AD-A031 388 NIELSEN ENGINEERING AND RESEARCH INC MOUNTAIN VIEW CALIF F/G 1/3
PREDICTION OF SUPERSONIC STORE SEPARATION CHARACTERISTICS. VOLU--ETC(U)
MAY 76 F K GOODWIN, M M KEIRSTEAD F33615-75-C-3053
UNCLASSIFIED NEAR-TR-106 AFFDL-TR-76-41-VOL-2 NL

1 OF 3
AD A031388



F
388

AD A031388

AFFDL-TR-76-41
VOLUME II

(12)

PREDICTION OF SUPERSONIC STORE SEPARATION CHARACTERISTICS
VOLUME II.-USERS MANUAL FOR THE COMPUTER PROGRAM

NIELSEN ENGINEERING & RESEARCH, INC.
MOUNTAIN VIEW, CALIFORNIA 94043

COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION

MAY 1976

DDC
RECEIVED
NOV 1 1976
B

FINAL REPORT FEBRUARY 1975 - MARCH 1976

Approved for public release: distribution unlimited

AIR FORCE FLIGHT DYNAMICS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

E. H. Flinn

E. H. Flinn, Chief
Control Criteria Branch
Flight Control Division

Calvin L. Dyer

Calvin L. Dyer
Project Engineer
Control Criteria Branch
Flight Control Division

Lt Col Larry M. Hadley, Chief
Flight Control Division

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

AIR FORCE - 1 OCTOBER 76 - 175

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
AFFDL-TR-76-41, Vol. II

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

(9) Final rept. 24 Feb 75-24 March 76

(6) 4. TITLE (and Subtitle)
PREDICTION OF SUPERSONIC STORE SEPARATION CHARACTERISTICS, Volume II. Users Manual for the Computer Program

5. TYPE OF REPORT & PERIOD COVERED
Final-Technical
2/24/75 to 3/24/76

6. PERFORMING ORG. REPORT NUMBER
NEAR-TR-106

(14)

(10) 7. AUTHOR(s)
Frederick K. Goodwin
Mary M. Keirstead
Marnix F. E. Dillenius

8. CONTRACT OR GRANT NUMBER(s)
(15) F33615-75-C-3053

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Nielsen Engineering & Research, Inc.
510 Clyde Avenue
Mountain View, California 94043

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
Program Element 62201 F
8219-01-13

11. CONTROLLING OFFICE NAME AND ADDRESS
Air Force Flight Dynamics Laboratory (FGC)
Wright-Patterson Air Force Base
Ohio 45433

(11) 12. REPORT DATE
May 1976

13. NUMBER OF PAGES
271

(16) 14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)
AF-8219

(12) 274 p.

15. SECURITY CLASS. (of this report)
Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

(17) 16. DISTRIBUTION STATEMENT (of this Report)
821901
Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

(18) AFFDL (19) TR-76-41-Vol-2

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Aerodynamic Loads	Flow Fields
Aerodynamic Interference	Store Separation
External Stores	Supersonic Flow

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Detailed instructions are presented for using a computer program which calculates the six-degree-of-freedom trajectories of external stores which are separated from fighter-bomber type aircraft flying at supersonic speeds. Single store configurations can be handled by the program. The program uses linear potential-flow theory to model the wing and pylon loading and thickness. Three-dimensional line sources and line doublets are used to model the fuselage and store. The program also calculates the trajectory

OVER

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

389 783

1 B

20. of the store as it separates from the aircraft. This report describes the program, presents instructions for preparing input for the program, describes the output from the program, and presents a sample case. The program draws heavily on an earlier program for subsonic speeds written by the present authors and described in AFFDL-TR- 74-130.

ACCESSION for
NTIS White Section ☒
DDC Buff Section ☐
UNANNOUNCED ☐
JUSTIFICATION
BY
DISTRIBUTION/AVAILABILITY CODES
Dist. AVAIL. and/or SPECIAL
A

FOREWORD

This report, "Prediction of Supersonic Store Separation Characteristics," describes a combined theoretical-experimental program directed toward developing a computer program for predicting the trajectory of an external store separated from an aircraft flying at supersonic speed. Volume I, "Theoretical Methods and Comparisons with Experiment," describes the theoretical approach and presents extensive comparisons with experimental data. This volume, Volume II.- "Users Manual for the Computer Program," presents detailed instructions on the use of the computer program.

The work was carried out by Nielsen Engineering & Research, Inc., 510 Clyde Avenue, Mountain View, California 94043, under Contract No. F33615-75-C-3053. The contract was initiated under Project 8219, Task 821901, of the Air Force Flight Dynamics Laboratory. The Air Force Project Engineer on the contract was Calvin L. Dyer, AFFDL/FGC. The report number assigned by Nielsen Engineering & Research, Inc. is NEAR TR 106.

The authors wish to thank Mr. Dyer, AFFDL/FGC, for his assistance during the course of the investigation. The computer program card deck is available upon request from Mr. Dyer. His address is AFFDL/FGC, Wright-Patterson AFB, Ohio 45433.

The work documented in this report was started on February 24, 1975 and was effectively concluded with the submission of this report. The report was submitted by the authors in March 1976.

TABLE OF CONTENTS

Section		Page No.
1.	INTRODUCTION	1
2.	GENERAL DESCRIPTION OF THE USE OF THE PROGRAM	3
	2.1 Input Section	3
	2.2 Wing-Pylon Loading Section	4
	2.3 Trajectory Section	5
3.	DETAILS OF THE USE OF THE PROGRAM	6
	3.1 *General Flow Chart*	7
	3.2 Input Data	10
	3.2.1 Input format	10
	3.2.2 Sample input data	29
	3.3 Output from the Program	34
4.	PROGRAM ERROR MESSAGES	38
5.	PROGRAM RUNNING TIMES	39
TABLE I		40
APPENDIX I - DETAILS FOR THE PROGRAMMER		117
APPENDIX II - DETERMINATION OF TAIL-FIN LIFT-CURVE SLOPE		262
REFERENCES		264

LIST OF ILLUSTRATIONS

Figure		Page
1.	Program listing.	43
2.	General flow chart of trajectory program.	81
3.	Trajectory program input format.	84
4.	Simplified layout of panels for wing-pylon and fuselage combination.	92
5.	Fuselage coordinate system.	93
6.	Definition of certain wing input variables.	94
7.	Example wing.	95
8.	Variables describing and locating pylon, input data item number 19.	96
9.	Wind-tunnel models used in sample calculations.	97
10.	Configuration for sample trajectory.	100
11.	Input data deck for sample trajectory calculation.	101
12.	Trajectory program output for sample case.	102
13.	Coordinate systems fixed in separated store and used in force and moment calculation.	116
14.	Coordinate systems used in trajectory calculation.	117

LIST OF SYMBOLS

a_{max}	maximum store body radius
c_{d_c}	section drag coefficient of a circular cylinder normal to airstream
C_A	axial-force coefficient, axial force/$q_{\infty_s} S_R$
C_ℓ	rolling-moment coefficient, rolling moment/$q_{\infty_s} S_R \ell_R$
C_m	pitching-moment coefficient, pitching moment/$q_{\infty_s} S_R \ell_R$
C_n	yawing-moment coefficient, yawing moment/$q_{\infty_s} S_R \ell_R$
C_N	normal-force coefficient, normal force/$q_{\infty_s} S_R$
C_Y	side-force coefficient, side force/$q_{\infty_s} S_R$
d	maximum diameter of store
F_T	thrust force
F_x	total force acting along the store longitudinal axis
g_x	component of the gravitational force acting along the store longitudinal axis
I_{xx}, I_{yy}, I_{zz}	moments of inertia about x,y,z axes of figure 13; taken about store moment center
I_{yz}, I_{xz}, I_{xy}	products of inertia about x,y,z axes of figure 13; taken about store moment center
ℓ	body length
ℓ_R	reference length; store maximum diameter, d
ℓ_s	length of separated store
m	mass of separated store
M_∞	aircraft free-stream Mach number
p,q,r	rotational velocities about x,y,z axes of figure 13; positive as shown in figure 14
q_∞	free-stream dynamic pressure; $1/2\ \rho_\infty V_\infty^2$

q_{∞_s}	ejected store free-stream dynamic pressure; $1/2\ \rho_\infty V^2_{\infty_s}$
r	local body radius
s	semispan of wing
S_R	reference area taken equal to body frontal area, πa^2_{max}
t	time
u_+/V_∞	strength of a constant u-velocity panel
U_s, V_s, W_s	total velocities as seen by store, positive in x_s, y_s, z_s directions, see figure 13
V_∞	aircraft free-stream velocity
V_{∞_s}	separated store free-stream velocity
x,y,z	coordinate system with origin at store moment center, see figure 13
x_B, y_B, z_B	coordinate system fixed in fuselage with the origin at nose, see figure 5
x_ℓ, z_ℓ	coordinate system fixed in local airfoil section of wing, see figure 8(b), with x_ℓ lying along chord line
x_s, y_s, z_s	coordinate system fixed in separated store with origin at store nose, see figure 13
$x_{s,m}$	x_s coordinate of store moment center
x_w, y_w, z_w	coordinate system fixed in wing with origin at root-chord leading edge, see figure 6
α_ℓ	local angle of attack due to wing twist and camber
θ	tan θ, local slope of wing thickness envelope
ξ, η, ζ	inertial coordinate system fixed in fuselage nose, positive forward along longitudinal axis, positive laterally to the right, and positive vertically downward, respectively, see figure 14
ρ_∞	free-stream mass density
Ψ, Θ, Φ	store yaw, pitch and roll angles specifying angular orientation of store x,y,z coordinate system relative to ξ, η, ζ inertial system, see figure 14

PREDICTION OF SUPERSONIC STORE SEPARATION CHARACTERISTICS

Volume II.- Users Manual for the Computer Program

1. INTRODUCTION

The purpose of this volume of the report is to describe and present instructions for using the computer program developed in conjunction with the theoretical work presented in Volume I, reference 1. This work represents a first year's effort on the development of an analytical method for predicting the trajectory of a store separated from an aircraft flying at supersonic speeds. As a result, the parent aircraft configuration which can be modeled does not have the generality of the subsonic store separation prediction method and program described in references 2 and 3. Preliminary versions of that method and program are described in references 4, 5, and 6.

In developing the present program every attempt has been made to keep the input data format the same as that used in the program described in reference 3. The printed output also has been kept as similar as possible.

The program as described in this report has been run on the CDC 6600 computer. The program should run with very little, if any, modification on other computers with large enough memory capacity. The program requires approximately 210,000 octal words of core storage on the 6600. No tapes, drums, or disks are used by the program. All input to the program is from cards and all output is printed.

The next sections of this report will describe in more detail the general use of the program and then present instructions for preparing the input data and interpreting the output. A sample case is presented. Details of the program are contained in an appendix.

The method of describing the aircraft components, the aircraft operational parameters, and the dynamical characteristics of the store are of importance. The various parameters which are included in the computer program are:

Wing Panels

Thickness distribution: Specified at large number of chordwise locations and at the same number of spanwise locations as used in the constant u-velocity panel layout which represents the wing as a lifting surface.

PRECEDING PAGE BLANK-NOT FILMED

Mean camber surface: May have both twist and camber.

Leading-edge shape: Represented by straight line segments of differing sweep.

Trailing-edge shape: Also represented by straight line segments of differing sweep.

Dihedral: None.

Fuselage

Shape: Body of revolution.

Pylon

Thickness distribution: Same method of description as for wing panel

Mean camber surface: Planar.

Orientation: Vertical and streamwise.

Leading-edge shape: Straight but can be swept.

Trailing-edge shape: Straight but can be swept.

Tip: Parallel to pylon root chord.

Number: One pylon per wing panel or one under fuselage centerline.

Store

Shape: Body of revolution.

Number of stores: One.

Attached orientation: Initial pitch angle arbitrary, initial yaw angle zero.

Empennage: Planar or cruciform empennage.

Initial roll orientation: Arbitrary.

Power: Option of specifying a thrust time history.

Airplane Operating Characteristics

Flight path: Straight but not necessarily horizontal.

Flight velocity: Constant.

Density: Constant.

Angle of attack: Constant.

Yaw angle: Zero.

Store Inertial Characteristics

Moments of inertia: Constant.

Products of inertia: Constant.

Center of gravity position: Not necessarily on store longitudinal axis.

Store Ejection Conditions

Initial translational velocities: Arbitrary.

Initial pitching velocity: Arbitrary.

Initial yawing velocity: Arbitrary.

Initial rolling velocity: Arbitrary.

2. GENERAL DESCRIPTION OF THE USE OF THE PROGRAM

The store separation trajectory program consists of three main sections. The first section reads in the input data and calculates quantities from these data, the second section solves for the singularity distribution which represents the wing-pylon loading including interference on the fuselage, and the third section calculates the trajectory.

2.1 Input Section

The input section of the program reads in the following information.

(1) Aircraft flight conditions
(2) Indices specifying what aircraft components are present
(3) Fuselage data
(4) Wing data
(5) Pylon data
(6) Store data

The aircraft flight conditions, item (1) above, which can be varied are the angle of attack, flight path angle, Mach number, free-stream air density, and flight velocity.

Item (2) consists of four indices. The first specifies whether a fuselage is present and the second specifies whether a pylon is present. The third index is not used and has been left in for compatibility with the input data for the subsonic program, reference 3. The fourth index specifies whether a store is present. The program in its present form will only handle one store.

The fuselage input data, item (3) above, are the length, maximum radius, and polynomials specifying the body radius as a function of axial location. Other fuselage input data specify the number of sources and doublets, to be used in representing the fuselage volume and angle of attack effects, and specify the fuselage interference panel layout.

The wing input data, item (4), locate the wing relative to the fuselage and supply information required to lay out the u-velocity panels and the thickness distribution. These data also include the twist and camber distribution and the slope distribution of the thickness envelope. The wing leading and trailing edges can have breaks in sweep.

Similarly, the pylon input data, item (5), locate the pylon and provide information required to lay out the u-velocity panels and the thickness distribution. The pylon is located laterally relative to the fuselage centerline and longitudinally relative to the leading edge of the local wing chord. The data also include the slope distribution of the thickness envelope. The pylon cannot have twist or camber. The leading and trailing edges can be swept at constant but different sweep angles.

Store data, item (6), are input which assign a store number and specify a shape number, length, and maximum radius. The store is located by specifying its lateral position relative to the fuselage centerline and the longitudinal and vertical position of the store nose relative to the wing chord immediately above the store. The incidence of the store relative to the wing root chord is also specified. The store must be under the fuselage centerline or to the left of this line as seen by the pilot. Data used to determine the source and doublet distributions to represent the store volume and angle of attack effects are also input. These data include the number of singularities and polynomials specifying the store shape.

2.2 Wing-Pylon Loading Section

The second main section of the program calculates the singularity distribution representing the wing-pylon loading including interference on the fuselage. This is done using the method described in section 3.3 of reference 1. The coefficient matrix is first calculated and then the right-hand side is determined. Finally, the strengths of the constant u-velocity panels divided by π are calculated by solving the set of simultaneous equations.

2.3 Trajectory Section

The third and last main section of the trajectory program is the trajectory calculation which consists of the following steps:

(1) Input additional information to describe separated store
(2) Input empennage data if store has one
(3) Input thrust time history if powered store
(4) Initialize for trajectory calculation
(5) Calculate aerodynamic forces and moments
(6) Calculate accelerations and rates of change of orientation angles
(7) Integrate equations of motion
(8) Repeat steps (5), (6), and (7) to end of trajectory

The additional data describing the store to be separated include indices specifying the store number, the number of segments the body is to be broken into for the force calculation, the flow separation location, whether the store has an empennage and whether it is powered. Also, the store mass and inertia characteristics are read in along with the location of the point about which the aerodynamic moments are to be calculated. This is the point about which the moments and products of inertia were calculated. The location of the store center of mass relative to this point is also specified as are the store axial-force coefficient and the value of the crossflow-drag coefficient to be used in the viscous crossflow force and moment calculation.

Three other indices are input which pertain to options included in the computer program. Provision has been made to include or exclude the damping terms in the velocity field calculation. Also, for a store with an empennage, rolling moment may or may not be included in the acceleration determination. The third option pertains to the calculation of free-flight trajectories as opposed to captive-store trajectories as obtained in the wind tunnel. In wind-tunnel captive-store testing it is customary to change the store pitch and yaw angles to account for translational motion only while measuring the aerodynamic forces and moments. This changes its position in the nonuniform flow field. Provision has been made in the computer program to simulate this.

For a store with an empennage, additional quantities must be specified. These data are an index indicating whether the enpennage is planar or cruciform, the tail-fin semispan, the average body radius in the tail-fin region, the initial roll orientation of the fins, and the lift-curve slope

of the fins alone. In addition, the axial position at which the forces are assumed to act must be specified.

For a powered store, a series of polynomials is used to specify the thrust time history.

In the trajectory initialization certain store separation conditions are specified. These are the initial translational velocities and rotational velocities.

The integration of the equations of motion is done by a standard numerical integration technique with the aerodynamic forces and moments calculated at each point required by the integration scheme. The calculation of the nonuniform velocity field and the resulting forces and moments is described in sections 3.4 and 4 of reference 1 and in section 5 of reference 4, respectively.

For a given aircraft-store combination and Mach number, it can be seen that a series of trajectories can be run with only minor changes to the input data deck. For example, the aircraft angle of attack can be varied by changing one number on one card as can the aircraft flight path angle. The altitude can be varied by changing the free-stream density and possibly the free-stream velocity to account for changes in the speed of sound. Among other things easily varied are the store mass and inertia properties, center of gravity location, and ejection conditions.

Provision has also been made for restarting a trajectory. This is accomplished by changing one card which specifies the initial and final times and adding two cards specifying the current values of the dependent variables. These are tabulated in the output at each integration step.

3. DETAILS OF THE USE OF THE PROGRAM

The program consists of a main program and 42 subroutines. Table I lists these subroutines in alphabetical order and gives a one-sentence description of what each subroutine does. A listing of the six-degree-of-freedom trajectory program and its 42 subroutines is presented in figure 1 and a general flow chart of the main program in figure 2. The program listing in figure 1 is specifically for the CDC 6600 computer. To run it on other machines the first card and the three cards following statement number 1000 in the main program may have to be removed or changed. A detailed description of the program and subroutines, including the equations programmed, is presented in Appendix I.

3.1 General Flow Chart

The computer program integrates the six-degree-of-freedom equations of motion which are derived in Appendix II of reference 4. The aerodynamic forces and moments are calculated by the methods presented in section 5.2 of that report.

A general flow chart of the program is presented in figure 2. Page 1 of the flow chart, figure 2(a), is the input section of the program. Constants are defined and heading information is read and printed. The aircraft flight conditions are input as are indices specifying what aircraft components are present. If the fuselage is present, the fuselage data are read in and printed in subroutine FUSEIO. This routine also calls BDYGEN to calculate the source and doublet distributions which represent the fuselage volume and angle of attack effects.

The next steps in the program read in the data required to model the wing. The data locating the wing are first read and then subroutine SWNGIN is called. This routine reads in the data required to lay out the constant u-velocity panels which will represent the loaded wing. In addition it reads in the twist and camber distribution at the panel control points. Subroutine WLYOUT is called to lay out the panels. The last wing input data is the thickness distribution. These data are read in by subroutine WITHIN.

A check is made in the program to determine whether or not a pylon is present. If one is, subroutine PLYOUT is called to read in the data required to lay out the constant u-velocity panels which will represent the pylon loading. This routine also lays out the panels. Subroutine PYTHIN is next called to read in the pylon thickness data.

The next two steps in the program are calls to subroutines THKOUT and THKLYT. These two routines, respectively, print the input data for the wing and pylon thickness and lay out the thickness panels.

A check is now made to see if a fuselage is present. If it is then subroutine BLYOUT is called to lay out the constant u-velocity panels on the fuselage.

Provision has been made for not including a store in the input data. This has been done so that the program can be used to determine the coordinates of the points at which the wing twist and camber distribution must be input. The next step in the program checks to see if there is a store

and if there is not one, the store input section of the program is bypassed. If there is a store, subroutine STORIO is called to read in and print all of the store data. These data consist of the location and incidence of the store and polynomials specifying its shape. This routine also calls subroutine BDYGEN to calculate the source and doublet distributions which represent the store volume and angle of attack effects. Following this the store is located in the fuselage coordinate system.

Boxes 2 through 4 of the left-hand column of figure 2(b) are associated with the calculation of the strengths of the constant u-velocity panels to represent the wing-pylon loading including interference effects on the fuselage. The equations used to solve for the strengths are given in section 3.3 of reference 1. The first step is to call subroutine DPCOEF which calculates the coefficient matrix, that is, the coefficients multiplying the unknown panel strengths divided by π, $(u_+/V_\infty)/\pi$. The next step is to call subroutine DPRHS which calculates the right-hand side of this set of simultaneous equations. The resulting set of equations is then solved by calling subroutine INVERS. The next section of the program prints the resulting panel strengths, control point coordinates, and interference velocities at these points. If a store is not present this ends the calculations and the program returns to the beginning to read another set of input data.

If a store is present the program continues and reads in the additional data required to describe the store to be separated and performs the initialization for the trajectory calculation. The remainder of the flow chart of figure 2(b) shows this part of the program. First the additional data describing the store are read in. These data consist of, among other things, the mass and inertia characteristics, polynomials defining the body shape, and center of gravity location. The next step is to calculate other quantities from these data.

If the store to be separated has an empennage the data required to describe the empennage are input and subroutine SEMPIN is called to initialize for the empennage force and moment calculation.

The program next reads in polynomials specifying the thrust time history if this option if being used.

Provision is made in the program for prescribing initial translational and rotational velocities relative to the parent aircraft. These are next

read in and then the initial values of the 12 dependent variables in the trajectory calculation are computed. These are:

(1) The three coordinates of the store center of moments relative to the fuselage nose.

(2) The three translational velocity components of the store center of moments relative to the fuselage.

(3) The three store rotational velocities.

(4) The three angles giving the store orientation relative to the fuselage.

Reference positions of the store nose, center of moments, and base are next calculated and the initial and final trajectory times, as well as the integration interval, are input. If the initial time is not zero, then the trajectory is being restarted from a previous run and the current values of the 12 dependent variables obtained from that run are read in. The last step in the initialization is to initialize subroutine ADAMS, the integration routine.

The last page of the flow chart, figure 2(c), is the integration loop of the program. The first steps are to calculate the aerodynamic forces and moments acting on the body and the empennage, if one is present, of the separated store. The body forces and moments are determined by using the equation presented in section 5.2 of reference 4. The empennage forces and moments are determined as discussed in section 5.3 and Appendix I of that reference.

One of the options in the computer program is to calculate a wind-tunnel captive-store trajectory as opposed to a free-flight trajectory. It is customary in the wind tunnel to change the store orientation relative to the parent aircraft while measuring the forces and moments in order to approximately account for the store translational motion relative to the aircraft. The computer program also does this during the force and moment calculation. Thus, if a captive-store trajectory is being calculated, the next step in the program is to put the store back to its correct orientation and call subroutine DIRCOS in order to calculate the free-flight direction cosines between the store body coordinate system and the inertial coordinate system which is fixed in the fuselage.

The next series of steps determines the store translational and rotational accelerations. This involves solving the set of six simultaneous

equations given by equations (II-16) through (II-18) and (II-41) through (II-43) of Appendix II of reference 4 making use of the relationships given in section 6.1 of that reference. The equations in section 6.1 are unchanged except for the expression for the force acting along the store axis, F_x, in equation (65). This now contains a thrust term so that

$$F_x = mg_x - q_{\infty_s} S_R C_A + F_T \tag{1}$$

The coefficient matrix is first calculated and then the right-hand sides are determined. Subroutine INVERS is called to solve the set of six equations for the accelerations. The rates of change of the orientation angles are next determined from equation (II-1) of Appendix II of reference 4.

A check is next made to see if output is to be printed at the end of an integration step. If output is not required the integration continues by calling subroutine ADAMS. If it is required, subroutine SOUTPT is called. Upon return from this subroutine a check is made to see if the time is equal to the final time which was read in and if so, the trajectory is stopped and control returns to the beginning of the program to read in a new set of data. If it is not, the integration is continued.

3.2 Input Data

This section of the report will describe in detail the preparation of the input data deck for the program. Whenever possible the input format has not been changed from that used in the subsonic store separation program described in reference 3. If an item of input has not been changed, the item number of figure 14 of reference 3 will be given. Other remarks which will aid the user who is familiar with the subsonic program will be made where appropriate.

3.2.1 Input format

The format for the input data for the trajectory program is shown in figure 3. Three lines of information are shown for each item. The first line gives the program variable names, the second line shows the card column fields into which the data are to be punched, and the third line shows the FORTRAN format type. Data punched in I and E formats are right

justified in the fields whereas data in F format can be punched anywhere in the field. A decimal point should be included in both E- and F-type data.

Item number 1 (item no. 1, ref. 3) is an index NCARDS which indicates how many cards of information are to follow to identify the run, item number 2. The value of NCARDS must be one or greater.

Item number 2 (item no. 2, ref. 3) is a set of NCARDS cards containing hollerith information identifying the run and may start and end anywhere on the card. The cards are reproduced in the output just as they are read in.

Item number 3 (item no. 3, ref. 3) consists of one card and contains the following information:

ALFAC	fuselage and wing angle of attack, degrees
GAMF	fuselage flight path angle, degrees
FMACH	Mach number
RHO	air density at flight altitude, slugs per cubic foot
VINF	aircraft free-stream velocity, feet per second

The aircraft is assumed to be flying in a straight line; however, it can be climbing or diving. For climbing flight, GAMF is positive. The Mach number should be between 1.2 and 3.0 and the angle of attack should not exceed 10^{o}. The wing-fuselage flow model is valid within these limits.

Item number 4 contains four indices which specify what aircraft components are present. They are

NFU	fuselage?	NFU = 0, no NFU = 1, yes
NPY	pylon?	NPY = 0, no NPY = 1, yes
NDMY	dummy variable which is not used; input as NDMY = 0	
NSTRS	store?	NSTRS = 0, no NSTRS = 1, yes

Provision has been made for omitting the fuselage. For cases with no fuselage the reference coordinate system is fixed at the wing root-chord leading edge and ALFAC and GAMF of item 3 should be the wing angle of attack and

flight path angle, respectively. The present version of the program is limited to one store, NSTRS = 1. The number of stores can be zero. This allows the input data deck through item 23 to be checked out without running a trajectory. In addition, the program can also be used as an aid in determining the points on the wing and pylon at which the slopes of the camberline and the thickness distribution are to be input.

Item number 5 (item no. 5, ref. 3) consists of two quantities which are

FLTHC length of fuselage, feet

FRMAX maximum fuselage radius, feet

This item and the next five are omitted if NFU = 0.

The next three items of input, items 6, 7, and 8 (item no. 9, 10, and 11, ref. 3) are included in the data deck only when NFU = 1. They are

NFPOLY number of polynomials specifying circular fuselage shape
$1 \leq$ NFPOLY ≤ 7

FXEND(J) x/ℓ of end points of polynomials specifying fuselage shape, NFPOLY values

FCOEF(J,K) coefficients of polynomials specifying shape

These data specify the radius distribution of the fuselage and are used in the calculation of the source-sink distribution which represents the fuselage volume and the doublet distribution which represents the fuselage angle of attack effects. Up to seven polynomials may be used. The polynomial programmed is

$$\frac{r}{\ell} = C_1 + C_7\sqrt{C_2\left(\frac{x}{\ell}\right)^2 + C_3\left(\frac{x}{\ell}\right) + C_4} + C_5\left(\frac{x}{\ell}\right) + C_6\left(\frac{x}{\ell}\right)^2 \qquad (2)$$

where C_1 through C_7 are the coefficients, r is the local fuselage radius, and ℓ is the fuselage length. The polynomials must be input for shapes which are made dimensionless by the body length since the trajectory program is written assuming this to be the case.

Item 6 specifies the number of polynomials. Item 7 consists of one card which contains the NFPOLY values of the end points of the polynomials describing the shape. The decimal point can be placed anywhere in the ten-column field.

Item number 8 on figure 3 is a set of NFPOLY cards specifying the values of the coefficients of the polynomials, equation (2). All seven coefficients are input even though some of them may be zero.

Items 9 and 10 specify the body interference panel layout and the number of line sources and line doublets to be used to model the fuselage volume and angle of attack effects. The various quantities are

NCWB	number of rings of body interference panels
NBDCR1	number of panels in a ring lying above the wing on the left half of the fuselage
NBDCR2	number of panels in a ring lying below the wing on the left half of the fuselage
NFSOR	number of line sources and line doublets to be used over the fuselage length
BODYPL	length of fuselage over which body interference panels are to be placed, feet

The definitions of some of the above quantities can be clarified by the use of figure 4. In this simplified layout there are five rings of body interference panels, NCWB = 5, laid out over the length BODYPL. In each of the rings one panel, NBDCR1 = 1, lies above the wing $z_w = 0$ plane and three panels, NBDCR2 = 3, lie below this plane. Some general rules for determining the input values of these quantities will now be given.

First, consider the length BODYPL. If the wing trailing edge is supersonic, that is, the component of the free-stream Mach number perpendicular to the trailing edge is greater than one, body interference panels should be laid out over the wing root-chord length, CRW, in figure 4. For a subsonic wing trailing edge, (perpendicular Mach number less than one) body interference panels should extend to the x_w coordinate of the wing tip-chord trailing edge. If the base of the store being separated is behind the point selected using the preceeding rules then the body interference panels should be extended to the x_w location of the store base. When a large rearward axial motion of the store is expected during its trajectory, the body panels should be extended aft to cover this motion.

The number of rings of body interference panels and the number of panels in each ring will now be discussed. The number of rings is dependent on the number of panels in a chordwise row on the wing, NCW. This quantity is input as part of item 13 and its selection will be discussed later. Since, if at all possible, the same number of chordwise panels should be

used on both the body and the wing, over the wing root-chord region, CRW in figure 4, the number of rings of body panels can be found using the following relation

$$NCWB = NCW\left(\frac{BODYPL}{CRW}\right)$$

The value of CRW is determined as described under the item 12 input data. In the above expression, the value of BODYPL should be adjusted so that NCWB is an integer.

