. ($0 \leq K^*_{v_{SE}} \leq 1.0$)
WLIMIT	$w/V\|_{max}$	Limit on vortex-induced velocities at wing and tail control points. ($0 \leq$ WLIMIT ≤ 1.0)

Input preparation.- A discussion of the input variables is presented in this section as an aid in the preparation of the input data deck. If a configuration has one set of lifting surfaces, this is denoted the wing regardless of its axial location on the body and data describing this lifting surface are input as wing data. If there are two lifting surfaces, the set nearest the nose is the wing and the aft set is the tail. For example, in a canard-body-wing configuration, the canard data are input as wing data and the wing data are input as tail data. In the following discussion, necessary geometric relations are illustrated in figure 2.

Item number 1 of the input data deck (fig. 1) is a card containing indices specifying particular program options. NHEAD indicates the number of identification cards following in Item 2. NTBL is the number of entries in the table describing the body shape. Note again that only circular cross-sectional bodies are permitted. NPRINT is an index which determines the quantity of output obtained from the program. Typically, this number is zero. For diagnostic purposes, a provision for detailed output information on lattice slopes, induced velocities, and circulation strengths is provided with NPRINT = 10. An abbreviated output summary is obtained by setting NPRINT = -1. Examples of the output variatons are described further in the sample case output. MPRINT is a special index controlling the quantity of output obtained during vortex trajectory calculations. Typically, MPRINT = 0, but if it is greater than zero, the trajectory information is printed as it is computed. The additional trajectory information is useful only if the program fails to compute a trajectory for some particular case, and the additional output may give some clue as to why the calculation failed. This option should only be used after a computational problem has been discovered.

The index NOSEV specifies whether or not there is (NOSEV = 1) or is not (NOSEV = 0) a symmetrical pair of vortices shed from the nose of the configuration. If the nose angle (THETAN) is less than or equal to four degrees, NOSEV is automatically set equal to zero. The next index, NTRIM, determines whether or not a trim calculation is made. No trim calculation is made when NTRIM = 0, and when NTRIM > 0, a trim condition is calculated. If NTRIM = 1, the wing incidence is varied to achieve trim and if NTRIM = 2, the tail incidence is varied.

The following five indices are associated with the wing. NCWW is the number of chordwise rows in the wing lattice. The only quantitative restrictions is that $2 \leq$ NCWW ≤ 10. MSWW is the number of spanwise columns in the wing lattice, and it must fall within the range $2 \leq$ MSWW ≤ 25. The total number of wing panels (NCWW $\times$ MSWW) must be less than or equal to 100. Some guidance in choosing a proper lattice arrangement for various shape wings is provided in reference 8. NCAMW specifies if wing camberline slopes are nonzero and must be input (NCAMW = 1) or are zero and need not be input (NCAMW = 0). The index NPSIW identifies straight leading edges and trailing edges (NPSIW = 0) which requires input of only one value for leading-edge and trailing-edge sweep angles, or broken leading and trailing edges (NPSIW = 1) which require input of the sweep angles at each spanwise station on the wing. NSEPW specifies the number of leading-edge separation vortices shed from the wing, and it must be either 1 or 2. If NSEPW = 2, a special value of YSEPW is required in Item 5.

The provision for multiple separation vortices is included to handle wings with breaks in leading-edge sweep. It has been observed that leading-edge separation vortices are shed from the wing regions inboard and outboard of the break. A maximum of two leading-edge separation vortices are allowed.

The next five indices in Item 1, NCWT, MSWT, NCAMT, NPSIT, and NSEPT are tail indices analogous to the previous five wing indices and subject to exactly the same restrictions. If no tail is present, all five must be zero.

The last index in Item 1, NBODY, determines whether or not the upwash field around the body is included (NBODY $\geq$ 0) in the wing and tail interference calculation or not included (NBODY < 0). This index is used to aid in determining the magnitude of the body-interference effect and generally should be set equal to zero.

Item 2 is a group of NHEAD cards containing identification information which is printed at the beginning of the output.

Item 3 is a group of NTBL cards describing the body shape. Each card contains an x-station, XBDY, and the corresponding body radius, RBDY. The cards should be in ascending order in x and there should be less than 100 cards in this item. The program internally sets up its own table of coordinates which is stored in the XBDY and RBDY arrays and is limited to 100 entries. A good rule of thumb to follow in inputting Item 3 is the following.

$$\text{NTBL} < (96 - \text{MSWW} - \text{MSWT})$$

Some care is required when describing the body shape via XBDY and RBDY. Linear interpolation is used throughout; therefore, where the body shape is changing rapidly, more points are required. There should be a minimum of five entries in the nose region ahead of the wing and there must be entries at x-stations identically equal to XWLE and XWTE, and XTLE and XTTE if a tail is present. The last entry in the table must be greater than XWTE or XTTE, whichever is greater, by an amount not less than DXI. If the body is made up of a nose section followed by a cylindrical afterbody, there should be two points on the cylinder very close together near the beginning of the cylinder. Points on a cylinder can be spaced large distances apart, but if the cylinder is followed by a section with changing radius, the last two points on the cylinder should be close together.

Item 4 consists of a single card containing XM and ZM, the coordinates of the center of moments, EMACH, the free-stream Mach number ($0 \leq$ EMACH < 1.0), and the reference area and reference length, REFS and REFL, respectively. THETAN is the nose semiapex angle in degrees (see fig. 2). The final two variables are associated with the free vortex trajectory calculations. DXOUT is the approximate increment in x at which trajectory coordinates are stored for use in induced velocity calculations. A lower limit for this variable is about 0.5 percent of the overall length of the body because of storage limitations. Typically, a reasonable value for DXOUT is about one half the maximum radius of the body. DXI is the initial integration interval for the trajectory calculations. The integration package will cut the interval in half if necessary for reasonable accuracy, and this halving process can occur ten times before the program automatically terminates execution with an appropriate message. If DXI is made too large, the program will stop because of unacceptable accuracy, and if DXI is made too small, the running time will become large. Experience has indicated that a value of DXI between 2 and 5 percent of the body length will work for most cases. Under rare circumstances when two vortices get very close together or when a vortex gets very near the wing or body, a smaller value of DXI may be required.

Item 5 contains geometric information for the wing. XLEW is the distance from the nose of the body to the intersection of the wing leading edge with the body. XTEW is the location of the trailing-edge intersection with the body. The wing hinge line at the wing-body juncture is located by the next two variables, XHLW and ZHLW. If an experimental center of pressure location is to be used for moment calculations, XCPW must contain the appropriate value. Otherwise, the program computes a center of pressure and XCPW must be identically zero. BS2W is the wing semispan measured from the centerline of the body. RAVGW is the average body radius in the vicinity of the wing. YSEPW is the y-station at which the wing is assumed broken for purposes of having two leading-edge separation vortices. If NSEPW = 1, then YSEPW must be equal to BS2W. If NSEPW = 2, YSEPW must be given some value greater than RAVGW and less than BS2W and the chosen value should coincide with one of the breaks in the lattice layout. That is, YSEPW will be equal to one of the values of YW to be described in Item 7. It is advised that there be at least three values of YW on either side of YSEPW to achieve reasonable accuracy in the separation vortex strength and position calculation.

Wing parameters are contined in Item 6. The first variable PHIW, is the dihedral angle in degrees for the entire wing. No breaks in dihedral are permitted. The second variable, PSIWLE(1), is the sweep angle, in degrees, of the leading edge at the wing-body juncture. If the leading edge has no breaks in sweep (NPSIW = 0), this value is the only sweep angle associated with the leading edge. PSITEW(1) is the sweep angle, in degrees, of the trailing edge at the wing-body juncture. If there are no breaks in sweep, it must be the trailing-edge sweep angle. Remember that a swept forward trailing edge has a negative sweep angle. The last variable in this item, CMTEST, is the convergence tolerance on pitching moment for a trim calculation (NTRIM > 0). A typical value for this quantity is about 1 percent of the magnitude of the untrimmed pitching moment. If it is made too small, the computer time required to converge to a trimmed solution can be very large.

Item 7 is a list of the spanwise locations, YW, of the outboard side of each column of vortices. These quantities are dimensional spanwise distances measured from the body centerline. There are MSWW values input. The last value must be equal to the wing semispan, BS2W. These spanwise columns forming the wing lattice need not be equally spaced, but for convenience in preparing input, it is quite acceptable to use an equal spacing along the wing.

Items 8 and 9 are optional and are included in the input deck only if there are breaks in sweep of the wing leading and trailing edges (NPSIW > 0) Item 8 includes the leading-edge sweep angle, in degrees, of each column of vortices from the wing-body juncture to the tip. If the wing sweep angle is continuously changing as in an ogee wing, the sweep angle at the center of each column of the lattice should be used.

Item 9 contains the wing trailing-edge sweep angle corresponding to the leading-edge angles in Item 8. If the trailing edge is unbroken and has constant sweep, the values must still be input even though they are all the same.

Item 10 is also optional and is included only if the wing is cambered or twisted (NCAMW > 0). If such is the case, ALPHLW, the tangent of the local camber angle α_ℓ of each element of the lattice is input. There are MSWW cards corresponding to the number of spanwise columns forming the lattice, one card for each column. The camber angles on each card run from leading edge to trailing edge with NCWW values per card. If there are more than eight chordwise rows, the ninth and tenth values follow on the next card. The ALPHLW value for the most forward area element in each column must start on a new card.

Values of ALPHLW are obtained as follows. Consider the sketch in figure 3 which shows the cambered and twisted section of the lifting surface at some spanwise station. At point P, corresponding to a control point on the wing mean surface, a tangent to the wing mean surface is constructed, which makes an angle α_ℓ with the wing root chord. The positive sense of α_ℓ is shown in this figure. The input value required is ALPHLW = tan α_ℓ. Near the leading edge of the section shown in figure 3, α_ℓ is negative. Item 10 completes the input description of the wing.

If a tail or aft lifting surface is not present on the configuration (MSWT = 0), the next portion of input is Item 17 which specifies angle of attack and other nongeometric-related parameters. If a tail is present (MSWT > 1), Items 11 through 16 are required input. These items specifying the tail geometry are analogous to the equivalent wing parameters in Items 5 through 10 and the rules and restrictions regarding preparation of tail input are the same as those described above for the wing.

Item 17 is a group of cards, one card for each run, which specifies the variables which are considered changeable for a given geometric configuration. The first entry on the card is the index, NDEX, which is simply used to control the stacking of additional cases. NDEX = 1 on each card represents a new angle of attack or incidence angle condition. If NDEX = 0, the card is ignored and the program returns to read in a new case beginning with Item 1. Thus, a blank card is used to separate different cases. When NDEX ≠ 0, the next value on the card is the configuration angle of attack in degrees, ALPHAD, taken as the angle between the axis of the body and the free-stream velocity. The second quantity is the incidence angle of the wing root chord in degrees, ALPIW. Its sense is such that a positive incidence is a leading edge up condition. The next three variables are the K_v^* factors which relate the actual realized vortex lift from the leading and side edges to that which is theoretically available. AKVLW1 is the fraction of leading-edge separation vortex lift which is obtained on the inboard portion of the wing if NSEPW = 2 or on the entire wing if NSEPW = 1. AKVLW1 is a number between zero and one and is generally geometry dependent. Its value can be obtained for sharp-edged delta wings from the correlation curves in figure 4. The source of figure 4 is described in detail in reference 1. The correlation curve can be used to get AKVLW1 for any swept wing, but since figure 4 was obtained for sharp-edged delta wings specifically, some judgement is necessary when other wings are considered. Instead of using the actual wing aspect ratio for nondelta wings, it is possible that some equivalent aspect ratio given by the delta wing expression

$$AR = \frac{4}{\tan \Lambda_{LE}} \tag{1}$$

would give a more reliable value. The factor is included as an input variable so that its effect can easily be examined by making a series of runs with AKVLW1 varied between zero and one.

If a wing is broken into two leading-edge vortex regions (NSEPW = 2), then AKVLW2 is the K^*_{VLE} factor which applies to the outboard portion of the wing. It is acceptable for AKVLW1 and AKVLW2 to be equal. In the case of wings with breaks in sweep, the appropriate values can be obtained from figure 4 using an effective aspect ratio in each region as calculated by equation (1). At the present stage in the development of program SUBSON, there is no reliable method of choosing the correct K^*_{VLE} factor. Many more data comparisons for double-delta wings or variable sweep wings should be made for this purpose.

For unswept leading edges, the vortex lift from the leading edge is usually small. In this case, the full amount of vortex lift should be retained and both AKVLW1 and AKVLW2 should be unity. For a wing with nonzero tip chord, the side-edge suction lift is generally very small compared to the potential lift except for very low-aspect-ratio wings. Comparisons with rectangular wing data indicate that the side-edge factor, AKVSW, should be unity at all times. When the tip chord is zero, such as on a delta wing, AKVSW should be identically zero.

The next four variables, ALPIT, AKVLT1, AKVLT2, and AKVST are the corresponding tail parameters. They fall under the same rules and guidelines set up for the respective wing parameters. If no tail is present, all four values should be set equal to zero.

The final quantity on this card is WLIMIT, the maximum allowable vortex-induced velocity nondimensionalized by free-stream velocity. The purpose of this variable is to limit the magnitude of the vortex-induced velocities on the wing or tail. In the course of program development, a canard-wing-body configuration developed a lift curve which appeared to exhibit a discontinuity around $\alpha = 8^{\circ}$. Close investigation showed that this occurred when the canard trailing-vortex trajectory abruptly changed from passing beneath the wing to passing over the wing. At this point, the vortex-induced velocities changed character rapidly and, because of the close proximity of the vortex to the wing, the velocities were large. This created very large local angles of attack on the wing, and the vortex-lattice scheme predicted large changes in wing loading. This, of course, is an unrealistic situation because a true viscous vortex does not behave as a potential vortex and induce infinite velocities at its center. For this reason a limit was introduced which arbitrarily sets any vortex-induced velocity greater than WLIMIT, equal to WLIMIT.

Generally, WLIMIT should be set equal to 1.0. If, in the process of running the program, unusual variations in the lift or pitching moment with angle of attack occur which can be attributed to unrealistic vortex-induced interference, WLIMIT can be used to limit the magnitude of the large induced velocities causing the problem. A value of WLIMIT = 0.1 has been used in some specific examples to reduce the apparent discontinuity in the predicted lift and moment curves and resulted in good agreement with experiment.

This discussion is not meant to suggest that an arbitrary velocity limit will cure the problems with the near flow fields of potential vortices. It is simply included to note that a simple, approximate fix is available. If WLIMIT = 0.0, the effect of the free vortices on the lifting surfaces is completely eliminated.

The above discussion includes all the input required for a typical run. The sample cases in the following section cover the options available in the program.

Sample cases.- Some sample cases are now presented to illustrate the preparation of input decks for various types of configurations. The airplane configurations chosen for these examples are the canard-wing-body combinations of reference 6 shown in figure 5 and the wing-body-tail configuration from reference 11 shown in figure 6.

In figure 7(a), the complete input deck for a canard-wing-body configuration is shown for sample case 1. The geometry corresponds to the sketch in figure 5 and this sample case considers the presence of both lifting surfaces. This series runs is for four angles of attack ($\alpha = 4^{\circ}$, 8°, 12°, and 16°) with no canard deflection ($\delta_w = 0^{\circ}$) and five angles of attack ($\alpha = 0^{\circ}$, 4°, 8°, 12°, and 16°) with positive canard deflection ($\delta_w = 10^{\circ}$). Sample case 1 requires approximately 720 seconds on the IBM 360/67 computer; however, this time is much less than would be required if each run were made individually. A single run of this type requires approximately 100 seconds.

The input deck for the second sample case is shown in figure 7(b). This deck is for the same configuration examined in the case above, but with the canard removed. In this case, the leading- and trailing-edge sweep angles are input at each spanwise station to illustrate the procedure for a wing with breaks in sweep.

Sample case 3 is the wing-body-tail combination shown in figure 6. This input deck specifies a minimum amount of output and a trim calculation with the tail incidence variable. Only one angle of attack is specified because of the uncertainty in the amount of time required to converge on a trimmed solution. This particular run requires approximately 100 seconds on the IBM 360/67. Note that the vortex-induced velocity is limited to 0.1 by the variable WLIMIT on the last card. This was necessary because of the large effect the wing shed vorticity had on the tail loading. The relative position between the trailing vortices and the tail was such that small changes in tail angle resulted in large nonlinear changes in tail loading which prevented convergence on a trim condition.

Description of Output

The output produced by the SUBSON computer program for sample case 1 is shown in figure 8. The first page of output from the program, figure 8(a is a tabulation of the input data in Items 1 and 2 of figure 1. The next page of output, figure 8(b), is a summary of the geometry of the configuration by component. The first quantities at the top of the page are the first angle of attack to be considered, the Mach number, and beta. The next two items are the reference area and the reference length from Item 4 of figure 1. If the wing and /or tail have alternate center-of-pressure locations input in Items 5 and 11 of figure 1, these are printed here. If a trim calculation is requested (NTRIM > 0), the value of CMTEST specified in Item 6 of figure 1 is printed here. These are followed by the geometry of the wing.

The wing quantities which are tabulated are:

OUTPUT NOTATION	ALGEBRAIC NOTATION
XLE	$x_{LE_{root}}$
XTE	$x_{TE_{root}}$
B/2	$b/2$
XHL	x_{HL}
ZHL	z_{HL}
C(ROOT)	c_{root}
C(TIP)	c_{tip}
Y(SEP)	YSEPW (Item 5)

The following block of data contains the wing dihedral angle in degrees and the leading-edge and trailing-edge sweep angles at the various y stations. The first entry denoted Y(RT) represents the wing-body juncture and the corresponding initial sweep angles of the wing leading and trailing edges. The following entries under the heading Y(WING) represent the y stations defining the spanwise lattice layout on the wing. The sweep angles at these stations are noted in the next two columns.

The same quantities are tabulated for the tail surface if one is present. The following quantities are listed for the body.

OUTPUT NOTATION	ALGEBRAIC NOTATION
THETA	θ_N (Item 4)
FINENESS	x_{LE_w}/r_N
R(BASE)	r_N
AVERAGE RADIUS WING	r_{avg_w}
AVERAGE RADIUS TAIL	r_{avg_e}
CENTER OF MOMENTS X	x_m
CENTER OF MOMENTS Z	z_m
DXI	Δx } Item 4
DXOUT	- } Item 4
X	x
R	r
S	s_w or s_e
DR/DX	dr/dx

This concludes the general geometric description of the configuration. This information is output once at the beginning of each case. The following output is dependent on the information input on each card of Item 17; that is, the angle of attack, incidence angles, K_v^* factors, and induced velocity limit.

Figure 8(c) is the first page of output for each run within the series of runs making up sample case 1. The first line summarizes the information input in Item 17 as follows:

OUTPUT NOTATION	ALGEBRAIC NOTATION
ALPHA	α
M	M

OUTPUT NOTATION	ALGEBRAIC NOTATION
INCIDENCE WING	δ_w
INCIDENCE TAIL	δ_e
WING KVLE*	$K^*_{v_{LE_w}}$
WING KVSE*	$K^*_{v_{SE_w}}$
TAIL KVLE*	$K^*_{v_{LE_e}}$
TAIL KVSE*	$K^*_{v_{SE_e}}$
W/V LIMIT	$(w/V)_{max}$

The next block of output on this page is a summary of the strength and position of the right-hand vortex (if present) of the symmetrical pair of vortices shed from the nose of the body. The last entry, XS/RB, is the body-vortex separation location.

OUTPUT NOTATION	ALGEBRAIC NOTATION
GAM/2*PI*V*RB	$\frac{\Gamma_B}{2\pi V r_N}$
Y/RB	y_B/r_N
Z/RB	z_B/r_N
XS/RB	x_s/r_N

The following block of output is the induced Beskin upwash at the wing control points (x,y,z) due to the presence of the body. The induced velocities, V/V(INF) and W/V(INF), expressed as a fraction of the free-stream velocity, are positive in the positive y- and z-directions, respectively. A summary of the nose-vortex position at the wing leading edge follows. The next block of data are the velocities induced at the same wing control points by the vortex pair shed from the nose of the body. These velocities have the same positive sense as the body-induced upwash above. The final block of data in figure 8(c) is the total induced velocity at each control point.

The next page of output, figure 8(d), contains the results from the lifting-surface calculations for the wing in a wing-alone coordinate system. Under FLOW CONDITIONS, the angle of attack is the incompressible angle of attack of the wing, including incidence. The next printed information is the lattice layout followed by the heading REFERENCE QUANTITIES. Under this heading, the actual exposed planform area of the wing and the average chord are listed. The aerodynamic coefficients on this page are based on these reference quantities.

The following block of information contains wing geometry for the wing alone. If the Mach number is nonzero, the geometry is for the wing in the incompressible plane.

The last half of this page contains the predicted aerodynamic characteristics of the wing in the presence of the body and other external interference velocity fields. Most of these quantities are self-explanatory and will not be described herein. It should be noted that in the coordinate system for the wing alone, x_W is measured from the leading edge of the root chord, positive forward. The same positive direction is taken for all x_W direction coefficients. A few of the more important coefficients are defined as follows:

OUTPUT NOTATION	ALGEBRAIC NOTATION
CNP	$C_{N_{W(B),p}}$
CNV	$C_{N_{W(B),v_{LE}}}$
CNVS	$C_{N_{W(B),v_{SE}}}$
CL*C/(2*B)	$\frac{cc_\ell}{2b}$
CSUC*C/(2*B)	$\frac{cc_s}{2b}$

OUTPUT NOTATION	ALGEBRAIC NOTATION
KVLE	$K_{V_{LE}}$
KVSE	$K_{V_{SE}}$
GAM/V2PI	$\frac{\Gamma}{2\pi V}$

The last line of figure 8(d) contains the predicted leading-edge and side-edge vortex-lift constants, $K_{V_{LE}}$ and $K_{V_{SE}}$, respectively. Since these values include effects of external interference on the wing, they are useful only in cases of wings alone with no interference effects included. The last two items are the lateral position and strength of the trailing vortex and the leading-edge separation vortex at the wing trailing edge.

Figure 8(e) is headed by a summary of the strengths and positions of the vortices shed from the configuration ahead of the wing trailing edge. The pairs of vortices are listed in the following order. Vortex 1 is the right-side body vortex shed from the nose. Vortex 2 is the trailing vortex shed from the wing. Vortex 3 is the leading-edge separation vortex shed from the wing. If more than one separation vortex is requested, vortex 3 is the vortex associated with the inboard region and vortex 4 is shed from the outboard region. If a vortex is missing for any reason, all following vortices are moved up in the table. For example, if no vortices are shed by the nose, vortex 1 becomes the trailing vortex shed by the wing, and so on. The remainder of figure 8(e) is the induced velocities at the tail control points. These velocities are analogous to the induced velocities on the wing shown in figure 8(c).

Figure 8(f) contains calculated results for the tail surface. All the quantities on this page are analogous to those described for the wing in figure 8(d). The last entry on this figure is a summary of the strengths and positions of all the vortices in the field just aft of the tail trailing edge. The first group of vortices are the same as described in connection with figure 8(e). The second group of vortices are defined as follows. Vortex 4 is the trailing vortex corresponding to the potential lift on the tail. Vortex 5 is the leading-edge separation vortex shed from the tail. If multiple vortices are shed from the tail leading edge, this vortex would be shed from the inboard tail region and vortex 6 would be shed from the outboard tail region.

The next page of output, figure 8(g), is a summary page of the force coefficients, pitching-moment coefficients, and centers of pressure of each component of the configuration and of the total configuration. The coefficients for the individual components are described in Table I. The total configuration variables are defined as follows:

OUTPUT NOTATION	ALGEBRAIC NOTATION
CN	$C_N = \frac{N}{qS}$
CM	$C_m = \frac{M}{qS\ell}$
XCP	$\bar{x}_{CP} = x_m - \frac{C_m}{C_N}\ell$
CL	$C_L = \frac{L}{qS}$
CDI	$C_{D_i} = C_L \tan\alpha$
CA	$C_A = \frac{A}{qS}$
CDI/CL**2	C_{D_i}/C_L^2

The last page of output for this run, figure 8(h), contains a summary of the trajectories of the shed vortices. At the top of the page the vortices are identified and their strengths listed. This is followed by blocks of output, one block for each x-station, describing the local crossflow geometry of the configuration and the position of the right-side vortices. The x stations of each block of results are approximately DXOUT apart. Notice that the trajectory calculation starts at the wing leading edge with a pair of body vortices. As the calculation moves downstream, other vortices are shed and added to the calculation. The trajectory calculation is carried downstream to a point aft of the tail trailing edge. For purposes of saving space in figure 8, only selected portions of the trajectory calculation is presented herein. The variables in each block are defined as follows:

OUTPUT NOTATION	ALGEBRAIC NOTATION
X	x
DX	Δx
A	a
S	s_w or s_e
RO	r_o, transformed circle radius
DA/DX	da/dx
SIGMA(REAL)	y
SIGMA(IMAG)	z

This completes the output for one card in Item 17 of the input deck. Additional runs will repeat the output of figures 8(c) through (h). The above set of output obtained with NPRINT = 0 is a considerable amount of output for production runs; therefore, an optional set of output can be obtained by setting NPRINT = -1. In this case, the complete output consists of figures 8(a), (b), and (g), with some shed vortex positions and strengths added.

Some extra output over and above that shown in figure 8 can be obtained when NPRINT = 10. This additional output is useful only for diagnostic purposes and is not described herein. This output is labeled and the user should have no trouble interpreting the results.

Program Listing

The SUBSON computer program consists of the main program and fifteen subroutine subprograms. Each source deck is identified in columns 73 through 80 by a four-character identification and a three-digit number sequencing the cards within that deck. The program listing is given on the following pages. The table below will act as a table of contents for the listing.

```
C         SUBSONIC PREDICTION METHOD
C                                              M.R. MENDENHALL     200/C
C
      COMPLEX SP,VP,SIGMA,EYE
C
      DIMENSION TITLE(20),GAM(4),YG(4),ZG(4),Y1(5),GAM1(2)
      DIMENSION XGAMN(10),YGAMN(10),ZGAMN(10),GAMN(10)
      DIMENSION XBDY(100),RBDY(100),BBDY(100),DRBDY(100)
      DIMENSION XV(200),YV(4,200),ZV(4,200),NVP(200),ARAY(5,200)
      DIMENSION XP(100),SP(100),VP(100),ALPHLW(100),ALPHLT(100),
     1          UEI(100),VEI(100),WEI(100),VEW(100),WEW(100)
      DIMENSION WB(100),VB(100)
      DIMENSION YW(25),PSILEW(25),PSITEW(25),YT(25),PSILET(25),
     1          PSITET(25)
      DIMENSION FVNW(100,102),FVNT(100,102),FVN(100,102)
      DIMENSION SUMYW(25),YLOCW(25),XLELW(25),ZLELW(25),CHLCLW(25),
     1          CHLCBW(25)
      DIMENSION SUMYT(25),YLOCT(25),XLELT(25),ZLELT(25),CHLCLT(25),
     1          CHLCBT(25)
      DIMENSION SWPVPW(100),SWW(100),PVXW(100),PVYW(100),PVZW(100),
     1          PCXW(100),PTLXW(100),PTLYW(100),PTLZW(100)
      DIMENSION SWPVPT(100),SWT(100),PVXT(100),PVYT(100),PVZT(100),
     1          PCXT(100),PTLXT(100),PTLYT(100),PTLZT(100)
      DIMENSION FUWW(2),FVWW(2),FWWW(2)
      DIMENSION FUWT(2),FVWT(2),FWWT(2),SIGMA(2),HEAD(9)
C
      COMMON /TRAJ1/ XI,XF,DXI,TITL,NPRNT
      COMMON /TRAJ2/ SP,VP,XP,NEXT
      COMMON /BODY/ NTBL,MTBL,XBDY,RBDY,BBDY,DRBDY
      COMMON /VORTEX/ NV,NV2,GAM,YG,ZG
      COMMON /VTABLE/ NTRY,XV,YV,ZV,NVP,ARAY
      COMMON /FLOW/ ALPHA,CSALP,EYE
      COMMON /PARAM/ DXOUT,XPRNT,MPRINT,XFINAL
      COMMON /VLS/ NPRINT,MMM,CNORM,CNV,CNVS,XCOPP,XCOPLE,XCOPSE,
     1             YTR,GAMT,Y1,GAM1,CNV1,CNV2,NSEP
      COMMON /B/ MDUM,M
C
      DATA HEAD/4HWING,4HTAIL,4HW(B),4HB(W),4HT(B),4HB(T),4HNOSE,
     1          4HBODY,4H
C
  700 FORMAT (20A4)
  701 FORMAT (15I5,I3,I2)
  702 FORMAT (5X,20A4)
  703 FORMAT (1H0)
  704 FORMAT (8F10.5)
  705 FORMAT (1H1)
  706 FORMAT(//5X24HCONFIGURATION GEOMETRY -//5X18HREFERENCE AREA   =,
     1          F9.3,/ 5X,18HREFERENCE LENGTH = ,F9.3)
  707 FORMAT(1H1,5X17HFLOW CONDITIONS -,2X5HALPHA,6X4HMACH,6X4HBETA/20X,
     1          F10.2,2F10.4)
  708 FORMAT (I10,4F10.5)
  709 FORMAT (//5X25HTABLE OF BODY COORDINATES/15X,1HX,9X1HR,9X1HB,
     1          9X5HDR/DX)
  710 FORMAT (18X,10H VORTEX    ,4X,5(1PE12.3))
  711 FORMAT (1H0,5X47HSUMMARY OF VORTEX STRENGTHS AND POSITION AT X = ,
     1          F8.3)
  712 FORMAT (/10X6HVORTEX,3X12HGAMMA/2*PI*V,6X1HY,9X1HZ,
     1          4(/12X,I2,6XF10.6,2F10.3))
  713 FORMAT (5X49HSUMMARY OF FORCE AND PITCHING MOMENT COEFFICIENTS //
     1          37X,2HCN,10X2HCM,10X3HXCP,9X2HCL,10X3HCDI,9X2HCA)
  714 FORMAT (/10X,A4,3H... )
  715 FORMAT (14X,A4,10H POTENTIAL ,4X,6(1PE12.3))
  716 FORMAT (14X,A4,10H VORTEX,LE ,4X,6(1PE12.3))
  717 FORMAT (14X,A4,10H VORTEX,SE ,4X,6(1PE12.3))
  718 FORMAT (/10X,12HAFTERBODY... ,10X, 6(1PE12.3))
  719 FORMAT (//10X,22HTOTAL CONFIGURATION... , 6(1PE12.3))
  720 FORMAT (1H1,5X,35HSUMMARY OF FREE VORTEX TRAJECTORIES //)
  721 FORMAT (//5X1HX,8X2HDX,11X1HA,9X1HB,8X2HR0,7X5HDA/DX)
  722 FORMAT (F9.4,F11.7,4F10.4)
  723 FORMAT (/1X6HVORTEX,1X11HSIGMA(REAL),3X6H(IMAG))
  724 FORMAT (I4,2X,4(1PE12.3,E11.3))
  725 FORMAT (/6X6HVORTEX,3X12HGAMMA/2*PI*V)
  726 FORMAT (I10,5X,F10.6)
  727 FORMAT (10X6HVORTEX,I2,28H ---  BODY VORTEX FROM NOSE )
  728 FORMAT (10X6HVORTEX,I2,28H ---  WING TRAILING VORTEX  )
  729 FORMAT (10X6HVORTEX,I2,28H ---  WING SEPARATION VORTEX)
  730 FORMAT(//5X,A4,4H ....7X3HXLE,7X3HXTE,7X3HB/2,7X3HXHL,7X5HZHL6X7HC
     1(ROOT),3X6HL(TIP),4X6HY(SEP)/15X,8F10.4)
  731 FORMAT(42X11HSWEEP ANGLE/1HX9H DIHEDRAL,2X8H Y(HT)  ,5X2HLE,8X2HTE
     1            /15X,6F10.3)
  732 FORMAT(29X2HY(,A4,1H),6X2HLE,8X2HTE)
  733 FORMAT(25X,3F10.3)
  734 FORMAT(//5X8HBODY ...  ,6X27H----------- NOSE -----------,5X14HAVER
     1AGE RADIUS,4X17HCENTER OF MOMENTS/20X5HTHETA,3X8HFINENESS,3X7HR(BA
     2SE),5X4HWING,6X4HTAIL,7X1HX,9X1HZ,9X5HDXI,5X5HDXOUT/15X9F10.3)
  735 FORMAT(//5X30HNOSE VORTICES -  GAM/2*PI*V*RB3X4HY/RB,6X4HZ/RB,
     15X5HXS/RB/22XF10.4,3F10.3)
  736 FORMAT(1H1,5X,A4)
  737 FORMAT (30X15H-- INCIDENCE -- ,5X15H---- WING ----- ,
     1        5X15H---- TAIL ----- / 10X5HALPHA,8X1HW,7X4HWING,6X4HTAIL,
     2        2(5X5HKVLE*,5X5HKVSE*),3X9HW/V LIMIT /5X2F10.2,7F10.3)
  738 FORMAT (5X,A4,14H X(CP)        = ,F9.3)
  739 FORMAT (5X,18HTRIM (DELTA CM)  = ,F9.5)
  740 FORMAT (///10X,35H*** TRIM CONFIGURATION ATTAINED *** )
  741 FORMAT (///10X,36H*** TRIM CONFIGURATION NOT FOUND *** )
  742 FORMAT (/35X12HCDI/CL**2 = ,1PE12.3)
  745 FORMAT (///5X46HVELOCITY INDUCED AT SPECIFIED FIELD POINTS ON ,A4,
     1           9H (BESKIN) )
  746 FORMAT (/ 10X1HX,9X1HY,9X1HZ,11X8HV/V(INF),4X8HW/V(INF))
  747 FORMAT (I4,3F10.3,5X,2(1PE12.4))
  748 FORMAT (//5X64H*** NOTE - VORTEX INDUCED VELOCITIES ARE NEGLECTED
     1, W/V LIMIT =,F10.4)
  749 FORMAT (///5X52HTOTAL INDUCED VELOCITY AT SPECIFIED FIELD POINTS (O
     1N , A4)
  750 FORMAT (I2,F6.2,9F8.4)
  751 FORMAT (45XF10.3,10X,F10.3)
  752 FORMAT (12X,I2,6XF10.6,2F10.3)
  798 FORMAT (//10X 63H*** NOSE VORTEX POSITION ERROR, ELIMINATED FROM C
     1ALCULATION ***  //)
  799 FORMAT (//5X40H***  EXECUTION STOP, BODY TABLE ERROR     )
C
      NAMELIST /CARD1/ NHEAD,NTBL,NPRINT,MPRINT,NOSEV,NTRIM,NCWW,MSWW,
     1                 NCAWW,NPSIW,NSEPW,NCWT,MSWT,NCAWT,NPSIT,NSEPT,NBODY
C
      EYE=CMPLX(0.0,1.0)
      MTBL=1
      NTIME=1
      NFLAG=0
      CM2=0.0
      AL2=0.0
      NMX=100
      NMY=102
      NZ=25
      NVMAX=4
      PI=3.1415926
      RAD=180./3.1415926
   10 READ (5,701) NHEAD,NTBL,NPRINT,MPRINT,NOSEV,NTRIM,NCWW,MSWW,NCAWW,
     1  NPSIW,NSEPW,NCWT,MSWT,NCAWT,NPSIT,NSEPT,NBODY
   98 CONTINUE
      NPRNT=NPRINT
C
C     INITIALIZE
C
      IF (MSWT.LE.0 .AND. NTRIM.GE.2) NTRIM=0
      NOSE=NOSEV
      DO 3008 J=1,NMX
      UEI(J)=0.0
      VEI(J)=0.0
      WEI(J)=0.0
      VEW(J)=0.0
      WEW(J)=0.0
      ALPHLT(J)=0.0
 3008 ALPHLW(J)=0.0
      DO 9 J=1,NVMAX
      YG(J)=0.0
      ZG(J)=0.0
    9 NVP(J)=0
      YW(1)=0.0
      YT(1)=0.0
      CNNP  =0.0
      CNNV  =0.0
      CNWBP =0.0
      CNWBVS=0.0
      CNWBVL=0.0
      CNBWP =0.0
      CNBWV =0.0
```

```fortran
      CNBWVS=0.0
      CNA   =0.0
      CNTBP =0.0
      CNTBVL=0.0
      CNTBVS=0.0
      CNBTP =0.0
      CNBTV =0.0
      CNBTVS=0.0
      CNWP  =0.0
      CNWV  =0.0
      CNWBP =0.0
      CNWBVS=0.0
      CNWBVL=0.0
      CNBWP =0.0
      CNBWV =0.0
      CNBWVS=0.0
      CMA   =0.0
      CMTBP =0.0
      CMTBVL=0.0
      CMTBVS=0.0
      CMBTP =0.0
      CMBTV =0.0
      CMBTVS=0.0
      XCPNP =0.0
      XCPNV =0.0
      XCPWBP=0.0
      XCPBWP=0.0
      XCPBWV=0.0
      XCPBWS=0.0
      XCPA  =0.0
      XCPTBP=0.0
      XCPBTP=0.0
      XCPBTV=0.0
      XCPBTS=0.0
      CAWBP =0.0
      CAWBVL=0.0
      CAWBVS=0.0
      CATBP =0.0
      CATBVL=0.0
      CATBVS=0.0
      XCPT=0.0
      RAVGT=0.0
      ALPIT=0.0
      WRITE (6,705)
      WRITE (6,CARD1)
      WRITE(6,703)
C
      DO 11 J=1,NHEAD
      READ (5,700) TITLE
   11 WRITE (6,702) TITLE
      DO 12 J=1,NTBL
   12 READ (5,704) XBDY(J),RBDY(J)
      READ (5,704) XM,ZM,EMACH,REFS,REFL,THETAN,DXOUT,DXI
      READ (5,704) XLEW,XTEW,XMLW,ZMLW,XCPW ,BS2W,RAVGW,YSEPW,
     1       PHIW,PSILEW(1),PSITEW(1),CMTEST
      IF (CMTEST.LE.0.0) CMTEST=0.01
      CRWW=XTEW-XLEW
      IMAX=MSWW+1
      MW=MSWW+NCWW
      READ(5,704) (YW(I),I=2,IMAX)
      IF (NPSIW.GT.0) READ (5,704) (PSILEW(I),I=2,IMAX)
      IF (NPSIW.GT.0) READ (5,704) (PSITEW(I),I=2,IMAX)
      IF (NPSIW.GT.0) GO TO 109
      DO 108 I=2,IMAX
      PSILEW(I)=PSILEW(1)
  108 PSITEW(I)=PSITEW(1)
C
C      INPUT TANGENT OF LOCAL WING ANGLES DUE TO CAMBER AND TWIST
C          (NO INCIDENCE SHOULD BE INCLUDED)
C
  109 IF (NCAMW.EQ.0) GO TO 1001
      MN=0
      DO 9910 JNW=1,MW,NCWW
      MN=MN+NCWW
 9910 READ (5,704) (ALPHLW(J),J=JNW,MN)
 1001 CONTINUE
      IF (MSWT.LE.0) GO TO 20
      READ (5,704) XLET,XTET,XMLT,ZMLT,XCPT,BS2T,RAVGT,YSEPT,
     1       PHIT,PSILET(1),PSITET(1)
      CRWT=XTET-XLET
      IMAXT=MSWT+1
      MT=MSWT+NCWT
      READ (5,704) (YT(I),I=2,IMAXT)
      IF (NPSIT.GT.0) READ (5,704) (PSILET(I),I=2,IMAXT)
      IF (NPSIT.GT.0) READ (5,704) (PSITET(I),I=2,IMAXT)
      IF (NPSIT.GT.0) GO TO 120
      DO 118 I=2,IMAXT
      PSILET(I)=PSILET(1)
  118 PSITET(I)=PSITET(1)
C
C      INPUT TANGENT OF LOCAL TAIL ANGLES DUE TO CAMBER AND TWIST
C          (NO INCIDENCE SHOULD BE INCLUDED)
C
  120 IF (NCAMT.EQ.0) GO TO 20
      MN=0
      DO 9911 JNW=1,MT,NCWT
      MN=MN+NCWT
 9911 READ (5,704) (ALPHLT(J),J=JNW,MN)
   20 CONTINUE
C
      READ (5,750) NDEX,ALPHAD,ALPIW,AKVLW1,AKVLW2,AKVSW,
     1                     ALPIT,AKVLT1,AKVLT2,AKVST,WLIMIT
      BETA=SQRT(1.0-EMACH*EMACH)
C
C      LAY OUT WING VORTEX LATTICE GEOMETRY
C
      SSPANW=BS2W-RAVGW
      ARAY(1,1)=PSILEW(1)
      ARAY(2,1)=PSITEW(1)
      DO 22 J=2,IMAX
      ARAY(1,J)=PSILEW(J)
      ARAY(2,J)=PSITEW(J)
   22 YW(J)=YW(J)-RAVGW
      IF (EMACH.LE.0.0) GO TO 221
      CRWW=CRWW/BETA
      DO 21 I=1,IMAX
      PSILEW(I)=ATAN(TAN(PSILEW(I)/RAD)/BETA)*RAD
   21 PSITEW(I)=ATAN(TAN(PSITEW(I)/RAD)/BETA)*RAD
      IF (NCAMW.EQ.0) GO TO 221
      DO 121 I=1,MW
  121 ALPHLW(I)=ALPHLW(I)*BETA
  221 CONTINUE
      ALPHAW=0.0
      CALL LATTUS (FVNW,NWX,NWY,NZ,NCWW,MSWW,CRWW,CTWW,SSPANW,ALPHAW,
     1             PSILEW,PSITEW,PHIW,SREFW,CAVEW, SUMYW,YLOCW,YW,
     2             XLELW,ZLELW,CHLCLW,CHLCBW,SWPVPW,SWW,
     3             PVXW,PVYW,PVZW,PCXW,PTLXW,PTLYW,PTLZW,FUWW,FVWW,FWWW)
      CRWW=CRWW*BETA
      CTWW=CTWW*BETA
      IF (DXI.GT.1.0E-03) GO TO 541
      DXI=0.01*CRWW
  541 IF (DXOUT.LT.DXI) DXOUT=DXI
C
C      OUTPUT WING CHARACTERISTICS
C
      WRITE(6,707)ALPHAD,EMACH,BETA
      IF (WLIMIT.LE.0.0) WRITE (6,748) WLIMIT
      IF (WLIMIT.LT.0.0) WLIMIT=0.0
      WRITE(6,706)REFS,REFL
      IF (XCPW.GT.0.0) WRITE (6,738) HEAD(1),XCPW
      IF (XCPT.GT.0.0 .AND. MSWT.GT.0) WRITE (6,738) HEAD(2),XCPT
      IF (NTRIM.GT.0) WRITE (6,739) CMTEST
      WRITE(6,730)HEAD(1),XLEW,XTEW,BS2W,XMLW,ZMLW,CRWW,CTWW,YSEPW
      WRITE (6,731) PHIW,RAVGW,ARAY(1,1),ARAY(2,1)
      WRITE(6,732)HEAD(1)
      DO 123 J=2,IMAX
      DUM=YW(J)+RAVGW
  123 WRITE (6,733) DUM,(ARAY(N,J),N=1,2)
      IF (MSWT.LE.0) GO TO 29
C
C       LAY OUT TAIL VORTEX LATTICE GEOMETRY
C
      SSPANT=BS2T-RAVGT
      ARAY(1,1)=PSILET(1)
```

```
      ARAY(2,1)=PSITET(1)
      DO 23 J=2,IMAXT
      ARAY(1,J)=PSILET(J)
      ARAY(2,J)=PSITET(J)
   23 YT(J)=YT(J)-RAVGT
      IF (EMACH.LE.0.0) GO TO 224
      CRWT=CRWT/BETA
      DO 24 I=1,IMAXT
      PSILET(I)=ATAN(TAN(PSILET(I)/RAD)/BETA)*RAD
   24 PSITET(I)=ATAN(TAN(PSITET(I)/RAD)/BETA)*RAD
      IF (NCAMT.EQ.0) GO TO 224
      DO 124 I=1,MT
  124 ALPHLT(I)=ALPHLT(I)*BETA
  224 CONTINUE
      ALPHAT=0.0
      CALL LATTUS (FVNT,NMX,NMY,NZ,NCWT,MSWT,CRWT,CTWT,SSPANT,ALPHAT,
     1             PSILET,PSITET,PHIT,SREFT,CAVET,SUMYT,YLOCT,YT,
     2             XLELT,ZLELT,CMLCLT,CMLCBT,BWPVPT,SWT,
     3             PVXT,PVYT,PVZT,PCXT,PTLXT,PTLYT,PTLZT,FUWT,FVWT,FWWT)
      CRWT=CRWT*BETA
      CTWT=CTWT*BETA
C
C     OUTPUT TAIL CHARACTERISTICS
C
      WRITE(6,730)HEAD(2),XLET,XTET,BB2T,XMLT,ZMLT,CRWT,CTWT,YSEPT
      WRITE (6,731) PHIT,RAVGT,ARAY(1,1),ARAY(2,1)
      WRITE(6,732)HEAD(2)
      DO 225 J=2,IMAXT
      DUM=YT(J)+RAVGT
  225 WRITE(6,733)DUM,(ARAY(N,J),N=1,2)
      GO TO 28
   29 XLET=XTEW+100.
   28 CONTINUE
C
C     SET UP TABLE OF BODY, WING, AND TAIL COORDINATES FOR USE IN
C         VORTEX TRAJECTORY CALCULATIONS
C
      NT=1
      J=1
   30 CONTINUE
      XV(J)=XBDY(NT)
      YV(1,J)=RBDY(NT)
      ZV(1,J)=RBDY(NT)
      IF (XV(J).GT.XLEW) GO TO 31
   32 NT=NT+1
      J=J+1
      IF (NT.LE.NTBL) GO TO 30
      GO TO 40
   31 IF (XV(J).GT.XTEW .AND. XV(J).LE.XLET) GO TO 32
      IF (XV(J).GT.XLET) GO TO 35
      IMIN=2
      IF (PSILEW(1).GT.1.0E-02) GO TO 135
      IMIN=IMAX
      XV(J-1)=XV(J-1)-.001
  135 DO 33 I=IMIN,IMAX
      IF (J .GE. 100) GO TO 40
      XV(J)=XLEW-XLELW(I)*BETA
      ZV(1,J)=RAVGW+YW(I)
  133 IF (XV(J)-XBDY(NT)) 233,333,433
  333 YV(1,J)=RBDY(NT)
      NT=NT+1
      GO TO 33
  433 NT=NT+1
      IF (NT.LE.NTBL) GO TO 133
      WRITE(6,799)
      WRITE(6,708) J,XV(J)
      J=433
      WRITE (6,708) J
      STOP
  233 DELTA=(XBDY(NT)-XV(J))/(XBDY(NT)-XBDY(NT-1))
      YV(1,J)=RBDY(NT)-DELTA*(RBDY(NT)-RBDY(NT-1))
   33 J=J+1
      IF (CTW.LT.(0.02*CRW)) GO TO 34
      XV(J)=XV(J-1)+CTW
      ZV(1,J)=RAVGW + SSPANW
  533 IF (XV(J)-XBDY(NT)) 633,933,833
  933 YV(1,J)=RBDY(NT)
      J=J+1
      IF (J .GE. 100) GO TO 40
      GO TO 34
  833 NT=NT+1
      IF (NT.LE.NTBL) GO TO 533
      WRITE (6,799)
      WRITE (6,708) J,XV(J)
      J=833
      WRITE (6,708) J
      STOP
  633 DELTA=(XBDY(NT)-XV(J))/(XBDY(NT)-XBDY(NT-1))
      YV(1,J)=RBDY(NT)-DELTA*(RBDY(NT)-RBDY(NT-1))
      J=J+1
      IF (J .GE. 100) GO TO 40
   34 XV(J)=XTEW + .001
      IF (XV(J).LE.XV(J-1)) XV(J)=XV(J-1)+0.001
      XINTE=XV(J)
  134 IF (XV(J)-XBDY(NT)) 234,334,434
  334 YV(1,J)=RBDY(NT)
      ZV(1,J)=RBDY(NT)
      NT=NT+1
      GO TO 934
  434 NT=NT+1
      IF (NT.LE.NTBL) GO TO 134
      WRITE (6,799)
      WRITE(6,708)J,XV(J)
      J=434
      WRITE (6,708) J
      STOP
  234 DELTA=(XBDY(NT)-XV(J))/(XBDY(NT)-XBDY(NT-1))
      YV(1,J)=RBDY(NT)-DELTA*(RBDY(NT)-RBDY(NT-1))
      ZV(1,J)=YV(1,J)
  934 J=J+1
      IF (J .GE. 100) GO TO 40
      GO TO 30
   35 IMIN=2
      IF (PSILET(1).GT. 1.0E-02) GO TO 138
      IMIN=IMAXT
      XV(J-1)=XV(J-1)-.001
  138 DO 36 I=IMIN,IMAXT
      IF (J .GE. 100) GO TO 40
      XV(J)=XLET-XLELT(I)*BETA
      ZV(1,J)=RAVGT + YT(I)
  136 IF (XV(J)-XBDY(NT)) 236,336,436
  336 YV(1,J)=RBDY(NT)
      NT=NT+1
      GO TO 36
  436 NT=NT+1
      IF (NT.LE.NTBL) GO TO 136
      WRITE(6,799)
      WRITE(6,708) J,XV(J)
      J=436
      WRITE (6,708) J
      STOP
  236 DELTA=(XBDY(NT)-XV(J))/(XBDY(NT)-XBDY(NT-1))
      YV(1,J)=RBDY(NT)-DELTA*(RBDY(NT)-RBDY(NT-1))
   36 J=J+1
      IF (CTWT.LT.(0.02*CRWT))GO TO 37
      XV(J)=XV(J-1)+CTWT
      ZV(1,J)=RAVGT+SSPANT
  536 IF (XV(J)-XBDY(NT)) 636,936,836
  936 YV(1,J)=RBDY(NT)
      J=J+1
      IF (J .GE. 100) GO TO 40
      GO TO 37
  836 NT=NT+1
      IF (NT.LE.NTBL) GO TO 536
      WRITE(6,799)
      WRITE(6,708) J,XV(J)
      J=836
      WRITE (6,708) J
      STOP
  636 DELTA=(XBDY(NT)-XV(J))/(XBDY(NT)-XBDY(NT-1))
      YV(1,J)=RBDY(NT)-DELTA*(RBDY(NT)-RBDY(NT-1))
      J=J+1
      IF (J .GE. 100) GO TO 40
   37 XV(J)=XTET + 0.001
```

```
      IF (XV(J).LE.XV(J-1)) XV(J)=XV(J-1)+0.001
  137 IF (XV(J)-XBDY(NT)) 237,337,437
  337 YV(1,J)=RBDY(NT)
      ZV(1,J)=RBDY(NT)
      NT=NT+1
      GO TO 38
  437 NT=NT+1
      IF (NT.LE.NTBL) GO TO 137
      WRITE(6,799)
      WRITE(6,708) J,XV(J)
      J=437
      WRITE (6,706) J
      STOP
  237 DELTA=(XBDY(NT)-XV(J))/(XBDY(NT)-XBDY(NT-1))
      YV(1,J)=RBDY(NT)-DELTA*(RBDY(NT)-RBDY(NT-1))
      ZV(1,J)=YV(1,J)
   38 J=J+1
      IF (J .GE. 100) GO TO 40
      DO 39 I=NT,NTBL
      XV(J)=XBDY(I)
      YV(1,J)=RBDY(I)
      ZV(1,J)=RBDY(I)
      J=J+1
      IF (J .GE. 100) GO TO 40
   39 CONTINUE
   40 CONTINUE
      NTBL=J-1
      DO 41 J=1,NTBL
      XBDY(J)=XV(J)
      RBDY(J)=YV(1,J)
      SBDY(J)=ZV(1,J)
      XV(J)=0.0
      YV(1,J)=0.0
   41 ZV(1,J)=0.0
      XBDY(NTBL+1)=XBDY(NTBL)+CRW
      RBDY(NTBL+1)=RBDY(NTBL)
      SBDY(NTBL+1)=SBDY(NTBL)
C
      DRBDY(1)=(RBDY(2)-RBDY(1))/(XBDY(2)-XBDY(1))
      DRBDY(NTBL)=(RBDY(NTBL)-RBDY(NTBL-1))/(XBDY(NTBL)-XBDY(NTBL-1))
      DRBDY(NTBL+1)=DRBDY(NTBL)
      I=NTBL-1
      DO 43 J=2,I
      DRZ1=(RBDY(J)-RBDY(J-1))/(XBDY(J)-XBDY(J-1))
      DRZ2=(RBDY(J+1)-RBDY(J))/(XBDY(J+1)-XBDY(J))
   43 DRBDY(J)=0.5*(DRZ1+DRZ2)
C
      CALL SHAPE (XLEW,RBASE,DUM,DUM)
      FINE=XLEW/(2.0*RBASE)
C
C     OUTPUT BODY CHARACTERISTICS
C
      WRITE(6,738)THETAN,FINE,RBASE,RAVGW,RAVGT,XM,ZM,DXI,DXOUT
      WRITE (6,709)
      DO 14 J=1,NTBL
   14 WRITE (6,708) J,XBDY(J),RBDY(J),SBDY(J),DRBDY(J)
C
C     BEGINNING OF CALCULATIONS DEPENDENT ON ANGLE OF ATTACK (ALPHAD)
C
   19 ALPHA=ALPHAD
      CSALP=COS(ALPHA/RAD)
      ALPHA=SIN(ALPHA/RAD)
      IF (WLIMIT.LT.0.0) WLIMIT=0.0
      WRITE (6,705)
      WRITE (6,737) ALPHAD,EMACH,ALPIW,ALPIT,AKVLW1,AKVSW,AKVLT1,AKVST
     1 ,              WLIMIT
      IF (NSEPW.GT.1 .OR. NSEPT.GT.1) WRITE (6,751) AKVLW2,AKVLT2
      DO 219 J=1,NMX
      VEW(J)=0.0
      WEW(J)=0.0
      VEI(J)=0.0
  219 WEI(J)=0.0
C
C     COMPUTE NORMAL FORCE AND MOMENT CONTRIBUTION OF NOSE
C
      VLM=0.
      DO 240 J=2,NTBL
      IF (XBDY(J).GT.XLEW) GO TO 241
      K=J
  240 VLM=VLM+(RBDY(J)**2 + RBDY(J)*RBDY(J-1) + RBDY(J-1)**2)
     1        *(XBDY(J)-XBDY(J-1))/3.0
  241 XCPNP=XLEW-VLM/(RBDY(K)**2)
      CNNP=2.0*PI*(RBDY(K)**2)/REFS*SIN(ALPHAD/RAD)/COS(ALPHAD/RAD)
      CMNP=CNNP*(XM-XCPNP)/REFL
      CNNV=0.0
      CMNV=0.0
      XCPNV=0.0
      IF (ABS(ALPHAD).LE.4.0 .OR. THETAN.LE.4.0) NOSEV=0
      IF (NOSEV.LE.0) GO TO 119
C
C     CALCULATE EFFECT OF BODY VORTEX SHED FROM NOSE
C
      MTBL=1
      XA=XLEW/RBASE
      CALL BDYVTX (ALPHAD,THETAN,FINE,XA,EMACH,GAMN,NG,XGAMN,YGAMN,
     1             ZGAMN,XSA)
      IF (NG.EQ.0) NOSEV=0
      IF (NOSEV.LE.0) GO TO 119
      WRITE(6,735)GAMN(NG),YGAMN(NG),ZGAMN(NG),XSA
      DO 16 J=1,NG
      GAMN(J)=GAMN(J)*RBASE
      XGAMN(J)=XGAMN(J)*RBASE
      CALL SHAPE (XGAMN(J),RL,DUM,DUM)
      YGAMN(J)=YGAMN(J)*RL
   16 ZGAMN(J)=ZGAMN(J)*RL
      DUM=SQRT(YGAMN(NG)**2 + ZGAMN(NG)**2)
      IF (DUM.GT.RBASE) GO TO 116
      NOSEV=0
      NG=1
      WRITE (6,798)
  116 CONTINUE
      GAMNG=GAMN(NG)
      YGAMNG=YGAMN(NG)
      ZGAMNG=ZGAMN(NG)
C
C     COMPUTE VORTEX INDUCED FORCES AND MOMENTS ON NOSE
C
      CALL CNVNZ (XGAMN,YGAMN,ZGAMN,GAMN,NG,XM,CN,CM)
      MTBL=1
      CNNV=2.0*PI*CN/REFS
      CMNV=2.0*PI*CM/(REFS*REFL)
      XCPNV=0.0
      IF (NG.GT.1) XCPNV=XM-CMNV*REFL/CNNV
C
C     BEGINNING OF CALCULATIONS DEPENDENT ON WING INCIDENCE ANGLE
C
  119 CONTINUE
      ALPHAW=ALPHAD+ALPIW
      SNALIW=SIN(ALPIW/RAD)
      CSALIW=COS(ALPIW/RAD)
      MDUM=0
      NTRY=0
      MTBL=1
      NV=0
      H=DXI
      NEXT=NCW*NSW
      I=2
      DO 141 J=1,NEXT
      XP(J)=XLEW-PCXW(J)*BETA
      IF (J .GT. ((I-1)*NCW)) I=I+1
      YDUM=SUMYW(I) + RAVGW
      ZDUM=ZHLW-PVZW(J)-(XP(J)-XHLW)*SNALIW
      SP(J)=CMPLX(YDUM,ZDUM)
      WB(J)=0.0
  141 VP(J)=CMPLX(0.0,0.0)
C
C     COMPUTE BESKIN UPWASH ON WING
C
      IF (NBODY.LT.0) GO TO 540
      IF (NPRINT.LT.0) GO TO 510
      WRITE (6,745) HEAD(1)
      WRITE (6,746)
  510 CONTINUE
      DO 340 J=1,NEXT
```

```
      CALL SHAPE (XP(J),A,DUM,DUM)
      VP(J)= EYE*ALPHA*A*A/(SP(J)*SP(J))
      WB(J)=-AIMAG(VP(J))
      VB(J)=REAL(VP(J))
      IF (NPRINT.GE.0) WRITE (6,747) J,XP(J),SP(J),VP(J)
      VP(J)=CMPLX(0.0,0.0)
  340 CONTINUE
  540 CONTINUE
      IF (NOSEV.LE.0) GO TO 440
C
C      COMPUTE TRAJECTORY OF NOSE VORTEX PAST WING
C
  341 NV=NV+1
      NV2=NV+2
      GAM(1)=GAMNG
      YG(1)=YGAMNG
      ZG(1)=ZGAMNG
   15 XI=XLEW
      WRITE (6,711) XI
      WRITE (6,712) (I,GAM(I),YG(I),ZG(I),I=1,NV)
      XF=XTEW + DXI
      TITL=HEAD(1)
      CALL TRJTRY
      IF (MODUM.GT.10) GO TO 53
      YG(1)=YV(1,NTRY)
C
C      CORRECT VERTICAL POSITION OF VORTEX TO ACCOUNT FOR
C        WING VERTICAL DISPLACEMENT
C
      ZG(1)=ZV(1,NTRY)+ZMLW
  140 CONTINUE
  440 CONTINUE
C
C      SET UP TOTAL INDUCED VELOCITY FIELD ON WING
C
      IF (NPRINT.LT.0) GO TO 511
      WRITE (6,749) HEAD(1)
      WRITE (6,746)
  511 CONTINUE
      DO 142 J=1,NEXT
      DUM=CABS(VP(J))
      IF (DUM.GT.WLIMIT)  VP(J)= VP(J)*WLIMIT/DUM
      WEW(J)=-AIMAG(VP(J))
      VEW(J)= REAL(VP(J))
      VEW(J)=(VEW(J)+VB(J))
      WEW(J)=(WEW(J)+WB(J))
      DUM=-WEW(J)
      IF (NPRINT.GE.0) WRITE (6,747) J,XP(J),SP(J),VEW(J),DUM
      VEW(J)=VEW(J)*BETA
      WEW(J)=WEW(J)*BETA
  142 CONTINUE
C
C      WING CHARACTERISTICS
C
      DO 42 I=1,NMX
      DO 42 J=1,NMY
   42 FVN(I,J)=FVNW(I,J)
      XMW=0.0
      ZMW=ZMLW-ZM
      ALPHAW=ALPHAW*BETA
      CRW=CRW/BETA
      CTW=CTW/BETA
      YSEP=YSEPW-RAVGW
      IF (NPRINT.GE.0) WRITE (6,736) HEAD(1)
      CALL VTXLAT (FVN,NMX,NMY,NZ,NCW,MSW,CRW,CTW,SSPANW,ALPHAW,
     1            PSILEW,PSITEW,PHIW,SREFW,CAVEW,YW,YLOCW,SUMYW,XLELW,
     2            CHLCBW,SWPVPW,SWW,PVXW,PVYW,PVZW,PTLXW,PTLYW,PTLZW,
     3           FUWW,FVWW,FWWW,UEI,VEW,WEW,ALPHLW,XMW,ZMW,PCXW,
     4            NSEPW,YSEP,BETA,AKVLW1,AKVLW2,AKVSW)
      ALPHAW=ALPHAW/BETA
      CRW=CRW*BETA
      CTW=CTW*BETA
      CNWBP=CNORM*SREFW/REFS*CSALIW/BETA
      CAWBP=CNWBP*SNALIW/CSALIW
      CNWBVL=(CNV1*AKVLW1+CNV2*AKVLW2)*SREFW/REFS*CSALIW/BETA
      CAWBVL=CNWBVL*SNALIW/CSALIW
      CNWBVS=CNVS*SREFW/REFS*CSALIW*AKVSW/BETA
      CAWBVS=CNWBVS*SNALIW/CSALIW
      XCPWBP=XLEW-XCUPP*BETA
      IF (XCPW.GT.0.0) XCPWBP=XCP-
      XCPWVL=XLEW-XCUPLE*BETA
      XCPWVS=XLEW-XCUPSL*BETA
      CMWBP=(XM-XCPWBP)*CNWBP/REFL
      CMWBVL=(XM-XCPWVL)*CNWBVL/REFL
      CMWBVS=(XM-XCPWVS)*CNWBVS/REFL
C
C      DETERMINE VERTICAL LOCATION OF SEPARATION VORTICES
C
      EPS=90.-ATAN(TAN(PSILEW(1)/RAD)*BETA)*RAD
      CALL ZVTX (ALPHAW,EPS,BTA2)
      IF (CNV.LT.0.0 .AND. ALPHAW.GT.0.0) BTA2=-BTA2
      GAM(NV+1)=GAMT/BETA
      IF (YTR .GT. SSPANW .OR. YTR.LE.0.0) YTR=SSPANW*PI/4.0
      YTR=YTR+RAVGW
      YG(NV+1)=YTR
      ZG(NV+1)=ZMLW-(XTEW-XHLW)*SNALIW
      NV=NV+1
      Z=CRW*SIN(BTA2/RAD)
      SUMGAM=0.0
      IF (AKVLW1.LE.0.0 .AND. AKVLW2.LE.0.0) GO TO 520
      DO 245 I=1,NSEPW
      GAM(NV+1)=GAM1(I)/BETA
      SUMGAM=SUMGAM+GAM(NV+1)
      YG(NV+1)=Y1(I)+RAVGW
      IF (I.GT.1) Z=(CTW+(CRW-CTW)*YSEPW/SSPANW)*SIN(BTA2/RAD)
      ZG(NV+1)=Z+ZMLW-(XTEW-XHLW)*SNALIW
      SIGMA(I)=CMPLX(YG(NV+1),ZG(NV+1))
      IF (GAM(NV+1) .NE. 0.0) NV=NV+1
  245 CONTINUE
  520 NV2=NV+2
C
C      COMPUTE NORMAL FORCE AND MOMENT ON BODY IN PRESENCE OF WING
C
      XCPBWP=XCPWBP
      XCPBWV=XCPWVL
      XCPBWS=XCPWVS
      DUM=(RAVGW*RAVGW*RAVGW/YTR)/(YTR*RAVGW)
      CNBWP=DUM*CNWBP
      CNBWV=0.0
      IF (AKVLW1.LE.0.0 .AND. AKVLW2.LE.0.0) GO TO 521
      DO 143 I=1,NSEPW
      DUM=(RAVGW*REAL(RAVGW*RAVGW/CONJG(SIGMA(I))))
     1               /(REAL(SIGMA(I))*RAVGW)
      NI=NV-NSEPW+I
  143 CNBWV=CNBWV+DUM*GAM(NI)/SUMGAM*CNWBVL
  521 CMBWP=(XM-XCPBWP)*CNBWP/REFL
      CMBWV=(XM-XCPBWV)*CNBWV/REFL
      DUM=(RAVGW*RAVGW*RAVGW/BS2W)/(BS2W*RAVGW)
      CNBWVS=DUM*CNWBVS
      CMBWVS=(XM-XCPBWS)*CNBWVS/REFL
      NTRYSV=NTRY
  151 CONTINUE
      ALPHAT=ALPHAD+ALPIT
      SNALIT=SIN(ALPIT/RAD)
      CSALIT=COS(ALPIT/RAD)
      NTRY=NTRYSV
C
C      COMPUTE TRAJECTORY OF NOSE VORTEX AND WING VORTICES PAST TAIL
C
      XI=XITE
      IF (NOSEV.GT.0) XI=XV(NTRY) + .001
      IF (NPRINT.GE.0) WRITE (6,705)
      WRITE (6,711) XI
      WRITE (6,712) (I,GAM(I),YG(I),ZG(I),I=1,NV)
      XINIT=XI
      TITL=HEAD(2)
      IF (MSWT.LE.0) XTET=XBDY(NTBL)-DXI
      XF=XTET + DXOUT
      IF (XF.LE.XI) GO TO 50
      DXI=H
      NEXT=NCWT*MSWT
      IF (NEXT.GT.0) GO TO 148
      CALL TRJTRY
      IF (MODUM.GT.10) GO TO 53
```

```
      GO TO 147
  148 CONTINUE
      I=2
      DO 145 J=1,NEXT
      XP(J)=XLET-PCAT(J)*BETA
      IF (J .GT. ((I-1)*NCWT)) I=I+1
      YDUM=SUMYT(I)*RAVGT
      ZDUM=ZHLT-PVZI(J)-(XP(J)-XHLT)*SNALIT
      SP(J)=CMPLX(YDUM,ZDUM)
      WB(J)=0.0
  145 VP(J)=CMPLX(0.0,0.0)
C
C      COMPUTE BESKIN UPWASH ON TAIL
C
      IF (NBODY.LT.0) GO TO 548
      IF (NPRINT.LT.0) GO TO 512
      WRITE (6,745) HEAD(2)
      WRITE (6,746)
  512 CONTINUE
      DO 348 J=1,NEXT
      CALL SHAPE ( XP(J),A,DUM,DUM)
      VP(J)=EYE*ALPHA*A*A/(SP(J)*SP(J))
      WB(J)=-AIMAG(VP(J))
      VB(J)=REAL(VP(J))
      IF (NPRINT.GE.0) WRITE (6,747) J,XP(J),SP(J),VP(J)
      VP(J)=CMPLX(0.0,0.0)
  348 CONTINUE
  548 CONTINUE
      CALL TRJTRY
      IF (MDUM.GT.10) GO TO 53
C
C      SET UP TOTAL INDUCED VELOCITY FIELD ON TAIL
C
      IF (NPRINT.LT.0) GO TO 513
      WRITE (6,749) HEAD(2)
      WRITE (6,746)
  513 CONTINUE
      DO 146 J=1,NEXT
      DUM=CABS(VP(J))
      IF (DUM.GT.WLIMIT)  VP(J)= VP(J)*WLIMIT/DUM
      VEI(J)= REAL(VP(J))
      WEI(J)=-AIMAG(VP(J))
      VEI(J)=VEI(J)+VB(J)
      WEI(J)=WEI(J)+WB(J)
      DUM=-WEI(J)
      IF (NPRINT.GE.0) WRITE (6,747) J,XP(J),SP(J),VEI(J),DUM
      VEI(J)=VEI(J)*BETA
      WEI(J)=WEI(J)*BETA
  146 CONTINUE
  147 CONTINUE
C
C      COMPUTE NORMAL FORCE AND MOMENT CONTRIBUTION OF AFTERBODY
C
      IF (MSWT.LE.0) XLET=XBODY(NTBL)
      XFINL=XLET
      DUM=0.5*(XFINL-XINIT)/RAVGW
      NG=DUM
      IF (NG.LE.0) GO TO 150
      CALL CNVTX (XINIT,XFINL,NG,XM,CN,CM)
      CNA=2.0*PI*CN/REFS
      CMA=2.0*PI*CM/(REFS*REFL)
      XCPA=XM-CMA*REFL/CNA
  150 IF (MSWT.LE.0) GO TO 50
C
C      TAIL CHARACTERISTICS
C
      DO 45 I=1,NMX
      DO 45 J=1,NMY
   45 FVN(I,J)=FVNT(I,J)
      XMT=0.0
      ZMT=ZMLT-ZM
      CRWT=CRWT/BETA
      CTWT=CTWT/BETA
      ALPHAT=ALPHAT*BETA
      YSEP=YSEPT-RAVGT
      IF (NPRINT.GE.0) WRITE (6,736) HEAD(2)
      CALL VTALAT (FVN,NMX,NMY,NZ,NCWT,MSWT,CRWT,CTWT,SSPANT,ALPHAT,
     1             PSILET,PSITET,PHIT,SREFT,CAVET,YT,YLOCT,SUMYT,XLELT,
     2             CHLCBT,SWPVPT,SWT,PVXT,PVYT,PVZT,PTLXT,PTLYT,PTLZT,
     3            FUWT,FVWT,FWWT,UEI,VEI,WEI,ALPHLT,XMT,ZMT,PCXI,
     4             NSEPT,YSEP,BETA,AKVLT1,AKVLT2,AKVST)
      ALPHAT=ALPHAT/BETA
      CRWT=CRWT*BETA
      CTWT=CTWT*BETA
      CNTBP=CNORM*SREFT/REFS*CSALIT/BETA
      CATBP=CNTBP*SNALIT/CSALIT
      CNTBVL=(CNV1*AKVLT1+CNV2*AKVLT2)*BREFT*CSALIT/REFS/BETA
      CATBVL=CNTBVL*SNALIT/CSALIT
      CNTBVS=CNVS*SREFT/REFS*CSALIT*AKVST/BETA
      CATBVS=CNTBVS*SNALIT/CSALIT
      XCPTBP=XLET-XCOPP*BETA
      IF (XCPT.GT.0.0) XCPTBP=XCPT
      XCPTVL=XLET-XCOPLE*BETA
      XCPTVS=XLET-XCOPSE*BETA
      CMTBP=(XM-XCPTBP)*CNTBP/REFL
      CMTBVL=(XM-XCPTVL)*CNTBVL/REFL
      CMTBVS=(XM-XCPTVS)*CNTBVS/REFL
      DO 344 I=1,NV
      YG(I)=YV(I,NTRY)
  344 ZG(I)=ZV(I,NTRY)
      WRITE (6,711) XV(NTRY)
      WRITE (6,712) (I,GAM(I),YG(I),ZG(I),I=1,NV)
C
C      DETERMINE VERTICAL LOCATION OF SEPARATION VORTICES
C
      EPS=90.-ATAN(TAN(PSILET(1)/RAD)*BETA)*RAD
      CALL ZVTX (ALPHAT,EPS,BTA2)
      IF (CNV.LT.0.0 .AND. ALPHAT.GT.0.0) BTA2=-BTA2
      Z=CRWT*SIN(BTA2/RAD)
      GAMT=GAMT/BETA
      IF (YTR.GT.SSPANT .OR. YTR.LE.0.0) YTR=SSPANT*PI/4.0
      YTR=YTR+RAVGT
      ZTR=ZHLT-(XTET-XHLT)*SNALIT
      NI=NV+1
      WRITE (6,752) NI,GAMT,YTR,ZTR
      IF (AKVLT1.LE.0.0 .AND. AKVLT2.LE.0.0) GO TO 530
      SUMGAM=0.0
      DO 244 I=1,NSEPT
      GAM1(I)=GAM1(I)/BETA
      Y1(I)=Y1(I)+RAVGT
      SUMGAM=SUMGAM+GAM1(I)
      IF (I.GT.1) Z=(CTWT+(CRWT-CTWT)*YSEPT/SSPANT)*SIN(BTA2/RAD)
      Z=Z+ZHLT-(XTET-XHLT)*SNALIT
      SIGMA(I)=CMPLX(Y1(I),Z)
      IF (ABS(GAM1(I)).LT.1.0E-06) GO TO 244
      NI=NI+1
      WRITE (6,752) NI,GAM1(I),Y1(I),Z
  244 CONTINUE
C
C      COMPUTE NORMAL FORCE AND MO---* ON BODY IN PRESENCE OF TAIL
C
  530 XCPBTP=XCPTBP
      XCPBTV=XCPTVL
      XCPBTS=XCPTVS
      DUM=(RAVGT-RAVGT*RAVGT/YTR)/(YTR-RAVGT)
      CNBTP=DUM*CNTBP
      CNBTV=0.0
      IF (AKVLT1.LE.0.0 .AND. AKVLT2.LE.0.0) GO TO 531
      DO 243 I=1,NSEPT
      DUM=(RAVGT-REAL(RAVGT*RAVGT/CONJG(SIGMA(I))))
     1              /(REAL(SIGMA(I))-RAVGT)
  243 CNBTV=CNBTV+DUM*GAM1(I)/SUMGAM*CNTBVL
  531 CMBTP=(XM-XCPBTP)*CNBTP/REFL
      CMBTV=(XM-XCPBTV)*CNBTV/REFL
      DUM=(RAVGT-RAVGT*RAVGT/BS2T)/(BS2T-RAVGT)
      CNBTVS=DUM*CNTBVS
      CMBTVS=(XM-XCPBTS)*CNBTVS/REFL
C
   50 CONTINUE
      CNTOT=CNNP + CNNV + CNWBP + CNWBVL + CNWBVS + CNBWP + CNBWV + CNA
     1    + CNTBP + CNTBVL + CNTBVS + CNBTP + CNBTV + CNBWVS + CNBTVS
      CMTOT=CMNP + CMNV + CMWBP + CMWBVL + CMWBVS + CMBWP + CMBWV + CMA
     1    + CMTBP + CMTBVL + CMTBVS + CMBTP + CMBTV + CMBWVS + CMBTVS
```

```
      XCPTOT=XM-CMTOT*REFL/CNTOT                                        SB01 926
      CATOT=CAWBP + CAWBVL + CAWBVS + CATBP + CATBVL + CATBVS           SB01 927
      IF (NSWT.GT.0) WRITE (6,705)                                      SB01 928
      CL=0.0                                                            SB01 929
      TNA=TAN(ALPHAD/RAD)                                               SB01 930
      CSA=COS(ALPHAD/RAD)                                               SB01 931
      CSAW=COS(ALPHAW/RAD)                                              SB01 932
      CSAT=COS(ALPHAT/RAD)                                              SB01 933
      WRITE (6,705)                                                     SB01 934
      WRITE (6,737) ALPHAD,EMACH,ALPIW,ALPIT,AKVLW1,AKVSW,AKVLT1,AKVST  SB01 935
     1 ,            WLIMIT                                              SB01 936
      IF (NSEPW.GT.1 .OR. NSEPT.GT.1) WRITE (6,751) AKVLW2,AKVLT2       SB01 937
      WRITE (6,703)                                                     SB01 938
      WRITE (6,713)                                                     SB01 939
      WRITE (6,714) HEAD(7)                                             SB01 940
      CL1=CNNP*CSA                                                      SB01 941
      CL2=CNNV*CSA                                                      SB01 942
      CD1=CL1*TNA                                                       SB01 943
      CD2=CL2*TNA                                                       SB01 944
      CL=CL+CL1+CL2                                                     SB01 945
      WRITE (6,715) HEAD(9),CNNP,CMNP,XCPNP,CL1,CD1                     SB01 946
      WRITE (6,710)  CNNV,CMNV,XCPNV,CL2,CD2                            SB01 947
      WRITE (6,714) HEAD(1)                                             SB01 948
      CL1=CNWBP*CSAW                                                    SB01 949
      CL2=CNWBVL*CSAW                                                   SB01 950
      CL3=CNWBVS*CSAW                                                   SB01 951
      CD1=CL1*TNA                                                       SB01 952
      CD2=CL2*TNA                                                       SB01 953
      CD3=CL3*TNA                                                       SB01 954
      WRITE (6,715) HEAD(3),CNWBP,CMWBP,XCPWBP,CL1,CD1,CAWBP            SB01 955
      WRITE (6,716) HEAD(3),CNWBVL,CMWBVL,XCPWVL,CL2,CD2,CAWBVL         SB01 956
      WRITE (6,717) HEAD(3),CNWBVS,CMWBVS,XCPWVS,CL3,CD3,CAWBVS         SB01 957
      CL=CL+CL1+CL2+CL3                                                 SB01 958
      WRITE (6,714) HEAD(8)                                             SB01 959
      CL1=CNBWP*CSA                                                     SB01 960
      CL2=CNBWV*CSA                                                     SB01 961
      CL3=CNBWVS*CSA                                                    SB01 962
      CD1=CL1*TNA                                                       SB01 963
      CD2=CL2*TNA                                                       SB01 964
      CD3=CL3*TNA                                                       SB01 965
      WRITE (6,715) HEAD(4),CNBWP,CMBWP,XCPBWP,CL1,CD1                  SB01 966
      WRITE (6,716) HEAD(4),CNBWV,CMBWV,XCPBWV,CL2,CD2                  SB01 967
      WRITE (6,717) HEAD(4),CNBWVS,CMBWVS,XCPBWS,CL3,CD3                SB01 968
      CL=CL+CL1+CL2+CL3                                                 SB01 969
      CL1=CNA*CSA                                                       SB01 970
      CD1=CL1*TNA                                                       SB01 971
      WRITE (6,718) CNA,CMA,XCPA,CL1,CD1                                SB01 972
      CL=CL+CL1                                                         SB01 973
      WRITE (6,714) HEAD(2)                                             SB01 974
      CL1=CNTBP*CSAT                                                    SB01 975
      CL2=CNTBVL*CSAT                                                   SB01 976
      CL3=CNTBVS*CSAT                                                   SB01 977
      CD1=CL1*TNA                                                       SB01 978
      CD2=CL2*TNA                                                       SB01 979
      CD3=CL3*TNA                                                       SB01 980
      CL=CL+CL1+CL2+CL3                                                 SB01 981
      WRITE (6,715) HEAD(5),CNTBP,CMTBP,XCPTBP,CL1,CD1,CATBP            SB01 982
      WRITE (6,716) HEAD(5),CNTBVL,CMTBVL,XCPTVL,CL2,CD2,CATBVL         SB01 983
      WRITE (6,717) HEAD(5),CNTBVS,CMTBVS,XCPTVS,CL3,CD3,CATBVS         SB01 984
      WRITE (6,714) HEAD(8)                                             SB01 985
      CL1=CNBTP*CSA                                                     SB01 986
      CL2=CNBTV*CSA                                                     SB01 987
      CL3=CNBTVS*CSA                                                    SB01 988
      CD1=CL1*TNA                                                       SB01 989
      CD2=CL2*TNA                                                       SB01 990
      CD3=CL3*TNA                                                       SB01 991
      CL=CL+CL1+CL2+CL3                                                 SB01 992
      WRITE (6,715) HEAD(6),CNBTP,CMBTP,XCPBTP,CL1,CD1                  SB01 993
      WRITE (6,716) HEAD(6),CNBTV,CMBTV,XCPBTV,CL2,CD2                  SB01 994
      WRITE (6,717) HEAD(6),CNBTVS,CMBTVS,XCPBTS,CL3,CD3                SB01 995
      CD1=CL*TNA                                                        SB01 996
      WRITE (6,719) CNTOT,CMTOT,XCPTOT,CL,CD1 ,CATOT                    SB01 997
      CD2=0.0                                                           SB01 998
      IF (CL.NE.0.0) CD2=CD1/(CL*CL)                                    SB01 999
      WRITE (6,742) CD2                                                 SB011000
      IF (NTRY.EQ.0 .OR. NPRINT.LT.0) GO TO 60                          SB011001
      IF (NTRIM.GT.0 .AND. NFLAG.EQ.0) GO TO 70                         SB011002
   52 WRITE (6,720)                                                     SB011003
   53 J=1                                                               SB011004
      IF (NOSEV.EQ.0) GO TO 56                                          SB011005
      WRITE (6,727) J                                                   SB011006
      J=J+1                                                             SB011007
   56 IF (J.GT.NV) GO TO 58                                             SB011008
      WRITE (6,728) J                                                   SB011009
      J=J+1                                                             SB011010
      IF (J.GT.NV) GO TO 58                                             SB011011
      DO 57 K=J,NV                                                      SB011012
   57 WRITE (6,729) K                                                   SB011013
   58 CONTINUE                                                          SB011014
      WRITE (6,725)                                                     SB011015
      DO 51 J=1,NV                                                      SB011016
   51 WRITE (6,726) J,GAM(J)                                            SB011017
      IF (NTRY.EQ.0) GO TO 65                                           SB011018
      DO 55 J=1,NTRY                                                    SB011019
      JV=NVP(J)                                                         SB011020
      WRITE (6,721)                                                     SB011021
      WRITE (6,722) XV(J),(ARAY(N,J),N=1,5)                             SB011022
      WRITE (6,723)                                                     SB011023
      DO 54 K=1,JV                                                      SB011024
   54 WRITE (6,724) K,YV(K,J),ZV(K,J)                                   SB011025
   55 CONTINUE                                                          SB011026
   60 IF (NTRIM.GT.0 .AND. NDUM.LE.10) GO TO 70                         SB011027
   65 NTIME=1                                                           SB011028
      NFLAG=0                                                           SB011029
      NOSEV=NOSE                                                        SB011030
      CM2=0.0                                                           SB011031
      AL2=0.0                                                           SB011032
      READ (5,750) NDEX,ALPHAD,ALPIW,AKVLW1,AKVLW2,AKVSW,               SB011033
     1                      ALPIT,AKVLT1,AKVLT2,AKVST,WLIMIT            SB011034
   61 IF (NDEX.GT.0) GO TO 19                                           SB011035
      GO TO 10                                                          SB011036
   70 CONTINUE                                                          SB011037
      NTIME=NTIME+1                                                     SB011038
      CM1=CM2                                                           SB011039
      AL1=AL2                                                           SB011040
      CM2=CMTOT                                                         SB011041
      IF (ABS(CMTOT).LE.CMTEST) GO TO 72                                SB011042
      IF (NTIME.GT.5) GO TO 73                                          SB011043
      IF (NTRIM.GE.2) GO TO 75                                          SB011044
C                                                                       SB011045
C      CALCULATE TRIM ASSUMING WING INCIDENCE IS VARIABLE               SB011046
C                                                                       SB011047
      AL2=ALPIW                                                         SB011048
      IF (NTIME.GT.2) GO TO 71                                          SB011049
      CMDUM=CMTOT-CMWBP-CMWBVL-CMWBVS-CMBWP-CMBWV-CMBWVS                SB011050
      CNTR=CMDUM*REFL/(XCPWBP-XM)                                       SB011051
      DUM=CNWBP+CNWBVL+CNWBVS+CNBWP+CNBWV+CNBWVS                        SB011052
      SAD2=CNTR*SIN(ALPHAW/RAD)/DUM                                     SB011053
      IF (ABS(SAD2).GT.0.70) GO TO 371                                  SB011054
      SAD2=ARSIN(SAD2)                                                  SB011055
      ALPIW=SAD2*RAD-ALPHAD                                             SB011056
      GO TO 119                                                         SB011057
  371 ALPIW=ALPIW+2.0*(CNTR-DUM)/(ABS(CNTR-DUM))                        SB011058
      GO TO 119                                                         SB011059
   71 ALPIW=(CM2*AL1-CM1*AL2)/(CM2-CM1)                                 SB011060
      GO TO 119                                                         SB011061
   72 IF (NFLAG.GT.0) GO TO 65                                          SB011062
      NFLAG=1                                                           SB011063
      WRITE (6,740)                                                     SB011064
      GO TO 52                                                          SB011065
   73 WRITE (6,741)                                                     SB011066
      IF (NFLAG.GT.0) GO TO 65                                          SB011067
      NFLAG=1                                                           SB011068
      GO TO 52                                                          SB011069
C                                                                       SB011070
C      CALCULATE TRIM ASSUMING TAIL INCIDENCE IS VARIABLE               SB011071
C                                                                       SB011072
   75 CONTINUE                                                          SB011073
      AL2=ALPIT                                                         SB011074
      IF (NTIME.GT.2) GO TO 76                                          SB011075
      CMDUM=CMTOT-CMTBP-CMTBVL-CMTBVS-CMBTP-CMBTV-CMBTVS                SB011076
      CNTR=CMDUM*REFL/(XM-XCPTBP)                                       SB011077
      DUM=CNTBP+CNTBVL+CNTBVS+CNBTP+CNBTV+CNBTVS                        SB011078
      SAD2=CNTR*SIN(ALPHAT/RAD)/DUM                                     SB011079
```

```
      IF (ABS(SAD2).GT.0.70) GO TO 375
      SAD2=ARSIN(SAD2)
      ALPIT=SAD2*RAD-ALPHAD
      GO TO 151
  375 ALPIT=ALPIT+2.0*(CNTN-DUM)/(ABS(CNTR-DUM))
      GO TO 151
   76 ALPIT=(CM2*AL1-CM1*AL2)/(CM2-CM1)
      GO TO 151
      END




      SUBROUTINE LATTUS (FVN,NMX,NMY,NZ,NCW,MSW,CRW,CTW,SSPAN,ALPHAD,
     1                   PSIWLE,PSIWTE,PHID,SREF,CAVE,SUMY,YLOC,Y,
     2                   XLEL,ZLEL,CHLOCL,CHLOCB,SWPVVP,SW,
     3                   PVX,PVY,PVZ,PCX,PTLX,PTLY,PTLZ,FUW,FVW,FWW)
C
C       SUBROUTINE TO SET UP LATTICE GEOMETRY
C
      DIMENSION FVN(NMX,NMY)
      DIMENSION SUMY(NZ),YLOC(NZ),XLEL(NZ),ZLEL(NZ),Y(NZ),
     1          CHLOCB(NZ),CHLOCL(NZ),PSIWLE(NZ),PSIWTE(NZ)
      DIMENSION SW(NMX),SWPVVP(NMX),PVX(NMX),PVY(NMX),PVZ(NMX),
     1          PCX(NMX),PTLX(NMX),PTLY(NMX),PTLZ(NMX),
     2          FUW(2),FVW(2),FWW(2),SWPWLE(25),SWPWTE(25),YWW(2)
      COMMON/INF/BOT
      PI=3.1415926
      DTR=PI/180.0
      IMAX=MSW+1
      M=MSW*NCW
      PHI=PHID*DTR
      SPHI=SIN(PHI)
      CSPHI=COS(PHI)
      TANPHI=TAN(PHI)
      DO 110 I=1,IMAX
      SWPWLE(I)=TAN(PSIWLE(I)*DTR)
  110 SWPWTE(I)=TAN(PSIWTE(I)*DTR)
      DO 111 JC=1,NMX
      DO 111 JV=1,NMY
  111 FVN(JC,JV)=0.0
      BOT= SSPAN
C     LAY-OUT VORTEX BOUND LEG MIDPOINTS AND CONTROL POINTS ON THE WING
C
      D=NCW
      SUMY(1)=0.0
      YLOC(1)=0.0
      XLEL(1)=0.0
      ZLEL(1)=0.0
      CHLOCB(1)=0.0
      CHLOCL(1)=CRW
      SREF=0.0
      DO 701 I=2,IMAX
      SUMY(I)=(Y(I)+Y(I-1))/2.0
      YLOC(I)=SUMY(I)/SSPAN
      XLEL(I)=(Y(I-1)-Y(I))*SWPWLE(I) + XLEL(I-1)
      ZLEL(I)=(Y(I-1)-Y(I))*TANPHI  + ZLEL(I-1)
      CHLOCL(I)=CHLOCL(I-1) - (Y(I)-Y(I-1))*(SWPWLE(I)-SWPWTE(I))
      IF (I .LT. IMAX) GO TO 115
      IF (CHLOCL(IMAX) .LT. 0.02*CRW) CHLOCL(IMAX)=0.0
  115 CHLOCB(I)=(CHLOCL(I)  +CHLOCL(I-1))*0.5
      SREF=SREF+(CHLOCL(I)+CHLOCL(I-1))*(Y(I)-Y(I-1))
      DO 720 JCW=1,NCW
      A=JCW
      SWPC=SWPWLE(I)-  ((A-0.25)/D)*(SWPWLE(I)-SWPWTE(I))
      J=(I-2)*NCW+JCW
      SWPV=SWPWLE(I)-((A-0.75)/D)*(SWPWLE(I)-SWPWTE(I))
C
C     DISTINCTION BETWEEN SWEEP IN CHORDAL(SWPVVP) AND PLANFORM(SWPV)
C     PLANES
C
      SWPVVP(J)=SWPV*CSPHI
      SWPVVP(J)= ATAN (SWPVVP(J))
C
C       COORDINATES OF BOUND VORTEX , PV
```

```
C       COORDINATES OF CONTROL POINTS , PC
C
      SW(J)=(Y(I)-Y(I-1))/(2.0*CSPHI)
      PVX(J)=XLEL(I-1)-((A-0.75)*(CHLOCL(I-1)/D))-((Y(I)-Y(I-1))*0.5)*
     1       SWPV
      PVY(J)=-SUMY(I)
      PVZ(J)=ZLEL(I-1)-((Y(I)-Y(I-1))*0.5)*TANPHI
      PCX(J)=XLEL(I-1)-((A-0.25)*(CHLOCL(I-1)/D))-((Y(I)-Y(I-1))*0.5)*
     1       SWPC
C
C     COORDINATES OF 3/4 CHORD ELEMENTAL PANEL LEFT SIDE POINT,PTL
C
      PTLX(J)=XLEL(I-1)-((A-0.25)*((CHLOCL(I-1)/D))-(Y(I)-Y(I-1))*SWPC
      PTLY(J)=-Y(I)
      PTLZ(J)=ZLEL(I-1)-(Y(I)-Y(I-1))*TANPHI
  720 CONTINUE
  701 CONTINUE
      CAVE=SREF/(2.*SSPAN)
      CTW=CHLOCL(IMAX)
C
C     BUILD UP L,M,S, OF BOUNDARY CONDITION, THE COEFFICIENT MATRIX
C
C     CONTROL POINTS ON WING, VORTICES ON WING
C
      DO 212 JC=1,M
      DO 212 JV=1,M
      XWW=PCX(JC)-PVX(JV)
      YWW(1)=PVY(JC)-PVY(JV)
      YWW(2)=PVY(JC)+PVY(JV)
      ZWW=PVZ(JC)-PVZ(JV)
      DO 261 I=1,2
      CALL INFWW(SWPVVP(JV),PHI,XWW,YWW(I),ZWW,SW(JV),FUW(I),FVW(I),
     1FWW(I))
      SWPVVP(JV)=-SWPVVP(JV)
      PHI=-PHI
  261 CONTINUE
      FVN(JC,JV)=(FWW(1)+FWW(2))*CSPHI-(FVW(1)+FVW(2))*SPHI
  212 CONTINUE
      RETURN
      END




      SUBROUTINE SHAPE (X,A,S,DS)
C
C      TABLE LOOK-UP OF BODY COORDINATES
C
      DIMENSION XBDY(100),RBDY(100),SBDY(100),DRBDY(100)
      COMMON /BODY/ NTBL,MTBL,XBDY,RBDY,SBDY,DRBDY
C
  700 FORMAT (///5X12H****** XBDY=,F10.4,2X2HX=, F10.4,47H, X OUTSIDE RA
     1NGE OF TABLE OF BODY COORDINATES /40X,23HEXECUTION STOP IN SHAPE )
      NT=NTBL+1
    9 DO 10 J=MTBL,NT
      K=J
      IF (X-XBDY(J)) 12,13,10
   10 CONTINUE
   11 WRITE (6,700)   XBDY(K),X
      STOP
   13 A=RBDY(K)
      S=SBDY(K)
      DS=DRBDY(K)
      GO TO 40
   12 IF (K.LE.1) GO TO 11
      IF (X.LE.XBDY(K-1)) GO TO 120
      DEL=(X-XBDY(K-1))/(XBDY(K)-XBDY(K-1))
      A=RBDY(K-1) + DEL*(RBDY(K)-RBDY(K-1))
      S=SBDY(K-1) + DEL*(SBDY(K)-SBDY(K-1))
      DS=DRBDY(K-1) + DEL*(DRBDY(K)-DRBDY(K-1))
   40 CONTINUE
      IF (S.LT.A) S=A
      IF (K.GT.2) MTBL=K-2
      RETURN
  120 MTBL=1
      GO TO 9
      END
```

```
      SUBROUTINE BDYVTX (ALPHAD,THETAN,FINE,XA,EMACH,GAMN,NG,
     1                   XGAMN,YGAMN,ZGAMN,XSA)
C
C      COMPUTE THE STRENGTH AND POSITION OF BODY VORTICES
C
      DIMENSION XST(11),ZA1(11),YA1(11),YA2(11),GAMT(11)
      DIMENSION XGAMN(10),YGAMN(10),ZGAMN(10),GAMN(10)
C
      DATA XST/0.0,1.0,2.0,3.0,4.0,5.0,6.0,7.0,8.0,9.0,10.0/
      DATA GAMT/0.3,0.32,0.34,0.40,0.48,0.62,0.77,0.90,1.0,1.08,1.15/
      DATA YA1/.45,.463,.477,.492,.505,0.52,.534,0.55,0.565,0.575,0.585/
      DATA YA2/.63,.64,.653,.665,.678,.69,.7,.715,.725,.735,.74/
      DATA ZA1/1.14,1.26,1.38,1.5,1.615,1.73,1.84,1.95,2.05,2.14,2.20/
C
  701 FORMAT (//1X5H*****,3X62HASYMMETRIC OR UNSTEADY BODY VORTEX SEPARA
     1TION POSSIBLE    *****//)
C
      NTABLE=11
      DELTA=0.0
      F=1.0
      IF (ALPHAD.LT.0.0) F=-1.0
      ALPHA=F*ALPHAD
C
      IF (THETAN.LE.30.0) GO TO 10
      XSA=10./(ALPHA -4.0) + 2.0
      GO TO 13
   10 XSA=32.0 - SQRT(1024.0*(ALPHA -4.0)/(THETAN-4.0))
   13 IF (XSA.LT.0.0) XSA=0.0
      NG=0
      IF (XSA.GE.XA) RETURN
C
C      COMPUTE UPPER LIMIT FOR SYMMETRIC VORTEX SEPARATION
C
   12 ALMT=((FINE-12.0)**2)/3.57 + 12.0
      IF (ALPHA .GT.ALMT) WRITE (6,701)
      SNALP=SIN(ALPHAD/57.2957795)*F
      DXPR=XA-XSA
      IF (DXPR.LT. 1.0E-02) GO TO 15
      XPAR=DXPR/2.0
      NG=XPAR
      IF (NG.GT.10) NG=10
      IF (NG.LE.1) NG=2
      GO TO 16
   15 NG=1
      DXPR=0.0
      XPAR=0.0
   16 CONTINUE
      DO 40 JG=1,NG
      DUM1=JG
      DUM2=NG
      DUM=0.0
      IF (NG.GT.1) DUM=(DUM1-1.0)/(DUM2-1.0)
      XPAR=XSA + DUM*DXPR
      IF (NG.EQ.1) XPAR=XA
      XGAMN(JG)=XPAR
      XPAR=(XPAR-XSA)*SNALP
      IF (XPAR .LT. 0.0) XPAR=0.0
      DO 20 J=1,NTABLE
      K=J
      IF (XPAR-XST(J)) 22,23,20
   20 CONTINUE
   23 GAMN(JG)=GAMT(K)*SNALP
      ZGAMN(JG)=ZA1(K)
      Y1=YA1(K)
      Y2=YA2(K)
      GO TO 30
   22 DELTA=(XPAR-XST(K-1))/(XST(K)-XST(K-1))
      GAMN(JG)=GAMT(K-1) + DELTA*(GAMT(K)-GAMT(K-1))
      GAMN(JG)=GAMN(JG)*SNALP
      ZGAMN(JG)=ZA1(K-1) + DELTA*(ZA1(K)-ZA1(K-1))
      Y1=YA1(K-1) + DELTA*(YA1(K)-YA1(K-1))
      Y2=YA2(K-1) + DELTA*(YA2(K)-YA2(K-1))
   30 YGAMN(JG)=Y1
      IF (EMACH .GT. 1.0) YGAMN(JG)=Y2
      GAMN(JG)=GAMN(JG)*F
   40 ZGAMN(JG)=ZGAMN(JG)*F
      RETURN
      END
```

```
      SUBROUTINE  CNVNZ (XGAMN,YGAMN,ZGAMN,GAMN,NG,XM,CN,CM)
C
C      COMPUTE NORMAL FORCE AND MOMENT ON NOSE DUE TO SHED BODY VORTEX
C
      COMPLEX SIGMA,SIGMAB
      DIMENSION XGAMN(10),YGAMN(10),ZGAMN(10),GAMN(10)
      DIMENSION A(2),SIGMA(2),SIGMAB(2)
C
      CN=0.0
      CND=0.0
      CM=0.0
      IF (NG.LE.1) RETURN
      XI=XGAMN(1)
      CALL SHAPE (XI,A(1),DUM,DUM)
      SIGMA(1)=CMPLX(YGAMN(1),ZGAMN(1))
      SIGMAB(1)=CMPLX(0.0,0.0)
      DUM=(YGAMN(1))**2 + (ZGAMN(1))**2
      IF (DUM.GT.0.0) SIGMAB(1)=SIGMA(1)-A(1)*A(1)/CONJG(SIGMA(1))
      DO 50 J=2,NG
      XI=XGAMN(J)
      SIGMA(2)=CMPLX(YGAMN(J),ZGAMN(J))
      CALL SHAPE (XI,A(2),DUM,DUM)
      SIGMAB(2)=SIGMA(2)-A(2)*A(2)/CONJG(SIGMA(2))
      CND=4.0*GAMN(J)*REAL(SIGMAB(2)-SIGMAB(1))
      CN=CN+CND
      XP=0.5*(XGAMN(J-1)+XGAMN(J))
      CM=CM+CND*(XM-XP)
      A(1)=A(2)
      SIGMA(1)=SIGMA(2)
      SIGMAB(1)=SIGMAB(2)
   50 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TRJTRY
C      SUBROUTINE FOR DETERMINING THE PATHS OF FREE VORTICES IN THE
C         PRESENCE OF A WING-BODY COMBINATION                    260/C
      COMPLEX  SP,VP,SIGMA,EYE
      EXTERNAL FCT,OUTP
C
      DIMENSION GAM(4),YG(4),ZG(4),XBDY(100),RBDY(100),SBDY(100)
      DIMENSION PRMT(5),AUX(16,8),Y(8),Z(8)
      DIMENSION XP(100),YP(4),ZP(4),SP(100),VP(100) ,DRBDY(100)
      DIMENSION SIGMA(4),XV(200),YV(4,200),ZV(4,200),NVP(200),
     1          ARAY(5,200)
C
      COMMON /B/ MM,M
      COMMON /TRAJ1/ XI,XF,DXI,TITL,NPRINT
      COMMON /TRAJ2/ SP,VP,XP,NEXT
      COMMON /BODY/ NTBL,MTBL,XBDY,RBDY,SBDY,DRBDY
      COMMON /VORTEX/ NV,NV2,GAM,YG,ZG
      COMMON /VTABLE/ NTRY,XV,YV,ZV,NVP,ARAY
      COMMON /FLOW/ ALPHA,CSALP,EYE
      COMMON /PARAM/ DXOUT,XPRNT,MPRINT,XFINAL
C
  701 FORMAT (//I10)
  713 FORMAT (//5X67HEXECUTION STOP IN TRJTRY, INTEGRATION INTERVAL HAS
     1BECOME TOO SMALL //I4,1PE12.4//)
  715 FORMAT (///5X46HVELOCITY INDUCED AT SPECIFIED FIELD POINTS ON ,A4)
  716 FORMAT (/ 10X1HX,9X1HY,9X1HZ,11X8HV/V(INF),4X8HW/V(INF))
  717 FORMAT (I4,3F10.3,5X,2(1PE12.4))
  719 FORMAT (/10X,28HFIELD POINT OUTSIDE OF TABLE,5XI3,3F10.4,I9,F10.4)
C
      XFINAL=XF
      PRMT(1)=XI
      PRMT(2)=XF
      PRMT(3)=DXI
      PRMT(4)=.01
      NV2=NV+2
      NDIM=2*NV
      DO 10 J=1,NV
      JN=2*J
```

```
      Y(JN-1)=YG(J)
   10 Y(JN)=ZG(J)
      DUM=NDIM
      DUM=1./DUM
      DO 15 J=1, NDIM
   15 Z(J)=DUM
      XPRNT=XI=.0001
      CALL HPCG (PRMT,Y,Z,NDIM,IHLF,FCT,OUTP,AUX)
      IF (IHLF.GT.10) GO TO 60
      IF (NEXT.LE.0) RETURN
C
C       CALCULATE INDUCED VELOCITIES AT SPECIFIED FIELD POINTS
C
      NTBL=1
      DO 50 J=1,NEXT
      DO 51 K=1,NTRY
      KN=K
      IF (XV(K)-XP(J)) 51,52,53
   51 CONTINUE
   54 WRITE (6,719) J,XP(J),SP(J),KN,XV(KN)
      VP(J)=CMPLX(0.0,0.0)
      GO TO 50
   52 DO 55 KK=1,NV
   55 SIGMA(KK)=CMPLX(YV(KK,KN),ZV(KK,KN))
      GO TO 58
   53 IF (KN.EQ.1) GO TO 54
      DELTA=(XV(KN)-XP(J))/(XV(KN)-XV(KN-1))
      DO 56 KK=1,NV
      YP(KK)=YV(KK,KN)-DELTA*(YV(KK,KN)-YV(KK,KN-1))
      ZP(KK)=ZV(KK,KN)-DELTA*(ZV(KK,KN)-ZV(KK,KN-1))
   56 SIGMA(KK)=CMPLX(YP(KK),ZP(KK))
   58 CALL SHAPE (XP(J),A,S,DB)
      CALL EXTVEL (NV,A,SIGMA,GAM,SP(J),VP(J))
   50 CONTINUE
      IF (NPRINT.LT.0) GO TO 510
      WRITE (6,715) TITL
      WRITE (6,716)
  510 CONTINUE
      DO 40 J=1,NEXT
      VP(J)=CONJG(VP(J))
      IF (NPRINT.LT.0) GO TO 40
      WRITE (6,717) J,XP(J),SP(J),VP(J)
   40 CONTINUE
      RETURN
   60 WRITE (6,713) IHLF,H
      MM=IHLF
      RETURN
      END
```

```
      SUBROUTINE FCT (X,Y,Z)
      COMPLEX SIGMA,ENU,CENU,G,EYE,ADUM,BDUM,CDUM,DDUM,ENUN,ENUM,
     1        CENUN,CENUM,DNUDS,DWDS
      DIMENSION SIGMA(4),ENU(4),G(6,6),CENU(4),DWDS(4),DNUDS(4)
      DIMENSION Y(8),Z(8),GAM(4),YG(4),ZG(4)
      COMMON /B/ MM,H
      COMMON /VORTEX/ NV,NV2,GAM,YG,ZG
      COMMON /FLOW/ ALPHA,CSALP,EYE
      COMMON /RESULT/ G,DWDS,SIGMA,ENU,A,S,RZ,DS,DNUDS
C
      MM=0
C
C      LOOK UP BODY RADIUS AND LOCAL WING OR TAIL SEMISPAN
C
      CALL SHAPE (X,A,S,DS)
      RZ=0.5*(S + A*A/S)
      ARZ=A/RZ
      DO 9 J=1,NV
      K=2*J
    9 SIGMA(J)=CMPLX(Y(K-1),Y(K))
      DE=(2.*RZ)**2
      DO 10 J=1,NV
      CDUM=SIGMA(J) + A*A/SIGMA(J)
      ENU(J)=0.5*CDUM + 0.5*CSQRT(CDUM*CDUM - DE4)
   10 CENU(J)=CONJG(ENU(J))
C
C      COMPUTE G-FUNCTIONS FOR COMPLEX VELOCITY EXPRESSIONS
C
      DO 20 N=1,NV
      DWDS(N)=CMPLX(0.0,0.0)
      ENUN=ENU(N)/RZ
      CENUN=CENU(N)/RZ
      ADUM=ENUN + 1.0/ENUN
      BDUM=CSQRT(ADUM*ADUM - 4.*ARZ*ARZ)
      CDUM=(ENUN*ENUN-1.0)*(ADUM+BDUM)
      DNUDS(N)=2.0*ENUN*ENUN*BDUM/CDUM
C
C      CALCULATE CROSSFLOW CONTRIBUTION , G(NV+1,N) , INCLUDING EFFECT
C        OF BODY DA/DX
C
      G(NV+1,N)=-EYE*(1.0 + 1.0/(ENUN*ENUN))*DNUDS(N)*ALPHA
     1             + DS*A*CSALP/SIGMA(N)
C
C      CALCULATE TRANSFORMATION CONTRIBUTION , G(NV+2,N)
C
      DDUM=ADUM/BDUM - 1.0 - 2.0*ENUN*BDUM/((ENUN*ENUN-1.0)**2)
      DDUM=DDUM/(ADUM        + BDUM)/RZ
      G(NV+2,N)=-EYE*GAM(N)*DDUM
C
C      CALCULATE VORTEX INTERFERENCE CONTRIBUTION , G(M,N)
C
      DO 22 M=1,NV
      IF (M.EQ.N) GO TO 21
      ENUM=ENU(M)/RZ
      CENUM=CENU(M)/RZ
      DDUM=1.0/(ENUN-ENUM) + 1.0/(ENUN-1.0/ENUM) - 1.0/(ENUN-CENUM)
     1     - 1.0/(ENUN-1.0/CENUM)
      G(M,N)=-EYE*GAM(M)/RZ*DDUM*DNUDS(N)
      GO TO 22
   21 DDUM=1.0/ADUM - 1.0/(ENUN-CENUN) - 1.0/(ENUN-1.0/CENUN)
      G(M,N)=-EYE*GAM(N)/RZ*DDUM*DNUDS(N)
   22 CONTINUE
   20 CONTINUE
      DO 26 N=1,NV
      DO 25 M=1,NV2
   25 DWDS(N)=DWDS(N) + G(M,N)
   26 DWDS(N)=CONJG(DWDS(N))
      DO 27 N=1,NV
      N2=2*N
      Z(N2)= AIMAG(DWDS(N))
   27 Z(N2-1)=REAL(DWDS(N))
      RETURN
      END
```

```
      SUBROUTINE HPCG (PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
C
      DIMENSION PRMT(5),Y(8),DERY(8),AUX(16,8)
      COMMON /B/ MM,H
      N=1
      IHLF=0
      X=PRMT(1)
      H=PRMT(3)
      PRMT(5)=0.
      DO 1 I=1,NDIM
      AUX(16,I)=0.
      AUX(15,I)=DERY(I)
    1 AUX(1,I)=Y(I)
      IF(H*(PRMT(2)-X))3,2,4
C
C      ERROR RETURNS
    2 IHLF=12
      GO TO 4
    3 IHLF=13
C
C      COMPUTATION OF DERY FOR STARTING VALUES
```

```
    4 CALL FCT(X,Y,DERY)
      IF(MM.GT.1) RETURN
C
C     RECORDING OF STARTING VALUES
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))6,5,6
    5 IF(IHLF)7,7,6
    6 RETURN
    7 DO 8 I=1,NDIM
    8 AUX(8,I)=DERY(I)
C
C     COMPUTATION OF AUX(2,I)
      ISW=1
      GO TO 100
C
    9 X=X+H
      DO 10 I=1,NDIM
   10 AUX(2,I)=Y(I)
C
C     INCREMENT H IS TESTED BY MEANS OF BISECTION
   11 IHLF=IHLF+1
      X=X-H
      DO 12 I=1,NDIM
   12 AUX(4,I)=AUX(2,I)
      H=.5000*H
      N=1
      ISW=2
      GO TO 100
C
   13 X=X+H
      CALL FCT(X,Y,DERY)
      IF(MM.GT.1) RETURN
      N=2
      DO 14 I=1,NDIM
      AUX(2,I)=Y(I)
   14 AUX(9,I)=DERY(I)
      ISW=3
      GO TO 100
C
C     COMPUTATION OF TEST VALUE DELT
   15 DELT=0.
      DO 16 I=1,NDIM
   16 DELT=DELT+AUX(15,I)*ABS(Y(I)-AUX(4,I))
      DELT=.066666667*DELT
      IF(DELT-PRMT(4))19,19,17
   17 IF(IHLF-10)11,18,18
C
C     NO SATISFACTORY ACCURACY AFTER 10 BISECTIONS, ERROR MESSAGE.
   18 IHLF=11
      X=X+H
      GO TO 4
C
C     THERE IS SATISFACTORY ACCURACY AFTER LESS THAN 11 BISECTIONS.
   19 X=X+H
      CALL FCT(X,Y,DERY)
      IF(MM.GT.1) RETURN
      DO 20 I=1,NDIM
      AUX(3,I)=Y(I)
   20 AUX(10,I)=DERY(I)
      N=3
      ISW=4
      GO TO 100
C
   21 N=1
      X=X+H
      CALL FCT(X,Y,DERY)
      IF(MM.GT.1) RETURN
      X=PRMT(1)
      DO 22 I=1,NDIM
      AUX(11,I)=DERY(I)
   22 Y(I)=AUX(1,I)+H*(.375*AUX(8,I)+.79166667*AUX(9,I)-.20833333*AUX(10
     1,I)+.041666667*DERY(I))
   23 X=X+H
      N=N+1
      CALL FCT(X,Y,DERY)
      IF(MM.GT.1) RETURN
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))6,24,6
   24 IF(N-4)25,200,200
   25 DO 26 I=1,NDIM
      AUX(N,I)=Y(I)
   26 AUX(N+7,I)=DERY(I)
      IF(N-3)27,29,200
C
   27 DO 28 I=1,NDIM
      DELT=AUX(9,I)+AUX(9,I)
      DELT=DELT+DELT
   28 Y(I)=AUX(1,I)+.33333333*H*(AUX(8,I)+DELT+AUX(10,I))
      GO TO 23
C
   29 DO 30 I=1,NDIM
      DELT=AUX(9,I)+AUX(10,I)
      DELT=DELT+DELT+DELT
   30 Y(I)=AUX(1,I)+.375*H*(AUX(8,I)+DELT+AUX(11,I))
      GO TO 23
C
C
C     ******************************************************************
C     THE FOLLOWING PART OF SUBROUTINE DHPCG COMPUTES BY MEANS OF
C     RUNGE-KUTTA METHOD STARTING VALUES FOR THE NOT SELF-STARTING
C     PREDICTOR-CORRECTOR METHOD.
  100 DO 101 I=1,NDIM
      Z=H*AUX(N+7,I)
      AUX(5,I)=Z
  101 Y(I)=AUX(N,I)+.4*Z
C     Z IS AN AUXILIARY STORAGE LOCATION
C
      Z=X+.4*H
      CALL FCT(Z,Y,DERY)
      IF(MM.GT.1) RETURN
      DO 102 I=1,NDIM
      Z=H*DERY(I)
      AUX(6,I)=Z
  102 Y(I)=AUX(N,I)+.29697761*AUX(5,I)+.15875964*Z
C
      Z=X+.45573725*H
      CALL FCT(Z,Y,DERY)
      IF(MM.GT.1) RETURN
      DO 103 I=1,NDIM
      Z=H*DERY(I)
      AUX(7,I)=Z
  103 Y(I)=AUX(N,I)+.21810039*AUX(5,I)-3.0509651*AUX(6,I)+3.8328647*Z
C
      Z=X+H
      CALL FCT(Z,Y,DERY)
      IF(MM.GT.1) RETURN
      DO 104 I=1,NDIM
  104 Y(I)=AUX(N,I)+.17476028*AUX(5,I)-.55148066*AUX(6,I)+1.2055356*AUX(
     17,I)+.17118478*H*DERY(I)
      GO TO(9,13,15,21),ISW
C     ******************************************************************
C
C
C     POSSIBLE BREAK-POINT FOR LINKAGE
C
C
C     STARTING VALUES ARE COMPUTED.
C     NOW START HAMMINGS MODIFIED PREDICTOR-CORRECTOR METHOD.
  200 ISTEP=3
  201 IF(N-8)204,202,204
C
C     N=8 CAUSES THE ROWS OF AUX TO CHANGE THEIR STORAGE LOCATIONS
  202 DO 203 N=2,7
      DO 203 I=1,NDIM
      AUX(N-1,I)=AUX(N,I)
  203 AUX(N+6,I)=AUX(N+7,I)
      N=7
C
C     N LESS THAN 8 CAUSES N+1 TO GET N
  204 N=N+1
C
C     COMPUTATION OF NEXT VECTOR Y
      DO 205 I=1,NDIM
      AUX(N-1,I)=Y(I)
```

```
  205 AUX(N+6,I)=DERY(I)
      X=X+H
  206 ISTEP=ISTEP+1
      DO 207 I=1,NDIM
      DELT=AUX(N-4,I)+1.33333333*H*(AUX(N+6,I)+AUX(N+6,I)-AUX(N+5,I)+AUX
     1(N+4,I)+AUX(N+4,I))
      Y(I)=DELT-.92561983*AUX(16,I)
  207 AUX(16,I)=DELT
C     PREDICTOR IS NOW GENERATED IN ROW 16 OF AUX, MODIFIED PREDICTOR
C     IS GENERATED IN Y, DELT MEANS AN AUXILIARY STORAGE.
C
      CALL FCT(X,Y,DERY)
      IF (MM.GT.1) RETURN
C     DERIVATIVE OF MODIFIED PREDICTOR IS GENERATED IN DERY
C
      DO 208 I=1,NDIM
      DELT=.12500*(9.00*AUX(N-1,I)-AUX(N-3,I)+3.00*H*(DERY(I)+AUX(N+6,I)
     1+AUX(N+6,I)-AUX(N+5,I)))
      AUX(16,I)=AUX(16,I)-DELT
  208 Y(I)=DELT+.07438016*AUX(16,I)
C
C     TEST WHETHER H MUST BE HALVED OR DOUBLED
      DELT=0.
      DO 209 I=1,NDIM
  209 DELT=DELT+AUX(15,I)*ABS(AUX(16,I))
      IF(DELT-PRMT(4))210,222,222
C
C     H MUST NOT BE HALVED. THAT MEANS Y(I) ARE GOOD.
  210 CALL FCT(X,Y,DERY)
      IF(MM.GT.1) RETURN
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))212,211,212
  211 IF(IHLF-11)213,212,212
  212 RETURN
  213 IF(H*(X-PRMT(2)))214,212,212
  214 IF(ABS(X-PRMT(2))-.1*ABS(H))212,215,215
  215 IF(DELT-.02*PRMT(4))216,216,201
C
C
C     H COULD BE DOUBLED IF ALL NECESSARY PRECEDING VALUES ARE
C     AVAILABLE
  216 IF(IHLF)201,201,217
  217 IF(N-7)201,218,218
  218 IF(ISTEP-4)201,219,219
  219 IMOD=ISTEP/2
      IF(ISTEP-IMOD-IMOD)201,220,201
  220 H=H+H
      IHLF=IHLF-1
      ISTEP=0
      DO 221 I=1,NDIM
      AUX(N-1,I)=AUX(N-2,I)
      AUX(N-2,I)=AUX(N-4,I)
      AUX(N-3,I)=AUX(N-6,I)
      AUX(N+6,I)=AUX(N+5,I)
      AUX(N+5,I)=AUX(N+3,I)
      AUX(N+4,I)=AUX(N+1,I)
      DELT=AUX(N+6,I)+AUX(N+5,I)
      DELT=DELT+DELT+DELT
  221 AUX(16,I)=8.962962962*(Y(I)-AUX(N-3,I))-3.361111111*H*(DERY(I)+DELT+
     1AUX(N+4,I))
      GO TO 201
C
C
C     H MUST BE HALVED
  222 IHLF=IHLF+1
      IF(IHLF-10)223,223,210
  223 H=.5*H
      ISTEP=0
      DO 224 I=1,NDIM
      Y(I)=.390625E-2*(8.E1*AUX(N-1,I)+135.00*AUX(N-2,I)+4.E1*AUX(N-3,I)
     1+AUX(N-4,I))-.1171875*(AUX(N+6,I)-6.*AUX(N+5,I)-AUX(N+4,I))*H
      AUX(N-4,I)=.390625E-2*(12.00*AUX(N-1,I)+135.00*AUX(N-2,I)+
     1108.00*AUX(N-3,I)+AUX(N-4,I))-.0234375*(AUX(N+6,I)+
     218.00*AUX(N+5,I)-9.00*AUX(N+4,I))*H
      AUX(N-3,I)=AUX(N-2,I)
  224 AUX(N+4,I)=AUX(N+5,I)
      X=X-H
      DELT=X-(H+H)
      CALL FCT(DELT,Y,DERY)
      IF(MM.GT.1) RETURN
      DO 225 I=1,NDIM
      AUX(N-2,I)=Y(I)
      AUX(N+5,I)=DERY(I)
  225 Y(I)=AUX(N-4,I)
      DELT=DELT-(H+H)
      CALL FCT(DELT,Y,DERY)
      IF(MM.GT.1) RETURN
      DO 226 I=1,NDIM
      DELT=AUX(N+5,I)+AUX(N+4,I)
      DELT=DELT+DELT+DELT
      AUX(16,I)=8.962962962*(AUX(N-1,I)-Y(I))
     1-3.361111111*H*(AUX(N+6,I)+DELT+DERY(I))
  226 AUX(N+3,I)=DERY(I)
      GO TO 206
      END


      SUBROUTINE OUTP (X,Y,Z,IHLF,NDIM,PRMT)
      COMPLEX G,SIGMA,ENU,DWDS,DNUDS
      DIMENSION Y(8),Z(8),PRMT(5),G(6,6),DWDS(4),SIGMA(4),ENU(4),
     1          GAM(4),YG(4),ZG(4),DNUDS(4),
     2          XV(200),YV(4,200),ZV(4,200),NVP(200),ARAY(5,200)
      COMMON /VORTEX/ NV,NV2,GAM,YG,ZG
      COMMON /RESULT/ G,DWDS,SIGMA,ENU,A,B,RZ,DS,DNUDS
      COMMON /PARAM/ DXOUT,XPRNT,MPRINT,XFINAL
      COMMON /B/ MM,H
      COMMON /VTABLE/ NTRY,XV,YV,ZV,NVP,ARAY
C
      IF (X.LT.XPRNT) RETURN
  700 FORMAT (//5X1HX,8X2HDX,11X1HA,9X1HB,8X2HRO,6X5HDA/DX)
  701 FORMAT (F9.4,F11.7,4F10.4)
  702 FORMAT (/1X6HVORTEX,1X11HSIGMA(REAL),3X6H(IMAG),5X8HNU(REAL),
     1          4X6H(IMAG),3X11HDW/DS(REAL),3X6H(IMAG),
     2          2X12HDNU/DS(REAL),3X6H(IMAG))
  703 FORMAT (I4,2X4(1PE12.3,E11.3)/5X,6(E11.3,E10.3))
      IF (MPRINT.LE.0) GO TO 15
      WRITE (6,700)
      WRITE (6,701) X,H,A,B,RZ,DS
      WRITE (6,702)
      DO 10 N=1,NV
   10 WRITE (6,703) N,SIGMA(N),ENU(N),DWDS(N),DNUDS(N),(G(M,N),M=1,NV2)
   15 XPRNT=X+DXOUT
      IF (XPRNT.GE.XFINAL) XPRNT=XFINAL-H
      NTRY=NTRY+1
      IF (NTRY.GT.200) RETURN
C
C     SAVE TABLE OF VORTEX POSITIONS
C
      XV(NTRY)=X
      NVP(NTRY)=NV
      ARAY(1,NTRY)=H
      ARAY(2,NTRY)=A
      ARAY(3,NTRY)=B
      ARAY(4,NTRY)=RZ
      ARAY(5,NTRY)=DS
      DO 20 N=1,NV
      YV(N,NTRY)=REAL(SIGMA(N))
   20 ZV(N,NTRY)=AIMAG(SIGMA(N))
      RETURN
      END
```

```
      SUBROUTINE EXTVEL (NV,A,SIGMA,GAM,S,V)
C
C     COMPUTE VELOCITY FIELD INDUCED BY VORTEX PAIRS AND THEIR IMAGES
C
      COMPLEX S,V,SIGMA,SPR,SGM,SGMC,EYE,CD1
      DIMENSION SIGMA(4),GAM(4)
      COMMON /FLOW/ ALPHA,CSALP,EYE
C
      SPR=S/A
      V=CMPLX(0.0,0.0)
      DO 10 J=1,NV
      SGM=SIGMA(J)/A
      SGMC=CONJG(SGM)
      CD1=SPR-SGM
      D3=1.0
      D2=CABS(CD1)
      IF (D2.GT.1.0E-04) GO TO 20
      CD1=CMPLX(1.0,0.0)
      D3=0.0
   20 CONTINUE
      V=V - EYE*GAM(J)/A*(  D3/CD1        + 1.0/(SPR+1.0/SGM)
     1                  - 1.0/(SPR+SGMC) - 1.0/(SPR-1.0/SGMC))
   10 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE VTXLAT (FVN,NMX,NMY,NZ,NCW,MSW,CRW,CTW,SSPAN,ALPHAD,
     1                  PSIWLE,PSIWTE,PHID,SREF,CAVE,Y,YLOC,SUMY,XLEL,
     2                  CHLOCB,SWPVVP,SW,PVX,PVY,PVZ,PTLX,PTLY,PTLZ,
     3                  FUW,FVW,FWW,UEI,VEI,WEI,ALPHAL,XM,ZM,PCX,
     4                  NSEPV,YSEPV,BETA,AKVL1,AKVL2,AKVSE)
C              VORTEX LATTICE PROGRAM
C
      DIMENSION FVN(NMX,NMY)
      DIMENSION PSIWLE(NZ),PSIWTE(NZ),Y(NZ),YLOC(NZ),SUMY(NZ),XLEL(NZ),
     1          CHLOCB(NZ)
      DIMENSION SWPVVP(NMX),SW(NMX),PVX(NMX),PVY(NMX),PVZ(NMX),
     1          PTLX(NMX),PTLY(NMX),PTLZ(NMX),ALPHAL(NMX),
     2          UEI(NMX),VEI(NMX),WEI(NMX),PCX(NMX)
      DIMENSION FUW(2),FVW(2),FWW(2),TIPSFC(10),AKVLE(2)
      DIMENSION CIR(100),SLDCO(25),SECLIF(25),SECSUC(25),SECSF(25),
     1          SECTHR(25),SWPWLE(25),SWPWTE(25) , Y1(3),GAM1(2)
C
      COMMON /VL1/ CIR,SLDCO,SECLIF,SECSUC,SECSF,SECTHR,SWPWLE,SWPWTE
      COMMON /VL3/ SUMLIF,SUMSUC,TIPSFC,FSLDCO,FECLIF,AKVLP,AKVSP
      COMMON /VL5/ NPRINT,MMM,CNORM,CNV,CNVS,XCOPP,XCOPLE,XCOPSE,
     1            YTR,GAMT,Y1,GAM1,CNV1,CNV2,NSEP
      COMMON /VL7/ CLTOT,CNTOT,TOTMOM
      COMMON/INF/BOT
C
C
    1 FORMAT (1H1,27X,32HHORSESHOE VORTEX CHARACTERISTICS//
     1  5X29HBOUNDLEG MIDPOINT COORDINATES,28X16HVORTEX STRENGTHS/ 5X
     21HJ,6X,1HX,10X,1HY,9X,1HZ, 8X5HSWEEP,3X9HSEMIWIDTH,10X7HGAMMA/V)
   10 FORMAT (/5X7HSTATION,5X1HY,4X8HSWEEP LE,2X8HSWEEP TE)
   11 FORMAT (I10,3F10.3)
   13 FORMAT (I4,1X,F9.4,2X,F9.4,2X,F8.4,2X,F8.4,2X,F8.4,11X,F8.4)
   21 FORMAT (//12X48HGEOMETRY ADJUSTED FOR COMPRESSIBILITY CORRECTION)
   22 FORMAT (//4X7HC(ROOT),3X6HC(TIP),5X5HB/2,6X8HDIHEDRAL,5X3HXCM,7X
     1        3HZCM,5X6HY(SEP),3X8HETA(SEP)/8F10.3)
   23 FORMAT (3X,I3,5X,F8.4,4X,F8.4,8X,F8.4,9X,F8.4,9X,F8.4,
     19X,F8.4,9X,F8.4,4X,F8.4)
   34 FORMAT (1H1,4X,36HCONTROL POINT COORDINATES AND SLOPES,11X,
     129HEXTERNALLY INDUCED VELOCITIES)
   35 FORMAT (I4,1X,F9.4,2X,F9.4,2X,F8.4,2X,F8.4,6X,F9.5,1X,F9.5,1X,F9.5
     1,1X,F9.4,2X,F9.4,2X,F8.4)
   51 FORMAT (36X,12HMEAN SURFACE/3X,1HJ,6X,1HX,10X1HY,9X,1HZ,8X,5HSLOPE
     1,10X,3HU/V,7X,3HV/V,7X,3HW/V)
   59 FORMAT (12X27HLIFTING SURFACE CALCULATION )
   60 FORMAT (/5X21HLATTICE ARRANGEMENT =,I3,14H CHORDWISE AND,I3,
     1       19H SPANWISE ELEMENTS  )
   65 FORMAT(/10X,26HSPANWISE LOAD DISTRIBUTION//2X,7HSTATION,3X,7HY/(B/
     12),3X,13HLOCAL CHORD,C,4X,12HCL*C/CL*CAVG, 6X,10HCL*C/(2*B),
     2 6X,10HCX*C/(2*B), 6X,10HCY*C/(2*B),6X,12HCSUC*C/(2*B),3X,
     38HETA,DEG.  )
   66 FORMAT(/5X,53HREFERENCE QUANTITIES =  PLANFORM AREA   AVERAGE CHOR
     1D/29X,3HS =,F10.3,3X6HCAVG =,F7.2)
   67 FORMAT (//5X14H***PREDICTED**,4X18HTRAILING VORTICITY,2X20HSEPARAT
     1ION VORTICITY/5X4HKVLE,6X4HKVSE,7X1HY5X8HGAM/V2PI,6X1HY,5X8HGAM/V2
     2PI )
   68 FORMAT (2F10.5,2(F10.4,F10.5)/40XF10.4,F10.5)
   72 FORMAT (1H1,33X,27HAERODYNAMIC LOADING RESULTS//)
   78 FORMAT (/5X17HFLOW CONDITIONS =,3X5HALPHA,5X4HBETA/20X,F10.3,
     1         F10.4)
      PI=3.1415926
      DTR=PI/180.0
      BOT= SSPAN
      IMAX=MSW+1
      M=MSW*NCW
      MP1=M+1
      IF (YSEPV.LE.0.0) NSEPV=1
      IF (NSEPV.GT.1 .AND. YSEPV.GE.SSPAN) NSEPV=1
      IF (NSEPV.LE.0 .OR. NSEPV.GT.2) NSEPV=1
      IF (NSEPV.EQ.1) YSEPV=SSPAN
      ETASEP=YSEPV/SSPAN
      AKVLE(1)=AKVL1
      AKVLE(2)=AKVL2
C
      PHI= PHID*DTR
      SPHI=SIN(PHI)
      CSPHI=COS (PHI)
      DO 110 I=1,IMAX
      IF (ETASEP.GT.YLOC(I)) NSEP=I
      SWPWLE(I)=TAN(PSIWLE(I)*DTR)
  110 SWPWTE(I)=TAN(PSIWTE(I)*DTR)
      IF (NSEPV.EQ.1 .AND. NSEP.LT.IMAX) NSEP=IMAX
C
      DO 111 JC=1,100
  111 CIR(JC)=0.0
C
      ALPHA= ALPHAD*DTR
      COSALP= COS(ALPHA)
C
      IF (NPRINT.LT.0) GO TO 510
      WRITE (6,59)
      IF (BETA.LT.1.0) WRITE (6,21)
      WRITE (6,78) ALPHAD,BETA
      WRITE (6,60) NCW,MSW
      WRITE (6,66) SREF,CAVE
      WRITE (6,22) CRW,CTW,SSPAN,PHID,XM,ZM,YSEPV,ETASEP
      WRITE (6,10)
      DO 113 I=2,IMAX
      JC=I-1
  113 WRITE (6,11)JC,Y(I),PSIWLE(I),PSIWTE(I)
  510 CONTINUE
C
C     BUILD-UP OF R.H.S. OF BOUNDARY CONDITION
C
      DO 2612 JC=1,M
 2612 CIR(JC)=12.566371*(SIN(ALPHA+ALPHAL(JC)) *CSPHI+
     1             VEI(JC)*SPHI-(UEI(JC)*  ALPHAL(JC)+WEI(JC))*CSPHI)
C
      DO 9811 JC=1,M
 9811 FVN(JC,MP1)=CIR(JC)
C
C     SOLVE FOR UNKNOWN VORTEX STRENGTHS
C
      CALL INVERS (FVN,1,M,NMX,NMY)
C
      DO 9812 JC=1,M
 9812 CIR(JC)=FVN(JC,MP1)
C
C     LOADING CALCULATIONS
C
      CALL LOAD1 (NCW,M,MP1,IMAX,NMX,NZ,Y,YLOC,CHLOCB,XLEL,CRW,CTW,
     1                SSPAN,PHI,CSPHI,SPHI,ALPHA,SREF,CAVE,
     2                PTLX,PTLY,PTLZ,PVX,PVY,PVZ,SW,SWPVVP,
     3                UEI,VEI,WEI,XM,ZM,BETA,AKVLE)
      IF (NPRINT.LT.10) GO TO 551
```

```
C
C     CONTROL POINTS COORDINATES, MEAN SURFACE SLOPES AND EXTERNALLY
C     INDUCED VELOCITIES OUTPUT
C
      WRITE (6,50)
      WRITE (6,51)
C
      DO 6001 J=1,M
 6001 WRITE (6,55) J,PCX(J),PVY(J),PVZ(J),ALPHAL(J),UEI(J),VEI(J),WEI(J)
C
C     HORSESHOE VORTEX CHARACTERISTICS OUTPUT
C
      WRITE (6,1)
      DO 550 J=1,M
      SWPVVD=       SWPVVP(J) *57.29578
  550 WRITE (6,13) J,PVX(J),PVY(J),PVZ(J),SWPVVD    ,SW(J),CIR(J)
  551 CONTINUE
C
C     OUTPUT SPAN LOADING COEFFICIENTS
C
      IF (NPRINT.GE.10) WRITE (6,72)
      IF (NPRINT.GE.0) WRITE (6,65)
      DO 3000 I=2,IMAX
      IM1=I-1
      ETA= PSIWLE(I)+(ATAN2(SECSUC(I),-SECSF(I)))*57.2957795
      IF (NPRINT.LT.0) GO TO 3000
      WRITE (6,23) IM1,YLOC(I),CHLOCB(I),SLDCO(I),SECLIF(I),SECSUC(I),
     1SECSF(I),SECTHR(I),ETA
 3000 CONTINUE
C
C      CALCULATE POSITION AND STRENGTH OF SEPARATION VORTEX
C       AND TRAILING VORTEX
C
      Y1(1)=0.0
      Y1(2)=0.0
      Y1(3)=0.0
      GAM1(1)=0.0
      GAM1(2)=0.0
      CNV2=0.0
      YBD=0.0
      YTR=0.0
      SFM=1.0
      SF=0.0
      IF (CTW.LT. 0.01*CRW) SFM=0.0
      DUM=NCW
      DUM=CTW/DUM
      DO 305 J=1,NCW
      TIPSFC(J)=-TIPSFC(J)*SFM
  305 SF=SF+TIPSFC(J)*DUM
      SF=Y(IMAX)*SF
      KSEP=IMAX
      DO 300 I=2,IMAX
      IF (ETASEP.GT.YLOC(I)) KSEP=I
      YBD=YBD+SECTHR(I)*(Y(I)-Y(I-1))
  300 YTR=YTR+SECLIF(I)*(Y(I)-Y(I-1))
      YTR=YTR/SECLIF(2)
      GAMT=CNORM*SREF*COSALP/(8.0*PI*YTR)
      II=2
      DO 303 J=1,NSEPV
      DUM=0.0
      IF (KSEP.EQ.IMAX) DUM=1.0
      Y1D=0.
      DO 301 I=II,KSEP
      Y1(J)=Y1(J)+SECTHR(I)* YLOC(I)*SSPAN*(Y(I)-Y(I-1))
      Y1D=Y1D+SECTHR(I)*(Y(I)-Y(I-1))
  301 CONTINUE
      CNV1=CNV
      IF (KSEP.LT.IMAX) CNV2=Y1D/YBD
      IF (KSEP.LT.IMAX) II=KSEP+1
      KSEP=IMAX
      IF (AKVLE(J).LE.0.0 .AND. DUM*AKVSE.LE.0.0) GO TO 313
      GAM1(J)=AKVLE(J)*CNV*Y1D  /YBD + DUM*AKVSE*CNVS
      Y1(J)=(Y1(J)*AKVLE(J)+DUM*AKVSF*SFM)/(Y1D*AKVLE(J)+DUM*AKVSE*SF)
      GO TO 303
  313 Y1(J)=SSPAN
  303 CONTINUE
      DO 304 J=1,NSEPV
      DUM=0.0
      IF (Y1(J).GT.0.0) DUM=1.0/(8.0*PI*Y1(J))
  304 GAM1(J)=GAM1(J)*SREF*COSALP*DUM
      IF (NPRINT.LT.0) GO TO 513
  306 WRITE (6,67)
      WRITE (6,68) AKVLP,AKVSP,YTR,GAMT,(Y1(J),GAM1(J),J=1,NSEPV)
  513 CONTINUE
      IF (NSEPV.LE.1) RETURN
      CNV1=CNV*CNV2
      CNV2=CNV-CNV1
C
      RETURN
      END
```

```
      SUBROUTINE LOAD1 (NCW,M,MP1,IMAX,NMX,NZ,Y,YLOC,CHLOCB,XLEL,CRW,
     1                  CTW,SSPAN,PHI,CSPHI,SPHI,ALPHA,SREF,CAVE,
     2                  PTLX,PTLY,PTLZ,PVX,PVY,PVZ,SW,SWPVVP,
     3                  UEI,VEI,WEI,XM,ZM,BETA,AKVLE)
C
      DIMENSION Y(NZ),YLOC(NZ),CHLOCB(NZ),XLEL(NZ)
      DIMENSION PTLX(NMX),PTLY(NMX),PTLZ(NMX),PVX(NMX),PVY(NMX),PVZ(NMX)
      DIMENSION SW(NMX),SWPVVP(NMX),UEI(NMX),VEI(NMX),WEI(NMX)
      DIMENSION BLLIFT(100),BLDRAG(100),TLLIFT(100),TLDRAG(100),
     1        BLSF(100),TLSF(100),
     2        BLLIF1(100),BLLIF2(100),BLLIF3(100),BLLIF4(100),BLLIF5(100)
     3 ,      BLDRG2(100),BLDRG3(100),BLDRG4(100),BLDRG5(100),
     4        BLSF1(100),BLSF2(100),BLSF3(100),BLSF4(100),
     5        TLSF1(100),TLSF2(100)
      DIMENSION U(100),V(100),W(100),VB(100),WB(100),T(100),ELPANL(100)
      DIMENSION CHLIFT(25),CHDSUC(25),CHRDSF(25),SLDCO1(25),CHLOC(25),
     1      CHDTHR(25),YARM(25),XMOM(25),CHDRAG(25)
      DIMENSION TIPSF(10),TIPSFC(10),YTW(2),YBW(2),FUW(2),FVW(2),FWW(2),
     1          Y1(3),GAM1(2),AKVLE(2)
      DIMENSION CIR(100),SLDCO(25),SECLIF(25),SECSUC(25),SECSF(25),
     1          SECTHR(25),SWPWLE(25),SWPWTE(25)
C
      COMMON /VL1/ CIR,SLDCO,SECLIF,SECSUC,SECSF,SECTHR,SWPWLE,SWPWTE
      COMMON /VL3/ SUMLIF,SUMSUC,TIPSFC,FSLDCO,FECLIF,AKVLP,AKVSP
      COMMON /VL5/ NPRINT,MMM,CNORM,CNV,CNVS,XCOPP,XCOPLE,XCOPSE,
     1             YTR,GAMT,Y1,GAM1,CNV1,CNV2,NSEP
      COMMON /VL7/ CLTOT,CNTOT,TOTMOM
      COMMON/INF/BOT
C
    1 FORMAT (1H1,17HBLLIFT COMPONENTS//5X,7HBLLIF1 ,5X,7HBLLIF2 ,5X,
     27HBLLIF3 ,5X,7HBLLIF4 ,5X,7HBLLIF5 ,5X,6HBLLIFT/)
    2 FORMAT (/1X,17HTLLIFT COMPONENTS/)
    3 FORMAT (1X,10E12.4)
    4 FORMAT (/1X,35H V/V AT 3/4 ELEM.CHORD OF LEFT TAIL/)
    5 FORMAT (/1X,25H U/V AT BOUNDLEG MIDPOINT/)
    6 FORMAT (/1X,25H V/V AT BOUNDLEG MIDPOINT/)
   14 FORMAT (//  5X3HCNP,7X3HCNV,7X4HCNVS,6X3HCLP,7X3HCMP,7X3HCOI,
     1         7X2HCA,6X7HCDI/CL2,5X3HCXV,7X3HCYV,7X3HCRP)
   15 FORMAT(11F10.5)
   16 FORMAT (6E12.4)
   17 FORMAT (5E12.6)
   20 FORMAT (/1X,17HBLDRAG COMPONENTS//5X,7HBLDRG2 ,5X,7HBLDRG3 ,5X,
     17HBLDRG4 ,5X,7HBLDRG5 ,5X,6HBLDRAG/)
   21 FORMAT (/1X,17HTLDRAG COMPONENTS/)
   22 FORMAT (/1X,25H W/V AT BOUNDLEG MIDPOINT/)
   23 FORMAT (/1X,35H W/V AT 3/4 ELEM.CHORD OF LEFT TAIL/)
   24 FORMAT (//1X,15HBLSF COMPONENTS//5X,7HBLSF1  ,5X,7HBLSF2  ,5X,
     17HBLSF3  ,5X,7HBLSF4  ,5X,6HBLSF/)
   25 FORMAT (//1X,15HTLSF COMPONENTS//5X,7HTLSF1  ,5X,7HTLSF2  ,5X,
     14HTLSF/)
   26 FORMAT (1X,3E12.4)
   33 FORMAT (    //10X,43HX COORDINATES OF CENTERS OF PRESSURE DUE TO/
     12X,14HPOTENTIAL LIFT,4X,20HLEADING EDGE SUCTION,3X,
     217HSIDE EDGE SUCTION,4X,7HOVERALL /5X,F10.4,10X,F10.4,10X,F10.4,
     3  7X,3F10.4)
C
       BETA2=BETA*BETA
       SINALP= SIN(ALPHA)
```

```
      COSALP= COS(ALPHA)
      DW=NCW
      I=1
      SUMLIF=0.0
      SUMDRG=0.0
      SUMSF=0.0
      SUMSUC=0.0
      SUMTHR=0.0
      SMMOMX=0.0
      SMCTHR=0.0
  702 I=I+1
      CHLIFT(I)=0.0
      CHDRAG(I)=0.0
      CHRDSF(I)=0.0
      CHDSUC(I)=0.0
C
C     COMPUTE VELOCITIES NEEDED FOR LOADING CALCULATIONS
C
C     SIDEWASH  V  INDUCED BY WING VORTICES AT THE 3/4 CHORD OF THE LEFT
C     TRAILING WING VORTEX LEGS
C
      DO 823 NV=1,NCW
      JT=(I-2)*NCW+NV
C
C     ADD EXTERNALLY INDUCED VELOCITY VEI
C
      V(JT)=VEI(JT)
      W(JT)=WEI(JT)
      DO 823 NN=1,M
      XTW=PTLX(JT)-PVX(NN)
      YTW(1)=PTLY(JT)-PVY(NN)
      YTW(2)=PTLY(JT)+PVY(NN)
      ZTW=PTLZ(JT)-PVZ(NN)
      DO 271 L=1,2
      CALL INFWW(SWPVVP(NN),PHI,XTW,YTW(L),ZTW,SW(NN),FUW(L),FVW(L),
     1FWW(L))
      SWPVVP(NN)=-SWPVVP(NN)
      PHI=-PHI
  271 CONTINUE
      V(JT)=V(JT)+((FVW(1)+FVW(2))*CIR(NN))/12.566371
      W(JT)=W(JT)+((FWW(1)+FWW(2))*CIR(NN))/12.566371
  823 CONTINUE
C
C     VELOCITIES  U  AND  V  INDUCED AT WING VORTEX BOUND LEG MIDPOINT
C     INDUCED BY WING VORTICES
C
      DO 826 NV=1,NCW
      JB=(I-2)*NCW+NV
C
C     ADD EXTERNALLY INDUCED VELOCITIES UEI,VEI
C
      U(JB)=UEI(JB)
      VB(JB)=VEI(JB)
      WB(JB)= WEI(JB)
      DO 826 NN=1,M
      XBW=PVX(JB)-PVX(NN)
      YBW(1)=PVY(JB)-PVY(NN)
      YBW(2)=PVY(JB)+PVY(NN)
      ZBW=PVZ(JB)-PVZ(NN)
      DO 268 L=1,2
      CALL INFWW(SWPVVP(NN),PHI,XBW,YBW(L),ZBW,SW(NN),FUW(L),FVW(L),
     1FWW(L))
      SWPVVP(NN)=-SWPVVP(NN)
      PHI=-PHI
  268 CONTINUE
      U(JB)=U(JB)+((FUW(1)+FUW(2))*CIR(NN))/12.566371
      VB(JB)=VB(JB)+((FVW(1)+FVW(2))*CIR(NN))/12.566371
      WB(JB)=WB(JB)+((FWW(1)+FWW(2))*CIR(NN))/12.566371
  826 CONTINUE
C
C     CALCULATE FORCES ON CHORDWISE ROW OF ELEMENTAL PANELS ON THE WING
C
      CHLOC(I)=CRW-Y(I)*(SWPWLE(I)-SWPWTE(I))
      CIRNET=0.0
      DO 829 KCW=1,NCW
      JT=(I-2)*NCW+KCW
      JS=(I-1)*NCW+KCW
      T(JT)=SSPAN/(SW(JT)*CSPHI)
      IF (I.EQ.IMAX) GO TO 2010
      CIRNET=CIRNET+CIR(JT)-CIR(JS)
      GO TO 2011
 2010 CIRNET=CIRNET+CIR(JT)
 2011 FAC=1.0
      IF (KCW.EQ.NCW) FAC=0.75
      ELPANL(JT)=CHLOC(I)/DW
C
C     TLLIFT=LIFT ACTING ON LEFT LEG OF WING VORTEX TAIL AT THE 3/4
C     ELEMENTAL PANEL CHORD
C     TLDRAG=FORCE ACTING ON LEFT LEG OF WING VORTEX TAIL AT THE 3/4
C     ELEMENTAL PANEL CHORD,IT ACTS ALONG VINF LINE AND POINTS FORWARD
C
      TLLIFT(JT)=ELPANL(JT)*FAC*V(JT)*(2.0/SREF)*CIRNET*COSALP
C
      TLDRAG(JT)=ELPANL(JT)*FAC*V(JT)*(-2.0/SREF)*CIRNET*SINALP
C
      TLSF1(JT)=ELPANL(JT)*FAC*W(JT)*(2.0/SREF)*CIRNET
      TLSF2(JT)=ELPANL(JT)*FAC*(-SINALP)*(2.0/SREF)*CIRNET
      TLSF(JT)=TLSF1(JT)+TLSF2(JT)
C
C     BLLIFT = LIFT ACTING ON BOUND LEG OF THE WING VORTEX AT THE
C     MIDPOINT
C     BLDRAG= -DRAG FORCE ACTING AT SAME POINT
C
      SWPVVT= TAN(SWPVVP(JT))
      FACLDW= (2.0/SREF)*2.0*SW(JT)*CIR(JT)
      BLLIF1(JT)= FACLDW*CSPHI
      BLLIF2(JT)= FACLDW*(-U(JT)*COSALP)*CSPHI
      BLLIF3(JT)= FACLDW*VB(JT)*SWPVVT     *COSALP
      BLLIF4(JT)= FACLDW*VB(JT)*SPHI*SINALP
      BLLIF5(JT)= FACLDW*(-WB(JT)) *CSPHI*SINALP
      BLLIFT(JT)=BLLIF1(JT)+BLLIF2(JT)+BLLIF3(JT)+BLLIF4(JT)+BLLIF5(JT)
C
      BLDRG2(JT)= FACLDW*U(JT)*SINALP*CSPHI
      BLDRG3(JT)= FACLDW*(-VB(JT))*SWPVVT     *SINALP
      BLDRG4(JT)= FACLDW*VB(JT)*SPHI*COSALP
      BLDRG5(JT)= FACLDW*(-WB(JT))*CSPHI*COSALP
      BLDRAG(JT)= BLDRG2(JT)+BLDRG3(JT)+BLDRG4(JT)+BLDRG5(JT)
C
      BLSF1(JT)=FACLDW*COSALP*SPHI
      BLSF2(JT)=FACLDW*(-SINALP*SWPVVT)
      BLSF3(JT)=FACLDW*(-U(JT)*SPHI)
      BLSF4(JT)=FACLDW*WB(JT)*SWPVVT
      BLSF(JT)=BLSF1(JT)+BLSF2(JT)+BLSF3(JT)+BLSF4(JT)
C
      CHLIFT(I)=CHLIFT(I)+(TLLIFT(JT)     )+BLLIFT(JT)
      CHDRAG(I)= CHDRAG(I)+(TLDRAG(JT)     )+BLDRAG(JT)
      CHRDSF(I)=CHRDSF(I)+                BLSF(JT)
      CHDSUC(I)= CHDSUC(I)+(           BLLIFT(JT))*SINALP+
     1(           BLDRAG(JT))*COSALP
      SLOCO1(I)=T(JT)*CHLIFT(I)
C
C     SECLIF=(SECTION LIFT*LOCAL CHORD)/2*SPAN
C
      SECLIF(I)=CHLIFT(I)*(SREF/(2.0*SW(JT)*CSPHI*4.0*SSPAN))
      SECSUC(I)= CHDSUC(I)*(SREF/(2.0*SW(JT)*CSPHI*4.0*SSPAN))
      SECSF(I)= CHRDSF(I)*(SREF/(2.0*SW(JT)*CSPHI*4.0*SSPAN))
      SECTHR(I)= SQRT (SECSUC(I)*SECSUC(I)+SECSF(I)*SECSF(I))
  829 CONTINUE
C
C     SPANWISE SUMMING UP OF LIFTS ACTING ON CHORDWISE ROWS
C
      SUMLIF=SUMLIF+2.0*CHLIFT(I)
      SUMDRG= SUMDRG+2.0*CHDRAG(I)
      DUM=SQRT((1.0+BETA2*(SWPWLE(I)**2))/(BETA2*(1.0+(SWPWLE(I)**2))))
      SUMSF=SUMSF+CHRDSF(I)*DUM
      SUMSUC=SUMSUC+2.0*CHDSUC(I)*DUM
      IF (I.EQ.IMAX) GO TO 663
      GO TO 664
  663 DO 665 JTIP=1,NCW
      JT=(IMAX-2)*NCW+JTIP
  665 TIPSF(JTIP)= TLSF(JT)
  664 CONTINUE
      JSEP=1
      IF (I.GT.NSEP) JSEP=2
      CHDTHR(I)= SQRT(CHDSUC(I)*CHDSUC(I)+CHRDSF(I)*CHRDSF(I))
      CHDTHR(I)=CHDTHR(I)*AKVLE(JSEP)
```

```
      YARM(I)= -(Y(I)+Y(I-1))/2.0
      XMOM(I)= CHDTHR(I)*YARM(I)
      SMCTHR=SMCTHR+CHDTHR(I)
      SMMOMX= SMMOMX+XMOM(I)
      IF(I.LT.IMAX) GO TO 702
      YBAR=0.0
      IF (SMCTHR.NE.0.0) YBAR= -SMMOMY/SMCTHR
      SUMTHR= SQRT((SUMSUC/2.0)*(SUMSUC/2.0)+SUMSF*SUMSF)
C     WING TIP SIDE FORCE,MOMENT ABOUT Z-AXIS AND CENTER OF PRESSURE
C     FORCES ON VORTEX TRAILING LEG ONLY ARE ACCOUNTED FOR AT WING TIP
C     TIPSFC= CY*C/2*b WHERE CY= (SIDEFORCE ACTING ON WINGTIP PANELS/
C     UNIT LENGTH ALONG WINGTIP)/Q*C
C
      COPTIP=0.0
      TOL= (1.0E-02)*CRW
      IF (CTW.LT.TOL) GO TO 666
      TIPEP=CTW/D
      ZTPMOM=0.0
      TIPFY=0.0
      DO 662 JTIP=1,NCW
      JT=(IMAX-2)*NCW+JTIP
      AJTIP= JTIP
      XLOC= (AJTIP-0.25)*TIPEP
      TIPFY=TIPFY+TIPSF(JTIP)
      TIPSFC(JTIP)= ((TIPSF(JTIP)/TIPEP)*SREF)/(4.0*SSPAN)
      ZTPMOM= ZTPMOM+  (TIPSF(JTIP)*PTLX(JT))/CAVE
  662 CONTINUE
      COPTIP=(-ZTPMOM/TIPFY)*(CAVE/CRW)
      GO TO 667
  666 ZTPMOM=0.0
      TIPFY=0.0
  667 CONTINUE
C
      CRBLT= SQRT(SUMLIF*SUMLIF+SUMDRG*SUMDRG)
      CNORM=SUMDRG*SINALP+SUMLIF*COSALP
      CA=SUMDRG*COSALP-SUMLIF*SINALP
      F=1.0
      IF (CNORM.LT.0.0) F=-1.0
      CNV=2.*SUMTHR*F
      CNVS=-2.0*TIPFY*F
      CNTOT= CNORM + CNV + CNVS
C
      IF (ABS(SINALP).LT.1.0E-07) GO TO 7401
      AKVLP= 2.0*SUMTHR/(SINALP*SINALP)
      AKVSP= (-2.0*TIPFY)/(SINALP*SINALP)
      GO TO 7402
 7401 AKVLP=0.0
      AKVSP=0.0
 7402 CONTINUE
C
      IF (ABS(SUMLIF).LT.1.0E-07) GO TO 311
      CDOCLS=SUMDRG/(SUMLIF*SUMLIF)
      GO TO 312
  311 CDOCLS=0.0
  312 CONTINUE
C
C     SLDCO=SPANLOAD COEFFICIENT CLC/CLCAVE FOR THE WING
C
      DO 3010 I=2,IMAX
      IF (ABS(SUMLIF).LT.1.0E-07) GO TO 3811
      SLDCO(I)=SLDCO1(I)/SUMLIF
      GO TO 3010
 3811 SLDCO(I)=0.0
 3010 CONTINUE
      CLTOT = SUMLIF
C
C     PITCHING MOMENT CALCULATION. MOMENTS TAKEN ABOUT POINT  XM,YM,ZM
C     AND DIVIDED BY  Q*SREF*CAVE
C
      WMOM=0.0
      ZMOM= 0.0
      ZLEMOM=0.0
      DO 9109 JT=1,M
      WMOM=WMOM+(2.0*((BLLIFT(JT)*SINALP+BLDRAG(JT)*COSALP)*(PVZ(JT)-ZM)
     1+(BLDRAG(JT)*SINALP-BLLIFT(JT)*COSALP)*(XM-PVX(JT))+(TLLIFT(JT)*
     2SINALP+TLDRAG(JT)*COSALP)*(PTLZ(JT)-ZM)+(TLDRAG(JT)*SINALP
     3-TLLIFT(JT)*COSALP)*(XM-PTLX(JT))))/CAVE)
      ZLEMOM=ZLEMOM+(-(BLLIFT(JT)*SINALP+BLDRAG(JT)*COSALP)*PVY(JT)+
     1BLSF(JT)*PVX(JT))/CAVE
 9109 ZMOM=ZMOM+(-(BLLIFT(JT)*SINALP+BLDRAG(JT)*COSALP)*PVY(JT)+BLSF(JT)
     1*PVX(JT)+TLSF(JT)*PTLX(JT))/CAVE
 8905 TOTMOM=WMOM
C
      CX=SUMSUC/2.
      IF (NPRINT.LT.0) GO TO 510
      WRITE (6,14)
      WRITE (6,15) CNORM,CNV,CNVS,CLTOT,WMOM,SUMDRG,CA,CDOCLS,CX,SUMSF,
     1             CRBLT
  510 CONTINUE
C
      IF (CTW.LT.TOL) GO TO 668
      CPTIP1= (ZMOM-ZLEMOM)/TIPFY
      GO TO 669
  668 CPTIP1=0.0
  669 CONTINUE
      XCOPP= (WMOM/CNORM)*CAVE+XM
      DO 670 I=2,IMAX
      IF (Y(I)-YBAR) 670,671,672
  670 CONTINUE
  671 XCOPLE=XLEL(I)
      GO TO 673
  672 DELTA=(Y(I)-YBAR)/(Y(I)-Y(I-1))
      XCOPLE=XLEL(I)-DELTA*(XLEL(I)-XLEL(I-1))
  673 CONTINUE
      XCOPSE= -COPTIP*CRW
      XCOP=(CNORM*XCOPP+2.0*SUMTHR*XCOPLE+2.0*TIPFY*XCOPSE)/CNTOT
      IF (NPRINT.GE.0) WRITE (6,33) XCOPP,XCOPLE,XCOPSE,XCOP
      IF(NPRINT.LT.100) RETURN
      WRITE (6,1)
      MN=0
      DO 8907 JT=1,M,NCW
      MN=MN+NCW
 8907 WRITE (6,16) (BLLIF1(J),BLLIF2(J),BLLIF3(J),BLLIF4(J),BLLIF5(J),
     1BLLIFT(J),J=JT,MN)
      WRITE (6,20)
      MN=0
      DO 8921 JT=1,M,NCW
      MN=MN+NCW
 8921 WRITE (6,17) (BLDRG2(J),BLDRG3(J),BLDRG4(J),BLDRG5(J),BLDRAG(J),
     1J=JT,MN)
      WRITE (6,24)
      MN=0
      DO 8925 JT=1,M,NCW
      MN=MN+NCW
 8925 WRITE (6,17) (BLSF1(J),BLSF2(J),BLSF3(J),BLSF4(J),BLSF(J),J=JT,MN)
      WRITE (6,2)
      MN=0
      DO 8908 JT=1,M,NCW
      MN=MN+NCW
 8908 WRITE (6,3) (TLLIFT(J),J=JT,MN)
      WRITE (6,21)
      MN=0
      DO 8922 JT=1,M,NCW
      MN=MN+NCW
 8922 WRITE (6,3) (TLDRAG(J),J=JT,MN)
      WRITE (6,25)
      MN=0
      DO 8926 JT=1,M,NCW
      MN=MN+NCW
 8926 WRITE (6,26) (TLSF1(J),TLSF2(J),TLSF(J),J=JT,MN)
      WRITE (6,4)
      MN=0
      DO 8909 JT=1,M,NCW
      MN=MN+NCW
 8909 WRITE (6,3) (V(J),J=JT,MN)
      WRITE (6,23)
      MN=0
      DO 8924 JT=1,M,NCW
      MN=MN+NCW
 8924 WRITE (6,3) (W(J),J=JT,MN)
      WRITE (6,5)
      MN=0
      DO 8910 JT=1,M,NCW
      MN=MN+NCW
```

```
8910 WRITE (6,3) (U(J),J=JT,MN)
     WRITE (6,6)
     MN=0
     DO 8911 JT=1,M,NCW
     MN=MN+NCW
8911 WRITE (6,3) (VB(J),J=JT,MN)
     WRITE (6,22)
     MN=0
     DO 8923 JT=1,M,NCW
     MN=MN+NCW
8923 WRITE (6,3) (WB(J),J=JT,MN)
C
     RETURN
     END
```

```
      SUBROUTINE INVERS(A,NSYS,N,NMAX,MMAX)
C     SUBROUTINE TO SOLVE SIMULTANEOUS EQUATIONS
C
      DIMENSION A(NMAX,MMAX),X(300)
  100 FORMAT (// 5X,18HMATRIX IS SINGULAR ,5X3HI =,I4)
    5 SIGN=1.0
      NPI=N+1
      NMI=N-1
      NPLSY=N+NSYS
    9 DO 14 I=1,NMI
      IPI=I+1
      MAX=I
      AMAX=ABS(A(I,I))
      DO 10 K=IPI,N
      AKMAX=ABS(A(K,I))
      IF(AKMAX.LE.AMAX) GO TO 10
      MAX=K
      AMAX=AKMAX
   10 CONTINUE
      IF(AMAX.LT.1.0E-12) GO TO 16
      IF(MAX.EQ.I) GO TO 12
      DO 11 L=I,NPLSY
      TEMP=A(I,L)
      A(I,L)=A(MAX,L)
   11 A(MAX,L)=TEMP
      SIGN=-SIGN
   12 DO 14 J=IPI,N
      IF (A(J,I)) 30,14,30
   30 CONST=-A(J,I)/A(I,I)
      DO 13 L=I,NPLSY
   13 A(J,L)=A(J,L)+A(I,L)*CONST
   14 CONTINUE
      DO 15 I=1,N
      ISING=I
      IF (A(I,I)) 15,16,15
   15 CONTINUE
      GO TO 18
   16 WRITE(6,100)  ISING
      STOP
   18 DO 21 I=NPI,NPLSY
      DO 20 KK=1,N
      K=NPI-KK
      X(K)=A(K,I)
      IF(K.EQ.N) GO TO 20
      J=K
   19 J=J+1
      X(K)=X(K)-A(K,J)*X(J)
      IF(J.NE.N) GO TO 19
   20 X(K)=X(K)/A(K,K)
      DO 21 J=1,N
   21 A(J,I)=X(J)
      RETURN
      END
```

```
      SUBROUTINE INFMM (PSII,APHII,XXX,YYY,ZZZ,SNN,FUI,FVI,FWI)
C
C
      COMMON/INF/OUT
C
      TOLRNC=(BOT*15.0E-05)**2
      DATA PSIID,APHIID/0., 0./ , FS,FC,FPS,FPC/0.,1.,0.,1./
      IF (PSII .EQ. PSIID) GO TO 10
      FS = SIN(PSII)
      FC = COS(PSII)
      FT = FS/FC
      PSIID=PSII
   10 CONTINUE
      IF (APHII.EQ.APHIID) GO TO 20
      FPS=SIN(APHII)
      FPC=COS(APHII)
      APHIID=APHII
   20 CONTINUE
      DUMY=SNN*FT
      F1=XXX+DUMY
      F4=XXX-DUMY
      DUMY=SNN*FPC
      F2=YYY+DUMY
      F5=YYY-DUMY
      DUMY=SNN*FPS
      F3=ZZZ+DUMY
      F6=ZZZ-DUMY
      DUMY2=YYY*FPS-ZZZ*FPC
      FFA=(XXX*FC-(YYY*FPC+ZZZ*FPS)*FS)**2+DUMY2*DUMY2
      FFE=F2*F2+F3*F3
      FFB=SQRT(F1*F1+FFE)
      FFD=F5*F5+F6*F6
      FFC=SQRT(F4*F4+FFD)
      DUMY3=FC*FPC
      DUMY4=FC*FPS
      FFF=(F1*FS+F2*DUMY3+F3*DUMY4)/FFB-(F4*FS+F5*DUMY3+F6*DUMY4)/FFC
C
C
      IF(ABS(FFA).LT.TOLRNC) GO TO 262
      DUMY=FFF/FFA
      FUONE=(ZZZ*FPC-YYY*FPS)*FC*DUMY
      FVONE=(XXX*DUMY4-ZZZ*FS)*DUMY
      FWONE=(YYY*FS-XXX*DUMY3)*DUMY
      GO TO 265
  262 FUONE=0.
      FVONE=0.
      FWONE=0.
  265 IF(ABS(FFD).LT.TOLRNC) GO TO 263
C
      DUMY=(1.-F4/FFC)/FFD
      FVTWO=F6*DUMY
      FWTWO=-F5*DUMY
      GO TO 266
  263 FVTWO=0.
      FWTWO=0.
  266 IF(ABS(FFE).LT.TOLRNC) GO TO 264
C
      DUMY=(1.-F1/FFB)/FFE
      FVTHRE=-F3*DUMY
      FWTHRE=F2*DUMY
C
      GO TO 267
  264 FVTHRE=0.
      FWTHRE=0.
  267 FUI=FUONE
      FVI=FVONE+FVTWO+FVTHRE
      FWI=FWONE+FWTWO+FWTHRE
      RETURN
      END
```

```
      SUBROUTINE ZVTX (ALPHAD,EPS,BTA2)
C
C       COMPUTE VERTICAL POSITION OF WING SEPARATION VORTEX
C
      DIMENSION A2E(8),B2E(8)
C
      DATA A2E/0.0,0.2,0.4,0.6,0.8,1.0,1.2,1.4/
      DATA B2E/0.0,0.11,0.22,0.3,0.375,0.45,0.52,0.575/
C
      F=1.0
      X=ALPHAD/(2.*EPS)
      IF (X .LT. 0.0) F=-1.0
      X=X*F
      DO 10 J=1,8
      K=J
      IF (X-A2E(J)) 12,14,10
   10 CONTINUE
      BTA2=ALPHAD/2.4
      RETURN
   14 BTA2=B2E(K)*EPS*F
      RETURN
   12 DELTA=(X-A2E(K-1))/(A2E(K)-A2E(K-1))
      BTA2=B2E(K-1) + DELTA*(B2E(K)-B2E(K-1))
      BTA2=BTA2*EPS*F
      RETURN
      END
```

```
      SUBROUTINE CNVTX (XINIT,XFINL,NSEC,XM,CN,CM)
C
C       COMPUTE THE FREE VORTEX INDUCED NORMAL FORCE AND MOMENT
C
      COMPLEX SIGMA,SIGMAB
      DIMENSION XBDY(100),RBDY(100),SBDY(100),DRBDY(100)
      DIMENSION XV(200),YV(4,200),ZV(4,200),NVP(200),ARAY(5,200)
      DIMENSION GAM(4),YG(4),ZG(4),SIGMA(4,2),SIGMAB(2),A(2)
C
      COMMON /BODY/ NTBL,MTBL,XBDY,RBDY,SBDY,DRBDY
      COMMON /VORTEX/ NV,NV2,GAM,YG,ZG
      COMMON /VTABLE/ NTRY,XV,YV,ZV,NVP,ARAY
C
      DUM=NSEC
      DX=(XFINL-XINIT)/DUM
      XI=XINIT
      K=1
      CN=0.0
      CM=0.0
      MN=1
      IN=1
      DO 30 J=1,NSEC
      DO 10 NK=K,2
      NN=MN
      DO 11 M=NN,NTRY
      MN=M
      IF (XI-XV(M)) 13,12,11
   11 CONTINUE
   12 DO 112 NVN=1,NV
      Y=YV(NVN,MN)
      Z=ZV(NVN,MN)
  112 SIGMA(NVN,NK)=CMPLX(Y,Z)
      GO TO 17
   13 IF (MN.EQ.1) GO TO 12
      DELTA=(XI-XV(MN-1))/(XV(MN)-XV(MN-1))
      DO 113 NVN=1,NV
      Y=YV(NVN,MN-1) + DELTA*(YV(NVN,MN)-YV(NVN,MN-1))
      Z=ZV(NVN,MN-1) + DELTA*(ZV(NVN,MN)-ZV(NVN,MN-1))
  113 SIGMA(NVN,NK)=CMPLX(Y,Z)
   17 IF (MN.GT.1) MN=MN-1
C
C       LOOK UP LOCAL BODY RADIUS
C
      JN=IN
      DO 14 I=JN,NTBL
      IN=I
      IF (XI-XBDY(I)) 16,15,14
   14 CONTINUE
   15 A(NK)=RBDY(IN)
      GO TO 18
   16 IF (IN.EQ.1) GO TO 15
      DELTA=(XI-XBDY(IN-1))/(XBDY(IN)-XBDY(IN-1))
      A(NK)=RBDY(IN-1)+DELTA*(RBDY(IN)-RBDY(IN-1))
   18 IF (IN.GT.1) IN=IN-1
   10 XI=XI+DX
      K=2
      XP=XI-DX-DX/2.
C
C       COMPUTE NORMAL FORCE AND MOMENT ON ONE SECTOR OF BODY
C
      CND=0.0
      DO 20 NVN=1,NV
      DO 21 NK=1,2
   21 SIGMAB(NK)=SIGMA(NVN,NK)-A(NK)*A(NK)/CONJG(SIGMA(NVN,NK))
      CND=CND+4.0*GAM(NVN)*REAL(SIGMAB(2)-SIGMAB(1))
   20 CONTINUE
      CM=CM+CND*(XM-XP)
      CN=CN+CND
      A(1)=A(2)
      DO 22 NVN=1,NV
   22 SIGMA(NVN,1)=SIGMA(NVN,2)
   30 CONTINUE
      RETURN
      END
```