Two rules can be given for selecting the number of panels in a ring on the left side of the body. If the store to be separated is under the fuselage at least eight panels should be used. If the store is under the wing, six panels is probably sufficient. The two input parameters specifying the number in a ring are NBDCR1 and NBDCR2. These are the number of panels above and below the wing $z_w = 0$ plane of figure 4. Assume that six panels are to be used in a ring, for example. Then, the values of NBDCR1 and NBDCR2 for three specific wing positions are:

Mid-wing

NBDCR1 = NBDCR2 = 3

Wing tangent with top of fuselage

NBDCR1 = 0, NBDCR2 = 6

Wing tangent with bottom of fuselage

NBDCR1 = 6, NBDCR2 = 0

For wings located intermediate between the mid-wing and high or low wing positions, the panels should be divided above and below the wing so that the panel widths are as equal as possible.

The computer program as documented in this report limits the total number of constant u-velocity panels or interference panels on the fuselage, wing, and pylon to 200. This limit is imposed by the dimensions of certain arrays in the program. Thus, the following relation which is based on the total panel layout must be satisfied.

$$NCWB*(NBDCR1 + NBDCR2) + NCW*MSW + NCP*MSP \leq 200 \qquad (3)$$

The variables NCW and MSW are input in item 13 and the variables NCP and MSP are input in item 20.

The last input variable in item number 9 is NFSOR, the number of line sources and line doublets to be used to model the fuselage volume and angle of attack effects. The general rule to use in determining NFSOR is that the distance between the origins of successive sources or doublets should be approximately equal to the chordwise length of the wing constant u-velocity panels at the root chord. Therefore,

$$\text{NFSOR} \cong \text{NCW}\left(\frac{\text{FLTHC}}{\text{CRW}}\right) \leq 100$$

The maximum value of 100 is imposed by dimension statements in the program. The fuselage length, FLTHC, was input as part of item 5. The wing root chord length, CRW, will be input as part of item 12 and the number of chordwise panels, NCW, will be input as part of item 13.

Item number 11 (item 15, ref. 3) contains two parameters which specify the wing location relative to the fuselage nose. The two parameters are shown pictorially in figure 5 and are

XBWOC — x_{B_W} location of intersection of wing leading edge with fuselage, feet; negative as shown in figure 5

ZBWO — z_{B_W} location of intersection of wing leading edge with fuselage, feet; negative as shown in figure 5

Item 12 (item 16, ref. 3) contains

CRW — wing root chord length, feet

SSPAN — wing semispan, feet

The definitions are shown in figure 6. The wing root chord is the wing chord at the spanwise station, Y(1), at which the wing leading edge intersects the fuselage. Both quantities are input as positive quantities.

Items 13 and 14 (items 17 and 18, ref. 3) are input data describing the left wing panel which are used to lay out the constant u-velocity panels. The quantities are

NCW — number of panels in a chordwise row on wing; NCW ≥ 4

MSW — number of panels in a spanwise row on wing; also number of thickness strips in a spanwise row on wing; MSW ≤ 19

I — wing u-velocity panel side-edge number; I = 1 to MSW+1

Y(I)	y_w location of I^{th} side edge on the left wing panel, feet (I = 1 value shown in figure 6, negative for all I's since on left panel; measured in wing planform plane)
PSIWLE(I)	leading-edge sweep of wing section to the right of the I^{th} side edge, degrees; positive swept back (measured in wing planform plane)
PSIWTE(I)	trailing-edge sweep of wing section to the right of the I^{th} side edge, degrees; positive swept back (measured in wing planform plane)

Based on these input data, the wing is divided chordwise and spanwise into trapezoidal shaped elemental panels and one constant u-velocity panel placed on each panel. All the NCW panels in a chordwise row have equal chords and spans, the spans being determined by the Y(I)'s.

The question arises as to the values to use for NCW and MSW. No specific rules can be given since the number needed is determined to some extent by the wing planform shape and the location of the store being separated. Fewer panels can be used on the wing if the store is under the fuselage than if the store is under the wing. The number of panels in a chordwise row, NCW, is also determined to some extent by the camber distribution, if any, since the camber is specified at the panel control points. If the wing is uncambered except near the leading edge then a fairly large number of panels is required in a chordwise row if this effect is to be included. In general a minimum of eight panels, NCW = 8, should be used.

The number of panels in a spanwise row, MSW, is controlled to some extent by the wing. A panel side edge must coincide with the root chord and each break in leading-edge sweep angle and trailing-edge sweep angle, and if a pylon is present, a trailing leg must coincide with the spanwise location of the pylon. One must also coincide with the wing tip. Consider the example wing alone of figure 7. There is a break in trailing-edge sweep at $y_w/s = -0.2$, a break in leading-edge sweep at -0.4, and a pylon at -0.6. To place a panel side edge at each of these positions plus the wing tip and the root chord requires four panels across the semispan. This is the minimum number which can be used for this wing and is probably not sufficient. Experience with the program has shown that in some cases six panels in a spanwise row has given good results.

The only sure way of determining convergence with number of panels, both NCW and MSW, is to examine the results obtained from the trajectory program. For a particular wing various panel layouts should be tried to

assure convergence. The minimum number of panels, consistent with the desired accuracy, should be used in order to minimize the trajectory calculation time.

The maximum number of panels that can be placed on the left wing panel, pylon, and fuselage, is 200. This limit is imposed by dimension statements in the computer program. Thus, limits may be imposed on NCW and NSW in order to satisfy equation (3).

Item number 14 consists of a deck of MSW+1 cards. The index I is the panel side-edge number. The side edges are numbered from the root chord, I = 1, to the tip, I = MSW + 1. Associated with each I are the spanwise location of the side edge, Y(I), and the sweep angles of the leading and trailing edges of the wing segment to the right of the side edge, PSIWLE(I) and PSIWTE(I). When I = 1 the sweep angles are zero.

The two indices of item number 15 (item 19, ref. 3) are associated with the wing twist and camber distribution.

NTAC — twist and/or camber? NTAC = 0, no; NTAC = 1, yes

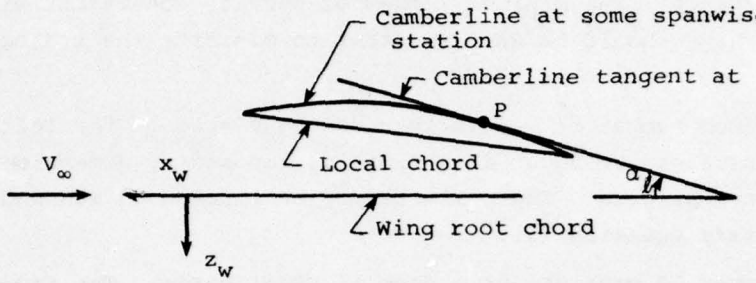
NUNI — if wing has no twist, and the camber distribution is similar at all spanwise stations, NUNI = 1; for all other cases, NUNI = 0 (omit if NTAC = 0)

If NTAC = 1, item number 16 is included in the input data deck. These data specify the wing twist and/or camber distribution in terms of the tangent of the local angle of attack of the camberline for a wing root chord angle of attack of zero degrees. The function of the index NUNI is explained below.

ALPHAL(J) — $\tan \alpha_\ell$ of the wing camberline at the constant u-velocity panel control points. If NUNI = 1, only data for the chordwise row adjacent to the root chord are input. The first value is for the control point nearest the leading edge. If NUNI = 0, data for all chordwise rows must be input starting nearest the root chord and working outboard. Data for each row start on a new card (omit if NTAC = 0)

The constant u-velocity panel control points are at 95 percent of the chord which passes through the centroid of area (see fig. 2, ref. 1) of each elemental panel laid out by NCW, MSW, and the Y(I)'s of items 13 and 14.

The values of ALPHAL(J) are obtained as follows. Consider the following sketch



which shows the cambered and twisted section of the lifting surface at some spanwise station for zero wing angle of attack. At the control point P, a tangent to the camberline is constructed, which makes an angle α_ℓ with the root chord (the x_w axis). The positive sense of α_ℓ is shown. The input value required is ALPHAL(J) = tan α_ℓ. For wings which have the same camber distribution at all spanwise stations and no twist, NUNI = 1, data are only input for the row of control points closest to the root chord. The program assigns these values to all other rows.

The two indices of item number 17 (item 21, ref. 3) are associated with the specification of the wing thickness distribution. They are

NCWS — number of thickness panels in a chordwise row on the wing; NCWS*MSW + NCPS*MSP $\leq$ 400

NUNIS — if wing has a similar thickness distribution at all spanwise stations, NUNIS = 1; if not, NUNIS = 0

A total of 400 thickness panels can be used on the wing and pylon. The thickness panels are also trapezoidal in shape. The span of each chordwise strip is the same as the corresponding row of constant u-velocity panels. Each of these chordwise strips is divided into NCWS equal chord panels. Usually more thickness panels should be used in a chordwise row than u-velocity panels. Experience with the program has shown 12 to 14 thickness panels usually to be sufficient. Again, this can only be checked by varying the number and examining the resulting store load distributions. The minimum number, consistent with the desired accuracy, should be used to minimize the trajectory calculation time.

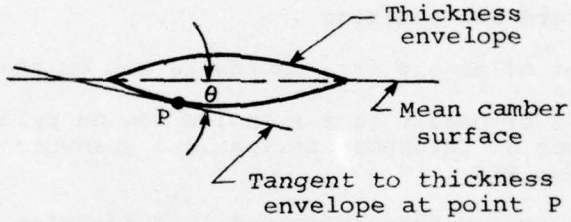
The data contained in item number 18 are

THETAL(J) — slope of the wing thickness distribution at the centroid of area of the thickness panels. If NUNIS = 1 only data for the chordwise row adjacent to the root chord are input. The first

value is for the panel at the leading edge. If NUNIS = 0, data for all chordwise rows must be input starting at the root chord and working outboard. Data for each row start on a new card.

Note that the values of the thickness slopes are input for the centroid of area of each panel. Also, for wings with similar distributions at all spanwise stations, NUNIS = 1, data are only input for the row of panels adjacent to the root chord. The program assigns these values to all other rows.

The values of THETAL(J) are obtained as follows. Consider the following sketch



which shows the thickness envelope at a spanwise station where the slopes are to be determined. At a point P at which the slope is required, a tangent to the surface of the thickness envelope is constructed which makes an angle θ with the line connecting the leading and trailing edges of the envelope. The input value required is THETAL(J) = tan θ. Forward of the point of maximum thickness θ is positive and aft of this point it is negative. In some cases for a wing with a blunt leading edge, the thickness slope should not exceed a certain value. The determination of this value is discussed in detail in section 3.4.1 of reference 1.

The next five items of input data, items 19 through 23, are associated with the pylon if one is present, NPY = 1. Item number 19 (item 23, ref. 3) consists of one card and contains the following quantities

IP — index of the y_w location of the pylon. This must be one of the Y(I)'s read in for the wing, item 14, if under the wing. If pylon is under the fuselage centerline, IP = 0.

PSIPLE — sweep angle of the pylon leading edge in the wing coordinate system, degrees. Positive sweep is swept back.

PSIPTE — sweep angle of the pylon trailing edge in the wing coordinate system, degrees. Positive sweep is swept back.

CRP	length of pylon root chord, feet
HP	height of pylon measured from wing chordal plane, feet
XPLE	location of pylon root chord leading edge measured from local wing chord leading edge, feet (negative behind)

The pylon location can be under the fuselage, IP = 0, or at one of the wing-panel side-edge locations outboard of the wing-fuselage juncture. The remaining five items are shown in figure 8 for the two cases.

The next two items of input data are associated with the constant u-velocity panels which represent the pylon loading. Item number 20 (item 24, ref. 3) is one card and contains

NCP	number of panels in a chordwise row on the pylon
MSP	number of panels in a spanwise row on pylon; also number of thickness strips in a spanwise row on the pylon; MSP $\leq$ 19

For a typical pylon, two or three vortices in a spanwise row are sufficient, MSP = 2 or 3. The number chordwise, NCP, depends on the length of the pylon root chord. The chordwise dimensions of the trapezoidal shaped area elements on the pylon should be approximately the same as those on the wing immediately above the pylon. That is, the local wing chord length divided by NCW of item 13 should be approximately equal to the pylon root chord length divided by NCP.

The program is limited to 200 constant u-velocity panels on the wing-pylon-fuselage combination. Thus,

$$NCWB*(NBDCR1 + NBDCR2) + NCW*MSW + NCP*MSP \leq 200$$

Item number 21 (item 25, ref. 3) consists of a deck of MSP+1 cards which contain the following information

K	pylon-panel side-edge number; K = 1 to MSP+1
Z(K)	z location of the K^{th} side edge, feet; measured from local wing chord

The first side edge should be placed on the pylon root chord and the last on the tip chord. The remaining side edges should be equally spaced between these two. For a pylon located under the fuselage, figure 8(a), Z(1) should be the z_w location of the pylon root chord, the bottom of the fuselage, and Z(MSP+1) should equal Z(1) + HP. For a pylon under the wing, Z(1) = 0 and Z(MSP+1) = HP.

Items 22 and 23 provide data required to model the pylon thickness distribution. The data are

NCPS	number of thickness panels in a chordwise row on the pylon
NUNIP	if pylon has a similar thickness distribution at all spanwise stations, NUNIP = 1; if not, NUNIP = 0
THETPL(J)	slope of the pylon thickness distribution at the centroids of the thickness panels. If NUNIP = 1 only data for the chordwise row adjacent to the root chord are input. The first value is for the panel at the leading edge. If NUNIP = 0, data for all chordwise rows must be input starting at the root chord and working outboard. Data for each row start on a new card.

These data are prepared in the same manner as were the corresponding data for the wing thickness, items 17 and 18. The comments made there concerning a blunt leading edge also apply to the pylon.

The program is limited to 400 thickness panels on the wing-pylon combination. Thus,

$$NCWS*MSW + NCPS*MSP \leq 400$$

If no store is present, NSTRS of item 4 is equal to zero, this concludes the input data deck. If NSTRS is not zero, then the next five items of input data describe the store to be separated. The store to be separated must be under the fuselage centerline or to the left of this line as seen by the pilot.

Item number 24 (item 32, ref. 3) is one card which contains the following information.

NUMSTR(1)	store number; ≤ 99
NSHAPE(1)	shape number of store; ≤ 99
SLTHC(1)	length of store, feet
SRMAX(1)	maximum radius of store, feet
XSNC(1)	x_ℓ location of store nose measured from wing chord leading edge immediately above store, feet; positive ahead
YSN(1)	y_w location of store nose measured from fuselage centerline, feet; positive to the right

ZSN(1) z_ℓ location of store nose measured from wing chord leading edge immediately above store, feet; positive below

SIC(1) store incidence angle measured relative to wing root chord, degrees; positive nose up

The subscripted form of the above variables is not required since the present program is limited to one store. It is used in anticipation of a future version of the program treating multiple stores.

Item number 25 is one card which contains

NSHPT number of different values of NSHAPE(J) from item number 24; since J is limited to one, NSHPT is input as NSHPT = 1

Item number 26 is one card. The data are

MSHAPE shape number of store represented by the shape data to be read in as item numbers 27, 28, and 29; input as MSHAPE = NSHAPE(1) which was input in item number 24

MSOR number of line sources and line doublets to be used to model the store volume and angle of attack effects; MSOR $\leq$ 100

A value of MSOR of 30 to 40 should be sufficient to model the store. The value can be varied to determine whether increasing the number affects the trajectory.

The next three items of input data, item numbers 27, 28, and 29, specify the store shape. They are

NSPOLJ number of polynomials specifying store shape; $1 \leq$ NSPOLJ ≤ 7

SXNDJ(J) x/ℓ of end points of polynomials specifying store shape; NSPOLJ values

SCOFJ(J,K) coefficients of polynomials specifying shape

These data specify the radius distribution of the store and are used in the calculation of the source-sink distribution which represents the store volume and the doublet distribution which represents the store angle of attack effects. Up to seven polynomials may be used. The polynomial programmed is given by equation (2). The polynomials must be input for a shape which is made dimensionless by the store length since the trajectory program is written assuming this to be the case.

Item 27 specifies the number of polynomials. Item 28 consists of one card which contains the NSPOLJ values of the end points of the polynomials describing the shape.

Item number 29 is a set of NSPOLJ cards specifying the values of the coefficients of the polynomials, equation (2). All seven coefficients are input even though some of them may be zero.

The remainder of the input data deck is additional data required to describe the store being separated and other data required for the force and moment and trajectory calculations.

Item number 30 is one card which contains nine indices. They are

NEJECT — number of the store being separated

NSEG — number of equal length segments the body is to be broken into for the force calculation; NSEG $\leq$ 40

NSEGXO — number of body segments to the flow separation location

NGAM — trajectory to simulate wind-tunnel captive-store trajectory? NGAM = 0, no
NGAM = 1, yes

NPOLY — number of polynomials specifying store shape; $1 \leq$ NPOLY ≤ 7

NROLL — rolling moment to be calculated?
NROLL = 0, no
NROLL = 1, yes

NEMP — empennage present?
NEMP = 0, no
NEMP = 1, yes

NDAMP — damping to be included in force calculation?
NDAMP = 0, no
NDAMP = 1, yes

NTHRUS — thrust time history to be specified?
NTHRUS = 0, no
NTHRUS = 1, yes

The index NEJECT must be the store number that was read into NUMSTR(1) in item 24; that is, NEJECT = NUMSTR(1).

In the store body force and moment calculation, the body is divided into NSEG equal length segments. Experience in using the program has shown that 20 body segments, NSEG = 20, usually yields converged forces and moments. This can only be checked for a specific case by varying NSEG

and comparing results. To minimize calculation time NSEG should be kept as small as possible.

No definite rules for the selection of a value of NSEGXO can be given. For stores with cylindrical afterbodies NSEGXO should probably be input as NSEG. For stores with boattailed afterbodies NSEGXO should probably be less than NSEG. In the subsonic store separation program, reference 3, an empirical relationship was given for estimating the flow separation location for boattailed bodies. This relationship was obtained from data taken at low subsonic speeds and is, therefore, probably not valid at supersonic speeds. In using the present program NSEGXO should be set equal to NSEG unless experimental data are available for determining the separation location.

The index NGAM is included as input solely for the purpose of allowing the program to be used to compare with captive store data obtained in the wind tunnel. Since the wind tunnel cannot produce a flow where the store sees a free-stream velocity coming from a different direction than that seen by the parent aircraft the captive store case must be handled differently by the computer program.

The number of polynomials specifying the store shape, NPOLY, is the number required to specify the shape from the store nose to its base. These polynomials are used in the force and moment calculation. The polynomials are of the form given by equation (2). The value of NPOLY should be equal to the value of NSPOLJ read in under item 27.

The next index, NROLL, indicates whether or not the rolling moment for a store with an empennage is to be calculated. NEMP specifies whether there is an empennage. The index, NDAMP, is used by the program to determine whether or not aerodynamic damping in pitch, yaw, and roll is to be included in the force and moment calculation. The last index, NTHRUS, specifies whether a thrust time history is to be read in.

Item number 31 (item 38, ref. 3) is one card and specifies the store mass and inertia characteristics. The quantities are

SMASS	store mass, slugs
FIXX	I_{xx} moment of inertia, slug-ft^2

FIYY	I_{yy}	moment of inertia, slug-ft^2
FIZZ	I_{zz}	moment of inertia, slug-ft^2
FIYZ	I_{yz}	product of inertia, slug-ft^2
FIXZ	I_{xz}	product of inertia, slug-ft^2
FIXY	I_{xy}	product of inertia, slug-ft^2

The equations defining the moments and products of inertia are given by equation (II-36) of Appendix II of reference 4.

The one card of item number 32 (item 39, ref. 3) contains

XMOM	location along store axis about which the pitching and yawing moments are to be taken, negative behind nose, feet; same point about which moments of inertia are taken
XBAR	x location of store c.g. measured from moment center, feet; positive forward
YBAR	y location of store c.g. measured from store axis, feet; positive to the right
ZBAR	z location of store c.g. measured from store axis, feet; positive below

The next two items of input, items 33 and 34 (items 40 and 41, ref. 3) describe the shape of the separated store. The quantities contained on the cards are

XEND(J)	x/ℓ of end points of polynomials specifying shape of ejected store, NPOLY values
COEF(J,K)	coefficients of polynomials specifying shape

These data are a repeat of the data input as items 28 and 29.

Item number 35 (item 42, ref. 3) contains two quantities which are

CA	store axial-force coefficient; reference area is store maximum cross-sectional area
CDC	crossflow-drag coefficient

The store axial-force coefficient is not calculated by the computer program so it is required input. The crossflow-drag coefficient is used in the viscous crossflow force and moment calculation which is used in place of the slender-body calculation behind the assumed separation location. It is defined as the section drag coefficient of a circular cylinder normal

to the airstream. That is,

$$CDC = c_{d_c} = \frac{\text{drag per unit length}}{q_\infty (2r)}$$

where q_∞ is the free-stream dynamic pressure and r is the cylinder radius. The value commonly used for this parameter is 1.2. If a more accurate value is desired, reference 7 should be used. The crossflow-drag coefficient is presented there as a function of crossflow Mach number and Reynolds number.

The next two items of input data are included in the input data deck if an empennage is present, NEMP = 1 in item 30. Item 36 (item 43, ref. 3) contains

IPLNR	IPLNR = 0, cruciform empennage IPLNR = 1, planar empennage
MSF	number of spanwise control points on each fin; must be odd and $5 \leq MSF \leq 11$

It should be noted that MSF must be odd and in the range $5 \leq MSF \leq 11$. For most stores, MSF = 5 has been found to give accurate results. The larger the fin span to body radius ratio, the more points required.

The next card, item number 37 (item 44, ref. 3) contains

XTAIL	x location at which empennage forces are to act measured from store nose, feet; negative number
RADAV	average store body radius in empennage region, feet; positive number
FINSS	tail fin semispan, measured from body longitudinal axis, feet; positive number
PHIROL	initial fin orientation, degrees; $0 \leq PHIROL \leq 90$; PHIROL = 0 if fins vertical and horizontal for a cruciform empennage or horizontal for a planar empennage.
CLALPH	lift-curve slope of two exposed panels joined together, per radian; reference area is store maximum cross-sectional area

The location at which the empennage forces are assumed to act, XTAIL, is arbitrary. It can be estimated using Chart 10 of reference 8. The lift-curve slope, CLALPH, can be estimated using the method described in Appendix II.

The next three items of input data, item numbers 38, 39, and 40, are included in the input data deck if a thrust time history is to be specified, NTHRUS = 1, in item 30. The input quantities are

NTPOLY — number of polynomials specifying thrust time history; $1 \leq$ NTPOLY ≤ 5

TEND(J) — final times over which the NTPOLY polynomials apply

TC(J,K) — coefficients of the polynomials

The thrust force as programmed acts along the store longitudinal axis and is specified by a series of polynomials of the form

$$F_T = \sum_{k=1}^{6} a_k t^{k-1} \tag{4}$$

where F_T is the thrust force in pounds. Up to five polynomials can be used to specify the time history and item 38 contains the number of polynomials, NTPOLY. Item 39 is one card and contains the NTPOLY values of the final time t over which each of the polynomials applies. That is, the first polynomial is used from $t = 0$ to $t =$ TEND(1), the second from $t =$ TEND(1) to $t =$ TEND(2), and so on. Item 40 is a set of NTPOLY cards which specify the six a_k coefficients, TC(J,K), for the polynomials.

Item number 41 (item 45, ref. 3) is one card which contains the initial velocities of the store relative to the parent aircraft such as might exist at the end of the ejection stroke of a store released from a pylon or rack. The six quantities are

VXZERO — store forward ejection velocity parallel to store longitudinal axis, ft/sec

VYZERO — store lateral ejection velocity perpendicular to store longitudinal axis, ft/sec

VZZERO — store downward ejection velocity perpendicular to store longitudinal axis, ft/sec

VAR(4) — store initial rolling rate, p, radians/sec

VAR(5) — store initial pitching rate, q, radians/sec

VAR(6) — store initial yawing rate, r, radians/sec

At the beginning of the trajectory, the end of the stroke, the store is oriented such that its y-axis is parallel to the y_B fuselage axis shown in figure 5. Thus the x,z store plane is parallel to the x_B, z_B fuselage plane and the initial vertical translational velocity of the store, VZZERO, is in the x,z store plane perpendicular to the store x-axis. The velocities VXZERO, VYZERO, and VZZERO are positive in the positive x,y,and z directions.

The next card, item number 42 (item 46, ref. 3) contains three items. They are:

DTIME	integration interval, seconds
TIMEI	initial time, seconds
TIMEF	final time, seconds

The first, DTIME, is the integration interval to be used in the integration subroutine, ADAMS. The version of this routine which is included in the present program does not adjust this interval to satisfy certain accuracy requirements but uses the input value of DTIME. A value of 0.025 or 0.01 seconds should probably work. However, different values should be tried and the trajectory results compared to determine that a small enough value is being used. Finned stores probably require a smaller value than finless stores because of the large moments produced by the empennage. The initial time, TIMEI, must be input as 0.0 unless a trajectory is to be restarted using the last page of output from a previous run to obtain the initial conditions. Then, TIMEI should be given the value that appears on that page of output. The final time, TIMEF, is the time at which the trajectory calculation is to be terminated. Except for very unusual trajectories, a value of 0.5 to 0.7 second is normally sufficient to determine if a store will or will not clear the aircraft.

If TIMEI and TIMEF are both input as zero, no trajectory calculation will be performed. However, the store load distributions and forces and moments will be calculated with the store in its initial position. This feature can be useful in checking out the entire input data deck prior to running a trajectory or for studying store loadings at specific points.

The last item of input, item number 43 (item 47, ref. 3) is input only if a trajectory is being restarted, TIMEI $\neq 0$. This item consists of two cards with VAR(1) through VAR(8) on the first card and VAR(9) through VAR(12) on the second. The following table gives the notation used to identify

VAR(1) through VAR(12) on the trajectory program output which will be discussed in section 3.3.

Program Notation	Output Notation
VAR(1)	DXF, ft/sec
VAR(2)	DYF, ft/sec
VAR(3)	DZF, ft/sec
VAR(4)	P, radians/sec
VAR(5)	Q, radians/sec
VAR(6)	R, radians/sec
VAR(7)	XF of XMOM, ft
VAR(8)	YF of XMOM, ft
VAR(9)	ZF of XMOM, ft
VAR(10)	PSI, degrees
VAR(11)	THETA, degrees
VAR(12)	PHI, degrees

3.2.2 Sample input data

A sample input data deck will now be presented. It will utilize the wind tunnel model components shown in figure 9 except that they have been scaled up by a factor of twenty in order to approximate a full-scale aircraft and store. The models were used in the wind tunnel test program described in reference 9.

The configuration for the sample calculation is shown in figure 10. It consists of the wing-fuselage combination of figure 9(a), the pylon of figure 9(b), and the store of figure 9(c). The store is shown in its initial position which is one store body radius below the carriage position on the pylon.

The input data deck is tabulated in figure 11. The item numbers indicated on the figure correspond to those of figure 3. The first item on figure 11 is item number 1 which contains the value of NCARDS, in this case 12. This is followed by item number 2, which consists of these 12 cards of identifying information.

Item number 3 specifies the aircraft flight conditions. The angle of attack is 2.0°, the flight path angle is -30.0°, and the Mach number is 1.5. The free-stream air density is 0.0008907 slugs per cubic foot, which corresponds to a flight altitude of 30,000 feet. The flight velocity is 1,492.0 feet per second.

The aircraft components which are present are specified by item 4. There is a circular fuselage, NFU = 1; there is a pylon, NPY = 1; and there is a store, NSTRS = 1.

Items 5 through 10 are the fuselage input data. Item 5 contains the fuselage length and maximum radius which are shown in figure 10. Items 6, 7, and 8 contain the data for the polynomials specifying the fuselage shape. The fuselage radius distribution, made dimensionless by the fuselage length is tabulated in figure 9(a). *Three polynomials*, item 6, are required. The nose is a circular arc and ends at $x/\ell = 0.32$, the centerbody is a cylinder ending at $x/\ell = 0.7534$, and the aft position is a circular arc ending at $x/\ell = 1.0$. These three values of x/ℓ are contained in item 7.

The next three cards, item 8, *contain the coefficients of the polynomials* for the three sections (see eq. (2)). The nose section is a circular arc so that when the following boundary conditions are imposed

$$\frac{x}{\ell} = 0 \qquad \frac{r}{\ell} = 0$$

$$\frac{x}{\ell} = 0.32 \qquad \frac{r}{\ell} = 0.0457$$

$$\frac{x}{\ell} = 0.32 \qquad \frac{dr}{dx} = 0$$

the equation of a circle gives

$$\frac{r}{\ell} = -1.0975 + \sqrt{-\left(\frac{x}{\ell}\right)^2 + 0.64\left(\frac{x}{\ell}\right) + 1.2045} \tag{5}$$

The cylindrical section runs from $x/\ell = 0.32$ to $x/\ell = 0.7534$ and in this region

$$\frac{r}{\ell} = 0.0457 \tag{6}$$

The last section is also a circular arc. By imposing the following three boundary conditions

$$\frac{x}{\ell} = 0.7534 \qquad \frac{r}{\ell} = 0.0457$$

$$\frac{x}{\ell} = 0.7534 \qquad \frac{dr}{dx} = 0$$

$$\frac{x}{\ell} = 1.0 \qquad \frac{r}{\ell} = 0.0302$$

we obtain

$$\frac{r}{\ell} = -1.9237 + \sqrt{-\left(\frac{x}{\ell}\right)^2 + 1.5068\left(\frac{x}{\ell}\right) + 3.3109} \tag{7}$$

Equations (5), (6), and (7) are the three polynomials describing the shape of the fuselage and the coefficients are contained in the three cards making up item 8.

Item number 9 contains four indices. The first three specify the constant u-velocity panel layout on the fuselage and the last specifies the number of line sources and line doublets to be used to represent the fuselage volume and angle of attack effects. Twelve (12) rings of panels are to be laid out on the fuselage with eight in each ring. Since this is a mid-wing configuration, four panels in each ring are to be placed above the wing and four below. The number of sources and doublets to be used is 52.

Item number 10 specifies the fuselage length over which the constant u-velocity panels are to be laid out. Since, in this case, the wing trailing edge is supersonic and no rearward motion of the store is expected, BODYPL is input as the length of the wing root chord shown in figure 10.

The next eight items, items 11 through 18, are the wing input data. Item 11 gives the position of the wing root chord leading edge relative to the fuselage and item 12 specifies the root chord length and the semispan. These quantities are shown in figure 10. Items 13 and 14 are data required by the program to lay out the constant u-velocity panels. There are to be 12 panels in each chordwise row and 8 of these rows across the semispan. This requires that the spanwise locations on the left wing panel of nine side edges and the sweep angles to the right of these points be specified.