COMPONENTS	TYPE	NORMAL-FORCE COEFFICIENT	LIFT COEFFICIENT	PITCHING-MOMENT COEFFICIENT	CENTER OF PRESSURE LOCATION	AXIAL-FORCE COEFFICIENT
NOSE	Potential	$C_{N_{N,p}}$	$C_{L_{N,p}}$	$C_{m_{N,p}}$	$\bar{x}_{N,p}$	---------
	Viscous	$C_{N_{N,v}}$	$C_{L_{N,v}}$	$C_{m_{N,v}}$	$\bar{x}_{N,v}$	---------
WING IN PRESENCE OF BODY	Potential	$C_{N_{W(B),p}}$	$C_{L_{W(B),p}}$	$C_{m_{W(B),p}}$	$\bar{x}_{W(B),p}$	$C_{A_{W(B),p}}$
	Viscous	$C_{N_{W(B),v}}$	$C_{L_{W(B),v}}$	$C_{m_{W(B),v}}$	$\bar{x}_{W(B),v}$	$C_{A_{W(B),v}}$
BODY IN PRESENCE OF WING	Potential	$C_{N_{B(W),p}}$	$C_{L_{B(W),p}}$	$C_{m_{B(W),p}}$	$\bar{x}_{B(W),p}$	---------
	Viscous	$C_{N_{B(W),v}}$	$C_{L_{B(W),v}}$	$C_{m_{B(W),v}}$	$\bar{x}_{B(W),v}$	---------
AFTERBODY	--------	C_{N_A}	C_{L_A}	C_{m_A}	$\bar{x}_A$	---------
TAIL IN PRESENCE OF BODY	Potential	$C_{N_{T(B),p}}$	$C_{L_{T(B),p}}$	$C_{m_{T(B),p}}$	$\bar{x}_{T(B),p}$	$C_{A_{T(B),p}}$
	Viscous	$C_{N_{T(B),v}}$	$C_{L_{T(B),v}}$	$C_{m_{T(B),v}}$	$\bar{x}_{T(B),v}$	$C_{A_{T(B),v}}$
BODY IN PRESENCE OF TAIL	Potential	$C_{N_{B(T),p}}$	$C_{L_{B(T),p}}$	$C_{m_{B(T),p}}$	$\bar{x}_{B(T),p}$	---------
	Viscous	$C_{N_{B(T),v}}$	$C_{L_{B(T),v}}$	$C_{m_{B(T),v}}$	$\bar{x}_{B(T),v}$	---------
COMPLETE CONFIGURATION	--------	C_N	C_L	C_m	$\bar{x}$	C_A

Table I.- Summary of force and moment coefficient notation.

ITEM 1 FORMAT (15I5,I3,I2)

1	6	11	16	21	26	31	36	41	46	51	56	61	66	71	76	79
NHEAD	NTBL	NPRINT	MPRINT	NOSEV	NTRIM	NCWW	MSWW	NCAMW	NPSIW	NSEPW	NCWT	MSWT	NCAMT	NPSIT	NSEPT	NBODY

ITEM 2 FORMAT (20A4); NHEAD cards

1
TITLE

ITEM 3 FORMAT (2F10.5); NTBL cards

1	11
XBDY(J)	RBDY(J)

ITEM 4 FORMAT (8F10.5)

1	11	21	31	41	51	61	71
XM	ZM	EMACH	REFS	REFL	THETAN	DXOUT	DXI

ITEM 5 FORMAT (8F10.5)

1	11	21	31	41	51	61	71
XLEW	XTEW	XHLW	ZHLW	XCPW	BS2W	RAVGW	YSEPW

ITEM 6 FORMAT (4F10.5)

1	11	21	31	41	51	61	71
PHIW	PSILEW(1)	PSITEW(1)	CMTEST				

ITEM 7 FORMAT (8F10.5)

1	11	21	31	41	51	61	71
YW(J)							

If NPSIW = 0, go to Item 10.

ITEM 8 FORMAT (8F10.5)

1	11	21	31	41	51	61	71
PSILEW(J)							

ITEM 9 FORMAT (8F10.5)

1	11	21	31	41	51	61	71
PSITEW(J)							

(a) Page 1.

Figure 1.- Input format for SUBSON program.

ITEM 10 If NCAMW = 0, go to Item 11.
FORMAT (8F10.5); MSWW cards.

1	11	21	31	41	51	61	71
ALPHLW(J)							

ITEM 11 If MSWT = 0, go to Item 17.
FORMAT (8F10.5)

1	11	21	31	41	51	61	71
XLET	XTET	XHLT	ZHLT	XCPT	BS2T	RAVGT	YSEPT

ITEM 12 FORMAT (8F10.5)

1	11	21	31	41	51	61	71
PHIT	PSILET(1)	PSITET(1)					

ITEM 13 FORMAT (8F10.5)

1	11	21	31	41	51	61	71
YT(J)							

ITEM 14 If NPSIT = 0, go to Item 16.
FORMAT (8F10.5)

1	11	21	31	41	51	61	71
PSILET(J)							

ITEM 15 FORMAT (8F10.5)

1	11	21	31	41	51	61	71
PSITET(J)							

ITEM 16 If NCAMT = 0, go to Item 17.
FORMAT (8F10.5); MSWT cards

1	11	21	31	41	51	61	71
ALPHLT(J)							

ITEM 17 FORMAT (I2,F6.2,9F8.3)

1	3	9	17	25	33	41	49	57	65	73
NDEX	ALPHAD	ALPIW	AKVLW1	AKVLW2	AKVSW	ALPIT	AKVLT1	AKVLT2	AKVST	WLIMIT

(b) Page 2.

Figure 1.- Concluded.

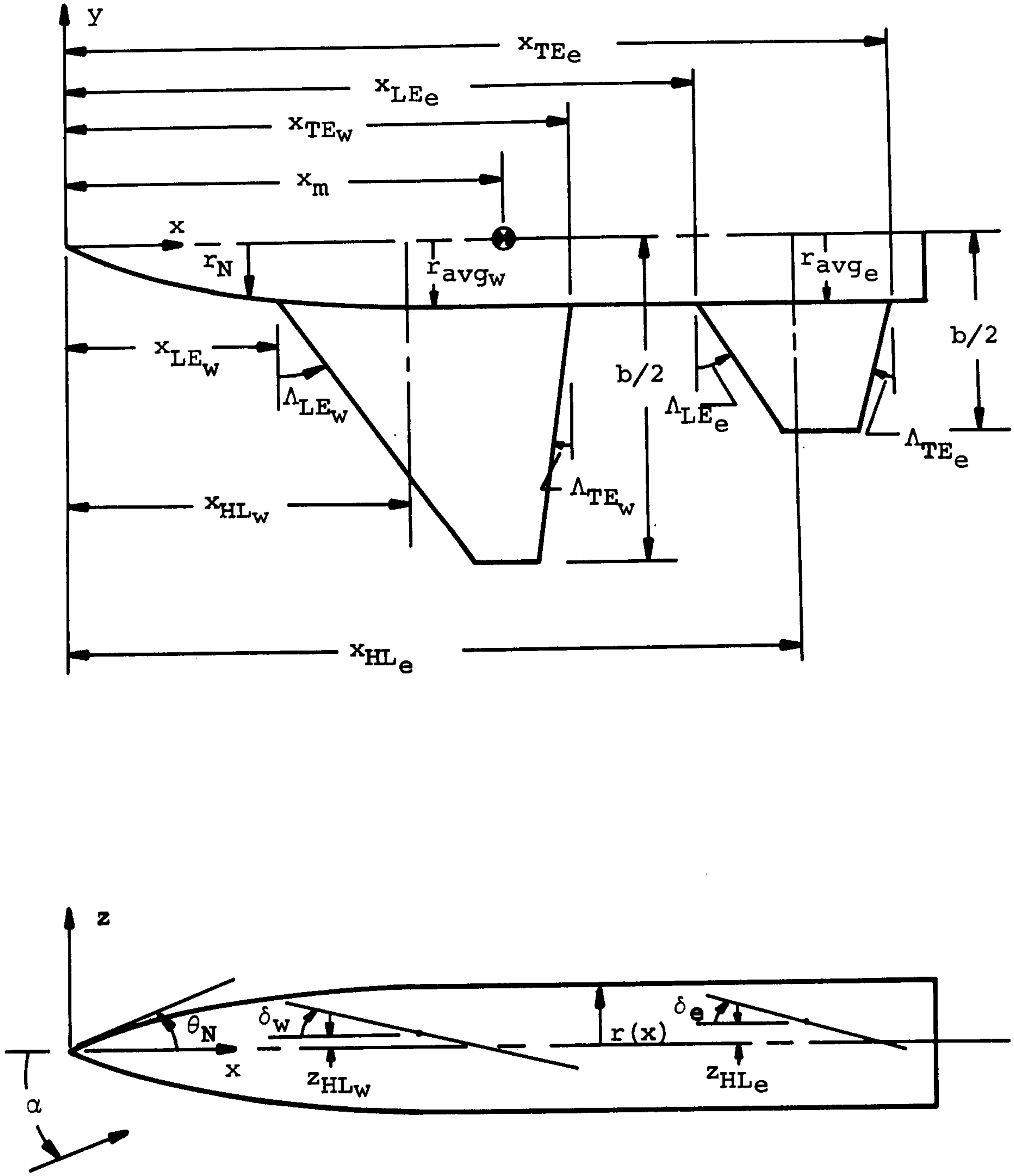


Figure 2.- Geometric nomenclature for SUBSON program.

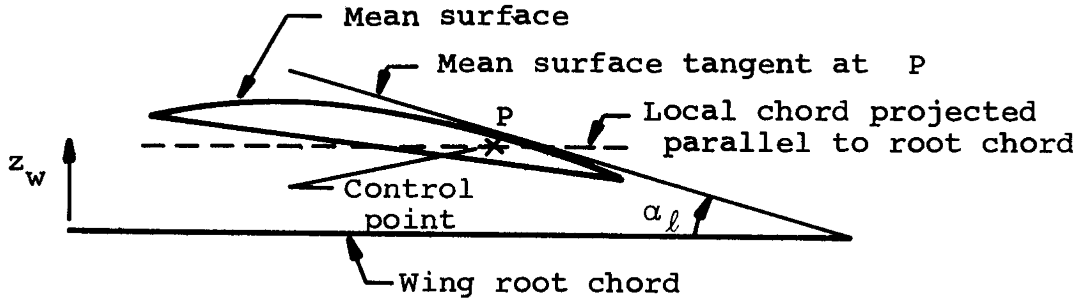


Figure 3.- Mean surface detail for a wing with camber, twist, and dihedral.

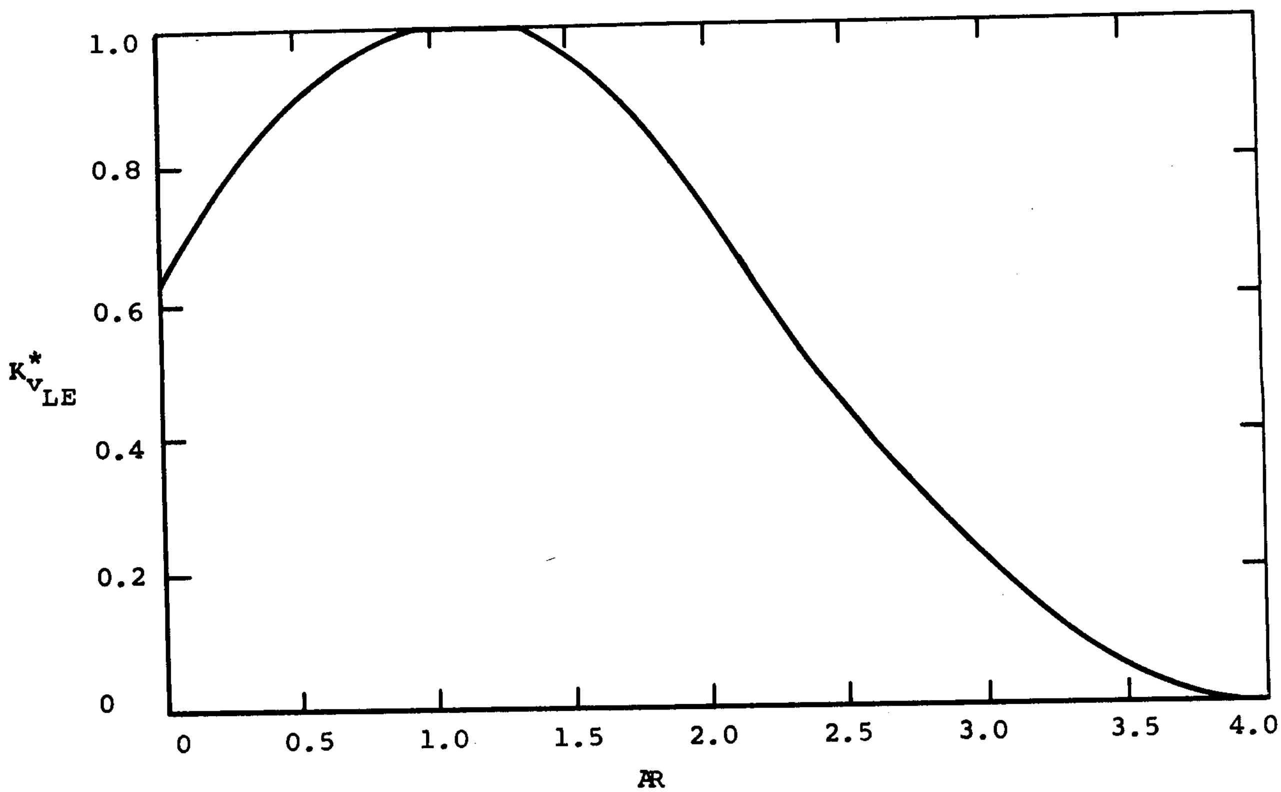


Figure 4.- Correlation curve for vortex-lift ratio on delta wings in incompressible flow.

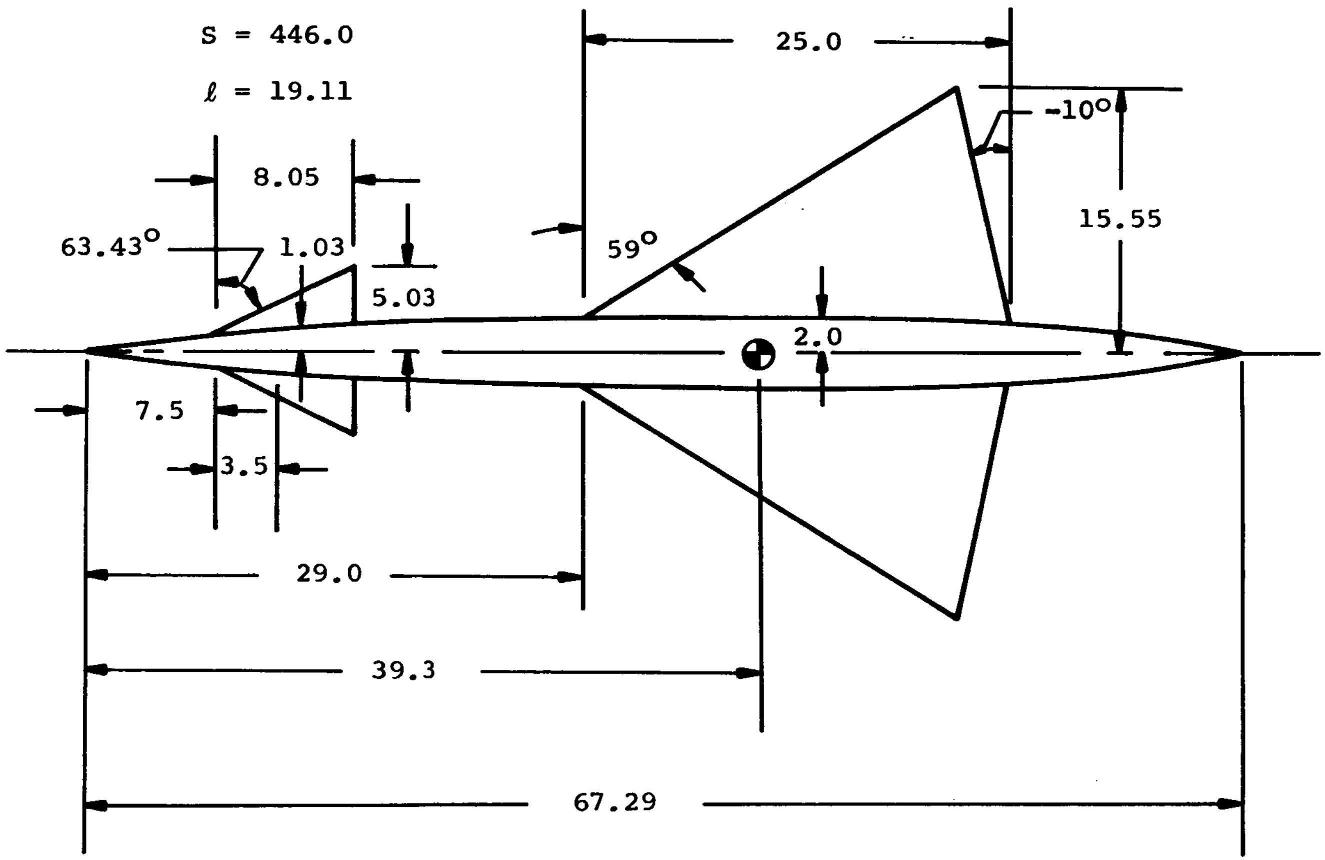


Figure 5.- Canard-body-wing configuration for sample cases 1 and 2.

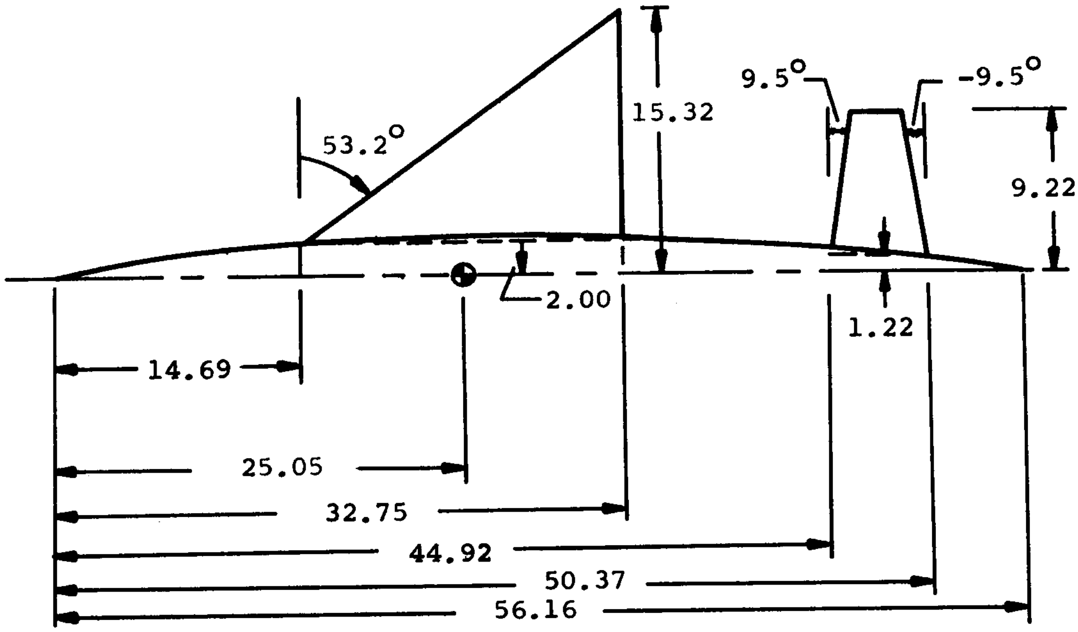


Figure 6.- Wing-body-tail configuration for sample case 3.

```
  2   35     0     0     1     0     4    10     0     0     1     8    10     0     0  1  0
 SAMPLE CASE 1                                CANARD - BODY - WING CONFIGURATION
   REF. NASA TM X-643        BRADY, PAGE, AND KOENIG        JANUARY 1962
  0.         0.02
 2.          .30
 4.          .53
 6.          .72
 7.5         .84
 8.          .88
 10.         1.03
 12.         1.18
 14.         1.33
 16.         1.46
 18.         1.58
 20.         1.68
 22.         1.76
  24.        1.83
 26.         1.88
 28.         1.92
 29.         1.935
 30.         1.95
 32.         1.98
 34.         2.
 36.         2.02
  38.        2.03
 44.         2.03
 46.         2.01
 48.         1.98
 50.         1.93
 52.         1.87
 54.         1.77
 56.         1.63
 58.         1.48
 60.         1.27
 62.         1.03
 64.         0.77
 66.         0.43
 67.         0.02
 39.3        0.          0.        446.       19.11      8.        1.0       0.5
 7.5         15.55       11.       0.         0.         5.03      1.03      5.03
 0.          63.43       0.        0.
 1.43        1.83        2.23      2.63       3.03       3.43      3.83      4.23
 4.63        5.03
 29.         54.         39.3      0.         0.         15.55     2.        15.55
 0.          59.0        -10.        0.0
 3.355       4.71        6.065     7.42       8.775      10.12     11.485    12.84
 14.195      15.55
1  4.     0.     .53     0.     0.     0.     .41     0.     0.     1.0
1  8.     0.     .53     0.     0.     0.     .41     0.     0.     1.0
1 12.     0.     .53     0.     0.     0.     .41     0.     0.     1.0
1 16.     0.     .53     0.     0.     0.     .41     0.     0.     1.0
1  0.    10.     .53     0.     0.     0.     .41     0.     0.     1.0
1  4.    10.     .53     0.     0.     0.     .41     0.     0.     1.0
1  8.    10.     .53     0.     0.     0.     .41     0.     0.     1.0
1 12.    10.     .53     0.     0.     0.     .41     0.     0.     1.0
1 16.    10.     .53     0.     0.     0.     .41     0.     0.     1.0
```

(a) Sample case 1.

Figure 7.- Sample input decks for SUBSON program.

```
  2   35     0     0     1     0     8    10     0     1     1     0     0     0     0  0 0
 SAMPLE CASE 2                                WING - BODY CONFIGURATION
   REF. NASA TM X-643        BRADY, PAGE, AND KOENIG        JANUARY 1962
 0.          0.02
2.           .30
4.           .53
6.           .72
7.5          .84
8.           .88
10.          1.03
12.          1.18
14.          1.33
16.          1.46
18.          1.58
20.          1.68
22.          1.76
 24.         1.83
26.          1.88
28.          1.92
29.          1.935
30.          1.95
32.          1.98
34.          2.
36.          2.02
 38.         2.03
44.          2.03
46.          2.01
48.          1.98
50.          1.93
52.          1.87
54.          1.77
56.          1.63
58.          1.48
60.          1.27
62.          1.03
64.          0.77
66.          0.43
67.          0.02
39.3         0.          0.          446.        19.11       8.          1.0         0.5
29.          54.         39.3        0.          0.          15.55       2.          15.55
0.           59.0        -10.         0.0
3.355        4.71        6.065       7.42        8.775       10.12       11.485      12.84
14.195       15.55
59.0          59.0        59.0        59.0        59.0        59.0        59.0        59.0
59.0          59.0
-10.0         -10.0       -10.0       -10.0       -10.0       -10.0       -10.0       -10.0
-10.0         -10.0
1  4.0   0.0     0.41     0.0     0.0     0.0     0.0     0.0     0.0     1.0
1  8.0   0.0     0.41     0.0     0.0     0.0     0.0     0.0     0.0     1.0
1 12.0   0.0     0.41     0.0     0.0     0.0     0.0     0.0     0.0     1.0
1 16.0   0.0     0.41     0.0     0.0     0.0     0.0     0.0     0.0     1.0
1 20.0   0.0     0.41     0.0     0.0     0.0     0.0     0.0     0.0     1.0
```

(b) Sample case 2.

Figure 7.- Continued.

```
   3   23   -1    0    1    2    6   10    0    0    1    4    8    0    0  1 0
    SAMPLE CASE 3                        WING - BODY - TAIL  CONFIGURATION
         REF, NACA RM A52L15   D, KOENIG      APRIL 1953
                         TRIM CALCULATION
  0,         ,01
  ,35        ,146
  1,4        ,393
  5,6        1,04
  8,4        1,35
  11,23      1,61
  14,04       1,81
  14,7       1,85
  16,85      1,97
  19,66      2,09
  22,46      2,18
  25,27      2,23
  28,08      2,25
  30,89      2,23
  33,67      2,18
  36,5       2,09
  39,3       1,97
  42,12      1,81
   44,92     1,61
  47,74      1,35
  50,5       1,04
  54,6       ,39
  56,2       ,01
  25,05       0,        0,13      313,76    13,65     22,       1,0       0,5
  14,7       32,5       25,05     0,        0,        15,32      2,       15,32
  0,         53,2       0,        0,01
  3,332      4,664      5,996     7,328     8,66      9,992     11,324    12,656
  13,988      15,32
  44,92      50,37      46,21     0,        0,0       9,22      1,22      9,22
  0,         9,5        -9,5
  2,22       3,22       4,22      5,22      6,22      7,22      8,22      9,22
 1  16,    0,       ,20      0,      1,0      -6,      1,      0,      1,     0,1
```

(c) Sample case 3.

Figure 7.- Concluded.

```
&CARD1
NHEAD= 2
NTBL= 35
NPRINT= 0
MPRINT= 0
NOSEV= 1
NTRIM= 0
NCWW= 4
MSWW= 10
NCAMW= 0
NPSIW= 0
NSEPW= 1
NCWT= 8
MSWT= 10
NCAMT= 0
NPSIT= 0
NSEPT= 1
NBODY= 0
&END

                                          CANARD - BODY - WING CONFIGURATION
      SAMPLE CASE 1
        REF. NASA TM X-643        BRADY, PAGE, AND KOENIG        JANUARY 1962
```

(a) Page 1.

Figure 8.- Sample case from SUBSON program.

FLOW CONDITIONS -

ALPHA	MACH	BETA
16.00	0.0000	1.0000

CONFIGURATION GEOMETRY -

REFERENCE AREA = 446.000
REFERENCE LENGTH = 19.110

WING ...

XLE	XTE	B/2	XHL	ZHL	C(ROOT)	C(TIP)	Y(SEP)
7.5000	15.5500	5.0300	11.0000	0.0000	8.0500	0.0000	5.0300

		SWEEP ANGLE	
DIHEDRAL	Y(RT)	LE	TE
0.000	1.030	63.430	0.000
	Y(WING)	LE	TE
	1.430	63.430	0.000
	1.830	63.430	0.000
	2.230	63.430	0.000
	2.630	63.430	0.000
	3.030	63.430	0.000
	3.430	63.430	0.000
	3.830	63.430	0.000
	4.230	63.430	0.000
	4.630	63.430	0.000
	5.030	63.430	0.000

TAIL ...

XLE	XTE	B/2	XHL	ZHL	C(ROOT)	C(TIP)	Y(SEP)
29.0000	54.0000	15.5500	39.3000	0.0000	25.0000	0.0000	15.5500

		SWEEP ANGLE	
DIHEDRAL	Y(RT)	LE	TE
0.000	2.000	59.000	-10.000
	Y(TAIL)	LE	TE
	3.355	59.000	-10.000
	4.710	59.000	-10.000
	6.065	59.000	-10.000
	7.420	59.000	-10.000
	8.775	59.000	-10.000
	10.120	59.000	-10.000
	11.485	59.000	-10.000
	12.840	59.000	-10.000
	14.195	59.000	-10.000
	15.550	59.000	-10.000

BODY ...

---------- NOSE ----------			AVERAGE RADIUS		CENTER OF MOMENTS			
THETA	FINENESS	R(BASE)	WING	TAIL	X	Z	DXI	DXOUT
8.000	4.464	0.840	1.030	2.000	39.300	0.000	0.500	1.000

TABLE OF BODY COORDINATES

	X	R	S	DR/DX
1	0.00000	0.02000	0.02000	0.14000
2	2.00000	0.30000	0.30000	0.12750
3	4.00000	0.53000	0.53000	0.10500
4	6.00000	0.72000	0.72000	0.08750
5	7.50000	0.84000	0.84000	0.07906
6	8.29982	0.90249	1.43000	0.07656
7	9.09965	0.96247	1.83000	0.07500
8	9.89947	1.02746	2.23000	0.07500
9	10.69930	1.08245	2.63000	0.07500
10	11.49912	1.14243	3.03000	0.07500
11	12.29895	1.20242	3.43000	0.07500
12	13.09877	1.26241	3.83000	0.07500
13	13.89860	1.32239	4.23000	0.07063
14	14.69842	1.37540	4.63000	0.06563
15	15.49825	1.42738	5.03000	0.06500
16	15.55100	1.43081	1.43081	0.06500
17	16.00000	1.46000	1.46000	0.06250
18	18.00000	1.58000	1.58000	0.05500
19	20.00000	1.68000	1.68000	0.04500
20	22.00000	1.76000	1.76000	0.03750
21	24.00000	1.83000	1.83000	0.03000
22	26.00000	1.88000	1.88000	0.02250
23	28.00000	1.92000	1.92000	0.01750
24	29.00000	1.93500	1.93500	0.01500
25	31.25508	1.96883	3.35500	0.01333
26	33.51018	1.99510	4.71000	0.01083
27	35.76527	2.01765	6.06500	0.00774
28	38.02036	2.03000	7.42000	0.00274
29	40.27545	2.03000	8.77500	0.00000
30	42.51390	2.03000	10.12000	-0.00173
31	44.78564	2.02214	11.48500	-0.00788
32	47.04073	1.99439	12.84000	-0.01653
33	49.29581	1.94760	14.19500	-0.02459
34	51.55089	1.88347	15.55000	-0.03739
35	54.00099	1.76993	1.76993	-0.05817
36	56.00000	1.63000	1.63000	-0.07250
37	58.00000	1.49000	1.48000	-0.09000
38	60.00000	1.27000	1.27000	-0.11250
39	62.00000	1.03000	1.03000	-0.12500
40	64.00000	0.77000	0.77000	-0.15000
41	66.00000	0.43000	0.43000	-0.29000
42	67.00000	0.02000	0.02000	-0.41000

(b) Page 2.

Figure 8.- Continued.

		-- INCIDENCE --		---- WING -----		---- TAIL -----		
ALPHA	M	WING	TAIL	KVLF*	KVSE*	KVLF*	KVSE*	W/V LIMIT
16.00	0.00	10.000	0.000	0.530	0.000	0.410	0.000	1.000

NOSE VORTICES -	GAM/2*PI*V*RB	Y/RB	Z/RB	XS/RB
	0.1013	0.484	1.435	0.000

VELOCITY INDUCED AT SPECIFIED FIELD POINTS ON WING (BESKIN)

	X	Y	Z	V/V(INF)	W/V(INF)
1	9.334	1.230	0.289	7.3907E-02	1.4845E-01
2	11.247	1.230	-0.043	-1.5989E-02	2.2914E-01
3	13.159	1.230	-0.375	-1.4927E-01	2.2207E-01
4	15.072	1.230	-0.707	-2.3183E-01	1.3501E-01
5	9.984	1.630	0.176	2.3220E-02	1.0602E-01
6	11.697	1.630	-0.121	-2.0400E-02	1.3666E-01
7	13.409	1.630	-0.418	-7.7487E-02	1.4100E-01
8	15.122	1.630	-0.716	-1.2603E-01	1.1584E-01
9	10.634	2.030	0.064	4.8533E-03	7.7436E-02
10	12.147	2.030	-0.199	-1.8259E-02	9.2183E-02
11	13.659	2.030	-0.462	-4.6810E-02	9.7566E-02
12	15.172	2.030	-0.724	-7.4273E-02	9.0812E-02
13	11.284	2.430	-0.049	-2.4004E-03	5.9141E-02
14	12.597	2.430	-0.277	-1.5569E-02	6.7343E-02
15	13.909	2.430	-0.505	-3.1219E-02	7.1841E-02
16	15.222	2.430	-0.733	-4.7004E-02	7.0810E-02
17	11.934	2.830	-0.162	-5.4094E-03	4.7053E-02
18	13.046	2.830	-0.355	-1.3267E-02	5.1996E-02
19	14.159	2.830	-0.549	-2.2243E-02	5.5218E-02
20	15.272	2.830	-0.742	-3.1525E-02	5.6004E-02
21	12.584	3.230	-0.275	-6.6405E-03	3.8717E-02
22	13.496	3.230	-0.433	-1.1426E-02	4.1803E-02
23	14.409	3.230	-0.592	-1.6674E-02	4.3960E-02
24	15.322	3.230	-0.750	-2.2157E-02	4.5106E-02
25	13.233	3.630	-0.388	-7.0755E-03	3.2734E-02
26	13.946	3.630	-0.512	-9.9608E-03	3.4635E-02
27	14.659	3.630	-0.635	-1.2992E-02	3.5975E-02
28	15.372	3.630	-0.759	-1.6176E-02	3.6982E-02
29	13.883	4.030	-0.501	-7.1396E-03	2.8290E-02
30	14.396	4.030	-0.590	-8.7461E-03	2.9244E-02
31	14.909	4.030	-0.679	-1.0431E-02	3.0087E-02
32	15.422	4.030	-0.768	-1.2184E-02	3.0814E-02
33	14.533	4.430	-0.614	-6.9727E-03	2.4690E-02
34	14.846	4.430	-0.668	-7.7663E-03	2.5172E-02
35	15.159	4.430	-0.722	-8.5818E-03	2.5622E-02
36	15.472	4.430	-0.777	-9.4201E-03	2.6045E-02
37	15.183	4.830	-0.726	-6.7264E-03	2.1858E-02
38	15.296	4.830	-0.746	-6.9634E-03	2.2005E-02
39	15.409	4.830	-0.766	-7.2029E-03	2.2150E-02
40	15.522	4.830	-0.785	-7.4449E-03	2.2293E-02

SUMMARY OF VORTEX STRENGTHS AND POSITION AT X = 7.500

VORTEX	GAMMA/2*PI*V	Y	Z
1	0.085127	0.406	1.206

(c) Page 3.

Figure 8.- Continued.

VELOCITY INDUCED AT SPECIFIED FIELD POINTS ON WING

	X	Y	Z	V/V(INF)	W/V(INF)
1	9.334	1.230	0.289	1.3685E-02	-1.4366E-02
2	11.247	1.230	-0.043	1.3146E-03	-9.5434E-03
3	13.159	1.230	-0.375	-1.2206E-03	-4.4134E-03
4	15.072	1.230	-0.707	-1.1452E-03	-2.1045E-03
5	9.984	1.630	0.176	8.6317E-03	-5.4274E-03
6	11.697	1.630	-0.121	3.2471E-03	-5.1893E-03
7	13.409	1.630	-0.418	7.9904E-04	-3.5265E-03
8	15.122	1.630	-0.716	-9.2118E-06	-2.2802E-03
9	10.634	2.030	0.064	5.6603E-03	-2.2443E-03
10	12.147	2.030	-0.199	3.1833E-03	-2.7330E-03
11	13.659	2.030	-0.462	1.5972E-03	-2.3859E-03
12	15.172	2.030	-0.724	7.7890E-04	-1.9164E-03
13	11.284	2.430	-0.049	3.8802E-03	-9.4032E-04
14	12.597	2.430	-0.277	2.6784E-03	-1.4122E-03
15	13.909	2.430	-0.505	1.7609E-03	-1.4827E-03
16	15.222	2.430	-0.733	1.1751E-03	-1.3987E-03
17	11.934	2.830	-0.162	2.7675E-03	-3.6009E-04
18	13.046	2.830	-0.355	2.1667E-03	-7.0150E-04
19	14.159	2.830	-0.549	1.6644E-03	-8.6118E-04
20	15.272	2.830	-0.742	1.2976E-03	-9.2238E-04
21	12.584	3.230	-0.275	2.0382E-03	-8.9085E-05
22	13.496	3.230	-0.433	1.7328E-03	-3.1138E-04
23	14.409	3.230	-0.592	1.4742E-03	-4.5484E-04
24	15.322	3.230	-0.750	1.2671E-03	-5.4875E-04
25	13.233	3.630	-0.388	1.5491E-03	4.0093E-05
26	13.946	3.630	-0.512	1.3995E-03	-9.5439E-05
27	14.659	3.630	-0.635	1.2743E-03	-1.9726E-04
28	15.372	3.630	-0.759	1.1650E-03	-2.7801E-04
29	13.883	4.030	-0.501	1.2079E-03	1.0062E-04
30	14.396	4.030	-0.590	1.1455E-03	2.7879E-05
31	14.909	4.030	-0.679	1.0917E-03	-3.5315E-05
32	15.422	4.030	-0.768	1.0378E-03	-9.0684E-05
33	14.533	4.430	-0.614	9.7275E-04	1.3155E-04
34	14.846	4.430	-0.668	9.5290E-04	9.7355E-05
35	15.159	4.430	-0.722	9.3188E-04	6.5137E-05
36	15.472	4.430	-0.777	9.0940E-04	3.4764E-05
37	15.183	4.830	-0.726	8.0665E-04	1.4557E-04
38	15.296	4.830	-0.746	8.0142E-04	1.3565E-04
39	15.409	4.830	-0.766	7.9604E-04	1.2591E-04
40	15.522	4.830	-0.785	7.9090E-04	1.1633E-04

TOTAL INDUCED VELOCITY AT SPECIFIED FIELD POINTS ON WING

	X	Y	Z	V/V(INF)	W/V(INF)
1	9.334	1.230	0.289	8.7592E-02	1.3409E-01
2	11.247	1.230	-0.043	-1.4674E-02	2.1960E-01
3	13.159	1.230	-0.375	-1.5049E-01	2.1766E-01
4	15.072	1.230	-0.707	-2.3297E-01	1.3290E-01
5	9.984	1.630	0.176	3.1852E-02	1.0060E-01
6	11.697	1.630	-0.121	-1.7152E-02	1.3147E-01
7	13.409	1.630	-0.418	-7.6687E-02	1.3748E-01
8	15.122	1.630	-0.716	-1.2604E-01	1.1356E-01
9	10.634	2.030	0.064	1.0514E-02	7.5192E-02
10	12.147	2.030	-0.199	-1.5076E-02	8.9450E-02
11	13.659	2.030	-0.462	-4.5213E-02	9.5180E-02
12	15.172	2.030	-0.724	-7.3495E-02	8.8895E-02
13	11.284	2.430	-0.049	1.4798E-03	5.8201E-02
14	12.597	2.430	-0.277	-1.2890E-02	6.5930E-02
15	13.909	2.430	-0.505	-2.9458E-02	7.0358E-02
16	15.222	2.430	-0.733	-4.5829E-02	6.9411E-02
17	11.934	2.830	-0.162	-2.6420E-03	4.6693E-02
18	13.046	2.830	-0.355	-1.1100E-02	5.1294E-02
19	14.159	2.830	-0.549	-2.0579E-02	5.4357E-02
20	15.272	2.830	-0.742	-3.0227E-02	5.5081E-02
21	12.584	3.230	-0.275	-4.6022E-03	3.8628E-02
22	13.496	3.230	-0.433	-9.6934E-03	4.1492E-02
23	14.409	3.230	-0.592	-1.5199E-02	4.3505E-02
24	15.322	3.230	-0.750	-2.0889E-02	4.4558E-02
25	13.233	3.630	-0.388	-5.5263E-03	3.2774E-02
26	13.946	3.630	-0.512	-8.5613E-03	3.4540E-02
27	14.659	3.630	-0.635	-1.1718E-02	3.5778E-02
28	15.372	3.630	-0.759	-1.5011E-02	3.6704E-02
29	13.883	4.030	-0.501	-5.9317E-03	2.8391E-02
30	14.396	4.030	-0.590	-7.6007E-03	2.9272E-02
31	14.909	4.030	-0.679	-9.3397E-03	3.0052E-02
32	15.422	4.030	-0.768	-1.1146E-02	3.0723E-02
33	14.533	4.430	-0.614	-5.9999E-03	2.4822E-02
34	14.846	4.430	-0.668	-6.8134E-03	2.5269E-02
35	15.159	4.430	-0.722	-7.6499E-03	2.5687E-02
36	15.472	4.430	-0.777	-8.5107E-03	2.6080E-02
37	15.183	4.830	-0.726	-5.9198E-03	2.2003E-02
38	15.296	4.830	-0.746	-6.1620E-03	2.2141E-02
39	15.409	4.830	-0.766	-6.4069E-03	2.2276E-02
40	15.522	4.830	-0.785	-6.6540E-03	2.2409E-02

(c) Concluded.

Figure 8.- Continued.

WING

LIFTING SURFACE CALCULATION

FLOW CONDITIONS - ALPHA 26.000, BETA 1.0000

LATTICE ARRANGEMENT - 4 CHORDWISE AND 10 SPANWISE ELEMENTS

REFERENCE QUANTITIES - PLANFORM AREA S = 32.386, AVERAGE CHORD CAVG = 4.05

C(ROOT)	C(TIP)	B/2	DIHEDRAL	XCM	ZCM	Y(SEP)	ETA(SEP)
8.050	0.000	4.000	0.000	0.000	0.000	4.000	1.000

STATION	Y	SWEEP LE	SWEEP TE
1	0.400	63.430	0.000
2	0.800	63.430	0.000
3	1.200	63.430	0.000
4	1.600	63.430	0.000
5	2.000	63.430	0.000
6	2.400	63.430	0.000
7	2.800	63.430	0.000
8	3.200	63.430	0.000
9	3.600	63.430	0.000
10	4.000	63.430	0.000

CNP	CNV	CNVS	CLP	CMP	CDI	CA	CDI/CL2	CXV	CYV	CRP
1.00133	0.79972	0.00000	1.07112	-1.11890	0.08809	-0.39037	0.07678	0.19519	-0.34899	1.07474

X COORDINATES OF CENTERS OF PRESSURE DUE TO

POTENTIAL LIFT	LEADING EDGE SUCTION	SIDE EDGE SUCTION	OVERALL
-4.5736	-4.3412	-0.0000	-4.4426

SPANWISE LOAD DISTRIBUTION

STATION	Y/(B/2)	LOCAL CHORD,C	CL*C/CL*CAVG	CL*C/(2*B)	CX*C/(2*B)	CY*C/(2*B)	CSUC*C/(2*B)	ETA,DEG.
1	0.0500	7.6501	1.1919	0.3230	0.0225	-0.0443	0.0497	90.3978
2	0.1500	6.8503	1.2551	0.3402	0.0832	-0.1434	0.1658	93.5590
3	0.2500	6.0504	1.2371	0.3353	0.0973	-0.1709	0.1966	93.0956
4	0.3500	5.2506	1.2043	0.3264	0.1106	-0.1966	0.2256	92.7838
5	0.4500	4.4508	1.1505	0.3118	0.1195	-0.2139	0.2450	92.6166
6	0.5500	3.6510	1.0746	0.2912	0.1241	-0.2230	0.2552	92.5148
7	0.6500	2.8511	0.9743	0.2641	0.1246	-0.2245	0.2568	92.4564
8	0.7500	2.0513	0.8438	0.2287	0.1205	-0.2172	0.2484	92.4529
9	0.8500	1.2515	0.6676	0.1809	0.1085	-0.1952	0.2233	92.5091
10	0.9500	0.4258	0.4008	0.1086	0.0768	-0.1369	0.1570	92.7293

---PREDICTED---		TRAILING VORTICITY		SEPARATION VORTICITY	
KVLE	KVSE	Y	GAM/V2PI	Y	GAM/V2PI
4.16155	-0.00000	3.3559	0.34558	2.1711	0.22611

(d) Page 4.

Figure 8.- Continued.

SUMMARY OF VORTEX STRENGTHS AND POSITION AT X = 16.001

VORTEX	GAMMA/2*PI*V	Y	Z
1	0.085127	0.397	1.788
2	0.345578	4.386	-0.790
3	0.226110	3.701	0.162

VELOCITY INDUCED AT SPECIFIED FIELD POINTS ON TAIL (BESKIN)

	X	Y	Z	V/V(INF)	W/V(INF)
1	32.354	2.677	0.000	0.0000	1.5098E-01
2	35.324	2.677	0.000	0.0000	1.5584E-01
3	38.293	2.677	0.000	0.0000	1.5844E-01
4	41.262	2.677	0.000	0.0000	1.5844E-01
5	44.231	2.677	0.000	0.0000	1.5752E-01
6	47.200	2.677	0.000	0.0000	1.5243E-01
7	50.169	2.677	0.000	0.0000	1.4215E-01
8	53.138	2.677	0.000	0.0000	1.2595E-01
9	34.376	4.032	0.000	0.0000	6.8058E-02
10	37.033	4.032	0.000	0.0000	6.9481E-02
11	39.690	4.032	0.000	0.0000	6.9852E-02
12	42.348	4.032	0.000	0.0000	6.9852E-02
13	45.005	4.032	0.000	0.0000	6.9128E-02
14	47.662	4.032	0.000	0.0000	6.6554E-02
15	50.320	4.032	0.000	0.0000	6.2388E-02
16	52.977	4.032	0.000	0.0000	5.5986E-02
17	36.397	5.387	0.000	0.0000	3.8792E-02
18	38.743	5.387	0.000	0.0000	3.9134E-02
19	41.088	5.387	0.000	0.0000	3.9134E-02
20	43.434	5.387	0.000	0.0000	3.9011E-02
21	45.779	5.387	0.000	0.0000	3.8363E-02
22	48.125	5.387	0.000	0.0000	3.6926E-02
23	50.471	5.387	0.000	0.0000	3.4796E-02
24	52.816	5.387	0.000	0.0000	3.1623E-02
25	38.418	6.742	0.000	0.0000	2.4985E-02
26	40.452	6.742	0.000	0.0000	2.4985E-02
27	42.486	6.742	0.000	0.0000	2.4985E-02
28	44.520	6.742	0.000	0.0000	2.4815E-02
29	46.554	6.742	0.000	0.0000	2.4262E-02
30	48.588	6.742	0.000	0.0000	2.3347E-02
31	50.621	6.742	0.000	0.0000	2.2117E-02
32	52.655	6.742	0.000	0.0000	2.0356E-02
33	40.439	8.097	0.000	0.0000	1.7323E-02
34	42.162	8.097	0.000	0.0000	1.7323E-02
35	43.884	8.097	0.000	0.0000	1.7242E-02
36	45.606	8.097	0.000	0.0000	1.7018E-02
37	47.328	8.097	0.000	0.0000	1.6621E-02
38	49.050	8.097	0.000	0.0000	1.6029E-02
39	50.772	8.097	0.000	0.0000	1.5265E-02
40	52.494	8.097	0.000	0.0000	1.4228E-02
41	42.453	9.447	0.000	0.0000	1.2726E-02
42	43.865	9.447	0.000	0.0000	1.2668E-02
43	45.276	9.447	0.000	0.0000	1.2552E-02
44	46.688	9.447	0.000	0.0000	1.2337E-02
45	48.099	9.447	0.000	0.0000	1.2014E-02
46	49.511	9.447	0.000	0.0000	1.1641E-02
47	50.922	9.447	0.000	0.0000	1.1164E-02
48	52.334	9.447	0.000	0.0000	1.0537E-02
49	44.475	10.802	0.000	0.0000	9.6688E-03
50	45.574	10.802	0.000	0.0000	9.5661E-03
51	46.674	10.802	0.000	0.0000	9.4378E-03
52	47.774	10.802	0.000	0.0000	9.2525E-03
53	48.874	10.802	0.000	0.0000	9.0404E-03
54	49.973	10.802	0.000	0.0000	8.7832E-03
55	51.073	10.802	0.000	0.0000	8.5006E-03
56	52.173	10.802	0.000	0.0000	8.1248E-03
57	46.503	12.162	0.000	0.0000	7.4608E-03
58	47.290	12.162	0.000	0.0000	7.3732E-03
59	48.077	12.162	0.000	0.0000	7.2527E-03
60	48.864	12.162	0.000	0.0000	7.1331E-03
61	49.651	12.162	0.000	0.0000	6.9949E-03
62	50.438	12.162	0.000	0.0000	6.8342E-03
63	51.224	12.162	0.000	0.0000	6.6755E-03
64	52.011	12.162	0.000	0.0000	6.4612E-03
65	48.525	13.517	0.000	0.0000	5.8164E-03
66	49.000	13.517	0.000	0.0000	5.7581E-03
67	49.475	13.517	0.000	0.0000	5.6921E-03
68	49.950	13.517	0.000	0.0000	5.6132E-03
69	50.425	13.517	0.000	0.0000	5.5348E-03
70	50.900	13.517	0.000	0.0000	5.4570E-03
71	51.375	13.517	0.000	0.0000	5.3798E-03
72	51.850	13.517	0.000	0.0000	5.2728E-03
73	50.546	14.872	0.000	0.0000	4.5559E-03
74	50.709	14.872	0.000	0.0000	4.5337E-03
75	50.873	14.872	0.000	0.0000	4.5117E-03
76	51.036	14.872	0.000	0.0000	4.4897E-03
77	51.199	14.872	0.000	0.0000	4.4677E-03
78	51.363	14.872	0.000	0.0000	4.4458E-03
79	51.526	14.872	0.000	0.0000	4.4240E-03
80	51.689	14.872	0.000	0.0000	4.3906E-03

(e) Page 5.

Figure 8.- Continued.

VELOCITY INDUCED AT SPECIFIED FIELD POINTS ON TAIL

	X	Y	Z	V/V(INF)	W/V(INF)
1	32.354	2.677	0.000	1.7874E-01	-2.2402E-01
2	35.324	2.677	0.000	3.1029E-01	-3.2266E-01
3	38.293	2.677	0.000	1.0757E-01	-3.3855E-01
4	41.262	2.677	0.000	7.0380E-02	-2.6807E-01
5	44.231	2.677	0.000	5.8058E-02	-2.2844E-01
6	47.200	2.677	0.000	5.3841E-02	-2.0138E-01
7	50.169	2.677	0.000	5.7258E-02	-1.8460E-01
8	53.138	2.677	0.000	6.5031E-02	-1.7365E-01
9	34.376	4.032	0.000	1.4847E-01	-6.5528E-02
10	37.033	4.032	0.000	1.4903E-01	-2.9389E-02
11	39.690	4.032	0.000	2.4147E-01	3.1539E-02
12	42.348	4.032	0.000	2.5815E-01	-2.1987E-01
13	45.005	4.032	0.000	1.3125E-01	-1.8253E-01
14	47.662	4.032	0.000	1.0230E-01	-1.4819E-01
15	50.320	4.032	0.000	9.3787E-02	-1.2672E-01
16	52.977	4.032	0.000	9.5011E-02	-1.1363E-01
17	36.397	5.387	0.000	1.1078E-01	3.4582E-03
18	38.743	5.387	0.000	1.0657E-01	1.1729E-02
19	41.088	5.387	0.000	1.0651E-01	2.5412E-02
20	43.434	5.387	0.000	1.1753E-01	4.7628E-02
21	45.779	5.387	0.000	1.6110E-01	7.2286E-02
22	48.125	5.387	0.000	2.4523E-01	3.5050E-02
23	50.471	5.387	0.000	2.3979E-01	-6.7861E-02
24	52.816	5.387	0.000	1.9439E-01	-8.6268E-02
25	38.418	6.742	0.000	7.2975E-02	2.6455E-02
26	40.452	6.742	0.000	7.1481E-02	2.8525E-02
27	42.486	6.742	0.000	7.0389E-02	3.1153E-02
28	44.520	6.742	0.000	6.9659E-02	3.4714E-02
29	46.554	6.742	0.000	6.9646E-02	3.9654E-02
30	48.588	6.742	0.000	7.1254E-02	4.6268E-02
31	50.621	6.742	0.000	7.5934E-02	5.4384E-02
32	57.655	6.742	0.000	8.1855E-02	5.9478E-02
33	40.439	8.097	0.000	4.7903E-02	3.0158E-02
34	42.162	8.097	0.000	4.7487E-02	3.0621E-02
35	43.884	8.097	0.000	4.7044E-02	3.1046E-02
36	45.606	8.097	0.000	4.6421E-02	3.1524E-02
37	47.328	8.097	0.000	4.5594E-02	3.2300E-02
38	49.050	8.097	0.000	4.4744E-02	3.3695E-02
39	50.772	8.097	0.000	4.4245E-02	3.5946E-02
40	52.494	8.097	0.000	4.4111E-02	3.8053E-02
41	42.453	9.447	0.000	3.2506E-02	2.7844E-02
42	43.865	9.447	0.000	3.2397E-02	2.7858E-02
43	45.276	9.447	0.000	3.2203E-02	2.7784E-02
44	46.688	9.447	0.000	3.1867E-02	2.7712E-02
45	48.099	9.447	0.000	3.1380E-02	2.7769E-02
46	49.511	9.447	0.000	3.0789E-02	2.8095E-02
47	50.922	9.447	0.000	3.0231E-02	2.8818E-02
48	52.334	9.447	0.000	2.9763E-02	2.9621E-02
49	44.475	10.802	0.000	2.2834E-02	2.4144E-02
50	45.574	10.802	0.000	2.2782E-02	2.4041E-02
51	46.674	10.802	0.000	2.2666E-02	2.3903E-02
52	47.774	10.802	0.000	2.2474E-02	2.3790E-02
53	48.874	10.802	0.000	2.2204E-02	2.3743E-02
54	49.973	10.802	0.000	2.1881E-02	2.3832E-02
55	51.073	10.802	0.000	2.1540E-02	2.4088E-02
56	52.173	10.802	0.000	2.1216E-02	2.4417E-02
57	46.503	12.162	0.000	1.6526E-02	2.0513E-02
58	47.290	12.162	0.000	1.6472E-02	2.0416E-02
59	48.077	12.162	0.000	1.6389E-02	2.0335E-02
60	48.864	12.162	0.000	1.6269E-02	2.0270E-02
61	49.651	12.162	0.000	1.6122E-02	2.0251E-02
62	50.438	12.162	0.000	1.5955E-02	2.0292E-02
63	51.224	12.162	0.000	1.5775E-02	2.0380E-02
64	52.011	12.162	0.000	1.5597E-02	2.0534E-02
65	48.525	13.517	0.000	1.2250E-02	1.7405E-02
66	49.000	13.517	0.000	1.2204E-02	1.7366E-02
67	49.475	13.517	0.000	1.2145E-02	1.7343E-02
68	49.950	13.517	0.000	1.2082E-02	1.7334E-02
69	50.425	13.517	0.000	1.2009E-02	1.7342E-02
70	50.900	13.517	0.000	1.1933E-02	1.7358E-02
71	51.375	13.517	0.000	1.1851E-02	1.7387E-02
72	51.850	13.517	0.000	1.1770E-02	1.7442E-02
73	50.546	14.872	0.000	9.2179E-03	1.4923E-02
74	50.709	14.872	0.000	9.1990E-03	1.4924E-02
75	50.873	14.872	0.000	9.1798E-03	1.4926E-02
76	51.036	14.872	0.000	9.1600E-03	1.4928E-02
77	51.199	14.872	0.000	9.1388E-03	1.4933E-02
78	51.363	14.872	0.000	9.1174E-03	1.4938E-02
79	51.526	14.872	0.000	9.0959E-03	1.4943E-02
80	51.689	14.872	0.000	9.0750E-03	1.4956E-02

(e) Continued.

Figure 8.- Continued.

TOTAL INDUCED VELOCITY AT SPECIFIED FIELD POINTS ON TAIL

	X	Y	Z	V/V(INF)	W/V(INF)
1	32.354	2.677	0.000	1.7874E-01	-7.3040E-02
2	35.324	2.677	0.000	3.1029E-01	-1.6683E-01
3	38.293	2.677	0.000	1.0757E-01	-1.8011E-01
4	41.262	2.677	0.000	7.0380E-02	-1.0962E-01
5	44.231	2.677	0.000	5.8058E-02	-7.0923E-02
6	47.200	2.677	0.000	5.3841E-02	-4.8957E-02
7	50.169	2.677	0.000	5.7258E-02	-4.2450E-02
8	53.138	2.677	0.000	6.5031E-02	-4.7702E-02
9	34.376	4.032	0.000	1.4847E-01	2.5303E-03
10	37.033	4.032	0.000	1.4903E-01	4.0092E-02
11	39.690	4.032	0.000	2.4147E-01	1.0139E-01
12	42.348	4.032	0.000	2.5815E-01	-1.5002E-01
13	45.005	4.032	0.000	1.3125E-01	-1.1340E-01
14	47.662	4.032	0.000	1.0230E-01	-8.1641E-02
15	50.320	4.032	0.000	9.3787E-02	-6.4330E-02
16	52.977	4.032	0.000	9.5011E-02	-5.7645E-02
17	36.397	5.387	0.000	1.1078E-01	4.2250E-02
18	38.743	5.387	0.000	1.0657E-01	5.0863E-02
19	41.088	5.387	0.000	1.0651E-01	6.4546E-02
20	43.434	5.387	0.000	1.1753E-01	8.6639E-02
21	45.779	5.387	0.000	1.6110E-01	1.1065E-01
22	48.125	5.387	0.000	2.4523E-01	7.1976E-02
23	50.471	5.387	0.000	2.3979E-01	-3.3064E-02
24	52.816	5.387	0.000	1.9439E-01	-5.4644E-02
25	38.418	6.742	0.000	7.2975E-02	5.1441E-02
26	40.452	6.742	0.000	7.1481E-02	5.3511E-02
27	42.486	6.742	0.000	7.0389E-02	5.6139E-02
28	44.520	6.742	0.000	6.9659E-02	5.9529E-02
29	46.554	6.742	0.000	6.9646E-02	6.3915E-02
30	48.588	6.742	0.000	7.1254E-02	6.9615E-02
31	50.621	6.742	0.000	7.5934E-02	7.6501E-02
32	52.655	6.742	0.000	8.1855E-02	7.9833E-02
33	40.439	8.097	0.000	4.7903E-02	4.7481E-02
34	42.162	8.097	0.000	4.7487E-02	4.7944E-02
35	43.884	8.097	0.000	4.7044E-02	4.8288E-02
36	45.606	8.097	0.000	4.6421E-02	4.8542E-02
37	47.328	8.097	0.000	4.5594E-02	4.8921E-02
38	49.050	8.097	0.000	4.4744E-02	4.9724E-02
39	50.772	8.097	0.000	4.4245E-02	5.1212E-02
40	52.494	8.097	0.000	4.4111E-02	5.2282E-02
41	42.453	9.447	0.000	3.2506E-02	4.0570E-02
42	43.865	9.447	0.000	3.2397E-02	4.0526E-02
43	45.276	9.447	0.000	3.2203E-02	4.0337E-02
44	46.688	9.447	0.000	3.1867E-02	4.0049E-02
45	48.099	9.447	0.000	3.1380E-02	3.9784E-02
46	49.511	9.447	0.000	3.0789E-02	3.9736E-02
47	50.922	9.447	0.000	3.0231E-02	3.9982E-02
48	52.334	9.447	0.000	2.9763E-02	4.0158E-02
49	44.475	10.802	0.000	2.2834E-02	3.3813E-02
50	45.574	10.802	0.000	2.2782E-02	3.3607E-02
51	46.674	10.802	0.000	2.2666E-02	3.3341E-02
52	47.774	10.802	0.000	2.2474E-02	3.3042E-02
53	48.874	10.802	0.000	2.2204E-02	3.2783E-02
54	49.973	10.802	0.000	2.1881E-02	3.2616E-02
55	51.073	10.802	0.000	2.1540E-02	3.2589E-02
56	52.173	10.802	0.000	2.1216E-02	3.2542E-02
57	46.503	12.162	0.000	1.6526E-02	2.7974E-02
58	47.290	12.162	0.000	1.6472E-02	2.7789E-02
59	48.077	12.162	0.000	1.6389E-02	2.7588E-02
60	48.864	12.162	0.000	1.6269E-02	2.7403E-02
61	49.651	12.162	0.000	1.6122E-02	2.7246E-02
62	50.438	12.162	0.000	1.5955E-02	2.7126E-02
63	51.224	12.162	0.000	1.5775E-02	2.7055E-02
64	52.011	12.162	0.000	1.5597E-02	2.6996E-02
65	48.525	13.517	0.000	1.2250E-02	2.3222E-02
66	49.000	13.517	0.000	1.2204E-02	2.3124E-02
67	49.475	13.517	0.000	1.2145E-02	2.3036E-02
68	49.950	13.517	0.000	1.2082E-02	2.2947E-02
69	50.425	13.517	0.000	1.2009E-02	2.2876E-02
70	50.900	13.517	0.000	1.1933E-02	2.2815E-02
71	51.375	13.517	0.000	1.1851E-02	2.2767E-02
72	51.850	13.517	0.000	1.1770E-02	2.2714E-02
73	50.546	14.872	0.000	9.2179E-03	1.9479E-02
74	50.709	14.872	0.000	9.1990E-03	1.9458E-02
75	50.873	14.872	0.000	9.1798E-03	1.9437E-02
76	51.036	14.872	0.000	9.1600E-03	1.9418E-02
77	51.199	14.872	0.000	9.1388E-03	1.9400E-02
78	51.363	14.872	0.000	9.1174E-03	1.9383E-02
79	51.526	14.872	0.000	9.0959E-03	1.9367E-02
80	51.689	14.872	0.000	9.0750E-03	1.9346E-02

(e) Concluded.

Figure 8.- Continued.

TAIL

LIFTING SURFACE CALCULATION

FLOW CONDITIONS - ALPHA 16.000, BETA 1.0000

LATTICE ARRANGEMENT - 8 CHORDWISE AND 10 SPANWISE ELEMENTS

REFERENCE QUANTITIES - PLANFORM AREA S = 339.479, AVERAGE CHORD CAVG = 12.53

C(ROOT)	C(TIP)	B/2	DIHEDRAL	XCM	ZCM	Y(SEP)	ETA(SEP)
25.000	0.000	13.550	0.000	0.000	0.000	13.550	1.000

STATION	Y	SWEEP LE	SWEEP TE
1	1.355	59.000	-10.000
2	2.710	59.000	-10.000
3	4.065	59.000	-10.000
4	5.420	59.000	-10.000
5	6.775	59.000	-10.000
6	8.120	59.000	-10.000
7	9.485	59.000	-10.000
8	10.840	59.000	-10.000
9	12.195	59.000	-10.000
10	13.550	59.000	-10.000

CNP	CNV	CNVS	CLP	CMP	CDI	CA	CDI/CL2	CXV	CYV	CRP
0.69736	0.24320	0.00000	0.70679	-0.75117	0.06511	-0.13223	0.13035	0.06611	-0.10206	0.70978

X COORDINATES OF CENTERS OF PRESSURE DUE TO

POTENTIAL LIFT	LEADING EDGE SUCTION	SIDE EDGE SUCTION	OVERALL
-13.4937	-12.7614	-0.0000	-13.3043

SPANWISE LOAD DISTRIBUTION

STATION	Y/(B/2)	LOCAL CHORD,C	CL*C/CL*CAVG	CL*C/(2*B)	CX*C/(2*B)	CY*C/(2*B)	CSUC*C/(2*B)	ETA,DEG.
1	0.0500	23.7530	1.1358	0.1855	0.0076	-0.0101	0.0126	96.2013
2	0.1500	21.2590	1.1961	0.1954	0.0175	-0.0258	0.0312	93.1809
3	0.2500	18.7650	1.2785	0.2089	0.0257	-0.0397	0.0473	91.8755
4	0.3500	16.2709	1.2946	0.2115	0.0331	-0.0513	0.0611	91.8737
5	0.4500	13.7769	1.2120	0.1980	0.0379	-0.0589	0.0700	91.7272
6	0.5496	11.2921	1.1069	0.1808	0.0406	-0.0633	0.0753	91.6849
7	0.6496	8.7981	0.9797	0.1600	0.0416	-0.0648	0.0770	91.6808
8	0.7500	6.2949	0.8243	0.1347	0.0404	-0.0629	0.0748	91.6962
9	0.8500	3.8009	0.6268	0.1024	0.0361	-0.0562	0.0668	91.7396
10	0.9500	1.2769	0.3463	0.0566	0.0251	-0.0387	0.0461	91.8927

---PREDICTED---		TRAILING VORTICITY		SEPARATION VORTICITY	
KVLF	KVSE	Y	GAM/V2PI	Y	GAM/V2PI
3.20097	-0.00000	11.9301	0.75897	7.6678	0.16884

SUMMARY OF VORTEX STRENGTHS AND POSITION AT X = 55.001

VORTEX	GAMMA/2*PI*V	Y	Z
1	0.085127	5.660	1.135
2	0.345578	2.941	5.499
3	0.226110	3.062	3.738
4	0.758969	13.930	0.000
5	0.168842	9.668	1.918

(f) Page 6.

Figure 8.- Continued.

		-- INCIDENCE --		---- WING -----		---- TAIL -----		
ALPHA	M	WING	TAIL	KVLE*	KVSE*	KVLE*	KVSE*	W/V LIMIT
16.00	0.00	10.000	0.000	0.530	0.000	0.410	0.000	1.000

SUMMARY OF FORCE AND PITCHING MOMENT COEFFICIENTS

		CN	CM	XCP	CL	CDI	CA
NOSE...							
	POTENTIAL	2.850E-03	5.191E-03	4.497E 00	2.740E-03	7.857E-04	
	VORTEX	1.005E-03	1.855E-03	4.026E 00	9.662E-04	2.771E-04	
WING...							
	W(B) POTENTIAL	7.161E-02	1.022E-01	1.202E 01	6.436E-02	1.845E-02	1.263E-02
	W(B) VORTEX,LE	3.031E-02	4.355E-02	1.184E 01	2.724E-02	7.812E-03	5.345E-03
	W(B) VORTEX,SE	0.000	0.000	7.500E 00	0.000	0.000	0.000
BODY...							
	B(W) POTENTIAL	1.682E-02	2.400E-02	1.202E 01	1.616E-02	4.635E-03	
	B(W) VORTEX,LE	9.765E-03	1.403E-02	1.184E 01	9.386E-03	2.692E-03	
	B(W) VORTEX,SE	0.000	0.000	7.500E 00	0.000	0.000	
AFTERBODY...		1.755E-03	4.069E-04	3.487E 01	1.687E-03	4.836E-04	
TAIL...							
	T(B) POTENTIAL	5.308E-01	-8.871E-02	4.249E 01	5.102E-01	1.463E-01	0.000
	T(B) VORTEX,LE	7.590E-02	-9.776E-03	4.176E 01	7.296E-02	2.092E-02	0.000
	T(B) VORTEX,SE	0.000	0.000	2.900E 01	0.000	0.000	0.000
BODY...							
	B(T) POTENTIAL	7.621E-02	-1.274E-02	4.249E 01	7.326E-02	2.101E-02	
	B(T) VORTEX,LE	1.586E-02	-2.042E-03	4.176E 01	1.524E-02	4.370E-03	
	B(T) VORTEX,SE	0.000	0.000	2.900E 01	0.000	0.000	
TOTAL CONFIGURATION...		8.329E-01	7.798E-02	3.751E 01	7.942E-01	2.277E-01	1.797E-02

CDI/CL**2 = 3.610E-01

(g) Page 7.

Figure 8.- Continued.

```
SUMMARY OF FREE VORTEX TRAJECTORIES

          VORTEX 1 ---  BODY VORTEX FROM NOSE
          VORTEX 2 ---  WING TRAILING VORTEX
          VORTEX 3 ---  WING SEPARATION VORTEX

     VORTEX    GAMMA/2*PI*V
        1        0.085127
        2        0.345578
        3        0.226110


     X        DX           A          S          RO         DA/DX
  7.5000  0.5000000     0.8400     0.8400     0.8400     0.0791

VORTEX SIGMA(REAL)    (IMAG)
  1      4.065E-01  1.206E 00


     X        DX           A          S          RO         DA/DX
  8.5000  0.2500000     0.9175     1.5301     1.0401     0.0762

VORTEX SIGMA(REAL)    (IMAG)
  1      3.893E-01  1.338E 00


     X        DX           A          S          RO         DA/DX
  9.5000  0.5000000     0.9925     2.0302     1.2577     0.0750

VORTEX SIGMA(REAL)    (IMAG)
  1      3.735E-01  1.441E 00

                              • • •

     X        DX           A          S          RO         DA/DX
 16.0000  0.5000000     1.4600     1.4600     1.4600     0.0625

VORTEX SIGMA(REAL)    (IMAG)
  1      3.973E-01  1.788E 00


     X        DX           A          S          RO         DA/DX
 16.0010  0.5000000     1.4601     1.4601     1.4601     0.0625

VORTEX SIGMA(REAL)    (IMAG)
  1      3.973E-01  1.788E 00
  2      4.386E 00 -7.901E-01
  3      3.201E 00  1.622E-01


     X        DX           A          S          RO         DA/DX
 17.0010  0.2500000     1.5201     1.5201     1.5201     0.0587

VORTEX SIGMA(REAL)    (IMAG)
  1      4.430E-01  1.810E 00
  2      4.481E 00 -4.610E-01
  3      3.117E 00  1.570E-01

                              • • •
```

(h) Page 8.

Figure 8.- Continued.

29.0010	0.5000000	1.9350	1.9356	1.9350	0.0150

VORTEX	SIGMA(REAL)	(IMAG)
1	1.909E 00	1.010E 00
2	3.552E 00	2.539E 00
3	4.880E 00	1.847E 00

X	DX	A	S	RO	DA/DX
30.0010	0.5000000	1.9500	2.5653	2.0238	0.0143

VORTEX	SIGMA(REAL)	(IMAG)
1	2.051E 00	8.237E-01
2	3.489E 00	2.600E 00
3	4.950E 00	2.284E 00

X	DX	A	S	RO	DA/DX
31.0010	0.5000000	1.9650	3.1950	2.2018	0.0135

VORTEX	SIGMA(REAL)	(IMAG)
1	2.165E 00	6.690E-01
2	3.457E 00	2.648E 00
3	4.957E 00	2.733E 00

• • •

X	DX	A	S	RO	DA/DX
51.0010	0.5000000	1.8991	15.2196	7.7283	-0.0343

VORTEX	SIGMA(REAL)	(IMAG)
1	5.553E 00	4.832E-01
2	3.557E 00	4.993E 00
3	2.528E 00	3.710E 00

X	DX	A	S	RO	DA/DX
52.0010	0.5000000	1.8626	13.0185	6.6425	-0.0412

VORTEX	SIGMA(REAL)	(IMAG)
1	5.650E 00	5.058E-01
2	3.410E 00	5.087E 00
3	2.668E 00	3.586E 00

X	DX	A	S	RO	DA/DX
53.0010	0.5000000	1.8163	7.3942	3.9202	-0.0497

VORTEX	SIGMA(REAL)	(IMAG)
1	5.655E 00	5.482E-01
2	3.243E 00	5.197E 00
3	2.800E 00	3.536E 00

X	DX	A	S	RO	DA/DX
54.0010	0.1250000	1.7699	1.7699	1.7699	-0.0582

VORTEX	SIGMA(REAL)	(IMAG)
1	5.607E 00	8.390E-01
2	3.085E 00	5.350E 00
3	2.923E 00	3.603E 00

X	DX	A	S	RO	DA/DX
55.0010	0.2500000	1.6999	1.6999	1.6999	-0.0653

VORTEX	SIGMA(REAL)	(IMAG)
1	5.660E 00	1.135E 00
2	2.941E 00	5.499E 00
3	3.062E 00	3.738E 00

(h) Concluded.

Figure 8.- Concluded.

PART IV - SUPSON COMPUTER PROGRAM

Introduction

This computer program predicts the static longitudinal aerodynamic characteristics of wing-body-tail combinations at supersonic speeds. It is an extension of the method of reference 2 to angles of attack for which symmetrical body vortices are shed from the nose of the configuration and leading-edge and side-edge separation vortices are shed from the wing and tail. A lifting-surface method (refs. 12 and 13), modified to include interference velocity fields in the form of induced camber and modified to compute leading-edge suction distributions, represents the wing and tail surfaces.

The program is written in FORTRAN IV for the IBM 360 series machines. No tapes, drums, or disks other than the standard input/output units are required. Minor changes are required to run the program on other machines such as the CDC 6600. Typical running time on the IBM 360/67 for a wing-body-tail configuration is approximately 1 minute per angle of attack. Actual running time depends on the number of panels representing the lifting-surfaces. Some specific running times are noted in the discussion of the sample cases.

The following sections present descriptions of the method, program, input, and output. A program listing and sample cases are included. The algebraic notation used in this section is the same as that used in reference 1. A list of symbols from reference 1 is included.

List of Symbols

AR	aspect ratio
a	local body radius
b	semispan
C_{D_i}	induced drag coefficient
C_L	lift coefficient, $\frac{L}{qS}$
C_A	axial-force coefficient
C_m	pitching-moment coefficient, $\frac{M}{qS\ell}$

C_N	normal-force coefficient, $\frac{N}{qS}$
c	local chord
c_n	section normal-force coefficient, $\frac{1}{c}\int_{LE}^{TE} \frac{\Delta p}{q}\,dx$
c_s	section leading-edge suction coefficient
c_x	x-direction section suction coefficient
c_y	y-direction section suction coefficient
K_v^*	vortex-lift ratio
L	lift force
ℓ	reference length
M	pitching moment about center of moments, or free-stream Mach number
N	normal force
Δp	static pressure difference between lower and upper surfaces of lifting surface
q	free-stream dynamic pressure
r	body radius
r_N	radius of base of nose
S	reference area
s	semispan of lifting surface
u,v,w	perturbation velocities along x,y,z directions, respectively
V	free-stream velocity
x,y,z	configuration coordinates with origin at body nose
x_{ac}	x location of aerodynamic center
x_{HL}	x location of lifting-surface hinge line
x_m	x location of center of moments

x_s	x position for onset of separation from body nose
$\bar{x}$	center-of-pressure location
z_{HL}	height of lifting-surface hinge line
α	body angle of attack
β	$\sqrt{M^2 - 1}$
Γ_B	right body-vortex strength, positive counterclockwise when viewed from rear of configurations
Γ_n	n'th separation vortex strength on right wing panel
Γ_t	trailing-vortex strength on right wing panel
δ	lifting-surface deflection angle, positive trailing edge down
θ_N	nose angle, degrees
Λ	sweep angle
ρ	density
σ	complex vortex position, y + iz

Subscripts

A	afterbody
avg	average
B(T)	body in presence of tail
B(W)	body in presence of wing
e	tail or empennage
HL	hinge line
LE	leading edge
N	nose
p	potential
root	root chord
SE	side edge

T(B)	tail in presence of body
TE	trailing edge
tip	tip chord
v	vortex
W(B)	wing in presence of body
w	wing

Description of Method

A brief description of the method is presented herein. The user should consult reference 1 for a complete description and details of the theoretical approach.

An axisymmetric nose at some moderate angle of attack sheds a symmetric pair of body vortices. These shed body vortices, whose strength and initial position are determined from data correlations, are tracked downstream past the wing using slender-body techniques in the crossflow plane. One exception to this trajectory calculation is that the vortices move parallel to the wing if the leading edge is supersonic. The vortex-induced velocities are computed at the wing control points and combined with the Beskin upwash induced by the body to obtain the total upwash induced on each wing panel. This, added to the free-stream contribution, results in a total local incidence angle distribution over the wing.

The wing is modeled by a constant pressure panel scheme in the form of a prediction program obtained from R. Carmichael of the NASA, Ames Research Center. The result from this method is a pressure distribution on the wing. An equivalent circulation distribution on the wing is obtained from the pressure loading and this is used to compute a distribution of leading-edge suction and side-edge suction (if present) and their associated vortex positions and strengths. The vortex lift on the leading edge and side edge is obtained from the suction distribution through the Polhamus vortex-lift analogy (ref. 9) and correlation curves.

The trajectories of the body vortices, the wing trailing vortex, and the wing leading-edge separation vortices are computed downstream past the afterbody and horizontal tail. These trajectories are computed in the crossflow plane considering mutual interference between the vortices and interference from their images in the body. If the tail leading edge is supersonic, the vortices move parallel to the tail surface. The induced velocity field on the tail panels is computed, and the tail loading is obtained in a manner similar to that just described for the wing. The forces on the body in the presence of a wing and tail are computed by the method of reference 2. The free vortex-induced forces on the nose and afterbody are computed using the method of Sacks (ref. 10).

The forces and moments on the entire configuration are obtained by summing the contributions of the various components. The forces are resolved into normal force and axial force (excluding frictional drag and wave drag), and lift and induced drag.

Description of Program

The SUPSON computer program consists of a main program, nine function subprograms, and twenty-five subroutine subprograms. The main program (SP01) accepts most of the input, prints a portion of the output, and generally directs the flow of the calculation. The subprograms provide specific services to the main program during the calculation procedure. Since the lifting-surface portion of the program was a separate wing-alone prediction program which was incorporated into the prediction scheme, there are several options present in this part of the program which are not used. These were left in the program so that the original lifting-surface procedure could be left intact as much as possible; however, this does result in portions of subprograms being carried along but never used. These surplus calculations are short and do not cause a large core storage penalty by their presence. The following is a list of subprograms and their general purpose.

Subroutine SSWING is the former main program of the Carmichael-Woodward lifting-surface method. It accepts description of the wing (or tail) geometry through namelist input and then sets up locations of control points and divides the wing up into the specified number of constant pressure panels. It then calculates an array of influence functions which are used later in computing wing-loading distributions.

Subroutine TABLE sets up a table of coordinates describing the configuration at specific axial stations required in the vortex trajectory calculations. The body radius at the required stations are obtained by linear interpolation in the input table of body coordinates. Body slopes are computed by a simple differencing scheme.

Subroutine FORCE computes the loading on the lifting surface considering both the geometric camber and induced camber. The bound circulation distribution corresponding to the potential loading is computed as is the strength and position of the equivalent trailing vortex.

Subroutine EDGFRC computes the distribution of leading-edge and side-edge suction. The leading-edge and side-edge vortex lift is computed along with the strength and lateral position of the associated separation vortex.

Subroutine MATRIX computes the aerodynamic influence matrix for each lifting surface.

Subroutine CNVTX computes the vortex-induced force and center of pressure on the afterbody using the method of reference 10.

Subroutine BDYVTX uses tables derived from data correlations to look up the strength and position of the pair of symmetric vortices shed from the body nose.

Subroutine CNVNZ computes the nose vortex-induced normal force and center of pressure on the body nose using the method of reference 10.

Subroutine SHAPE does a table look-up for the body radius and slope and local lifting-surface semispan at any prescribed axial station.

Subroutine FILL is used to fill in intermediate locations in an array using a linear interpolation procedure.

Subroutine ZSECT is a general airfoil section calculation routine which is used only when the lifting surfaces are assumed to have finite thickness. The available section shapes are specified by the variable SECT as described in the discussion of input preparation.

Subroutine TCOMP computes the pressure function for a wedge-shaped airfoil of triangular planform. This routine is used only when the lifting surface has a finite thickness.

Subroutine TAINT is used to interpolate in tables of airfoil section thicknesses and surface slopes.

Subroutines LINEQS and SOLVE are used together to solve the set of linear simultaneous equations for the loading on the lifting surfaces.

Subroutine COMP computes the downwash function for a uniformly loaded triangular shaped panel.

Subroutine EXTVEL computes the free vortex-induced velocity at wing or tail control points.

Subroutine TRJTRY computes the trajectories of the free vortices past the configuration using the subroutines FCT, OUTP, HPCG, and SHAPE.

Subroutine FCT computes the derivatives for the equations of motion for each free vortex.

Subroutine HPCG is a predictor-corrector integration package which uses a Runge-Kutta starting procedure.

Subroutine OUTPUT stores the vortex positions in a table at specified intervals in x. When necessary, some diagnostic information on the vortex trajectories is available as optional output.

Subroutine ZVTX determines the vertical position of the leading-edge separation vortex using a table look-up of correlated data for delta wings.

Subroutine WPANL calculates the upwash induced by the constant pressure panels comprising the lifting surface.

Subprograms KFACT, CH1416, EQ14, CHRT8, EQ21, EQ24, EQ24L, EQ26, EQ30, EQ30L, and EQ31 are used to compute the lift and center of pressure on the body in the presence of a lifting surface. These subprograms perform the same functions in SUPSON as they do in R1307 and the individual descriptions are not repeated here.

Description of Input

Variable definitions.- The format of the input cards for the SUBSON program is shown in figure 1. In this figure the program variable name is shown as well as the card columns in which the value is punched and the format in which it is punched. The following is a table of the input variables along with the algebraic symbol where applicable. The variable is defined and its limits shown where necessary. The algebraic notation used in defining the configuration is shown in figure 2. A discussion of the preparation of the input is presented in the section following the table.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
Item 1		
NHEAD		Number of heading cards.
NTBL		Number of entries in table of body coordinates. $5 \le$ NTBL $\le$ (96 less the total number of columns on both wing and tail).
NPRINT		Index controlling optional output: NPRINT = -2 Minimum output. Final aerodynamic characteristics only. = -1 Abbreviated output. Final aerodynamic characteristics plus details of lifting-surface results. = 0 No optional output. = 1 Some optional output. = 4 Large amount of optional output. > 4 All optional output including diagnostic information (not recommended for general use).
MPRINT		Index controlling diagnostic output: MPRINT = 0 No additional output. = 1 Output vortex trajectories during calculations. (This option should be used only if program has terminated execution during a previous trajectory calculation.)
NOSEV		NOSEV = 0 No nose separation vortex pair. = 1 Nose separation vortex pair included.
NREGNW		Number of regions into which wing is divided. $1 \le$ NREGNW ≤ 20
NALPW		NALPW = 0 Uncambered and untwisted wing. > 0 Cambered and/or twisted wing, local angles must be input.
NSEPW		Number of leading-edge separation vortices shed from wing. $1 \le$ NSEPW ≤ 2
NREGNT		Number of regions into which tail is divided. $1 \le$ NREGNT ≤ 20 NREGNT = 0 No tail.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
NALPT		NALPT = 0 Uncambered and untwisted tail. $>$ 0 Cambered and/or twisted tail, local angles must be input.
NSEPT		Number of leading-edge separation vortices shed from tail. $1 \leq$ NSEPT ≤ 2
NBODY		Index controlling body upwash: NBODY = 0 Body upwash included in wing and tail interference calculation. $<$ 0 Body upwash not included.
NAFT		Index controlling presence of afterbody behind rear lifting surface. NAFT = 0 No afterbody. = 1 Afterbody included.
Item 2		
TITLE		Any alphabetical or numerical identification information.
Item 3		
EM	M	Mach number, $M > 1.0$.
REFS	S	Reference area.
REFL	ℓ	Reference length.
XM	x_m	x-location of center of moments
THETAN	θ_N	Nose semiapex angle, degrees.
DXOUT		x-increment in table of free vortex trajectories. DXOUT $\leq$ RAVGW , typically.
DXI		Maximum integration interval for vortex trajectory calculation. DXI $\leq$ DXOUT, typically.
Item 4		
XBDY	x	x-stations at which body coordinates are defined.
RBDY	r	Body radius at above x-station.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
Item 5		
RAVGW	r_{avg_w}	Average body radius at wing.
YSEPW	y_{sep_w}	Spanwise location of second leading-edge separation vortex. If NSEPW = 1, YSEPW = b/2.
XHLW	x_{HL_w}	x-coordinate of wing hinge line at wing-body juncture.
ZHLW	z_{HL_w}	z-coordinate of wing hinge line at wing-body juncture.
XWCP	x_{CP_w}	x-coordinate of alternate wing center-of-pressure location.
Item 6		Item 6 is made up of NREGNW namelist decks.
$INPUT		Namelist identification.
PER		Location of wing control point in fraction of panel chord. Typically, PER = 0.95.
ROWS		Number of chordwise divisions into which wing region is divided. $1 \leq$ ROWS ≤ 10
COLS		Number of spanwise divisions into which wing region is divided. $1 \leq$ COLS ≤ 20

NOTE: Sum of ROWS × COLS over all wing regions must not exceed 100 and total number of columns on wing semispan must not exceed 20.

ROOTLE	x_{LE_w}	x-coordinate of leading-edge of root chord of wing region now under consideration.
ROOTTE	x_{TE_w}	x-coordinate of trailing edge of root chord. $x_{TE} > x_{LE}$
ROOTY		y-coordinate of root chord. Typically, ROOTY = RAVGW in first wing region.
TIPLE	$x_{LE_{tip}}$	x-coordinate of leading edge of tip chord.
TIPTE	$x_{TE_{tip}}$	x-coordinate of trailing edge of tip chord. $x_{TE} \geq x_{LE}$
TIPY		y-coordinate of tip chord of this wing region. If NREGNW = 1, TIPY = b/2.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
TCROOT	$t/c\|_{root}$	Root chord thickness ratio. May be equal to zero.
TCTIP	$t/c\|_{tip}$	Tip chord thickness ratio. May be equal to zero.
SECT		Specification of airfoil section. SECT = 1 Parabolic arc. = 2 Double wedge. = 3 30-70 hexagon. = 4 Wedge. = 5 NACA 000X = 6 NACA 6400X = 7 NACA 6500X = 8 RAE 101
$END		End of namelist.
Item 7		
ALPHAW	α_w	Local angle of wing mean surface due to camber and twist, radians.
Item 8		
RAVGT	r_{avg_e}	Average body radius at tail.
YSEPT	y_{sep_e}	Spanwise location of second leading-edge separation vortex. If NSEPT = 1, YSEPT = b/2.
XHLT	x_{HL_e}	x-coordinate of tail hinge line at wing-tail juncture.
ZHLT	z_{HL_e}	z-coordinate of tail hinge line at wing-tail juncture.
XTCP	x_{CP_e}	x-coordinate of alternate tail center-of-pressure location.
Item 9		Item 9 is made up of NREGNT namelist decks.
$INPUT		Namelist identification.
PER		Location of tail control point in fraction of panel chord. Typically, PER = 0.95.

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
ROWS COLS ROOTLE ROOTTE ROOTY TIPLE TIPTE TIPY TCROOT TCTIP SECT $END		Corresponding tail variables having same limits and restrictions as those on wing.
Item 10		
ALPHAT	α_e	Local angle of tail mean surface due to camber and twist, radians.
Item 11		
NDEX		Index controlling next case of input. NDEX = 1 Execute program using variables on this card. = 0 Ignore this card and return to beginning for new case.
ALPHAD	α	Angle of attack of configuration, degrees.
ALPIW	δ_w	Incidence angle of wing relative to body axis, degrees.
AKVLW1	$K^*_{v_{LE_w}}$	Fraction of leading-edge suction converted to lift in inboard wing region. ($0 \leq K^*_{v_{LE}} \leq 1.0$)
AKVLW2	$K^*_{v_{LE_w}}$	Fraction of leading-edge suction converted to lift in outboard wing region. ($0 \leq K^*_{v_{LE}} \leq 1.0$)
AKVSW	$K^*_{v_{SE_w}}$	Fraction of side-edge suction converted to lift on wing. ($0 \leq K^*_{v_{SE}} \leq 1.0$)
ALPIT	δ_e	Incidence angle of tail relative to body axis, degrees.
AKVLT1	$K^*_{v_{LE_e}}$	Fraction of leading-edge suction converted to lift in inboard tail region. ($0 \leq K^*_{v_{LE}} \leq 1.0$)

PROGRAM NOTATION	ALGEBRAIC NOTATION	DEFINITION
AKVLT2	$K^*_{v_{LE_e}}$	Fraction of leading-edge suction converted to lift in outboard tail region. $(0 \leq K^*_{v_{LE}} \leq 1.0)$
AKVST	$K^*_{v_{SE_e}}$	Fraction of side-edge suction converted to lift on tail. $(0 \leq K^*_{v_{SE}} \leq 1.0)$
WLIMIT	$(w/V)_{max}$	Limit on vortex-induced velocities at wing and tail control points. $(0 \leq \text{WLIMIT} \leq 1.0)$

Input preparation.- A discussion of the input variables is presented in this section as an aid in the preparation of the input data deck. If a configuration has one set of lifting surfaces, this is denoted the wing regardless of its axial location on the body and data describing this lifting surface are input as wing data. If there are two lifting surfaces, the set nearest the nose is the wing and the aft set is the tail. For example, in a canard-body-wing configuration, the canard data are input as wing data and the wing data are input as tail data. In the following discussion, necessary geometric relations are illustrated in figure 2.

Item number 1 of the input data deck (fig. 1) is a card containing indices specifying particular program options. NHEAD indicates the number of identification cards following in Item 2. NTBL is the number of entries in the table describing the body shape. NPRINT is an index which determines the quantity of output obtained from the program. Typically, this number is zero, but varying degrees of additional output can be obtained by increasing this index from 1 to 4. An abbreviated output summary is obtained by setting NPRINT = -1 or -2. MPRINT is a special index controlling the quantity of output obtained during vortex trajectory calculations. Typically, this index is zero, but if it is greater than zero, the trajectory coordinates are printed as they are computed. This additional trajectory information is useful only if the program fails to compute a trajectory for some particular case, and the additional output may give some clue as to why the calculation failed. This option should only be used after a computational problem has been discovered.

The index NOSEV specifies whether there is (NOSEV = 1) or is not (NOSEV = 0) a symmetrical pair of vortices shed from the nose of the configuration. If the nose angle (THETAN) is less than or equal to four degrees, NOSEV is automatically set equal to zero.

The following three indices are associated with the wing. NREGNW is the number of spanwise regions describing the wing. This number is typically equal to zero unless there are breaks in sweep or it is desired that the panel spacing be different in different wing regions. NALPW specifies if wing camberline slopes are nonzero and must be input (NALPW = 1) or are zero and need not be input (NALPW = 0). NSEPW specifies the number of leading-edge separation vortices shed from the wing. This index must be either 1 or 2, and if NSEPW = 2, a special value of YSEPW is required in Item 5.

The provision for multiple separation vortices is included to handle wings with breaks in leading-edge sweep. It has been observed that leading-edge separation vortices are shed from the wing regions inboard and outboard of the break. A maximum of two leading-edge separation vortices are allowed.

The next three indices in Item 1, NREGNT, NALPT, and NSEPT are tail indices analogous to the preceeding wing indices and subject to exactly the same restrictions. If no tail is present, all three must be zero.

The next index in Item 1, NBODY, determines whether the upwash field around the body is included (NBODY = 0) in the wing-and tail-interference calculations or not included (NBODY < 0). This index is used to determine the magnitude of the body-interference effect and generally should be equal to zero.

The last index, NAFT, is used to specify the presence of an afterbody behind the aft lifting surface. If there is an afterbody, NAFT = 1, and if not, NAFT = 0.

Item 2 is a group of NHEAD cards containing identification information which is printed on the first page of output. The information on the last card of this group is retained and printed as a heading card at various points in the output.

Item 3 is a single card containing EM, the free-stream Mach number (greater than 1); REFS and REFL, the reference area and reference length, respectively; XM, the x-coordinate of the center of moments; and THETAN, the nose semiapex angle in degrees. The final two variables are associated with the free vortex trajectory calculations. DXOUT is the approximate increment in x at which trajectory coordinates are stored for use in induced velocity calculations. A lower limit for this variable is about 0.5 percent of the overall length of the body because of storage limitations Typically, a reasonable value for DXOUT is about one half the maximum radius of the body. DXI is the initial integration interval for the trajectory calculations. The integration package will cut the interval in half if necessary for reasonable accuracy, and this halving process can occur ten times before the program automatically terminates execution with an appropriate message. If the input value of DXI is too large, the program will stop because of unacceptable accuracy, and if DXI is too small, the running time will become large. Experience has indicated a value of DXI between 2 and 5 percent of the body length will work for most cases. Under rare circumstances when two vortices get very close together or when a vortex gets very near the wing or body, a smaller value of DXI may be required. DXI should not be larger than DXOUT.

Item 4 is a group of NTBL cards describing the body shape. Each card contains an x-station, XBDY, and the corresponding body radius, RBDY. The cards should be in ascending order in x and there should be less than 75 cards in this item in a typical run. The program internally sets up its own table of coordinates which is stored in the XBDY and RBDY arrays and is limited to 100 entries. A good rule of thumb to follow in inputting Item 4 is the following.

$$\text{NTBL} < (96 - \text{Total number of spanwise columns on wing and tail})$$

Some care is required when describing the body shape via XBDY and RBDY. Linear interpolation is used throughout; therefore, if the body shape is changing rapidly, more points are required. There should be a minimum of five entries in the nose region ahead of the wing and there must be entries at x-stations identically equal to XWLE and XWTE, and XTLE and XTTE if a tail is present. The last entry in the table must be greater than XWTE or XTTE, whichever is greater, by an amount not less than DXI. If the body is made up of a nose section followed by a cylindrical afterbody, there should be two points on the cylinder very close together near the beginning of the cylinder. Points on a cylinder can be spaced large distances apart, but if the cylinder is followed by a section with changing radius, the last two points on the cylinder should be close together.

Item 5 contains geometric information for the wing. RAVGW is the average body radius in the vicinity of the wing. YSEPW is the y-station at which the wing is assumed broken for purposes of having two leading-edge separation vortices. If NSEPW = 1, then YSEPW must be equal to the wing semispan. If NSEPW = 2, YSEPW must have some value greater than RAVGW and less than the wing semispan. The chosen value should coincide with one of the breaks in the panel layout. It is advised that there be at least three columns of panels on either side of YSEPW to achieve some reasonable accuracy in the separation vortex strength and position calculation. The next two variables, XHLW and ZHLW, are the coordinates of the wing hinge line at the wing-body juncture. The last variable, XWCP, is an experimental center-of-pressure location for the wing potential lift which may be used if desired. If this is not to be used, XWCP must be identically zero.

Item 6 is a namelist (INPUT) describing the geometry of the wing and the panel arrangement. There must be one complete namelist deck for each region of the wing (NREGNW) and they must be in order from wing root to wing tip. The first variable, PER, is the chordwise location of the control point of each constant pressure panel presented as a fraction of the local panel chord. This number is typically 0.95 for supersonic flow. The next two variables, ROWS and COLS, are the number of chordwise and spanwise divisions into which the wing region is divided. Thus, ROWS $\times$ COLS is the number of panels in the wing region under consideration. There are certain limitations on the size of these numbers, as the total number of panels on the entire wing semispan cannot exceed 100 and the total number of spanwise divisions on the entire wing semispan cannot exceed 20.

The following six variables specify the planform geometry of the wing region. ROOTLE is the x-station of the leading edge of the root chord of the region, ROOTTE is the x-station of the trailing edge of the root chord, and ROOTY is the y-station of the root chord. If this is the first wing region, these values correspond to the actual wing root chord where ROOTLE and ROOTTE are the intersections of the leading and trailing edges with the body, and ROOTY has the same value as RAVGW in Item 5. If NREGNW > 1, these values specify the chord of the inboard side of the wing region. In a similar manner, TIPLE and TIPTE are the x-stations of the tip leading and trailing edges, respectively. TIPY is the y-station of the outboard side of the region. If the region under consideration is the last (or only) region of the wing, TIPY must be equal to the semispan of the wing. Note that on wings with multiple regions and continuous leading and trailing edges, the tip chord of one region is identical to the root chord of the adjacent region.

The remaining variables in the namelist are optional. TCROOT and TCTIP are the thickness-to-chord ratios of the root and tip of the wing region. These values may be input as zero or omitted entirely from the deck. Since wing thickness is not a prime consideration in the calculation procedure, it is suggested that these values be omitted. If wing thickness is to be input, a value of SECT must be input to specify the wing section shape. The options available for airfoil sections are noted in the tables of variable definitions. This ends the namelist description of the wing region. All other regions should follow in the input deck immediately; therefore, there will be NREGNW sets of the namelist INPUT decks.

Item 7 is optional input and is included only if the wing is cambered or twisted (NALPW $>$ 0). If such is the case, ALPHAW, the local panel angle, α_ℓ, of each element of the wing must be input. These angles, in radians, are relative to the wing root chord and thus do not include any wing incidence. There are eight values per card and the angles are input from leading edge to trailing edge, from wing root to tip. There are a total of NPANLW values, where NPANLW is the total number of panels on the wing and is computed internally in the program.

Values of ALPHAW are obtained as follows. Consider the sketch in figure 2(b) which shows the cambered and twisted section of the lifting surface at some spanwise station. At point P, corresponding to a control point on the wing mean surface, a tangent to the wing mean surface is constructed, which makes an angle α_ℓ with the wing root chord. The positive sense of α_ℓ is shown in this figure. The input value required is α_ℓ in radians. Near the leading edge of the section shown in figure 2(b), α_ℓ is negative. Item 10 completes the input description of the wing.

If NREGNT = 0; that is, no tail is present, go directly to Item 11. If a tail is present, Item 8 is the next card in the input deck. The variables in Item 8, RAVGT, YSEPT, XHLT, ZHLT, and XTCP are analogous to the wing parameters in Item 5 and subject to the same restrictions and limitations.

Item 9 is a namelist INPUT describing the tail and there must be NREGNT decks of this namelist. The tail variables in this namelist are in direct correspondence with the wing variables described in Item 6.

If NALPT is not zero, the local panel incidence angles in radians of the tail are input in Item 10 in an analogous manner to Item 7 for the wing.

Item 11 is a group of cards, one card for each run, which specifies the variables which are considered changeable for a given geometric configuration. The first entry on the card is the index NDEX which is simply used to control the stacking of additional cases. NDEX = 1 on each card represents a new flow condition. If NDEX = 0, the card is ignored and the program returns to read in a completely new case beginning with Item 1. Thus, a blank card is used to separate calculations with different configurations. When NDEX $\neq$ 0, the next value on the card is the configuration angle of attack in degrees, ALPHAD, taken as the angle between the axis of the body and the free-stream velocity. The second quantity is the incidence angle of the wing root chord in degrees, ALPIW. Its sense is such that a positive incidence is a leading-edge up condition. The next three variables are the K_V^* factors which relate the actual realized vortex lift from the leading and side edges to that which is theoretically available. AKVLW1 is the fraction of leading-edge separation vortex lift which is obtained on the inboard portion of the wing if NSEPW = 2 or on the entire wing if NSEPW = 1. If NSEPW = 2, AKVLW2 is the leading-edge suction factor for the outboard portion of the wing. The values of AKVLW1 and AKVLW2 are obtained from the correlation curve in figure 3. This figure is reproduced from figure 9 of reference 1. The data on which the curve is based are for sharp-edged delta wings. To use this curve for another wing shape, an effective aspect ratio computed as

$$\text{Æ} = \frac{4}{\tan \Lambda_{LE}}$$

should be used. As described in reference 1, care must be taken when using this correlation curve for anything other than sharp-edged wings. A rounded leading edge can cause the suction factor, K_{VLE}^*, to go to zero.

AKVSW is the side-edge suction factor for wings with tips. The side-edge suction lift is usually small compared to the potential lift except for very low-aspect-ratio wings. Comparisons with subsonic rectangular wing data indicate that this factor should be unity, but the magnitude for supersonic flow is not known because of a lack of appropriate data for correlation purposes. Unless other information is available to the user, it is suggested that a value of 1.0 be used for all wings with non-zero tip chords. When there is no wing tip chord, such as on a delta wing, AKVSW should be identically zero.

The next four variables in Item 11, ALPIT, AKVLT1, AKVLT2, and AKVST are the corresponding tail parameters. They fall under the same rules and guidelines set up for the respective wing parameters. If no tail is present, all four values should be set equal to zero.

The final quantity on this card is WLIMIT, the maximum allowable vortex-induced velocity nondimensionalized by the free-stream velocity. The purpose of this variable is to limit the magnitude of the vortex-induced velocities on the wing or tail. In the course of program development, a canard-body-wing configuration exhibited a discontinuous lift curve at some angle of attack around 8°. Close investigation showed that this occurred when the canard trailing vortex passed very near to some wing control points, inducing inordinately large velocities, which resulted in large local angles of attack. The lifting-surface method reacted accordingly and large loading gradients were predicted. This, of course, is an unrealistic situation because a true viscous vortex does not behave as a potential vortex and induce infinite velocities at its center. For this reason a limit was introduced which arbitrarily sets any vortex-induced velocity greater than WLIMIT, equal to WLIMIT.

Generally, WLIMIT should be set equal to 1.0. If , in the process of running the program, unusual variations in the lift or pitching moment with angle of attack occur which can be attributed to unrealistic vortex-induced interference, WLIMIT can be used to limit the magnitude of the large induced velocities causing the problem. A value of WLIMIT = 0.1 has been used in some specific examples to reduce the apparent discontinuity in the predicted lift and moment curves and resulted in good agreement with experiment. This discussion is not meant to suggest that an arbitrary velocity limit will cure the problems with the near flow fields of potential vortices. It is simply included to note that a simple, approximate fix is available. If WLIMIT = 0.0, the effect of the free vortices on the lifting surfaces is completely eliminated.

The above discussion includes all the input required for a typical run. The sample cases in the following section cover most of the options available in the program.

Sample cases.- Listings of the input data decks for three sample cases are presented in figure 4 and sketches of the configurations chosen for these sample cases are shown in figures 5 and 6. These configurations were used in the tests of references 14 and 15, respectively.

Sample case 1 is the canard-body-wing combination shown in figure 5 at a Mach number of 2.01. No optional output is requested and the canard and wing are each defined by a single region. A complete range of angles of attack for one canard deflection angle is specified. This input deck, shown in figure 4(a), requires approximately 300 seconds running time on the IBM 360/67.

Sample case 2, shown in figure 4(b), is the same configuration with the canard removed. No optional output is requested. For demonstration purposes, the wing is specified by two regions. The inboard and outboard regions are given the same panel spacing as that used in sample case 1. This is not normally the situation in laying out this type of lifting surface in multiple regions. Generally, some change in spacing between the regions is necessary. Of course, a break in the sweep angle of the leading or trailing edge dictates the use of multiple regions and the panel spacing may or may not be changed.

Sample case 3 is the wing-body-tail combination shown in figure 6. Minimum output is requested and the wing and tail surfaces are each defined by a single region. A complete range in angle of attack for three tail deflection angles is specified. This input deck, shown in figure 4(c), requires approximately 540 seconds on the IBM 360/67.

Description of Output

The output produced by the SUPSON computer program for sample case 1 is shown in figure 7. The first page of output from the program, figure 7(a), is a tabulation of most of the input data in Items 1, 2, and 3 of figure 1. The next page of output, figure 7(b), is a listing of the namelist INPUT which describes the geometry of the wing. If the wing is divided into more than one region, the variables in namelist INPUT describing each region are listed here. The next page of output, figure 7(c), is a similar listing of the namelist INPUT describing the geometry of the tail. The first entry in the namelist distinguishes between the wing (NSF = 1) or the tail (NSF = 2).

The following page of output, figure 7(d), summarizes the geometry of the configuration by component. The first line at the top of the page is the last identification card from the heading in Item 2 of figure 1. The wing quantities which are tabulated are:

OUTPUT NOTATION	ALGEBRAIC NOTATION
XLE	${x_{LE}}_{root}$
XTE	${x_{TE}}_{root}$
XLE(TIP)	${x_{LE}}_{tip}$
XTE(TIP)	${x_{TE}}_{tip}$
CROOT	c_{root}
CTIP	c_{tip}
B/2	$b/2$
RAVG	${r_{avg}}_w$
YSEP	YSEPW (Item 5)
XHL	${x_{HL}}_w$
ZHL	${z_{HL}}_w$
XCP	$\bar{x}_{CP}$ or XWCP (Item 5)

The same quantities are tabulated for the tail surface if one is present. The following quantities are listed for the body.

OUTPUT NOTATION	ALGEBRAIC NOTATION
THETA	θ_N (Item 3)
FINENESS	x_{LE_w}/r_N
R(BASE)	r_N
AVERAGE RADIUS WING	r_{avg_w}
AVERAGE RADIUS TAIL	r_{avg_e}
CENTER OF MOMENTS X	x_m
CENTER OF MOMENTS Z	z_m
DXI	Δx } Item 3
DXOUT	- } Item 3
X	x
R	r
S	s_w or s_e
DR/DX	dr/dx

This concludes the general geometric description of the configuration. This information is output once at the beginning of each case. The following output is dependent on the information input on Item 11; that is, the angle of attack, incidence angles, K_V^* factors, and induced velocity limit.

Figure 7(e) is the first page of output for each run within the series of runs making up sample case 1. The first line summarizes the information input in Item 11 as follows:

OUTPUT NOTATION	ALGEBRAIC NOTATION
ALPHA	α
M	M
INCIDENCE WING	δ_w

OUTPUT NOTATION	ALGEBRAIC NOTATION
INCIDENCE TAIL	δ_e
WING KVLE*	$K^*_{V_{LE_W}}$
WING KVSE*	$K^*_{V_{SE_W}}$
TAIL KVLE*	$K^*_{V_{LE_e}}$
TAIL KVSE*	$K^*_{V_{SE_e}}$
W/V LIMIT	$(w/V)_{max}$

The next block of output on this page is a summary of the strength and position of the right-hand vortex (if present) of the symmetrical pair of vortices shed from the nose of the body.