These data are contained on the nine cards of item 14. Note that the first side edge coincides with the wing-fuselage juncture. Item 15 indicates that the wing has neither twist nor camber, and thus item 16 is omitted from the input data deck. Items 17 and 18 specify the wing thickness distribution. The two indices on item 17 indicate that 20 thickness panels are to be placed in a chordwise row and that the wing has a similar thickness distribution at all spanwise stations. Consequently, item 18 consists of three cards with the 20 values of the slope of the thickness distribution. The airfoil section specification is shown in figure 9(a). The wing leading edge in this case is subsonic. The maximum value of the slope has been selected in accordance with the discussion in section 3.4 of reference 1.

Since there is a pylon, items 19 through 23 are input and contain the pylon data. The pylon is the one shown in figure 9(b), except that it has been scaled up by a factor of 20. Item 19 of the input data specifies that the pylon is located below wing-panel side-edge number 3 (IP = 3), the leading-edge and trailing-edge sweep angles are 0.0°, the root chord length is 4.43 feet, the height is 1.65 feet, and the leading edge is 2.51 feet behind the local wing chord leading edge (see fig. 10). Items 20 and 21 are the pylon constant u-velocity panel data. There are to be four panels in a chordwise row and two of these rows spanwise. The three side edge locations are given by item 21. Items 22 and 23 specify the pylon thickness distribution. There are to be 9 thickness panels in a chordwise row and the thickness distribution is similar at all spanwise stations.

Item 24 begins the store data. The store is shown in figure 9(c) but has been scaled up by a factor of twenty. Item 24 contains the number assigned to the store and its shape number. The next five quantities in this item are the store length, maximum radius, and location of the store nose. This information is shown in figure 10. The last number is the store incidence angle which, in this case, is 0.0°.

Item number 25 contains the value of NSHPT. In the present version of this program this variable must be input as NSHPT = 1. Items 26 through 29 contain the shape data for the store. Item 26 contains the value of MSHAPE, which must be equal to the value of NSHAPE(1) read in as part of item 24. It also specifies the number of line sources and line doublets to be used for the store.

Items 27 through 29 contain the data specifying the store shape. The body shape shown in figure 9(c) is defined by two polynomials. The first is a circular arc representing the nose which ends at $x/\ell = 1.5/6.375 = 0.2353$. The second is one which specifies a constant radius, $r/\ell = 0.05882$, to the base of the store, $x/\ell = 1.0$. Item 28 contains the end points and item 29 contains the coefficients of the polynomials (see eq.(2)). The coefficients are determined in the manner previously described for the fuselage.

Item 30 contains nine indices which indicate the following. Store number 10, NEJECT = 10, is the store being separated and the body is to be broken into 20 segments, NSEG = 20, for the force and moment calculation with flow separation occurring at the base of the store, NSEGXO = 20. In addition, a free-flight trajectory is to be calculated, NGAM = 0; two polynomials are to be input describing the body shape, NPOLY = 2; rolling moment is to be calculated, NROLL = 1; the store has an empennage, NEMP = 1; aerodynamic damping is to be included, NDAMP = 1; and a thrust time history is to be specified, NTHRUS = 1.

Item 31 contains the mass and inertia characteristics of the separated store. The particular numbers used in this example have been assigned and have not been determined by specifying a density distribution of the store and performing the required integrations. Item 32 specifies that the inertia characteristics are assumed to have been calculated about a point 7.4 feet behind the store nose and that the store center of gravity is not offset from this point.

Items 33 and 34 contain the end points and coefficients of the polynomials describing the store shape. These data must be identical to the data read in as items 28 and 29.

The store axial force coefficient, 0.65, and crossflow drag coefficient, 0.0, are input in item 35. A crossflow drag coefficient is not used in this example since flow separation is not assumed to occur ahead of the store base.

Items 36 and 37 are data describing the store empennage. Item 36 specifies a cruciform empennage and five control points on each tail panel. Item 37 indicates that the empennage forces act 9.558 feet behind the store nose, the body radius in the fin region is 0.625 feet, the fin semispan is 1.45833 feet, the initial fin orientation is 45° from the vertical and horizontal, and the lift-curve slope is 3.491. The fin details are obtained from figure 9(c) and scaled up by a factor of 20.

Items 38, 39, and 40 specify the thrust time history. Two polynomials of the type given by equation (4) are to be specified. Item 39 indicates that the first polynomial will apply from $t = 0$ seconds to $t = 0.2$ seconds and the second from $t = 0.2$ seconds to $t = 1.0$ second. Item 40 contains the coefficients of the polynomials. The first polynomial is a constant thrust of 1,500 pounds. The second polynomial starts with this value at $t = 0.2$ seconds and allows the thrust to linearly decrease to zero at $t = 1.0$ second.

Item 41 specifies initial axial and lateral translational velocities of the store of 0 feet per second, an initial downward translational velocity of 10 feet per second, and initial rotational velocities of 0.0 radians per second.

The last card on input, item 42, provides the integration interval and the initial and final times. Since the initial time is 0.0, item 43 is not included.

3.3 Output from the Program

Figure 12 presents the output from the computer program for the sample case, the configuration of figure 10, and the data deck of figure 11.

The first page of output, figure 12(a), tabulates the input identifying information, the aircraft flight conditions, and the fuselage input data. Near the bottom of the page, a tabulation of the fuselage shape, as calculated from the input polynomials specifying the shape, and the source and doublet singularity origins and strengths begins. This tabulation is continued on the next page, figure 12(b).

Figure 12(c), 12(d), and 12(e) tabulate most of the wing and pylon input data. Figure 12(c) tabulates the wing data exclusive of the twist and camber and thickness distributions. Figure 12(d) tabulates the pylon data except for the thickness distribution. The input thickness distributions for the wing and pylon are tabulated in figure 12(e).

Figure 12(f) tabulates the input data for the store. Following this, a tabulation of the store shape, as calculated from the input polynomials, and the source and doublet singularity origins and strengths is presented.

The next four pages of output, figure 12(g) through figure 12(j), tabulate quantities associated with the constant u-velocity panel layout on the wing, pylon, and fuselage and the boundary condition at the control

points of these panels. The x,y,z coordinates are those of the control points in the wing coordinate system. The next three columns, U/VINF, V/VINF, and W/VINF, are the sums of the dimensionless perturbation velocities in the x_w,y_w,and z_w directions, respectively, induced at the control points by the other aircraft components. These include fuselage and store on the wing and pylon, wing thickness on pylon, pylon thickness on wing, both wing and pylon thickness on the fuselage, and store on fuselage. VINF is the free-stream velocity. The next to last column tabulates the twist and camber distribution input for the wing. The last column tabulates the singularity strengths of the constant u-velocity panels laid out on the wing, pylon, and fuselage.

Page 11 of the output, figure 12(k), indicates the number of the store separated and tabulates the additional data which were read in to describe this store.

The last three parts of figure 12, parts (l), (m), and (n), are output for three points in the trajectory, the first point, an intermediate point, and the point at which the trajectory is terminated. The complete output from the program will contain a page like this for each integration step, in this case every 0.025 seconds.

At the top of each page is the trajectory time in seconds. Following this, the forces and moments, components as well as totals, and the body load distributions and the velocity distributions along the store axis from which these load distributions were calculated are listed. The load distributions are the sums of those due to buoyancy, slender body, and viscous crossflow. The velocity field calculation is discussed in detail in sections 3.4 of reference 1 and 5.1 of reference 4 and the force and moment calculation in sections 5.2 and 5.3 of reference 4.

The following table relates the program output variables to the x_s, y_s,z_s coordinate system and positive directions shown in figure 13.

Program Notation	Notation of figure 13
CN	C_N
CY	C_Y
CLM	C_m
CLN	C_n
CLL	C_ℓ

Program Notation	Notation of figure 13
X,FT	x_s,feet
X/L	x_s/ℓ_s
DCN/DX	dC_N/dx_s,per foot
DCY/DX	dC_Y/dx_s,per foot
U/VS	U_s/V_{∞_s}
V/VS	V_s/V_{∞_s}
W/VS	W_s/V_{∞_s}

As the store pitches, yaws, and rolls during the trajectory, the x_s,y_s,z_s coordinate system pitches, yaws, and rolls with it. The velocities and forces are always calculated in this coordinate system.

The remaining quantities tabulated on each page of trajectory output specify the store location, orientation, velocities, and accelerations relative to the parent aircraft at that particular time. Before discussing these quantities, the coordinate systems must be mentioned. Figure 13 shows another coordinate system, x, y, and z, which is fixed in the store and moves with the store as it yaws, pitches, and rolls. The origin of this system is fixed at the store moment center. This coordinate system is also shown in figure 14 along with another system, ξ,η,ζ. This latter system is an inertial coordinate system whose origin is fixed in the fuselage nose and is parallel to the x_B,y_B,z_B system of figure 5. At any point in time, the two coordinate systems are orientated with respect to each other by a system of angles. The angles are those shown in figure 14 and consist of three rotations in the yaw, Ψ, pitch, Θ, and roll, Φ, sequence. The positive senses of the three store rotational velocities about the x,y,z axes are also shown in the figure.

Following the load and velocity distribution output on, for example, figure 12(m), the location of the store in the fuselage, or inertial, coordinate system is tabulated. The locations of the store nose, NOSE, moment center, XMOM, and base, BASE, are tabulated relative to the fuselage nose and also relative to the position of the store at time t = 0. In this tabulation XF is x_B of figure 5 or ξ of figure 14. Likewise, YF is y_B or η and ZF is z_B or ζ.

The next output are the translational velocities and accelerations of the store relative to the moving aircraft. For example,

$$DXF = \frac{dx_B}{dt} \text{ or } \frac{d\xi}{dt} \text{ , ft/sec}$$

$$D2XF = \frac{d^2x_B}{dt^2} \text{ or } \frac{d^2\xi}{dt^2} \text{ , ft/sec}^2$$

Next the rotational velocities shown in figure 14 are listed as are the rotational accelerations. The notation is

$$P = p, \text{ radians/sec}$$

$$Q = q, \text{ radians/sec}$$

$$R = r, \text{ radians/sec}$$

$$PDOT = \frac{dp}{dt} \text{ , radians/sec}^2$$

$$QDOT = \frac{dq}{dt} \text{ , radians/sec}^2$$

$$RDOT = \frac{dr}{dt} \text{ , radians/sec}^2$$

The last output printed at each integration step are the values of the three orientation angles shown in figure 14 and their time rates of change. The notation is

$$PSI = \Psi, \text{ deg.}$$

$$THETA = \Theta, \text{ deg.}$$

$$PHI = \Phi, \text{ deg.}$$

$$DPSI = \frac{d\Psi}{dt} \text{ , radians/sec}$$

$$DTHETA = \frac{d\Theta}{dt} \text{ , radians/sec}$$

$$DPHI = \frac{d\Phi}{dt} \text{ , radians/sec}$$

As the yaw angle, Ψ, and/or the pitch angle, θ, increase in magnitude, the velocity field which is used in the store force and moment calculation may not be calculated along the store longitudinal axis. This is the case when all or a portion of the store lies in the region where the velocity field is smoothed as described is section 3.4.2 of reference 1. The velocity field to be smoothed is calculated by making an axial traverse parallel to the fuselage longitudinal axis at the y_B, z_B location of the store center of gravity. When Ψ and θ are zero the store axis lies along this traverse. As the angles increase in magnitude, this is no longer the case. Except for an extremely long store, sufficiently accurate velocity fields are calculated up to angles of $\pm$ 10 degrees.

4. PROGRAM ERROR MESSAGES

The possible occurrence of certain fatal errors during program execution has been anticipated in the store separation trajectory program. If such an error should occur, the program has been designed to print a diagnostic error message and then halt. Each message with the routine in which the corresponding error is detected is presented in the following table. The table is arranged in alphabetical order by the first word of the message.

Routine	Error Message
BDYGEN	at base of body radial distance to Mach cone emanating from body nose is less than maximum body radius; check body input data and Mach number
NUMACH	axial location of traverse above wing chordal plane
INVERS	matrix is singular
NUMACH	no negative velocity due to wing thickness found in traverse to trailing-edge Mach cone
SSTORE	NSTRS input as (n > 1); program limited to one store
STORIO	shape polynomials not input for all stores
WITHIN	slope of wing thickness envelope at leading edge not positive for some chordwise row
THRCAL	time greater than end of specified thrust time history

If one of the above messages is printed out during program execution, the user should examine the input data for possible error.

5. PROGRAM RUNNING TIMES

The program described in this report has been run on the CDC 6600. Because of machine differences, the running time varies from one machine to the other. As a consequence, only an approximate running time can be given.

The running time of the trajectory program is a function of a number of factors, some of which are

(a) Number of sources and doublets required to represent the bodies

(b) Number of constant u-velocity panels on wing, pylon, and fuselage

(c) Number of thickness panels on wing and pylon

(d) Number of body segments and tail fin control points used in force and moment calculation

(e) Integration interval

(f) Real time duration of trajectory

All of the above should be kept to a minimum except (e) which should be as large as possible. For the sample trajectory which was presented, the running time was about 5 minutes on the CDC 6600.

TABLE I

SUBROUTINES USED IN COMPUTER PROGRAM

Subroutine Name	Function
ADAMS	numerical integration routine to integrate differential equations
BDYGEN	calculates line sources and doublets to give a required body shape and angle of attack
BLYOUT	lays out fuselage constant u-velocity panels
CEL1	calculates complete elliptic integral of the first kind
CEL2	calculates complete elliptic integral of the second kind
DIRCOS	calculates direction cosines between inertial and store body coordinate systems
DOUBLT	calculates the strength of a linear line doublet
DPCOEF	calculates coefficient matrix of the set of equations to be solved for the wing-pylon-fuselage constant u-velocity panel strengths
DPRHS	calculates right-hand side of the set of equations to be solved for the wing-pylon-fuselage constant u-velocity panel strengths
ELI1	calculates generalized elliptic integral of the first kind
ELI2	calculates generalized elliptic integral of the second kind
FLSQFY	calculates least squares polynomial approximation of specified degree to a given set of data points
FUSEIO	reads and prints fuselage data and calls BDYGEN to calculate line source and doublet distributions
INTOST	transforms a vector with components in the inertial coordinate system to one with component in the store body coordinate system
INVERS	solves a system of simultaneous linear algebraic equations
NUMACH	calculates the local Mach number to be used in calculating the wing-fuselage induced velocities, calculates these velocities and calls FLSQFY to perform a least squares smoothing

TABLE I.- (CONT.)

PLYOUT	reads and prints pylon data and lays out constant u-velocity panels on pylon
PYTHIN	reads pylon thickness data
RESVEL	calculates velocities induced at a field point by all aircraft components
SEMFOR	calculates store empennage forces and moments
SEMPIN	initializes for empennage force and moment calculation
SFORCE	calculates the store-body forces and moments
SHAPE	calculates radius and surface slope at a point on a body from input polynomials
SIMSON	performs a Simpson rule integration
SOURCE	calculates the strength of a linear line source
SOUTPT	prints forces, moments, load distributions, and trajectory data at the end of each integration step
STORIO	reads and prints store data and calls BDYGEN to calculate line source and doublet distributions
STTOIN	transforms a vector with components in the store body coordinate system to one with components in the inertial coordinate system
SWNGIN	reads and prints wing constant u-velocity panel data and twist and camber data, if any
THK.YT	lays out wing and pylon thickness panels
THKOUT	prints wing and pylon thickness input data
THRCAL	calculates thrust from input polynomials
VELBD	calculates velocities or influence functions due to fuselage constant u-velocity panels or panel
VELCAL	calculates velocities at a field point due to fuselage or store line sources and doublets
VELO	calculates the influence of a semi-infinite triangle associated with a constant u-velocity panel
VELOTH	calculates the influence of a semi-infinite triangle associated with a thickness source panel
VELPP	calculates velocities or influence functions due to pylon constant u-velocity panels or panel

TABLE I.- (CONC.)

VELPTH	calculates velocities at a field point due to pylon thickness source panels
VELWP	calculates velocities or influence functions due to wing constant u-velocity panels or panel
VELWTH	calculates velocities at a field point due to wing thickness source panels
WTHIN	reads wing thickness data
WLYOUT	lays out constant u-velocity panels on wing

```
      PROGRAM SSTORE(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)                 SPA01  1
C                                                                           SPA01  2
C     PROGRAM TO CALCULATE SIX-DEGREE-OF-FREEDOM STORE TRAJECTORIES         SPA01  3
C           AT SUPERSONIC SPEEDS                                            SPA01  4
C                                                                           SPA01  5
      LOGICAL CENTER                                                        SPA01  6
C                                                                           SPA01  7
      DIMENSION STATEMENTS                                                  SPA01  8
C                                                                           SPA01  9
      DIMENSION CUBF(7,7),XLND(7)                                           SPA01 10
      DIMENSION HEAD(20)                                                    SPA01 11
C                                                                           SPA01 12
C     COMMON STATEMENTS                                                     SPA01 13
C                                                                           SPA01 14
      COMMON FVN(200,201)                                                   SPA01 15
      COMMON/BPGEOM/                    BNT2(100),CST2(100),ZLC(200),       SPA01 16
     1    ZRC(200),TMTI(100),SANGM,CANGM                                    SPA01 17
      COMMON/CAMBER/ALPHAL(200)                                             SPA01 18
      COMMON /CFORCE/ CDC,CLMRY,CLMCP,CLMSB,CLNBY,CLNCF,CLNSB,CNBY,CNCF,    SPA01 19
     1CNSB,CNX(40),CYBY,CYCF,CYSB,CYX(40),DC(3,3),DELX,DX,FDRDX(81),        SPA01 20
     2ERAD(81),ESTRMX,UT(81),VAR(12),VSTORE,VT(81),WT(81),XMOM              SPA01 21
      COMMON /CONFIG/NFU,NPY,NSTHS,LYB=P                                    SPA01 22
      COMMON /CONSTB/PI,PI2,DTOR,WTUD,FOURPI                                SPA01 25
      COMMON /COUTPT/ DVAR(12),ESTLGC,EXBT(81),TIME                         SPA01 24
      COMMON /EFORCE/ CNBM,CLMEM,CYBM,CLNBM,CLLBM                           SPA01 25
      COMMON /EMPDAT/ FINSS,RADAV,STAIL,PHIROL,MBF,IPLNR,CLALPH             SPA01 26
      COMMON/EXVEL/UEI(200),VEI(200),WEI(200),CIR(200)                      SPA01 27
      COMMON /FLOW/ALFACR,GAMF,FMACH,RHO,VINF,BETA,BETASQ,FMCHSQ            SPA01 28
      COMMON /FBGEOM/FRMAX,SSPAN,    FLTHC,RUDTPL                           SPA01 29
      COMMON/ICVEL/UP,VP,WP,II,IF,       DELTP(200)                         SPA01 30
      COMMON /IFORCE/ NDAMP,NBJSTR,NEMP,NGAM,NMBEG,NMBEGU,NROLL             SPA01 31
      COMMON/INDEX/NCW,MSW,MSWP,NPANLS,NCWH,NBDCR1,NBDCR2,NBD,NWIP,MP,      SPA01 32
     1    NCP,MSP,N1P,N2,N2P,NPTOT                                          SPA01 33
      COMMON /OUTIN1/ XNOSEI,YNUSEI,ZNOSEI,XCGI,YCGI,ZCGI,XBASEI,YBASEI,    SPA01 34
     1ZBASEI                                                                SPA01 35
      COMMON /PYGEOM/Z(20),XPLE,YPL,CHP,WP,PSIPLE,IP,BLLE,BLTE,PSLPDF,      SPA01 36
     1       CENTER                                                         SPA01 37
      COMMON /SMPDAT/NSPOLY,FXEND(9),FCUBF(7,7)                             SPA01 38
      COMMON /SBOB/BXL(101,1),BSS(100,1),BDS(100,1),NSBOR(1)                SPA01 39
      COMMON /STGEOM/BRMAX(1),BLTHC(1),XMBOC(1),YMBO(1),ZMBO(1),            SPA01 40
     1    NBPOLY(1),BXEND(7,1),BCUBF(7,7,1),XBBO(1),YBBO(1),ZBBO(1),        SPA01 41
     2    BIBCR(1),NUMBTR(1),BBIBCR(1),CBIBCR(1)                            SPA01 42
      COMMON /THKDAT/NCHB,NCPB,MB,MPB  ,NTHP,XRBT(400),XRBT(400),           SPA01 43
     1    XLBT(400),XLBT(400),YRCT(400),YLCT(400),ZRCT(400),ZLCT(400),      SPA01 44
     2    THETAL(400),THETRL(200),SLLET(400),SLTET(400),DZDX(400)           SPA01 45
      COMMON /VELARG/X,DUMY,DUMZ,U,V,W,PM,TLRNC,TIPY,PYPNL                  SPA01 46
      COMMON /WGEOM/XBWUC,ZBWU,CHW,SLPWLE,PSIWTE,PSIWLE(20),PSINTE(20),      SPA01 47
     1  Y(20)                                                               SPA01 48
      COMMON/WPGEOM/XRF(200),XRB(200),XLF(200),XLB(200),YRC(200),           SPA01 49
     1    YLC(200),XCPT(200),YCPT(200),ZCPT(200),BWPPLE(200),BWPPTE(200)    SPA01 50
      COMMON/WUFLOW/FNUMCH,BETANU,INUMCH,IBMTH,XTS,XTB                      SPA01 51
      COMMON /THRUBT/ NTPOLY,TEND(5),TC(9,6),FTHRUS,NTHRUS                  SPA01 52
C                                                                           SPA01 53
C     FORMAT STATEMENTS                                                     SPA01 54
C                                                                           SPA01 55
  701 FORMAT(10I5)                                                          SPA01 56
  702 FORMAT(1H1,36X,51HSUPERSONIC SIX-DEGREE-OF-FREEDOM TRAJECTORY PROG    SPA01 57
     1RAM//)                                                                SPA01 58
  703 FORMAT(20A4)                                                          SPA01 59
  704 FORMAT(10X,20A4)                                                      SPA01 60
  705 FORMAT(/1H )                                                          SPA01 61
  706 FORMAT(6F10.0)                                                        SPA01 62
  707 FORMAT(10X,26HAIRCRAFT FLIGHT CONDITIONS/15X,17HANGLE OF ATTACK =,   SPA01 63
     1F6.2,4H DEGREES/15X,19HFLIGHT PATH ANGLE =,F6.2,8H DEGREES/15X,11HS  SPA01 64
     2MACH NUMBER =,F5.2/15X,26HFREE STREAM MASS DENSITY =,F11.7,21H SLUG  SPA01 65
     3GS PER CUBIC FOOT/15X,22HFREE STREAM VELOCITY =,F8.2,16H FEET PER     SPA01 66
     4SECOND//)                                                             SPA01 67
  709 FORMAT(/20X,33HTHIS STORE HAS A PLANAR EMPENNAGE)                     SPA01 68
  710 FORMAT(/20X,36HTHIS STORE HAS A CRUCIFORM EMPENNAGE)                  SPA01 69
  711 FORMAT(20X,20HTHE EMPENNAGE FORCES ACT,F9.5,17H FEET BEHIND NOSE/     SPA01 70
     120X,50HTHE AVERAGE BODY RADIUS IN THE EMPENNAGE REGION IS,F9.5,5H     SPA01 71
     2FEET)                                                                 SPA01 72
  712 FORMAT(1H1,10X,15HREING INPUT DATA)                                   SPA01 73
  713 FORMAT(/15X,60HLOCATION OF WING ROOT CHORD LEADING EDGE RELATIVE TO   SPA01 74
     1O FUSELAGE NOSE/25X,4HXF =,F10.5,5H FEET/25X,4HZF =,F10.5,5H FEET)    SPA01 75
  714 FORMAT(20X,26HAXIAL-FORCE COEFFICIENT IS,F10.5/20X,29HCROSS-FLOW-DR   SPA01 76
     1AG COEFFICIENT IS,F10.5)                                              SPA01 77
  715 FORMAT(25X,51HCOEFFICIENTS OF POLYNOMIALS DESCRIBING EACH SECTION/    SPA01 78
     120X,7HSECTION,5X,2HC1,8X,2HC2,8X,2HC3,8X,2HC4,8X,2HC5,8X,2HC6,8X,     SPA01 79
     222HC7)                                                                SPA01 80
  716 FORMAT(1H1,10X,12HSTORE NUMBER,I5,21H IS THE STORE EJECTED/)          SPA01 81
  717 FORMAT(15X,31HADDITIONAL INPUT FOR THIS STORE/20X,12HSTORE MASS =,    SPA01 82
     1F10.5,6H SLUGS/20X,45HMOMENTS AND PRODUCTS OF INERTIA, SLUG - SQ FS   SPA01 83
     2T/25X,5HIXX =,F10.5/25X,5HIYY =,F10.5/25X,5HIZZ =,F10.5/25X,5HIYZ     SPA01 84
     3=,F10.5/25X,5HIXZ =,F10.5/25X,5HIXY =,F10.5)                          SPA01 85
  718 FORMAT(20X,22HSTORE MOMENT CENTER IS,F9.5,17H FEET BEHIND NOSE/20X    SPA01 86
     1,55HSTORE CENTER OF GRAVITY OFFSET FROM MOMENT CENTER, FEET/25X,6H    SPA01 87
     2XBAR =,F9.5/25X,6HYBAR =,F9.5/25X,6HZBAR =,F9.5)                      SPA01 88
  719 FORMAT(20X,34HPOLYNOMIALS SPECIFYING STORE SHAPE/                     SPA01 89
     123X,26HX/L OF END OF EACH SECTION/26X,7HSECTION,5X,3HX/L)             SPA01 90
  720 FORMAT(28X,I2,5X,7F10.5)                                              SPA01 91
  722 FORMAT(20X,18HSEPARATION ASSUMED,F10.5,15H FEET FROM NOSE)            SPA01 92
  729 FORMAT(20X,52HTHE TAIL FIN SEMISPAN MEASURED FROM THE BODY AXIS IS    SPA01 93
     1,F9.5,5H FEET/20X,29HTHE FINS ARE INITIALLY ROLLED,F6.2,41H DEGREES   SPA01 94
     25 FROM THE VERTICAL AND HORIZONTAL)                                   SPA01 95
  735 FORMAT(20X,27HTHE FIN LIFT-CURVE SLOPE IS,F9.5,11H PER RADIAN)        SPA01 96
  740 FORMAT(1H1,45X,  43HU-VELOCITY PANEL CONTROL POINT COORDINATES,/      SPA01 97
     1 43X,48HINTERFERENCE VELOCITIES INDUCED AT THESE POINTS,/             SPA01 98
     2 45X,45HINPUT T-1ST AND CAMBER SLOPE AT THESE POINTS,/                SPA01 99
     3 52X,31HAND PANEL SINGULARITY STRENGTHS/                              SPA01100
     4 55X,24H(-ING COORDINATE SYSTEM))                                     SPA01101
  741 FORMAT(/10X,19H-ING CONTROL POINTS/17X,3HRUN,2X,5HPANEL,5X,5HX, FTS   SPA01102
     1,6X,5HY, FT,6X,5HZ, FT,6X,6HU/VINF,5X,6HV/VINF,5X,6HW/VINF,5X,       SPA01103
     26HALPHAL,3X,10HUPLUS/VINF)                                            SPA01104
  742 FORMAT(15X,215,3X,8F11.5)                                             SPA01105
  743 FORMAT(/10X,20HPYLON CONTROL POINTS/17X,3HRUN,2X,5HPANEL,5X,5HX, FS   SPA01106
     1T,6X,5HY, FT,6X,5HZ, FT,17X,6HV/VINF,25X,10HUPLUS/VINF)               SPA01107
  744 FORMAT(15X,215,3X,3F11.5,11X,F11.5,22X,F11.5)                         SPA01108
  745 FORMAT(/10X,23HFUSELAGE CONTROL POINTS/16X,4HRING,2X,5HPANEL,5X,      SPA01109
     15HX, FT,6X,5HY, FT,6X,5HZ, FT,17X,6HV/VINF,5X,6HW/VINF,14X,          SPA01110
     210HUPLUS/VINF)                                                        SPA01111
  746 FORMAT(15X,215,3X,3F11.5,11X,2F11.5,11X,F11.5)                        SPA01112
  750 FORMAT(/20X,46HPOLYNOMIALS SPECIFYING STORE THRUST TIME HISTORY/      SPA01113
     123X,30HTIME AT END OF EACH POLYNOMIAL/20X,21HPOLYNOMIAL  TIME, SEC    SPA01114
     2)                                                                     SPA01115
  751 FORMAT(25X,I6,4X,F11.5)                                               SPA01116
  752 FORMAT(23X,27HCOEFFICIENTS OF POLYNOMIALS/26X,10HPOLYNOMIAL,7X,2HC5   SPA01117
     11,9X,2HC2,9X,2HC3,9X,2HC4,9X,2HC5,9X,2HC6)                            SPA01118
  753 FORMAT(25X,I6,6X,6F11.3)                                              SPA01119
  754 FORMAT(///5X,20H****** NSTRS INPUT 48,13,30H, PROGRAM LIMITED TO UNS  SPA01120
     1E STORE)                                                              SPA01121
C                                                                           SPA01122
      DATA PI/3.1415926535/,PI2/1.5707963267/,DTOR/.0174532925/             SPA01123
      DATA RTOD/57.29577951/,FOURPI/12.56637062/                            SPA01124
      DATA ACCG/32.174/                                                     SPA01125
C                                                                           SPA01126
 1000 READ (5,701) NCARDS                                                   SPA01127
      IF (EOF(5)) 2000,2001                                                 SPA01128
 2000 STOP                                                                  SPA01129
 2001 CONTINUE                                                              SPA01130
      WRITE(6,702)                                                          SPA01131
      DO 1 JN1,NCARDS                                                       SPA01132
      READ(5,703) HEAD                                                      SPA01133
    1 WRITE(6,704) HEAD                                                     SPA01134
      WRITE(6,705)                                                          SPA01135
C                                                                           SPA01136
C     SPECIFY FLIGHT CONDITIONS                                             SPA01137
C                                                                           SPA01138
      READ(5,706) ALFAC,GAMF,FMACH,RHO,VINF                                 SPA01139
      WRITE(6,707) ALFAC,GAMF,FMACH,RHO,VINF                                SPA01140
      FMCHSQ=FMACH*FMACH                                                    SPA01141
      BETASQ=FMCHSQ-1.0                                                     SPA01142
      BETA=SQRT(BETASQ)                                                     SPA01143
      ALFACR=ALFAC*DTOR                                                     SPA01144
```

(a) Page 1.

Figure 1.- Program listing.

```
C
C     SPECIFY AIRCRAFT COMPONENTS TO BE INCLUDED
C
      READ(5,701) NFU,NPY,NDMY,NSTRS
      IF(NSTRS.LT.2) GO TO 2
      WRITE (6,754) NSTRS
      STOP
C
C     INPUT FUSELAGE DATA IF FUSELAGE IS PRESENT
C
    2 NBIP=0
      IF(NFU.EQ.0) GO TO 10
      CALL FUSEID
   10 CONTINUE
C
C     INPUT WING DATA
C
      READ(5,706) XRMUC,ZRMU
      WRITE(6,712)
      WRITE(6,713) XRMUC,ZRMU
      CALL SWNGIN
      TIPY=SPANY(1)
      TLENC=(TIPY+15E-5)**2
C
C     LAY OUT CONSTANT U-VELOCITY PANELS ON WING
C
      CALL WLYOUT
C
C     INPUT WING THICKNESS DATA
C
      CALL WITHIN
C
C     INPUT PYLON LAYOUT INFORMATION IF PYLON IS PRESENT
C
      NPY=0
      NPS=0
      IF (NPY .EQ. 0) GO TO 20
      CALL PLYOUT
      CENTER=YPL,EQ,0,0
C
C     INPUT PYLON THICKNESS DATA
C
      CALL PYTHIN
   20 CONTINUE
      NTMP=NS+NPS
C
C     OUTPUT THICKNESS DISTRIBUTION
C
      CALL THKOUT
C
C     LAY OUT THICKNESS PANELS ON WING AND PYLON IF PRESENT
C
      CALL THKLYT
C
C     LAY OUT CONSTANT U-VELOCITY PANELS ON FUSELAGE IF PRESENT
C
      N1P=NPANL$+1
      N2=NPANL$+NP
      N2P=N2+1
      IF(NFU.NE.0) CALL FLYOUT
      NPTOT=N2+NBIP
      NPTOTP=NPTOT+1
C
C     INPUT STORE DATA IF PRESENT
C
      IF (NSTRS .EQ. 0) GO TO 30
      CALL STORID
C
C     LOCATE STORE IN FUSELAGE COORDINATE SYSTEM
C
      DO 35 N=1,NSTRS
```

```
      SSTMCR(N)=SIN(STMCR(N))                                          SPA01217
      CSTMCR(N)=COS(STMCR(N))                                          SPA01218
      XHSU(N)=XRMUC+X=SUC(N)                                           SPA01219
      YHSU(N)=Y=SU(N)                                                  SPA01220
   35 ZHSU(N)=ZRMU+ZHSU(N)                                             SPA01221
   30 CONTINUE                                                         SPA01222
C                                                                      SPA01223
C     SET UP COEFFICIENT MATRIX AND RIGHT HAND SIDE OF EQUATIONS,      SPA01224
C        ARRANGE IN FVN ARRAY IN AUGMENTED MATRIX FORM                 SPA01225
C                                                                      SPA01226
      INUMC=0                                                          SPA01227
      CALL DPCOEF                                                      SPA01228
      CALL DPRHS                                                       SPA01229
      DO 40 I=1,NPTOT                                                  SPA01230
      FVN(I,NPTOTP)=CIR(I)                                             SPA01231
   40 CONTINUE                                                         SPA01232
C                                                                      SPA01233
C     SOLVE LINEAR SYSTEM                                              SPA01234
C                                                                      SPA01235
      CALL INVERS(FVN,1,NPTOT,200,201)                                 SPA01236
      DO 43 J=1,NPTOT                                                  SPA01237
      DELTP(J)=FVN(J,NPTOTP)                                           SPA01238
   43 CONTINUE                                                         SPA01239
C                                                                      SPA01240
C     OUTPUT CONTROL POINT COORDINATES, INTERFERENCE VELOCITIES, INPUT  SPA01241
C          TWIST AND CAMBER ANGLES, AND U-VELOCITY PANEL SINGULARITY    SPA01242
C          STRENGTHS                                                   SPA01243
C                                                                      SPA01244
C     WING PANELS                                                      SPA01245
C                                                                      SPA01246
      WRITE (6,740)                                                    SPA01247
      WRITE (6,741)                                                    SPA01248
      DO 100 J=1,NPANLS                                                SPA01249
      NRO=(J-1)/NCP+1                                                  SPA01250
      NPAN=J-NCP*(NRO-1)                                               SPA01251
      UPVINF=PI*DELTP(J)                                               SPA01252
  100 WRITE (6,742) NRO,NPAN,XCPT(J),YCPT(J),ZCPT(J),TWIST(J),VFI(J),  SPA01253
     1WEI(J),ALPHAL(J),UPVINF                                          SPA01254
C                                                                      SPA01255
C     PYLON PANELS                                                     SPA01256
C                                                                      SPA01257
      IF (NPY.EQ.0) GO TO 110                                          SPA01258
      WRITE (6,745)                                                    SPA01259
      DO 101 J=N1P,N2                                                  SPA01260
      NRO=(J-NPANLS-1)/NCP+1                                           SPA01261
      NPAN=J-NPANLS-NCP*(NRO-1)                                        SPA01262
      UPVINF=PI*DELTP(J)                                               SPA01263
  101 WRITE (6,746) NRO,NPAN,XCPT(J),YCPT(J),ZCPT(J),VFI(J),UPVINF     SPA01264
C                                                                      SPA01265
C     FUSELAGE PANELS                                                  SPA01266
C                                                                      SPA01267
  110 IF (NFU.EQ.0) GO TO 120                                          SPA01268
      WRITE (6,705)                                                    SPA01269
      DO 102 J=N2P,NPTOT                                               SPA01270
      NRO=(J-N2-1)/NBO+1                                               SPA01271
      NPAN=J-N2-NBO*(NRO-1)                                            SPA01272
      UPVINF=PI*DELTP(J)                                               SPA01273
  102 WRITE (6,746) NRO,NPAN,XCPT(J),YCPT(J),ZCPT(J),VFI(J),WEI(J),    SPA01274
     1 UPVINF                                                          SPA01275
  120 CONTINUE                                                         SPA01276
C                                                                      SPA01277
C     INPUT ADDITIONAL INFORMATION REQUIRED TO DESCRIBE EJECTED STORE  SPA01278
C                                                                      SPA01279
      IF (NSTRS.EQ.0) GO TO 1000                                       SPA01280
      READ (5,701) NEJECT,NSEG,NSEGAD,NGAM,NPOLY,NROLL,NEMP,NDAMP,NTMPJSPA01281
      READ (5,706) SMASS,FIXX,FIYY,FIZZ,FIYZ,FIXZ,FIXY                SPA01282
      READ (5,706) XMOM,XRAM,YRAM,ZRAM                                 SPA01283
      WRITE (6,716) NEJECT                                             SPA01284
      WRITE (6,717) SMASS,FIXX,FIYY,FIZZ,FIYZ,FIXZ,FIXY               SPA01285
      WRITE (6,718) XMOM,XRAM,YRAM,ZRAM                                SPA01286
      XMOM=XMB(XMOM)                                                   SPA01287
      READ (5,706) (XEND(J),J=1,NPOLY)                                 SPA01288
```

(b) Page 2.

Figure 1.- Continued.

```
      DO 50 J=1,NPOLY                                                   SPA01284
   50 READ (5,706) (COEF(J,K),K=1,7)                                    SPA01285
      WRITE (6,719)                                                     SPA01286
      DO 51 J=1,NPOLY                                                   SPA01287
   51 WRITE (6,720) J,XEND(J)                                           SPA01288
      WRITE (6,715)                                                     SPA01289
      DO 52 J=1,NPOLY                                                   SPA01290
   52 WRITE (6,720) J,(COEF(J,K),K=1,7)                                 SPA01291
C                                                                       SPA01292
C     DETERMINE GEOMETRIC PARAMETERS DESCRIBING STORE                   SPA01293
C                                                                       SPA01294
      DO 53 J=1,NSTMS                                                   SPA01295
      IF (NEJPLT-NUMSTM(J)) 53,54,53                                    SPA01296
   54 NEJSTR=J                                                          SPA01297
   53 CONTINUE                                                          SPA01298
      ESTLGC=BLTMC(NEJSTR)                                              SPA01299
      ESTMX=BMMAX(NEJSTR)                                               SPA01300
      SREF=PI*EBTRMX**2                                                 SPA01301
      REFL=2.0*EBTRMX                                                   SPA01302
      RSEG=NSEG                                                         SPA01303
      DELX=ESTLGC/RSEG                                                  SPA01304
      DAMREL=.2                                                         SPA01305
      EXST(1)=0.0                                                       SPA01306
      NMSEG=2*NSEG+1                                                    SPA01307
      DO 55 J=2,NMBEG                                                   SPA01308
   55 EXST(J)=EXST(J-1)+DX                                              SPA01309
      DO 56 J=2,NMBEG,2                                                 SPA01310
      XXX=EXST(J)/ESTLGC                                                SPA01311
      CALL SHAPE (XXX,NPOLY,XEND,COEF,RR,RDRDX(J))                      SPA01312
   56 ERAD(J)=ESTLGC*RR                                                 SPA01313
      RSEGX0=NSEGX0                                                     SPA01314
      XBEP=RSEGX0*DELX                                                  SPA01315
      WRITE(6,722) XBEP                                                 SPA01316
      READ (5,706) CA,CDC                                               SPA01317
      WRITE(6,716) CA,CDC                                               SPA01318
      NMSEG=2*NSEGX0+1                                                  SPA01319
      NASYM=0                                                           SPA01320
      IF (ABS(XBAR),GT,1.0E-05) NASYM=1                                 SPA01321
      IF (ABS(YBAR),GT,1.0E-05) NASYM=1                                 SPA01322
      IF (ABS(ZBAR),GT,1.0E-05) NASYM=1                                 SPA01323
      SAPG=SIN(ALFACR+GAMP+DTOR)                                        SPA01324
      CAPG=COS(ALFACR+GAMP+DTOR)                                        SPA01325
C                                                                       SPA01326
C     INPUT EMPENNAGE DATA IF EMPENNAGE IS PRESENT                      SPA01327
C                                                                       SPA01328
      CNEM=0.0                                                          SPA01329
      CLMEM=0.0                                                         SPA01330
      CYEM=0.0                                                          SPA01331
      CLNEM=0.0                                                         SPA01332
      CLLEM=0.0                                                         SPA01333
      IF (NEMP,EQ,0) GO TO 130                                          SPA01334
      READ (5,701) IPLNR,MSF                                            SPA01335
      READ (5,706) XTAIL,RADAV,FINBB,PHIROL,CLALPH                      SPA01336
      IF (IPLNR,EQ,1) WRITE (6,709)                                     SPA01337
      IF (IPLNR,EQ,0) WRITE (6,710)                                     SPA01338
      WRITE (6,711) XTAIL,RADAV                                         SPA01339
      WRITE (6,729) FINBB,PHIROL                                        SPA01340
      WRITE (6,735) CLALPH                                              SPA01341
      CALL SEMPIN                                                       SPA01342
C                                                                       SPA01343
C     INPUT THRUST TIME HISTORY IF POWERED STORE                        SPA01344
C                                                                       SPA01345
  130 IF (ATMRUN,EQ,0) GO TO 57                                         SPA01346
      READ (5,701) NTPOLY                                               SPA01347
      READ (5,706) (TEND(J),J=1,NTPOLY)                                 SPA01348
      DO 131 J=1,NTPOLY                                                 SPA01349
  131 READ (5,706) (TC(J,K),K=1,6)                                      SPA01350
      WRITE (6,750)                                                     SPA01351
      DO 132 J=1,NTPOLY                                                 SPA01352
  132 WRITE (6,751) J,TEND(J)                                           SPA01353
      WRITE (6,752)                                                     SPA01354
      DO 133 J=1,NTPOLY                                                 SPA01355
  133 WRITE (6,753) J,(TC(J,K),K=1,6)                                   SPA01356
C                                                                       SPA01357
C     INITIALIZE FOR TRAJECTORY CALCULATION                             SPA01358
C                                                                       SPA01359
   57 VAR(11)=SIB(NEJSTR)                                               SPA01360
      VAR(10)=0.0                                                       SPA01361
      VAR(12)=0.0                                                       SPA01362
      READ (5,706) VXZERO,VYZERO,VZZERO,                                SPA01363
     1             VAR(4),VAR(5),VAR(6)                                 SPA01364
      VAR(1)=VZZERO*SSIBCB(NEJSTR)+VXZERO*CSIBCB(NEJSTR)                SPA01365
      VAR(2)=VYZERO                                                     SPA01366
      VAR(3)=VZZERO*CSIBCB(NEJSTR)-VXZERO*SSIBCB(NEJSTR)                SPA01367
      CONTINUE                                                          SPA01368
      VAR(7)=XHSO(NEJSTR)+XMOM*CSIBCB(NEJSTR)                           SPA01369
      VAR(8)=YHSO(NEJSTR)                                               SPA01370
      VAR(9)=ZHSO(NEJSTR)+XMOM*SSIBCB(NEJSTR)                           SPA01371
      XNOSEI=XHSO(NEJSTR)                                               SPA01372
      YNOSEI=YHSO(NEJSTR)                                               SPA01373
      ZNOSEI=ZHSO(NEJSTR)                                               SPA01374
      XCGI=VAR(7)                                                       SPA01375
      YCGI=VAR(8)                                                       SPA01376
      ZCGI=VAR(9)                                                       SPA01377
      XBASEI=XNOSEI+BLTMC(NEJSTR)+CSIBCB(NEJSTR)                        SPA01378
      YBASEI=YNOSEI                                                     SPA01379
      ZBASEI=ZNOSEI+BLTMC(NEJSTR)+SSIBCB(NEJSTR)                        SPA01380
      READ (5,706) DTIME,TIMEI,TIMEF                                    SPA01381
      NEQ=12                                                            SPA01382
      DDTIME=DTIME                                                      SPA01383
      TIME=TIMEI                                                        SPA01384
      IF (TIMF,LE,0,0) GO TO 60                                         SPA01385
      READ (5,706) VAR                                                  SPA01386
      DO 61 J=10,12                                                     SPA01387
   61 VAR(J)=DTOR*VAR(J)                                                SPA01388
   60 NDIFEQ=3                                                          SPA01389
      CALL ADAMS (DTIME,DDTIME, VAR,NVAR,NEQ,NDIFEQ,TIME)               SPA01390
      NOUT=1                                                            SPA01391
C                                                                       SPA01392
C     CALCULATE AERODYNAMIC FORCES AND MOMENTS                          SPA01393
C                                                                       SPA01394
   62 CALL SFORCE                                                       SPA01395
      IF (NEMP,EQ,1) CALL SEMFOR                                        SPA01396
      IF (NGAM,EQ,1) CALL DIRCOS(VAR,DC)                                SPA01397
C                                                                       SPA01398
C     CALCULATE ACCELERATIONS                                           SPA01399
C                                                                       SPA01400
C     CALCULATE COEFFICIENT MATRIX                                      SPA01401
C                                                                       SPA01402
      DO 70 J=1,6                                                       SPA01403
      DO 70 K=1,6                                                       SPA01404
   70 FVN(J,K)=0.0                                                      SPA01405
      FVN(1,1)=1.0                                                      SPA01406
      FVN(2,2)=1.0                                                      SPA01407
      FVN(3,3)=1.0                                                      SPA01408
      FVN(4,4)=FIXX                                                     SPA01409
      FVN(5,5)=FIYY                                                     SPA01410
      FVN(6,6)=FIZZ                                                     SPA01411
      FVN(4,5)=-FIXY                                                    SPA01412
      FVN(4,6)=-FIXZ                                                    SPA01413
      FVN(5,6)=-FIYZ                                                    SPA01414
      FVN(5,4)=-FIXY                                                    SPA01415
      FVN(6,4)=-FIXZ                                                    SPA01416
      FVN(6,5)=-FIYZ                                                    SPA01417
      IF (NASYM,EQ,0) GO TO 80                                          SPA01418
      FVN(1,4)=YBAR*DC(1,3)-ZBAR*DC(1,2)                                SPA01419
      FVN(1,5)=ZBAR*DC(1,1)-XBAR*DC(1,3)                                SPA01420
      FVN(1,6)=XBAR*DC(1,2)-YBAR*DC(1,1)                                SPA01421
      FVN(2,4)=YBAR*DC(2,3)-ZBAR*DC(2,2)                                SPA01422
      FVN(2,5)=ZBAR*DC(2,1)-XBAR*DC(2,3)                                SPA01423
      FVN(2,6)=XBAR*DC(2,2)-YBAR*DC(2,1)                                SPA01424
      FVN(3,4)=YBAR*DC(3,3)-ZBAR*DC(3,2)                                SPA01425
      FVN(3,5)=ZBAR*DC(3,1)-XBAR*DC(3,3)                                SPA01426
      FVN(3,6)=XBAR*DC(3,2)-YBAR*DC(3,1)                                SPA01427
```

(c) Page 3.

Figure 1.- Continued.

```
      DO 71 J=4,6                                                       SPA01433
      DO 71 K=1,3                                                       SPA01434
   71 FVN(J,K)=SMASS*FVN(K,J)                                           SPA01435
C                                                                       SPA01436
C     CALCULATE RIGHT HAND SIDE                                         SPA01437
C                                                                       SPA01438
   80 GXX=-GCG*(SAMG*DC(1,1)-CAMG*DC(3,1))                              SPA01439
      GYY=-GCG*(SAMG*DC(1,2)-CAMG*DC(3,2))                              SPA01440
      GZZ=-GCG*(SAMG*DC(1,3)-CAMG*DC(3,3))                              SPA01441
      QSTORE=0.5*RHO*VBTOMF**2                                          SPA01442
      QBREF=QSTORE*SREF                                                 SPA01443
      QSREFL=QBREF*REFL                                                 SPA01444
      CNORM=CNBB+CNBY+CNCF                                              SPA01445
      CSIDE=CYBB+CYBY+CYCF                                              SPA01446
      CROLL=0.0                                                         SPA01447
      CPITCH=CLMBB+CLMBY+CLMCF                                          SPA01448
      CYA=CLNBB+CLNBY+CLNCF                                             SPA01449
      IF (NEMP.EQ.0) GO TO 81                                           SPA01450
      CNORM=CNORM*CNEM                                                  SPA01451
      CSIDE=CSIDE+CYEM                                                  SPA01452
      CROLL=CROLL+CLLEM                                                 SPA01453
      CPITCH=CPITCH+CLMEM                                               SPA01454
      CYA=CYA+CLNEM                                                     SPA01455
   81 CONTINUE                                                          SPA01456
      CALL INTOST(VAR(1),VAR(2),VAR(3),UU,VV,WW,DC)                     SPA01457
      RTHRUS=0.0                                                        SPA01458
      IF (NTHRUS.NE.0) CALL THRCAL(TIME)                                SPA01459
      RONE=GXX*QBREF*CA/SMASS+VAR(6)*VV-VAR(5)*WW                       SPA01460
      RUNE=RONE+RTHRUS/SMASS                                            SPA01461
      RTHR=GYY*QBREF+CBIDE/SMASS+VAR(4)*WW-VAR(6)*UU                    SPA01462
      RTHRR=GZZ-QBREF*CNORM/SMASS+VAR(5)*UU-VAR(4)*VV                   SPA01463
      FVN(4,7)=QBREFL*CROLL                                             SPA01464
      FVN(5,7)=QBREFL*CPITCH                                            SPA01465
      FVN(6,7)=QBREFL*CYA                                               SPA01466
      IF (NASYM.EW.0) GO TO 85                                          SPA01467
      RONE=RONE+XBAR*(VAR(5)**2+VAR(6)**2)  BAR*VAR(4)*VAR(5)-ZBAR*VAR(4SPA01468
     1)*VAR(6)                                                          SPA01469
      RTHR=RTHR-XBAR*VAR(4)*VAR(5)+YBAR*(VAR(4)**2+VAR(6)**2)-ZBAR*VAR(5SPA01470
     1)*VAR(6)                                                          SPA01471
      RTHRR=RTHR+XBAR*VAR(4)*VAR(6)+YBAR*VAR(5)*VAR(6)+ZBAR*(VAR(4)**2+VASPA01472
     1R(5)**2)                                                          SPA01473
      FVN(4,7)=FVN(4,7)+SMASS*(YBAR*(VAR(6)*VV-VAR(5)*UU)-ZBAR*(VAR(6)* SPA01474
     1UU-VAR(4)*WW))                                                    SPA01475
      FVN(5,7)=FVN(5,7)-SMASS*(ZBAR*(VAR(5)*WW-VAR(6)*VV)-XBAR*(VAR(4)* SPA01476
     1VV-VAR(5)*UU))                                                    SPA01477
      FVN(6,7)=FVN(6,7)-SMASS*(XBAR*(VAR(6)*UU-VAR(4)*WW)-YBAR*(VAR(5)* SPA01478
     1WW-VAR(6)*VV))                                                    SPA01479
      FVN(4,7)=FVN(4,7)+SMASS*(GZZ*YBAR-GYY*ZBAR)                       SPA01480
      FVN(5,7)=FVN(5,7)+SMASS*(GXX*ZBAR-GZZ*XBAR)                       SPA01481
      FVN(6,7)=FVN(6,7)+SMASS*(GYY*XBAR-GXX*YBAR)                       SPA01482
   85 FVN(1,7)=RONE*DC(1,1)+RTHR*DC(2,1)+RTHRR*DC(3,1)                  SPA01483
      FVN(2,7)=RONE*DC(2,1)+RTHR*DC(2,2)+RTHRR*DC(3,2)                  SPA01484
      FVN(3,7)=RONE*DC(3,1)+RTHR*DC(3,2)+RTHRR*DC(3,3)                  SPA01485
      FVN(4,7)=FVN(4,7)+VAR(6)*VAR(5)*(FIZZ-FIYY)+(VAM(5)**2-VAM(6)**2)*SPA01486
     1FIYZ+VAM(4)*(VAR(5)*FIXZ-VAR(6)*FIXY)                            SPA01487
      FVN(5,7)=FVN(5,7)+VAR(6)*VAR(4)*(FIXX-FIZZ)+(VAR(6)**2-VAR(4)**2)*SPA01488
     1FIXZ+VAR(5)*(VAR(6)*FIXY-VAR(4)*FIYZ)                            SPA01489
      FVN(6,7)=FVN(6,7)+VAR(4)*VAR(5)*(FIYY-FIXX)+(VAR(4)**2-VAR(5)**2)*SPA01490
     1FIXY+VAR(6)*(VAR(4)*FIYZ-VAR(5)*FIXZ)                            SPA01491
C                                                                       SPA01492
C     SOLVE FOR ACCELERATIONS                                           SPA01493
C                                                                       SPA01494
      CALL INVERS(FVN,1,6,200,201)                                      SPA01495
      DO 86 J=1,6                                                       SPA01496
   86 DVAM(J)=FVN(J,7)                                                  SPA01497
      DO 87 J=7,9                                                       SPA01498
   87 DVAM(J)=VAR(J-6)                                                  SPA01499
      SPHI=SIN(VAR(12))                                                 SPA01500
      CPHI=COS(VAM(12))                                                 SPA01501
      DVAM(10)=(VAM(5)*SPHI+VAM(6)*CPHI)/CUS(VAM(11))                   SPA01502
      DVAM(11)=VAM(5)*CPHI-VAM(6)*SPHI                                  SPA01503
      DVAM(12)=VAM(4)+SIN(VAM(11))*DVAM(10)                             SPA01504
C                                                                       SPA01505
C     OUTPUT TRAJECTORY DATA IF AT END OF INTEGRATION STEP              SPA01506
C                                                                       SPA01507
C                                                                       SPA01508
      IF (NOUT.EQ.0) GO TO 91                                           SPA01509
      CALL SOUTPT                                                       SPA01510
      NOUT=0                                                            SPA01511
      IF (TIMP+1.0E-05*TIMEF) 91,1000,1000                              SPA01512
C                                                                       SPA01513
C     CALL INTEGRATION ROUTINE                                          SPA01514
C                                                                       SPA01515
   91 CALL ADAMS (DTIMP, DDTIME,VAR,DVAM,NEQ,NDIFEQ,TIME)               SPA01516
      IF (NDIFEQ.EQ.1) GO TO 1000                                       SPA01517
      IF (NDIFEQ.GT.7) NOUT=1                                           SPA01518
      GO TO 62                                                          SPA01519
      END
```

```
      SUBROUTINE ADAMS(H,DS,Y,DY,NEQ,NDIFEQ,S)                          SPA02  1
C     ADAMS INTEGRATION ROUTINE                                         SPA02  2
C      FIXED INTEGRATION INTERVAL                                       SPA02  3
C                                                                       SPA02  4
      DIMENSION Y(12),DY(12)                                            SPA02  5
      DIMENSION Y1(12),DY3(12),Y5(12),E(12),Y3(12),DY1(12),Y2(12),PA(12)SPA02  6
     1,PF(12),DY2(12)                                                   SPA02  7
      GO TO (100,200,300,400,500,600,700,800,900),NDIFEQ               SPA02  8
C                                                                       SPA02  9
C     START BY RUNGE-KUTTA                                              SPA02 10
C                                                                       SPA02 11
  100 H=DS                                                              SPA02 12
 1101 JB=1                                                              SPA02 13
      DO 101 I=1,NEQ                                                    SPA02 14
  101 Y1(I)=Y(I)                                                        SPA02 15
  102 SX1=S                                                             SPA02 16
  103 NDIFEQ=2                                                          SPA02 17
      RETURN                                                            SPA02 18
  200 DO 201 I=1,NEQ                                                    SPA02 19
      DY3(I)=DY(I)                                                      SPA02 20
      Y5(I)=Y(I)                                                        SPA02 21
      TEMP=H*DY(I)                                                      SPA02 22
      Y(I)=0.5*TEMP+Y5(I)                                               SPA02 23
  201 E(I)=TEMP                                                         SPA02 24
      S=0.5*H+S                                                         SPA02 25
      NDIFEQ=3                                                          SPA02 26
      RETURN                                                            SPA02 27
  300 DO 301 I=1,NEQ                                                    SPA02 28
      TEMP=H*DY(I)                                                      SPA02 29
      Y(I)=0.5*TEMP+Y5(I)                                               SPA02 30
  301 E(I)=E(I)+2.0*TEMP                                                SPA02 31
      NDIFEQ=4                                                          SPA02 32
      RETURN                                                            SPA02 33
  400 DO 401 I=1,NEQ                                                    SPA02 34
      TEMP=H*DY(I)                                                      SPA02 35
      Y(I)=Y5(I)+TEMP                                                   SPA02 36
  401 E(I)=E(I)+2.0*TEMP                                                SPA02 37
      S=0.5*H+S                                                         SPA02 38
      NDIFEQ=5                                                          SPA02 39
      RETURN                                                            SPA02 40
  500 DO 501 I=1,NEQ                                                    SPA02 41
  501 Y(I)=(H*DY(I)+E(I))*0.16666667+Y5(I)                              SPA02 42
      GO TO (502,507,509,902),JB                                        SPA02 43
  502 CONTINUE                                                          SPA02 44
  507 DO 508 I=1,NEQ                                                    SPA02 45
      DY1(I)=DY3(I)                                                     SPA02 46
```

(d) Page 4.