OUTPUT NOTATION	ALGEBRAIC NOTATION
GAM/2*PI*V*RB	$\frac{\Gamma_B}{2\pi V r_N}$
Y/RB	y_B/r_N
Z/RB	z_B/r_N
XS/RB	x_s/r_N
GAM/2*PI*V	$\frac{\Gamma_B}{2\pi V}$
Y	y_B
Z	z_B

The following block of output is the induced Beskin upwash at the wing control points (x,y,z) due to the presence of the body. The induced velocities, V/V(INF) and W/V(INF), expressed as a fraction of the free-stream velocity, are positive in the positive y and z directions, respectively. The next block of data are the induced velocities induced at the same wing control points by the vortex pair shed from the nose of the body. These velocities have the same positive sense as the body-induced upwash above. The final block of data in figure 7(e) is the total induced velocity at each control point.

The next page of output, figure 7(f), contains the results from the lifting-surface calculations for the wing. The first several lines are reiteration of some input quantities. The remainder of the results are calculated quantities associated with the potential forces and moments on the wing. The potential span-loading distribution is followed by the spanwise position and strength of the trailing vortex. The above output in figure 7(f) is printed in subroutine FORCE. The definitions of this output are as follows.

OUTPUT NOTATION	ALGEBRAIC NOTATION
CNP	$C_{N_{W(B),p}}$
DCN/DALPHA	$dC_N/d\alpha$
CMP	$C_{m_{W(B),p}}$
DCM/DCN	dC_m/dC_N
XAC	x_{ac}
XCP	x_{CP}
YCP	y_{CP}
B	b
CCN/2B	$\frac{cc_n}{2b}$
GAM/V2PI	$\frac{\Gamma_t}{2\pi V}$

The next block of output on this page is the spanwise distribution of the leading-edge suction. If the wing has a nonzero tip chord, a second block of output containing the distribution of side-edge suction is printed. This latter output is printed in subroutine EDGFRC and is defined as follows.

OUTPUT NOTATION	ALGEBRAIC NOTATION	
Y/(B/2)	$\frac{y}{b/2}$	
CCS/2B	$\frac{cc_s}{2b}$	
EPS(DEG)	ϵ	
CCX/2B	$\left(\frac{cc_x}{2b}\right)_{LE}$	
CCY/2B	$\left(\frac{cc_y}{2b}\right)_{LE}$	
CYC/2B	$\left(\frac{cc_y}{2b}\right)_{SE}$	(This variable does not appear in the sample case because $c_{tip} = 0.$)

Figure 7(g) is headed by a summary of the strengths and positions of the vortices shed from the configuration ahead of the wing trailing edge. The pairs of vortices are listed in the following order. Vortex 1 is the right-side body vortex shed from the nose. Vortex 2 is the trailing vortex shed from the wing. Vortex 3 is the leading-edge separation vortex shed from the wing. If more than one separation vortex is requested, vortex 3 is the vortex associated with the inboard region and vortex 4 is shed from the outboard region. If a vortex is missing for any reason, all following vortices are moved up in the table. For example, if no vortices are shed by the nose, vortex 1 becomes the trailing vortex shed by the wing, and so on. The remainder of figure 7(g) indicates the induced velocities at the tail control points. These velocities are analogous to the induced velocities on the wing shown in figure 7(e).

Figure 7(h) contains calculated results for the tail surface. All the quantities on this page are analogous to those described for the wing in figure 7(f). The last entry on this figure is a summary of the strengths and positions of all the vortices in the field just aft of the tail trailing edge. The first group of vortices are the same as described in connection with figure 7(g). The second group of vortices are defined as follows. Vortex 4 is the trailing vortex corresponding to the potential lift on the tail. Vortex 5 is the leading-edge separation vortex shed from the tail. If multiple vortices are shed from the tail leading edge, this vortex would be shed from the inboard tail region and vortex 6 would be shed from the outboard tail region.

The next page of output, figure 7(i), is a summary page of the force coefficients, pitching-moment coefficients, and centers of pressure on each component of the configuration and of the total configuration. The coefficients for the individual components are described in Table I. The total configuration variables are defined as follows.

OUTPUT NOTATION	ALGEBRAIC NOTATION
CN	$C_N = \frac{N}{qS}$
CM	$C_m = \frac{M}{qS\ell}$
XCP	$\bar{x}_{CP} = x_m - \frac{C_m}{C_N}\ell$
CL	$C_L = \frac{L}{qS}$
CDI	$C_{D_i} = C_L \tan\alpha$
CA	$C_A = \frac{A}{qS}$
CDI/CL**2	C_{D_i}/C_L^2

The last page of output for this run, figure 7(j), contains a summary of the trajectories of the shed vortices. At the top of the page the vortices are identified and their strengths listed. This is followed by blocks of output, one block for each x station, describing the local crossflow geometry of the configuration and the position of the right-side vortices. Each block of results is separated by approximately DXOUT. Notice that the trajectory calculation starts at the wing leading edge with a pair of body vortices. As the calculation moves downstream, other vortices are shed and added to the calculation. The trajectory calculation is carried downstream to a point aft of the tail trailing edge. The variables in each block are defined as follows.

OUTPUT NOTATION	ALGEBRAIC NOTATION
X	x
DX	Δx
A	a
S	s_w or s_e
RO	r_o
DA/DX	da/dx
SIGMA(REAL)	y
SIGMA(IMAG)	z

This completes the output for one card in Item 11 of the input deck. Additional runs will repeat the output of figures 7(e) through (j). The above set of output obtained with NPRINT = 0 is a considerable amount of output for production runs; therefore, an optional set of output can be obtained by setting NPRINT = -2. In this case, the complete output consists of figures 7(a), (b), (c), (d), and (i) with some shed vortex positions and strengths added.

Some extra output over and above that shown in figure 7 can be obtained when NPRINT > 0. This additional output is useful only for diagnostic purposes and is not described herein. This output is labeled and the user should have no trouble interpreting the results.

Program Listing

The SUPSON computer program consists of the main program, nine function subprograms, and twenty-five subroutine subprograms. Each source deck is identified in columns 73 through 80 by a four-character identification and a three-digit number sequencing the cards within that deck.

The program listing is given on the following pages. The table below will act as a table of contents for the listing.

```
C       SUPERSONIC PREDICTION METHOD
C                                                       M.R.MENDENHALL  260/C
      COMPLEX SP,VP,EYE,SIGMA
      DIMENSION AW(10000),AT(10000)
      DIMENSION XP(100),SP(100),VP(100),WB(100),WV(100)
      DIMENSION XYW(6,100),XCPW(100),YCPW(100),XBARW(100),AREAW(100),
     1          XYT(6,100),XCPT(100),YCPT(100),XBART(100),AREAT(100),
     2          SLEW(100),STEW(100),DZDXW(100),NRCW(2, 20),XPNLFW(100),
     3          SLET(100),STET(100),DZDXT(100),NRCT(2, 20),XPNLFT(100),
     4          XPNLBW(100),IPW(100),YSPANW( 20),PNLCW(100),ALPHAW(100),
     5          XPNLBT(100),IPT(100),YSPANT( 20),PNLCT(100),ALPHAT(100),
     6          XSPANW(20),XSPANT(20)
      DIMENSION GAMN(10),XGAMN(10),YGAMN(10),ZGAMN(10)
      DIMENSION TITLE(20),ALPHA(100),ARAY(6),HEAD(9)
      DIMENSION UTW(100),UTT(100),SIGMA(2)
C
      COMMON /BODY/ NTBL,MTBL,XBDY(100),RBDY(100),SBDY(100),DRBDY(100)
      COMMON /VTABLE/ NTRY,XV(200),YV(4,200),ZV(4,200),NVP(200),
     1                ARRAY(5,200)
      COMMON /RESLT2/ YTR,GAMT,CLP,CMP,XCP
      COMMON /RESLT1/ CNV1,CNV2,CNVS,Y1(2),GAM1(2),COPLE,COPTIP
      COMMON /B/ MDUMY,M
      COMMON /VORTEX/ NV,NV2,GAM(4),YG(4),ZG(4)
      COMMON /TRAJ1/ XI,XF,DXI,TITL,NPRNT
      COMMON /TRAJ2/ XP,SP,VP,NEXT
      COMMON /FLOW/ ALPD,CSALP,EYE
      COMMON /PARAM/ DXOUT,XPRNT,MPRNT,XFINAL
C
      DATA HEAD /4HWING,4HTAIL,4HW(B),4HB(W),4HT(B),4HB(T),4HNOSE,
     1           4HBODY,4H    /
C
      NAMELIST /INPUTM/ NHEAD,NTBL,NPRINT,MPRINT,NOSEV,NREGNW,NALPW,
     1                  NSEPW,NREGNT,NALPT,NSEPT,NBODY,NAFT
C
    1 FORMAT(20A4)
  700 FORMAT (16I5)
  701 FORMAT (1H1)
  702 FORMAT(10X,20A4)
  703 FORMAT (1H0)
  705 FORMAT (8F10.5)
  706 FORMAT (10X,10F10.5)
  707 FORMAT (I2,F6.4,9F8.4)
  708 FORMAT (I10,4F10.5)
  709 FORMAT (//5X25HTABLE OF BODY COORDINATES/15X1HX,9X1HR,9X1HS,
     1           9X5HDR/DX)
  710 FORMAT (18X,10H VORTEX    ,4X,5(1PE12.3))
  711 FORMAT (1H0,5X47HSUMMARY OF VORTEX STRENGTHS AND POSITION AT X = ,
     1           F8.3)
  712 FORMAT (/10X6HVORTEX,3X12HGAMMA/2*PI*V,6X1HY,9X1HZ,
     1           4(/12X,I2,6XF10.6,2F10.3)/)
  713 FORMAT (5X49HSUMMARY OF FORCE AND PITCHING MOMENT COEFFICIENTS //
     1           37X,2HCN,10X2HCM,10X3HXCP,9X2HCL,10X3HCDI,9X2HCA)
  714 FORMAT (/10X,A4,3H... )
  715 FORMAT (14X,A4,10H POTENTIAL ,4X,6(1PE12.3))
  716 FORMAT (14X,A4,10H VORTEX,LE ,4X,6(1PE12.3))
  717 FORMAT (14X,A4,10H VORTEX,SE ,4X,6(1PE12.3))
  718 FORMAT (/10X,12HAFTERBODY... ,10X, 6(1PE12.3))
  719 FORMAT (//10X,22HTOTAL CONFIGURATION... , 6(1PE12.3))
  720 FORMAT (1H1,5X,35HSUMMARY OF FREE VORTEX TRAJECTORIES //)
  721 FORMAT (//5X1HX,8X2HDX,11X1HA,9X1HS,8X2HR0,7X5HDA/DX)
  722 FORMAT (F9.4,F11.7,4F10.4)
  723 FORMAT (/1X6HVORTEX,1X11HSIGMA(REAL),3X6H(IMAG))
  724 FORMAT (I4,2X,4(1PE12.3,E11.3))
  725 FORMAT (/6X6HVORTEX,3X12HGAMMA/2*PI*V)
  726 FORMAT (I10,5X,F10.6)
  727 FORMAT (10X6HVORTEX,I2,28H ---  BODY VORTEX FROM NOSE )
  728 FORMAT (10X6HVORTEX,I2,28H ---  WING TRAILING VORTEX  )
  729 FORMAT (10X6HVORTEX,I2,28H ---  WING SEPARATION VORTEX)
  732 FORMAT (12X,I2,6XF10.6,2F10.3)
  734 FORMAT(//5X8HBODY ... ,6X27H---------- NOSE ----------,5X14HAVER
     1AGE RADIUS,4X17HCENTER OF MOMENTS/20X5HTHETA,3X8HFINENESS,3X7HR(BA
     2SE),5X4HWING,6X4HTAIL,7X1HX,9X1HZ,9X3HDXI,5X5HDXOUT/15X9F10.3)
  735 FORMAT (//5X30HNOSE VORTICES -  GAM/2*PI*V*RB,3X4HY/RB,6X4HZ/RB,
     1           5X5HXS/RB,/22XF10.4,3F10.3)
  737 FORMAT (//30X15H-- INCIDENCE -- ,5X15H---- WING ----- ,
     1           5X15H---- TAIL ----- / 10X5HALPHA,8X1HM,7X4HWING,6X4HTAIL,
     2           2(5X5HKVLE*,5X5HKVSE*),3X9HW/V LIMIT /5X2F10.2,7F10.3)
  742 FORMAT (/35X12HCDI/CL**2 = ,1PE12.3)
  745 FORMAT (///5X46HVELOCITY INDUCED AT SPECIFIED FIELD POINTS ON ,A4,
     1           9H (BESKIN)  )
  746 FORMAT (/ 10X1HX,9X1HY,9X1HZ,11X8HV/V(INF),4X8HW/V(INF))
  747 FORMAT (I4,3F10.3,5X,2(1PE12.4))
  748 FORMAT (//5X64H*** NOTE - VORTEX INDUCED VELOCITIES ARE NEGLECTED
     1, W/V LIMIT =,F10.4)
  749 FORMAT (///5X52HTOTAL INDUCED VELOCITY AT SPECIFIED FIELD POINTS O
     1N , A4)
  751 FORMAT (45XF10.3,10X,F10.3)
  752 FORMAT (/10X14HCONTROL POINTS,15X6HCAMBER/2X5HPANEL,
     1           5X1HX,9X1HY,7X4HAREA,6X5HALPHA)
  753 FORMAT (I5,4F10.5)
  755 FORMAT (/5X,A4)
  756 FORMAT (/15X1HM,9X4HREFS,6X4HREFL,6X3HXCM,5X5HDXOUT,5X3HDXI)
  757 FORMAT (/15X4HRAVG,6X4HYSEP,6X3HXHL,7X3HZHL,7X3HXCP)
  758 FORMAT (15X3HXLE,7X3HXTE,5X8HXLE(TIP),2X8HXTE(TIP),4X5HCROOT,
     1           5X4HCTIP,7X3HB/2 / 10X,10F10.5)
  798 FORMAT (//10X 63H*** NOSE VORTEX POSITION ERROR, ELIMINATED FROM C
     1ALCULATION ***   //)
C
      EYE=CMPLX(0.0,1.0)
      PI=3.1415926
      RAD=180./3.1415926
      ZM=0.0
  200 READ (5,700) NHEAD,NTBL,NPRINT,MPRINT,NOSEV,NREGNW,NALPW,NSEPW,
     1                               NREGNT,NALPT,NSEPT,NBODY,NAFT
      NAFTW=1
      IF (NREGNT.LE.0) NAFTW=NAFT
      NPRNT=NPRINT
      MPRNT=MPRINT
      WRITE (6,701)
      WRITE (6,INPUTM)
      IF (NHEAD.LE.0 .OR. NREGNW.LE.0) GO TO 996
       DO 201 J=1,NHEAD
      READ (5,1) TITLE
  201 WRITE (6,702)   TITLE
      READ (5,705) EM,REFS,REFL,XM,THETAN,DXOUT,DXI
      WRITE (6,756)
      BETA=SQRT(EM*EM-1.0)
      WRITE (6,706) EM,REFS,REFL,XM,DXOUT,DXI,BETA
      IF (EM.LE.0.0) STOP
      EMU=ARSIN(1.0/EM)*RAD
C
C      INITIALIZE
C
      NPANLT=0
      NR=20
      NMAT=10000
      NPL=100
      NSURF=1
      NSPNT=0
      CRNT=0.0
      CTNT=0.0
      NOSE=NOSEV
      DO 203 J=1,NTBL
  203 READ (5,705) XBDY(J),RBDY(J)
      DO 202 J=1,NPL
      XP(J)=0.0
      ALPHA(J)=0.0
      ALPHAW(J)=0.0
      ALPHAT(J)=0.0
      SLEW(J)=0.0
      STEW(J)=0.0
      SLET(J)=0.0
      STET(J)=0.0
      UTW(J)=0.0
  202 UTT(J)=0.0
      CNNP  =0.0
      CNNV  =0.0
      CNWBP =0.0
      CNWBVS=0.0
      CNWBVL=0.0
      CNBWP =0.0
      CNBWV =0.0
      CNBWVS=0.0
      CNA   =0.0
```

```
      CNTBP  =0.0
      CNTBVL=0.0
      CNTBVS=0.0
      CNBTP  =0.0
      CNBTV  =0.0
      CNBTVS=0.0
      CMWP   =0.0
      CMWV   =0.0
      CMWBP  =0.0
      CMWBVL=0.0
      CMWBVS=0.0
      CMBWP  =0.0
      CMBWV  =0.0
      CMBWVS=0.0
      CMA    =0.0
      CMTBP  =0.0
      CMTBVL=0.0
      CMTBVS=0.0
      CMBTP  =0.0
      CMBTV  =0.0
      CMBTVS=0.0
      XCPNP  =0.0
      XCPNV  =0.0
      XCPWBP=0.0
      XCPBWP=0.0
      XCPBWV=0.0
      XCPBWS=0.0
      XCPA   =0.0
      XCPTBP=0.0
      XCPBTP=0.0
      XCPBTV=0.0
      XCPBTS=0.0
      CAWBP  =0.0
      CAWBVL=0.0
      CAWBVS=0.0
      CATBP  =0.0
      CATBVL=0.0
      CATBVS=0.0
C
C       COMPUTE WING GEOMETRIC CHARACTERISTICS
C
      READ (5,705)  RAVGW,YSEPW,XHLW,ZHLW,XWCP
      WRITE (6,701)
      CALL SSWING (NSURF,NREGNW,NR,NPL,NMAT,NPRINT,MSPNW,
     1            EM,REFS,REFL,XM,TITLE,TIPYW,SLOPW1,CTIPW,CROOTW,TOLW,
     2            NPANLW,NRCW,AW,XYW,XCPW,YCPW,XBARW,AREAW,SLEW,STEW,
     3            DZDXW,XPNLFW,XPNLBW,XSPANW,YSPANW,PNLCW,IPW,UTW,ARAY)
      SSPANW=TIPYW
      IF (YSEPW.LE.0.0) NSEPW=1
      IF (NSEPW.LT.1 .OR. NSEPW.GT.2) NSEPW=1
      IF (NSEPW.EQ.2 .AND. YSEPW.GE.SSPANW) NSEPW=1
      IF (NSEPW.EQ.1) YSEPW=SSPANW
      CRWW=CROOTW
      CTWW=CTIPW
      XLEW=ARAY(1)
      XTEW=ARAY(2)
      XLEWTP=ARAY(4)
      XTEWTP=ARAY(5)
      XLET=XBDY(NTBL)
      XTET=XLET+0.001
      IF (NALPW.GT.0) READ (5,705) (ALPHAW(I),I=1,NPANLW)
      IF (NPRINT.LE.1) GO TO 204
      WRITE (6,752)
      WRITE (6,753) (I,XCPW(I),YCPW(I),AREAW(I),ALPHAW(I),I=1,NPANLW)
  204 IF (NREGNT.LT.1) GO TO 50
C
C       COMPUTE TAIL GEOMETRIC CHARACTERISTICS
C
      NSURF=2
      READ (5,705)  RAVGT,YSEPT,XHLT,ZHLT,XTCP
      WRITE (6,701)
      CALL SSWING (NSURF,NREGNT,NR,NPL,NMAT,NPRINT,MSPNT,
     1            EM,REFS,REFL,XM,TITLE,TIPYT,SLOPT1,CTIPT,CROOTT,TOLT,
     2            NPANLT,NRCT,AT,XYT,XCPT,YCPT,XBART,AREAT,SLET,STET,
     3            DZDXT,XPNLFT,XPNLBT,XSPANT,YSPANT,PNLCT,IPT,UTT,ARAY)
      SSPANT=TIPYT
      IF (YSEPT.LE.0.0) NSEPT=1
      IF (NSEPT.LT.1 .OR. NSEPT.GT.2) NSEPT=1
      IF (NSEPT.EQ.2 .AND. YSEPT.GE.SSPANT) NSEPT=1
      IF (NSEPT.EQ.1) YSEPT=SSPANT
      CRWT=CROOTT
      CTWT=CTIPT
      XLET=ARAY(1)
      XTET=ARAY(2)
      XLETTP=ARAY(4)
      XTETTP=ARAY(5)
      IF (NALPT.GT.0) READ (5,705) (ALPHAT(I),I=1,NPANLT)
      IF (NPRINT.LE.1) GO TO 50
      WRITE (6,752)
      WRITE (6,753) (I,XCPT(I),YCPT(I),AREAT(I),ALPHAT(I),I=1,NPANLT)
   50 CONTINUE
C
C       SET UP TABLE OF BODY COORDINATES AND SLOPES
C
      CALL TABLE (XLEW,XTEW,CRWW,CTWW,SSPANW,XSPANW,YSPANW,MSPNW,SLOPW1,
     1            XLET,XTET,CRWT,CTWT,SSPANT,XSPANT,YSPANT,MSPNT,SLOPT1)
C
      CALL SHAPE (XLEW,RBASE,DUM,DUM)
      FINE=XLEW/(2.0*RBASE)
C
C       OUTPUT BODY CHARACTERISTICS
C
      WRITE (6,701)
      WRITE (6,1) TITLE
      WRITE (6,755) HEAD(1)
      WRITE (6,758) XLEW,XTEW,XLEWTP,XTEWTP,CRWW,CTWW,SSPANW
      WRITE (6,757)
      WRITE (6,706) RAVGW,YSEPW,XHLW,ZHLW,XWCP
      IF (NREGNT.LT.1) GO TO 205
      WRITE (6,755) HEAD(2)
      WRITE (6,758) XLET,XTET,XLETTP,XTETTP,CRWT,CTWT,SSPANT
      WRITE (6,757)
      WRITE (6,706) RAVGT,YSEPT,XHLT,ZHLT,XTCP
  205 CONTINUE
      WRITE(6,754)THETAN,FINE,RBASE,RAVGW,RAVGT,XM,ZM,DXI,DXOUT
      WRITE (6,709)
      DO 14 J=1,NTBL
   14 WRITE (6,708) J,XBDY(J),RBDY(J),SBDY(J),DRBDY(J)
C
C       BEGINNING OF CALCULATIONS DEPENDING ON ANGLE OF ATTACK
C
   19 READ (5,707) NDEX,ALPHAD,ALPIW,AKVLW1,AKVLW2,AKVSW,
     1                        ALPIT,AKVLT1,AKVLT2,AKVST,WLIMIT
   98 IF (NDEX.LE.0) GO TO 200
      IF (ABS(ALPHAD + ALPIW).LE.0.0) GO TO 19
      WRITE (6,701)
      WRITE (6,737) ALPHAD,EM,ALPIW,ALPIT,AKVLW1,AKVSW,AKVLT1,AKVST,
     1             WLIMIT
      IF (NSEPW.GT.1 .OR. NSEPT.GT.1) WRITE (6,751) AKVLW2,AKVLT2
      IF (WLIMIT.LE.0.0) WRITE (6,748) WLIMIT
      NOSEV=NOSE
      NV=0
      ALPW=ALPHAD+ALPIW
      ALPO=ALPHAD/RAD
      ALPI=ALPIW/RAD
      CSALP=COS(ALPO)
      IF (WLIMIT.LT.0.0) WLIMIT=0.0
C
C       COMPUTE NORMAL FORCE AND MOMENT CONTRIBUTION OF NOSE
C
      VLM=0.
      DO 240 J=2,NTBL
      IF (XBDY(J).GT.XLEW) GO TO 241
      K=J
  240 VLM=VLM+(RBDY(J)**2 + RBDY(J)*RBDY(J-1) + RBDY(J-1)**2)
     1        *(XBDY(J)-XBDY(J-1))/3.0
  241 XCPNP=XLEW-VLM/(RBDY(K)**2)
      CNNP=2.0*PI*(RBDY(K)**2)/REFS*SIN(ALPHAD/RAD)/COS(ALPHAD/RAD)
      CMNP=CNNP*(XM-XCPNP)/REFL
      CNNV=0.0
      CMNV=0.0
      XCPNV=0.0
      IF (ABS(ALPHAD).LE.4.0 .OR. THETAN.LE.4.0) NOSEV=0
      IF (NOSEV.LE.0) GO TO 119
C
C       CALCULATE EFFECT OF BODY VORTEX SHED FROM NOSE
```

```
C
      MTBL=1
      XA=XLEW/RBASE
      CALL BDYVTX (ALPHAD,THETAN,FINF,XA,EM,GAMN,NG,XGAMN,YGAMN,
     1             ZGAMN,XSA)
      IF (NG.EQ.0) NOSEV=0
      IF (NOSEV.LE.0) GO TO 119
      WRITE(6,735)GAMN(NG),YGAMN(NG),ZGAMN(NG),XSA
      DO 16 J=1,NG
      GAMN(J)=GAMN(J)*RBASE
      XGAMN(J)=XGAMN(J)*RBASE
      CALL SHAPE (XGAMN(J),RL,S,SD)
      YGAMN(J)=YGAMN(J)*RL
   16 ZGAMN(J)=ZGAMN(J)*RL
      DUM=SQRT(YGAMN(NG)**2 + ZGAMN(NG)**2)
      IF (DUM.GT.RBASE) GO TO 116
      NOSEV=0
      NG=1
      WRITE (6,798)
  116 CONTINUE
      GAMNG=GAMN(NG)
      YGAMNG=YGAMN(NG)
      ZGAMNG=ZGAMN(NG)
  341 NV=NV+1
      NV2=NV+2
      GAM(1)=GAMNG
      YG(1)=YGAMNG
      ZG(1)=ZGAMNG
      WRITE (6,711) XLEW
      WRITE (6,712) (I,GAM(I),YG(I),ZG(I),I=1,NV)
C
C       COMPUTE VORTEX INDUCED FORCES AND MOMENTS ON NOSE
C
      MTBL=1
      CALL CNVNZ (XGAMN,YGAMN,ZGAMN,GAMN,NG,XM,CN,CM)
      CNNV=2.0*PI*CN/REFS
      CMNV=2.0*PI*CM/(REFS*REFL)
      XCPNV=0.0
      IF (NG.GT.1) XCPNV=XM-CMNV*REFL/CNNV
  119 CONTINUE
C
      SNALIW=SIN(ALPIW/RAD)
      CSALIW=COS(ALPIW/RAD)
      H=DXI
      NTRY=0
      MDUMY=0
      MTBL=1
      NEXT=NPANLW
      DO 141 J=1,NEXT
      XP(J)=XCPW(J)
      YDUM=YCPW(J)
      ZDUM=ZHLW-(XP(J)-XHLW)*SNALIW
      SP(J)=CMPLX(YDUM,ZDUM)
      WB(J)=0.0
      WV(J)=0.0
  141 CONTINUE
C
C       COMPUTE BESKIN UPWASH ON WING
C
      TITL=HEAD(1)
      DUM=1.0
      EPS=90.-ATAN2(SLEW(1),DUM)*RAD
      IF (NBODY.LT.0) GO TO 540
      IF (NPRINT.LT.0) GO TO 510
      WRITE (6,745) HEAD(1)
      WRITE (6,746)
  510 CONTINUE
      DO 340 J=1,NEXT
      CALL SHAPE(XP(J),A,DUM,DUM)
      VP(J)=EYE*ALPUW*A*A/(SP(J)*SP(J))
      WB(J)=AIMAG(VP(J))
      IF (NPRINT.LT.0) GO TO 340
      WRITE (6,747) J,XP(J),SP(J),VP(J)
  340 CONTINUE
  540 CONTINUE
      IF (NOSEV.LE.0) GO TO 440
C
C       COMPUTE TRAJECTORY OF NOSE VORTEX PAST WING
C
      IF (EPS.LT.EMU) GO TO 15
C
C       SUPERSONIC LEADING EDGE
C
      NTRY=NTRY+1
      NVP(NTRY)=NV
      ARRAY(1,NTRY)=H
      ARRAY(2,NTRY)=RBASE
      ARRAY(3,NTRY)=S
      ARRAY(4,NTRY)=RL
      ARRAY(5,NTRY)=SD
      XV(NTRY)=XLEW
      YV(1,NTRY)=YG(1)
      ZV(1,NTRY)=ZG(1)
      NTRY=NTRY+1
      XV(NTRY)=XTEW+0.001
      IF (XYW(4,NPANLW).GT.XTEW) XV(NTRY)=XYW(4,NPANLW)+0.001
      YV(1,NTRY)=YG(1)
      ZV(1,NTRY)=ZG(1)-CROOTW*SNALIW
      CALL SHAPE (XV(NTRY),A,S,SD)
      NVP(NTRY)=NV
      ARRAY(1,NTRY)=H
      ARRAY(2,NTRY)=A
      ARRAY(3,NTRY)=S
      ARRAY(4,NTRY)=A
      ARRAY(5,NTRY)=SD
      XF=-99.9
      CALL TRJTRY
      GO TO 17
   15 XI=XLEW
      XF=XTEW+DXI/4.
      IF (XYW(4,NPANLW).GT.XF) XF=XYW(4,NPANLW)+0.001
      CALL TRJTRY
      IF (MDUMY.GT.10) GO TO 953
   17 CONTINUE
      YG(1)=YV(1,NTRY)
C
C       CORRECT VERTICAL POSITION OF VORTEX TO ACCOUNT FOR WING
C         VERTICAL DISPLACEMENT
C
      ZG(1)=ZV(1,NTRY) + ZHLW
C
C       SET UP INDUCED VELOCITY FIELD ON WING
C
      IF (NPRINT.LT.0) GO TO 511
      WRITE (6,749) HEAD(1)
      WRITE (6,746)
  511 CONTINUE
      DO 142 J=1,NEXT
      DUM=CABS(VP(J))
      IF (DUM.GT.WLIMIT) VP(J)=VP(J)*WLIMIT/DUM
      WV(J)=AIMAG(VP(J))
      A=REAL(VP(J))
      IF (NPRINT.LT.0) GO TO 142
      DUM=WV(J) + WB(J)
      WRITE (6,747) J,XP(J),SP(J),A,DUM
  142 CONTINUE
  440 CONTINUE
C
C       COMPUTE LOADS AND FORCE ON WING
C
      DO 31 J=1,NPANLW
   31 ALPHA(J)=ALPHAW(J) + ALPD + ALPI + WB(J) + WV(J)
      NCASE=-1
      NSURF=1
      CDFW=0.
      CALL FORCE (NSURF,NR,NPL,NMAT,NPRINT,NCASE,NREGNW,TITLE,
     1            NPANLW,EM,REFS,REFL,XM,AW,XYW,XCPW,YCPW,XBARW,AREAW,
     2            SLEW,STEW,DZDXW,NRCW,CDFW,ALPHA,UTW,IPW)
      CNWBP=CLP*CSALIW
      IF (XWCP.GT.0.0) XCP=XWCP
      XCPWBP=XCP
      CAWBP=CNWBP*SNALIW/CSALIW
      CMWBP=(XM-XCPWBP)*CNWBP/REFL
      CALL WPANL (NSURF,NR,NPL,NMAT,NPRINT,NPANLW,EM,XYW,XCPW,YCPW,
     1            XBARW,NPRINT,SLEW,STEW,TOLW,XPNLFW,XPNLBW)
```

```
      CALL EDGFRC (NSURF,NR,NPL,NMAT,NPANLW,NREGNW,REFS,
     1             XYW,YCPW,SLEW,STEW,DZDXW,NRCW,YSPANW,PNLCW,
     2             TIPYW,SLOPW1,TOLW,CTIPW,CROOTW,ALPHA,NPRINT,
     3             YSEPW,NSEPW,AKVLW1,AKVLW2,AKVSW)
      DO 120 J=1,NSEPW
      DUM=0.0
      IF (Y1(J).GT.0.0) DUM=1.0/(8.0*PI*Y1(J))
      GAM1(J)=GAM1(J)*REFS*CSALP*DUM
      Y1(J)=Y1(J)+XYW(5,1)
  120 CONTINUE
      CNWBVL=(CNV1*AKVLW1 + CNV2*AKVLW2)*CSALIW
      CNWBVS=CNVS*AKVSW*CSALIW
      XCPWVL=COPLE
      XCPWVS=COPTIP
      CMWBVL=(XM-XCPWVL)*CNWBVL/REFL
      CMWBVS=(XM-XCPWVS)*CNWBVS/REFL
      CAWBVL=CNWBVL*SNALIW/CSALIW
      CAWBVS=CNWBVS*SNALIW/CSALIW
C
C      DETERMINE VERTICAL LOCATION OF SEPARATION VORTICES
C
      CALL ZVTX (ALPW,EPS,BTA2)
      IF (CNWBVL.LT.0.0 .AND. ALPW.GT.0.0) BTA2=-BTA2
      GAM(NV+1)=GAMT
      IF (YTR .GT. SSPANW) YTR=SSPANW
      YG(NV+1)=YTR
      ZG(NV+1)=ZHLW-(XTEW-XHLW)*SNALIW
      NV=NV+1
      Z=CRWW*SIN(BTA2/RAD)
      SUMGAM=0.0
      IF (AKVLW1.LE.0.0  .AND. AKVLW2.LE.0.0) GO TO 520
      DO 245 I=1,NSEPW
      GAM(NV+1)=GAM1(I)
      SUMGAM=SUMGAM+GAM(NV+1)
      YG(NV+1)=Y1(I)
      IF (I.GT.1) Z=(CTWW+(CRWW-CTWW)*YSEPW/SSPANW)*SIN(BTA2/RAD)
      ZG(NV+1)=Z+ZHLW-(XTEW-XHLW)*SNALIW
      SIGMA(I)=CMPLX(YG(NV+1),ZG(NV+1))
      IF (GAM(NV+1) .NE. 0.0) NV=NV+1
  245 CONTINUE
  520 NV2=NV+2
C
C      COMPUTE NORMAL FORCE AND MOMENT ON BODY IN PRESENCE OF WING
C
      TAPER=CTWW/CRWW
      DUM=1.0
      SWPLE=ATAN2(SLEW(1),DUM)*RAD
      SMBW=1.0E+08
      IF (SWPLE.GT.0.0) SMBW=BETA/TAN(SWPLE/RAD)
      ROSW=RAVGW/SSPANW
      CSW=(SSPANW-RAVGW)*(CTWW+CRWW)
      BETAAW=4.0*BETA*((SSPANW-RAVGW)**2)/CSW
      BDCRW=BETA*2.0*RAVGW/CRWW
      PARAM=BETAAW*(1.0+1.0/SMBW)*(1.0+TAPER)
      CALL CH1416 (SSPANW,RAVGW,SWPLE,CTWW,CRWW,EM,PARAM,NAFTW,BDCRW,
     1             XCPBWP,SMBW,TAPER,ROSW,BETAAW)
      XCPBWP=XCPBWP*CRWW+XLEW
      XCPBWV=XCPBWP
      XCPBWS=XCPBWP
      CLAW=CHRTB (BETAAW,SSPANW,RAVGW,CTWW,CRWW,SWPLE)/(BETA*RAD)
      CLAR=CLAW*RAD
      CALL KFACT(ROSW,CKWB,EM,PARAM,CKBW,NAFTW,SMBW,TAPER,BDCRW,CLAR,
     1             BETA,SWPLE,BETAAW,DUM,DUM)
      CNBWP=CKBW*CNWBP/CKWB
      CNBWV=CKBW*CNWBVL/CKWB
      CNBWVS=CKBW*CNWBVS/CKWB
  521 CMBWP=(XM-XCPBWP)*CNBWP/REFL
      CMBWV=(XM-XCPBWV)*CNBWV/REFL
      CMBWVS=(XM-XCPBWS)*CNBWVS/REFL
C
      ALPT=ALPHAD+ALPIT
      ALPI=ALPIT/RAD
      SNALIT=SIN(ALPI)
      CSALIT=COS(ALPI)
C
C      COMPUTE TRAJECTORY OF NOSE VORTEX AND WING VORTICES PAST TAIL
C
      XI=XTEW + .002
      IF (NOSEV.GT.0) XI=XV(NTRY) + .001
      IF (NPRINT.GE.0) WRITE (6,701)
      WRITE (6,711) XI
      WRITE (6,712) (I,GAM(I),YG(I),ZG(I),I=1,NV)
      XINIT=XI
      TITL=HEAD(2)
      IF (NREGNT.LE.0) XTET=XBODY(NTHL)-DXI
      XF=XTET + DXI/4.
      IF (NREGNT.LE.0) GO TO 149
      IF (XYT(4,NPANLT).GT.XF) XF=XYT(4,NPANLT)+0.001
  149 CONTINUE
      IF (XF.LE.XI) GO TO 60
      DXI=H
      NEXT=NPANLT
      IF (NEXT.GT.0) GO TO 148
      CALL TRJTRY
      IF (MDUMY.GT.10) GO TO 53
      GO TO 147
  148 CONTINUE
      DO 145 J=1,NEXT
      XP(J)=XCPT(J)
      YDUM=YCPT(J)
      ZDUM=ZHLT-(XP(J)-XHLT)*SNALIT
      SP(J)=CMPLX(YDUM,ZDUM)
      WB(J)=0.0
      WV(J)=0.0
  145 CONTINUE
C
C      COMPUTE BESKIN UPWASH ON TAIL
C
      DUM=1.0
      EPS=90.-ATAN2(SLET(1),DUM)*RAD
      IF (NBODY.LT.0) GO TO 548
      IF (NPRINT.LT.0) GO TO 512
      WRITE (6,745) HEAD(2)
      WRITE (6,746)
  512 CONTINUE
      DO 348 J=1,NEXT
      CALL SHAPE (XP(J),A,DUM,DUM)
      VP(J)=EYE*ALPD*A*A/(SP(J)*SP(J))
      WB(J)=AIMAG(VP(J))
      IF (NPRINT.LT.0) GO TO 348
      WRITE (6,747) J,XP(J),SP(J),VP(J)
  348 CONTINUE
  548 CONTINUE
      IF (EPS.LT.EMU) GO TO 349
C
C      SUPERSONIC LEADING EDGE
C
      NEXT=0
      XF=XLET
      CALL TRJTRY
      IF (MDUMY.GT.10) GO TO 53
      NTRY=NTRY+1
      XV(NTRY)=XTET+0.001
      IF (XYT(4,NPANLT).GT.XTET) XV(NTRY)=XYT(4,NPANLT)+0.001
      CALL SHAPE (XV(NTRY),A,S,SD)
      NVP(NTRY)=NV
      ARRAY(1,NTRY)=H
      ARRAY(2,NTRY)=A
      ARRAY(3,NTRY)=B
      ARRAY(4,NTRY)=A
      ARRAY(5,NTRY)=SD
      DO 345 J=1,NV
      YV(J,NTRY)=YV(J,NTRY-1)
  345 ZV(J,NTRY)=ZV(J,NTRY-1)-CROOTT*SNALIT
      NEXT=NPANLT
      XF=99.9
      CALL TRJTRY
      GO TO 347
  349 CONTINUE
      CALL TRJTRY
      IF (MDUMY.GT.10) GO TO 53
C
C      SET UP TOTAL INDUCED VELOCITY FIELD ON TAIL
C
  347 CONTINUE
      IF (NPRINT.LT.0) GO TO 513
```

```
      WRITE (6,749) HEAD(2)
      WRITE (6,746)
  513 CONTINUE
      DO 146 J=1,NEXT
      DUM=CABS(VP(J))
      IF (DUM.GT.WLIMIT)  VP(J)= VP(J)*WLIMIT/DUM
      WV(J)=AIMAG(VP(J))
      A=REAL(VP(J))
      IF (NPRINT.LT.0) GO TO 146
      DUM=WV(J)+WB(J)
      WRITE (6,747) J,XP(J),SP(J),A,DUM
  146 CONTINUE
  147 CONTINUE
C
C      COMPUTE NORMAL FORCE AND MOMENT CONTRIBUTION OF AFTERBODY
C
      IF (NREGNT.LE.0) XLET=XBDY(NTBL)
      XFINL=XLET
      DUM=0.0
      IF (RAVGW.GT.0.0) DUM=0.5*(XFINL-XINIT)/RAVGW
      NG=DUM
      IF (NG.LE.0) GO TO 150
      CALL CNVTX (XINIT,XFINL,NG,XM,CN,CM)
      CNA=2.0*PI*CN/REFS
      CMA=2.0*PI*CM/(REFS*REFL)
      XCPA=XM-CMA*REFL/CNA
  150 IF (NREGNT.LE.0) GO TO 60
C
C      COMPUTE LOADS AND FORCE ON TAIL
C
      DO 41 J=1,NPANLT
   41 ALPHA(J)=ALPHAT(J) + ALPU + ALPI + WB(J) + WV(J)
      NSURF=2
      COFT=0.
      CALL FORCE (NSURF,NR,NPL,NMAT,NPRINT,NCASE,NREGNT,TITLE,
     1            NPANLT,EM,REFS,REFL,XM,AT,XYT,XCPT,YCPT,XBART,AREAT,
     2            SLET,STET,DZDXT,NRCT,COFT,ALPHA,UTT,IPT)
      CNTBP=CLP*CSALIT
      IF (XTCP.GT.0.0) XCP=XTCP
      XCPTBP=XCP
      CATBP=CNTBP*SNALIT/CSALIT
      CMTBP=(XM-XCPTBP)*CNTBP/REFL
      CALL WPANL (NSURF,NR,NPL,NMAT,NPRINT,NPANLT,EM,XYT,XCPT,YCPT,
     1            XBART,NPRINT,SLET,STET,TOLT,XPNLFT,XPNLBT)
      CALL EDGFRC (NSURF,NR,NPL,NMAT,NPANLT,NREGNT,REFS,
     1             XYT,YCPT,SLET,STET,DZDXT,NRCT,YSPANT,PNLCT,
     2             TIPYT,SLOPT1,TOLT,CT1PT,CROOTT,ALPHA,NPRINT,
     3             YSEPT,NSEPT,AKVLT1,AKVLT2,AKVST)
      DO 220 J=1,NSEPT
      DUM=0.0
      IF (Y1(J).GT.0.0) DUM=1.0/(8.0*PI*Y1(J))
      GAM1(J)=GAM1(J)*REFS*CSALP*DUM
  220 Y1(J)=Y1(J) + XYT(5,1)
      DO 344 J=1,NV
      YG(J)=YV(J,NTRY)
  344 ZG(J)=ZV(J,NTRY)
      WRITE (6,711) XV(NTRY)
      WRITE (6,712) (I,GAM(I),YG(I),ZG(I),I=1,NV)
      ZTR=ZHLT - (XTET-XHLT)*SNALIT
      NI=NV+1
      WRITE (6,732) NI,GAMT,YTR,ZTR
      CNTBVL=(CNV1*AKVLT1 + CNV2*AKVLT2)*CSALIT
      CNTBVS=CNVS*AKVST*CSALIT
      XCPTVL=COPLE
      XCPTVS=COPTIP
      CMTBVL=(XM-XCPTVL)*CNTBVL/REFL
      CMTBVS=(XM-XCPTVS)*CNTBVS/REFL
      CATBVL=CNTBVL*SNALIT/CSALIT
      CATBVS=CNTBVS*SNALIT/CSALIT
C
C      DETERMINE VERTICAL LOCATION OF SEPARATION VORTICES
C
      CALL ZVTX (ALPT  ,EPS,BTA2)
      IF (CNTBVL.LT.0.0 .AND. ALPT.GT.0.0) BTA2=-BTA2
      Z=CRWT*SIN(BTA2/RAD)
      SUMGAM=0.0
      IF (AKVLT1.LE.0.0 .AND. AKVLT2.LE.0.0) GO TO 530
      DO 244 I=1,NSEPT
      SUMGAM=SUMGAM+GAM1(I)
      IF (I.GT.1) Z=(CTWT+(CRWT-CTWT)*YSEPT/SSPANT)*SIN(BTA2/RAD)
      Z=Z+ZHLT-(XTET-XHLT)*SNALIT
      SIGMA(I)=CMPLX(Y1(I),Z)
      IF (ABS(GAM1(I)).LT.1.0E-06) GO TO 244
      NI=NI+1
      WRITE (6,732) NI,GAM1(I),Y1(I),Z
  244 CONTINUE
C
C      COMPUTE NORMAL FORCE AND MOMENT ON BODY IN PRESENCE OF TAIL
C
  530 CONTINUE
      TAPER=CTWT/CRWT
      DUM=1.0
      SWPLE=ATAN2(SLET(1),DUM)*RAD
      SMBW=1.0E+08
      IF (SWPLE.GT.0.0) SMBW=BETA/TAN(SWPLE/RAD)
      ROSW=RAVGT/SSPANT
      CSW=(SSPANT-RAVGT)*(CTWT+CRWT)
      BETAAW=4.0*BETA*((SSPANT-RAVGT)**2)/CSW
      BDCRW=BETA*2.0*RAVGT/CRWT
      CALL CH1416 (SSPANT,RAVGT,SWPLE,CTWT,CRWT,EM,PARAM,NAFT ,BDCRW,
     PARAM=BETAAW*(1.0+1.0/SMBW)*(1.0+TAPER)
     1             XCPBTP,SMBW,TAPER,ROSW,BETAAW)
      XCPBTP=XCPBTP*CRWT+XLET
      XCPBTV=XCPBTP
      XCPBTS=XCPBTP
      CLAW=CHRT8 (BETAAW,SSPANT,RAVGT,CTWT,CRWT,SWPLE)/(BETA*RAD)
      CLAR=CLAW*RAD
      CALL KFACT(ROSW,CKWB,EM,PARAM,CKBW,NAFT ,SMBW,TAPER,BDCRW,CLAR,
     1              BETA,SWPLE,BETAAW,DUM,DUM)
      CNBTP=CKBW*CNTBP/CKWB
      CNBTV=CKBW*CNTBVL/CKWB
      CNBTVS=CKBW*CNTBVS/CKWB
  531 CMBTP=(XM-XCPBTP)*CNBTP/REFL
      CMBTV=(XM-XCPBTV)*CNBTV/REFL
      CMBTVS=(XM-XCPBTS)*CNBTVS/REFL
C
C
   60 CONTINUE
      CNTOT=CNNP + CNNV + CNWBP + CNWBVL + CNWBVS + CNBWP + CNBWV + CNA
     1     + CNTBP + CNTBVL + CNTBVS + CNBTP + CNBTV + CNBWVS + CNBTVS
      CMTOT=CMNP + CMNV + CMWBP + CMWBVL + CMWBVS + CMBWP + CMBWV + CMA
     1     + CMTBP + CMTBVL + CMTBVS + CMBTP + CMBTV + CMBWVS + CMBTVS
      XCPTOT=XM-CMTOT*REFL/CNTOT
      CATOT=CAWBP + CAWBVL + CAWBVS + CATBP + CATBVL + CATBVS
      CL=0.0
      TNA=TAN(ALPHAD/RAD)
      CBA=COS(ALPHAD/RAD)
      CSAW=COS(ALPW/RAD)
      CSAT=COS(ALPT/RAD)
      WRITE (6,701)
      WRITE (6,737) ALPHAD,EM,ALPIW,ALPIT,AKVLW1,AKVSW,AKVLT1,AKVST
     1 ,              WLIMIT
      IF (NSEPW.GT.1 .OR. NSEPT.GT.1) WRITE (6,751) AKVLW2,AKVLT2
      WRITE (6,703)
      WRITE (6,713)
      WRITE (6,714) HEAD(7)
      CL1=CNNP*CBA
      CL2=CNNV*CBA
      CD1=CL1*TNA
      CD2=CL2*TNA
      CL=CL+CL1+CL2
      WRITE (6,715) HEAD(9),CNNP,CMNP,XCPNP,CL1,CD1
      WRITE (6,710)  CNNV,CMNV,XCPNV,CL2,CD2
      WRITE (6,714) HEAD(1)
      CL1=CNWBP*CSAW
      CL2=CNWBVL*CSAW
      CL3=CNWBVS*CSAW
      CD1=CL1*TNA
      CD2=CL2*TNA
      CD3=CL3*TNA
      WRITE (6,715) HEAD(3),CNWBP,CMWBP,XCPWBP,CL1,CD1,CAWBP
      WRITE (6,716) HEAD(3),CNWBVL,CMWBVL,XCPWVL,CL2,CD2,CAWBVL
      WRITE (6,717) HEAD(3),CNWBVS,CMWBVS,XCPWVS,CL3,CD3,CAWBVS
      CL=CL+CL1+CL2+CL3
```

```
      WRITE (6,714) HEAD(8)
      CL1=CNBWP*CBA
      CL2=CNBWV*CBA
      CL3=CNBWVS*CSA
      CD1=CL1*TNA
      CD2=CL2*TNA
      CD3=CL3*TNA
      WRITE (6,715) HEAD(4),CNBWP,CMBWP,XCPBWP,CL1,CD1
      WRITE (6,716) HEAD(4),CNBWV,CMBWV,XCPBWV,CL2,CD2
      WRITE (6,717) HEAD(4),CNBWVS,CMBWVS,XCPBWS,CL3,CD3
      CL=CL+CL1+CL2+CL3
      CL1=CNA*CSA
      CD1=CL1*TNA
      WRITE (6,718) CNA,CMA,XCPA,CL1,CD1
      CL=CL+CL1
      WRITE (6,714) HEAD(2)
      CL1=CNTBP*CSAT
      CL2=CNTBVL*CSAT
      CL3=CNTBVS*CSAT
      CD1=CL1*TNA
      CD2=CL2*TNA
      CD3=CL3*TNA
      CL=CL+CL1+CL2+CL3
      WRITE (6,715) HEAD(5),CNTBP,CMTBP,XCPTBP,CL1,CD1,CATBP
      WRITE (6,716) HEAD(5),CNTBVL,CMTBVL,XCPTVL,CL2,CD2,CATBVL
      WRITE (6,717) HEAD(5),CNTBVS,CMTBVS,XCPTVS,CL3,CD3,CATBVS
      WRITE (6,714) HEAD(8)
      CL1=CNBTP*CBA
      CL2=CNBTV*CBA
      CL3=CNBTVS*CBA
      CD1=CL1*TNA
      CD2=CL2*TNA
      CD3=CL3*TNA
      CL=CL+CL1+CL2+CL3
      WRITE (6,715) HEAD(6),CNBTP,CMBTP,XCPBTP,CL1,CD1
      WRITE (6,716) HEAD(6),CNBTV,CMBTV,XCPBTV,CL2,CD2
      WRITE (6,717) HEAD(6),CNBTVS,CMBTVS,XCPBTS,CL3,CD3
      CD1=CL*TNA
      WRITE (6,719) CNTOT,CMTOT,XCPTOT,CL,CD1 ,CATOT
      CD2=0.0
      IF (CL.NE.0.0) CD2=CD1/(CL*CL)
      WRITE (6,742) CD2
      IF (NTRY.EQ.0 .OR. NPRINT.LT.0) GO TO 19
   52 WRITE (6,720)
   53 J=1
      IF (NOSEV.EQ.0) GO TO 56
      WRITE (6,727) J
      J=J+1
   56 IF (J.GT.NV) GO TO 58
      WRITE (6,728) J
      J=J+1
      IF (J.GT.NV) GO TO 58
      DO 57 K=J,NV
   57 WRITE (6,729) K
   58 CONTINUE
      WRITE (6,725)
      DO 51 J=1,NV
   51 WRITE (6,726) J,GAM(J)
      IF (NTRY.EQ.0) GO TO 19
      DO 55 J=1,NTRY
      JV=NVP(J)
      WRITE (6,721)
      WRITE (6,722) XV(J),(ARRAY(N,J),N=1,5)
      WRITE (6,723)
      DO 54 K=1,JV
   54 WRITE (6,724) K,YV(K,J),ZV(K,J)
   55 CONTINUE
      GO TO 19
  953 CONTINUE
  996 STOP 182
      END
```

```
      SUBROUTINE SSWING (NSURF,NREGN,NR,NPL,NMAT,OUTCOD,MSPN,
     1                   MACH,SREF,CBAR,REFMOM,TITLE,BS2 ,SLOPE1,CTIP,
     2                   CROOT,TOLRNC,PANELS,ROWCOL,A,XY,XCPT,YCPT,XBAR,
     3            AREA,SLE,STE,DZDX,XPNLF,XPNLB,XSPAN,YSPAN,
     4            PNLC,IP,UT,ARAY)
C      FORMERLY MAIN PROGRAM OF NASA/AMES WING PROGRAM
      DIMENSION XY(6,100),XCPT(NPL),YCPT(NPL),XBAR(NPL),AREA(NPL)
      DIMENSION SLE(NPL),STE(NPL),DZDX(NPL),ROWCOL(2,20)
      DIMENSION PNLC(NPL),A(NMAT),IP(NPL),UT(NPL)
      DIMENSION XPNLF(NPL),XPNLB(NPL)
      DIMENSION DVROOT(20),DVTIP(20),DVSPAN(20),SLOPE(20),NTRCPT(20)
      DIMENSION THICK(2),ARAY(6),XSPAN(20),YSPAN(20)
      DIMENSION ZROOT(20),ZTIP(20),SCF(20),CHORD(20),TITLE(20),HEAD(2)
      REAL LE,MACH,MAC,NTRCPT
      INTEGER TYPE,PANELS,REGION,ROWS,COLS,ROWCOL,SECT,OUTCOD
      LOGICAL ASYM,NOTAPR
      COMMON /PARAMS/ ASYM
      DATA HEAD /4HWING,4HTAIL/
C
      NAMELIST /INPUT/ NSF,ROWS,COLS,PER,ROOTLE,ROOTTE,ROOTY,
     1              TIPLE,TIPTE,TIPY,TCROOT,TCTIP,ASYM,SECT,ALT,SCALE,TYPE
    6 FORMAT (1H0 34H GEOMETRICAL DESCRIPTION OF REGION,I3,I6,
     2  8H COLUMNS,I4,5H ROWS)
    7 FORMAT (/10X10HGRID LINES/5X7HY(SPAN),5X7HX(ROOT),
     1          6X6HX(TIP),7X5HSLOPE)
    8 FORMAT (4F12.5)
    9 FORMAT(F12.5)
   10 FORMAT (12X,3F12.5)
   71 FORMAT(40H0SURFACE AREA (UPPER+LOWER)..............    F12.4/
     X       40H VOLUME.................................    F12.4)
  151 FORMAT(23H1NASA/AMES WING PROGRAM,5X,11HSUMMARY OF  ,A4,
     1      9H GEOMETRY / 1X,20A8 /
     1 40H0TOTAL AREA OF PANELS ON WING............   F12.4/
     2 40H WING REFERENCE AREA.....................   F12.4/
     3 40H WING WETTED AREA........................   F12.4/
     4 40H MEAN AERODYNAMIC CHORD..................   F12.4/
     5 40H REFERENCE CHORD LENGTH..................   F12.4/
     6 40H WING VOLUME.............................   F12.4/
     7 40H REYNOLDS NUMBER.........................   1PE14.2)
  751 FORMAT (37H0ERROR IN DATA DEFINING WING PLANFORM)
  787 FORMAT (39H0 MACH NUMBER MUST BE GREATER THAN 1.0  )
  791 FORMAT(29H0PANEL COUNT WOULD EXCEED 100)
  795 FORMAT(36H0NEITHER ROWS NOR COLS MAY EXCEED 20)
  785 FORMAT(34H0NO MORE THAN 20 REGIONS PERMITTED)
  798 FORMAT(51H0PROGRAM TERMINATED BY COST-CONSCIOUS ERROR CATCHER)
  792 FORMAT(21H0STOP IN SSWING
      RNL=0.0
      IF (MACH.LE.1.0) GO TO 786
      XX=0.
      TVOLUM=0.
      SWET=0.
      CDF=0.
      MAC=0.
      STOTAL=0.
      REGION=0
      PANELS=0
      MSPN=0
      NSPN=1
C      NAMELIST VARIABLES INITIALIZED
      ASYM=.FALSE.
      TCROOT=0.0
      TCTIP=0.0
      SECT=1
      ALT=-1.0
      SCALE=1.0
      PER=0.95
      TYPE=4
      NREG=0
      NSF=NSURF
 1000 IF (NREG - NREGN) 1001,899,796
C
C      INPUT DESCRIPTION OF EACH REGION
C
 1001 READ (5,INPUT)
      WRITE (6,INPUT)
      NREG=NREG+1
      IF (NREG.GT.1) GO TO 1002
```

```
      ARAY(1)=ROOTLE
      ARAY(2)=ROOTTE
      ARAY(3)=ROOTY
 1002 IF (NREG.LT.NREGN) GO TO 1003
      ARAY(4)=TIPLE
      ARAY(5)=TIPTE
      ARAY(6)=TIPY
 1003 K=ROWS
      BS2=TIPY
      ROOTY=ROOTY-ARAY(3)
      TIPY=TIPY-ARAY(3)
      THICK(1)=TCROOT
      THICK(2)=TCTIP
C-----TEST FOR IMPOSSIBLE GEOMETRICAL SITUATIONS.
      IF(ROOTLE.GT.ROOTTE.OR.TIPLE.GT.TIPTE.OR.ROOTY.GE.TIPY)GO TO 750
C-----TEST FOR OVERFLOW OF TABLES.
  995 IF(PANELS+ROWS*COLS.GT.NPL)GO TO 790
      IF(ROWS.GT.NR.OR.COLS.GT.NR)GO TO 794
      IF(REGION.GE.NR)GO TO 784
      IF(TYPE.LE.0.OR.TYPE.GT.6) TYPE=4
      REGION=REGION+1
      ROWCOL(1,REGION)=ROWS
      ROWCOL(2,REGION)=COLS
      DY=TIPY-ROOTY
      CROOT=ROOTTE-ROOTLE
      CTIP=TIPTE-TIPLE
      MAC=MAC+DY*(CROOT*(CROOT+CTIP)+CTIP*CTIP)
      M1=COLS+1
      N1=ROWS+1
C
C     SET TOLERANCE USED IN SUBROUTINE COMP
C
      TOLRNC= (TIPY*15.0E-05)**2 * 1000.
      SLOPE1=(TIPLE-ROOTLE)/DY
C
C-----NOW THE DVROOT ARRAY WILL BE COMPUTED IF IT HAS NOT BEEN  INPUT
C-----DVROOT MUST BE COMPUTED IF TYPE= 3 OR 4
      IF(TYPE.LT.3.OR.TYPE.GT.4) GO TO 16
      DVROOT(1)=ROOTLE
      DVROOT(N1)=ROOTTE
      CALL FILL(DVROOT,ROWS)
   16 CONTINUE
C-----NOW THE DVTIP ARRAY MUST BE COMPUTED IF IT HAS NOT BEEN INPUT
C-----DVTIP MUST BE COMPUTED IF TYPE=3,4,5,OR 6
      IF (TYPE.LE.2) GO TO 30
C-----DVTIP IS COMPUTED EVENLY FOR TYPE=3,4 CONICALLY FOR TYPE=5 OR  6
      IF (TYPE.GT.4) GO TO 26
      DVTIP(1)=TIPLE
      DVTIP(N1)=TIPTE
      CALL FILL(DVTIP,ROWS)
      GO TO 30
C-----THE CONICAL DIVISION SCHEME USES THE INTERSECTION OF THE LEADING
C-----AND TRAILING EDGES AND THE DVROOT ARRAY TO DEFINE THE SPANWISE
C-----PANEL DIVISION LINES.
C-----IF THE LEADING AND TRAILING EDGES ARE PARALLEL,THEN THE SPANWISE
C-----DIVISION LINES ARE ALSO PARALLEL.
C-----THE EQUATION OF THE LEADING EDGE IS      X=SLOPE1*Y+C1
C-----THE EQUATION OF THE TRAILING EDGE IS     X=SLOPE2*Y+C2
C-----THE INTERSECTION OF THE LEADING AND TRAILING EDGES IS  (XEND,YEND)
C-----NOTAPR IS TRUE IF LEADING + TRAILING EDGES ARE PARALLEL (NO TAPER)
   26 SLOPE1=(TIPLE-ROOTLE)/DY
      SLOPE2=(TIPTE-ROOTTE)/DY
      C1=ROOTLE-SLOPE1*ROOTY
      C2=ROOTTE-SLOPE2*ROOTY
      NOTAPR=SLOPE1.EQ.SLOPE2
      IF(NOTAPR)GO TO 28
      YEND=(C1-C2)/(SLOPE2-SLOPE1)
      XEND=SLOPE1*YEND+C1
   28 DO 27 I=1,N1
      SLOPE(I)=SLOPE1
      IF(.NOT.NOTAPR)SLOPE(I)=(XEND-DVROOT(I))/(YEND-ROOTY)
      NTRCPT(I)=DVROOT(I)-SLOPE(I)*ROOTY
   27 DVTIP(I)=SLOPE(I)*TIPY+NTRCPT(I)
      GO TO 35
C-----THE ARRAYS SLOPE, AND NTRCPT MUST NOW BE COMPUTED FOR TYPE=1,2,3,4
   30 DO 31 I=1,N1
      SLOPE(I)=(DVTIP(I)-DVROOT(I))/DY
   31 NTRCPT(I)=DVROOT(I)-SLOPE(I)*ROOTY
C-----THE DVSPAN ARRAY MUST BE COMPUTED IF IT HAS NOT BEEN INPUT
C-----DVSPAN MUST BE COMPUTED IF TYPE=2,4,6
   35 IF (MOD(TYPE,2).EQ.1) GO TO 39
C-----TEST ABOVE SENDS CONTROL TO 39 IF AND ONLY IF TYPE IS ODD.
      DVSPAN(1)=ROOTY
      DVSPAN(M1)=TIPY
      CALL FILL(DVSPAN,COLS)
   39 CONTINUE
C-----THE ARRAYS DVROOT,DVTIP,DVSPAN,SLOPE,NTRCPT ARE ALL COMPUTED.
      IF(OUTCOD.LT.2)GO TO 15
      WRITE(6,6)  REGION,COLS,ROWS
      WRITE(6,7)
      MM=MIN0(M1,N1)
      WRITE(6,8)(DVSPAN(I),DVROOT(I),DVTIP(I),SLOPE(I),I=1,MM)
      J=MM+1
      IF(M1.GT.MM)WRITE(6,9)(DVSPAN(I),I=J,M1)
      IF(N1.GT.MM)WRITE(6,10)(DVROOT(I),DVTIP(I),SLOPE(I),I=J,N1)
C-----FOR EACH PANEL ON THE WING, THE CORNER POINTS, AREA, CENTROID,
C-----DOWNWASH CONTROL POINT ARE COMPUTED.
   15 DO 50 J=1,COLS
      NSPN=NSPN+1
      Y1=DVSPAN(J)
      Y2=DVSPAN(J+1)
C-----(Y1+Y2)/2 IS MIDSPAN OF THIS COLUMN OF PANELS.
      Y3=(Y1+Y2)/2.0
      DELTA=(Y3-ROOTY)/(TIPY-ROOTY)
      THCK=TCROOT-DELTA*(TCROOT-TCTIP)
      CHORD(J)=CROOT-DELTA*(CROOT-CTIP)
      XLE=ROOTLE + Y3*SLOPE1
      DO 50 I=1,ROWS
      PANELS=PANELS+1
C-----LOAD XY ARRAY WITH CORNER POINTS.
      XY(1,PANELS)=NTRCPT(I)+SLOPE(I)*Y1
      XY(2,PANELS)=NTRCPT(I)+SLOPE(I)*Y2
      XY(3,PANELS)=NTRCPT(I+1)+SLOPE(I+1)*Y1
      XY(4,PANELS)=NTRCPT(I+1)+SLOPE(I+1)*Y2
      XY(5,PANELS)=Y1
      XY(6,PANELS)=Y2
      IF (J.EQ.1 .AND. I.EQ.1 .AND. NREG.EQ.1) XSPAN(1)=XY(1,PANELS)
      IF (I.EQ.1) XSPAN(NSPN)=XY(2,PANELS)
      CR=XY(3,PANELS)-XY(1,PANELS)
      CT=XY(4,PANELS)-XY(2,PANELS)
      YPER=(CR  + 2.*CT)/(3.0*(CR +  CT))
      YPER1=1.0-YPER
      YC=YPER*Y2+YPER1*Y1
      YCPT(PANELS)=YC
      LE=NTRCPT(I)+SLOPE(I)*YC
      TE=NTRCPT(I+1)+SLOPE(I+1)*YC
      PNLC(PANELS)=TE-LE
C-----LEADING +TRAILING EDGES OF PANEL AT CHORD THRU CENTROID OF PANEL.
      SLE(PANELS)=SLOPE(I)
      STE(PANELS)=SLOPE(I+1)
C-----SLE,STE ARE SLOPES (DX/DY) OF LEADING+TRAILING EDGES OF PANEL
C-----NEXT, COMPUTE CONTROL POINT(XCPT,YCPT) AND CENTROID(XBAR,YCPT)
      XCPT(PANELS)=LE+PER*(TE-LE)
      XBAR(PANELS)=(LE+TE)*0.5
      XPNLF(PANELS)= LE + 0.25*(TE-LE)
      XPNLB(PANELS) = LE + 0.75*(TE-LE)
      AREA(PANELS)=(CR+CT)*(Y2-Y1)*0.5
      STOTAL=STOTAL+AREA(PANELS)
C-----COMPUTE DZDX FOR THIS PANEL UNLESS SECT=999   (USE INPUT TABLE)
      IF(SECT.GT.999)GO TO 50
      IF(THCK.GT.0.)CALL ZSECT((XBAR(PANELS)-XLE)/CHORD(J),DUMMY,XX,
     1SECT)
      DZDX(PANELS)=XX*THCK
   50 CONTINUE
C-----ENDS BIG DOUBLE LOOP.
C-----VOLUME AND SURFACE AREA CALCULATION
      CROOT=ROOTTE-ROOTLE
      CTIP=TIPTE-TIPLE
C-----LOAD ARRAYS ZROOT+ZTIP WITH HALF THICKNESS AT PTS. OF DVROOT+DVTIP
      DO 60 I=1,N1
      XX=0.
      IF(CROOT.GT.0.)CALL ZSECT((DVROOT(I)-ROOTLE)/CROOT,XX,DUMMY,SECT)
      ZROOT(I)=XX*TCROOT
      XX=0.
```

```
      IF(CTIP.GT.0.)CALL ZSECT((DVTIP(I)-TIPLE)/CTIP,XX,DUMMY,SECT)
      ZTIP(I)=XX*TCTIP
   60 CONTINUE
      DVTERM=0.
      S1=0.
      S2=0.
C-----COMPUTE S1, THE ARC LENGTH OF ROOT UPPER SURFACE.
C-----COMPUTE S2, THE ARC LENGTH OF TIP  UPPER SURFACE.
C-----COMPUTE VOLUME OF THIS WING REGION.
      DO 70 I=1,N
      DX1=DVROOT(I+1)-DVROOT(I)
      DX2=DVTIP(I+1)-DVTIP(I)
      DZ1=ZROOT(I)+ZROOT(I+1)
      DZ2=ZTIP(I)+ZTIP(I+1)
      DVTERM=DVTERM+DX1*(DZ1+DZ1+DZ2)+DX2*(DZ1+DZ2+DZ2)
      ZZ=ZROOT(I+1)-ZROOT(I)
      DS1=SQRT(DX1*DX1+ZZ*ZZ)
      S1=S1+DS1
      ZZ=ZTIP(I+1)-ZTIP(I)
      DS2=SQRT(DX2*DX2+ZZ*ZZ)
   70 S2=S2+DS2
      VOLUME=DY*DVTERM/6.
C-----VOLUME IS VOLUME OF THIS REGION AND TVOLUM IS RUNNING TOTAL.
C-----SAREA IS SURFACE AREA OF THIS REGION, BOTH UPPER AND LOWER.
C-----SWET IS THE RUNNING TOTAL OF ALL REGIONS.
      SAREA=(S1+S2)*DY
C-----DOUBLE THE SURFACE AREA AND VOLUME FOR SYMMETRIC PLANFORMS.
      IF(.NOT.ASYM)SAREA=SAREA+SAREA
      IF(.NOT.ASYM)VOLUME=VOLUME+VOLUME
      SWET=SWET+SAREA
      TVOLUM=TVOLUM+VOLUME
      IF(OUTCOD.GT.1)WRITE(6,71)SAREA,VOLUME
C-----TURBULENT SKIN FRICTION CALCULATION
C-----FOR EACH POINT IN DVSPAN(COLS+1 OF THEM), COMPUTE A CF BASED ON
C     LOCAL REYNOLDS NUMBER.  THEN MULTIPLY BY LOCAL ARC LENGTH TO
C----- GET LOCAL C*CF AT EACH POINT IN DVSPAN
C-----THEN INTEGRATE C*CF VS. Y TO GET D/Q(FRICTION).
C-----THE NEXT GROUP OF STATEMENTS ARE A POOR MAN*S ATMOSPHERE ROUTINE.
C-----IT IS VALID TO 20 KM.  ALT IS GEOMETRICAL ALTITUDE IN KM.
C
      IF(ALT.LT.0..OR.MACH.EQ.0.)GO TO 1010
C
C-----CONVERT GEOMETRICAL ALTITUDE TO GEOPOTENTIAL ALTITUDE.
      H=ALT/(1.0+1.57338E-4*ALT)
      IF(H.GT.11.)GO TO 73
C-----NEXT THREE STATEMENTS APPLY TO TROPOSPHERE  (ALT)11 KM.)
      TEMP=288.15-6.5*H
      PRESS=101325.*(288.15/TEMP)**(-5.25588)
      ASOUND=SQRT(401.8743*TEMP)
      GO TO 74
C-----NEXT THREE STATEMENTS APPLY TO STRATOSPHERE (ALT *11 KM.)
   73 TEMP=216.65
      ASOUND=295.069
      PRESS=22632.*EXP(.15769*(11.-H))
   74 DEN=3.483676E-3*PRESS/TEMP
      VISC=1.458E-6*SQRT(TEMP*TEMP*TEMP)/(TEMP+110.4)
      RNL=DEN*ASOUND*MACH/VISC
C-----RNL WILL BE IN UNITS OF 1/METERS.
C-----SCALE IS USED TO ALLOW INPUT IN UNITS OTHER THAN METERS.
C-----IF INPUT IS IN METERS,      SCALE=1.
C                   CENTIMETERS,  SCALE=.01
C                   INCHES,       SCALE=.0254
C                   FEET,         SCALE=.3048
   75 DO 80 J=1,M1
      Y=DVSPAN(J)
      THETA=(Y-ROOTY)/(TIPY-ROOTY)
      SC=S2*THETA+S1*(1.-THETA)
C-----SC IS THE LOCAL ARC LENGTH AT Y.
      IF(SC.LE.0.)GO TO 81
      RX=RNL*SC*SCALE
      XX=ALOG10(RX)-1.5
      CFI=.088/(XX*XX)
      Z=1.0+.115*MACH*MACH
      RSTAR=RX*(Z*TEMP+198.6)/((TEMP+198.6)*(Z**2.5))
      CF=CFI/Z*(XX/(ALOG10(RSTAR)-1.5))**2
      SCF(J)=SC*CF
      GO TO 80
   81 SCF(J)=0.
   80 CONTINUE
C-----INTEGRATE TO GET SKIN FRICTION USING TRAPEZOIDAL RULE.
C-----NOTE ABSENCE OF 0.5 IN TRAP RULE FORMULA. WE ARE GETTING UPPER
C-----AND LOWER SIDES OF THE WING IN THIS CALCULATION.
      XX=0.
      DO 90 I=1,COLS
   90 XX=XX+(DVSPAN(I+1)-DVSPAN(I))*(SCF(I+1)+SCF(I))
      CDF=CDF+XX
      IF(.NOT.ASYM)CDF=CDF+XX
 1010 IF (NREG.GT.1) MSPN=MSPN-1
      DO 1011 J=1,M1
      MSPN=MSPN+1
 1011 YSPAN(MSPN)=DVSPAN(J)
      GO TO 1000
C
C-----ENDS LOOP FOR THIS WING REGION.
C-----GO BACK TO 1000 TO LOOK FOR ANOTHER WING REGION.
C
  899 MAC=MAC/(STOTAL*3.)
      IF(.NOT.ASYM)STOTAL=STOTAL+STOTAL
      IF(SREF.EQ.0.) SREF=STOTAL
      IF(CBAR.EQ.0.) CBAR=MAC
      CDF=CDF/SREF
      RE=RNL*CBAR*SCALE
  150 IF(OUTCOD.GT.0)WRITE(6,151) HEAD(NSURF),TITLE,STOTAL,SREF,SWET,
     1                 MAC,CBAR,TVOLUM,RE
  160 CALL MATRIX (NSURF,NR,NPL,NMAT,A,XY,XCPT,YCPT,XBAR,SLE,STE,
     1             DZDX,PANELS,MACH,TOLRNC,IP,UT)
      NREG=NREG+1
      CTIP=ARAY(5)-ARAY(4)
      CROOT=ARAY(2)-ARAY(1)
      ROOTY=ARAY(3)
      TIPY=ARAY(6)
      BS2=TIPY
      DO 200 J=1,PANELS
      XY(5,J)=XY(5,J)+ROOTY
      XY(6,J)=XY(6,J)+ROOTY
      YCPT(J)=YCPT(J)+ROOTY
  200 CONTINUE
      DO 201 J=1,MSPN
      YSPAN(J)=YSPAN(J)+ROOTY
  201 CONTINUE
      RETURN
C-----FOLLOWING SECTION OF CODE IS FOR ERROR HANDLING
  750 WRITE (6,751)
      GO TO 799
  790 WRITE(6,791)
      GO TO 799
  794 WRITE(6,795)
      GO TO 799
  784 WRITE(6,785)
      GO TO 799
  786 WRITE (6,787)
  799 WRITE(6,798)
  796 WRITE(6,792)
      STOP 145
      END
```

```
      SUBROUTINE TABLE (XLEW,XTEW,CRW,CTW,SSPANW,XLELW,YW,IMAX,PSILEW,
     1                  XLET,XTET,CRT,CTT,SSPANT,XLELT,YT,IMAXT,PSILET)
C
C     SET UP TABLE OF BODY, WING, AND TAIL COORDINATES FOR USE IN
C         VORTEX TRAJECTORY CALCULATIONS
C
      DIMENSION XLELW(20),XLELT(20),YW(20),YT(20)
      COMMON /VTABLE/ NTRY,XV(200),YV(4,200),ZV(4,200),NVP(200),
     1                ARRAY(5,200)
      COMMON /BODY/ NTBL,MTBL,XBDY(100),RBDY(100),SBDY(100),DRBDY(100)
  708 FORMAT (I10,4F10.5)
  798 FORMAT (//5X63H***** ERROR IN SUBROUTINE TABLE, TOO FEW ENTRIES IN
     1 BODY TABLE  )
  799 FORMAT (//5X40H****  EXECUTION STOP, BODY TABLE EXCEEDED )
      MTBL=1
      RAVGW=0.0
      RAVGT=0.0
      BETA=1.0
      NT=1
      J=1
   30 CONTINUE
      XV(J)=XBDY(NT)
      YV(1,J)=RBDY(NT)
      ZV(1,J)=RBDY(NT)
      IF (XV(J).GT.XLEW) GO TO 31
   32 NT=NT+1
      J=J+1
      IF (NT.LE.NTBL) GO TO 30
      GO TO 40
   31 IF (XV(J).GT.XTEW .AND. XV(J).LE.XLET) GO TO 32
      IF (XV(J).GT.XLET) GO TO 35
      IMIN=2
      IF (PSILEW   .GT.1.0E-02) GO TO 135
      IMIN=IMAX
      XV(J-1)=XV(J-1)-.001
  135 DO 33 I=IMIN,IMAX
      IF (J .GE. 100) GO TO 40
      XV(J)=XLELW(I)
      ZV(1,J)=YW(I)
  133 IF (XV(J)-XBDY(NT)) 233,333,433
  333 YV(1,J)=RBDY(NT)
      NT=NT+1
      GO TO 33
  433 NT=NT+1
      IF (NT.LE.NTBL) GO TO 133
      WRITE(6,799)
      WRITE(6,708) J,XV(J)
      J=433
      WRITE (6,708) J
      STOP
  233 DELTA=(XBDY(NT)-XV(J))/(XBDY(NT)-XBDY(NT-1))
      YV(1,J)=RBDY(NT)-DELTA*(RBDY(NT)-RBDY(NT-1))
   33 J=J+1
      IF (CTW.LT.(0.02*CRW)) GO TO 34
      XV(J)=XV(J-1)+CTW
      ZV(1,J)=SSPANW
  533 IF (XV(J)-XBDY(NT)) 633,933,833
  933 YV(1,J)=RBDY(NT)
      J=J+1
      IF (J .GE. 100) GO TO 40
      GO TO 34
  833 NT=NT+1
      IF (NT.LE.NTBL) GO TO 533
      WRITE (6,799)
      WRITE (6,708) J,XV(J)
      J=833
      WRITE (6,708) J
      STOP
  633 DELTA=(XBDY(NT)-XV(J))/(XBDY(NT)-XBDY(NT-1))
      YV(1,J)=RBDY(NT)-DELTA*(RBDY(NT)-RBDY(NT-1))
      J=J+1
      IF (J .GE. 100) GO TO 40
   34 XV(J)=XTEW + .001
      IF (XV(J).LE.XV(J-1)) XV(J)=XV(J-1)+0.001
      XINTE=XV(J)
  134 IF (XV(J)-XBDY(NT)) 234,334,434
  334 YV(1,J)=RBDY(NT)
      ZV(1,J)=RBDY(NT)
      NT=NT+1
      GO TO 934
  434 NT=NT+1
      IF (NT.LE.NTBL) GO TO 134
      WRITE (6,799)
      WRITE(6,708)J,XV(J)
      J=434
      WRITE (6,708) J
      STOP
  234 DELTA=(XBDY(NT)-XV(J))/(XBDY(NT)-XBDY(NT-1))
      YV(1,J)=RBDY(NT)-DELTA*(RBDY(NT)-RBDY(NT-1))
      ZV(1,J)=YV(1,J)
  934 J=J+1
      IF (J .GE. 100) GO TO 40
      GO TO 30
   35 IMIN=2
      IF (PSILET   .GT. 1.0E-02) GO TO 138
      IMIN=IMAXT
      XV(J-1)=XV(J-1)-.001
  138 DO 36 I=IMIN,IMAXT
      IF (J .GE. 100) GO TO 40
      XV(J)=XLELT(I)
      ZV(1,J)=YT(I)
  136 IF (XV(J)-XBDY(NT)) 236,336,436
  336 YV(1,J)=RBDY(NT)
      NT=NT+1
      GO TO 36
  436 NT=NT+1
      IF (NT.LE.NTBL) GO TO 136
      WRITE(6,799)
      WRITE(6,708) J,XV(J)
      J=436
      WRITE (6,708) J
      STOP
  236 DELTA=(XBDY(NT)-XV(J))/(XBDY(NT)-XBDY(NT-1))
      YV(1,J)=RBDY(NT)-DELTA*(RBDY(NT)-RBDY(NT-1))
   36 J=J+1
      IF (CTT.LT.(0.02*CRT))GO TO 37
      XV(J)=XV(J-1)+CTT
      ZV(1,J)=SSPANT
  536 IF (XV(J)-XBDY(NT)) 636,936,836
  936 YV(1,J)=RBDY(NT)
      J=J+1
      IF (J .GE. 100) GO TO 40
      GO TO 37
  836 NT=NT+1
      IF (NT.LE.NTBL) GO TO 536
      WRITE(6,799)
      WRITE(6,708) J,XV(J)
      J=836
      WRITE (6,708) J
      STOP
  636 DELTA=(XBDY(NT)-XV(J))/(XBDY(NT)-XBDY(NT-1))
      YV(1,J)=RBDY(NT)-DELTA*(RBDY(NT)-RBDY(NT-1))
      J=J+1
      IF (J .GE. 100) GO TO 40
   37 XV(J)=XTET + 0.001
      IF (XV(J).LE.XV(J-1)) XV(J)=XV(J-1)+0.001
  137 IF (XV(J)-XBDY(NT)) 237,337,437
  337 YV(1,J)=RBDY(NT)
      ZV(1,J)=RBDY(NT)
      NT=NT+1
      GO TO 38
  437 NT=NT+1
      IF (NT.LE.NTBL) GO TO 137
      WRITE(6,799)
      WRITE(6,708) J,XV(J)
      J=437
      WRITE (6,708) J
      STOP
  237 DELTA=(XBDY(NT)-XV(J))/(XBDY(NT)-XBDY(NT-1))
      YV(1,J)=RBDY(NT)-DELTA*(RBDY(NT)-RBDY(NT-1))
      ZV(1,J)=YV(1,J)
   38 J=J+1
      IF (J .GE. 100) GO TO 40
      DO 39 I=NT,NTBL
```

```
      XV(J)=XBDY(I)
      YV(1,J)=RBDY(I)
      ZV(1,J)=RBDY(I)
      J=J+1
      IF (J .GE. 100) GO TO 40
   39 CONTINUE
   40 CONTINUE
      NTBL=J-1
      DO 41 J=1,NTBL
      XBDY(J)=XV(J)
      RBDY(J)=YV(1,J)
      SBDY(J)=ZV(1,J)
      XV(J)=0.0
      YV(1,J)=0.0
   41 ZV(1,J)=0.0
      XBDY(NTBL+1)=XBDY(NTBL)+CR**
      RBDY(NTBL+1)=RBDY(NTBL)
      SBDY(NTBL+1)=SBDY(NTBL)
C
      DRBDY(1)=(RBDY(2)-RBDY(1))/(XBDY(2)-XBDY(1))
      DRBDY(NTBL)=(RBDY(NTBL)-RBDY(NTBL-1))/(XBDY(NTBL)-XBDY(NTBL-1))
      I=NTBL-1
      DRBDY(NTBL+1)=DRBDY(NTBL)
      DO 43 J=2,I
      DRZ1=(RBDY(J)-RBDY(J-1))/(XBDY(J)-XBDY(J-1))
      DRZ2=(RBDY(J+1)-RBDY(J))/(XBDY(J+1)-XBDY(J))
   43 DRBDY(J)=0.5*(DRZ1+DRZ2)
      IF (NTBL.LE.1) WRITE (6,798)
      RETURN
      END


      SUBROUTINE FONCE (NSURF,NR,NPL,NMAT,OUTCOD,CASE,REGION,TITLE,
     1                  PANELS,MACH,SREF,CBAR,REFMOM,A,XY,XCPT,YCPT,XBAR,
     2                  AREA,SLE,STE,DZDX,ROWCOL,CDF,ALPHA,UT,IP)
C     THIS SUBROUTINE COMPUTES THE SHAPE OR LOAD ON THE WING.
C-----   CASE=1  DIRECT PROBLEM, CP IS GIVEN, ALPHA IS COMPUTED.
C-----       =2  DIRECT PROBLEM, UPPER SURFACE CP IS GIVEN.
C-----       =-1  INDIRECT PROBLEM, ALPHA IS GIVEN AND CP IS COMPUTED
C-----       =-2 TO COMPUTE A FLAT WING AT A GIVEN LIFT COEFF. INPUT CL
C        CASE = -1 IS ONLY OPTION USED IN THIS VERSION
C
C-----   ALPHA= ARRAY OF POINTS CONTAINING THE INCIDENCE OF EACH PANEL.
C         POSITIVE ALPHA IS POSITIVE ANGLE OF ATTACK. THERE ARE *PANELS*
C         VALUES
C-----   CP= ARRAY OF DELTA-CP VALUES ON EACH INDIVIDUAL PANEL.  THERE
C         ARE *PANELS* VALUES
C
      DIMENSION CPT(100),CPU(100),CPL(100),F(100),LQ(100)
      DIMENSION XY(6,NPL),XCPT(NPL),YCPT(NPL),XBAR(NPL),AREA(NPL)
      DIMENSION SLE(NPL),STE(NPL),ROWCOL(2,NR),DZDX(NPL)
      DIMENSION A(NMAT),ALPHA(NPL),UT(NPL),IP(NPL)
      DIMENSION YSLD(20),SLOAD(20),TITLE(20),YWIDE(20),HEAD(2)
C
      REAL MACH,LQ,KPOLAR
      INTEGER PANELS,CASE,REGION,ROWS,COLS,ROWCOL,OUTCOD
C
      LOGICAL ASYM
C
      COMMON /RESLT2/ YTR,GAMT,CL,CM,CPXBAR
      COMMON/PARAMS/ ASYM
      COMMON/WNGFRC/CLW
      COMMON/PRESS/CP(100)
      DATA HEAD /4HWING,4HTAIL/
C
   17 FORMAT(23H1NASA/AMES WING PROGRAM,20X5H*****,A4,5H*****//5X,20A4)
   20 FORMAT(//13X,40H DRAG COEFFICIENT........................  F10.5/
     A 14X,39HDRAG COEFF. DUE TO THICKNESS............  F10.5/
     B 14X,39HDRAG COEFF. DUE TO SKIN FRICTION....... , F10.5/
     5       13X,40H PITCHING MOMENT COEFF AT ZERO LIFT.....    F10.5/
     6       13X,40H DRAG COEFFICIENT AT ZERO LIFT..........    F10.5/
     9       13X,40H MINIMUM DRAG COEFFICIENT...............    F10.5/
     A       13X,40H LIFT COEFF. FOR MINIMUM DRAG...........    F10.5/
     B       13X,40H LIFT COEFF. FOR MAX L/D................    F10.5/
     C       13X,40H MAXIMUM LIFT/DRAG......................    E10.3)
   21 FORMAT (//25X31HPROPERTIES OF INDIVIDUAL PANELS/2X1HJ,2X5HALPHA,
     2  2X,5HDT/DX,6X,2HCP,5X,8HDELTA-CP,6X,2HCP,8X,2HCP,6X,4HLOAD/
     3                 20X,9HTHICKNESS,15X,5HUPPER,5X,5HLOWER,5X,5H(L/Q))
   22 FORMAT(I4,F7.4,F7.3,F11.5,4F10.5)
  710 FORMAT(/4X23H SPAN LOAD DISTRIBUTION,5X 5H 2B =,F8.4/
     1          9X1HY,6X6HCCN/2B /(2X,2F10.5))
  711 FORMAT (/5X21HTRAILING VORTEX .... ,2X3HY =,F8.4,5X10HGAM/V2PI =,
     1             1PE12.4)
  712 FORMAT (/5X17HFLOW CONDITIONS = ,3X4HMACH,6X4HBETA /20X2F10.5)
  713 FORMAT (/5X22HREFERENCE QUANTITIES = ,3X4HAREA,5X6HLENGTH,
     1             6X4HXMOM,6X3HB/2,4X6HPANELS/25X4F10.3,I7)
  714 FORMAT (//17X,28H----------DCN/DALPHA--------/10X3HCNP,4X
     1             27H(RAD**-1) (DEG**-1)  * BETA,6X3HCMP,5X7HDCM/DCN,
     2             6X3HXAC,7X3HXCP,7X3HYCP)
  715 FORMAT (5X6F10.5,3F10.3)
C
      SHALF=SREF
      IF(.NOT.ASYM)SHALF=0.5*SHALF
      AINC=0.0
C-----COMPUTE CL-ALPHA AND A.C.
      CDT=0.
   10 DO 15 I=1,PANELS
      CPT(I)=-UT(I)-UT(I)
      CDT=CDT+CPT(I)*DZDX(I)*AREA(I)
   15 F(I)=-1.0
      CDT=2.0*CDT/SHALF
      CALL SOLVE(F,A,PANELS,NPL,IP)
      AC=0.
      CLA=0.
      DO 25 I=1,PANELS
      XX=AREA(I)*F(I)
      CLA=CLA+XX
   25 AC=AC+XBAR(I)*XX
      AC=AC/CLA
      CLA=CLA/SHALF
      CMCL=(REFMOM-AC)/CBAR
      KPOLAR=1./CLA
      CLAD=CLA*.0174533
      BETA=SQRT(ABS(MACH*MACH-1.0))
      BCLA=CLA*BETA
      IF (OUTCOD.LT.-1) GO TO 125
      WRITE (6,17) HEAD(NSURF),TITLE
      WRITE (6,712) MACH,BETA
      WRITE (6,713) SREF,CBAR,REFMOM,XY(6,PANELS),PANELS
  125 CONTINUE
  160 IF(CASE.LT.0) GO TO 500
      IF(CASE.EQ.1)GO TO 204
C-----INPUT VALUES OF CP ARE UPPER SURFACE CP,CONVERT TO DELTA-CP.
      DO 202 I=1,PANELS
  202 CP(I)=2.0*(CPT(I)-CP(I))
  204 DO 205 I=1,PANELS
  205 ALPHA(I)=0.
      IJ=0
      DO 210 J=1,PANELS
      DO 210 I=1,PANELS
      IJ=IJ+1
  210 ALPHA(I)=ALPHA(I)+A(IJ)*CP(J)
      GO TO 250
  500 CONTINUE
  605 IF(CASE+2)615,610,621
  610 XX=CL/CLA
      DO 611 I=1,PANELS
  611 ALPHA(I)=XX
  615 DO 616 I=1,PANELS
  616 ALPHA(I)=ALPHA(I)+AINC
  621 DO 625 I=1,PANELS
  625 CP(I)=-ALPHA(I)
      CALL SOLVE(CP,A,PANELS,NPL,IP)
  250 CL=0.
      CD=0.
      CPXBAR=0.
      T2=0.
      CPYBAR=0.
      DO 700 I=1,PANELS
```

```
      CPU(I)=CPT(I)-0.5*CP(I)                                           SP04 118
      CPL(I)=CPT(I)+0.5*CP(I)                                           SP04 119
      LG(I)=CP(I)*AREA(I)                                               SP04 120
      CL=CL+LG(I)                                                       SP04 121
      CD=CD+LG(I)*ALPHA(I)                                              SP04 122
      T2=T2+(F(I)*ALPHA(I)+CP(I))*AREA(I)                               SP04 123
      CPXBAR=CPXBAR+LG(I)*XBAR(I)                                       SP04 124
  700 CPYBAR=CPYBAR+LG(I)*YCPT(I)                                       SP04 125
      T2=T2/SHALF                                                       SP04 126
      IF(CL.EQ.0.)GO TO 701                                             SP04 127
      CPXBAR=CPXBAR/CL                                                  SP04 128
      CPYBAR=CPYBAR/CL                                                  SP04 129
  701 CL=CL/SHALF                                                       SP04 130
      CD=CD/SHALF+CDT+CDF                                               SP04 131
      CM=(REFMOM-CPXBAR)*CL/CBAR                                        SP04 132
      CMO=CM-CMCL*CL                                                    SP04 133
      CDO=CD-CL*(T2-CL)/CLA                                             SP04 134
      APOLAR=(T2-CL-CL)/CLA                                             SP04 135
      CLMIN=-.5*APOLAR/KPOLAR                                           SP04 136
      CDMIN=CDO+CLMIN*(APOLAR+KPOLAR*CLMIN)                             SP04 137
      CLOPT=SQRT(ABS(CDO*KPOLAR))                                       SP04 138
      CDOPT=CDO+CLOPT*(APOLAR+KPOLAR*CLOPT)                             SP04 139
      XLDOPT=CLOPT/CDOPT                                                SP04 140
      IF(CL.EQ.0.)GO TO 775                                             SP04 141
C                                                                       SP04 142
C      COMPUTE SPAN LOAD DISTRIBUTION                                   SP04 143
C                                                                       SP04 144
      I=0                                                               SP04 145
      M=1                                                               SP04 146
      CLCAV=(XY(6,PANELS)-XY(5,1))/(CL*SHALF)                           SP04 147
      TB=4.0*XY(6,PANELS)                                               SP04 148
      DO 705  J=1,REGION                                                SP04 149
      ROWS=ROWCOL(1,J)                                                  SP04 150
      COLS=ROWCOL(2,J)                                                  SP04 151
  703 DO 706 K=1,COLS                                                   SP04 152
      I=I+1                                                             SP04 153
      IMAX=I                                                            SP04 154
      YSLD(I)=YCPT(M)                                                   SP04 155
      WIDTH=XY(6,M)-XY(5,M)                                             SP04 156
      YWIDE(I)=WIDTH                                                    SP04 157
      SUM1=0.                                                           SP04 158
      SUM2=0.                                                           SP04 159
  704 DO 707 L=1,ROWS                                                   SP04 160
      SUM1=SUM1+LG(M)                                                   SP04 161
      SUM2=SUM2+AREA(M)                                                 SP04 162
  707 M=M+1                                                             SP04 163
      CLL=SUM1/SUM2                                                     SP04 164
  706 SLOAD(I)=SUM1/WIDTH/TB                                            SP04 165
  705 CONTINUE                                                          SP04 166
  775 CONTINUE                                                          SP04 167
      CLW=CL                                                            SP04 168
      IF (OUTCOD.LT.-1) GO TO 510                                       SP04 169
      WRITE (6,714)                                                     SP04 170
      WRITE (6,715) CL,CLA,CLAD,SCLA,CM,CMCL,AC,CPXBAR,CPYBAR           SP04 171
      IF (OUTCOD.GT.0) WRITE (6,20) CD,CDT,CDF,CMO,CDO,                 SP04 172
     1                           CDMIN,CLMIN,CLOPT,XLDOPT               SP04 173
      WRITE(6,710) TB,(YSLD(J),SLOAD(J),J=1,I)                          SP04 174
  510 CONTINUE                                                          SP04 175
C                                                                       SP04 176
C     COMPUTE POSITION AND STRENGTH OF TRAILING VORTEX                  SP04 177
C                                                                       SP04 178
      YTR=0.0                                                           SP04 179
      DO 50 J=1,IMAX                                                    SP04 180
   50 YTR=YTR+SLOAD(J)*YWIDE(J)                                         SP04 181
      YTR=YTR/SLOAD(1)                                                  SP04 182
      DUM=XY(6,PANELS)-XY(5,1)                                          SP04 183
      IF (YTR.GT.DUM .OR. YTR.LE.0.0) YTR=DUM*3.1415926/4.0             SP04 184
      GAMT=CL*SREF /(8.*3.1415926*YTR)                                  SP04 185
      YTR=YTR+XY(5,1)                                                   SP04 186
      IF (OUTCOD.LT.-1) RETURN                                          SP04 187
      WRITE (6,711) YTR,GAMT                                            SP04 188
      IF (OUTCOD.LE.1) RETURN                                           SP04 189
C                                                                       SP04 190
C      OUTPUT PROPERTIES OF INDIVIDUAL PANELS                           SP04 191
C                                                                       SP04 192
      WRITE(6,21)                                                       SP04 193
      WRITE(6,22)(I,ALPHA(I),DZDX(I),CPT(I),CP(I),CPU(I),CPL(I),LG(I),  SP04 194
     1    I=1,PANELS)                                                   SP04 195
  795 CONTINUE                                                          SP04 196
      RETURN                                                            SP04 197
      END                                                               SP04 198
```

```
      SUBROUTINE EDGFRC (NSURF,NR,NPL,NMAT,PANELS,REGION,SREF,          SP05 001
     1                  XY,YCPT,SLE,STE,DZDX,ROWCOL,DVSPAN,PNLC,TIPY,   SP05 002
     2                  SLOPE1,TOLRNC,CTIP,CROOT,ALPHA,NPRINT,          SP05 003
     3                  YSEP,NSEPV,AKVL1,AKVL2,AKVSF)                   SP05 004
C                                                                       SP05 005
C     THIS SUBROUTINE CALCULATES THE FORCES IN THE X AND Y DIRECTIONS   SP05 006
C     IN THE PLANE OF THE WING.                                         SP05 007
C     SUMFX---POINTS AFT, FORCE/(Q*SREF)                                SP05 008
C     SUMFY---POINTS TO STARBOARD, FORCE/(Q*SREF)                       SP05 009
C                                                                       SP05 010
      DIMENSION  CIRC(100), PNLRMC(100),PNLC(100),THOS(100),SWPMC(100)  SP05 011
     1           ,DLTCIR(100),FX(100),FY1(100),FY2(100),FT2(20),AKVLE(2) SP05 012
      DIMENSION XY(6,100),YCPT(NPL)                                     SP05 013
      DIMENSION SLE(NPL),STE(NPL),DZDX(NPL),ROWCOL(2,20),DVSPAN(NR)     SP05 014
      DIMENSION SECFX(20),SECFY1(20),SECFY2(20),SECSUC(20),ETA(20),     SP05 015
     1    YBAR(20),YLUC(20),THOSC(20)                                   SP05 016
      DIMENSION TIPSFC(20),ALPHA(NPL)                                   SP05 017
      DIMENSION SWPLE(20),MPNLS(20),SECFYS(20),SUMSCT(20)               SP05 018
      COMMON/PRESS/CP(100)                                              SP05 019
      COMMON/WVEL/WC4(100),WCT(100)                                     SP05 020
      COMMON/WNGFRC/CL                                                  SP05 021
      COMMON /RESLT1/ CNV1,CNV2,CNVS,Y1(2),GAM1(2),COPLE,COPTIP         SP05 022
C                                                                       SP05 023
      INTEGER       PANELS, ROWS,COLS,ROWCOL ,REGION,RM1,PNLSMR         SP05 024
C                                                                       SP05 025
    1 FORMAT (///19X,36HELEMENTAL PANEL GEOMETRY AND LOADING//5X,1HJ,   SP05 026
     14X,11HALPHA,TOTAL,4X,8HCENTROID,5X,6HR.H.S.,7X,5HPANEL,4X,        SP05 027
     220HMID CHORD,9X,7HGAMMA/V /                                       SP05 028
     313X,4HDEG.,9X,5HCHORD,8X,5HCHORD,8X,4HSPAN,4X,10HSWEEP,DEG.)      SP05 029
    2 FORMAT (3X,I3,2X,5(2X,F10.5),2X,2(5XF10.5,2XF10.5))               SP05 030
    3 FORMAT (///10X,37HSPANWISE DISTRIBUTION OF L.E. SUCTION/          SP05 031
     1   1X10HREGION COL,2X7HY/(B/2),4X6HCCS/2B,3X8HEPS(DEG),           SP05 032
     2    3X6HCCX/2B,4X6HCCY/2B)                                        SP05 033
    8 FORMAT (///5X,23HSIDE FORCE DISTRIBUTION //                       SP05 034
     1     11H REGION ROW,3X6HCYC/2B)                                   SP05 035
  701 FORMAT (2I5,2F10.5,F10.3,5F10.5)                                  SP05 036
C                                                                       SP05 037
      AKVLE(1)=AKVL1                                                    SP05 038
      AKVLE(2)=AKVL2                                                    SP05 039
      SSPAN=TIPY                                                        SP05 040
      CNST=SREF/(4.0*SSPAN)                                             SP05 041
      SUMFX=0.0                                                         SP05 042
      SUMFY1=0.0                                                        SP05 043
      SUMFY2=0.0                                                        SP05 044
      ROWS=ROWCOL(1,1)                                                  SP05 045
      CROOT=XY(3,ROWS)-XY(1,1)                                          SP05 046
      SWPLE(1)=SLE(1)                                                   SP05 047
      DO 100 JREG=1,REGION                                              SP05 048
      ROWS=ROWCOL(1,JREG)                                               SP05 049
      COLS=ROWCOL(2,JREG)                                               SP05 050
      MPNLS(JREG)= ROWS*COLS                                            SP05 051
      JREGM1= JREG-1                                                    SP05 052
      IF (JREG.GT.1)  MPNLS(JREG)=MPNLS(JREG)+MPNLS(JREGM1)             SP05 053
      JP1=MPNLS(JREGM1) + 1                                             SP05 054
      IF (JREG.GT.1) SWPLE(JREG)=SLE(JP1)                               SP05 055
C                                                                       SP05 056
C     CALCULATE PANEL SPAN---------------------------------THOS         SP05 057
C               PANEL CHORD AT MID PANEL SPAN--------------PNLC,SSWING  SP05 058
C               CHORD OF R.H. PANEL SIDE-------------------PNLRMC       SP05 059
C               SWEEP BACK OF PANEL MIDCHORD---------------SWPMC        SP05 060
C                                                                       SP05 061
C     CONVERT THE DELTA PRESSURE/Q OF A PANEL TO A LUMPED CIRCULATION   SP05 062
C     VALUE    CIRC=GAMMA/V                                             SP05 063
C                                                                       SP05 064
      DO 107 I=1,COLS                                                   SP05 065
```

```
      CIRNET=0,0
      DO 101 K=1,ROWS
      IF (JREG,EQ,1) GO TO 501
      GO TO 502
  501 J=(I-1)*ROWS+K
      JS=I*ROWS+K
      GO TO 503
  502 J=(I-1)*ROWS+K+MPNLS(JREGM1)
      JS=I*ROWS+K+MPNLS(JREGM1)
  503 IF (I,EQ,COLS,AND,JREG,EQ,REGION) GO TO 2010
      IF (I,EQ,1,AND,JREG,EQ,1) GO TO 4010
      GO TO 4011
 4010 CIRC(J)=(CP(J)*PNLC(J))/2,0
 4011 CIRC(JS)=(CP(JS)*PNLC(JS))/2,0
      DLTCIR (J)=CIRNET+CIRC(J)-CIRC(JS)
      GO TO 2011
 2010 DLTCIR(J)=CIRNET+CIRC(J)
 2011 TWOS(J)=XY(6,J)-XY(5,J)
      IF (K,EQ,ROWS) GO TO 2012
      GO TO 2013
 2012 PNLRMC(J)=0,5*(XY(4,J)-XY(2,J))
      GO TO 2014
 2013 PNLRMC(J)=XY(4,J)-XY(2,J)
 2014 SWPMC(J)=SLE(J)-0,5*(SLE(J)-STE(J))
      CIRNET=DLTCIR(J)
C
C     COMPUTE FX, FORCE ON ELEMENTAL PANEL IN X-DIRECTION,
C             FY, FORCE ON ELEMENTAL PANEL IN Y-DIRECTION,
C     FORCE/(Q*SREF)
C
      FX(J)=-(2,0*TWOS(J)/SREF)*CIRC(J)*( ALPHA(J)+WCT(J))
      FY1(J)=(2,0*TWOS(J)/SREF)*S*PMC(J)*CIRC(J)*( ALPHA(J)+WCT(J))
      FY2(J)=(2,0*PNLRMC(J)/SREF)*DLTCIR(J)*( ALPHA(J)+WC4(J))
      SUMFX=SUMFX+FX(J)
      SUMFY1=SUMFY1+FY1(J)
      IF (JREG,EQ,REGION ,AND, I,EQ,COLS) GO TO 102
      SUMFY2=SUMFY2+FY2(J)
      GO TO 101
  102 FY2(K)=FY2(J)
      FY2(J)=0,0
  101 CONTINUE
  107 CONTINUE
  100 CONTINUE
C
C     COEFFICIENT OF FORCE DETERMINED AS RESULTANT OF SUMFX AND SUMFY1
C        SUMSUC, ASSUMED TO ACT ON L,E,
C     COEFFICIENT OF FORCE DETERMINED AS RESULTANT OF SUMFX AND SUMFY1
C     AND SUMFY2 IS SMSCY2  (FORCE ON TRAILING LEGS)
C
      SUMSUC=SQRT(SUMFX*SUMFX+SUMFY1*SUMFY1)
      SMSCY2= SQRT(SUMFX**2+(SUMFY1+SUMFY2)**2)
C
C
C     NORMAL FORCE AUGMENTATION DETERMINED FROM L,E, VORTEX LIFT THEORY
C
      CNMD=CL
      CLMD= CNMD*COS(ALPHA(1))-2,0*SUMFX*SIN(ALPHA(1))
      CDI=CNMD*SIN(ALPHA(1))+2,0*SUMFX*COS(ALPHA(1))
      CDICLS= CDI/(CLMD*CLMD)
      CNADD=2,0*SMSCY2
C
      IF (NPRINT,LT,2) GO TO 117
C
C     OUTPUT PANEL ASSOCIATED PARAMETERS
C
      WRITE (6,1)
      DO 106 J=1,PANELS
      ALPHAD=ALPHA(J)*57,2957795
      DUM=1,0
      S*PMCD=(ATAN2(S*PMC(J),DUM))*57,2957795
  106 WRITE (6,2)J,ALPHAD,PNLC(J),PNLRMC(J),TWOS(J),S*PMCD,LIMC(J)
  117 CONTINUE
C
C     SPANWISE DISTRIBUTION OF WING PLANE FORCES
C     AND LEADING EDGE SUCTION CENTER OF PRESSURE
C
      IF (NPRINT,GE,0) WRITE (6,3)
      IMAX=0
      SUMSCT(1)=0,0
      XMOM=0,0
      DO 800 JREG=1,REGION
      ROWS= ROWCOL(1,JREG)
      COLS= ROWCOL(2,JREG)
      JREGM1= JREG-1
      DO 104 I=1,COLS
      SECFX(I)=0,0
      SECFY1(I)=0,0
      SECFY2(I)=0,0
      SECFY3(I)=0,0
      DO 103 K=1,ROWS
      IF (JREG,EQ,1) GO TO 801
      GO TO 802
  801 J=(I-1)*ROWS+K
      GO TO 803
  802 J=(I-1)*ROWS+K+MPNLS(JREGM1)
  803 SECFX(I)=SECFX(I)+(FX(J))*(CNST/TWOS(J))
      SECFY1(I)=SECFY1(I)+(FY1(J))*(CNST/TWOS(J))
      IF (I,EQ,1,AND,JREG,EQ,1) GO TO 3004
      IF (JREG,EQ,1) GO TO 804
      GO TO 805
  804 JS=(I-2)*ROWS+K
      GO TO 806
  805 JS = (I-2) * ROWS + K +MPNLS(JREGM1)
  806 SECFY2(I) = SECFY2(I) + FY2(J)*(CNST/TWOS(J))
      SECFY3(I)=SECFY3(I)+(FY1(J)+(FY2(J)+FY2(JS))/2,0)*(CNST/TWOS(J))
      GO TO 103
 3004 SECFY2(I) = SECFY2(I) + (FY2(J)) * (CNST/TWOS(J))
      SECFY3(I)=SECFY3(I)+(FY1(J)+FY2(J))*(CNST/TWOS(J))
  103 CONTINUE
C
      IMAX=IMAX+1
      IM1=IMAX-1
      IF (IM1,EQ,0) IM1=1
      SECSUC(IMAX)=SQRT(SECFX(I)*SECFX(I)+SECFY3(I)*SECFY3(I))
C          SECSUC = C CS/2 B   LOADING COEFFICIENT
      SUCT=SECSUC(IMAX)*(TWOS(J)/CNST)
      TWOSC(IMAX)=TWOS(J)
      JM=1
      YLOC(IMAX)=((XY(6,J)+XY(5,J))/2,0)/SSPAN
      DUM=1,0
      ETA(IMAX)=(ATAN2( SWPLE(JREG),DUM)-
     1                 ATAN2(SECFX(I),SECFY3(I)))*57,2957795
      IF (NSEPV,GT,1 ,AND, YSEP,LT,YLOC(IMAX)*SSPAN) JM=2
      IF (AKVLE(JM),LE,0,0) GO TO 70
      XMOM=XMOM + SUCT*YCPT(J)*AKVLE(JM)
      SUMSCT(IMAX)=SUMSCT(IM1)+SUCT*AKVLE(JM)
      YBAR(IMAX)=(XMOM/SUMSCT(IMAX))
      GO TO 71
   70 YBAR(IMAX)=XY(5,1)
   71 CONTINUE
      IF (YBAR(IMAX),GT,XY(6,PANELS)) YBAR(IMAX)=XY(6,PANELS)
      IF (YBAR(IMAX),LT,XY(5,1)) YBAR(IMAX)=XY(5,1)
C
      IF (NPRINT,GE,0) WRITE (6,701) JREG,I,YLOC(I),SECSUC(IMAX),
     1                 ETA(IMAX),SECFX(I),SECFY3(I)
  104 CONTINUE
  800 CONTINUE
C
      COPLE=YBAR(IMAX)
      DO 20 I=1,PANELS
      J=I
      IF (XY(6,I) -COPLE) 20,25,21
   20 CONTINUE
   25 COPLE=XY(2,J)
      GO TO 30
   21 DELTA=(COPLE-XY(5,J))/(XY(6,J)-XY(5,J))
      COPLE=XY(1,J) + DELTA*(XY(2,J)-XY(1,J))
   30 CONTINUE
C
C     TIP EDGE FORCE DISTRIBUTION AND CENTER OF PRESSURE
C
      ROWS= ROWCOL (1,REGION)
      RM1= REGION-1
      AROWS = ROWS
```

```
      SFM=0.0
      SF=0.0
      COPTIP=0.0
      YMOM=0.0
      TIPFY=0.0
      TOL=(1.E-2)*CROOT
      PNLSMR=PANELS-ROWS+1
      TIPCHD=XY(4,PANELS)-XY(2,PNLSMR)
      IF (TIPCHD.LT.TOL) GO TO 667
      TIPEPL=TIPCHD/AROWS
      IF (NPRINT.GE.0) WRITE (6,8)
      DO 662 JTIP=1,ROWS
      AJTIP=JTIP
      XLOC=AJTIP*TIPEPL
      TIPFY=TIPFY+FT2(JTIP)
      TIPSFC(JTIP)=(FT2(JTIP)/TIPEPL)*CNST
      YMOM=YMOM + (FT2(JTIP)*XLOC)
      XLOCOC=XLOC/TIPCHD
      SFM=SFM + TIPSFC(JTIP)*TIPEPL
      IF (NPRINT.GE.0) WRITE (6,701)REGION,JTIP,TIPSFC(JTIP)
  662 CONTINUE
      COPTIP=(YMOM/TIPFY) + XY(2,PNLSMR)
  667 CONTINUE
C
C      COMPUTE POSITION AND STRENGTH OF L.E. SEPARATION VORTICES
C      TIP VORTEX IS INCLUDED WITH OUTER REGION L.E. VORTEX
C
      RAVG=XY(5,1)
      Y1(1)=0.0
      Y1(2)=0.0
      GAM1(1)=0.0
      GAM1(2)=0.0
      CNV2=0.0
      CNVS=2.0*TIPFY
      CNV=CNADD
      YBD=0.0
      KSEP=IMAX
      SF=SFM
      SFM=SFM*(TIPY-RAVG)
      ESEP=YSEP/SSPAN
      DO 300 I=1,IMAX
      IF (ESEP.GT.YLOC(I)) KSEP=I
      YBD=YBD + SECSUC(I)*THOSC(I)
  300 CONTINUE
      II=1
      DO 303 J=1,NSEPV
      DUM=0.0
      IF (KSEP.EQ.IMAX) DUM=1.0
      Y1D=0.0
      DO 301 I=II,KSEP
      YETA=YLOC(I)*SSPAN - RAVG
      Y1(J)=Y1(J) + SECSUC(I)*YETA*THOSC(I)
      Y1D=Y1D + SECSUC(I)*THOSC(I)
  301 CONTINUE
      CNV1=CNV
      IF (KSEP.LT.IMAX) CNV2=Y1D/YBD
      IF (KSEP.LT.IMAX) II=KSEP+1
      KSEP=IMAX
      IF (AKVLE(J).LE.0.0 .AND. DUM*AKVSE.LE.0.0) GO TO 313
      GAM1(J)=AKVLE(J)*CNV*Y1D/YBD + DUM*AKVSE*CNVS
      Y1(J)=(Y1(J)*AKVLE(J) + DUM*AKVSE*SFM)/
     1                            (Y1D*AKVLE(J) + DUM*AKVSE*SF)
      GO TO 303
  313 Y1(J)=SSPAN
  303 CONTINUE
      IF (NSEPV.LE.1) RETURN
      CNV1=CNV*CNV2
      CNV2=CNV-CNV1
      RETURN
      END
```

```
      SUBROUTINE MATRIX (NSURF,NR,NPL,NMAT,A,XY,XCPT,YCPT,XBAR,SLE,
     1                  STE,DZDX,PANELS,MACH,TOLRNC,IP,UT)
C-----THIS ROUTINE COMPUTES AND STORES THE AERODYNAMIC MATRIX AND ALSO
C-----COMPUTES UT, THE ARRAY OF VELOCITIES (X-COMPONENTS) INDUCED
C-----BY WING THICKNESS.
      DIMENSION XY(6,100),XCPT(NPL),YCPT(NPL),XBAR(NPL)
      DIMENSION SLE(NPL),STE(NPL),DZDX(NPL),UT(NPL),A(NMAT),IP(NPL)
      COMMON /MAT1/ X,XT,Y,B,BTERM,RESULT,UTHICK,SUB
      COMMON /PARAMS/ ASYM
      REAL MACH
      LOGICAL SUB,ASYM,THICK
      INTEGER PANELS
      DATA PI/3.141593/
      DATA ZERO/0.0/
      BETA=SQRT(ABS(MACH*MACH-1.0))
      BETA1=1.0/BETA
      SUB=MACH.LT.1.
      IF(SUB)GO TO 5
      XTERM=-1.0D0
      CONS=BETA/(4.0*PI)
      TCONS=1.0/(PI*BETA)
      GO TO 6
    5 XTERM=1.D0
      CONS=BETA/(8.0*PI)
      TCONS=0.5/(PI*BETA)
    6 CONTINUE
C-----ZERO THE UT ARRAY
      DO 10 I=1,PANELS
   10 UT(I)=0.
C     BEGIN GRAND LOOP
      JJ=0
      SUMT=ZERO
      DO 900 I=1,PANELS
C   I IS INDEX OF INFLUENCING PANEL
      B1=SLE(I)*BETA1
      B2=STE(I)*BETA1
      BTERM2= SQRT( ABS(B2*B2+XTERM))
      BTERM1= SQRT( ABS(B1*B1+XTERM))
      X1=XY(1,I)*BETA1
      X2=XY(2,I)*BETA1
      X3=XY(3,I)*BETA1
      X4=XY(4,I)*BETA1
      Y1=XY(5,I)
      Y2=XY(6,I)
      THICK=DZDX(I).NE.0.
C-----BEGIN INNER LOOP.  J IS THE INDEX OF THE INFLUENCED PANEL
      DO 850 J=1,PANELS
      JJ=JJ+1
      XC=XCPT(J)*BETA1
      YC=YCPT(J)
      XCT=XBAR(J)*BETA1
      BTERM=BTERM1
      IF (B1.LT.ZERO) GO TO 820
      B=B1
      X=XC-X1
      Y=YC-Y1
      CALL COMP (TOLRNC)
      SUM=RESULT
      IF(.NOT.THICK)GO TO 11
      XT=XCT-X1
      CALL TCOMP
      SUMT=UTHICK
   11 IF(ASYM)GO TO 12
      Y=-YC-Y1
      CALL COMP (TOLRNC)
      SUM=SUM+RESULT
      IF(.NOT.THICK)GO TO 12
      CALL TCOMP
      SUMT=SUMT+UTHICK
   12 X=XC-X2
      Y=YC-Y2
      CALL COMP (TOLRNC)
      SUM=SUM-RESULT
      IF(.NOT.THICK)GO TO 13
      XT=XCT-X2
      CALL TCOMP
      SUMT=SUMT-UTHICK
```

```
 13 IF(ASYM)GO TO 825
    Y=-YC-Y2
    CALL COMP (TOLRNC)
    SUM=SUM-RESULT
    IF(.NOT.THICK)GO TO 825
    CALL TCOMP
    SUMT=SUMT-UTHICK
    GO TO 825
820 B=-B1
    X=XC-X1
    Y=Y1-YC
    CALL COMP (TOLRNC)
    SUM=-RESULT
    IF(.NOT.THICK)GO TO 15
    XT=XCT-X1
    CALL TCOMP
    SUMT=-UTHICK
 15 IF(ASYM)GO TO 16
    Y=Y1+YC
    CALL COMP (TOLRNC)
    SUM=SUM-RESULT
    IF(.NOT.THICK)GO TO 16
    CALL TCOMP
    SUMT=SUMT-UTHICK
 16 X=XC-X2
    Y=Y2-YC
    CALL COMP (TOLRNC)
    SUM=SUM+RESULT
    IF(.NOT.THICK)GO TO 17
    XT=XCT-X2
    CALL TCOMP
    SUMT=SUMT+UTHICK
 17 IF(ASYM)GO TO 825
    Y=Y2+YC
    CALL COMP (TOLRNC)
    SUM=SUM+RESULT
    IF(.NOT.THICK)GO TO 825
    CALL TCOMP
    SUMT=SUMT+UTHICK
825 BTERM=BTERM2
    IF (B2.LT.ZERO) GO TO 830
    B=B2
    X=XC-X3
    Y=YC-Y1
    CALL COMP (TOLRNC)
    SUM=SUM-RESULT
    IF(.NOT.THICK)GO TO 21
    XT=XCT-X3
    CALL TCOMP
    SUMT=SUMT-UTHICK
 21 IF(ASYM)GO TO 22
    Y=-YC-Y1
    CALL COMP (TOLRNC)
    SUM=SUM-RESULT
    IF(.NOT.THICK)GO TO 22
    CALL TCOMP
    SUMT=SUMT-UTHICK
 22 X=XC-X4
    Y=YC-Y2
    CALL COMP (TOLRNC)
    SUM=SUM+RESULT
    IF(.NOT.THICK)GO TO 23
    XT=XCT-X4
    CALL TCOMP
    SUMT=SUMT+UTHICK
 23 IF(ASYM)GO TO 840
    Y=-YC-Y2
    CALL COMP (TOLRNC)
    SUM=SUM+RESULT
    IF(.NOT.THICK)GO TO 840
    CALL TCOMP
    SUMT=SUMT+UTHICK
    GO TO 840
830 B=-B2
    X=XC-X3
    Y=Y1-YC
    CALL COMP (TOLRNC)
    SUM=SUM+RESULT
    IF(.NOT.THICK)GO TO 25
    XT=XCT-X3
    CALL TCOMP
    SUMT=SUMT+UTHICK
 25 IF(ASYM)GO TO 26
    Y=Y1+YC
    CALL COMP (TOLRNC)
    SUM=SUM+RESULT
    IF(.NOT.THICK)GO TO 26
    CALL TCOMP
    SUMT=SUMT+UTHICK
 26 X=XC-X4
    Y=Y2-YC
    CALL COMP (TOLRNC)
    SUM=SUM-RESULT
    IF(.NOT.THICK)GO TO 28
    XT=XCT-X4
    CALL TCOMP
    SUMT=SUMT-UTHICK
 28 IF(ASYM)GO TO 840
    Y=Y2+YC
    CALL COMP (TOLRNC)
    SUM=SUM-RESULT
    IF(.NOT.THICK)GO TO 840
    CALL TCOMP
    SUMT=SUMT-UTHICK
840 UT(J)=UT(J)+SUMT*TCONS*DZDX(I)
850 A(JJ)=CONS*SUM
900 CONTINUE
C-----MATRIX A WITH PANELS*PANELS ENTRIES IS NOW COMPUTED.
C-----NOW THE L-U DECOMPOSITION OF A IS PERFORMED.
      CALL LINEQS(PANELS,A,NPL,IP)
      RETURN
      END
```

```
      SUBROUTINE CNVTX (XINIT,XFINL,NSEC,XM,CN,CM)
C
C      COMPUTE THE FREE VORTEX INDUCED NORMAL FORCE AND MOMENT
C
      COMPLEX SIGMA,SIGMAB
      DIMENSION XBDY(100),RBDY(100),SBDY(100),DRBDY(100)
      DIMENSION XV(200),YV(4,200),ZV(4,200),NVP(200),ARAY(5,200)
      DIMENSION GAM(4),YG(4),ZG(4),SIGMA(4,2),SIGMAB(2),A(2)
C
      COMMON /BODY/ NTBL,MTBL,XBDY,RBDY,SBDY,DRBDY
      COMMON /VORTEX/ NV,NV2,GAM,YG,ZG
      COMMON /VTABLE/ NTRY,XV,YV,ZV,NVP,ARAY
C
      DUM=NSEC
      DX=(XFINL-XINIT)/DUM
      XI=XINIT
      K=1
      CN=0.0
      CM=0.0
      MN=1
      IN=1
      DO 30 J=1,NSEC
      DO 10 NK=K,2
      NN=MN
      DO 11 M=NN,NTRY
      MN=M
      IF (XI-XV(M)) 13,12,11
   11 CONTINUE
   12 DO 112 NVN=1,NV
      Y=YV(NVN,MN)
      Z=ZV(NVN,MN)
  112 SIGMA(NVN,NK)=CMPLX(Y,Z)
      GO TO 17
   13 IF (MN.EQ.1) GO TO 12
      DELTA=(XI-XV(MN-1))/(XV(MN)-XV(MN-1))
      DO 113 NVN=1,NV
      Y=YV(NVN,MN-1) + DELTA*(YV(NVN,MN)-YV(NVN,MN-1))
      Z=ZV(NVN,MN-1) + DELTA*(ZV(NVN,MN)-ZV(NVN,MN-1))
  113 SIGMA(NVN,NK)=CMPLX(Y,Z)
   17 IF (MN.GT.1) MN=MN-1
C
C      LOOK UP LOCAL BODY RADIUS
C
      JN=IN
      DO 14 I=JN,NTBL
      IN=I
      IF (XI-XBDY(I)) 16,15,14
   14 CONTINUE
   15 A(NK)=RBDY(IN)
      GO TO 18
   16 IF (IN.EQ.1) GO TO 15
      DELTA=(XI-XBDY(IN-1))/(XBDY(IN)-XBDY(IN-1))
      A(NK)=RBDY(IN-1)+DELTA*(RBDY(IN)-RBDY(IN-1))
   18 IF (IN.GT.1) IN=IN-1
   10 XI=XI+DX
      K=2
      XP=XI-DX-DX/2.
C
C      COMPUTE NORMAL FORCE AND MOMENT ON ONE SECTOR OF BODY
C
      CND=0.0
      DO 20 NVN=1,NV
      DO 21 NK=1,2
   21 SIGMAB(NK)=SIGMA(NVN,NK)-A(NK)*A(NK)/CONJG(SIGMA(NVN,NK))
      CND=CND+4.0*GAM(NVN)*REAL(SIGMAB(2)-SIGMAB(1))
   20 CONTINUE
      CM=CM+CND*(XM-XP)
      CN=CN+CND
      A(1)=A(2)
      DO 22 NVN=1,NV
   22 SIGMA(NVN,1)=SIGMA(NVN,2)
   30 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE BODYVTX (ALPHAD,THETAN,FINE,XA,EMACH,GAMN,NG,
     1                   XGAMN,YGAMN,ZGAMN,XSA)
C
C      COMPUTE THE STRENGTH AND POSITION OF BODY VORTICES
C
      DIMENSION XST(11),ZA1(11),YA1(11),YA2(11),GAMT(11)
      DIMENSION XGAMN(10),YGAMN(10),ZGAMN(10),GAMN(10)
C
      DATA XST/0.0,1.0,2.0,3.0,4.0,5.0,6.0,7.0,8.0,9.0,10.0/
      DATA GAMT/0.3,0.32,0.34,0.40,0.48,0.62,0.77,0.90,1.0,1.08,1.15/
      DATA YA1/.45,.463,.477,.492,.505,0.52,.534,0.55,0.565,0.575,0.585/
      DATA YA2/.63,.64,.653,.665,.678,.69,.7,.715,.725,.735,.74/
      DATA ZA1/1.14,1.26,1.38,1.5,1.615,1.73,1.84,1.95,2.05,2.14,2.20/
C
  701 FORMAT (//1X5H*****,3X62HASYMMETRIC OR UNSTEADY BODY VORTEX SEPARA
     1TION POSSIBLE    *****//)
C
      NTABLE=11
      DELTA=0.0
      F=1.0
      IF (ALPHAD.LT.0.0) F=-1.0
      ALPHA=F*ALPHAD
C
      IF (THETAN.LE.30.0) GO TO 10
      XSA=10./(ALPHA -4.0) + 2.0
      GO TO 13
   10 XSA=32.0 - SQRT(1024.0*(ALPHA -4.0)/(THETAN-4.0))
   13 IF (XSA.LT.0.0) XSA=0.0
      NG=0
      IF (XSA.GE.XA) RETURN
C
C      COMPUTE UPPER LIMIT FOR SYMMETRIC VORTEX SEPARATION
C
   12 ALMT=((FINE-12.0)**2)/3.57 + 12.0
      IF (ALPHA .GT.ALMT) WRITE (6,701)
      SNALP=SIN(ALPHAD/57.2957795)*F
      DXPR=XA-XSA
      IF (DXPR.LT. 1.0E-02) GO TO 15
      XPAR=DXPR/2.0
      NG=XPAR
      IF (NG.GT.10) NG=10
      IF (NG.LE.1) NG=2
      GO TO 16
   15 NG=1
      DXPR=0.0
      XPAR=0.0
   16 CONTINUE
      DO 40 JG=1,NG
      DUM1=JG
      DUM2=NG
      DUM=0.0
      IF (NG.GT.1) DUM=(DUM1-1.0)/(DUM2-1.0)
      XPAR=XSA + DUM*DXPR
      IF (NG.EQ.1) XPAR=XA
      XGAMN(JG)=XPAR
      XPAR=(XPAR-XSA)*SNALP
      IF (XPAR .LT. 0.0) XPAR=0.0
      DO 20 J=1,NTABLE
      K=J
      IF (XPAR-XST(J)) 22,23,20
   20 CONTINUE
   23 GAMN(JG)=GAMT(K)*SNALP
      ZGAMN(JG)=ZA1(K)
      Y1=YA1(K)
      Y2=YA2(K)
      GO TO 30
   22 DELTA=(XPAR-XST(K-1))/(XST(K)-XST(K-1))
      GAMN(JG)=GAMT(K-1) + DELTA*(GAMT(K)-GAMT(K-1))
      GAMN(JG)=GAMN(JG)*SNALP
      Y1=YA1(K-1) + DELTA*(YA1(K)-YA1(K-1))
      ZGAMN(JG)=ZA1(K-1) + DELTA*(ZA1(K)-ZA1(K-1))
      Y2=YA2(K-1) + DELTA*(YA2(K)-YA2(K-1))
   30 YGAMN(JG)=Y1
      IF (EMACH .GT. 1.0) YGAMN(JG)=Y2
      GAMN(JG)=GAMN(JG)*F
   40 ZGAMN(JG)=ZGAMN(JG)*F
      RETURN
      END
```

```
      SUBROUTINE  CNVNZ (XGAMN,YGAMN,ZGAMN,GAMN,NG,XM,CN,CM)
C
C     COMPUTE NORMAL FORCE AND MOMENT ON NOSE DUE TO SHED BODY VORTEX
C
      COMPLEX SIGMA,SIGMAB
      DIMENSION XGAMN(10),YGAMN(10),ZGAMN(10),GAMN(10)
      DIMENSION A(2),SIGMA(2),SIGMAB(2)
C
      CN=0.0
      CND=0.0
      CM=0.0
      IF (NG.LE.1) RETURN
      XI=XGAMN(1)
      CALL SHAPE (XI,A(1),DUM,DUM)
      SIGMA(1)=CMPLX(YGAMN(1),ZGAMN(1))
      SIGMAB(1)=CMPLX(0.0,0.0)
      DUM=(YGAMN(1))**2 + (ZGAMN(1))**2
      IF (DUM.GT.0.0) SIGMAB(1)=SIGMA(1)-A(1)*A(1)/CONJG(SIGMA(1))
      DO 50 J=2,NG
      XI=XGAMN(J)
      SIGMA(2)=CMPLX(YGAMN(J),ZGAMN(J))
      CALL SHAPE (XI,A(2),DUM,DUM)
      SIGMAB(2)=SIGMA(2)-A(2)*A(2)/CONJG(SIGMA(2))
      CND=8.0*GAMN(J)*REAL(SIGMAB(2)-SIGMAB(1))
      CN=CN+CND
      XP=0.5*(XGAMN(J-1)+XGAMN(J))
      CM=CM+CND*(XM-XP)
      A(1)=A(2)
      SIGMA(1)=SIGMA(2)
      SIGMAB(1)=SIGMAB(2)
   50 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SHAPE (X,A,B,DB)
C
C      TABLE LOOK-UP OF BODY COORDINATES
C
      DIMENSION XBDY(100),RBDY(100),SBDY(100),DRBDY(100)
      COMMON /BODY/ NTBL,MTBL,XBDY,RBDY,SBDY,DRBDY
C
  700 FORMAT (///5X12H*****  XBDY=,F10.4,2X2HX=, F10.4,47H, X OUTSIDE RA
     1NGE OF TABLE OF BODY COORDINATES /40X,23HEXECUTION STOP IN SHAPE )
      NT=NTBL+1
    9 DO 10 J=MTBL,NT
      K=J
      IF (X-XBDY(J)) 12,13,10
   10 CONTINUE
   11 WRITE (6,700)    XBDY(K),X
      STOP
   13 A=RBDY(K)
      B=SBDY(K)
      DB=DRBDY(K)
      GO TO 40
   12 IF (K.LE.1) GO TO 11
      IF (X.LE.XBDY(K-1)) GO TO 120
      DEL=(X-XBDY(K-1))/(XBDY(K)-XBDY(K-1))
      A=RBDY(K-1) + DEL*(RBDY(K)-RBDY(K-1))
      B=SBDY(K-1) + DEL*(SBDY(K)-SBDY(K-1))
      DB=DRBDY(K-1) + DEL*(DRBDY(K)-DRBDY(K-1))
   40 CONTINUE
      IF (B.LT.A) B=A
      IF (K.GT.2) MTBL=K-2
      RETURN
  120 MTBL=1
      GO TO 9
      END
```

```
      SUBROUTINE FILL(AFILL,NFILL)
      REAL AFILL(1)
C     FILLS INTERMEDIATE LOCATION IN AN ARRAY
      IF(NFILL.LE.1)RETURN
      DEL=(AFILL(NFILL+1)-AFILL(1))/FLOAT(NFILL)
      DO 20 I=2,NFILL
   20 AFILL(I)=AFILL(I-1)+DEL
      RETURN
      END
```

```
      SUBROUTINE ZSECT (X,Z,DZDX,ISECT)
C
C     GENERAL AIRFOIL SECTION CALCULATION ROUTINE
C       X IS THE FRACTION OF CHORD
C       ISECT IS THE VARIABLE WHICH DETERMINES THE CHOICE OF AIRFOIL
C
C       ISECT=1   PARABOLIC ARC
C       ISECT=2   DOUBLE WEDGE (DIAMOND)
C       ISECT=3   30-70 HEX
C       ISECT=3   WEDGE  (THICK BASE)
C       ISECT=5   NACA 000X
C       ISECT=6   NACA 6400X
C       ISECT=7   NACA 6500X
C       ISECT=8   RAE 101
C
      DIMENSION XTAB(19),ZTAB1(19),ZPTAB1(19),ZTAB2(19),ZPTAB2(19),
     1 ZTAB3(19),ZPTAB3(19),ZTAB4(19),ZPTAB4(19)
      DATA     XTAB/0.,.005,.0125,.0250,.05,.075,.1,.15,.2,.25,.3,.4,
     X  .5,.6,.7,.8,.9,.95,1./
      DATA    ZTAB1/0.,.0925,.1578,.2178,.2962,.35,.3902,.4455,.4782,
     1  .4952,.5002,.4837,.4412,.3803,.3053,.2187,.1207,.0672,.0105/
      DATA     ZPTAB1/18.46,12.624,6.275,3.69,2.644,1.88,1.273,.88,.497,
     X .22,-.0767,-.295,-.517,-.6795,-.808,-.923,-1.01,-1.102,-1.134/
      DATA    ZTAB2/0.,.082,.125,.1701,.2343,.2826,.3221,.3842,.4302,
     1 .4639,.4864,.4988,.4586,.3820,.2827,.1722,.0671,.0248,0./
      DATA     ZPTAB2/16.4,10.,4.405,2.915,2.25,1.756,1.354,1.081,.797,
     X .562,.2327,-.139,-.584,-.8795,-1.049,-1.078,-.983,-.671,-.496/
      DATA     ZTAB3/0.,.0772,.1169,.1574,.2177,.2647,.3040,.3666,.4143,
     X  .4503,.4760,.4996,.4812,.4146,.3156,.1987,.0810,.0306,0./
      DATA     ZPTAB3/15.44,9.352,4.01,2.688,2.146,1.726,1.359,1.103,
     X  .837,.617,.329,.026,-.425,-.828,-1.079,-1.175,-1.121,-.81,-.612/
      DATA     ZTAB4/0.,.087,.1335,.1915,.2659,.3191,.3607,.4220,.4630,
     X .4885,.4997,.4801,.4267,.3531,.2681,.1789,.0894,.0447,0./
      DATA     ZPTAB4/22.,10.5,5.32,3.71,2.446,1.863,1.465,1.000,.6589,
     X .3725,.0464,-.3966,-.6529,-.8073,-.8837,4*-.894/
C
C
C
      IF (X.LT.0..OR.X.GT.1.) GO TO 980
      IF (ISECT.GT.4) GO TO 200
      IF (ISECT.GT.1) GO TO 110
  100 Z=2.*X*(1.-X)
      DZDX=2.*(1.-2.*X)
      RETURN
  110 IF (ISECT.EQ.3) GO TO 2
      IF (ISECT.EQ.4) GO TO 3
    1 IF(X.GT..5)GO TO 101
      Z=X
      DZDX=1.
      RETURN
  101 Z=1.-X
      DZDX=-1.
      RETURN
    2 IF(X.GE..3)GO TO 111
      Z=1.666667*X
      DZDX=1.666667
      RETURN
  111 IF(X.GT..7)GO TO 112
      Z=.5
      DZDX=0.
      RETURN
  112 Z=1.666667*(1.-X)
      DZDX=-1.666667
```

```
      RETURN                                                            SP12 064
    3 Z=X/2.                                                            SP12 065
      DZDX=.5                                                           SP12 066
      RETURN                                                            SP12 067
  200 I=ISECT-4                                                         SP12 068
      IF (I.GT.4) GO TO 980                                             SP12 069
      GO TO(201,202,203,204),I                                          SP12 070
  201 CALL TAINT(XTAB,ZTAB1,X,Z,19,2)                                   SP12 071
      CALL TAINT(XTAB,ZPTAB1,X,DZDX,19,2)                               SP12 072
      RETURN                                                            SP12 073
  202 CALL TAINT(XTAB,ZTAB2,X,Z,19,2)                                   SP12 074
      CALL TAINT(XTAB,ZPTAB2,X,DZDX,19,2)                               SP12 075
      RETURN                                                            SP12 076
  203 CALL TAINT(XTAB,ZTAB3,X,Z,19,2)                                   SP12 077
      CALL TAINT(XTAB,ZPTAB3,X,DZDX,19,2)                               SP12 078
      RETURN                                                            SP12 079
  204 CALL TAINT(XTAB,ZTAB4,X,Z,19,2)                                   SP12 080
      CALL TAINT(XTAB,ZPTAB4,X,DZDX,19,2)                               SP12 081
      RETURN                                                            SP12 082
C-----YOU MUST BE OFF THE WING OR HAVE ENTERED AN ILLEGAL SECTION NUMBERSP12 083
  980 Z=0.                                                              SP12 084
      DZDX=0.                                                           SP12 085
      RETURN                                                            SP12 086
      END                                                               SP12 087
```

```
      SUBROUTINE TCOMP                                                  SP13 001
C-----ROUTINE TO COMPUTE PRESSURE FUNCTION FOR A WEDGE OF TRIANGULAR    SP13 002
C-----PLANFORM IN SUBSONIC OR SUPERSONIC FLOW.                          SP13 003
C-----(XT,Y)= COORDINATES OF FIELD POINT RELATIVE TO ORIGIN OF TRIANGLE SP13 004
C-----B IS SLOPE (DX/DY) OF LEADING EDGE.                               SP13 005
C-----BTERM  (SEE SUBROUTINE MATRIX)                                    SP13 006
C-----UTHICK IS THE RATIO OF INDUCED VELOCITY TO FREESTREAM VELOCITY    SP13 007
C-----SUB=.TRUE. IF MACH LT 1                                           SP13 008
      COMMON /MAT1/ X,XT,Y,B,BTERM,RESULT,UTHICK,SUB                    SP13 009
      LOGICAL SUB                                                       SP13 010
      DATA ZERO/0.0/,EPS/1.E-5/,ONE/1.0/,PI/3.141593/                   SP13 011
      IF(SUB)GO TO 400                                                  SP13 012
      IF(XT.LE.ZERO)GO TO 350                                           SP13 013
      X2=XT*XT                                                          SP13 014
      Y2=Y*Y                                                            SP13 015
  525 IF(B.GT.ONE)GO TO 560                                             SP13 016
C-----SUPERSONIC FLOW, SUPERSONIC LEADING EDGE.                         SP13 017
      IF(X2.GT.Y2)GO TO 550                                             SP13 018
C-----INSIDE MACH CONE.                                                 SP13 019
      IF(Y.LE.EPS.OR.XT.LE.B*Y)GO TO 350                                SP13 020
      UTHICK=-PI/BTERM                                                  SP13 021
      GO TO 300                                                         SP13 022
C-----OUTSIDE MACH CONE.                                                SP13 023
  550 UTHICK=- ATAN2(BTERM* SQRT(X2-Y2),B*XT-Y)/BTERM                   SP13 024
      GO TO 300                                                         SP13 025
C-----SUPERSONIC FLOW,  SUBSONIC LEADING EDGE.                          SP13 026
  560 IF (X2.LE.Y2)GO TO 350                                            SP13 027
      UTHICK=-ALOG((B*XT-Y+BTERM* SQRT(X2-Y2))/ ABS(XT-B*Y))/BTERM      SP13 028
      GO TO 300                                                         SP13 029
C-----SUBSONIC FLOW                                                     SP13 030
  400 AA=(B*XT+Y+BTERM* SQRT(XT*XT+Y*Y))/ ABS(XT-B*Y)                   SP13 031
      IF(AA.LE.ZERO)GO TO 350                                           SP13 032
      UTHICK=-ALOG(AA)/BTERM                                            SP13 033
      GO TO 300                                                         SP13 034
  350 UTHICK=ZERO                                                       SP13 035
  300 RETURN                                                            SP13 036
      END                                                               SP13 037
```

```
      SUBROUTINE TAINT(XTAB,FTAB,X,FX,N,K)                              SP14 001
      DIMENSION XTAB(1),FTAB(1),T(10),C(10)                             SP14 002
      IF (N - K) 1,1,2                                                  SP14 003
    1 NER=2                                                             SP14 004
      RETURN                                                            SP14 005
    2 IF (K-9) 3,3,1                                                    SP14 006
    3 IF ( MON) 4,4,5                                                   SP14 007
    5 IF ( MON-2) 6,7,4                                                 SP14 008
    4 J=0                                                               SP14 009
      NM1=N-1                                                           SP14 010
      DO 8 I=1,NM1                                                      SP14 011
      IF (XTAB(I)-XTAB(I+1)) 9,11,10                                    SP14 012
   11 NER=3                                                             SP14 013
      RETURN                                                            SP14 014
    9 J=J-1                                                             SP14 015
      GO TO 8                                                           SP14 016
   10 J=J+1                                                             SP14 017
    8 CONTINUE                                                          SP14 018
      MON=1                                                             SP14 019
      IF (J) 12,6,6                                                     SP14 020
   12 MON=2                                                             SP14 021
    7 DO 13 I=1,N                                                       SP14 022
      IF (X-XTAB(I)) 14,14,13                                           SP14 023
   14 J=I                                                               SP14 024
      GO TO 18                                                          SP14 025
   13 CONTINUE                                                          SP14 026
      GO TO 15                                                          SP14 027
    6 DO 16 I=1,N                                                       SP14 028
      IF(X-XTAB(I)) 16,17,17                                            SP14 029
   17 J=I                                                               SP14 030
      GO TO 18                                                          SP14 031
   16 CONTINUE                                                          SP14 032
   15 J=N                                                               SP14 033
   18 J=J-(K+1)/2                                                       SP14 034
      IF (J) 19,19,20                                                   SP14 035
   19 J=1                                                               SP14 036
   20 M=J+K                                                             SP14 037
      IF (M-N) 21,21,22                                                 SP14 038
   22 J=J-1                                                             SP14 039
      GO TO 20                                                          SP14 040
   21 KP1=K+1                                                           SP14 041
      JSAVE=J                                                           SP14 042
      DO 23 L=1,KP1                                                     SP14 043
      C(L)=X-XTAB(J)                                                    SP14 044
      T(L)=FTAB(J)                                                      SP14 045
   23 J=J+1                                                             SP14 046
      DO 24 J=1,K                                                       SP14 047
      I=J+1                                                             SP14 048
   25 T(I)=(C(J)*T(I)-C(I)*T(J))/(C(J)-C(I))                            SP14 049
      I=I+1                                                             SP14 050
      IF (I-KP1) 25,25 ,24                                              SP14 051
   24 CONTINUE                                                          SP14 052
      FX=T(KP1)                                                         SP14 053
      NER=1                                                             SP14 054
      RETURN                                                            SP14 055
      END                                                               SP14 056
```

```
      SUBROUTINE LINEQS(N,A,NPL,IP)
      DIMENSION A(N,N),IP(NPL)
      IP(N)=1
      DO 6 K=1,N
      IF(K.EQ.N)GO TO 5
      KP1=K+1
      M=K
      DO 1 I=KP1,N
    1 CONTINUE
      IP(K)=M
      IF(M.NE.K)IP(N)=-IP(N)
      T=A(M,K)
      A(M,K)=A(K,K)
      A(K,K)=T
      IF(T.EQ.0.)GO TO 5
      DO 2 I=KP1,N
    2 A(I,K)=-A(I,K)/T
      DO 4 J=KP1,N
      T=A(M,J)
      A(M,J)=A(K,J)
      A(K,J)=T
      IF(T.EQ.0.)GO TO 4
      DO 3 I=KP1,N
    3 A(I,J)=A(I,J)+A(I,K)*T
    4 CONTINUE
    5 IF(A(K,K).EQ.0.)IP(N)=0
    6 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SOLVE (B,A,N,NPL,IP)
      DIMENSION B(1)
      DIMENSION A(N,N),IP(NPL)
      IF(N.EQ.1)GO TO 9
      NM1=N-1
      DO 7 K=1,NM1
      KP1=K+1
      M=IP(K)
      T=B(M)
      B(M)=B(K)
      B(K)=T
      DO 7 I=KP1,N
    7 B(I)=B(I)+A(I,K)*T
      DO 8 KB=1,NM1
      KM1=N-KB
      K=KM1+1
      B(K)=B(K)/A(K,K)
      T=-B(K)
      DO 8 I=1,KM1
    8 B(I)=B(I)+A(I,K)*T
    9 B(1)=B(1)/A(1,1)
      RETURN
      END
```

```
      SUBROUTINE COMP (TOLRNC)
C-----ROUTINE TO COMPUTE THE DOWNWASH FUNCTION FOR A UNIFORMLY LOADED
C-----INFINITE TRIANGULAR PANEL IN SUBSONIC OR SUPERSONIC FLOW.
C-----REF. WOOD-AND,F.A.& J.AIRCRAFT NOV-DEC 1968.
C-----(X,Y) =COORDINATES OF FIELD POINT RELATIVE TO ORIGIN OF TRIANGLE
C-----B IS SLOPE (DX/DY) OF LEADING EDGE.
C-----BTERM   (REFER TO SUBROUTINE MATRIX)
C-----RESULT=ANSWER  (RATIO OF VERTICAL VELOCITY TO FREESTREAM VELOCITY)
C-----SUB=.TRUE. IF MACH)1  AND =.FALSE. IF MACH)1
C-----XT,UTHICK,DUMMY NOT USED IN THIS ROUTINE
C
      COMMON /MAT1/ X,XT,Y,B,BTERM,RESULT,UTHICK,SUB
C
      LOGICAL SUB
      DATA ZERO/0.0/,ONE/1.0/,PI/3.141593/
      R=ABS(Y)
      IF(SUB)GO TO 100
      IF (X.LT.TOLRNC) GO TO 50
      RSNGLP=R
      IF (RSNGLP.LT.TOLRNC) GO TO 50
      IF(B.GE.ONE)GO TO 60
C-----SUPERSONIC FLOW, SUPERSONIC LEADING EDGE.
      IF(X.LE.R)GO TO 20
C-----INSIDE MACH CONE
      D= SQRT(X*X-Y*Y)
      YSNGLP=Y
      IF (ABS(YSNGLP).LT.TOLRNC) GO TO 50
      RESULT=-(BTERM* ATAN2(BTERM*D,B*X-Y)+B*ALOG((X+D)/R)+D/Y)
      RETURN
C-----OUTSIDE MACH CONE
   20 DIFBNG=B*Y-X
      IF (Y.LT.ZERO.OR.X.LT.B*Y.OR.ABS(DIFBNG).LT.TOLRNC) GO TO 50
      RESULT=-BTERM*PI
      RETURN
   50 RESULT=ZERO
      RETURN
C-----SUPERSONIC FLOW, SUBSONIC LEADING EDGE.
   60 IF(X.LT.R)GO TO 50
      D= SQRT(X*X-Y*Y)
      YSNGLP=Y
      IF (ABS(YSNGLP).LT.TOLRNC) GO TO 50
      DUMARG=X-B*Y
      IF ((ABS(DUMARG)).LT.TOLRNC) GO TO 50
      RESULT=-(B*ALOG((X+D)/R)+D/Y
     1 -BTERM*ALOG((B*X-Y+BTERM*D)/ ABS(X-B*Y)))
      RETURN
C-----SUBSONIC FLOW
  100 D= SQRT(X*X+Y*Y)
      IF(B.EQ.ZERO)GO TO 102
      A= ABS(X-B*Y)
      YF6=(X+D)/Y
      RESULT=-(YF6-BTERM*ALOG( ABS(B*X+Y+BTERM*D)/A)+B*ALOG(YF6*A/(B*Y))
     1)
  101 RETURN
  102 RESULT=-((X+D)/Y-ALOG((Y+D)/ ABS(X)))
      RETURN
      END
```

```
      SUBROUTINE EXTVEL (NV,A,SIGMA,GAM,S,V)
C
C      COMPUTE VELOCITY FIELD INDUCED BY VORTEX PAIRS AND THEIR IMAGES
C
      COMPLEX S,V,SIGMA,SPR,SGM,SGMC,EYE,CD1
      DIMENSION SIGMA(4),GAM(4)
      COMMON /FLOW/ ALPHA,CSALP,EYE
C
      SPR=S/A
      V=CMPLX(0.0,0.0)
      DO 10 J=1,NV
      SGM=SIGMA(J)/A
      SGMC=CONJG(SGM)
      CD1=SPR-SGM
      D3=1.0
      D2=CABS(CD1)
      IF (D2.GT.1.0E-04) GO TO 20
      CD1=CMPLX(1.0,0.0)
      D3=0.0
   20 CONTINUE
      V=V - EYE*GAM(J)/A*(  D3/CD1      + 1.0/(SPR+1.0/SGM)
     1                  - 1.0/(SPR+SGMC) - 1.0/(SPR-1.0/SGMC))
   10 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TRJTRY
C      SUBROUTINE FOR DETERMINING THE PATHS OF FREE VORTICES IN THE
C         PRESENCE OF A WING-BODY COMBINATION                  260/C
      COMPLEX  SP,VP,SIGMA,EYE
      EXTERNAL FCT,OUTP
C
      DIMENSION GAM(4),YG(4),ZG(4),XBDY(100),RBDY(100),SBDY(100)
      DIMENSION PRMT(5),AUX(16,8),Y(8),Z(8)
      DIMENSION XP(100),YP(4),ZP(4),SP(100),VP(100) ,DRBDY(100)
      DIMENSION SIGMA(4),XV(200),YV(4,200),ZV(4,200),NVP(200),
     1            ARAY(5,200)
C
      COMMON /B/ MM,M
      COMMON /TRAJ1/ XI,XF,DXI,TITL,NPRINT
      COMMON /TRAJ2/ XP,SP,VP,NEXT
      COMMON /BODY/ NTBL,MTBL,XBDY,RBDY,SBDY,DRBDY
      COMMON /VORTEX/ NV,NV2,GAM,YG,ZG
      COMMON /VTABLE/ NTRY,XV,YV,ZV,NVP,ARAY
      COMMON /FLOW/ ALPHA,CSALP,EYE
      COMMON /PARAM/ DXOUT,XPRNT,MPRINT,XFINAL
C
  701 FORMAT (//I10)
  713 FORMAT (//5X67HEXECUTION STOP IN TRJTRY, INTEGRATION INTERVAL HAS
     1BECOME TOO SMALL //I4,1PE12.4//)
  715 FORMAT (///5X46HVELOCITY INDUCED AT SPECIFIED FIELD POINTS ON ,A4,
     1          9H (VORTEX) )
  716 FORMAT (/ 10X1HX,9X1HY,9X1HZ,11X8HV/V(INF),4X8HW/V(INF))
  717 FORMAT (I4,3F10.3,5X,2(1PE12.4))
  719 FORMAT (/10X,28HFIELD POINT OUTSIDE OF TABLE,5XI3,3F10.4,I9,F10.4)
C
      IF (XF.LT.-99.0) GO TO 20
      XFINAL=XF
      PRMT(1)=XI
      PRMT(2)=XF
      PRMT(3)=DXI
      PRMT(4)=.01
      NV2=NV+2
      NDIM=2*NV
      DO 10 J=1,NV
      JN=2*J
      Y(JN-1)=YG(J)
   10 Y(JN)=ZG(J)
      DUM=NDIM
      DUM=1./DUM
      DO 15 J=1, NDIM
   15 Z(J)=DUM
      XPRNT=XI-.0001
      CALL HPCG (PRMT,Y,Z,NDIM,IHLF,FCT,OUTP,AUX)
      IF (IHLF.GT.10) GO TO 60
      IF (NEXT.LE.0) RETURN
C
C       CALCULATE INDUCED VELOCITIES AT SPECIFIED FIELD POINTS
C
   20 CONTINUE
      MTBL=1
      DO 50 J=1,NEXT
      DO 51 K=1,NTRY
      KN=K
      IF (XV(K)-XP(J)) 51,52,53
   51 CONTINUE
   54 WRITE (6,719) J,XP(J),SP(J),KN,XV(KN)
      VP(J)=CMPLX(0.0,0.0)
      GO TO 50
   52 DO 55 KK=1,NV
   55 SIGMA(KK)=CMPLX(YV(KK,KN),ZV(KK,KN))
      GO TO 58
   53 IF (KN.EQ.1) GO TO 54
      DELTA=(XV(KN)-XP(J))/(XV(KN)-XV(KN-1))
      DO 56 KK=1,NV
      YP(KK)=YV(KK,KN)-DELTA*(YV(KK,KN)-YV(KK,KN-1))
      ZP(KK)=ZV(KK,KN)-DELTA*(ZV(KK,KN)-ZV(KK,KN-1))
   56 SIGMA(KK)=CMPLX(YP(KK),ZP(KK))
   58 CALL SHAPE (XP(J),A,B,DB)
      CALL EXTVEL (NV,A,SIGMA,GAM,SP(J),VP(J))
   50 CONTINUE
      IF (NPRINT.LT.0) GO TO 510
      WRITE (6,715) TITL
      WRITE (6,716)
  510 CONTINUE
      DO 40 J=1,NEXT
      VP(J)=CONJG(VP(J))
      IF (NPRINT.LT.0) GO TO 40
      WRITE (6,717) J,XP(J),SP(J),VP(J)
   40 CONTINUE
      RETURN
   60 WRITE (6,713) IHLF,M
      MM=IHLF
      RETURN
      END
```

```
      SUBROUTINE FCT (X,Y,Z)
      COMPLEX SIGMA,ENU,CENU,G,EYE,ADUM,BDUM,CDUM,DDUM,ENUN,ENUM,
     1        CENUN,CENUM,DNUDS,DWDS
      DIMENSION SIGMA(4),ENU(4),G(6,6),CENU(4),DWDS(4),DNUDS(4)
      DIMENSION Y(8),Z(8),GAM(4),YG(4),ZG(4)
      COMMON /B/ MM,M
      COMMON /VORTEX/ NV,NV2,GAM,YG,ZG
      COMMON /FLOW/ ALPHA,CSALP,EYE
      COMMON /RESULT/ G,DWDS,SIGMA,ENU,DNUDS,A,S,RZ,DS
C
      MM=0
C
C      LOOK UP BODY RADIUS AND LOCAL WING OR TAIL SEMISPAN
C
      CALL SHAPE (X,A,S,DS)
      RZ=0.5*(S + A*A/S)
      ARZ=A/RZ
      DO 9 J=1,NV
      K=2*J
    9 SIGMA(J)=CMPLX(Y(K-1),Y(K))
      DEM=(2.*RZ)**2
      DO 10 J=1,NV
      CDUM=SIGMA(J) + A*A/SIGMA(J)
      ENU(J)=0.5*CDUM + 0.5*CSQRT(CDUM*CDUM - DEM)
   10 CENU(J)=CONJG(ENU(J))
C
C      COMPUTE G-FUNCTIONS FOR COMPLEX VELOCITY EXPRESSIONS
C
      DO 20 N=1,NV
```

```
      D*DS(N)=CMPLX(0.0,0.0)
      ENUN=ENU(N)/RZ
      CENUN=CENU(N)/RZ
      ADUM=ENUN + 1.0/ENUN
      BDUM=CSQRT(ADUM*ADUM - 4.*ANZ*ARZ)
      CDUM=(ENUN*ENUN-1.0)*(ADUM+BDUM)
      DNUDS(N)=2.0*ENUN*ENUN*BDUM/CDUM
C
C       CALCULATE CROSSFLOW CONTRIBUTION , G(NV+1,N) , INCLUDING EFFECT
C          OF BODY DA/DX
C
      G(NV+1,N)=-EYE*(1.0 + 1.0/(ENUN*ENUN))*DNUDS(N)*ALPHA
     1             + DS*A*CSALP/SIGMA(N)
C
C       CALCULATE TRANSFORMATION CONTRIBUTION , G(NV+2,N)
C
      DDUM=ADUM/BDUM - 1.0 - 2.0*ENUN*BDUM/((ENUN*ENUN-1.0)**2)
      DDUM=DDUM/(ADUM       + BDUM)/RZ
      G(NV+2,N)=-EYE*GAM(N)*DDUM
C
C       CALCULATE VORTEX INTERFERENCE CONTRIBUTION , G(M,N)
C
      DO 22 M=1,NV
      IF (M.EQ.N) GO TO 21
      ENUM=ENU(M)/RZ
      CENUM=CENU(M)/RZ
      DDUM=1.0/(ENUN-ENUM) + 1.0/(ENUN+1.0/ENUM) - 1.0/(ENUN-CENUM)
     1    - 1.0/(ENUN-1.0/CENUM)
      G(M,N)=-EYE*GAM(M)/RZ*DDUM*DNUDS(N)
      GO TO 22
   21 DDUM=1.0/ADUM - 1.0/(ENUN-CENUN) - 1.0/(ENUN-1.0/CENUN)
      G(M,N)=-EYE*GAM(N)/RZ*DDUM*DNUDS(N)
   22 CONTINUE
   20 CONTINUE
      DO 26 N=1,NV
      DO 25 M=1,NV2
   25 DWDS(N)=DWDS(N) + G(M,N)
   26 DWDS(N)=CONJG(DWDS(N))
      DO 27 N=1,NV
      N2=2*N
      Z(N2)= AIMAG(DWDS(N))
   27 Z(N2-1)=REAL(DWDS(N))
      RETURN
      END
```

```
      SUBROUTINE HPCG (PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
C
      DIMENSION PRMT(5),Y(8),DERY(8),AUX(16,8)
      COMMON /B/ HH,H
      N=1
      IHLF=0
      X=PRMT(1)
      H=PRMT(3)
      PRMT(5)=0.
      DO 1 I=1,NDIM
      AUX(16,I)=0.
      AUX(15,I)=DERY(I)
    1 AUX(1,I)=Y(I)
      IF(H*(PRMT(2)-X))3,2,4
C
C       ERROR RETURNS
    2 IHLF=12
      GO TO 4
    3 IHLF=13
C
C       COMPUTATION OF DERY FOR STARTING VALUES
    4 CALL FCT(X,Y,DERY)
      IF(HH.GT.1) RETURN
C
C       RECORDING OF STARTING VALUES
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))6,5,6
    5 IF(IHLF)7,7,6
    6 RETURN
    7 DO 8 I=1,NDIM
    8 AUX(8,I)=DERY(I)
C
C       COMPUTATION OF AUX(2,I)
      ISW=1
      GO TO 100
C
    9 X=X+H
      DO 10 I=1,NDIM
   10 AUX(2,I)=Y(I)
C
C       INCREMENT H IS TESTED BY MEANS OF BISECTION
   11 IHLF=IHLF+1
      X=X-H
      DO 12 I=1,NDIM
   12 AUX(4,I)=AUX(2,I)
      H=.500*H
      N=1
      ISW=2
      GO TO 100
C
   13 X=X+H
      CALL FCT(X,Y,DERY)
      IF(HH.GT.1) RETURN
      N=2
      DO 14 I=1,NDIM
      AUX(2,I)=Y(I)
   14 AUX(9,I)=DERY(I)
      ISW=3
      GO TO 100
C
C       COMPUTATION OF TEST VALUE DELT
   15 DELT=0.
      DO 16 I=1,NDIM
   16 DELT=DELT+AUX(15,I)*ABS(Y(I)-AUX(4,I))
      DELT=.06666667*DELT
      IF(DELT-PRMT(4))19,19,17
   17 IF(IHLF-10)11,18,18
C
C       NO SATISFACTORY ACCURACY AFTER 10 BISECTIONS. ERROR MESSAGE.
   18 IHLF=11
      X=X+H
      GO TO 4
C
C       THERE IS SATISFACTORY ACCURACY AFTER LESS THAN 11 BISECTIONS.
   19 X=X+H
      CALL FCT(X,Y,DERY)
      IF(HH.GT.1) RETURN
      DO 20 I=1,NDIM
      AUX(3,I)=Y(I)
   20 AUX(10,I)=DERY(I)
      N=3
      ISW=4
      GO TO 100
C
   21 N=1
      X=X+H
      CALL FCT(X,Y,DERY)
      IF(HH.GT.1) RETURN
      X=PRMT(1)
      DO 22 I=1,NDIM
      AUX(11,I)=DERY(I)
   22 Y(I)=AUX(1,I)+H*(.375*AUX(8,I)+.7916667*AUX(9,I)-.20833333*AUX(10
     1,I)+.04166667*DERY(I))
   23 X=X+H
      N=N+1
      CALL FCT(X,Y,DERY)
      IF(HH.GT.1) RETURN
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))6,24,6
   24 IF(N-4)25,200,200
   25 DO 26 I=1,NDIM
      AUX(N,I)=Y(I)
   26 AUX(N+7,I)=DERY(I)
```

```
      IF(N-3)27,29,200
C
   27 DO 28 I=1,NDIM
      DELT=AUX(9,I)+AUX(9,I)
      DELT=DELT+DELT
   28 Y(I)=AUX(1,I)+,33333333*H*(AUX(8,I)+DELT+AUX(10,I))
      GO TO 23
C
   29 DO 30 I=1,NDIM
      DELT=AUX(9,I)+AUX(10,I)
      DELT=DELT+DELT+DELT
   30 Y(I)=AUX(1,I)+,375*H*(AUX(8,I)+DELT+AUX(11,I))
      GO TO 23
C
C
C     *****************************************************************
C     THE FOLLOWING PART OF SUBROUTINE DHPCG COMPUTES BY MEANS OF
C     RUNGE-KUTTA METHOD STARTING VALUES FOR THE NOT SELF-STARTING
C     PREDICTOR-CORRECTOR METHOD.
  100 DO 101 I=1,NDIM
      Z=H*AUX(N+7,I)
      AUX(5,I)=Z
  101 Y(I)=AUX(N,I)+,4*Z
C     Z IS AN AUXILIARY STORAGE LOCATION
C
      Z=X+,4*H
      CALL FCT(Z,Y,DERY)
      IF(MM,GT,1) RETURN
      DO 102 I=1,NDIM
      Z=H*DERY(I)
      AUX(6,I)=Z
  102 Y(I)=AUX(N,I)+,29697761*AUX(5,I)+,15875964*Z
C
      Z=X+,45573725*H
      CALL FCT(Z,Y,DERY)
      IF(MM,GT,1) RETURN
      DO 103 I=1,NDIM
      Z=H*DERY(I)
      AUX(7,I)=Z
  103 Y(I)=AUX(N,I)+,21810039*AUX(5,I)-3,05096515*AUX(6,I)+3,83286476*Z
C
      Z=X+H
      CALL FCT(Z,Y,DERY)
      IF(MM,GT,1) RETURN
      DO 104 I=1,NDIM
  104 Y(I)=AUX(N,I)+,17476028*AUX(5,I)-,55148066*AUX(6,I)+1,2055356*AUX(
     17,I)+,17118478*H*DERY(I)
      GO TO(9,13,15,21),ISW
C     *****************************************************************
C
C
C     POSSIBLE BREAK-POINT FOR LINKAGE
C
C
C     STARTING VALUES ARE COMPUTED.
C     NOW START HAMMINGS MODIFIED PREDICTOR-CORRECTOR METHOD.
  200 ISTEP=3
  201 IF(N-8)204,202,204
C
C     N=8 CAUSES THE ROWS OF AUX TO CHANGE THEIR STORAGE LOCATIONS
  202 DO 203 N=2,7
      DO 203 I=1,NDIM
      AUX(N-1,I)=AUX(N,I)
  203 AUX(N+6,I)=AUX(N+7,I)
      N=7
C
C     N LESS THAN 8 CAUSES N+1 TO GET N
  204 N=N+1
C
C     COMPUTATION OF NEXT VECTOR Y
      DO 205 I=1,NDIM
      AUX(N-1,I)=Y(I)
  205 AUX(N+6,I)=DERY(I)
      X=X+H
  206 ISTEP=ISTEP+1
      DO 207 I=1,NDIM
      DELT=AUX(N-4,I)+1,33333333*H*(AUX(N+6,I)+AUX(N+6,I)-AUX(N+5,I)+AUX
     1(N+4,I)+AUX(N+4,I))
      Y(I)=DELT-,92561983*AUX(16,I)
  207 AUX(16,I)=DELT
C     PREDICTOR IS NOW GENERATED IN ROW 16 OF AUX, MODIFIED PREDICTOR
C     IS GENERATED IN Y. DELT MEANS AN AUXILIARY STORAGE.
C
      CALL FCT(X,Y,DERY)
      IF (MM,GT,1) RETURN
C     DERIVATIVE OF MODIFIED PREDICTOR IS GENERATED IN DERY
C
      DO 208 I=1,NDIM
      DELT=,12500*(9,00*AUX(N-1,I)-AUX(N-3,I)+3,00*H*(DERY(I)+AUX(N+6,I)
     1+AUX(N+6,I)-AUX(N+5,I)))
      AUX(16,I)=AUX(16,I)-DELT
  208 Y(I)=DELT+,074380165*AUX(16,I)
C
C     TEST WHETHER H MUST BE HALVED OR DOUBLED
      DELT=0,
      DO 209 I=1,NDIM
  209 DELT=DELT+AUX(15,I)*ABS(AUX(16,I))
      IF(DELT-PRMT(4))210,222,222
C
C     H MUST NOT BE HALVED, THAT MEANS Y(I) ARE GOOD.
  210 CALL FCT(X,Y,DERY)
      IF(MM,GT,1) RETURN
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))212,211,212
  211 IF(IHLF-11)213,212,212
  212 RETURN
  213 IF(H*(X-PRMT(2)))214,212,212
  214 IF(ABS(X-PRMT(2))-,1*ABS(H))212,215,215
  215 IF(DELT-,02*PRMT(4))216,216,201
C
C
C     H COULD BE DOUBLED IF ALL NECESSARY PRECEDING VALUES ARE
C     AVAILABLE
  216 IF(IHLF)201,201,217
  217 IF(N-7)201,218,218
  218 IF(ISTEP-4)201,219,219
  219 IMOD=ISTEP/2
      IF(ISTEP-IMOD-IMOD)201,220,201
  220 H=H+H
      IHLF=IHLF-1
      ISTEP=0
      DO 221 I=1,NDIM
      AUX(N-1,I)=AUX(N-2,I)
      AUX(N-2,I)=AUX(N-4,I)
      AUX(N-3,I)=AUX(N-6,I)
      AUX(N+6,I)=AUX(N+5,I)
      AUX(N+5,I)=AUX(N+3,I)
      AUX(N+4,I)=AUX(N+1,I)
      DELT=AUX(N+6,I)+AUX(N+5,I)
      DELT=DELT+DELT+DELT
  221 AUX(16,I)=8,96296296*(Y(I)-AUX(N-3,I))-3,36111111*H*(DERY(I)+DELT+
     1AUX(N+4,I))
      GO TO 201
C
C
C     H MUST BE HALVED
  222 IHLF=IHLF+1
      IF(IHLF-10)223,223,210
  223 H=,5*H
      ISTEP=0
      DO 224 I=1,NDIM
      Y(I)=,390625E-2*(8,E1*AUX(N-1,I)+135,00*AUX(N-2,I)+4,E1*AUX(N-3,I)
     1+AUX(N-4,I))-,1171875*(AUX(N+6,I)-6,*AUX(N+5,I)-AUX(N+4,I))*H
      AUX(N-4,I)=,390625E-2*(12,00*AUX(N-1,I)+135,00*AUX(N-2,I)+
     1108,00*AUX(N-3,I)+AUX(N-4,I))-,0234375*(AUX(N+6,I)+
     218,00*AUX(N+5,I)-9,00*AUX(N+4,I))*H
      AUX(N-3,I)=AUX(N-2,I)
  224 AUX(N+4,I)=AUX(N+5,I)
      X=X-H
      DELT=X-(H+H)
      CALL FCT(DELT,Y,DERY)
      IF(MM,GT,1) RETURN
      DO 225 I=1,NDIM
      AUX(N-2,I)=Y(I)
```

```
      AUX(N+5,I)=DERY(I)                                              SP21 258
  225 Y(I)=AUX(N+4,I)                                                 SP21 259
      DELT=DELT*(H+H)                                                 SP21 260
      CALL FCT(DELT,Y,DERY)                                           SP21 261
      IF(MM.GT.1) RETURN                                              SP21 262
      DO 226 I=1,NDIM                                                 SP21 263
      DELT=AUX(N+5,I)+AUX(N+4,I)                                      SP21 264
      DELT=DELT+DELT+DELT                                             SP21 265
      AUX(16,I)=8.962962963*(AUX(N-1,I)-Y(I))                         SP21 266
     1-3.361111111*H*(AUX(N+6,I)+DELT+DERY(I))                        SP21 267
  226 AUX(N+3,I)=DERY(I)                                              SP21 268
      GO TO 206                                                       SP21 269
      END                                                             SP21 270


      SUBROUTINE OUTP (X,Y,Z,IHLF,NDIM,PRMT)                          SP22 001
      COMPLEX G,SIGMA,ENU,DWDS,DNUDS                                  SP22 002
      DIMENSION Y(8),Z(8),PRMT(5),G(6,6),DWDS(4),SIGMA(4),ENU(4),     SP22 003
     1          GAM(4),YG(4),ZG(4),DNUDS(4),                          SP22 004
     2          XV(200),YV(4,200),ZV(4,200),NVP(200),ARAY(5,200)      SP22 005
      COMMON /VORTEX/ NV,NV2,GAM,YG,ZG                                SP22 006
      COMMON /RESULT/ G,DWDS,SIGMA,ENU,DNUDS,A,B,RZ,DS                SP22 007
      COMMON /PARAM/ DXOUT,XPRNT,MPRINT,XFINAL                        SP22 008
      COMMON /B/ MM,M                                                 SP22 009
      COMMON /VTABLE/ NTRY,XV,YV,ZV,NVP,ARAY                          SP22 010
C                                                                     SP22 011
      IF (X.LT.XPRNT) RETURN                                          SP22 012
  700 FORMAT (//5X1HX,8X2HDX,11X1HA,9X1HB,8X2HRO,6X5HDA/DX)           SP22 013
  701 FORMAT (F9.4,F11.7,4F10.4)                                      SP22 014
  702 FORMAT (/1X6HVORTEX,1X11HSIGMA(REAL),3X6H(IMAG),5X8HNU(REAL),   SP22 015
     1       4X6H(IMAG),3X11HDW/DS(REAL),3X6H(IMAG),                  SP22 016
     2       2X12HDNU/DS(REAL),3X6H(IMAG))                            SP22 017
  703 FORMAT (I4,2X4(1PE12.3,E11.3)/5X,6(E11.3,E10.3))                SP22 018
      IF (MPRINT.LE.0) GO TO 15                                       SP22 019
      WRITE (6,700)                                                   SP22 020
      WRITE (6,701) X,M,A,B,RZ,DS                                     SP22 021
      WRITE (6,702)                                                   SP22 022
      DO 10 N=1,NV                                                    SP22 023
   10 WRITE (6,703) N,SIGMA(N),ENU(N),DWDS(N),DNUDS(N),(G(M,N),M=1,NV2) SP22 024
   15 XPRNT=X+DXOUT                                                   SP22 025
      IF (XPRNT.GE.XFINAL) XPRNT=XFINAL-M                             SP22 026
      NTRY=NTRY+1                                                     SP22 027
      IF (NTRY.GT.200) RETURN                                         SP22 028
C                                                                     SP22 029
C     SAVE TABLE OF VORTEX POSITIONS                                  SP22 030
C                                                                     SP22 031
      XV(NTRY)=X                                                      SP22 032
      NVP(NTRY)=NV                                                    SP22 033
      ARAY(1,NTRY)=M                                                  SP22 034
      ARAY(2,NTRY)=A                                                  SP22 035
      ARAY(3,NTRY)=B                                                  SP22 036
      ARAY(4,NTRY)=RZ                                                 SP22 037
      ARAY(5,NTRY)=DS                                                 SP22 038
      DO 20 N=1,NV                                                    SP22 039
      YV(N,NTRY)=REAL(SIGMA(N))                                       SP22 040
   20 ZV(N,NTRY)=AIMAG(SIGMA(N))                                      SP22 041
      RETURN                                                          SP22 042
      END                                                             SP22 043


      SUBROUTINE ZVIX (ALPHAD,EPS,BTA2)                               SP23 001
C                                                                     SP23 002
C     COMPUTE VERTICAL POSITION OF WING SEPARATION VORTEX             SP23 003
C                                                                     SP23 004
      DIMENSION A2E(8),B2E(8)                                         SP23 005
C                                                                     SP23 006
      DATA A2E/0.0,0.2,0.4,0.6,0.8,1.0,1.2,1.4/                       SP23 007
      DATA B2E/0.0,0.11,0.22,0.3,0.375,0.45,0.52,0.575/               SP23 008
C                                                                     SP23 009
      F=1.0                                                           SP23 010
      X=ALPHAD/(2.*EPS)                                               SP23 011
      IF (X .LT. 0.0) F=-1.0                                          SP23 012
      X=X*F                                                           SP23 013
      DO 10 J=1,8                                                     SP23 014
      K=J                                                             SP23 015
      IF (X-A2E(J)) 12,14,10                                          SP23 016
   10 CONTINUE                                                        SP23 017
      BTA2=ALPHAD/2.4                                                 SP23 018
      RETURN                                                          SP23 019
   14 BTA2=B2E(K)*EPS*F                                               SP23 020
      RETURN                                                          SP23 021
   12 DELTA=(X-A2E(K-1))/(A2E(K)-A2E(K-1))                            SP23 022
      BTA2=B2E(K-1) + DELTA*(B2E(K)-B2E(K-1))                         SP23 023
      BTA2=BTA2*EPS*F                                                 SP23 024
      RETURN                                                          SP23 025
      END                                                             SP23 026


      SUBROUTINE WPANL (NSURF,NR,NPL,NMAT,NPRINT,PANELS,MACH,         SP24 001
     1             XY,XCPT,YCPT,XBAR,OUTCOD,SLE,STE,TOLRNC,XPNLF,XPNLB) SP24 002
C     THIS SUBROUTINE CALCULATES THE UPWASH W/V INDUCED BY CONSTANT   SP24 003
C     PRESSURE PANELS WITH KNOWN DELTA P/Q AT THE LOWER OUTBOARD CORNER SP24 004
C     AND CENTROID OF AN ELEMENTAL PANEL                              SP24 005
C                                                                     SP24 006
      DIMENSION XY(6,100),XCPT(NPL),YCPT(NPL),XBAR(NPL),SLE(NPL),     SP24 007
     1          STE(NPL),XPNLF(NPL),XPNLB(NPL)                        SP24 008
      DIMENSION SUMTA(100),SUMTC(100)                                 SP24 009
      COMMON/WVEL/WC4(100),WCT(100)                                   SP24 010
      COMMON/PARAMS/ ASYM                                             SP24 011
      COMMON /MAT1/ X,XT,Y,B,BTERM,RESULT,UTHICK,SUB                  SP24 012
      COMMON/PRESS/CP(100)                                            SP24 013
C                                                                     SP24 014
      REAL MACH                                                       SP24 015
      LOGICAL SUB,ASYM                                                SP24 016
      INTEGER PANELS,OUTCOD                                           SP24 017
      DATA PI/3.141593/,ZERO/0.0/                                     SP24 018
C                                                                     SP24 019
      BETA=SQRT(ABS(MACH*MACH-1.0))                                   SP24 020
      BETAI=1.0/BETA                                                  SP24 021
      CONS=BETA/(4.0*PI)                                              SP24 022
      XTERM=1.000                                                     SP24 023
C                                                                     SP24 024
    1 FORMAT (1H1,5X,46HUPWASH INDUCED AT CENTROID OF ELEMENTAL PANELS, SP24 025
     1//2X,1HJ,6X,1HX,14X,1HY,19X,20HW/V,POSITIVE UPWARDS )           SP24 026
    2 FORMAT (1X,I3 ,1X,F10.5,5X,F10.5,15X,F10.5)                     SP24 027
    3 FORMAT (///5X,59HUPWASH INDUCED AT LOWER OUTBOARD CORNER OF ELEMENSP24 028
     1TAL PANELS,//2X,1HJ,6X,1HX,14X,1HY,19X,20HW/V,POSITIVE UPWARDS ) SP24 029
  701 FORMAT(///30X,25H***** INFLUENCE OF PANEL ,I2,1X,5H*****)       SP24 030
  702 FORMAT(/12X,5HSUMTA/)                                           SP24 031
  703 FORMAT(10(2X,F10.4))                                            SP24 032
  704 FORMAT(/12X,5HSUMTC/)                                           SP24 033
C                                                                     SP24 034
C     INITIALIZE UPWASHES                                             SP24 035
C                                                                     SP24 036
      DO 180 J=1,PANELS                                               SP24 037
      WC4(J)=0.0                                                      SP24 038
  180 WCT(J)=0.0                                                      SP24 039
C                                                                     SP24 040
C                                                                     SP24 041
C                                                                     SP24 042
C                                                                     SP24 043
C                                                                     SP24 044
      DO 900 I=1,PANELS                                               SP24 045
```

```
C
C     I IS INDEX OF INFLUENCING CONSTANT PRESSURE PANEL
C
      B1=SLE(I)*BETA1
      B2=STE(I)*BETA1
      BTERM1= SQRT( ABS(B1*B1+XTERM))
      BTERM2= SQRT( ABS(B2*B2+XTERM))
      X1=XY(1,I)*BETA1
      X2=XY(2,I)*BETA1
      X3=XY(3,I)*BETA1
      X4=XY(4,I)*BETA1
      Y1=XY(5,I)-XY(5,1)
      Y2=XY(6,I)-XY(5,1)
      Y3=Y1
      Y4=Y2
C
C     J IS INDEX OF THE PANEL WITH THE TWO FIELDPOINTS AT WHICH UPWASH
C     IS TO BE CALCULATED
C
      DO 850 J=1,PANELS
      SUMT4(J)=0.
      SUMTC(J)=0.
      XC4=XY(4,J)*BETA1
      XCT=XBAR(J)*BETA1
      YC4=XY(6,J)-XY(5,1)
      YCT=YCPT(J)-XY(5,1)
      BTERM=BTERM1
C
      IF (B1.LT.0.0) GO TO 820
      B=B1
      X=XC4-X1
      Y=YC4-Y1
      CALL COMP (TOLRNC)
      SUMHC4=RESULT
      IF(ASYM) GO TO 412
      Y=-YC4-Y1
      CALL COMP (TOLRNC)
      SUMHC4=SUMHC4+RESULT
  412 X=XCT-X1
      Y=YCT-Y1
      CALL COMP (TOLRNC)
      SUMHCT=RESULT
      IF (ASYM) GO TO 12
      Y=-YCT-Y1
      CALL COMP (TOLRNC)
      SUMHCT=SUMHCT+RESULT
   12 CONTINUE
  521 X=XC4-X2
      Y=YC4-Y2
      CALL COMP (TOLRNC)
      SUMHC4=SUMHC4-RESULT
      IF (ASYM) GO TO 413
      Y=-YC4-Y2
      CALL COMP (TOLRNC)
      SUMHC4=SUMHC4-RESULT
  413 CONTINUE
  502 X=XCT-X2
      Y=YCT-Y2
      CALL COMP (TOLRNC)
      SUMHCT=SUMHCT-RESULT
      IF (ASYM) GO TO 403
      Y=-YCT-Y2
      CALL COMP (TOLRNC)
      SUMHCT=SUMHCT-RESULT
  403 CONTINUE
      GO TO 522
  820 B=-B1
      X=XC4-X1
      Y=Y1-YC4
      CALL COMP (TOLRNC)
      SUMHC4= -RESULT
      IF(ASYM) GO TO 16
      Y=Y1 + YC4
      CALL COMP (TOLRNC)
   16 X=XCT - X1
      SUMHC4 = SUMHC4 - RESULT
      Y=Y1-YCT
      CALL COMP (TOLRNC)
      SUMHCT = -RESULT
      IF(ASYM) GO TO 17
      Y=Y1 + YCT
      CALL COMP (TOLRNC)
      SUMHCT=SUMHCT-RESULT
   17 X=XC4 - X2
      Y=Y2 - YC4
      CALL COMP(TOLRNC)
      SUMHC4=SUMHC4 + RESULT
      IF(ASYM) GO TO 18
      Y=Y2 + YC4
      CALL COMP (TOLRNC)
   18 X=XCT - X2
      SUMHC4=SUMHC4 + RESULT
      Y=Y2 - YCT
      CALL COMP (TOLRNC)
      SUMHCT=SUMHCT + RESULT
      IF(ASYM) GO TO 522
      Y=Y2 + YCT
      CALL COMP (TOLRNC)
      SUMHCT=SUMHCT + RESULT
  522 BTERM=BTERM2
      IF (B2.LT.0.0) GO TO 830
      B=B2
      X=XC4-X3
      Y=YC4-Y3
      CALL COMP (TOLRNC)
      SUMHC4=SUMHC4-RESULT
      IF (ASYM) GO TO 414
      Y=-YC4-Y3
      CALL COMP (TOLRNC)
      SUMHC4=SUMHC4-RESULT
  414 X=XCT-X3
      Y=YCT-Y3
      CALL COMP (TOLRNC)
      SUMHCT=SUMHCT-RESULT
      IF (ASYM) GO TO 23
      Y=-YCT-Y3
      CALL COMP (TOLRNC)
      SUMHCT=SUMHCT-RESULT
   23 CONTINUE
  583 X=XC4-X4
      Y=YC4-Y4
      CALL COMP (TOLRNC)
      SUMHC4=SUMHC4+RESULT
      IF (ASYM) GO TO 415
      Y=-YC4-Y4
      CALL COMP (TOLRNC)
      SUMHC4=SUMHC4+RESULT
  415 X=XCT-X4
      Y=YCT-Y4
      CALL COMP (TOLRNC)
      SUMHCT=SUMHCT+RESULT
      IF (ASYM) GO TO 840
      Y=-YCT-Y4
      CALL COMP (TOLRNC)
      SUMHCT=SUMHCT+RESULT
  840 CONTINUE
      GO TO 504
  830 B= -B2
      X=XC4-X3
      Y=Y3-YC4
      CALL COMP (TOLRNC)
      SUMHC4=SUMHC4 + RESULT
      IF(ASYM) GO TO 20
      Y= Y3 + YC4
      CALL COMP(TOLRNC)
      SUMHC4 = SUMHC4 + RESULT
   20 X= XCT - X3
      Y=Y3 - YCT
      CALL COMP (TOLRNC)
      SUMHCT = SUMHCT + RESULT
      IF(ASYM) GO TO 21
      Y=Y3 + YCT
      CALL COMP(TOLRNC)
      SUMHCT = SUMHCT + RESULT
```

```
   21 X=XC4 - X4                                                        SP24 200
      Y=Y4-YC4                                                          SP24 201
      CALL COMP(TOLRNC)                                                 SP24 202
      SUMWC4=SUMWC4 - RESULT                                            SP24 203
      IF(ASYM) GO TO 22                                                 SP24 204
      Y=Y4+YC4                                                          SP24 205
      CALL COMP (TOLRNC)                                                SP24 206
      SUMWC4=SUMWC4 - RESULT                                            SP24 207
   22 X=XCT-X4                                                          SP24 208
      Y=Y4-YCT                                                          SP24 209
      CALL COMP(TOLRNC)                                                 SP24 210
      SUMWCT=SUMWCT-RESULT                                              SP24 211
      IF(ASYM) GO TO 504                                                SP24 212
      Y= Y4 + YCT                                                       SP24 213
      CALL COMP (TOLRNC)                                                SP24 214
      SUMWCT = SUMWCT - RESULT                                          SP24 215
  504 WC4(J)=CP(I)*SUMWC4*CONS+WC4(J)                                   SP24 216
      WCT(J)=CP(I)*SUMWCT*CONS+WCT(J)                                   SP24 217
      SUMT4(J)=CP(I)*SUMWC4*CONS                                        SP24 218
      SUMTC(J)=CP(I)*SUMWCT*CONS                                        SP24 219
  850 CONTINUE                                                          SP24 220
      IF (OUTCOD.LT.5) GO TO 900                                        SP24 221
      WRITE(6,701) 1                                                    SP24 222
      WRITE(6,702)                                                      SP24 223
      WRITE(6,703)  (SUMT4(K),K=1,PANELS)                               SP24 224
      WRITE(6,704)                                                      SP24 225
      WRITE(6,703) (SUMTC(K),K=1,PANELS)                                SP24 226
  900 CONTINUE                                                          SP24 227
      IF (OUTCOD.LT.4) RETURN                                           SP24 228
C                                                                       SP24 229
C     PRINT UPWASH INDUCED AT VARIOUS PANEL POINTS                      SP24 230
C                                                                       SP24 231
      WRITE(6,1)                                                        SP24 232
      DO 600 J=1,PANELS                                                 SP24 233
  600 WRITE(6,2)J,XBAR(J),YCPT(J),WCT(J)                                SP24 234
      WRITE(6,3)                                                        SP24 235
      DO 601 J=1,PANELS                                                 SP24 236
  601 WRITE(6,2)J,XY(4,J),XY(6,J),WC4(J)                                SP24 237
C                                                                       SP24 238
      RETURN                                                            SP24 239
      END                                                               SP24 240
```

```
      SUBROUTINE KFACT (ROB,CKWB,FMACH,PARAM,CKBW,NAFTER,SMB,TAPER,BDCR,SP25 001
     1CLA,BETA,SWPLE,BETAA,SKWB,SKBW)                                   SP25 002
C                                                                       SP25 003
C     SUBROUTINE TO CALCULATE INTERFERENCE FACTORS, BOXES 47-50 OR      SP25 004
C          63-66 OF TABLE I OF REPORT 1307                              SP25 005
C                                                                       SP25 006
      CKWB=EQ14(ROB)                                                    SP25 007
      IF (FMACH.GT.1.0.AND.PARAM.GT.4.0) GO TO 10                       SP25 008
      CKBW=EQ21(ROB)                                                    SP25 009
      GO TO 100                                                         SP25 010
   10 IF (NAFTER.EQ.0) GO TO 30                                         SP25 011
      IF (SMB.GT.1.0) GO TO 15                                          SP25 012
      CKBW=EQ26(SMB,TAPER,BDCR,ROB)/(BETA*CLA)                          SP25 013
      GO TO 100                                                         SP25 014
   15 IF (SWPLE.GT.0.0) GO TO 20                                        SP25 015
      CKBW=EQ28L(TAPER,BDCR,ROB)/(BETA*CLA)                             SP25 016
      GO TO 100                                                         SP25 017
   20 CKBW=EQ28(SMB,TAPER,BDCR,ROB)/(BETA*CLA)                          SP25 018
      GO TO 100                                                         SP25 019
   30 IF (SMB.GT.1.0) GO TO 35                                          SP25 020
      CKBW=EQ31(SMB,TAPER,BDCR,ROB)/(BETA*CLA)                          SP25 021
      GO TO 100                                                         SP25 022
   35 IF (SWPLE.GT.0.0) GO TO 40                                        SP25 023
      CKBW=EQ30L(TAPER,BDCR,ROB)/(BETA*CLA)                             SP25 024
      GO TO 100                                                         SP25 025
   40 CKBW=EQ30(SMB,TAPER,BDCR,ROB)/(BETA*CLA)                          SP25 026
  100 CONTINUE                                                          SP25 027
      RETURN                                                            SP25 028
      END                                                               SP25 029
```

```
      SUBROUTINE CH1416(S,R,SWPLE,CT,CR,FMACH,ARPARM,NAFTER,ROCR,XBCR,  SP26 001
     1SMB,TAPER,ROS,BAR)                                                SP26 002
C                                                                       SP26 003
C     SUBROUTINE TO CALCULATE BODY CENTER OF PRESSURE DUE TO WING OR    SP26 004
C          TAIL, CHART 14, 15, OR 16 OF REPORT 1307.  IN TABLE I OF THATSP26 005
C          REPORT THIS QUANTITY IS IN BOX 57 OR 73.                     SP26 006
C                                                                       SP26 007
      DIMENSION BD14B(6),XB14B(6,2),BA15(7),               TA15(7,4),   SP26 008
     1TB15(7,4),TC15(7,4),TD15(7,4),TE15(7,4),TF15(7,4),TG15(7,4),TH15(7SP26 009
     2,4),TI15(7,4),BA16(8),                TA16(8,4),TB16(8,4),TC16(8,4)SP26 010
     3,TD16(8,4),TE16(8,4),TF16(8,4),TG16(8,4),TH16(8,4),TI16(8,4)       SP26 011
      DIMENSION XB15A(7,4,3),XB15B(7,4,3),XB15C(7,4,3),                  SP26 012
     1XB16A(8,4,3),XB16B(8,4,3),XB16C(8,4,3)                            SP26 013
      DIMENSION VA(4,3,3),V(3,3)                                        SP26 014
      EQUIVALENCE (XB15A(1,1,1),TA15(1,1)),(XB15A(1,1,2),TB15(1,1)),    SP26 015
     1(XB15A(1,1,3),TC15(1,1)),(XB15B(1,1,1),TD15(1,1)),                SP26 016
     2(XB15B(1,1,2),TE15(1,1)),(XB15B(1,1,3),TF15(1,1)),                SP26 017
     3(XB15C(1,1,1),TG15(1,1)),(XB15C(1,1,2),TH15(1,1)),                SP26 018
     4(XB15C(1,1,3),TI15(1,1))                                          SP26 019
      EQUIVALENCE (XB16A(1,1,1),TA16(1,1)),(XB16A(1,1,2),TB16(1,1)),    SP26 020
     1(XB16A(1,1,3),TC16(1,1)),(XB16B(1,1,1),TD16(1,1)),                SP26 021
     2(XB16B(1,1,2),TE16(1,1)),(XB16B(1,1,3),TF16(1,1)),                SP26 022
     3(XB16C(1,1,1),TG16(1,1)),(XB16C(1,1,2),TH16(1,1)),                SP26 023
     4(XB16C(1,1,3),TI16(1,1))                                          SP26 024
      DATA RAD/57.29578/                                                SP26 025
      DATA BD14B/0.0, 0.2, 0.4, 0.6, 0.8, 1.0/                          SP26 026
      DATA XB14B/0.500, 0.590, 0.637, 0.658, 0.665, 0.667,              SP26 027
     1           0.500, 0.575, 0.622, 0.646, 0.660, 0.667/              SP26 028
      DATA BA15/ 0.0, 0.5, 1.0, 1.5, 2.0, 3.0, 4.0/                     SP26 029
      DATA BA16/ 0.0, 0.5, 1.0, 1.5, 2.0, 3.0, 4.0, 1000.0/             SP26 030
      DATA TA15/ 0.00, 0.13, 0.25, 0.34, 0.40, 0.48, 0.50,              SP26 031
     1           0.00, 0.15, 0.30, 0.41, 0.50, 0.63, 0.72,              SP26 032
     2           0.00, 0.25, 0.45, 0.60, 0.75, 0.95, 1.13,              SP26 033
     3           0.00, 0.37, 0.65, 0.90, 1.12, 1.50, 1.80/              SP26 034
      DATA TB15/ 0.00, 0.14, 0.27, 0.38, 0.46, 0.50, 0.50,              SP26 035
     1           0.00, 0.21, 0.40, 0.57, 0.67, 0.82, 0.90,              SP26 036
     2           0.00, 0.32, 0.58, 0.79, 0.95, 1.20, 1.42,              SP26 037
     3           0.00, 0.60, 0.97, 1.24, 1.48, 1.90, 2.31/              SP26 038
      DATA TC15/ 0.00, 0.09, 0.34, 0.46, 0.50, 0.50, 0.50,              SP26 039
     1           0.00, 0.26, 0.55, 0.69, 0.77, 0.89, 1.00,              SP26 040
     2           0.00, 0.48, 0.78, 0.98, 1.13, 1.42, 1.68,              SP26 041
     3           0.00, 0.77, 1.17, 1.50, 1.80, 2.39, 2.98/              SP26 042
      DATA TD15/ 0.26, 0.35, 0.43, 0.47, 0.50, 0.50, 0.50,              SP26 043
     1           0.26, 0.40, 0.51, 0.59, 0.65, 0.73, 0.80,              SP26 044
     2           0.26, 0.47, 0.63, 0.75, 0.85, 1.02, 1.15,              SP26 045
     3           0.26, 0.60, 0.84, 1.05, 1.21, 1.53, 1.85/              SP26 046
      DATA TE15/ 0.14, 0.30, 0.42, 0.47, 0.50, 0.50, 0.50,              SP26 047
     1           0.14, 0.38, 0.54, 0.64, 0.71, 0.81, 0.90,              SP26 048
     2           0.14, 0.48, 0.70, 0.87, 1.00, 1.23, 1.43,              SP26 049
     3           0.14, 0.67, 1.00, 1.27, 1.52, 1.96, 2.43/              SP26 050
      DATA TG15/ 0.50, 0.50, 0.50, 0.50, 0.50, 0.50, 0.50,              SP26 051
     1           0.50, 0.54, 0.58, 0.61, 0.65, 0.71, 0.77,              SP26 052
     2           0.50, 0.59, 0.68, 0.76, 0.84, 1.01, 1.16,              SP26 053
     3           0.50, 0.70, 0.88, 1.05, 1.21, 1.52, 1.83/              SP26 054
      DATA TH15/ 0.27, 0.41, 0.48, 0.50, 0.50, 0.50, 0.50,              SP26 055
     1           0.27, 0.46, 0.58, 0.66, 0.71, 0.81, 0.90,              SP26 056
     2           0.27, 0.58, 0.76, 0.88, 1.00, 1.20, 1.41,              SP26 057
     3           0.27, 0.74, 1.04, 1.28, 1.50, 1.96, 2.40/              SP26 058
      DATA TA16/ .000, .108, .162, .195, .215, .238, .250, .250,        SP26 059
     1           .000, .085, .128, .152, .168, .187, .194, .194,        SP26 060
     2           .000, .070, .110, .132, .147, .165, .170, .170,        SP26 061
     3           .000, .062, .099, .120, .135, .153, .160, .160/        SP26 062
      DATA TB16/ .000, .095, .155, .197, .223, .244, .250, .250,        SP26 063
     1           .000, .085, .140, .173, .194, .215, .221, .221,        SP26 064
     2           .000, .075, .125, .162, .183, .205, .210, .210,        SP26 065
     3           .000, .068, .115, .154, .175, .197, .204, .204/        SP26 066
      DATA TC16/ .000, .100, .158, .190, .210, .230, .250, .250,        SP26 067
     1           .000, .100, .158, .190, .210, .230, .250, .250,        SP26 068
     2           .000, .100, .158, .190, .210, .230, .250, .250,        SP26 069
     3           .000, .100, .158, .190, .210, .230, .250, .250/        SP26 070
      DATA TD16/ .250, .250, .250, .250, .250, .250, .250, .250,        SP26 071
     1           .250, .263, .275, .284, .290, .301, .305, .305,        SP26 072
     2           .250, .270, .286, .300, .309, .321, .325, .325,        SP26 073
     3           .250, .275, .294, .308, .317, .329, .335, .335/        SP26 074
      DATA TE16/ .140, .163, .190, .210, .225, .242, .250, .250,        SP26 075
     1           .140, .177, .210, .233, .250, .269, .275, .275,        SP26 076
     2           .140, .185, .220, .243, .260, .279, .285, .285,        SP26 077
     3           .140, .195, .230, .253, .268, .287, .291, .291/        SP26 078
```

```
      DATA TG16/ .500, .465, .432, .400, .370, .320, .286, .286,
     1           .500, .483, .467, .457, .440, .417, .409, .409,
     2           .500, .496, .492, .487, .483, .476, .473, .473,
     3           .500, .504, .507, .510, .511, .512, .514, .514/
      DATA TM16/ .250, .250, .250, .250, .250, .250, .250, .250,
     1           .250, .275, .292, .305, .315, .327, .330, .330,
     2           .250, .280, .304, .321, .336, .353, .360, .360,
     3           .250, .290, .315, .332, .353, .372, .380, .380/
      DO 1 J=1,7
      DO 1 K=1,4
      TF15(J,K)=TC15(J,K)
    1 TI15(J,K)=TC15(J,K)
      DO 2 J=1,8
      DO 2 K=1,4
      TF16(J,K)=TC16(J,K)
    2 TI16(J,K)=TC16(J,K)
      CTLE=(S-R)*TAN(SWPLE/RAD)
      SWPMC=RAD*ATAN((CTLE+0.5*(CT-CR))/(S-R))
      SWPTE=RAD*ATAN((CTLE+CT-CR)/(S-R))
      IF (FMACH.LE.1.0) GO TO 100
      IF (ARPARM.LT.4.0) GO TO 100
C
C     CHART 14
C
      IF (NAFTER.EQ.0) GO TO 50
C
C     CHART 14(A)
C
      IF (BDCR.GT.1.0) GO TO 25
      IF (BDCR.GT.0.4) GO TO 10
      XBCR=0.5+0.55*BDCR
      RETURN
   10 IF (BDCR.GT.0.8) GO TO 15
      XBCR=0.72+0.5*(BDCR-0.4)
      RETURN
   15 XBCR=0.92+0.4*(BDCR-0.8)
      RETURN
   25 IF (SMB.GT.0.2) GO TO 30
      XBCR=1.0+0.378*(BDCR-1.0)
      RETURN
   30 IF (SMB.GT.2.0) GO TO 35
      XBCR=1.0+0.4*(BDCR-1.0)
      RETURN
   35 XBCR=1.0+0.411*(BDCR-1.0)
      RETURN
C
C     CHART 14(B)
C
   50 IF (BDCR.LT.1.0) GO TO 55
      XBCR=0.667
      RETURN
   55 L=1
      IF (SMB.GT.1.0) L=2
      J=0
   60 J=J+1
      IF (BDCR.GT.BD14B(J)) GO TO 60
      JM=J-1
      XBCR=XB14B(JM,L)+(XB14B(J,L)-XB14B(JM,L))*(BDCR-BD14B(JM))/
     1(BD14B(J)-BD14B(JM))
      RETURN
C
C     CHART 15 AND 16
C
  100 IF (SWPMC.LE.0.0) GO TO 105
      LM=2
      LP=3
      DXCL=SWPMC/(SWPMC-SWPTE)
      GO TO 110
  105 LM=1
      LP=2
      DXCL=1.0
      IF (TAPER.NE.1.0) DXCL=SWPLE/(SWPLE-SWPMC)
  110 IF (TAPER.LE.0.5) GO TO 115
      KM=2
      KP=3
      DXSL=(TAPER-0.5)/0.5
      GO TO 120
  115 KM=1
      KP=2
      DXSL=TAPER/0.5
  120 IF (ROS.GT.0.2) GO TO 125
      JM=1
      JP=2
      DXRS=ROS/0.2
      GO TO 135
  125 IF (ROS.GT.0.4) GO TO 130
      JM=2
      JP=3
      DXRS=(ROS-0.2)/0.2
      GO TO 135
  130 JM=3
      JP=4
      DXRS=(ROS-0.4)/0.2
  135 IF (FMACH.LE.1.0) GO TO 200
C
C     CHART 15
C
      I=0
  140 I=I+1
      IF (BAR.GE.BA15(I)) GO TO 140
      IM=I-1
      IP=I
      DXBA=(BAR-BA15(IM))/(BA15(IP)-BA15(IM))
      DO 145 K=KM,KP
      DO 145 J=JM,JP
      VA(J,K,1)=XB15A(IM,J,K)+DXBA*(XB15A(IP,J,K)-XB15A(IM,J,K))
      VA(J,K,2)=XB15B(IM,J,K)+DXBA*(XB15B(IP,J,K)-XB15B(IM,J,K))
  145 VA(J,K,3)=XB15C(IM,J,K)+DXBA*(XB15C(IP,J,K)-XB15C(IM,J,K))
      DO 150 L=LM,LP
      DO 150 K=KM,KP
  150 V(K,L)=VA(JM,K,L)+DXRS*(VA(JP,K,L)-VA(JM,K,L))
      VALA=V(KM,LM)+DXSL*(V(KP,LM)-V(KM,LM))
      VALB=V(KM,LP)+DXSL*(V(KP,LP)-V(KP,LP))
      XBCR=VALA+DXCL*(VALB-VALA)
      RETURN
C
C     CHART 16
C
  200 I=0
  210 I=I+1
      IF (BAR.GE.BA16(I)) GO TO 210
      IM=I-1
      IP=I
      DXBA=(BAR-BA16(IM))/(BA16(IP)-BA16(IM))
      DO 220 K=KM,KP
      DO 220 J=JM,JP
      VA(J,K,1)=XB16A(IM,J,K)+DXBA*(XB16A(IP,J,K)-XB16A(IM,J,K))
      VA(J,K,2)=XB16B(IM,J,K)+DXBA*(XB16B(IP,J,K)-XB16B(IM,J,K))
  220 VA(J,K,3)=XB16C(IM,J,K)+DXBA*(XB16C(IP,J,K)-XB16C(IM,J,K))
      DO 230 L=LM,LP
      DO 230 K=KM,KP
  230 V(K,L)=VA(JM,K,L)+DXRS*(VA(JP,K,L)-VA(JM,K,L))
      VALA=V(KM,LM)+DXSL*(V(KP,LM)-V(KM,LM))
      VALB=V(KM,LP)+DXSL*(V(KP,LP)-V(KM,LP))
      XBCR=VALA+DXCL*(VALB-VALA)
      RETURN
      END
```

```
      FUNCTION EQ14(X)
C
C     CALCULATE KW(B) OR KT(B) FROM EQ. 14 OF REPORT 1307
C
      IF (X.LE.0.0) GO TO 10
      IF (X.GE.1.0) GO TO 20
      X2=X*X
      TA=1+X2*X2
      TB=0.5*ATAN(0.5*(1.0/X-X))+3.14159/4.0
      TC=X2*(1.0/X-X+2.0*ATAN(X))
      EQ14=2.0*(TA*TB-TC)/(3.14159*(1.0-X)**2)
      RETURN
   10 EQ14=1.0
      RETURN
   20 EQ14=2.0
      RETURN
      END
```

```
      FUNCTION EQ21(X)
C
C     CALCULATE KB(W) OR KB(T) FROM EQ 21 OF REPORT 1307
C
      IF (X.LE.0.0) GO TO 10
      IF (X.GE.1.0) GO TO 20
      EQ21=(1.0-X*X)**2/(1.0-X)**2-EQ14(X)
      RETURN
   10 EQ21=0.0
      RETURN
   20 EQ21=2.0
      RETURN
      END
```

```
      FUNCTION EQ24(BM,TAPER,BDCR,ROB)
C
C     CALCULATE KB(W)*BCLAW OR KB(T)*BCLAT FROM EQ 24 FOR M GREATER THAN
C          1.0, VALUE OF THE ASPECT RATIO PARAMETER GREATER THAN 4, AND
C          MB GREATER THAN 1
C
      IF (ROB.LE.0.0) GO TO 10
      IF (ROB.GE.1.0) GO TO 20
      TA=1.0+BM
      TB=TA*BDCR
      TC=SQRT(BM*BM-1.0)
      TD=BM/TA
      TE=TD*(((TB+BM)/BM)**2)*ARCOS((1.0+TB)/(BM+TB))
      TF=TC*(SQRT(1.0+2.0*BDCR)-1.0)/TA
      TG=1.0+1.0/BDCR
      TH=TC*BDCR*BDCR*ALOG(TG+SQRT(TG*TG-1.0))/BM
      TI=BM*ARCOS(1.0/BM)/TA
      EQ24=8.0*BM*(TE+TF-TH-TI)/(3.14159*(1.0+TAPER)*BDCR*(1.0/ROB-1.0))
      EQ24=EQ24/TC
      RETURN
   10 EQ24=0.0
      RETURN
   20 EQ24=2.0
      RETURN
      END
```

```
      FUNCTION EQ24L(TAPER,BDCR,ROS)
C
C     LIMITING FORM OF EQ 24 WHEN BM EQUAL INFINITY, NO L.E. SWEEP
C
      IF (ROS.LE.0.0) GO TO 10
      IF (ROS.GE.1.0) GO TO 20
      TA=1.0+BDCR
      TB=1.0+1.0/BDCR
      TC=TA*TA*ARCOS(BDCR/TA)+SQRT(1.0+2.0*BDCR)-1.0-BDCR*BDCR*ALOG(TB+
     1SQRT(TB*TB-1.0))-1.5708
      EQ24L=8.0*TC/(3.14159*(1.0+TAPER)*BDCR*(1.0/ROS-1.0))
      RETURN
   10 EQ24L=0.0
      RETURN
   20 EQ24L=2.0
      RETURN
      END
```

```
      FUNCTION EQ26(BM,TAPER,BDCR,ROB)
C
C     CALCULATE KB(W)*BCLAW OR KB(T)*BCLAT FROM EQ 26 FOR M GREATER THAN
C          1.0, VALUE OF ASPECT RATIO PARAMETER GREATER THAN 4, AND
C          MB LESS THAN OR EQUAL TO 1
C
      IF (ROB.LE.0.0) GO TO 10
      IF (ROB.GE.1.0) GO TO 20
      TA=BM/(1.0+BM)
      TB=BDCR/TA
      TC=1.0+TB
      TD=SQRT(1.0/TC)
      TE=SQRT(TC)*(TC+1.0)-2.0-TB*TB*(0.5*ALOG(1.0+TD)-0.5*ALOG(1.0-TD))
      EQ26=16.0*TA*TA*TE/(3.14159*(1.0+TAPER)*BDCR*(1.0/ROB-1.0))
      RETURN
   10 EQ26=0.0
      RETURN
   20 EQ26=2.0
      RETURN
      END
```

```
      FUNCTION EQ30(BM,TAPER,BDCRA,ROB)
C
C     CALCULATE KB(W)*BCLAW OR KB(T)*BCLAT FROM EQ 30 FOR M GREATER THAN
C          1.0, VALUE OF THE ASPECT RATIO PARAMETER GREATER THAN 4, AND
C          MB GREATER THAN 1.  NO AFTERBODY.
C
      IF (ROB.LE.0.0) GO TO 10
      IF (ROB.GE.1.0) GO TO 20
      BDCR=BDCRA
      RRF=1.0
      IF (BDCR.LE.1.0) GO TO 1
      BDCR=1.0
      RRF=BDCRA
    1 TA=SQRT(BM*BM-1.0)
      TB=1.0+BM/BDCR
      TC=1.0/BDCR
      TD=TC*TC
      TE=TB*TB*ARCOS((BM+TC)/TB)-BM*BM*TD*ARCOS(1.0/BM)+BM*TD*TA*ARSIN(B
     1DCR)-TA*ALOG(TC+SQRT(TC*TC-1.0))
      EQ30=8.0*BDCR*TE/(3.14159*TA*(TAPER+1.0)*(1.0/ROB-1.0)*RRF)
      RETURN
   10 EQ30=0.0
      RETURN
   20 EQ30=2.0
      RETURN
      END
```

```
      FUNCTION EQ30L(TAPER,BDCRA,ROS)
C
C     LIMITING FORM OF EQUATION 30 WHEN BM EQUAL INFINITY, AT L.E. SWEEP
C
      IF (ROS.LE.0.0) GO TO 10
      IF (ROS.GE.1.0) GO TO 20
      BDCR=BDCRA
      RRF=1.0
      IF (BDCR.LT.1.0) GO TO 1
      TC=1.0
      RRF=BDCRA
    3 EQ30L=8.0*TC/(3.14159*(TAPER+1.0)*(1.0/ROS-1.0)*RRF)
      RETURN
    1 TA=1.0/BDCR
      TB=BDCR**2
      TC=2.0*ARCOS(BDCR)-BDCR*ALOG(TA+SQRT(TA*TA-1.0))+BDCR
      TD=1.0-TB
      TE=TD*BDCR/2.0
      SUM=TE
      FN=1.0
    2 FN=FN+2.0
      TE=TD*FN*TB*TE/(FN+1.0)
      SUM=SUM+TE
      IF (TE.GT.0.00001) GO TO 2
      TC=TC-SUM
      GO TO 3
   10 EQ30L=0.0
      RETURN
   20 EQ30L=2.0
      RETURN
      END
```

```
      FUNCTION EQ31(BM,TAPER,BDCRA,ROB)
C
C     CALCULATE KB(N)*BCLAN OR KB(T)*BCLAT FROM EQ 31 FOR M GREATER THAN
C          1.0, VALUE OF ASPECT RATIO PARAMETER GREATER THAN 4, AND
C          MB GREATER THAN 1.  NO AFTERBODY.
C
      IF (ROB.LE.0.0) GO TO 10
      IF (ROB.GE.1.0) GO TO 20
      BDCR=BDCRA
      RRF=1.0
      IF (BDCR.LE.1.0) GO TO 1
      BDCR=1.0
      RRF=BDCRA
    1 TA=SQRT(BM)
      TB=1.0+BM/BDCR
      TC=1.0/BDCR-1.0
      TD=1.0+BM
      TE=TB-SQRT(TC*TB)
      TF=((1.0/BDCR)**2)*BM*TA
      TG=BM*((1.0/BDCR)**2)*TD*(ATAN(SQRT(1.0/BM))-ATAN(SQRT(TC/TB)))
      TH=SQRT(BM*TC/TB)
      TI=TD*(0.5*ALOG(1.0+TH)-0.5*ALOG(1.0-TH))/TA
      EQ31=16.0*TA*BDCR*(TE-TF+TG-TI)/(3.14159*TD*(TAPER+1.0)*(1.0/ROB
     1-1.0)*RRF)
      RETURN
   10 EQ31=0.0
      RETURN
   20 EQ31=2.0
      RETURN
      END
```

```
      FUNCTION CHRT8 (BAR,S,R,CT,CR,SWPLE)
C
C     CALCULATE BETA*D(CL)/D(ALPHA) FOR SUPERSONIC MACH NUMBERS, CHART 8
C          OF REPORT 1307
C
      DIMENSION BA(11),CL(3),CH8(11,4)
      DATA RAD/57.29578/
      DATA BA/0.0,0.5,1.0,1.5,2.0,2.5,3.0,4.0,5.0,6.0,10.0/
      DATA CH8/
     1 0.0, 0.75, 1.45, 2.07, 2.59, 3.02, 3.41, 4.00, 4.00, 4.00, 4.00,
     2 0.0, 0.86, 2.09, 2.73, 3.41, 3.55, 3.65, 3.76, 3.82, 3.86, 4.00,
     3 0.0, 0.80, 1.81, 2.65, 3.12, 3.36, 3.52, 3.70, 3.79, 3.82, 4.00,
     4 0.0, 0.86, 2.02, 2.67, 3.00, 3.20, 3.34, 3.51, 3.60, 3.65, 4.00/
      IF (BAR.LE.10.0) GO TO 10
      CHRT8=4.0
      RETURN
   10 I=0
   11 I=I+1
      IF (BAR.GE.BA(I)) GO TO 11
      IM=I-1
      IP=I
      DXBA=(BAR-BA(IM))/(BA(IP)-BA(IM))
      TAPER=CT/CR
      IF (TAPER.LT.0.5) GO TO 20
      DXSL=(TAPER-0.5)/0.5
      TA=CH8(IM,3)+DXBA*(CH8(IP,3)-CH8(IM,3))
      TB=CH8(IM,4)+DXBA*(CH8(IP,4)-CH8(IM,4))
      CHRT8=TA+DXSL*(TB-TA)
      RETURN
   20 CTLE=(S-R)*TAN(SWPLE/RAD)
      SWPMC=RAD*ATAN((CTLE+0.5*(CT-CR))/(S-R))
      SWPTE=RAD*ATAN((CTLE+CT-CR)/(S-R))
      DXSL=TAPER/0.5
      IF (SWPMC.LE.0.0) GO TO 30
      JM=2
      JP=1
      DXCL=SWPMC/(SWPMC-SWPTE)
      GO TO 40
   30 JM=1
      JP=2
      DXCL=SWPLE/(SWPLE-SWPMC)
   40 DO 50 K=1,3
   50 CL(K)=CH8(IM,K)+DXBA*(CH8(IP,K)-CH8(IM,K))
      TA=CL(JM)+DXCL*(CL(JP)-CL(JM))
      TB=CL(3)
      CHRT8=TA+DXSL*(TB-TA)
      RETURN
      END
```