Figure 1.- Continued.

```
  508 Y2(I)=Y(I)                                                        SPA02 47
      JB=5                                                              SPA02 48
      GO TO 10$                                                         SPA02 49
  509 S=S+H                                                             SPA02 50
C                                                                       SPA02 51
C     OUTPUT OF RUNGE-KUTTA                                             SPA02 52
C                                                                       SPA02 53
      IF (JB-4) 521,802,802                                             SPA02 54
  521 NDIFEQ=8                                                          SPA02 55
      DO 522 I=1,NEQ                                                    SPA02 56
      PX(I)=Y(I)                                                        SPA02 57
      PF(I)=DY(I)                                                       SPA02 58
  522 Y(I)=Y2(I)                                                        SPA02 59
      RETURN                                                            SPA02 60
  800 DO 801 I=1,NEQ                                                    SPA02 61
      Y(I)=PX(I)                                                        SPA02 62
      DY(I)=PF(I)                                                       SPA02 63
  801 DY2(I)=DY3(I)                                                     SPA02 64
      JB=4                                                              SPA02 65
      S=S+H                                                             SPA02 66
  802 NDIFEQ=9                                                          SPA02 67
      RETURN                                                            SPA02 68
  900 IF (JA-5) 103,901,9901                                            SPA02 69
  901 CON=0.0416666667*H                                                SPA02 70
      JB=6                                                              SPA02 71
 9901 NDIFEQ=6                                                          SPA02 72
      GO TO 600                                                         SPA02 73
  902 JB=5                                                              SPA02 74
      GO TO 802                                                         SPA02 75
C                                                                       SPA02 76
C     ADAMS INTEGRATION                                                 SPA02 77
C                                                                       SPA02 78
  600 DO 601 I=1,NEQ                                                    SPA02 79
      Y1(I)=Y(I)                                                        SPA02 80
      Y(I)=Y(I)+CON*(55.0*DY(I)-59.0*DY3(I)+37.0*DY2(I)-9.0*DY1(I))     SPA02 81
      DY1(I)=DY2(I)                                                     SPA02 82
      DY2(I)=DY3(I)                                                     SPA02 83
      DY3(I)=DY(I)                                                      SPA02 84
  601 Y3(I)=Y(I)                                                        SPA02 85
      S=S+H                                                             SPA02 86
      NDIFEQ=7                                                          SPA02 87
      RETURN                                                            SPA02 88
  700 DO 701 I=1,NEQ                                                    SPA02 89
  701 Y(I)=Y1(I)+CON*(9.0*DY(I)+19.0*DY3(I)-5.0*DY2(I)+DY1(I))          SPA02 90
      GO TO 802                                                         SPA02 91
      END                                                               SPA02 92
```

```
      SUBROUTINE BDYGEN(NXBODY,RADIUS,LBODY,NSEG,XEND,SCOEF,T,TC,TX,    SPA03  1
     1 ALPHA)                                                           SPA03  2
C                                                                       SPA03  3
C SUBROUTINE TO CALCULATE SUPERSONIC LINE SOURCES AND DOUBLETS TO GIVE A SPA03  4
C REQUIRED BODY SHAPE AND ANGLE OF ATTACK.                              SPA03  5
C                                                                       SPA03  6
      DIMENSION XBODY(101),RBODY(101),RPBODY(101),DRDX(101),T(100)      SPA03  7
      DIMENSION TC(100),TX(101)                                         SPA03  8
      DIMENSION A(100),XF(100),RF(100)                                  SPA03  9
C                                                                       SPA03 10
      COMMON /FLO/ALFACM,DUM1,MACH,DUM2(2),BETA,BETASQ,MACHSQ           SPA03 11
      COMMON/SHLP/RFIELD,RFIELD,SSQ,X2,      U,V,VT                     SPA03 12
C                                                                       SPA03 13
      REAL MACH,MACHSQ,LBODY                                            SPA03 14
C                                                                       SPA03 15
  700 FORMAT(/// 5X,108H***** AT BASE OF BODY RADIAL DISTANCE TO MACH CUSPA03 16
     1NE EMANATING FROM BODY NOSE IS LESS THAN MAXIMUM BODY RADIUS/13X, SPA03 17
     2STHEREFOR BODY INPUT DATA AND MACH NUMBER)                        SPA03 18
  701 FORMAT(/35X,10HBODY SHAPE,40X,11HSINGULARITY,5X,6HSOURCE,9X,      SPA03 19
     17HDOUBLET/47X,6HORIGIN,7X,9HCONSTANTS,6X,9HCONSTANTS/24X,5HX, FT, SPA03 20
     210X,5HR, FT,10X,5HDR/DX,20X,1HM,7X,6HMACH(N), FT,8X,4HA(N),11X,   SPA03 21
     35HK0(N))                                                          SPA03 22
  702 FORMAT(15X,3(0PF15.5),15X,I5,F15.5,2(1PE15.5))                    SPA03 23
C                                                                       SPA03 24
      IF(BETA*RADIUS.LT.LBODY) GO TO 20                                 SPA03 25
      WRITE (6,700)                                                     SPA03 26
      STOP                                                              SPA03 27
   20 N=NXBODY-1                                                        SPA03 28
      XBODY(1)=0.0                                                      SPA03 29
      XBODY(NXBODY)=LBODY                                               SPA03 30
C                                                                       SPA03 31
C SETUP OF POINTS ON BODY AXIS BY DIVIDING BODY LENGTH INTO N  EQUAL     SPA03 32
C     LENGTH SEGMENTS                                                   SPA03 33
C                                                                       SPA03 34
      DEL=LBODY/N                                                       SPA03 35
      DO 33 I=2,N                                                       SPA03 36
   33 XBODY(I)=XBODY(I-1)+DEL                                           SPA03 37
C                                                                       SPA03 38
C CALCULATION OF RADII AND RPBODY TO GO WITH THE AXIS POINTS            SPA03 39
C                                                                       SPA03 40
      OLBOD=1./LBODY                                                    SPA03 41
      DO 35 I=1,NXBODY                                                  SPA03 42
      XOL=XBODY(I)*OLBOD                                                SPA03 43
      CALL SHAPE(XOL,NSEG,XEND,SCOEF,ROL,RPBODY(I))                     SPA03 44
   35 RBODY(I)=ROL*LBODY                                                SPA03 45
C                                                                       SPA03 46
C  CALCULATION OF DRDX AND RADIUS AT MIDPOINTS BETWEEN AXIS POINTS      SPA03 47
C                                                                       SPA03 48
      DO 36 I=1,N                                                       SPA03 49
      XF(I)=.5*(XBODY(I)+XBODY(I+1))                                    SPA03 50
      XOL=XF(I)*OLBOD                                                   SPA03 51
      CALL SHAPE(XOL,NSEG,XEND,SCOEF,ROL,DRDX(I))                       SPA03 52
   36 RF(I)=ROL*LBODY                                                   SPA03 53
C                                                                       SPA03 54
C THIS LOOP DETERMINES THE LOCATIONS OF THE ORIGINS OF THE CONICAL LINE SPA03 55
C SOURCES AND DOUBLETS.  THE ORIGINS ARE LOCATED AT THE TX(I).          SPA03 56
C                                                                       SPA03 57
      DO 10 I=1,NXBODY                                                  SPA03 58
   10 TX(I)=XBODY(I)-BETA*RBODY(I)                                      SPA03 59
C                                                                       SPA03 60
C ROUTINE FOR DROPPING CONTROL POINTS IF THEY ARE OUTSIDE THE MACH      SPA03 61
C CONE OF THE NOSE.                                                     SPA03 62
C                                                                       SPA03 63
  100 IF(BETA*RF(1).LT.XF(1)) GO TO 199                                 SPA03 64
      N=N-1                                                             SPA03 65
      NXBODY=NXBODY-1                                                   SPA03 66
      DO 101 I=2,NXBODY                                                 SPA03 67
      XBODY(I)=XBODY(I+1)                                               SPA03 68
      RBODY(I)=RBODY(I+1)                                               SPA03 69
      RPBODY(I)=RPBODY(I+1)                                             SPA03 70
  101 TX(I)=TX(I+1)                                                     SPA03 71
      DO 102 I=2,N                                                      SPA03 72
      XF(I)=XF(I+1)                                                     SPA03 73
      RF(I)=RF(I+1)                                                     SPA03 74
  102 DRDX(I)=DRDX(I+1)                                                 SPA03 75
      XF(1)=.5*(XBODY(2)+XBODY(1))                                      SPA03 76
      XOL=XF(1)*OLBOD                                                   SPA03 77
      CALL SHAPE(XOL,NSEG,XEND,SCOEF,ROL,DRDX(1))                       SPA03 78
      RF(1)=ROL*LBODY                                                   SPA03 79
      GO TO 100                                                         SPA03 80
  199 CONTINUE                                                          SPA03 81
C                                                                       SPA03 82
C     DETERMINATION OF SOURCE STRENGTHS AT CONTROL POINTS MIDWAY BETWEENSPA03 83
C      BODY DEFINITION POINTS.                                          SPA03 84
C                                                                       SPA03 85
C CALCULATION OF THE FIRST SOURCE STRENGTH.                             SPA03 86
C                                                                       SPA03 87
      XFIELD=XF(1)                                                      SPA03 88
      RFIELD=RF(1)                                                      SPA03 89
      SLOPE=DRDX(1)                                                     SPA03 90
      CALL SOURCE(TX(1))                                                SPA03 91
      A(1)=V-SLOPE*U                                                    SPA03 92
```

(e) Page 5.

Figure 1.- Continued.

```
      T(I)=RHODX(I)/A(I)                                                SPA03 93
C                                                                       SPA03 94
C CALCULATION OF THE REST OF SOURCE STRENGTHS                           SPA03 95
C                                                                       SPA03 96
      DO 210 I=2,N                                                      SPA03 97
      XFIELD=XF(I)                                                      SPA03 98
      RFIELD=RF(I)                                                      SPA03 99
      SLOPE=DRDX(I)                                                     SPA03100
      DO 205 J=1,I                                                      SPA03101
      CALL SOURCE(TX(J))                                                SPA03102
  205 A(J)=V*SLOPE+U                                                    SPA03103
      SUM=0.                                                            SPA03104
      IM1=I-1                                                           SPA03105
      DO 201 J=1,IM1                                                    SPA03106
  201 SUM=T(J)*A(  J)+SUM                                               SPA03107
  210 T(I)=(DRDX(I)-SUM)/A(I)                                           SPA03108
      T(NXBODY)=0.                                                      SPA03109
C                                                                       SPA03110
C     DETERMINATION OF DOUBLET STRENGTHS AT CONTROL POINTS MIDWAY       SPA03111
C      BETWEEN BODY DEFINITION POINTS                                   SPA03112
C                                                                       SPA03113
C CALCULATION OF THE FIRST DOUBLET STRENGTH.                            SPA03114
C                                                                       SPA03115
      XFIELD=XF(1)                                                      SPA03116
      RFIELD=RF(1)                                                      SPA03117
      SLOPE=DRDX(1)                                                     SPA03118
      CALL DOUBLT(TX(1))                                                SPA03119
      A(1)=SLOPE*U+V                                                    SPA03120
      TC(1)=ALPHA/A(1)                                                  SPA03121
C                                                                       SPA03122
C CALCULATION OF THE REST OF THE DOUBLET STRENGTHS                      SPA03123
C                                                                       SPA03124
      DO 215 I=2,N                                                      SPA03125
      XFIELD=XF(I)                                                      SPA03126
      RFIELD=RF(I)                                                      SPA03127
      SLOPE=DRDX(I)                                                     SPA03128
      DO 212 J=1,I                                                      SPA03129
      CALL DOUBLT(TX(J))                                                SPA03130
  212 A(  J)=SLOPE*U+V                                                  SPA03131
      SUM=0.                                                            SPA03132
      IM1=I-1                                                           SPA03133
      DO 203 J=1,IM1                                                    SPA03134
  203 SUM=TC(J)*A(  J)+SUM                                              SPA03135
  215 TC(I)=(ALPHA-SUM)/A(I)                                            SPA03136
      TC(NXBODY)=0.                                                     SPA03137
C                                                                       SPA03138
C PRINT OUT OF BODY CHARACTERISTICS                                     SPA03139
C                                                                       SPA03140
      WRITE (6,701)                                                     SPA03141
      DO 800 I=1,NXBODY                                                 SPA03142
      II=I-1                                                            SPA03143
  800 WRITE(6,702)   XBODY(I),RBODY(I),NPBODY(I),II,TX(I),T(I),TC(I)    SPA03144
      RETURN                                                            SPA03145
      END                                                               SPA03146
```

```
      SUBROUTINE BLYOUT                                                 SPAOW  1
C                                                                       SPAOW  2
C     SUBROUTINE TO LAY OUT FUSELAGE CONSTANT U-VELOCITY PANELS         SPAOW  3
C                                                                       SPAOW  4
      COMMON/RPGEOM/                  SNT2(100),CST2(100),ZLC(200),     SPAOW  5
     1   ZRC(200),THTI(100),SANG,CANG                                   SPAOW  6
      COMMON /CONSTS/PI,PI2,DTOR,RTOD,FOURPI                            SPAOW  7
      COMMON /PSGEOM/FMAX,SSPAN,    RLTMC,BODYRL                        SPAOW  8
      COMMON/INDEX/NC,MS,MSP,NPANLS,NCOB,NBODY1,NHBODY2,NBD,NBIP,MP,    SPAOW  9
     1   NCP,MSP,N1P,N2,N2P,NPTOT                                       SPAOW 10
      COMMON /RGEOM/XBODC,ZBODC,CHD,SLPHLP,SLPHTE,PSI-LE(20),PSI-TE(20), SPAOW 11
     1       T(20)                                                      SPAOW 12
      COMMON/RPGEOM/XRF(200),XRH(200),XLF(200),XLB(200),YRC(200),       SPAOW 13
     1   YLC(200),XCPT(200),YCPT(200),ZCPT(200),SAPPLE(200),SAPPTE(200) SPAOW 14
C                                                                       SPAOW 15
      DATA EAL/0,95/                                                    SPAOW 16
C                                                                       SPAOW 17
C     FUNCTION STATEMENTS WHICH TRANSFORM BODY PANEL                    SPAOW 18
C        COORDINATES TO WING COORDINATE SYSTEM.                         SPAOW 19
C                                                                       SPAOW 20
      Y-WING(YPAN)=YPAN*CS2+FMAX*LS                                     SPAOW 21
      Z-WING(YPAN)=YPAN*SN2+FMAX*SN+ZH+U                                SPAOW 22
C                                                                       SPAOW 23
C     LAY OUT FUSELAGE CONSTANT U-VELOCITY PANELS                       SPAOW 24
C                                                                       SPAOW 25
      ANC=R=NC+8                                                        SPAOW 26
      DX=BODYPL/ANC+8                                                   SPAOW 27
      NBD=NHBDY1+NBODY2                                                 SPAOW 28
      NBIP=NBD+NC+8                                                     SPAOW 29
      SANG=ZB+U/FRMAX                                                   SPAOW 30
      ANG=ASIN(SANG)                                                    SPAOW 31
      IF(ABS(ANG-PI2),LE,.0017) GO TO 270                               SPAOW 32
      IF(ABS(ANG+PI2),LE,.0017) GO TO 280                               SPAOW 33
C                                                                       SPAOW 34
C     WING IS NOT TANGENT TO FUSELAGE                                   SPAOW 35
C                                                                       SPAOW 36
      IL=1                                                              SPAOW 37
      IU=NBDY1                                                          SPAOW 38
      ANGB=-ANG+PI2                                                     SPAOW 39
      AN1=NBDY1                                                         SPAOW 40
      THTS=(PI2+ANG)/AN1                                                SPAOW 41
      THTE=ANG                                                          SPAOW 42
      ITIMES=1                                                          SPAOW 43
  210 XBIP=2.0*FRMAX*SIN(THTS/2.0)                                      SPAOW 44
      CPY=0.5*XBIP                                                      SPAOW 45
      A1=0.0                                                            SPAOW 46
      DO 200 I=IL,IU                                                    SPAOW 47
      XFBIP=0.0                                                         SPAOW 48
      A1=A1+1.0                                                         SPAOW 49
      ANG=ANGB+(A1-0.5)*THTS                                            SPAOW 50
      SN=SIN(THT)                                                       SPAOW 51
      CS=COS(THT)                                                       SPAOW 52
      SN2=SIN(ANG)                                                      SPAOW 53
      CS2=COS(ANG)                                                      SPAOW 54
      DO 255 J=I,NBIP,NBD                                               SPAOW 55
      K=J+N2                                                            SPAOW 56
      SNT2(J)=SN2                                                       SPAOW 57
      CST2(J)=CS2                                                       SPAOW 58
C                                                                       SPAOW 59
C     TRANSFORM CONTROL POINT Y AND Z COORDINATES INTO WING SYSTEM      SPAOW 60
C                                                                       SPAOW 61
      YCPT(K)=Y-WING(CPY)                                               SPAOW 62
      ZCPT(K)=Z-WING(CPY)                                               SPAOW 63
C                                                                       SPAOW 64
C     TRANSFORM CORNER POINT Y AND Z COORDINATES INTO WING SYSTEM       SPAOW 65
C                                                                       SPAOW 66
      YLC(K)=Y-WING(0.)                                                 SPAOW 67
      ZLC(K)=Z-WING(0.)                                                 SPAOW 68
      YRC(K)=Y-WING(-BIP)                                               SPAOW 69
      ZRC(K)=Z-WING(-BIP)                                               SPAOW 70
      XLF(K)=XFBIP                                                      SPAOW 71
      XRF(K)=XFBIP                                                      SPAOW 72
      XFBIP=XFBIP-DX                                                    SPAOW 73
      XLB(K)=XFBIP                                                      SPAOW 74
      XRB(K)=XFBIP                                                      SPAOW 75
C                                                                       SPAOW 76
C     CONTROL POINT X-COORDINATE                                        SPAOW 77
C                                                                       SPAOW 78
      XCPT(K)=XFBIP+(1.0-FAC)*DX                                        SPAOW 79
C                                                                       SPAOW 80
C     POLAR ANGLE OF THIS PANEL, DEGREES                                SPAOW 81
C                                                                       SPAOW 82
      THTI(J)=(PI-THT)*RTOD                                             SPAOW 83
  255 CONTINUE                                                          SPAOW 84
      THT=THT+THTS                                                      SPAOW 85
  200 CONTINUE                                                          SPAOW 86
```

(f) Page 6.

Figure 1.- Continued.

```
      IF(ITIMES.EQ.2) GO TO 300                                          SPA04 87
      IL=NADCRI+1                                                        SPA04 88
      IU=NB                                                              SPA04 89
      AN2=NADCR2                                                         SPA04 90
      ANG=.PI                                                            SPA04 91
      THTS=(PI2-ANG)/AN2                                                 SPA04 92
      THT=.PI2                                                           SPA04 93
      ITIMES=2                                                           SPA04 94
      GO TO 210                                                          SPA04 95
C                                                                        SPA04 96
C     WING TANGENT TO BOTTOM OF FUSELAGE                                 SPA04 97
C                                                                        SPA04 98
  270 IL=1                                                               SPA04 99
      IU=NBOCB1                                                          SPA04100
      AN1=NADCR1                                                         SPA04101
      ANG=.PI                                                            SPA04102
      THTS=.PI/AN1                                                       SPA04103
      THT=.PI2                                                           SPA04104
      ITIMES=2                                                           SPA04105
      GO TO 210                                                          SPA04106
C                                                                        SPA04107
C     WING TANGENT TO TOP OF FUSELAGE                                    SPA04108
C                                                                        SPA04109
  280 IL=1                                                               SPA04110
      IU=NBOCR2                                                          SPA04111
      ANG=.PI                                                            SPA04112
      AN2=NBOCR2                                                         SPA04113
      THTS=PI/AN2                                                        SPA04114
      THT=.PI2                                                           SPA04115
      ITIMES=2                                                           SPA04116
      GO TO 210                                                          SPA04117
  300 CONTINUE                                                           SPA04118
      RETURN                                                             SPA04119
      END                                                                SPA04120
```

```
      SUBROUTINE CEL1(RES,AK,IER)                                        SPA05  1
C                                                                        SPA05  2
C     ******************************************************************SPA05  3
C                                                                        SPA05  4
C       SUBROUTINE CEL1                                                  SPA05  5
C                                                                        SPA05  6
C       PURPOSE                                                          SPA05  7
C          CALCULATE COMPLETE ELLIPTIC INTEGRAL OF FIRST KIND            SPA05  8
C                                                                        SPA05  9
C       USAGE                                                            SPA05 10
C          CALL CEL1(RES,AK,IER)                                         SPA05 11
C                                                                        SPA05 12
C       DESCRIPTION OF PARAMETERS                                        SPA05 13
C          RES   - RESULT VALUE                                          SPA05 14
C          AK    - MODULUS (INPUT)                                       SPA05 15
C          IER   - RESULTANT ERROR CODE WHERE                            SPA05 16
C                    IER=0  NO ERROR                                     SPA05 17
C                    IER=1  AK NOT IN RANGE -1 TO +1                     SPA05 18
C                                                                        SPA05 19
C       REMARKS                                                          SPA05 20
C          FOR AK=+1,-1 THE RESULT IS SET TO 1.E30.                      SPA05 21
C          FOR MODULUS AK AND COMPLEMENTARY MODULUS CK,                  SPA05 22
C          EQUATION AK**2+CK**2=1.0 IS USED.                             SPA05 23
C          AK MUST BE IN THE RANGE -1 TO +1                              SPA05 24
C                                                                        SPA05 25
C       SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                    SPA05 26
C          NONE                                                          SPA05 27
C                                                                        SPA05 28
C       METHOD                                                           SPA05 29
C          DEFINITION                                                    SPA05 30
C          CEL1(AK)=INTEGRAL(1/SQRT((1+T**2)*(1+(CK*T)**2))), SUMMED     SPA05 31
C          OVER T FROM 0 TO INFINITY).                                   SPA05 32
C          EQUIVALENT ARE THE DEFINITIONS                                SPA05 33
C          CEL1(AK)=INTEGRAL(1/((COS(T)SQRT(1+(CK*TAN(T))**2)),SUMMED    SPA05 34
C          OVER T FROM 0 TO PI/2),                                       SPA05 35
C          CEL1(AK)=INTEGRAL(1/SQRT(1-(AK*SIN(T))**2),SUMMED OVER T      SPA05 36
C          FROM 0 TO PI/2), WHERE K=SQRT(1.-CK*CK),                      SPA05 37
C          EVALUATION                                                    SPA05 38
C          LANDEN TRANSFORMATION IS USED FOR CALCULATION.                SPA05 39
C          REFERENCE                                                     SPA05 40
C          R.BULIRSCH, 'NUMERICAL CALCULATION OF ELLIPTIC INTEGRALS      SPA05 41
C          AND ELLIPTIC FUNCTIONS', HANDBOOK SERIES SPECIAL FUNCTIONS,   SPA05 42
C          NUMERISCHE MATHEMATIK VOL. 7, 1965, PP. 78-90.                SPA05 43
C                                                                        SPA05 44
C     ******************************************************************SPA05 45
C                                                                        SPA05 46
C                                                                        SPA05 47
      IER=0                                                              SPA05 48
C                                                                        SPA05 49
C     TEST MODULUS                                                       SPA05 50
C                                                                        SPA05 51
      GEO=1.-AK*AK                                                       SPA05 52
      IF(GEO)1,2,3                                                       SPA05 53
    1 IER=1                                                              SPA05 54
      RETURN                                                             SPA05 55
C                                                                        SPA05 56
C     SET RESULT VALUE TO UFLOW                                          SPA05 57
C                                                                        SPA05 58
    2 RES=1.E30                                                          SPA05 59
      RETURN                                                             SPA05 60
    3 GEO=SQRT(GEO)                                                      SPA05 61
      ARI=1.                                                             SPA05 62
    4 AARI=ARI                                                           SPA05 63
      TEST=AARI*1.E-4                                                    SPA05 64
      ARI=GEO+ARI                                                        SPA05 65
C                                                                        SPA05 66
C     TEST OF ACCURACY                                                   SPA05 67
C                                                                        SPA05 68
      IF(AARI-GEO-TEST)6,6,5                                             SPA05 69
    5 GEO=SQRT(AARI*GEO)                                                 SPA05 70
      ARI=0.5*ARI                                                        SPA05 71
      GO TO 4                                                            SPA05 72
    6 RES=3.1415926/ARI                                                  SPA05 73
      RETURN                                                             SPA05 74
      END                                                                SPA05 75
```

```
      SUBROUTINE CEL2(RES,AK,A,B,IER)                                    SPA06  1
C                                                                        SPA06  2
C     ******************************************************************SPA06  3
C                                                                        SPA06  4
C       SUBROUTINE CEL2                                                  SPA06  5
C                                                                        SPA06  6
C       PURPOSE                                                          SPA06  7
C          COMPUTES THE GENERALIZED COMPLETE ELLIPTIC INTEGRAL OF        SPA06  8
C          SECOND KIND.                                                  SPA06  9
C                                                                        SPA06 10
C       USAGE                                                            SPA06 11
C          CALL CEL2(RES,AK,A,B,IER)                                     SPA06 12
C                                                                        SPA06 13
C       DESCRIPTION OF PARAMETERS                                        SPA06 14
C          RES   - RESULT VALUE                                          SPA06 15
C          AK    - MODULUS (INPUT)                                       SPA06 16
C          A     - CONSTANT TERM IN NUMERATOR                            SPA06 17
C          B     - FACTOR OF QUADRATIC TERM IN NUMERATOR                 SPA06 18
C          IER   - RESULTANT ERROR CODE WHERE                            SPA06 19
C                    IER=0  NO ERROR                                     SPA06 20
C                    IER=1  AK NOT IN RANGE -1 TO +1                     SPA06 21
C                                                                        SPA06 22
C       REMARKS                                                          SPA06 23
C          FOR AK = +1,-1 THE RESULT VALUE IS SET TO 1.E30 IF B IS       SPA06 24
C          POSITIVE, TO -1.E30 IF B IS NEGATIVE.                         SPA06 25
```

(g) Page 7.

Figure 1.- Continued.

```
C          SPECIAL CASES ARE                                                    SPA06 26
C          R(K) OBTAINED WITH A = 1, B = 1                                      SPA06 27
C          P(K) OBTAINED WITH A = 1, B = CK**2 WHERE CK IS                      SPA06 28
C          COMPLEMENTARY MODULUS,                                               SPA06 29
C          A(K) OBTAINED WITH A = 1, P = 0                                      SPA06 30
C          D(K) OBTAINED WITH A = 0, M = 1                                      SPA06 31
C          WHERE M, E, B, D DEFINE SPECIAL CASES OF THE GENERALIZED             SPA06 32
C          COMPLETE ELLIPTIC INTEGRAL OF SECOND KIND IN THE USUAL               SPA06 33
C          NOTATION, AND THE ARGUMENT X OF THESE FUNCTIONS MEANS                SPA06 34
C          THE MODULUS,                                                         SPA06 35
C                                                                               SPA06 36
C       SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                           SPA06 37
C          NONE                                                                 SPA06 38
C                                                                               SPA06 39
C       METHOD                                                                  SPA06 40
C          DEFINITION                                                           SPA06 41
C          RES=INTEGRAL((A+B*T*T)/(SQRT((1+T*T)*(1+(CK*T)**2))*(1+T*T)))SPA06 42
C          SUMMED OVER T FROM 0 TO INFINITY,                                    SPA06 43
C          EVALUATION                                                           SPA06 44
C          LANDENS TRANSFORMATION IS USED FOR CALCULATION,                      SPA06 45
C          REFERENCE                                                            SPA06 46
C          R.BULIRSCH, 'NUMERICAL CALCULATION OF ELLIPTIC INTEGRALS             SPA06 47
C          AND ELLIPTIC FUNCTIONS', HANDBOOK SERIES SPECIAL FUNCTIONS,          SPA06 48
C          NUMERISCHE MATHEMATIK VOL. 7, 1965, PP. 78-90,                       SPA06 49
C                                                                               SPA06 50
C     ***************************************************************SPA06 51
C                                                                               SPA06 52
C                                                                               SPA06 53
      IER=0                                                                     SPA06 54
C                                                                               SPA06 55
C     TEST MODULUS                                                              SPA06 56
C                                                                               SPA06 57
      GEO=1.-AK*AK                                                              SPA06 58
      IF(GEO)1,2,6                                                              SPA06 59
    1 IER=1                                                                     SPA06 60
      RETURN                                                                    SPA06 61
C                                                                               SPA06 62
C     SET RESULT VALUE = OVERFLOW                                               SPA06 63
C                                                                               SPA06 64
    2 IF(N)3,3,4                                                                SPA06 65
    3 RES=1.E50                                                                 SPA06 66
      RETURN                                                                    SPA06 67
    4 RES=1.E50                                                                 SPA06 68
      RETURN                                                                    SPA06 69
    5 RES=A                                                                     SPA06 70
      RETURN                                                                    SPA06 71
C                                                                               SPA06 72
C     COMPUTE INTEGRAL                                                          SPA06 73
C                                                                               SPA06 74
    6 GEO=SQRT(GEO)                                                             SPA06 75
      AMI=1.                                                                    SPA06 76
      AA=A                                                                      SPA06 77
      AN=A+B                                                                    SPA06 78
      P=B                                                                       SPA06 79
    7 AR=P+AA*GEO                                                               SPA06 80
      BR=P+B                                                                    SPA06 81
      AA=AN                                                                     SPA06 82
      AAMI=ARI                                                                  SPA06 83
      AMI=GEO+AMI                                                               SPA06 84
      AN=2./ARI+AN                                                              SPA06 85
C                                                                               SPA06 86
C     TEST OF ACCURACY                                                          SPA06 87
C                                                                               SPA06 88
      IF(AARI-GEO-1.E-9*AARI)9,9,8                                              SPA06 89
    8 GEO=SQRT(GEO*ARI)                                                         SPA06 90
      GEO=GEO+GEO                                                               SPA06 91
      GO TO 7                                                                   SPA06 92
    9 RES=.785398164*AN/ARI                                                     SPA06 93
      RETURN                                                                    SPA06 94
      END                                                                       SPA06 95
```

```
      SUBROUTINE DIRCOS (A,D)                                                   SPA07  1
C     SUBROUTINE TO CALCULATE DIRECTION COSINES                                 SPA07  2
C                                                                               SPA07  3
      DIMENSION A(12),D(3,3)                                                    SPA07  4
      SPSI=SIN(A(10))                                                           SPA07  5
      CPSI=COS(A(10))                                                           SPA07  6
      STHE=SIN(A(11))                                                           SPA07  7
      CTHE=COS(A(11))                                                           SPA07  8
      SPHI=SIN(A(12))                                                           SPA07  9
      CPHI=COS(A(12))                                                           SPA07 10
      D(1,1)=CTHE*CPSI                                                          SPA07 11
      D(1,2)=SPHI*STHE*CPSI-CPHI*SPSI                                           SPA07 12
      D(1,3)=CPHI*STHE*CPSI+SPHI*SPSI                                           SPA07 13
      D(2,1)=CTHE*SPSI                                                          SPA07 14
      D(2,2)=SPHI*STHE*SPSI+CPHI*CPSI                                           SPA07 15
      D(2,3)=CPHI*STHE*SPSI-SPHI*CPSI                                           SPA07 16
      D(3,1)=-STHE                                                              SPA07 17
      D(3,2)=SPHI*CTHE                                                          SPA07 18
      D(3,3)=CPHI*CTHE                                                          SPA07 19
      RETURN                                                                    SPA07 20
      END                                                                       SPA07 21
```

```
      SUBROUTINE DOUBLT(TX)                                                     SPA08  1
C                                                                               SPA08  2
C SUBROUTINE TO CALCULATE THE STRENGTH OF A LINEAR LINE DOUBLET OF              SPA08  3
C UNIT SLOPE WITH ORIGIN AT TX(J),                                              SPA08  4
C                                                                               SPA08  5
      COMMON/BRCE/XFIELD,RFIELD,BSQ,X2,     U,V,VT                              SPA08  6
      COMMON /FLOW/DUM1(5),BETA,BETABQ,DUM2                                     SPA08  7
C                                                                               SPA08  8
      X1=XFIELD-TX                                                              SPA08  9
      BR=BETA*RFIELD                                                            SPA08 10
      IF(X1-1.E-8*BR     )GO TO 10                                              SPA08 11
      XBR=X1/BR                                                                 SPA08 12
      XX=SQRT(XBR*XBR+1.)                                                       SPA08 13
      U=BETA*XX                                                                 SPA08 14
      ACOSH=ALOG(XBR+XX)                                                        SPA08 15
      XX=XBR*XX                                                                 SPA08 16
      V=-.5*BETABQ*(ACOSH+XX)                                                   SPA08 17
      RETURN                                                                    SPA08 18
C                                                                               SPA08 19
C FIELD POINT IS AHEAD OF MACH CONE FROM DOUBLET ORIGIN,                        SPA08 20
C                                                                               SPA08 21
   10 U=0.                                                                      SPA08 22
      V=0.                                                                      SPA08 23
      RETURN                                                                    SPA08 24
      END                                                                       SPA08 25
```

```
      SUBROUTINE OPCOEF                                                         SPA09  1
C                                                                               SPA09  2
C     SUBROUTINE TO CALCULATE THE LEFT HAND SIDE OF THE EQUATIONS               SPA09  3
C     FOR DOWNWASH ON THE WING, PYLON AND FUSELAGE INTERFERENCE PANELS          SPA09  4
C                                                                               SPA09  5
      COMMON A(N(200,201)                                                       SPA09  6
      COMMON/BPGEOM/                 SAT2(100),CST2(100),ZLE(200),              SPA09  7
     1   ZRE(200),THTI(100),SANG=,CANG=                                         SPA09  8
      COMMON /CONFIG/NFU,NPY,NSTRS,LVSWP                                        SPA09  9
      COMMON/ICVEL/UP,VP,=P,TI.IF,          DELTP(200)                          SPA09 10
      COMMON/INDEX/NC=,MS=,MSP,NPANLB,NCHB,NBPCH1,NHUCR2,NBD,NBIP,MP,           SPA09 11
     1   NCP,MSP,N1P,N2,N2P,NPTOT                                               SPA09 12
      COMMON /WGPGM/XBHUC,ZHHU,CRW,SLPWLE,SLPWTE,PSIWLE(20),PSIWTE(20),         SPA09 13
```

(h) Page 8.

Figure 1.- Continued.

```
     1  Y(20)
      COMMON/WPGEOM/XRF(200),XRB(200),XLF(200),XLB(200),YMC(200),
     1    YLC(200),XCPT(200),YCPT(200),ZCPT(200),SWPLE(200),SWPTE(200)
C
C
C     I IS THE INDEX OF THE INFLUENCING PANEL
C
C
C     INFLUENCE OF WING PANELS
C
      DO 100 I=1,NPANLS
      II=I
      IF=1
C
C     WING PANELS ON WING CONTROL POINTS
C
      DO 50 J=1,NPANLS
      CALL VELWP(XCPT(J),YCPT(J),ZCPT(J))
   50 FVN(J,I)=WP
C
C     WING PANELS ON PYLON CONTROL POINTS
C
      IF(NPY,EQ,0) GO TO 61
      DO 60 J=NIP,N2
      CALL VELWP(XCPT(J),YCPT(J),ZCPT(J))
   60 FVN(J,I)=WVP
   61 CONTINUE
C
C     WING PANELS ON FUSELAGE CONTROL POINTS
C
      IF(NFU,EQ,0) GO TO 100
      DO 80 J=N2P,NPTOT
      CALL VELWP(XCPT(J),YCPT(J),ZCPT(J))
      NB=J-N2
      WB=WP*CST2(NB)+VP*SNT2(NB)
   80 FVN(J,I)=WB
  100 CONTINUE
C
C     INFLUENCE OF PYLON PANELS
C
      IF(NPY,EQ,0) GO TO 201
      DO 200 I=NIP,N2
      II=I
      IF=1
C
C     PYLON PANELS ON WING CONTROL POINTS
C
      DO 150 J=1,NPANLS
      CALL VELPP(XCPT(J),YCPT(J),ZCPT(J))
  150 FVN(J,I)=WP
C
C     PYLON PANELS ON PYLON CONTROL POINTS
C
      DO 160 J=NIP,N2
      CALL VELPP(XCPT(J),YCPT(J),ZCPT(J))
  160 FVN(J,I)=WVP
C
C     PYLON PANELS ON FUSELAGE CONTROL POINTS
C
      IF(NFU,EQ,0) GO TO 200
      DO 180 J=N2P,NPTOT
      CALL VELPP(XCPT(J),YCPT(J),ZCPT(J))
      NB=J-N2
      WB=WP*CST2(NB)+VP*SNT2(NB)
  180 FVN(J,I)=WB
  200 CONTINUE
C
C     INFLUENCE OF FUSELAGE INTERFERENCE PANELS
C
  201 IF(NFU,EQ,0) RETURN
      DO 300 I=N2P,NPTOT
      II=I
      IF=1
C
C     FUSELAGE PANELS ON WING CONTROL POINTS
C
      DO 250 J=1,NPANLS
      CALL VELBD(XCPT(J),YCPT(J),ZCPT(J))
  250 FVN(J,I)=WP
C
C     FUSELAGE PANELS ON PYLON CONTROL POINTS
C
      IF(NPY,EQ,0) GO TO 261
      DO 260 J=NIP,N2
      CALL VELBD(XCPT(J),YCPT(J),ZCPT(J))
  260 FVN(J,I)=WVP
  261 CONTINUE
C
C     FUSELAGE PANELS ON FUSELAGE CONTROL POINTS
C
      DO 280 J=N2P,NPTOT
      CALL VELBD(XCPT(J),YCPT(J),ZCPT(J))
      NB=J-N2
      WB=WP*CST2(NB)+VP*SNT2(NB)
  280 FVN(J,I)=WB
  300 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DFRHS
C
C     SUBROUTINE TO CALCULATE THE RIGHT HAND SIDE OF THE EQUATIONS
C     FOR U+/VINF ON THE WING, PYLON, AND FUSELAGE INTERFERENCE PANELS
C
C     LOGICAL CENTER
C
      COMMON/BPGEOM/                    SNT2(100),CST2(100),ZLC(200),
     1    ZRC(200),THTI(100),SANG+,CANG+
      COMMON/CAMBER/ALPHAL(200)
      COMMON /CONFIG/NFU,NPY,NSTRS,LVS+P
      COMMON/EXVEL/UEI(200),VEI(200),WEI(200),CIM(200)
      COMMON /FLO+/ALPACR,GAMP,FMACH,RHO,VINF,BETA,METASU,FMCHSU
      COMMON /FSOR/FXL(101),FSS(100),FDS(100),NFSUM
      COMMON/ICVEL/UP,VP,WP,II,IF,          DELTP(200)
      COMMON/INDEX/NCN,NSN,NSWP,NPANLS,NCNB,NWDCN1,NWDCN2,NWD,NBIP,MP,
     1   NCP,NSP,N1P,N2,N2P,NPTOT
      COMMON/PYGEOM/DMZ(20),XPLE,YPL,CRP,WP,PBIPLE,IP,SLLE,SLTE,PSLPDP,
     1       CENTER
      COMMON /SBOR/BXL(101,1),SSS(100,1),SDS(100,1),NSSOR(1)
      COMMON /STGEOM/BMAX(1),BLTHC(1),XWBO(1),YWBU(1),ZWBU(1),
     1      NBPOLY(1),BXEND(7,1),SCOPE(7,7,1),YBSO(1),YBSU(1),ZBSO(1),
     2      BIBCR(1),NUMSTR(1),SSIBCR(1),CBIBCR(1)
      COMMON /THKDAT/NCNS,NCPS,NS,NPS  ,NTHP,XMFT(400),XMBT(400),
     1   XLFT(400),XLBT(400),YMCT(400),YLCT(400),ZMCT(400),ZLCT(400),
     2   THETAL(400),THETPL(200),BLLPT(400),BLTET(400),DZDX(400)
      COMMON /VELARG/X,Y,Z,U,V,W,EM,TLRNC,TIPY,PYPNL
      COMMON /WPGEOM/XRC,ZRWU,CHW,BLP+LE,SLP+TE,PBI+LE(20),PSI+TE(20),
     1      DM(20)
      COMMON/WPGEOM/XRF(200),XRB(200),XLF(200),XLB(200),YMC(200),
     1   YLC(200),XCPT(200),YCPT(200),ZCPT(200),SWPLE(200),SWPTE(200)
C
C     SET ALL VELOCITIES EQUAL TO ZERO
C
      DO 90 J=1,NPTOT
      UEI(J)=0.0
      VEI(J)=0.0
   90 WEI(J)=0.0
C
C     FUSELAGE-INDUCED VELOCITIES ON WING AND PYLON CONTROL POINTS
C
      IF(NFU,EQ,0) GO TO 101
```

(i) Page 9.

Figure 1.- Continued.

```
      DO 100 J=1,N2                                                     SPA10 43
      XVC=XCPT(J)+XB=UC                                                 SPA10 44
      ZVC=ZCPT(J)+ZB=U                                                  SPA10 45
      CALL VELCAL(BSS,BDS,BXL,NBSUM,XVC,YCPT(J),ZVC,UB,VB,WB)           SPA10 46
      UEI(J)=UEI(J)+UB                                                  SPA10 47
      VEI(J)=VEI(J)+VB                                                  SPA10 48
  100 WEI(J)=WEI(J)+WB                                                  SPA10 49
  101 CONTINUE                                                          SPA10 50
C                                                                       SPA10 51
C     STORE INDUCED VELOCITIES ON WING, PYLON, AND FUSELAGE             SPA10 52
C     CONTROL POINTS                                                    SPA10 53
C                                                                       SPA10 54
      IF(NBTRS,EQ,0) GO TO 149                                          SPA10 55
      DO 110 J=1,NPTOT                                                  SPA10 56
      DO 105 K=1,NBTRS                                                  SPA10 57
      XVC=XCPT(J)-X=BOC(K)                                              SPA10 58
      YVCS=YCPT(J)-Y=SO(K)                                              SPA10 59
      ZVC=ZCPT(J)-Z=BO(K)                                               SPA10 60
      XVCB=XVC*CBIBCR(K)-ZVC*SSIBCR(K)                                  SPA10 61
      ZVCS=XVC*SSIBCR(K)+ZVC*CBIBCR(K)                                  SPA10 62
      CALL VELCAL(BSS(1,K),BDS(1,K),BXL(1,K),NSSUM(K),                  SPA10 63
     1            XVCB,YVCS,ZVCB,US,VS,WS)                              SPA10 64
      UB=US*CSIHCH(K)+S*SSIHCH(K)                                       SPA10 65
      WB=-US*SSIBCR(K)+S*CSIBCR(K)                                      SPA10 66
      UEI(J)=UEI(J)+UB                                                  SPA10 67
      VEI(J)=VEI(J)+VB                                                  SPA10 68
  105 WEI(J)=WEI(J)+WB                                                  SPA10 69
  110 CONTINUE                                                          SPA10 70
C                                                                       SPA10 71
C     THICKNESS INDUCED VELOCITIES                                      SPA10 72
C       NOTE: NO WING-ON-WING OR PYLON-ON-PYLON VELOCITIES CALCULATED   SPA10 73
C                                                                       SPA10 74
C     IF PYLON NOT PRESENT, SKIP ALL PYLON CALCULATIONS                 SPA10 75
C                                                                       SPA10 76
C                                                                       SPA10 77
  149 IF(NPY ,EQ, 0) GO TO 152                                          SPA10 78
      II=NS+1                                                           SPA10 79
      IF=NTWP                                                           SPA10 80
C                                                                       SPA10 81
C     PYLON ON WING CONTROL POINTS                                      SPA10 82
C                                                                       SPA10 83
C         J IS THE CONTROL POINT INDEX                                  SPA10 84
C                                                                       SPA10 85
      DO 150 J=1,NPANLS                                                 SPA10 86
      CALL VELPTH(XCPT(J),YCPT(J),ZCPT(J))                              SPA10 87
      UEI(J)=UEI(J)+UP                                                  SPA10 88
      VEI(J)=VEI(J)+VP                                                  SPA10 89
      WEI(J)=WEI(J)+WP                                                  SPA10 90
  150 CONTINUE                                                          SPA10 91
C                                                                       SPA10 92
C     WING ON PYLON CONTROL POINTS                                      SPA10 93
C                                                                       SPA10 94
      II=1                                                              SPA10 95
      IF=NS                                                             SPA10 96
      DO 151 J=NP1P,N2                                                  SPA10 97
      CALL VELWTH(XCPT(J),YCPT(J),ZCPT(J))                              SPA10 98
      UEI(J)=UEI(J)+UP                                                  SPA10 99
      VEI(J)=VEI(J)+VP                                                  SPA10100
      WEI(J)=WEI(J)+WP                                                  SPA10101
  151 CONTINUE                                                          SPA10102
C                                                                       SPA10103
C     WING AND PYLON ON FUSELAGE CONTROL POINTS                         SPA10104
C                                                                       SPA10105
  152 IF (NFU ,EQ, 0) GO TO 111                                         SPA10106
C                                                                       SPA10107
C                 WING THICKNESS                                        SPA10108
C                                                                       SPA10109
      II=1                                                              SPA10110
      IF=NS                                                             SPA10111
      DO 156 J=N2P,NPTOT                                                SPA10112
      CALL VELWTH(XCPT(J),YCPT(J),ZCPT(J))                              SPA10113
      UEI(J)=UEI(J)+UP                                                  SPA10114
      VEI(J)=VEI(J)+VP                                                  SPA10115
      WEI(J)=WEI(J)+WP                                                  SPA10116
  156 CONTINUE                                                          SPA10117
C                                                                       SPA10118
C                 PYLON THICKNESS                                       SPA10119
C                                                                       SPA10120
      IF (NPY ,EQ, 0) GO TO 111                                         SPA10121
      II=NS+1                                                           SPA10122
      IF=NTWP                                                           SPA10123
      DO 155 J=N2P,NPTOT                                                SPA10124
      CALL VELPTH(XCPT(J),YCPT(J),ZCPT(J))                              SPA10125
      UEI(J)=UEI(J)+UP                                                  SPA10126
      VEI(J)=VEI(J)+VP                                                  SPA10127
      WEI(J)=WEI(J)+WP                                                  SPA10128
  155 CONTINUE                                                          SPA10129
C                                                                       SPA10130
C     CALCULATE RIGHT HAND SIDE                                         SPA10131
C                                                                       SPA10132
C     -- WING BOUNDARY CONDITION --                                     SPA10133
C                                                                       SPA10134
  111 CONTINUE                                                          SPA10135
      DO 120 J=1,NPANLS                                                 SPA10136
  120 CIR(J)=ALPACH-ALPHAL(J)-(UEI(J)*ALPHAL(J)+WEI(J))                 SPA10137
C                                                                       SPA10138
C     -- PYLON BOUNDARY CONDITION --                                    SPA10139
C                                                                       SPA10140
      IF(NPY,EQ,0) GO TO 131                                            SPA10141
      DO 130 J=NP1P,N2                                                  SPA10142
      IF(CENTPH) VEI(J)=0,                                              SPA10143
  130 CIR(J)=VEI(J)                                                     SPA10144
C                                                                       SPA10145
C     -- FUSELAGE INTERFERENCE PANEL BOUNDARY CONDITION --              SPA10146
C          ONLY V NORMAL IS USED IN FUSELAGE BOUNDARY CONDITION         SPA10147
C                                                                       SPA10148
  131 IF(NFU,EQ,0) RETURN                                               SPA10149
      DO 140 J=N2P,NPTOT                                                SPA10150
      NB=J-N2                                                           SPA10151
      A=-WEI(J)*CBT2(NB)+VEI(J)*SNT2(NB)                                SPA10152
  140 CIR(J)=A                                                          SPA10153
      RETURN                                                            SPA10154
      END                                                               SPA10155
```

```
      SUBROUTINE ELII(RES,X,CK)                                         SPA11  1
C     ******************************************************************SPA11  2
C     ******************************************************************SPA11  3
C     *                                                                *SPA11  4
C     *     SUBROUTINE ELII                                            *SPA11  5
C     *                                                                *SPA11  6
C     *     PURPOSE                                                    *SPA11  7
C     *        COMPUTES THE ELLIPTIC INTEGRAL OF FIRST KIND            *SPA11  8
C     *                                                                *SPA11  9
C     *     USAGE                                                      *SPA11 10
C     *        CALL ELII(RES,X,CK)                                     *SPA11 11
C     *                                                                *SPA11 12
C     *     DESCRIPTION OF PARAMETERS                                  *SPA11 13
C     *        RES   - RESULT VALUE                                    *SPA11 14
C     *        X     - UPPER INTEGRATION BOUND (ARGUMENT OF ELLIPTIC   *SPA11 15
C     *                INTEGRAL OF FIRST KIND)                         *SPA11 16
C     *        CK    - COMPLEMENTARY MODULUS                           *SPA11 17
C     *                                                                *SPA11 18
C     *     REMARKS                                                    *SPA11 19
C     *        MODULUS K = SQRT(1.-CK*CK).                             *SPA11 20
C     *                                                                *SPA11 21
C     *     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED              *SPA11 22
C     *        NONE                                                    *SPA11 23
C     *                                                                *SPA11 24
C     *     METHOD                                                     *SPA11 25
C     *        DEFINITION                                              *SPA11 26
```

(j) Page 10.

Figure 1.- Continued.

```
C          RES=INTEGRAL(1/SQRT((1+T*T)*(1+(CK*T)**2))), SUMMED          SPA11 27
C          OVER T FROM 0 TO X),                                         SPA11 28
C          EQUIVALENT ARE THE DEFINITIONS                               SPA11 29
C          RES=INTEGRAL(1/(COS(T)*SQRT(1+(CK*TAN(T))**2))), SUMMED      SPA11 30
C          OVER T FROM 0 TO ATAN(X)),                                   SPA11 31
C          RES=INTEGRAL(1/SQRT(1-(K*SIN(T))**2)), SUMMED OVER           SPA11 32
C          T FROM 0 TO ATAN(X)),                                        SPA11 33
C          EVALUATION                                                   SPA11 34
C          LANDENS TRANSFORMATION IS USED FOR CALCULATION,              SPA11 35
C          REFERENCE                                                    SPA11 36
C          M, BULIRSCH, NUMERICAL CALCULATION OF ELLIPTIC INTEGRALS ANDSPA11 37
C                ELLIPTIC FUNCTIONS,                                    SPA11 38
C                HANDBOOK SERIES OF SPECIAL FUNCTIONS                   SPA11 39
C                NUMERISCHE MATHEMATIK VOL, 7, 1965, PP, 78-90.         SPA11 40
C                                                                       SPA11 41
C     ****************************************************************SPA11 42
C                                                                       SPA11 43
C                                                                       SPA11 44
      IF(X)2,1,2                                                        SPA11 45
    1 RES=0.                                                            SPA11 46
      RETURN                                                            SPA11 47
    2 IF(CK)4,3,4                                                       SPA11 48
    3 RES=ALOG(ABS(X)+SQRT(1.+X*X))                                     SPA11 49
      GOTO 13                                                           SPA11 50
    4 ANGLE=ABS(1./X)                                                   SPA11 51
      GEO=ABS(CK)                                                       SPA11 52
      ARI=1.                                                            SPA11 53
      PIM=0.                                                            SPA11 54
    5 SUGEO=AMI*GEO                                                     SPA11 55
      AAMI=ARI                                                          SPA11 56
      AMI=GEO+ARI                                                       SPA11 57
      ANGLE=-SUGEO/ANGLE+ANGLE                                          SPA11 58
      SUGEO=SQRT(SUGEO)                                                 SPA11 59
      IF(ANGLE)7,6,7                                                    SPA11 60
C     REPLACE 0 BY SMALL VALUE                                          SPA11 61
    6 ANGLE=SUGEO*1.E-8                                                 SPA11 62
    7 TEST=AAMI*1.E-8                                                   SPA11 63
      IF(ABS(AAMI-GEO)-TEST)10,10,8                                     SPA11 64
    8 GEO=SUGEO+SUGEO                                                   SPA11 65
      PIM=PIM*PIM                                                       SPA11 66
      IF(ANGLE)9,5,5                                                    SPA11 67
    9 PIM=PIM*3.1415927                                                 SPA11 68
      GOTO 5                                                            SPA11 69
   10 IF(ANGLE)11,12,12                                                 SPA11 70
   11 PIM=PIM*3.1415927                                                 SPA11 71
   12 RES=(ATAN(ARI/ANGLE)+PIM)/ARI                                     SPA11 72
   13 IF(X)14,15,15                                                     SPA11 73
   14 RES=-RES                                                          SPA11 74
   15 RETURN                                                            SPA11 75
      END                                                               SPA11 76
```

```
      SUBROUTINE ELI2(R,X,CK,A,B)                                       SPA12  1
C                                                                       SPA12  2
C     ****************************************************************SPA12  3
C                                                                       SPA12  4
C        SUBROUTINE ELI2                                                SPA12  5
C                                                                       SPA12  6
C        PURPOSE                                                        SPA12  7
C           COMPUTES THE GENERALIZED ELLIPTIC INTEGRAL OF SECOND KIND   SPA12  8
C                                                                       SPA12  9
C        USAGE                                                          SPA12 10
C           CALL ELI2(R,X,CK,A,B)                                       SPA12 11
C                                                                       SPA12 12
C        DESCRIPTION OF PARAMETERS                                      SPA12 13
C           R      - RESULT VALUE                                       SPA12 14
C           X      - UPPER INTEGRATION BOUND (ARGUMENT OF ELLIPTIC      SPA12 15
C                    INTEGRAL OF SECOND KIND)                           SPA12 16
C           CK     - COMPLEMENTARY MODULUS                              SPA12 17
C           A      - CONSTANT TERM IN NUMERATOR                         SPA12 18
C           B      - QUADRATIC TERM IN NUMERATOR                        SPA12 19
C                                                                       SPA12 20
C        REMARKS                                                        SPA12 21
C           MODULUS K = SQRT(1.-CK*CK),                                 SPA12 22
C           SPECIAL CASES OF THE GENERALIZED ELLIPTIC INTEGRAL OF       SPA12 23
C           SECOND KIND ARE                                             SPA12 24
C           F(ATAN(X),K) OBTAINED WITH A=1., B=1.,                      SPA12 25
C           E(ATAN(X),K) OBTAINED WITH A=1., B=CK*CK,                   SPA12 26
C           B(ATAN(X),K) OBTAINED WITH A=1., B=0.,                      SPA12 27
C           D(ATAN(X),K) OBTAINED WITH A=0., B=1.,                      SPA12 28
C                                                                       SPA12 29
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                  SPA12 30
C           NONE                                                        SPA12 31
C                                                                       SPA12 32
C        METHOD                                                         SPA12 33
C           DEFINITION                                                  SPA12 34
C           R=INTEGRAL((A+B*T*T)/(SQRT((1+T*T)*(1+(CK*T)**2)))*(1+T*T)), SPA12 35
C                 SUMMED OVER T FROM 0 TO X),                           SPA12 36
C           EQUIVALENT IS THE DEFINITION                                SPA12 37
C           R=INTEGRAL((A+(B-A)*(SIN(T))**2)/SQRT(1-(K*SIN(T))**2),     SPA12 38
C                 SUMMED OVER T FROM 0 TO ATAN(X)),                     SPA12 39
C           EVALUATION                                                  SPA12 40
C           LANDENS TRANSFORMATION IS USED FOR CALCULATION,             SPA12 41
C           REFERENCE                                                   SPA12 42
C           M, BULIRSCH, NUMERICAL CALCULATION OF ELLIPTIC INTEGRALS ANDSPA12 43
C                 ELLIPTIC FUNCTIONS                                    SPA12 44
C                 HANDBOOK SERIES OF SPECIAL FUNCTIONS                  SPA12 45
C                 NUMERISCHE MATHEMATIK VOL, 7, 1965, PP, 78-90,        SPA12 46
C                                                                       SPA12 47
C     ****************************************************************SPA12 48
C                                                                       SPA12 49
C           TEST ARGUMENT                                               SPA12 50
      IF(X)2,1,2                                                        SPA12 51
    1 R=0.                                                              SPA12 52
      RETURN                                                            SPA12 53
C           TEST MODULUS                                                SPA12 54
    2 C=0.                                                              SPA12 55
      D=0.5                                                             SPA12 56
      IF(CK)7,3,7                                                       SPA12 57
    3 R=SQRT(1.+X*X)                                                    SPA12 58
      R=(A-B)*ABS(X)/R+B*ALOG(ABS(X)+R)                                 SPA12 59
C           TEST SIGN OF ARGUMENT                                       SPA12 60
    4 R=R+C*(A-B)                                                       SPA12 61
      IF(X)5,6,6                                                        SPA12 62
    5 R=-R                                                              SPA12 63
    6 RETURN                                                            SPA12 64
C           INITIALIZATION                                              SPA12 65
    7 AN=(B+A)*0.5                                                      SPA12 66
      AABA                                                              SPA12 67
      R=B                                                               SPA12 68
      ANG=ABS(1./X)                                                     SPA12 69
      PIM=0.                                                            SPA12 70
      ISI=0                                                             SPA12 71
      ARI=1.                                                            SPA12 72
      GEO=ABS(CK)                                                       SPA12 73
C           LANDEN TRANSFORMATION                                       SPA12 74
    8 R=AA*GEO+R                                                        SPA12 75
      SGEO=ARI*GEO                                                      SPA12 76
      AABAN                                                             SPA12 77
      AARI=ARI                                                          SPA12 78
C           ARITHMETIC MEAN                                             SPA12 79
      AMI=GEO+AMI                                                       SPA12 80
C           SUM OF SINE VALUES                                          SPA12 81
      AN=(R/ARI+AA)*0.5                                                 SPA12 82
      AANG=ABS(ANG)                                                     SPA12 83
      ANG=-SGEO/ANG+ANG                                                 SPA12 84
      PIMA=PIM                                                          SPA12 85
      IF(ANG)10,9,11                                                    SPA12 86
    9 ANG=-1.E-8*AANG                                                   SPA12 87
   10 PIM=PIM+3.1415927                                                 SPA12 88
      ISI=ISI+1                                                         SPA12 89
```

(k) Page 11.

Figure 1.- Continued.

```
   11 AANG=AARI*ANI+ANG*ANG                                              SPA12 90
      P=D/SQRT(AANG)                                                     SPA12 91
      IF(ISI)13,12,12                                                    SPA12 92
   12 ISI=ISI-4                                                          SPA12 93
   13 IF(ISI+2)15,14,14                                                  SPA12 94
   14 P=-P                                                               SPA12 95
   15 C=C+P                                                              SPA12 96
      D=D*(AARI*GEO)+0.5/ARI                                             SPA12 97
      IF(ABS(AANI*GEO)-1.E-09*AARI)17,17,16                              SPA12 98
   16 SGE=SQRT(SGEO)                                                     SPA12 99
C        GEOMETRIC MEAN                                                 SPA12100
      GEO=SGEO*SGEO                                                      SPA12101
      PI=PI*PI*A                                                         SPA12102
      ISI=ISI+ISI                                                        SPA12103
      GOTO 8                                                             SPA12104
C        ACCURACY WAS SUFFICIENT                                        SPA12105
   17 A=(ATAN(ARI/ANG)+PI+)*AN/ARI                                       SPA12106
      C=C+D*ANG/AANG                                                     SPA12107
      GOTO 8                                                             SPA12108
      END                                                                SPA12109
```

```
      SUBROUTINE FLSQFY (M,N,X,Y,W,MN1,S1,A,IER)                         SPA13  1
C******                                                                  SPA13  2
C        PURPOSE                                                         SPA13  3
C           FLSQFY CONSTRUCTS A LEAST SQUARES POLYNOMIAL APPROXIMATION   SPA13  4
C           OF SPECIFIED DEGREE TO A GIVEN SET OF DATA POINTS WITH       SPA13  5
C           GIVEN WEIGHTS USING ORTHOGONAL POLYNOMIALS.                  SPA13  6
C        USAGE                                                           SPA13  7
C           DIMENSION X(M),Y(M),W(M),S1(MN1,6),A(N+1)                    SPA13  8
C           CALL FLSQFY (M,N,X,Y,W,MN1,S1,A,IER)                         SPA13  9
C        INPUT PARAMETERS                                                SPA13 10
C           M   - NUMBER OF DATA POINTS                                  SPA13 11
C           N   - DEGREE OF POLYNOMIAL DESIRED, N.LT.M                   SPA13 12
C           X   - ARRAY OF INDEPENDENT VARIABLE                          SPA13 13
C           Y   - ARRAY OF DEPENDENT VARIABLE                            SPA13 14
C           W   - ARRAY OF POSITIVE WEIGHTS                              SPA13 15
C           MN1 - ROW DIMENSION OF SCRATCH ARRAY S1, MN1.GE.M            SPA13 16
C           S1  - SCRATCH ARRAY                                          SPA13 17
C                 IN THE COLUMN DEFINITIONS BELOW,                       SPA13 18
C                    I   REFERS TO POLYNOMIAL ORDER                      SPA13 19
C                    J   REFERS TO ROW INDEXING WITHIN A COLUMN          SPA13 20
C                 COL 1  POLYNOMIAL P(I-1), VALUE AT EACH X(J) THRU 240  SPA13 21
C                        COEFFICIENT OF EACH X**(J-1) TERM AFTER 240     SPA13 22
C                 COL 2  POLYNOMIAL P(I), SAME FORM AS COL 1             SPA13 23
C                 COL 3  ALPHA(I+1), WHERE I=J                           SPA13 24
C                 COL 4  BETA(I), WHERE I=J-1                            SPA13 25
C                 COL 5  S(I), WHERE I=J-1                               SPA13 26
C                 COL 6  SIGMA**2, WHERE I=J-1                           SPA13 27
C        OUTPUT PARAMETERS                                               SPA13 28
C           A   - ARRAY OF COMPUTED COEFFICIENTS                         SPA13 29
C                 A(1) THRU A(N+1) CONTAIN COMPUTED POLYNOMIAL           SPA13 30
C                 COEFFICIENTS, IN ORDER OF INCREASING DEGREE,           SPA13 31
C           IER - ERROR INDICATOR                                        SPA13 32
C                 =0   SUCCESS                                           SPA13 33
C                 =1   (N.GE.M),OR.(N.LE.0)                              SPA13 34
C                 =2   MN1.LT.M                                          SPA13 35
C                 =3   M.LE.1                                            SPA13 36
C                 =4   W(I).LE.0                                         SPA13 37
C        REFERENCE                                                       SPA13 38
C           FORSYTHE, G. E., GENERATION AND USE OF ORTHOGONAL            SPA13 39
C           POLYNOMIALS FOR DATA-FITTING WITH A DIGITAL COMPUTER,        SPA13 40
C           J. SIAM, VOL. 5, NO. 2, (JUNE 1957) PP. 74-88.               SPA13 41
C           NOTE - MOST NOTATION AND MOST LOCAL VARIABLE NAMES ARE       SPA13 42
C           BASED ON THE REFERENCE, IE XPP IS W(I,I)*(P(I),P(I)),        SPA13 43
C           BUT W, WT AND W(I) REFER TO THE WEIGHTS ARRAY.               SPA13 44
C******                                                                  SPA13 45
      DIMENSION X(1),Y(1),W(1),S1(MN1,1),A(1)                            SPA13 46
C                                                                        SPA13 47
C  ***  INITIAL SET-UP  ***                                             SPA13 48
C          TEST FOR USER ERRORS                                          SPA13 49
  100 IER = 4                                                            SPA13 50
      IF ((N.GE.M).OR.(N.LE.0)) IER=1                                    SPA13 51
      IF (MN1.LT.M) IER=2                                                SPA13 52
      IF (M.LE.1) IER=3                                                  SPA13 53
      IF (IER.NE.4) GO TO 990                                            SPA13 54
C          BETA(0)=0, P(0)=1, P(-1)=0, W(0,0)=SUM(W(I))                  SPA13 55
C          DELTA**2(-1)=(WF,F)                                           SPA13 56
      N1 = N+1                                                           SPA13 57
      S1(1,4)=0.                                                         SPA13 58
      DSQ=0.                                                             SPA13 59
      WPP=0.                                                             SPA13 60
      DO 110 J=1,M                                                       SPA13 61
      S1(J,2)=1.                                                         SPA13 62
      S1(J,1)=0.                                                         SPA13 63
      WT = W(J)                                                          SPA13 64
      IF (WT.LE.0.) GO TO 990                                            SPA13 65
      WPP = WPP+WT                                                       SPA13 66
  110 DSQ = DSQ+WT*Y(J)*Y(J)                                             SPA13 67
C                                                                        SPA13 68
C  ***  GENERATE ORTHOGONAL POLYNOMIALS - THRU 240  ***                  SPA13 69
      DO 240 I=1,N1                                                      SPA13 70
      WYP=0.                                                             SPA13 71
      WXPP=0.                                                            SPA13 72
C          COMPUTE (WP(I),XP(I)), AND OMEGA(I)=(P,P(I))                  SPA13 73
      DO 200 J=1,M                                                       SPA13 74
      TEMP=W(J)*S1(J,2)                                                  SPA13 75
      IF (I.LT.N1) WXPP=WXPP+TEMP*X(J)*S1(J,2)                           SPA13 76
  200 WYP=WYP+TEMP*Y(J)                                                  SPA13 77
C          S(I)=OMEGA(I)/W(I,I)                                          SPA13 78
      S1(I,5)=WYP/WPP                                                    SPA13 79
C          DELTA**2(I)=DELTA**2(I-1)*B(I)**2*W(I,I)                      SPA13 80
      DSQ=DSQ-S1(I,5)*S1(I,5)*WPP                                        SPA13 81
C          SIGMA**2(I)=DELTA**2(I)/(DEGREES OF FREEDOM)                  SPA13 82
C          SIGMA**2(M)=0                                                 SPA13 83
      RR = M-I                                                           SPA13 84
      IF (RR.NE.0) RR = DSQ/RR                                           SPA13 85
      S1(I,6) = RR                                                       SPA13 86
C          THRU 240 - CALCULATIONS FOR NEXT I, SKIP WHEN I=N+1           SPA13 87
C          ALPHA(I+1)=(WP(I),XP(I))/W(I,I)                               SPA13 88
      IF (I.GE.N1) GO TO 240                                             SPA13 89
      S1(I,3)=WXPP/WPP                                                   SPA13 90
      WPPU=WPP                                                           SPA13 91
      WPP=0.                                                             SPA13 92
      BT=S1(I,4)                                                         SPA13 93
      AL=S1(I,3)                                                         SPA13 94
C          P(I+1)=(X-ALPHA(I+1))P(I)-BETA(I)P(I-1)                       SPA13 95
C          W(I+1,I+1)=(WP(I+1),P(I+1)), B(I+1)=W(I+1,I+1)/W(I,I)         SPA13 96
      DO 230 J=1,M                                                       SPA13 97
      TEMP=(X(J)-AL)*S1(J,2)-BT*S1(J,1)                                  SPA13 98
      WPP=WPP+W(J)*TEMP**2                                               SPA13 99
      S1(J,1)=S1(J,2)                                                    SPA13100
  230 S1(J,2)=TEMP                                                       SPA13101
      S1(I+1,4)=WPP/WPPU                                                 SPA13102
  240 CONTINUE                                                           SPA13103
C                                                                        SPA13104
C  ***  COMPUTE COEFFICIENTS OF LEAST SQUARES POLYNOMIAL                SPA13105
C         A = B(0)P(0)+...+B(N)P(N)  ***                                 SPA13106
C          P(0)=1, P(-1)=0, A(INITIAL)=B(0)P(0)=B(0)                     SPA13107
      DO 300 IR=1,N1                                                     SPA13108
      A(IR) = 0.                                                         SPA13109
      S1(IR,1)=0.                                                        SPA13110
  300 S1(IR,2)=0.                                                        SPA13111
      S1(1,2)=1.                                                         SPA13112
      A(1)=S1(1,5)                                                       SPA13113
C          SUM LOOP THRU 310, I = ORDER OF POLYNOMIAL FORMED             SPA13114
      DO 310 I=1,N                                                       SPA13115
      AL=S1(I,3)                                                         SPA13116
      BT=S1(I,4)                                                         SPA13117
      T2=0.                                                              SPA13118
      II = I+1                                                           SPA13119
C          FORM P(I+1)=XP(I)-ALPHA(I+1)P(I)-BETA(I)P(I-1)                SPA13120
```