COMPONENTS	TYPE	NORMAL-FORCE COEFFICIENT	LIFT COEFFICIENT	PITCHING-MOMENT COEFFICIENT	CENTER OF PRESSURE LOCATION	AXIAL-FORCE COEFFICIENT
NOSE	Potential	$C_{N_{N,p}}$	$C_{L_{N,p}}$	$C_{m_{N,p}}$	$\bar{x}_{N,p}$	--------
	Viscous	$C_{N_{N,v}}$	$C_{L_{N,v}}$	$C_{m_{N,v}}$	$\bar{x}_{N,v}$	--------
WING IN PRESENCE OF BODY	Potential	$C_{N_{W(B),p}}$	$C_{L_{W(B),p}}$	$C_{m_{W(B),p}}$	$\bar{x}_{W(B),p}$	$C_{A_{W(B),p}}$
	Viscous	$C_{N_{W(B),v}}$	$C_{L_{W(B),v}}$	$C_{m_{W(B),v}}$	$\bar{x}_{W(B),v}$	$C_{A_{W(B),v}}$
BODY IN PRESENCE OF WING	Potential	$C_{N_{B(W),p}}$	$C_{L_{B(W),p}}$	$C_{m_{B(W),p}}$	$\bar{x}_{B(W),p}$	--------
	Viscous	$C_{N_{B(W),v}}$	$C_{L_{B(W),v}}$	$C_{m_{B(W),v}}$	$\bar{x}_{B(W),v}$	--------
AFTERBODY	-------	C_{N_A}	C_{L_A}	C_{m_A}	$\bar{x}_A$	--------
TAIL IN PRESENCE OF BODY	Potential	$C_{N_{T(B),p}}$	$C_{L_{T(B),p}}$	$C_{m_{T(B),p}}$	$\bar{x}_{T(B),p}$	$C_{A_{T(B),p}}$
	Viscous	$C_{N_{T(B),v}}$	$C_{L_{T(B),v}}$	$C_{m_{T(B),v}}$	$\bar{x}_{T(B),v}$	$C_{A_{T(B),v}}$
BODY IN PRESENCE OF TAIL	Potential	$C_{N_{B(T),p}}$	$C_{L_{B(T),p}}$	$C_{m_{B(T),p}}$	$\bar{x}_{B(T),p}$	--------
	Viscous	$C_{N_{B(T),v}}$	$C_{L_{B(T),v}}$	$C_{m_{B(T),v}}$	$\bar{x}_{B(T),v}$	--------
COMPLETE CONFIGURATION	-------	C_N	C_L	C_m	$\bar{x}$	C_A

Table I.- Summary of force and moment coefficient notation.