(f) Page 12.

Figure 1.- Continued.

```
C          AND ADD TO POLYNOMIAL SUM IN A                                SPA13121
      DO 510 I=1,11                                                      SPA13122
      T1=T2-AL*S1(IH,2)+HT*S1(IH,1)                                      SPA13123
      T2=S1(IH,2)                                                        SPA13124
      S1(IH,1)=S1(IH,2)                                                  SPA13125
      S1(IH,2)=T1                                                        SPA13126
  510 A(I0)=A(I0)+T1*S1(I+1,5)                                           SPA13127
      IER = 0                                                            SPA13128
  900 RETURN                                                             SPA13129
      END                                                                SPA13130
```

```
      SUBROUTINE FUSEIO                                                  SPA14  1
C                                                                        SPA14  2
C                                                                        SPA14  3
C     SUBROUTINE TO INPUT AND OUTPUT FUSELAGE DATA                       SPA14  4
C        LINE SOURCE AND DOUBLET DISTRIBUTIONS ARE CALCULATED            SPA14  5
C        USING SUBROUTINE BODYGEN                                        SPA14  6
C                                                                        SPA14  7
C                                                                        SPA14  8
      COMMON /FLO/ALFACR,GAMS,FMACH,RHO,VINF,BETA,BETASQ,FMCHSQ          SPA14  9
      COMMON /FSGEOM/FRMAX,SSPAN,   FLTHC,BODYPL                         SPA14 10
      COMMON /FSUM/FXL(101),FSS(100),FDS(100),NFSUM                      SPA14 11
      COMMON /BMPOAT/NFPOLY,FXEND(9),FCOEF(7,7)                          SPA14 12
      COMMON/INDEX/NCW,NSW,MSW,NPANLS,NCWB,NBDCR1,NBDCR2,NBD,NBIP,MP,    SPA14 13
     1   NCP,MSP,N1P,N2,N2P,NPTOT                                        SPA14 14
  701 FORMAT(1015)                                                       SPA14 15
  706 FORMAT(8F10.0)                                                     SPA14 16
  708 FORMAT(10X,19HFUSELAGE INPUT DATA)                                 SPA14 17
  709 FORMAT(15X,17HFUSELAGE LENGTH =,F10.5,5H FEET/15X,16HMAXIMUM RADIUSSPA14 18
     1S =,F10.5,5H FEET/)                                                SPA14 19
  729 FORMAT(   15X,37HPOLYNOMIALS SPECIFYING FUSELAGE SHAPE             SPA14 20
     1 /18X,20HX/L OF END OF EACH SECTION/21X,7HSECTION,5X,3HR/L)        SPA14 21
  730 FORMAT(23X,I2,3X,7F10.5)                                           SPA14 22
  731 FORMAT(18X,51HCOEFFICIENTS OF POLYNOMIALS DESCRIBING EACH SECTION/SPA14 23
     121X,7HSECTION,5X,2HC1,8X,2HC2,8X,2HC3,8X,2HC4,8X,2HC5,8X,2HC6,8X, SPA14 24
     22HC7)                                                              SPA14 25
  732 FORMAT(/14X,I3,  85H RINGS OF U-VELOCITY PANELS ARE TO BE LAID OUTSPA14 26
     1 ON THE FUSELAGE FROM X/L= 0 FEET TO X/L=,F10.5,5H FEET/20X,9HTHERESPA14 27
     2ARE,I3,54H PANELS IN EACH RING ON THE LEFT HALF OF THE FUSELAGE,, SPA14 28
     313,19H ABOVE THE WING AND,I3,15H BELOW THE WING/20X,25HTOTAL NUMBESPA14 29
     4R OF PANELS IS,I4)                                                 SPA14 30
  733 FORMAT(/15X,105HFUSELAGE SHAPE AS CALCULATED FROM THE INPUT POLYNOSPA14 31
     1MIALS, ORIGINS OF SOURCES AND DOUBLETS REPRESENTING THE/20X,56HFUSESPA14 32
     2ELAGE, AND VALUES OF THE SOURCE AND DOUBLET CONSTANTS)             SPA14 33
C                                                                        SPA14 34
C     READ IN AND WRITE FUSELAGE GEOMETRY AND POLYNOMIALS                SPA14 35
C        DESCRIBING SHAPE                                                SPA14 36
C                                                                        SPA14 37
      READ (5,706) FLTHC,FRMAX                                           SPA14 38
      WRITE (6,708)                                                      SPA14 39
      WRITE (6,709) FLTHC,FRMAX                                          SPA14 40
      READ (5,701) NFPOLY                                                SPA14 41
      READ (5,706) (FXEND(J),J=1,NFPOLY)                                 SPA14 42
      DO 3 J=1,NFPOLY                                                    SPA14 43
    3 READ (5,706) (FCOEF(J,K),K=1,7)                                    SPA14 44
      WRITE (6,729)                                                      SPA14 45
      DO 4 J=1,NFPOLY                                                    SPA14 46
    4 WRITE (6,730) J,FXEND(J)                                           SPA14 47
      WRITE (6,731)                                                      SPA14 48
      DO 5 J=1,NFPOLY                                                    SPA14 49
    5 WRITE (6,730) J,(FCOEF(J,K),K=1,7)                                 SPA14 50
C                                                                        SPA14 51
C     READ AND WRITE FUSELAGE PANEL DATA                                 SPA14 52
C                                                                        SPA14 53
      READ (5,701) NCW,NBDCR1,NBDCR2,NFSUM                               SPA14 54
      READ (5,706) BODYPL                                                SPA14 55
      NBD=NBDCR1+NBDCR2                                                  SPA14 56
      NPTOT=NCW+NBD                                                      SPA14 57
      BODYPL=ABS(BODYPL)                                                 SPA14 58
      BHF=BODYPL                                                         SPA14 59
      WRITE (6,732) NCW,MW,NBD,NBDCR1,NBDCR2,NPTOT                       SPA14 60
C                                                                        SPA14 61
C     CALCULATE LINE SOURCE AND DOUBLET DISTRIBUTIONS                    SPA14 62
C                                                                        SPA14 63
      NXBODY=NFSUM+1                                                     SPA14 64
      WRITE (6,733)                                                      SPA14 65
      CALL BDYGEN(NXBODY,FRMAX,FLTHC,NFPOLY,FXEND,FCOEF,FSS,FDS,FXL,     SPA14 66
     1            ALFACR)                                                SPA14 67
      RETURN                                                             SPA14 68
      END                                                                SPA14 69
```

```
      SUBROUTINE INTOST (XI,ETA,ZETA,X,Y,Z,DC)                           SPA15  1
C     SUBROUTINE TO TRANSFORM FROM INERTIAL TO STUNE SYSTEM              SPA15  2
C                                                                        SPA15  3
      DIMENSION DC(3,3)                                                  SPA15  4
      X=XI*DC(1,1)+ETA*DC(2,1)+ZETA*DC(3,1)                              SPA15  5
      Y=XI*DC(1,2)+ETA*DC(2,2)+ZETA*DC(3,2)                              SPA15  6
      Z=XI*DC(1,3)+ETA*DC(2,3)+ZETA*DC(3,3)                              SPA15  7
      RETURN                                                             SPA15  8
      END                                                                SPA15  9
```

```
      SUBROUTINE INVERS(A,NSYS,N,NMAX,MMAX)                              SPA16  1
C     SUBROUTINE TO SOLVE SIMULTANEOUS EQUATIONS                         SPA16  2
C                                                                        SPA16  3
      DIMENSION A(NMAX,MMAX),X(200)                                      SPA16  4
    5 SIGN=1.0                                                           SPA16  5
      NPI=N+1                                                            SPA16  6
      NMI=N-1                                                            SPA16  7
      NPLSY=N+NSYS                                                       SPA16  8
    9 DO 14 I=1,NMI                                                      SPA16  9
      IPI=I+1                                                            SPA16 10
      MAX=I                                                              SPA16 11
      AMAX=ABS(A(I,I))                                                   SPA16 12
      DO 10 K=IPI,N                                                      SPA16 13
      AKMAX=ABS(A(K,I))                                                  SPA16 14
      IF(AKMAX.LE.AMAX) GO TO 10                                         SPA16 15
      MAX=K                                                              SPA16 16
      AMAX=AKMAX                                                         SPA16 17
   10 CONTINUE                                                           SPA16 18
      IF(AMAX.LT.1.0E-12) GO TO 16                                       SPA16 19
      IF(MAX.EQ.I) GO TO 12                                              SPA16 20
      DO 11 L=I,NPLSY                                                    SPA16 21
      TEMP=A(I,L)                                                        SPA16 22
      A(I,L)=A(MAX,L)                                                    SPA16 23
   11 A(MAX,L)=TEMP                                                      SPA16 24
      SIGN=-SIGN                                                         SPA16 25
   12 DO 14 J=IPI,N                                                      SPA16 26
      IF (A(J,I)) 30,14,30                                               SPA16 27
   30 CONST=-A(J,I)/A(I,I)                                               SPA16 28
      DO 13 L=I,NPLSY                                                    SPA16 29
   13 A(J,L)=A(J,L)+A(I,L)*CONST                                         SPA16 30
   14 CONTINUE                                                           SPA16 31
      DO 15 I=1,N                                                        SPA16 32
      IF (A(I,I)) 15,16,15                                               SPA16 33
   15 CONTINUE                                                           SPA16 34
      GO TO 18                                                           SPA16 35
   16 WRITE(6,100)                                                       SPA16 36
  100 FORMAT(///5X,24H***** MATRIX IS SINGULAR)                          SPA16 37
      STOP                                                               SPA16 38
   18 DO 21 I=NPI,NPLSY                                                  SPA16 39
      DO 20 K=1,N                                                        SPA16 40
```


(m) Page 13.

Figure 1.- Continued.

```
      K=NPI-KK                                                          SPA16 41
      X(K)=A(K,I)                                                       SPA16 42
      IF(K.EQ.N) GO TO 20                                               SPA16 43
      J=K                                                               SPA16 44
   19 J=J+1                                                             SPA16 45
      X(K)=X(K)-A(K,J)*X(J)                                             SPA16 46
      IF(J.LT.N) GO TO 19                                               SPA16 47
   20 X(K)=X(K)/A(K,K)                                                  SPA16 48
      DO 21 J=1,N                                                       SPA16 49
   21 A(J,I)=X(J)                                                       SPA16 50
      RETURN                                                            SPA16 51
      END                                                               SPA16 52
      SUBROUTINE MACH (Y,Z)                                             SPA17  1
C                                                                       SPA17  2
C     THIS SUBROUTINE DETERMINES A LOCAL MACH NUMBER BASED ON THE FIRST SPA17  3
C     SET OF NONZERO FLOW VELOCITIES BEHIND THE FREE STREAM MACH CONE   SPA17  4
C     ASSOCIATED WITH WING THICKNESS.                                   SPA17  5
C                                                                       SPA17  6
C     NOTE: X,Y,Z IN WING COORDINATE SYSTEM                             SPA17  7
C                                                                       SPA17  8
C     THIS ROUTINE ALSO CALCULATES THE VELOCITIES BETWEEN XT3 AND XT4   SPA17  9
C     TO BE SMOOTHED BY LEAST SQUARES. THESE ARE THE SUM OF             SPA17 10
C          (A) FUSELAGE SOURCE AND DOUBLETS                             SPA17 11
C          (B) WING PANELS                                              SPA17 12
C          (C) BODY PANELS                                              SPA17 13
C          (D) WING THICKNESS                                           SPA17 14
C                                                                       SPA17 15
      COMMON /CONFIG/NFU,NPY,NSTRS,LVS=P                                SPA17 16
      COMMON /FLOW/DUM6(2),FMACH,DUM7(2),BETA,DUM8(2)                   SPA17 17
      COMMON /FSOR/FXL(101),FSS(100),FDS(100),NFSUM                     SPA17 18
      COMMON /IGVEL/UP,VP,WP,II,IF,DUM9(200)                            SPA17 19
      COMMON/INDEX/NCW,MSW,MS=P,NPANLS,NCWB,NBDCR1,NBDCR2,NBD,NBIP,MP,  SPA17 20
     1   NCP,MSP,N1P,N2,N2P,NPTOT                                       SPA17 21
      COMMON/NUMLIM/SNUMCH,BETANU,INUMCH,IBMTH,XT3,XT4                  SPA17 22
      COMMON /THRDAT/NCWB,NCPB,MB,DUM1(5002)                            SPA17 23
      COMMON /WGEOM/XBW(UC,ZBW(U,CRW,DUM9(62)                           SPA17 24
      COMMON /WPGEOM/XWF(200),XRB(200),XLP(200),XLB(200),YHC(200),      SPA17 25
     1 YLC(200),DUM4(600),S=PPLE(200),S=PPTE(200)                       SPA17 26
      COMMON /WSMOTH/ NUMDWS,WSCUE(6,3),XSMFTW                          SPA17 27
      DIMENSION XBW(20),VEL(20,3),W(20),B(20,6)                         SPA17 28
C                                                                       SPA17 29
      DATA W/20*1.0/                                                    SPA17 30
C                                                                       SPA17 31
  500 FORMAT(///5X58H***** AXIAL LOCATION OF TRAVERSE ABOVE WING CHORDAL SPA17 32
     1 PLANE)                                                           SPA17 33
  501 FORMAT(//5X,94H***** NO NEGATIVE VELOCITY DUE TO WING THICKNESS FS SPA17 34
     1OUND IN TRAVERSE TO TRAILING EDGE MACH CONE)                      SPA17 35
C                                                                       SPA17 36
      IF(Z.GT.0.0) GO TO 5                                              SPA17 37
      WRITE(6,500)                                                      SPA17 38
      STOP                                                              SPA17 39
C                                                                       SPA17 40
C     CALCULATE LOCAL WING LE X COORDINATE                              SPA17 41
C                                                                       SPA17 42
    5 YY=Y                                                              SPA17 43
      IF(YY.GT.0.) YY=-YY                                               SPA17 44
      IF(YY.GE.YLC(1)) GO TO 15                                         SPA17 45
      DO 10  I=2,MSP                                                    SPA17 46
      JLE=(I-2)*NCW+1                                                   SPA17 47
      IF(YY.GE.YLC(JLE)) GO TO 20                                       SPA17 48
   10 CONTINUE                                                          SPA17 49
C                                                                       SPA17 50
C     YY LESS THAN Y OF WING TIP, USE TIP                               SPA17 51
C     CHORD FOR CHRD.                                                   SPA17 52
C                                                                       SPA17 53
      XLE=XLE(JLE)                                                      SPA17 54
      JTE=JLE+NCW-1                                                     SPA17 55
      XTE=XLE(JTE)                                                      SPA17 56
      GO TO 25                                                          SPA17 57
C                                                                       SPA17 58
C     YY GE RIGHT SIDE OF FIRST CHORDWISE NUM.                          SPA17 59
C     USE ROOT-CHORD FOR CHRD.                                          SPA17 60
C                                                                       SPA17 61
   15 XLE=0.0                                                           SPA17 62
      XTE=CRW                                                           SPA17 63
      GO TO 25                                                          SPA17 64
C                                                                       SPA17 65
C     YY BRACKETED.  INTERPOLATE AND COMPUTE CHRD.                      SPA17 66
C                                                                       SPA17 67
   20 DELT=YY-YHC(JLE)                                                  SPA17 68
      XLE=XRF(JLE)+DELT*S=PPLE(JLE)                                     SPA17 69
      JTE=JLE+NCW-1                                                     SPA17 70
      XTE=XRB(JTE)+DELT*S=PPTE(JTE)                                     SPA17 71
   25 CHRD=XLE-XTE                                                      SPA17 72
      INUMCH=0                                                          SPA17 73
C                                                                       SPA17 74
      II=1                                                              SPA17 75
      IF=MB                                                             SPA17 76
C                                                                       SPA17 77
C     HUNT FOR BEGINNING OF WING THICKNESS VELOCITY FIELD, FIRST USING  SPA17 78
C          THE FREE-STREAM MACH NUMBER AND THEN USING THE CALCULATED    SPA17 79
C          LOCAL MACH NUMBER                                            SPA17 80
C                                                                       SPA17 81
  109 DELX1=0.1*CHRD                                                    SPA17 82
      WSAVE=0.0                                                         SPA17 83
      X=0.0                                                             SPA17 84
      XSAVE=0.0                                                         SPA17 85
      B=BETA                                                            SPA17 86
      IF(INUMCH.EQ.1) B=BETANU                                          SPA17 87
      AN=(XLE-CHRD-Z*B)/DELX1                                           SPA17 88
C                                                                       SPA17 89
C     SEARCH FOR FIRST NEGATIVE W VELOCITY USING DELTA X=0.1*CHORD      SPA17 90
C                                                                       SPA17 91
      DO 102 K=1,N                                                      SPA17 92
      X=X+DELX1                                                         SPA17 93
      CALL VELWTH (X,Y,Z)                                               SPA17 94
      WP=WP                                                             SPA17 95
      IF(WP.GE.0.0) GO TO 102                                           SPA17 96
      WSAVE=WP                                                          SPA17 97
      XSAVE=X                                                           SPA17 98
      GO TO 103                                                         SPA17 99
  102 CONTINUE                                                          SPA17100
      WRITE(6,501)                                                      SPA17101
      STOP                                                              SPA17102
C                                                                       SPA17103
C     BACK UP 1 STEP AND REPEAT SEARCH USING ONE THIRD OF INTERVAL      SPA17104
C                                                                       SPA17105
  103 CONTINUE                                                          SPA17106
      X=XSAVE+DELX1                                                     SPA17107
      DELX1=DELX1/3.0                                                   SPA17108
      DO 104 K=1,2                                                      SPA17109
      X=X+DELX1                                                         SPA17110
      CALL VELWTH(X,Y,Z)                                                SPA17111
      WP=WP                                                             SPA17112
      IF (WP.GE.0.0) GO TO 104                                          SPA17113
      XSAVE=X                                                           SPA17114
      WSAVE=WP                                                          SPA17115
      GO TO 106                                                         SPA17116
  104 CONTINUE                                                          SPA17117
C                                                                       SPA17118
C     BACK UP 1 STEP AND REPEAT USING 1/9TH ORIGINAL STEP SIZE          SPA17119
C                                                                       SPA17120
  106 X=XSAVE+DELX1                                                     SPA17121
      DELX1=DELX1/3.0                                                   SPA17122
      DO 105 K=1,2                                                      SPA17123
      X=X+DELX1                                                         SPA17124
      CALL VELWTH (X,Y,Z)                                               SPA17125
      WP=WP                                                             SPA17126
      IF (WP.GE.0.0) GO TO 105                                          SPA17127
```

(n) Page 14.

Figure 1.- Continued.

```
      XSAVE=X                                                           SPA17128
      MSAVE=MP                                                          SPA17129
      GO TO 107                                                         SPA17130
  105 CONTINUE                                                          SPA17131
C                                                                       SPA17132
C     SET XT3 EQUAL TO XSAVE                                            SPA17133
C                                                                       SPA17134
  107 XT3=XSAVE                                                         SPA17135
C                                                                       SPA17136
C     CALCULATE FLOW ANGLE TNU AT (XT3,Y,Z) POSITION IF INUMCH=0        SPA17137
C                                                                       SPA17138
      IF(INUMCH.EQ.1) GO TO 108                                         SPA17139
      CALL VEL=TM(XT3,Y,Z)                                              SPA17140
      UP=1.0=UP                                                         SPA17141
      MP=MP                                                             SPA17142
      TANNU= (SQRT(VP=VP+WP=WP))/UP                                     SPA17143
      TNU=ATAN(TANNU)=57.2957795                                        SPA17144
      ANGNU1=2.4049=ATAN(0.40825=BETA)=ATAN(BETA)                       SPA17145
      ANGNU1=ANGNU1=57.2957795                                          SPA17146
      ANGNU2=ANGNU1=TNU                                                 SPA17147
C                                                                       SPA17148
C     CALCULATE LOCAL MACH NUMBER DUE TO ISENTROPIC COMPRESSION AT      SPA17149
C     (XT3,Y,Z) POSITION USING I.M.HALL METHOD (AERO, JOUR, SEPT 75)    SPA17150
C     NOTE:THIS NEW MACH NUMBER IS TAKEN IN X DIRECTION                 SPA17151
C                                                                       SPA17152
      YNU=(ANGNU2/130.4540769)**0.6666666667                            SPA17153
      YNUSQ=YNU=YNU                                                     SPA17154
      YNUC=YNU=YNUSQ                                                    SPA17155
      FNUMCH=((1.0+1.3004=YNU+0.0962=YNUSQ-0.5127=YNUC)/(1.0+0.6722=YNU SPA17156
     1 +0.5270=YNUSQ))=COS(TNU/57.2957795)                              SPA17157
      INUMCH=1                                                          SPA17158
      BETANU=SQRT(FNUMCH=FNUMCH-1.0)                                    SPA17159
C                                                                       SPA17160
C     GO BACK AND REPEAT SEARCH FOR XT3 USING NEW MACH NUMBER           SPA17161
C                                                                       SPA17162
      GO TO 109                                                         SPA17163
C                                                                       SPA17164
C     DETERMINE DOWNSTREAM POINT WHERE M VELOCITY DUE TO WING THICKNESS SPA17165
C          JUMPS BACK TO A VALUE NEAR ZERO                              SPA17166
C                                                                       SPA17167
  108 CONTINUE                                                          SPA17168
      DELX1=0.1=CHRD                                                    SPA17169
      X=XT3-0.7=CHRD                                                    SPA17170
      CALL VEL=TM (X,Y,Z)                                               SPA17171
      MP=MP                                                             SPA17172
      MSAVE=MP                                                          SPA17173
      XSAVE=X                                                           SPA17174
      K=0                                                               SPA17175
  110 X=X-DELX1                                                         SPA17176
      K=K+1                                                             SPA17177
      CALL VEL=TM(X,Y,Z)                                                SPA17178
      MP=MP                                                             SPA17179
      IF (MP.GT.MSAVE) GO TO 111                                        SPA17180
      DELT=ABS(MP-MSAVE)                                                SPA17181
      IF(DELT.GE.(0.50=MSAVE)) GO TO 120                                SPA17182
  111 MSAVE=MP                                                          SPA17183
      XSAVE=X                                                           SPA17184
      GO TO 110                                                         SPA17185
C                                                                       SPA17186
C     BACK UP 1 STEP AND REPEAT SEARCH USING 1/3 OF INTERVAL            SPA17187
C                                                                       SPA17188
  120 X=XSAVE                                                           SPA17189
      DELX1=DELX1/3.0                                                   SPA17190
      DO 121 K=1,2                                                      SPA17191
      X=X-DELX1                                                         SPA17192
      CALL VEL=TM(X,Y,Z)                                                SPA17193
      MP=MP                                                             SPA17194
      IF (MP.GT.MSAVE) GO TO 122                                        SPA17195
      DELT=ABS(MP-MSAVE)                                                SPA17196
      IF(DELT.GE.(0.50=MSAVE)) GO TO 130                                SPA17197
  122 MSAVE=MP                                                          SPA17198
      XSAVE=X                                                           SPA17199
  121 CONTINUE                                                          SPA17200
C                                                                       SPA17201
C     BACK UP 1 STEP AND REPEAT SEARCH USING 1/9TH OF INTERVAL          SPA17202
C                                                                       SPA17203
  130 X=XSAVE                                                           SPA17204
      DELX1=DELX1/3.0                                                   SPA17205
      DO 131 K=1,2                                                      SPA17206
      X=X-DELX1                                                         SPA17207
      CALL VEL=TM(X,Y,Z)                                                SPA17208
      MP=MP                                                             SPA17209
      IF (MP.GT.MSAVE) GO TO 132                                        SPA17210
      DELT=ABS(MP-MSAVE)                                                SPA17211
      IF(DELT.GE.(0.50=MSAVE)) GO TO 140                                SPA17212
  132 MSAVE=MP                                                          SPA17213
      XSAVE=X                                                           SPA17214
  131 CONTINUE                                                          SPA17215
C                                                                       SPA17216
C     SET XT4 EQUAL TO XSAVE                                            SPA17217
C                                                                       SPA17218
  140 XT4=XSAVE                                                         SPA17219
C                                                                       SPA17220
C     CALCULATE VELOCITIES TO BE SMOOTHED                               SPA17221
C                                                                       SPA17222
      NPTSS=NCM                                                         SPA17223
      FNC=NCM                                                           SPA17224
      DELX=(XT4-XT3)/FNCM                                               SPA17225
      XX=XT3-DELX/2.0                                                   SPA17226
      NORD=3                                                            SPA17227
      DO 200 N=1,NPTSSM                                                 SPA17228
      XX=XX+DELX                                                        SPA17229
      XSM(N)=XX                                                         SPA17230
      UTU=0.0                                                           SPA17231
      VTV=0.0                                                           SPA17232
      WTW=0.0                                                           SPA17233
      II=1                                                              SPA17234
      IF=NPANLS                                                         SPA17235
      CALL VELMP(XX,Y,Z)                                                SPA17236
      UTU=UTU+UP                                                        SPA17237
      VTV=VTV+VP                                                        SPA17238
      WTW=WTW+WP                                                        SPA17239
      IF=MS                                                             SPA17240
      CALL VEL=TM(XX,Y,Z)                                               SPA17241
      UTU=UTU+UP                                                        SPA17242
      VTV=VTV+VP                                                        SPA17243
      WTW=WTW+WP                                                        SPA17244
      IF (NFU.EQ.0) GO TO 210                                           SPA17245
      II=NZP                                                            SPA17246
      IF=NPTUT                                                          SPA17247
      CALL VELBD(XX,Y,Z)                                                SPA17248
      UTU=UTU+UP                                                        SPA17249
      VTV=VTV+VP                                                        SPA17250
      WTW=WTW+WP                                                        SPA17251
      XB=XX+XBHUC                                                       SPA17252
      ZB=Z+ZBHO                                                         SPA17253
      CALL VELCAL(FSS,FDS,FXL,NFSUN,XB,Y,ZH,UP,VP,WP)                   SPA17254
      UTU=UTU+UP                                                        SPA17255
      VTV=VTV+VP                                                        SPA17256
      WTW=WTW+WP                                                        SPA17257
  210 CONTINUE                                                          SPA17258
      VEL(N,1)=UTU                                                      SPA17259
      VEL(N,2)=VTV                                                      SPA17260
      VEL(N,3)=WTW                                                      SPA17261
  200 CONTINUE                                                          SPA17262
C                                                                       SPA17263
C     TRANSFORM X COORDINATE SO ORIGIN IS CENTERED IN XT3, XT4 RANGE    SPA17264
C                                                                       SPA17265
      XSHFT=0.5=(XT3+XT4)                                               SPA17266
      DO 215 N=1,NPTSSM                                                 SPA17267
  215 XSM(N)=XSM(N)-XSHFT                                               SPA17268
C                                                                       SPA17269
C     CALL LEAST SQUARES ROUTINE FOR THREE VELOCITY COMPONENTS          SPA17270
C                                                                       SPA17271
```

(o) Page 15.