ITEM 1 FORMAT (18I5)

1	6	11	16	21	26	31	36	41	46	51	56	61	66	71	76
NHEAD	NTBL	NPRINT	MPRINT	NOSEV	NREGNW	NALPW	NSEPW	NREGNT	NALPT	NSEPT	NBODY	NAFT			

ITEM 2 FORMAT (20A4); NHEAD cards

1 TITLE NOTE: Last card saved for additional output in various subroutines.

ITEM 3 FORMAT (8F10.5)

1	11	21	31	41	51	61
EM	REFS	REFL	XM	THETAN	DXOUT	DXI

ITEM 4 FORMAT (2F10.5); NTBL cards

1	11
XBDY(J)	RBDY(J)

ITEM 5 FORMAT (5F10.5)

1	11	21	31	41
RAVGW	YSEPW	XHLW	ZHLW	XWCP

ITEM 6 NAMELIST, NREGNW blocks

2

```
$INPUT
PER=0.95,
SECT=
ROWS=
COLS=
ROOTLE=
ROOTTE=
ROOTY=
TIPLE=
TIPTE=
TIPY=
TCROOT=
TCTIP=
$END
```

(a) Page 1.

Figure 1.- Input format for SUPSON program.

ITEM 7

Omit Item 7 if NALPW = 0.

FORMAT (8F10.5)

1 ALPHAW(I)	11	21	31	41	51	61	71

Go to Item 11 if NREGNT = 0.

ITEM 8

1 RAVGT	11 YSEPT	21 XHLT	31 ZHLT	41 XTCP

ITEM 9

NAMELIST, NREGNT blocks

2

```
$INPUT
PER=0.95,
SECT=
ROWS=
COLS=
ROOTLE=
ROOTTE=
ROOTY=
TIPLE=
TIPTE=
TIPY=
TCROOT=
TCTIP=
$END
```

ITEM 10

Omit Item 10 if NALPT = 0.

FORMAT (8F10.5)

1 ALPHAT(I)	11	21	31	41	51	61	71

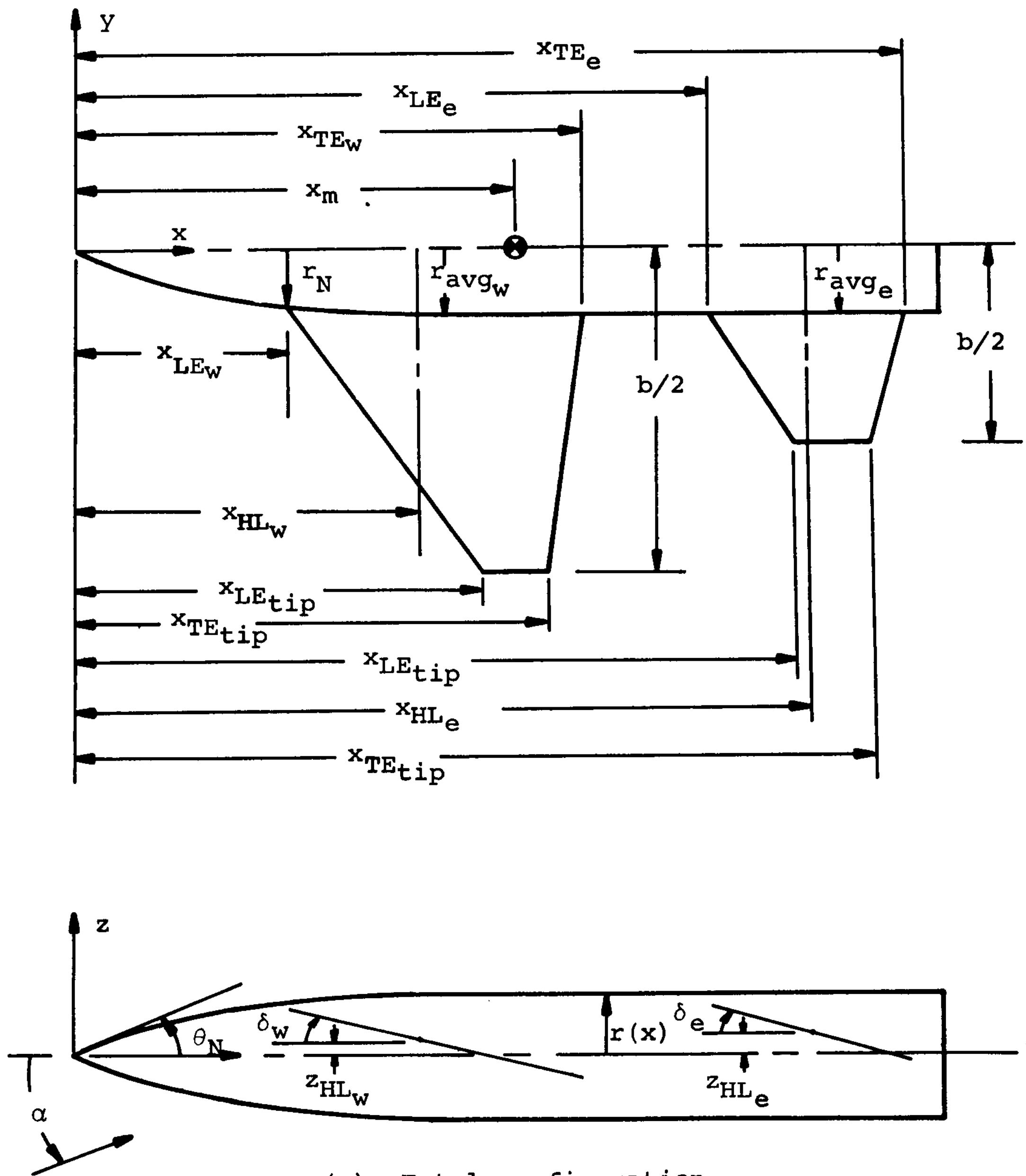
ITEM 11

FORMAT (I2,F6.2,9F8.3)

1 NDEX	3 ALPHAD	9 ALPIW	17 AKVLW1	25 AKVLW2	33 AKVSW	41 ALPIT	49 AKVLT1	57 AKVLT2	65 AKVST	73 WLIMIT

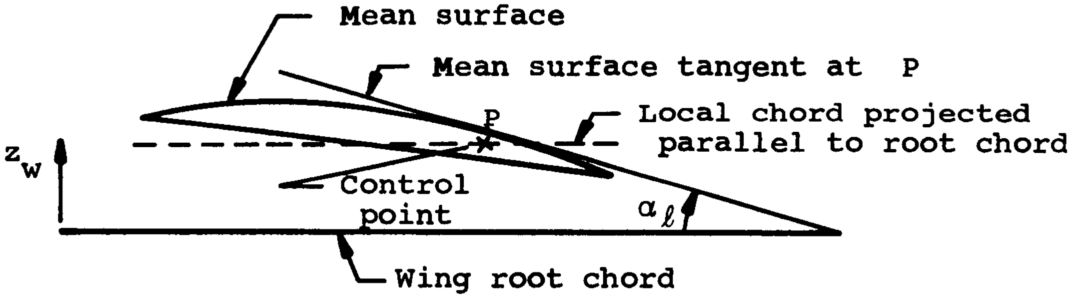
(b) Page 2.

Figure 1.- Concluded.



(a) Total configuration.

Figure 2.- Geometric nomenclature for SUPSON program.



(b) Detail of local wing section.

Figure 2.- Concluded.

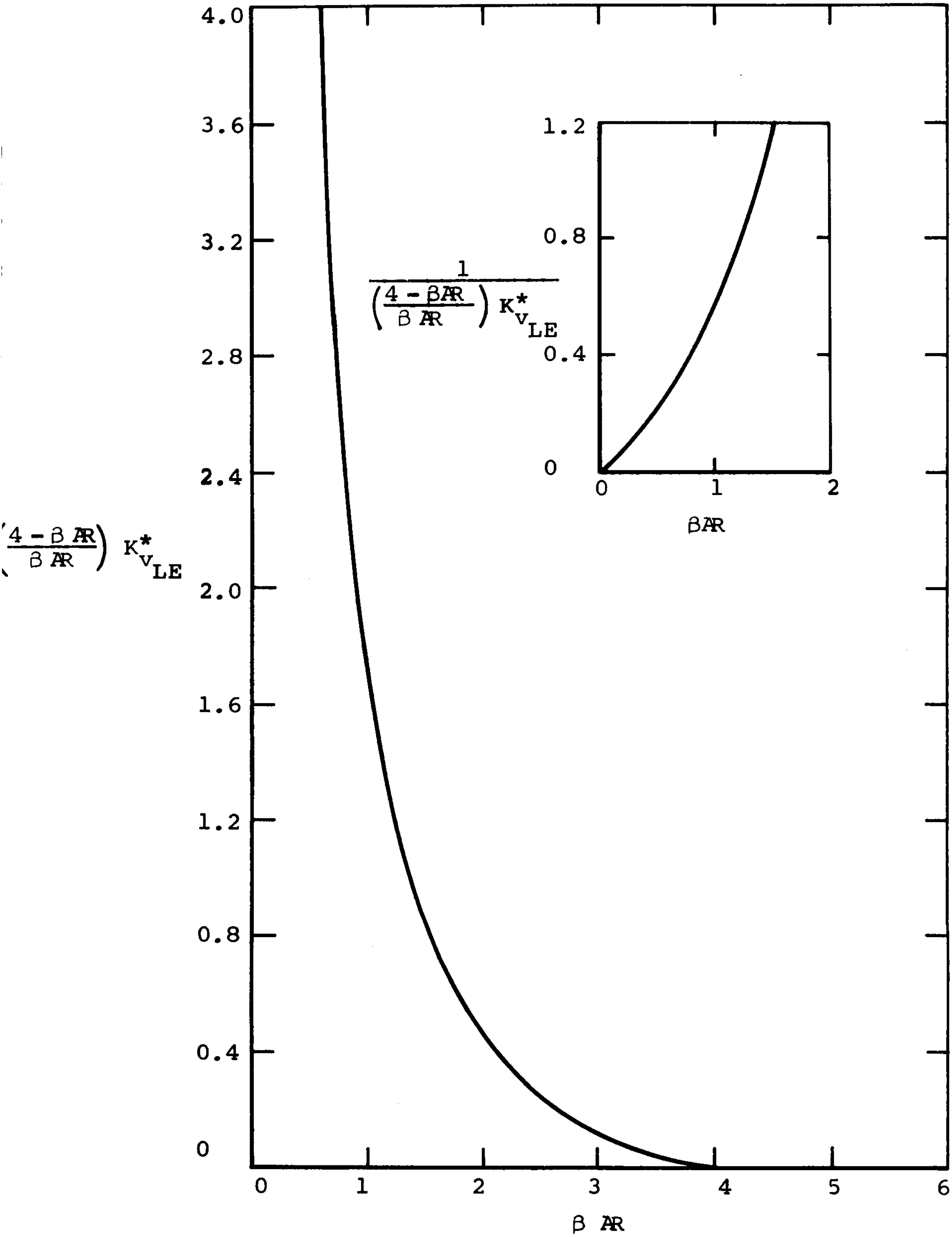


Figure 3.- Vortex-lift ratio on delta wings in supersonic flow.

```
  2   20     0     0     1     1     0     1     1     0     1     0     0
SAMPLE CASE 1    REF. NASA TM X-328    SEPT. 1960    C. DRIVER
  CANARD - WING - BODY  CONFIGURATION
2.01        192.0       15.33     25.0       10.        1.         1.
0.          0.
.297        .076
.956        .233
1.945        .445
2.605       .573
3.267       .682
3.929       .78
4.275       .826
5.255       .94
5.92        .996
6.583       1.042
11.375       1.31
17.749        1.667
17.75         1.667
17.8          1.667
18.61         1.667
25.           1.667
30.           1.667
37.           1.667
38.         1.667
 1.34       3.37        9.125      0.          0.
+INPUT
PER= 0.95,
ROWS= 4,
COLS= 4,
ROOTLE= 4.275,
ROOTTE=11.375,
ROOTY= 1.34,
TIPLE=11.375,
TIPTE=11.375,
TIPY= 3.37,
TCROOT= 0.0,
TCTIP= 0.0,
+END
 1.667      5.0135      25.        0.          0.
+INPUT
PER= 0.95,
ROWS= 6,
COLS=10,
ROOTLE=18.61,
ROOTTE=37.0,
ROOTY= 1.667,
TIPLE=37.0,
TIPTE=37.0,
TIPY= 8.36,
TCROOT= 0.0,
TCTIP= 0.0,
+END
1 00.0  10.      .5      0.      0.      0.      .5      0.      0.      1.
1 02.0  10.      .5      0.      0.      0.      .5      0.      0.      1.
1 04.0  10.      .5      0.      0.      0.      .5      0.      0.      1.
1 06.0  10.      .5      0.      0.      0.      .5      0.      0.      1.
1 08.0  10.      .5      0.      0.      0.      .5      0.      0.      1.
1 10.0  10.      .5      0.      0.      0.      .5      0.      0.      1.
1 12.0  10.      .5      0.      0.      0.      .5      0.      0.      1.
1 16.0  10.      .5      0.      0.      0.      .5      0.      0.      1.
1 20.0  10.      .5      0.      0.      0.      .5      0.      0.      1.
1 20.0  10.      .0      0.      0.      0.      .0      0.      0.      1.
```

(a) Sample case 1.

Figure 4.- Sample input decks for SUPSON program.

```
  2   20     0     0     1     2     0     1     0     0     0     0     0
 SAMPLE CASE  2    PROGRAM SUPSON
    WING - BODY  CONFIGURATION         REF. NASA TM X-328  C. DRIVER
2.01       192.0       15.33    25.0     10.       1.        1.
0.        0.
.297      .076
.956      .233
1.945      .445
2.605     .573
3.267     .682
3.929     .78
4.275     .826
5.255     .94
5.92      .996
6.563     1.042
11.375     1.31
17.749       1.667
17.75        1.667
17.8         1.667
18.61        1.667
25.          1.667
30.          1.667
37.          1.667
38.       1.667
1.667     8.36       25.       0.        0.
$INPUT
ROWS=6,
COLS=5,
SECT=1,
ROOTLE=18.61,
ROOTTE=37.0,
ROOTY=1.667,
TIPLE=27.805,
TIPTE=37.0,
TIPY=5.0135,
TCROOT=0.0,
TCTIP=0.0,
$END
$INPUT
ROWS=6,
COLS=5,
SECT=1,
ROOTLE=27.805,
ROOTTE=37.0,
ROOTY=5.0135,
TIPLE=37.0,
TIPTE=37.0,
TIPY=8.36,
TCROOT=0.0,
TCTIP=0.0,
$END
1  10.  0.      .5      0.0     0.0     0.0     0.0     0.0     0.0     1.0
```

(b) Sample case 2.

Figure 4.- Continued.

```
  2   14   -2    0    1    1    0    1    1    0    1    0    0
SAMPLE CASE 3  REF. NASA MEMO 6-11-59A  MAY 1959   PETERSON AND MENEES
 WING - BODY - TAIL  CONFIGURATION
1.30      349.      11.32      23.59      12.     1.       1.
0.        .0
5.        .984
10.       1.54
15.       1.93
20.29     2.2
25.16     2.34
29.75     2.38
31.11     2.376
34.26     2.34
38.       2.24
40.85     2.13
42.43     2.05
46.11     1.82
47.       1.75
2.38      16.42     23.59     0.        0.
+INPUT
PER=0.95,
ROWS=6,
COLS=10,
ROOTLE=20.29,
ROOTTE=34.26,
ROOTY=2.38,
TIPLE=25.16,
TIPTE=31.11,
TIPY=16.42,
TCROOT=0.0,
TCTIP=0.0,
+END
 2.       8.34      42.43      0.       0.
+INPUT
PER=0.95,
ROWS=4,
COLS=8,
ROOTLE=40.85,
ROOTTE=46.11,
ROOTY=2.0,
TIPLE=41.8,
TIPTE=43.89,
TIPY =8.34,
TCROOT=0.0,
TCTIP=0.0,
+END
1   2.    0.     0.     0.     0.     0.     0.     0.     0.     1.
1   4.    0.     0.     0.     0.     0.     0.     0.     0.     1.
1   6.    0.     0.     0.     0.     0.     0.     0.     0.     1.
1   8.    0.     0.     0.     0.     0.     0.     0.     0.     1.
1  10.    0.     0.     0.     0.     0.     0.     0.     0.     1.
1  12.    0.     0.     0.     0.     0.     0.     0.     0.     1.
1  16.    0.     0.     0.     0.     0.     0.     0.     0.     1.
1   2.    0.     0.     0.     0.   -10.     0.     0.     0.     1.
1   4.    0.     0.     0.     0.   -10.     0.     0.     0.     1.
1   6.    0.     0.     0.     0.   -10.     0.     0.     0.     1.
1   8.    0.     0.     0.     0.   -10.     0.     0.     0.     1.
1  10.    0.     0.     0.     0.   -10.     0.     0.     0.     1.
1  12.    0.     0.     0.     0.   -10.     0.     0.     0.     1.
1  16.    0.     0.     0.     0.   -10.     0.     0.     0.     1.
1   2.    0.     0.     0.     0.   -20.     0.     0.     0.     1.
1   4.    0.     0.     0.     0.   -20.     0.     0.     0.     1.
1   6.    0.     0.     0.     0.   -20.     0.     0.     0.     1.
1   8.    0.     0.     0.     0.   -20.     0.     0.     0.     1.
1  10.    0.     0.     0.     0.   -20.     0.     0.     0.     1.
1  12.    0.     0.     0.     0.   -20.     0.     0.     0.     1.
1  16.    0.     0.     0.     0.   -20.     0.     0.     0.     1.
```

(c) Sample case 3.

Figure 4.- Concluded.

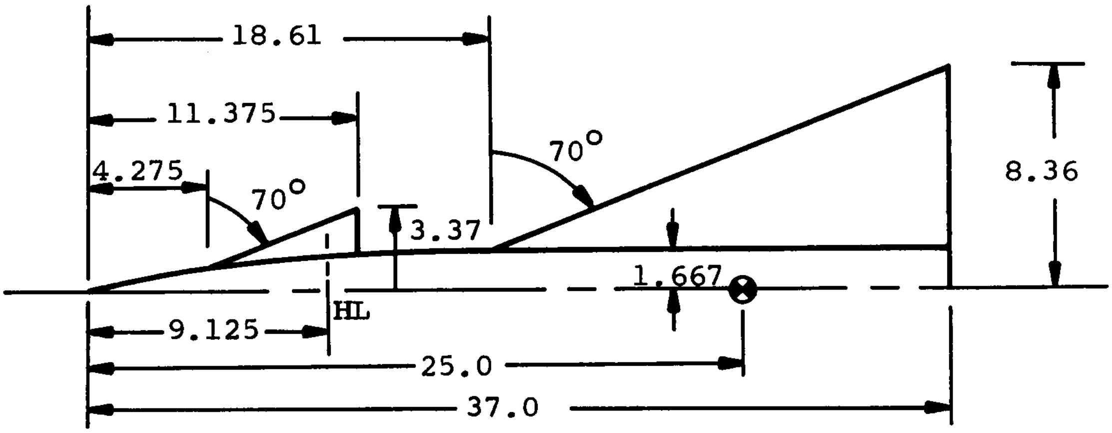


Figure 5.- Canard-body-wing configuration
for sample cases 1 and 2.

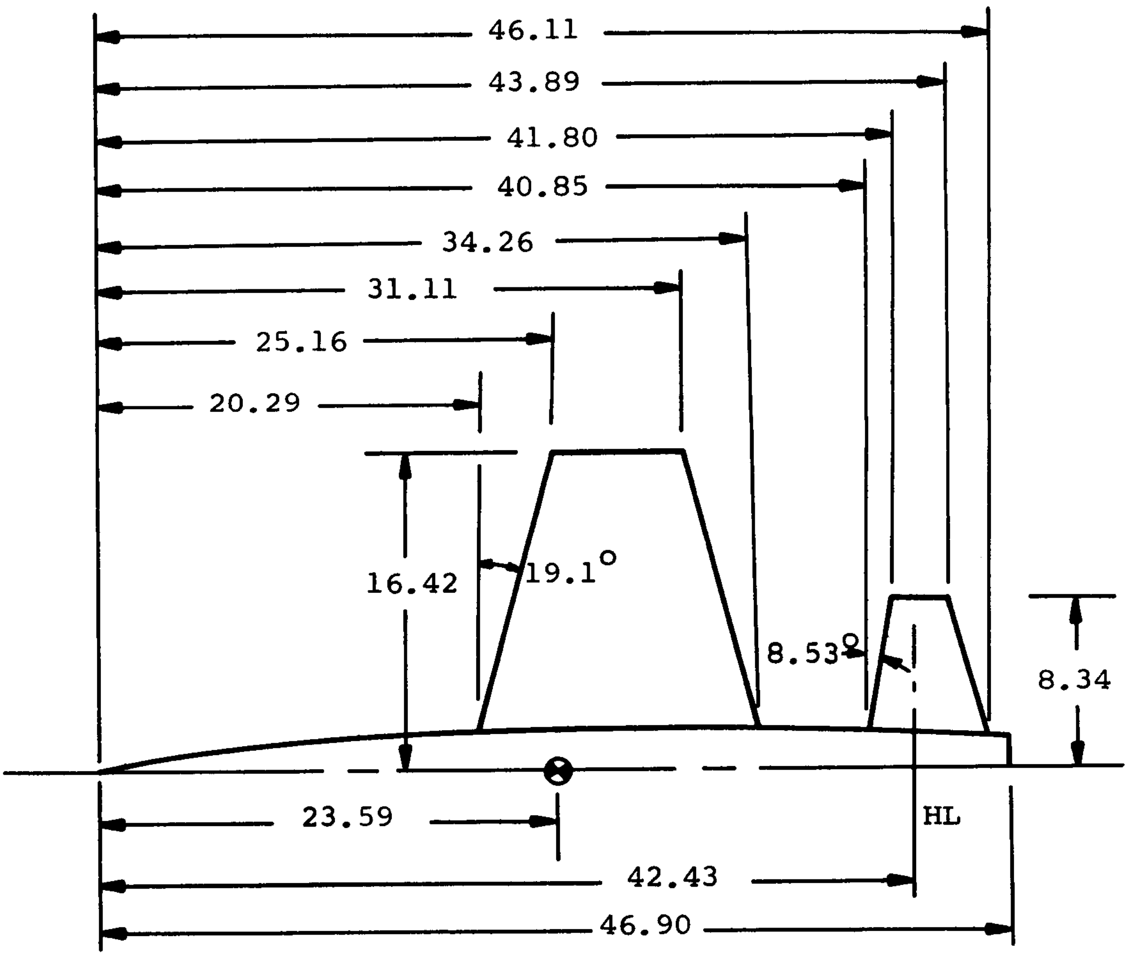


Figure 6.- Wing-body-tail configuration for sample case 3.

```
&INPUTM
NHEAD= 2
NTBL= 20
NPRINT= 0
MPRINT= 0
NOSEV= 1
NREGNW= 1
NALPW= 0
NSEPW= 1
NREGNT= 1
NALPT= 0
NSEPT= 1
NBODY= 0
NAFT= 0
&END
              SAMPLE CASE 1    REF.  NASA TM X-328   SEPT. 1960   C. DRIVER
                CANARD - WING - BODY  CONFIGURATION

                  M           REFS        REFL        XCM       DXOUT       DXI
               2.01000  192.00000  15.33000  25.00000    1.00000    1.00000    1.74359
```

(a) Page 1.

Figure 7.- Output from SUPSON program
for sample case 1.

```
&INPUT
NSF= 1
ROWS= 4
COLS= 4
PER= 0.950
ROOTLE= 4.2750
ROOTTE= 11.3750
ROOTY= 1.339999
TIPLE= 11.3750
TIPTE= 11.3750
TIPY= 3.370
TCROOT= 0.0
TCTIP= 0.0
ASYM= F
SECT= 1
ALT= -1.0
SCALE= 1.0
TYPE= 4
&END
```

(b) Page 2.

Figure 7.- Continued.

```
&INPUT
NSF= 2
ROWS= 6
COLS= 10
PER= 0.950
ROOTLE= 18.60999
ROOTTE= 37.0
ROOTY= 1.6670
TIPLE= 37.0
TIPTE= 37.0
TIPY= 8.360
TCROOT= 0.0
TCTIP= 0.0
ASYM= F
SECT= 1
ALT= -1.0
SCALE= 1.0
TYPE= 4
&END
```

(c) Page 3.

Figure 7.- Continued.

CANARD - WING - BODY CONFIGURATION

WING

XLE	XTE	XLE(TIP)	XTE(TIP)	CROOT	CTIP	B/2
4.27500	11.37500	11.37500	11.37500	7.10000	0.00000	3.37000

RAVG	YSEP	XHL	ZHL	XCP
1.34000	3.37000	9.12500	0.00000	0.00000

TAIL

XLE	XTE	XLE(TIP)	XTE(TIP)	CROOT	CTIP	B/2
18.60999	37.00000	37.00000	37.00000	18.39001	0.00000	8.36000

RAVG	YSEP	XHL	ZHL	XCP
1.66700	8.36000	25.00000	0.00000	0.00000

BODY ...	---------- NOSE ----------			AVERAGE RADIUS		CENTER OF MOMENTS			
	THETA	FINENESS	R(BASE)	WING	TAIL	X	Z	DXI	DXOUT
	10.000	2.588	0.826	1.340	1.667	25.000	0.000	1.000	1.000

TABLE OF BODY COORDINATES

	X	R	S	DR/DX
1	0.00000	0.00000	0.00000	0.25589
2	0.29700	0.07600	0.07600	0.24707
3	0.95600	0.23300	0.23300	0.22630
4	1.94500	0.44500	0.44500	0.20415
5	2.60500	0.57300	0.57300	0.17930
6	3.26700	0.68200	0.68200	0.15634
7	3.92900	0.78000	0.78000	0.14049
8	4.27500	0.82600	0.82600	0.11690
9	6.05000	1.00502	1.84750	0.08041
10	7.82500	1.11146	2.35500	0.05795
11	9.60000	1.21073	2.86250	0.05593
12	11.37500	1.31000	3.37000	0.05561
13	11.37600	1.31005	1.31005	0.05565
14	17.74899	1.66700	1.66700	0.02800
15	17.75000	1.66700	1.66700	0.00000
16	17.79999	1.66700	1.66700	0.00000
17	18.60999	1.66700	1.66700	0.00000
18	20.44897	1.66700	2.33630	0.00000
19	22.28798	1.66700	3.00560	0.00000
20	24.12698	1.66700	3.67490	0.00000
21	25.96597	1.66700	4.34420	0.00000
22	27.80498	1.66700	5.01350	0.00000
23	29.64397	1.66700	5.68280	0.00000
24	31.48297	1.66700	6.35209	0.00000
25	33.32198	1.66700	7.02139	0.00000
26	35.16096	1.66700	7.69069	0.00000
27	36.99998	1.66700	8.36000	0.00000
28	37.00099	1.66700	1.66700	0.00000
29	38.00000	1.66700	1.66700	0.00000

(d) Page 4.

Figure 7.- Continued.

ALPHA	M	-- INCIDENCE -- WING	TAIL	---- WING ----- KVLE*	KVSE*	---- TAIL ----- KVLE*	KVSE*	W/V LIMIT
16.00	2.01	10.000	0.000	0.500	0.000	0.500	0.000	1.000

NOSE VORTICES -	GAM/2*PI*V*RB	Y/RB	Z/RB	XS/RB
	0.0906	0.646	1.311	0.000

SUMMARY OF VORTEX STRENGTHS AND POSITION AT X = 4.275

VORTEX	GAMMA/2*PI*V	Y	Z
1	0.074799	0.533	1.083

VELOCITY INDUCED AT SPECIFIED FIELD POINTS ON WING (BESKIN)

	X	Y	Z	V/V(INF)	W/V(INF)
1	6.606	1.582	0.437	5.7449E-02	9.5909E-02
2	8.169	1.582	0.166	2.9297E-02	1.3809E-01
3	9.733	1.582	-0.106	-2.1923E-02	1.6345E-01
4	11.297	1.582	-0.377	-8.1243E-02	1.6068E-01
5	7.946	2.084	0.205	1.5486E-02	7.8089E-02
6	9.070	2.084	0.009	8.1490E-04	8.9664E-02
7	10.195	2.084	-0.186	-1.7450E-02	9.7132E-02
8	11.319	2.084	-0.381	-3.7576E-02	9.9363E-02
9	9.270	2.581	-0.025	-1.1602E-03	5.9592E-02
10	9.960	2.581	-0.145	-7.0939E-03	6.2933E-02
11	10.650	2.581	-0.265	-1.3584E-02	6.5480E-02
12	11.340	2.581	-0.385	-2.0474E-02	6.7140E-02
13	10.473	3.032	-0.234	-7.3537E-03	4.7348E-02
14	10.769	3.032	-0.285	-9.1522E-03	4.8180E-02
15	11.064	3.032	-0.337	-1.1005E-02	4.8926E-02
16	11.360	3.032	-0.388	-1.2907E-02	4.9582E-02

VELOCITY INDUCED AT SPECIFIED FIELD POINTS ON WING (VORTEX)

	X	Y	Z	V/V(INF)	W/V(INF)
1	6.606	1.582	0.437	1.6056E-02	-5.3901E-03
2	8.169	1.582	0.166	7.6823E-03	-7.9791E-03
3	9.733	1.582	-0.106	2.7704E-03	-6.3888E-03
4	11.297	1.582	-0.377	6.1840E-04	-4.2710E-03
5	7.946	2.084	0.205	7.2136E-03	-2.1890E-03
6	9.070	2.084	0.009	5.0521E-03	-3.0338E-03
7	10.195	2.084	-0.186	3.3162E-03	-3.0911E-03
8	11.319	2.084	-0.381	2.0716E-03	-2.7641E-03
9	9.270	2.581	-0.025	4.0101E-03	-9.7394E-04
10	9.960	2.581	-0.145	3.3966E-03	-1.2613E-03
11	10.650	2.581	-0.265	2.8322E-03	-1.4172E-03
12	11.340	2.581	-0.385	2.3305E-03	-1.4747E-03
13	10.473	3.032	-0.234	2.5851E-03	-4.9036E-04
14	10.769	3.032	-0.285	2.4367E-03	-5.6838E-04
15	11.064	3.032	-0.337	2.2920E-03	-6.3389E-04
16	11.360	3.032	-0.388	2.1511E-03	-6.8815E-04

TOTAL INDUCED VELOCITY AT SPECIFIED FIELD POINTS ON WING

	X	Y	Z	V/V(INF)	W/V(INF)
1	6.606	1.582	0.437	1.6056E-02	9.0519E-02
2	8.169	1.582	0.166	7.6823E-03	1.3011E-01
3	9.733	1.582	-0.106	2.7704E-03	1.5706E-01
4	11.297	1.582	-0.377	6.1840E-04	1.5641E-01
5	7.946	2.084	0.205	7.2136E-03	7.5900E-02
6	9.070	2.084	0.009	5.0521E-03	8.6630E-02
7	10.195	2.084	-0.186	3.3162E-03	9.4041E-02
8	11.319	2.084	-0.381	2.0716E-03	9.6599E-02
9	9.270	2.581	-0.025	4.0101E-03	5.8618E-02
10	9.960	2.581	-0.145	3.3966E-03	6.1672E-02
11	10.650	2.581	-0.265	2.8322E-03	6.4063E-02
12	11.340	2.581	-0.385	2.3305E-03	6.5665E-02
13	10.473	3.032	-0.234	2.5851E-03	4.6858E-02
14	10.769	3.032	-0.285	2.4367E-03	4.7611E-02
15	11.064	3.032	-0.337	2.2920E-03	4.8292E-02
16	11.360	3.032	-0.388	2.1511E-03	4.8894E-02

(e) Page 5.

Figure 7.- Continued.

```
NASA/AMES WING PROGRAM                              *****WING*****

      CANARD - WING - BODY  CONFIGURATION

    FLOW CONDITIONS -    MACH        BETA
                       2.01000    1.74359

    REFERENCE QUANTITIES -    AREA      LENGTH      XMOM       B/2     PANELS
                            192.000    15.330     25.000     3.370     16


                ----------DCN/DALPHA--------
       CNP     (RAD**-1) (DEG**-1)   * BETA       CMP     DCM/DCN       XAC        XCP        YCP
     0.06369    0.11375   0.00199    0.19833   0.06657   1.04514      8.978      8.977      2.145

    SPAN LOAD DISTRIBUTION        2B = 13.4800
        Y        CCN/2B
    1.58167    0.30599
    2.08433    0.26667
    2.58055    0.21207
    3.03166    0.10898

    TRAILING VORTEX ....   Y =   2.8223      GAM/V2PI =   3.2823E-01


          SPANWISE DISTRIBUTION OF L.E. SUCTION
REGION COL   Y/(B/2)      CCS/2B      EPS(DEG)    CCX/2B      CCY/2B
   1     1   0.47292     0.07680      90.460   -0.02170    0.07366
   1     2   0.62352     0.14431      89.745   -0.03905    0.13892
   1     3   0.77411     0.22702      88.989   -0.05855    0.21934
   1     4   0.92470     0.27876      90.778   -0.08027    0.26696
```

(f) Page 6.

Figure 7.- Continued.

SUMMARY OF VORTEX STRENGTHS AND POSITION AT X = 11.776

VORTEX	GAMMA/2*PI*V	Y	Z
1	0.074799	0.373	1.968
2	0.328235	2.822	-0.391
3	0.150326	2.595	0.360

VELOCITY INDUCED AT SPECIFIED FIELD POINTS ON TAIL (BESKIN)

	X	Y	Z	V/V(INF)	W/V(INF)
1	22.282	1.996	0.000	0.0000	1.9482E-01
2	25.196	1.996	0.000	0.0000	1.9482E-01
3	28.111	1.996	0.000	0.0000	1.9482E-01
4	31.025	1.996	0.000	0.0000	1.9482E-01
5	33.940	1.996	0.000	0.0000	1.9482E-01
6	36.854	1.996	0.000	0.0000	1.9482E-01
7	23.828	2.664	0.000	0.0000	1.0931E-01
8	26.437	2.664	0.000	0.0000	1.0931E-01
9	29.045	2.664	0.000	0.0000	1.0931E-01
10	31.653	2.664	0.000	0.0000	1.0931E-01
11	34.261	2.664	0.000	0.0000	1.0931E-01
12	36.870	2.664	0.000	0.0000	1.0931E-01
13	25.374	3.333	0.000	0.0000	6.9863E-02
14	27.676	3.333	0.000	0.0000	6.9863E-02
15	29.978	3.333	0.000	0.0000	6.9863E-02
16	32.281	3.333	0.000	0.0000	6.9863E-02
17	34.583	3.333	0.000	0.0000	6.9863E-02
18	36.885	3.333	0.000	0.0000	6.9863E-02
19	26.919	4.001	0.000	0.0000	4.8477E-02
20	28.915	4.001	0.000	0.0000	4.8477E-02
21	30.912	4.001	0.000	0.0000	4.8477E-02
22	32.908	4.001	0.000	0.0000	4.8477E-02
23	34.904	4.001	0.000	0.0000	4.8477E-02
24	36.900	4.001	0.000	0.0000	4.8477E-02
25	28.463	4.669	0.000	0.0000	3.5602E-02
26	30.154	4.669	0.000	0.0000	3.5602E-02
27	31.844	4.669	0.000	0.0000	3.5602E-02
28	33.535	4.669	0.000	0.0000	3.5602E-02
29	35.225	4.669	0.000	0.0000	3.5602E-02
30	36.915	4.669	0.000	0.0000	3.5602E-02
31	30.006	5.336	0.000	0.0000	2.7257E-02
32	31.391	5.336	0.000	0.0000	2.7257E-02
33	32.776	5.336	0.000	0.0000	2.7257E-02
34	34.161	5.336	0.000	0.0000	2.7257E-02
35	35.546	5.336	0.000	0.0000	2.7257E-02
36	36.931	5.336	0.000	0.0000	2.7257E-02
37	31.546	6.002	0.000	0.0000	2.1545E-02
38	32.626	6.002	0.000	0.0000	2.1545E-02
39	33.706	6.002	0.000	0.0000	2.1545E-02
40	34.786	6.002	0.000	0.0000	2.1545E-02
41	35.866	6.002	0.000	0.0000	2.1545E-02
42	36.946	6.002	0.000	0.0000	2.1545E-02
43	33.079	6.664	0.000	0.0000	1.7472E-02
44	33.855	6.664	0.000	0.0000	1.7472E-02
45	34.632	6.664	0.000	0.0000	1.7472E-02
46	35.408	6.664	0.000	0.0000	1.7472E-02
47	36.185	6.664	0.000	0.0000	1.7472E-02
48	36.961	6.664	0.000	0.0000	1.7472E-02
49	34.592	7.319	0.000	0.0000	1.4487E-02
50	35.069	7.319	0.000	0.0000	1.4487E-02
51	35.546	7.319	0.000	0.0000	1.4487E-02
52	36.023	7.319	0.000	0.0000	1.4487E-02
53	36.499	7.319	0.000	0.0000	1.4487E-02
54	36.976	7.319	0.000	0.0000	1.4487E-02
55	35.968	7.914	0.000	0.0000	1.2391E-02
56	36.172	7.914	0.000	0.0000	1.2391E-02
57	36.377	7.914	0.000	0.0000	1.2391E-02
58	36.581	7.914	0.000	0.0000	1.2391E-02
59	36.785	7.914	0.000	0.0000	1.2391E-02
60	36.990	7.914	0.000	0.0000	1.2391E-02

(g) Page 7.

Figure 7.- Continued.

VELOCITY INDUCED AT SPECIFIED FIELD POINTS ON TAIL (VORTEX)

	X	Y	Z	V/V(INF)	W/V(INF)
1	22.282	1.996	0.000	1.1757E-01	-3.8098E-01
2	25.196	1.996	0.000	1.3732E-01	-3.1656E-01
3	28.111	1.996	0.000	1.3330E-01	-4.6734E-01
4	31.025	1.996	0.000	6.8128E-02	-3.2909E-01
5	33.940	1.996	0.000	5.6476E-02	-2.7537E-01
6	36.854	1.996	0.000	5.1282E-02	-2.5169E-01
7	23.828	2.664	0.000	1.9245E-01	-1.9361E-01
8	26.437	2.664	0.000	1.7936E-01	-1.1332E-01
9	29.045	2.664	0.000	4.7276E-01	-1.7480E-01
10	31.653	2.664	0.000	1.4355E-01	-2.5320E-01
11	34.261	2.664	0.000	1.1303E-01	-2.0003E-01
12	36.870	2.664	0.000	1.0179E-01	-1.7988E-01
13	25.374	3.333	0.000	1.6672E-01	-5.9287E-02
14	27.676	3.333	0.000	1.5478E-01	-2.8686E-02
15	29.978	3.333	0.000	1.9363E-01	5.7063E-02
16	32.281	3.333	0.000	3.2934E-01	-1.9808E-01
17	34.583	3.333	0.000	1.6661E-01	-1.7577E-01
18	36.885	3.333	0.000	1.3429E-01	-1.4616E-01
19	26.919	4.001	0.000	1.2679E-01	3.8934E-03
20	28.915	4.001	0.000	1.1837E-01	1.4526E-02
21	30.912	4.001	0.000	1.2399E-01	4.8676E-02
22	32.908	4.001	0.000	1.8053E-01	8.6594E-02
23	34.904	4.001	0.000	2.8443E-01	2.4177E-02
24	36.900	4.001	0.000	2.4060E-01	-8.0302E-02
25	28.463	4.669	0.000	9.2526E-02	2.6850E-02
26	30.154	4.669	0.000	8.6790E-02	3.3516E-02
27	31.844	4.669	0.000	9.0593E-02	4.7019E-02
28	33.535	4.669	0.000	9.5842E-02	5.8034E-02
29	35.225	4.669	0.000	1.1234E-01	7.1089E-02
30	36.915	4.669	0.000	1.3756E-01	6.9675E-02
31	30.006	5.336	0.000	6.5651E-02	3.4335E-02
32	31.391	5.336	0.000	6.5420E-02	3.9962E-02
33	32.776	5.336	0.000	6.6716E-02	4.3205E-02
34	34.161	5.336	0.000	6.6593E-02	4.6975E-02
35	35.546	5.336	0.000	7.0247E-02	5.2231E-02
36	36.931	5.336	0.000	7.5330E-02	5.4015E-02
37	31.546	6.002	0.000	4.9274E-02	3.7214E-02
38	32.626	6.002	0.000	4.9898E-02	3.8327E-02
39	33.706	6.002	0.000	4.9317E-02	3.8843E-02
40	34.786	6.002	0.000	4.9281E-02	4.0998E-02
41	35.866	6.002	0.000	5.0686E-02	4.3004E-02
42	36.946	6.002	0.000	5.2233E-02	4.3356E-02
43	33.079	6.664	0.000	3.8066E-02	3.4316E-02
44	33.855	6.664	0.000	3.7679E-02	3.4392E-02
45	34.632	6.664	0.000	3.7439E-02	3.5232E-02
46	35.408	6.664	0.000	3.7757E-02	3.6245E-02
47	36.185	6.664	0.000	3.8319E-02	3.6853E-02
48	36.961	6.664	0.000	3.8964E-02	3.6811E-02
49	34.592	7.319	0.000	2.9216E-02	3.0988E-02
50	35.069	7.319	0.000	2.9235E-02	3.1407E-02
51	35.546	7.319	0.000	2.9378E-02	3.1817E-02
52	36.023	7.319	0.000	2.9593E-02	3.2080E-02
53	36.499	7.319	0.000	2.9813E-02	3.2163E-02
54	36.976	7.319	0.000	3.0125E-02	3.2031E-02
55	35.968	7.914	0.000	2.3844E-02	2.8540E-02
56	36.172	7.914	0.000	2.3916E-02	2.8576E-02
57	36.377	7.914	0.000	2.3987E-02	2.8600E-02
58	36.581	7.914	0.000	2.4055E-02	2.8612E-02
59	36.785	7.914	0.000	2.4127E-02	2.8607E-02
60	36.990	7.914	0.000	2.4282E-02	2.8490E-02

(g) Continued.

Figure 7.- Continued.

TOTAL INDUCED VELOCITY AT SPECIFIED FIELD POINTS ON TAIL

	X	Y	Z	V/V(INF)	W/V(INF)
1	22.282	1.996	0.000	1.1757E-01	-1.8615E-01
2	25.196	1.996	0.000	1.3732E-01	-1.2174E-01
3	28.111	1.996	0.000	1.3330E-01	-2.7251E-01
4	31.025	1.996	0.000	6.8128E-02	-1.3427E-01
5	33.940	1.996	0.000	5.6476E-02	-8.0545E-02
6	36.854	1.996	0.000	5.1282E-02	-5.6861E-02
7	23.828	2.664	0.000	1.9245E-01	-8.4293E-02
8	26.437	2.664	0.000	1.7936E-01	-4.0053E-03
9	29.045	2.664	0.000	4.7276E-01	-6.5490E-02
10	31.653	2.664	0.000	1.4355E-01	-1.4389E-01
11	34.261	2.664	0.000	1.1303E-01	-9.0713E-02
12	36.870	2.664	0.000	1.0179E-01	-7.0564E-02
13	25.374	3.333	0.000	1.6672E-01	1.0576E-02
14	27.676	3.333	0.000	1.5478E-01	4.1177E-02
15	29.978	3.333	0.000	1.9363E-01	1.2693E-01
16	32.281	3.333	0.000	3.2934E-01	-1.2822E-01
17	34.583	3.333	0.000	1.6661E-01	-1.0590E-01
18	36.885	3.333	0.000	1.3429E-01	-7.6296E-02
19	26.919	4.001	0.000	1.2679E-01	5.2371E-02
20	28.915	4.001	0.000	1.1837E-01	6.3003E-02
21	30.912	4.001	0.000	1.2399E-01	9.7154E-02
22	32.908	4.001	0.000	1.8053E-01	1.3507E-01
23	34.904	4.001	0.000	2.8443E-01	7.2655E-02
24	36.900	4.001	0.000	2.4060E-01	-3.1824E-02
25	28.463	4.669	0.000	9.2526E-02	6.2452E-02
26	30.154	4.669	0.000	8.6790E-02	6.9118E-02
27	31.844	4.669	0.000	9.0593E-02	8.2621E-02
28	33.535	4.669	0.000	9.5842E-02	9.3636E-02
29	35.225	4.669	0.000	1.1234E-01	1.0669E-01
30	36.915	4.669	0.000	1.3756E-01	1.0528E-01
31	30.006	5.336	0.000	6.5651E-02	6.1592E-02
32	31.391	5.336	0.000	6.5420E-02	6.7219E-02
33	32.776	5.336	0.000	6.6716E-02	7.0462E-02
34	34.161	5.336	0.000	6.6593E-02	7.4232E-02
35	35.546	5.336	0.000	7.0247E-02	7.9488E-02
36	36.931	5.336	0.000	7.5330E-02	8.1272E-02
37	31.546	6.002	0.000	4.9274E-02	5.8759E-02
38	32.626	6.002	0.000	4.9898E-02	5.9872E-02
39	33.706	6.002	0.000	4.9317E-02	6.0388E-02
40	34.786	6.002	0.000	4.9281E-02	6.2543E-02
41	35.866	6.002	0.000	5.0686E-02	6.4549E-02
42	36.946	6.002	0.000	5.2233E-02	6.4901E-02
43	33.079	6.664	0.000	3.8066E-02	5.1788E-02
44	33.855	6.664	0.000	3.7679E-02	5.1864E-02
45	34.632	6.664	0.000	3.7439E-02	5.2704E-02
46	35.408	6.664	0.000	3.7757E-02	5.3717E-02
47	36.185	6.664	0.000	3.8319E-02	5.4325E-02
48	36.961	6.664	0.000	3.8964E-02	5.4283E-02
49	34.592	7.319	0.000	2.9216E-02	4.5475E-02
50	35.069	7.319	0.000	2.9235E-02	4.5894E-02
51	35.546	7.319	0.000	2.9378E-02	4.6304E-02
52	36.023	7.319	0.000	2.9593E-02	4.6567E-02
53	36.499	7.319	0.000	2.9813E-02	4.6650E-02
54	36.976	7.319	0.000	3.0125E-02	4.6519E-02
55	35.968	7.914	0.000	2.3844E-02	4.0931E-02
56	36.172	7.914	0.000	2.3916E-02	4.0967E-02
57	36.377	7.914	0.000	2.3987E-02	4.0991E-02
58	36.581	7.914	0.000	2.4055E-02	4.1003E-02
59	36.785	7.914	0.000	2.4127E-02	4.0998E-02
60	36.990	7.914	0.000	2.4282E-02	4.0881E-02

(g) Concluded.

Figure 7.- Continued.

NASA/AMES WING PROGRAM *****TAIL*****

CANARD - WING - BODY CONFIGURATION

FLOW CONDITIONS - MACH 2.01000 BETA 1.74359

REFERENCE QUANTITIES -	AREA	LENGTH	XMOM	B/2	PANELS
	192.000	15.330	25.000	8.360	60

CNP	----------DCN/DALPHA-------- (RAD**-1)	(DEG**-1)	* BETA	CMP	DCM/DCN	XAC	XCP	YCP
0.30934	1.12533	0.01964	1.96210	-0.12993	-0.39058	30.988	31.439	4.599

SPAN LOAD DISTRIBUTION 2B = 33.4400

Y	CCN/2B
1.99578	0.15634
2.66439	0.15320
3.33281	0.15181
4.00097	0.15348
4.66870	0.15809
5.33575	0.15104
6.00151	0.13822
6.66443	0.11990
7.31886	0.09521
7.91379	0.04957

TRAILING VORTEX .... Y = 7.3476 GAM/V2PI = 4.1602E-01

SPANWISE DISTRIBUTION OF L.E. SUCTION

REGION	COL	Y/(B/2)	CCS/2B	EPS(DEG)	CCX/2B	CCY/2B
1	1	0.23943	0.00666	86.565	-0.00190	0.00638
1	2	0.31949	0.00425	92.983	-0.00166	0.00391
1	3	0.39955	0.00854	104.707	-0.00486	0.00702
1	4	0.47961	0.01663	93.895	-0.00674	0.01521
1	5	0.55967	0.02867	92.705	-0.01107	0.02645
1	6	0.63973	0.04133	90.049	-0.01417	0.03882
1	7	0.71979	0.05274	89.082	-0.01724	0.04984
1	8	0.79985	0.06464	88.450	-0.02045	0.06132
1	9	0.87991	0.07851	87.837	-0.02405	0.07474
1	10	0.95997	0.08993	90.052	-0.03083	0.08448

SUMMARY OF VORTEX STRENGTHS AND POSITION AT X = 37.276

VORTEX	GAMMA/2*PI*V	Y	Z
1	0.074799	4.162	0.546
2	0.328235	2.552	3.083
3	0.150326	1.794	2.933
4	0.416016	7.348	0.000
5	0.069799	6.440	1.411

(h) Page 8.

Figure 7.- Continued.

		-- INCIDENCE --		---- WING -----		---- TAIL -----		
ALPHA	M	WING	TAIL	KVLE*	KVSE*	KVLE*	KVSE*	W/V LIMIT
16.00	2.01	10.000	0.000	0.500	0.000	0.500	0.000	1.000

SUMMARY OF FORCE AND PITCHING MOMENT COEFFICIENTS

	CN	CM	XCP	CL	CDI	CA
NOSE...						
POTENTIAL	6.402E-03	9.357E-03	2.596E 00	6.154E-03	1.765E-03	
VORTEX	2.777E-03	4.141E-03	2.138E 00	2.669E-03	7.653E-04	
WING...						
W(B) POTENTIAL	6.272E-02	6.555E-02	8.977E 00	5.637E-02	1.616E-02	1.106E-02
W(B) VORTEX,LE	2.531E-02	2.723E-02	8.505E 00	2.275E-02	6.523E-03	4.462E-03
W(B) VORTEX,SE	0.000	0.000	0.000	0.000	0.000	0.000
BODY...						
B(W) POTENTIAL	2.319E-02	2.223E-02	1.030E 01	2.229E-02	6.392E-03	
B(W) VORTEX,LE	9.358E-03	8.971E-03	1.030E 01	8.995E-03	2.579E-03	
B(W) VORTEX,SE	0.000	0.000	1.030E 01	0.000	0.000	
AFTERBODY...	-6.246E-03	-4.384E-03	1.424E 01	-6.004E-03	-1.722E-03	
TAIL...						
T(B) POTENTIAL	3.093E-01	-1.299E-01	3.144E 01	2.974E-01	8.527E-02	0.000
T(B) VORTEX,LE	4.537E-02	-1.956E-02	3.161E 01	4.361E-02	1.251E-02	0.000
T(B) VORTEX,SE	0.000	0.000	0.000	0.000	0.000	0.000
BODY...						
B(T) POTENTIAL	5.034E-02	-1.629E-02	2.996E 01	4.839E-02	1.388E-02	
B(T) VORTEX,LE	7.383E-03	-2.390E-03	2.996E 01	7.097E-03	2.035E-03	
B(T) VORTEX,SE	0.000	0.000	2.996E 01	0.000	0.000	
TOTAL CONFIGURATION...	5.359E-01	-3.508E-02	2.600E 01	5.097E-01	1.461E-01	1.552E-02

CDI/CL**2 = 5.626E-01

(i) Page 9.

Figure 7.- Continued.

SUMMARY OF FREE VORTEX TRAJECTORIES

```
     VORTEX 1 ---  BODY VORTEX FROM NOSE
     VORTEX 2 ---  WING TRAILING VORTEX
     VORTEX 3 ---  WING SEPARATION VORTEX

 VORTEX    GAMMA/2*PI*V
    1        0.074799
    2        0.328235
    3        0.150326


    X       DX           A         S        RO       DA/DX
 4.2750  1.0000000    0.8260    0.8260   0.8260    0.1169

VORTEX SIGMA(REAL)    (IMAG)
  1     5.332E-01  1.083E 00


    X       DX           A         S        RO       DA/DX
 5.2750  0.5000000    0.9269    1.4015   1.0072    0.0963

VORTEX SIGMA(REAL)    (IMAG)
  1     5.171E-01  1.276E 00


    X       DX           A         S        RO       DA/DX
 6.2750  0.5000000    1.0185    1.9118   1.2272    0.0776

VORTEX SIGMA(REAL)    (IMAG)
  1     4.963E-01  1.432E 00


    X       DX           A         S        RO       DA/DX
 7.2750  0.5000000    1.0785    2.1977   1.3635    0.0649

VORTEX SIGMA(REAL)    (IMAG)
  1     4.731E-01  1.562E 00
                          •    •    •
    X       DX           A         S        RO       DA/DX
11.7750  0.5000000    1.3324    1.3324   1.3324    0.0539

VORTEX SIGMA(REAL)    (IMAG)
  1     3.732E-01  1.968E 00


    X       DX           A         S        RO       DA/DX
11.7760  0.5000000    1.3325    1.3325   1.3325    0.0539

VORTEX SIGMA(REAL)    (IMAG)
  1     3.732E-01  1.968E 00
  2     2.822E 00 -3.907E-01
  3     2.595E 00  3.603E-01


    X       DX           A         S        RO       DA/DX
12.7760  0.2500000    1.3885    1.3885   1.3885    0.0496

VORTEX SIGMA(REAL)    (IMAG)
  1     3.622E-01  1.984E 00
  2     2.988E 00 -9.758E-02
  3     2.321E 00  2.655E-01
```

• • •

(j) Page 10.

Figure 7.- Continued.

X	DX	A	S	RO	DA/DX
24.7760	0.5000000	1.6670	3.9111	2.3108	0.0000

VORTEX	SIGMA(REAL)	(IMAG)
1	1.717E 00	6.512E-01
2	2.551E 00	2.158E 00
3	3.282E 00	1.953E 00

X	DX	A	S	RO	DA/DX
25.7760	0.5000000	1.6670	4.2751	2.4625	0.0000

VORTEX	SIGMA(REAL)	(IMAG)
1	1.875E 00	3.794E-01
2	2.544E 00	2.087E 00
3	3.155E 00	2.528E 00

X	DX	A	S	RO	DA/DX
26.7760	0.5000000	1.6670	4.6390	2.6190	0.0000

VORTEX	SIGMA(REAL)	(IMAG)
1	2.051E 00	2.585E-01
2	2.659E 00	2.096E 00
3	2.761E 00	2.879E 00

• • •

X	DX	A	S	RO	DA/DX
36.7760	0.5000000	1.6670	8.2785	4.3071	0.0000

VORTEX	SIGMA(REAL)	(IMAG)
1	4.116E 00	4.262E-01
2	2.565E 00	2.927E 00
3	1.804E 00	3.094E 00

X	DX	A	S	RO	DA/DX
37.0260	0.2500000	1.6670	1.6670	1.6670	0.0000

VORTEX	SIGMA(REAL)	(IMAG)
1	4.140E 00	4.597E-01
2	2.562E 00	2.992E 00
3	1.789E 00	3.004E 00

X	DX	A	S	RO	DA/DX
37.1510	0.1250000	1.6670	1.6670	1.6670	0.0000

VORTEX	SIGMA(REAL)	(IMAG)
1	4.151E 00	5.068E-01
2	2.558E 00	3.039E 00
3	1.789E 00	2.970E 00

(j) Concluded.

Figure 7.- Concluded.

REFERENCES

1. Mendenhall, M. R. and Nielsen, J. N.: Effect of Symmetrical Vortex Shedding on the Longitudinal Aerodynamic Characteristics of Wing-Body-Tail Combinations. NEAR TR 69, May 1974; NASA CR 2473, 1974.

2. Pitts, W. C., Nielsen, J. N., and Kaattari, C. E.: Lift and Center of Pressure of Wing-Body-Tail Combinations at Subsonic, Transonic, and Supersonic Speeds. NACA Rept. No. 1307, 1957.

3. Jorgensen, L. H.: Prediction of Static Aerodynamic Characteristics for Space-Shuttle-Like and Other Bodies at Angles of Attack from 0° to 180°. NASA TN D-6996, Jan. 1973.

4. Jorgensen, L. H.: A Method for Estimating Static Aerodynamic Characteristics for Slender Bodies of Circular and Noncircular Cross Section Alone and With Lifting Surfaces at Angles of Attack From 0° to 90°. NASA TN D-7228, Apr. 1973.

5. DeYoung, J. and Harper, C. W.: Theoretical Symmetric Span Loading at Subsonic Speeds for Wings Having Arbitrary Planform. NACA Rept. 921, 1948.

6. Brady, J. A., Page, V. R., and Koenig, D. G.: Large-Scale Low-Speed Wind-Tunnel Tests of a Delta Winged Supersonic Transport Model with a Delta Canard Control Surface. NASA TM X-643, Jan. 1962.

7. Allen, H. J.: Estimation of the Forces and Moments Acting on Inclined Bodies of Revolution of High Fineness Ratio. NACA RMA9I26, 1949.

8. Dillenius, M. F. E., Mendenhall, M. R. and Spangler, S. B.: Calculation of the Longitudinal Aerodynamic Characteristics of STOL Aircraft with Externally-Blown Jet-Augmented Flaps. NASA CR-2358, Feb. 1974.

9. Polhamus, E. C.: Predictions of Vortex-Lift Characteristics Based on a Leading-Edge Suction Analogy. AIAA Paper No. 69-1133, Oct. 1969.

10. Sacks, A. H.: Vortex Interference on Slender Airplanes. NACA TN 3525, Nov. 1955.

11. Koenig, D. G.: Tests in the Ames 40- by 80-Foot Wind Tunnel of an Airplane Configuration With an Aspect Ratio 3 Triangular Wing and an All-Movable Horizontal Tail - Longitudinal and Lateral Characteristics. NACA RM A52L15, Apr. 1953.

12. Woodward, F. A. and Larsen, J. V.: A Method of Optimizing Camber Surfaces for Wing-Body Combinations at Supersonic Speeds. Part I - Theory and Applications. Rept. D6-10741, The Boeing Co., 1965.

13. Carmichael, R. L. and Woodward, F. A.: An Integrated Approach to the Analysis and Design of Wings and Wing-Body Combinations in Supersonic Flow. NASA TN D-3685, Oct. 1966.

14. Driver, C.: Effects of Wing Height on the Stability and Control Characteristics at a Mach Number of 2.01 of a Canard Airplane Configuration with a 70° Delta Wing. NASA TM X-328, Sept. 1960.

15. Peterson, V. L. and Menees, G. P.: Static Longitudinal Stability and Control Characteristics of an Unswept Wing and Unswept Horizontal-Tail Configuration at Mach Numbers 0.70 to 2.22. NASA MEMO 6-11-59A, May 1959.

☆ U. S. GOVERNMENT PRINTING OFFICE: 1975—635-047 / 52