Figure 1.- Continued.

```
      DO 21A N=1,5
  21A CALL PLSWRY(NPTS5M,NUMDWS,XSM,VEL(1,N),M,26,5,MSCDE(1,N),IER)
      RETURN
      END                                                              SPA17272
                                                                       SPA17273
                                                                       SPA17274
                                                                       SPA17275
      SUBROUTINE PLYOUT
C                                                                      SPA18  1
C     SUBROUTINE TO READ AND PRINT PYLON DATA AND LAY OUT ELEMENTAL    SPA18  2
C     PANELS ON THE PYLON                                              SPA18  3
C     PANEL COORDINATES ARE IN WING SYSTEM                             SPA18  4
C                                                                      SPA18  5
C     CORNER ORIENTATIONS  RIGHT SIDE IS CLOSER TO THE WING            SPA18  6
C                                                                      SPA18  7
      COMMON/BPGEOM/                          SNT2(100),CST2(100),ZLC(200),SPA18  8
     1   ZRC(200),THTI(100),SANGM,CANGM                                SPA18  9
      COMMON /CONSTS/PI,PI2,DTOR,RTOD,FOURPI                           SPA18 10
      COMMON/INDEX/NCW,MSW,MSWP,NPANLS,NCWH,NBDCH1,NBDCH2,NHD,NBIP,MP, SPA18 11
     1   NCP,MSP,N1P,N2,N2P,NPTOT                                      SPA18 12
      COMMON /PYGEOM/Z(20),XPLE,YPL,CRP,MP,MSIPLE,IP,SLLE,SLTE,MSLPDF, SPA18 13
     1      CENTER                                                     SPA18 14
      COMMON /WGEOM/XBMAC,ZBMAC,CHW,SLPALE,SLPATE,PSILE(20),PSITE(20), SPA18 15
     1   Y(20)                                                         SPA18 16
      COMMON/BPGEOM/XRF(200),XRB(200),XLF(200),XLB(200),YRC(200),      SPA18 17
     1   YLC(200),XLPT(200),YCPT(200),ZCPT(200),SMPPLE(200),SMPPTE(200) SPA18 18
C                                                                      SPA18 19
  701 FORMAT(1015)                                                     SPA18 20
  703 FORMAT(I5,7F10.0)                                                SPA18 21
  704 FORMAT(1H1,10X,16HPYLON INPUT DATA)                              SPA18 22
  705 FORMAT(/15X,41HLEADING EDGE OF PYLON ROOT CHORD IS AT XM=,F10.5,4SMSPA18 24
     1 FEET (MEASURED FROM LOCAL WING LEADING EDGE)/15X,23HSPANWISE LOCASPA18 25
     2TION IS YM=,F10.5,5H FEET/15X,11HROOTCHORD =,F10.5,5H FEET/15X,8HMESPA18 26
     3IGHT =,F10.5,5H FEET/15X,20HLEADING EDGE SWEEP ANGLE =,F10.5,8H DESPA18 27
     4GREES/15X,27HTRAILING EDGE SWEEP ANGLE =,F10.5,8H DEGREES)       SPA18 28
  706 FORMAT(/14X,I4,  50H U-VELOCITY PANELS ARE TO BE LAID OUT ON THE PSPA18 29
     1YLON/20X,                                                        SPA18 30
     2  I3,20H CHORDWISE ROWS WITH,I3,12H IN EACH ROW)                 SPA18 31
  709 FORMAT(/15X,38HSPANWISE LOCATIONS OF PANEL SIDE EDGES/20X,1HN,9X, SPA18 32
     1 1HZ/2RX,4HFEET)                                                 SPA18 33
  710 FORMAT(18X,I3,F14.5)                                             SPA18 34
C                                                                      SPA18 35
      DATA FAC/0.95/                                                   SPA18 36
C                                                                      SPA18 37
C     READ PYLON DATA                                                  SPA18 38
C                                                                      SPA18 39
      READ(5,703) IP,PSIPLE,PBIPTE,CRP,MP,XPLE                        SPA18 40
      WRITE(6,704)                                                     SPA18 41
      YPL=0.0                                                          SPA18 42
      IF(IP.GT.0) YPL=Y(IP)                                            SPA18 43
      WRITE(6,705) XPLE,YPL,CRP,MP,PSIPLE,PBIPTE                       SPA18 44
      READ(5,701) NCP,MSP                                              SPA18 45
      NP=NCP*MSP                                                       SPA18 46
      WRITE(6,706) MP,MSP,NCP                                          SPA18 47
      KMAX=MSP+1                                                       SPA18 48
      DO 1 J=1,KMAX                                                    SPA18 49
    1 READ(5,703) K,Z(K)                                               SPA18 50
      WRITE (6,709)                                                    SPA18 51
      DO 2 K=1,KMAX                                                    SPA18 52
    2 WRITE (6,710) K,Z(K)                                             SPA18 53
C                                                                      SPA18 54
C     LAY OUT ELEMENTAL PANELS                                         SPA18 55
C                                                                      SPA18 56
      SLLE=TAN(PSIPLE*DTOR)                                            SPA18 57
      SLTE=TAN(PBIPTE*DTOR)                                            SPA18 58
      PSLPDF=SLLE-SLTE                                                 SPA18 59
      PLEX=XPLE                                                        SPA18 60
      INDX=(IP+2)*NCW+1                                                SPA18 61
      IF(IP.GT.1) PLEX=XPLE+XLF(INDX)                                  SPA18 62
      ANCP=NCP                                                         SPA18 63
      CSIDE=CRP                                                        SPA18 64
C                                                                      SPA18 65
C     LOOP OVER CHORDWISE ROWS OF PANELS                               SPA18 66
C                                                                      SPA18 67
      DO 100 I=2,KMAX                                                  SPA18 68
      CSIDEP=CSIDE                                                     SPA18 69
      CSIDE=CRP*(Z(I)-Z(I))*PSLPDF                                     SPA18 70
      IM=I-1                                                           SPA18 71
      IF(I.EQ.2) GO TO 30                                              SPA18 72
      JLE=(I-3)*NCP+1+NPANLS                                           SPA18 73
      PLEX=XLF(JLE)                                                    SPA18 74
   30 CONTINUE                                                         SPA18 75
C                                                                      SPA18 76
C     LOOP OVER ELEMENTAL PANELS IN THIS CHORDWISE ROW                 SPA18 77
C                                                                      SPA18 78
      DO 130 K=1,NCP                                                   SPA18 79
      J=(I-2)*NCP+K+NPANLS                                             SPA18 80
      AKM=K-1                                                          SPA18 81
      AK=K                                                             SPA18 82
      SMPPLE(J)=SLLE-AKM*PSLPDF/ANCP                                   SPA18 83
      SMPPTE(J)=SLLE-AK*PSLPDF/ANCP                                    SPA18 84
C                                                                      SPA18 85
C     CORNER POINT COORDINATES                                         SPA18 86
C                                                                      SPA18 87
      XRF(J)=AKM*CSIDEP/ANCP+PLEX                                      SPA18 88
      XRB(J)=XRF(J)+CSIDEP/ANCP                                        SPA18 89
      XLF(J)=AKM*CSIDE/ANCP+(Z(IM)-Z(I))*SLLE+PLEX                     SPA18 90
      XLB(J)=XLF(J)+CSIDE/ANCP                                         SPA18 91
      YLC(J)=YPL                                                       SPA18 92
      YRC(J)=YPL                                                       SPA18 93
      ZRC(J)=Z(IM)                                                     SPA18 94
      ZLC(J)=Z(I)                                                      SPA18 95
C                                                                      SPA18 96
C     Z-COORDINATES OF CONTROL POINT AND PANEL CENTROID                SPA18 97
C                                                                      SPA18 98
      A1=XRB(J)-XRF(J)                                                 SPA18 99
      A2=XLB(J)-XLF(J)                                                 SPA18100
      H=Z(I)-Z(IM)                                                     SPA18101
      ZBAR=(2.0*A1+A2)*H/(3.0*(A1+A2))                                 SPA18102
      ZCPT(J)=Z(IM)+ZBAR                                               SPA18103
      YCPT(J)=YPL                                                      SPA18104
C                                                                      SPA18105
C     X-COORDINATES OF CONTROL POINT                                   SPA18106
C                                                                      SPA18107
      XPCLE=XRF(J)+ZBAR*SMPPLE(J)                                      SPA18108
      XPCTE=XRB(J)+ZBAR*SMPPTE(J)                                      SPA18109
      PNLC=XPCLE-XPCTE                                                 SPA18110
  130 XCPT(J)=XPCLE+FAC*PNLC                                           SPA18111
  100 CONTINUE                                                         SPA18112
      RETURN                                                           SPA18113
      END                                                              SPA18114
      SUBROUTINE PYTHIN
C                                                                      SPA19  1
C     SUBROUTINE TO INPUT PYLON THICKNESS DATA                         SPA19  2
C                                                                      SPA19  3
      COMMON/INDEX/NCW,MSW,MSWP,NPANLS,NCWH,NBDCH1,NBDCH2,NHD,NBIP,MP, SPA19  4
     1   NCP,MSP,N1P,N2,N2P,NPTOT                                      SPA19  5
      COMMON /THKDAT/NCWS,NCPS,NS,MPS  ,NTHP,XWFT(400),FHHT(400),      SPA19  6
     1   XLFT(400),XLHT(400),YRCT(400),YLCT(400),ZRCT(400),ZLCT(400),  SPA19  7
     2   THETAL(400),THETPL(200),SLLFT(400),SLTET(400),DZCF(400)       SPA19  8
C                                                                      SPA19  9
  701 FORMAT(1015)                                                     SPA19 10
  702 FORMAT(8F10.0)                                                   SPA19 11
  703 FORMAT(//14X,I4,49H THICKNESS PANELS ARE TO BE LAID OUT ON THE PYLSPA19 12
     1ON/20X,I3,20H CHORDWISE ROWS WITH,I3,12H IN EACH ROW/21X,43HTHE CHSPA19 13
     2ORDWISE ROWS COINCIDE WITH THOSE USED/21X,                       SPA19 14
                                                                       SPA19 15
```


(p) Page 16.

Figure 1.- Continued.

```
      5   (5X,FOR THE U-VELOCITY PANELS//)                                  SPA19  16
C                                                                           SPA19  17
      READ (5,701)NCPS,NUNIP                                                SPA19  18
      MPS=NCPS*MSP                                                          SPA19  19
      WRITE(6,703)MKS,MSP,NCPS                                              SPA19  20
      IF(NUNIP .NE. 0)GO TO 5                                               SPA19  21
      MN=0                                                                  SPA19  22
      DO 1 JNP=1,MPS,NCPS                                                   SPA19  23
      MN=MN+NCPS                                                            SPA19  24
    1 READ(5,702) (THETPL(JP),JP=JNP,MN)                                    SPA19  25
      RETURN                                                                SPA19  26
    5 READ(5,702) (THETPL(JP),JP=1,NCPS)                                    SPA19  27
      DO 6 J=2,MSP                                                          SPA19  28
      JJ=(J-1)*NCPS                                                         SPA19  29
      DO 6 K=1,NCPS                                                         SPA19  30
      KK=JJ+K                                                               SPA19  31
    6 THETPL(KK)=THETPL(K)                                                  SPA19  32
      RETURN                                                                SPA19  33
      END                                                                   SPA19  34
```

```
      SUBROUTINE RESVEL(XB,YB,ZB,UTU,VTV,WTW)                               SPA20   1
C                                                                           SPA20   2
C     SUBROUTINE TO CALCULATE RESULTANT VELOCITIES AT POINT (XB,YB,ZB)      SPA20   3
C     DUE TO ALL AIRCRAFT COMPONENTS EXCEPT SEPARATED STORE                 SPA20   4
C     COORDINATES XB,YB,ZB AND RETURNED VELOCITIES,UTU,VTV,WTW,             SPA20   5
C     ARE IN FUSELAGE SYSTEM                                                SPA20   6
C                                                                           SPA20   7
      COMMON /CONFIG/NFU,NPY,NSTHS,LVSWP                                    SPA20   8
      COMMON /FSOR/FXL(101),FSS(100),FDS(100),NFSOR                         SPA20   9
      COMMON/ICVEL/UP,VP,WP,II,IF,          DELTP(200)                      SPA20  10
      COMMON/INDEX/NCW,MSW,MSWP,NPANLS,NCWB,NBDCR1,NBDCR2,NBD,NBIP,MP,      SPA20  11
     1   NCP,MSP,N1P,N2,N2P,NPTOT                                           SPA20  12
      COMMON /TWDAT/NCWB,NCPS,MS,NPS  ,NTWP,XRFT(400),XRBT(400),            SPA20  13
     1   XLFT(400),XLBT(400),YRCT(400),YLCT(400),ZRCT(400),ZLCT(400),       SPA20  14
     2   THETAL(400),THETPL(200),SLLFT(400),SLTET(400),DZDX(400)            SPA20  15
      COMMON /WGEOM/XBWOC,ZBWO,CRW,SLPWLE,SLPWTE,PSIWLE(20),PSIWTE(20),     SPA20  16
     1      DUMY(20)                                                        SPA20  17
      COMMON/NUFLOW/FNUMCH,BETANU,INUMCH,ISMTH,XT3,XT4                      SPA20  18
      COMMON /WSMOTN/ NORDWS,WSCOE(6,3),XSWFTW                              SPA20  19
C                                                                           SPA20  20
C                                                                           SPA20  21
C     TRANSLATE COORDINATES TO WING SYSTEM                                  SPA20  22
C                                                                           SPA20  23
      XW=XB-XBWOC                                                           SPA20  24
      YW=YB                                                                 SPA20  25
      ZW=ZB-ZBWO                                                            SPA20  26
C                                                                           SPA20  27
      UTU=0.0                                                               SPA20  28
      VTV=0.0                                                               SPA20  29
      WTW=0.0                                                               SPA20  30
C                                                                           SPA20  31
C     IF FIELD POINT IS BETWEEN WING LEADING AND TRAILING EDGE MACH         SPA20  32
C        CONES, CALCULATE WING-FUSELAGE INDUCED VELOCITIES USING            SPA20  33
C        LEAST SQUARE POLYNOMIALS.  DO THIS ONLY IF ISMTH=1                 SPA20  34
C                                                                           SPA20  35
      IF (ISMTH.NE.1) GO TO 100                                             SPA20  36
      IF (XW.GT.XT3.OR.XW.LT.XT4) GO TO 100                                 SPA20  37
      XP=XSWFTW-XW                                                          SPA20  38
      MM=NORDW+1                                                            SPA20  39
      UP=WSCOE(1,1)                                                         SPA20  40
      VP=WSCOE(1,2)                                                         SPA20  41
      WP=WSCOE(1,3)                                                         SPA20  42
      XX=1.0                                                                SPA20  43
      DO 150 M=2,MM                                                         SPA20  44
      XX=XX*XP                                                              SPA20  45
      UP=UP+XX*WSCOE(M,1)                                                   SPA20  46
      VP=VP+XX*WSCOE(M,2)                                                   SPA20  47
  150 WP=WP+XX*WSCOE(M,3)                                                   SPA20  48
      UTU=UTU+UP                                                            SPA20  49
      VTV=VTV+VP                                                            SPA20  50
      WTW=WTW+WP                                                            SPA20  51
      GO TO 200                                                             SPA20  52
C                                                                           SPA20  53
C     CALCULATE FUSELAGE SOURCE-DOUBLET VELOCITIES IF ISMTH=0 OR FIELD      SPA20  54
C          POINT IS OUTSIDE MACH CONE RANGE WHEN ISMTH=1                    SPA20  55
C                                                                           SPA20  56
  100 IF (NFU.EQ.0) GO TO 175                                               SPA20  57
      CALL VELCAL(FSS,FDS,FXL,NFSOR,XW,YW,ZW,UP,VP,WP)                      SPA20  58
      UTU=UTU+UP                                                            SPA20  59
      VTV=VTV+VP                                                            SPA20  60
      WTW=WTW+WP                                                            SPA20  61
C                                                                           SPA20  62
C     CALCULATE FUSELAGE INTERFERENCE PANEL VELOCITIES IF ISMTH=0 OR        SPA20  63
C       FIELD POINT OUTSIDE XT3, XT4 RANGE WHEN ISMTH=1                     SPA20  64
C                                                                           SPA20  65
      IF (ISMTH.EQ.1.AND.XW.GT.0.0) GO TO 175                               SPA20  66
      INUMCH=1                                                              SPA20  67
      II=N2P                                                                SPA20  68
      IF=NPTOT                                                              SPA20  69
      CALL VELND(XW,YW,ZW)                                                  SPA20  70
      UTU=UTU+UP                                                            SPA20  71
      VTV=VTV+VP                                                            SPA20  72
      WTW=WTW+WP                                                            SPA20  73
C                                                                           SPA20  74
C     CALCULATE WING PANEL AND WING THICKNESS VELOCITIES IF ISMTH=0 OR      SPA20  75
C          FIELD POINT BEHIND TRAILING EDGE MACH CONE                       SPA20  76
C                                                                           SPA20  77
  175 IF (ISMTH.EQ.1.AND.XW.GT.XT4) GO TO 200                               SPA20  78
      INUMCH=1                                                              SPA20  79
      II=1                                                                  SPA20  80
      IF=NPANLS                                                             SPA20  81
      CALL VELNP(XW,YW,ZW)                                                  SPA20  82
      UTU=UTU+UP                                                            SPA20  83
      VTV=VTV+VP                                                            SPA20  84
      WTW=WTW+WP                                                            SPA20  85
      II=1                                                                  SPA20  86
C                                                                           SPA20  87
      IF=MS                                                                 SPA20  88
      CALL VELWTH(XW,YW,ZW)                                                 SPA20  89
      UTU=UTU+UP                                                            SPA20  90
      VTV=VTV+VP                                                            SPA20  91
      WTW=WTW+WP                                                            SPA20  92
C                                                                           SPA20  93
C     CALCULATE PYLON PANEL AND PYLON THICKNESS VELOCITIES                  SPA20  94
C                                                                           SPA20  95
  200 IF (NPY.EQ.0) RETURN                                                  SPA20  96
      INUMCH=0                                                              SPA20  97
      II=N1P                                                                SPA20  98
      IF=N2                                                                 SPA20  99
      CALL VELPP(XW,YW,ZW)                                                  SPA20100
      UTU=UTU+UP                                                            SPA20101
      VTV=VTV+VP                                                            SPA20102
      WTW=WTW+WP                                                            SPA20103
      II=MS+1                                                               SPA20104
      IF=NTWP                                                               SPA20105
      CALL VELPTH(XW,YW,ZW)                                                 SPA20106
      UTU=UTU+UP                                                            SPA20107
      VTV=VTV+VP                                                            SPA20108
      WTW=WTW+WP                                                            SPA20109
      RETURN                                                                SPA20110
      END
```

```
      SUBROUTINE SEMFOR                                                     SPA21   1
C     SUBROUTINE TO CALCULATE EMPENNAGE FORCES                              SPA21   2
C                                                                           SPA21   3
      DIMENSION UTL(11,4),VTL(11,4),WTL(11,4),VN(11,4),SARG(11)            SPA21   4
      COMMON /CFORCE/ CDC,CLMHY,CLMCP,CLMSB,CLNHY,CLNCP,CLNSW,CNRY,CNCP,    SPA21   5
```


(q) Page 17.

Figure 1.- Continued.

```
      1CNSB,CNB(40),CYBY,CYCP,CYSB,CYB(40),DC(3,3),DELX,DX,EURDX(81),
     2EWAU(81),ESTRMX,UT(81),VAN(12),VBTUBE,VT(81),WT(81),XMOM
      COMMON /CONSTS/PI,PI2,DTOR,RTOD,FOURPI
      COMMON /EFORCE/ CNEM,CLMEM,CYEM,CLNEM,CLLEM
      COMMON /EMPCON/ NFIN(11),XTAIL(11,4),ZTAIL(11,4),RMULE(4),FCONA,
     1FCONB,FCONC,CCLS(11),CCLS(11),XTAIL1,DYFIN
      COMMON /EMPDAT/ FINSS,RADAX,RTAIL,PHINCL,MSF,IPLNR,CLALPH
      COMMON /FLOW/ALPALN,GAMP,FMACH,RHO,VINF,BETA,BETASQ,FMCHSQ
      COMMON /IFORCE/ NDAMP,NEJSTR,NEMP,NGAM,NMSEG,NMSEGD,NROLL
      COMMON/NUFLO/FNUMCH,BETANU,INUMCH,ISMTH,XT3,XT4
      DO 1 J=1,4
      DO 1 K=1,MSF
      UTL(K,J)=0.0
      VTL(K,J)=0.0
    1 WTL(K,J)=0.0
      XXX=XTAIL+XMOM
      JMAX=4
      IF (IPLNR.EQ.1) JMAX=2
C
C     CALCULATE VELOCITIES IN STORE COORDINATE SYSTEM
C
      ISMTH=0
      DO 2 J=1,JMAX
      DO 2 K=1,MSF
      YYY=YTAIL(K,J)
      ZZZ=ZTAIL(K,J)
      CALL STTOIN (XXX,YYY,ZZZ,XI,ETA,ZETA,DC)
      XB=VAR(7)+XI
      YB=VAR(8)+ETA
      ZB=VAR(9)+ZETA
C
C     CALCULATE PERTURBATION VELOCITY FIELD
C
      CALL REBVEL(XB,YB,ZB,URES,VRES,WRES)
      CALL INTOBT(URES,VRES,WRES,UTL(K,J),VTL(K,J),WTL(K,J),DC)
    2 CONTINUE
C
C     CALCULATE STORE FREE-STREAM VELOCITY COMPONENTS
C
      VXI=VINF*COS(ALFACR)
      VETA=0.0
      VZETA=VINF*SIN(ALFACR)
      IF (NGAM.EQ.1) GO TO 3
      VXI=VXI+VAR(1)
      VETA=VETA+VAR(2)
      VZETA=VZETA+VAR(3)
    3 VBTUBE=SQRT(VXI**2+VETA**2+VZETA**2)
      CALL INTOBT (VXI,VETA,VZETA,VX,VY,VZ,DC)
C
C     ADD FREE-STREAM COMPONENTS TO PERTURBATION VELOCITIES
C     RESULTING VELOCITIES ARE POSITIVE IN +X, +Y, AND +Z DIRECTIONS
C
      VRATIO=VINF/VBTUBE
      FA=VX/VBTUBE
      FB=VY/VBTUBE
      FC=VZ/VBTUBE
      DO 4 J=1,JMAX
      DO 4 K=1,MSF
      UTL(K,J)=+VRATIO+UTL(K,J)+FA
      VTL(K,J)=+VRATIO+VTL(K,J)+FB
    4 WTL(K,J)=+VRATIO+WTL(K,J)+FC
C
C     ADD IN Q AND R DAMPING
C
      IF (NDAMP.EQ.0) GO TO 6
      DUM=-XXX/VBTUBE
      FA=VAR(6)*DUM
      FB=VAR(5)*DUM
      DO 5 J=1,JMAX
      DO 5 K=1,MSF
      VTL(K,J)=VTL(K,J)+FA
    5 WTL(K,J)=WTL(K,J)+FB
C
C     DETERMINE VELOCITY COMPONENTS NORMAL TO PANELS EXCLUDING
C          ROLL DAMPING CONTRIBUTION
C
    6 DUM=1.0
      DO 7 J=1,JMAX
      SPHI=SIN(RMULE(J))
      CPHI=COS(RMULE(J))
      DUM=-DUM
      DO 7 K=1,MSF
    7 VN(K,J)=DUM*(-UTL(K,J)*SPHI+VTL(K,J)*CPHI)
C
C     CALCULATE EMPENNAGE NORMAL FORCE, SIDE FORCE, PITCHING MOMENT,
C          AND YAWING MOMENT
C
      DUM=-XTAIL/DX
      IT=DUM+0.5
      IT1=IT+1
      V80=VT(IT)
      W80=WT(IT)
      ANG=PTOR*PHINCL
      WD=-V80*COS(ANG)+W80*SIN(ANG)
      VU=-V80*SIN(ANG)+W80*COS(ANG)
      A2=RADAV**2
      DO 8 K=1,MSF
      AS=(VN(K,1)+VN(K,2))/2.0+WD*A2/(RFIN(K)**2)
    8 SARG(K)=AS*CCL3(K)
      CALL SIMSUN (MSF,SARG,DYFIN,SUM)
      CNEM=FCONA*SUM
      CLMEM=-FCONB*SUM*FCONC
      CYEM=0.0
      CLNEM=0.0
      IF (IPLNR.EQ.1) GO TO 10
      DO 9 K=1,MSF
      BS=(VN(K,3)+VN(K,4))/2.0+VU*A2/(RFIN(K)**2)
    9 SARG(K)=BS*CCL3(K)
      CALL SIMSUN (MSF,SARG,DYFIN,SUM)
      CYEM=FCONA*SUM
      CLNEM=-FCONB*SUM*FCONC
   10 TA=CNEM
      TB=CYEM
      CA=COS(ANG)
      SA=SIN(ANG)
      CNEM=TA*CA-TB*SA
      CYEM=TA*SA+TB*CA
      TA=CLMEM
      TB=CLNEM
      CLMEM=TA*CA-TB*SA
      CLNEM=TA*SA+TB*CA
C
C     CALCULATE ROLLING MOMENT
C
      CLLEM=0.0
      IF (NROLL.EQ.0) RETURN
      DO 11 K=1,MSF
      ABU=(VN(K,1)+VN(K,2))/2.0
      IF (NDAMP.GT.0) ABU=ABU+VAR(4)*RFIN(K)/VBTUBE
      IF (IPLNR.EQ.1) GO TO 12
      ABU=ABU+(VN(K,3)+VN(K,4))/2.0
      IF (NDAMP.GT.0) ABU=ABU+VAR(4)*RFIN(K)/VBTUBE
   12 SARG(K)=ABU*CCLS(K)
   11 CONTINUE
      CALL SIMSUN (MSF,SARG,DYFIN,SUM)
      CLLEM=-FCONB*SUM
      RETURN
      END


      SUBROUTINE SEMPIN
C     SUBROUTINE TO INITIALIZE FOR EMPENNAGE CALCULATION
C
      COMMON /EFORCE/ CDC,CLMBY,CLMCP,CLMSB,CLNBY,CLNCP,CLNSB,CNBY,CNCP,
     1CNSB,CNX(40),CYBY,CYCP,CYSB,CYB(40),DC(3,3),DELX,DX,EURDX(81),
```

(r) Page 18.

Figure 1.- Continued.

```
      ZERAD(81),ESTRMX,UT(81),VAR(12),VSTORE,VT(81),WT(81),XMUM
      COMMON /CONSTS/PI,PI2,DTOR,RTOD,FOURPI
      COMMON /EMPDAT/ FINSS,RADAV,STAIL,PHIROL,MSF,IPLNR,CLALPH
      COMMON /EMPCON/ RFIN(11),YTAIL(11,4),ZTAIL(11,4),PHIRLE(4),FCONA,
     1FCONR,FCONC,CCL3(11),CCL5(11),RTAIL,DYFIN
      COMMON /IFORCE/ NOAMP,NEJSTR,NEMP,NGAM,NMSEG,NMSEGU,NROLL
C
C     LOCATE SPAN-WISE CONTROL POINTS
C
      TAPMSF=1
      TBRFINSS=RADAV
      DYFIN=TB/TA
      DO 1 J=1,MSF
      TC=J-1
    1 RFIN(J)=RADAV-TC*DYFIN
C
C     LOCATE CONTROL POINTS ON FINS IN STORE COORDINATE SYSTEM
C
      IF (XTAIL.GT.0.0) XTAIL=-XTAIL
      FRULE(1)=DTOR*(PHIROL+90.0)
      FRULE(2)=DTOR*(PHIROL+270.0)
      FRULE(3)=DTOR*(PHIROL+180.0)
      FRULE(4)=DTOR*(PHIROL)
      JMAX=4
      IF (IPLNR.EQ.1) JMAX=2
      DO 2 J=1,JMAX
      SPHI=SIN(FRULE(J))
      CPHI=COS(FRULE(J))
      DO 2 K=1,MSF
      YTAIL(K,J)=RFIN(K)*SPHI
    2 ZTAIL(K,J)=-RFIN(K)*CPHI
C
C     COMPUTE CONSTANTS
C
      FCONA=CLALPH/(3.1415927*TB*TB)
      FCONB=FCONA/(2.0*ESTRMX)
      FCONC=-XTAIL-XMUM
      SMS=FINSS**2
      AA=RADAV**4
      A2=RADAV**2
      TOVPI=2.0/3.1415927
      DO 3 J=1,MSF
      R2=RFIN(J)**2
      ARG=(SMS+M2-A4)*(SMS-R2)/(SMS*R2)
    3 CCL3(J)=0.0-SQRT(ARG)
      IF (NROLL.EQ.0) RETURN
      IF (IPLNR.EQ.0) GO TO 10
C
C     COMPUTE LOADING COEFFICIENTS FOR ROLLING MOMENT CALCULATION
C
C     PLANAR EMPENNAGE
C
      TA=1.0+TOVPI*ACOS(2.0*FINSS*RADAV/(SMS*A2))
      TB=(SMS-A2)/(SMS+A2)
      DO 4 J=1,MSF
      R=RFIN(J)
      R2=R*R
      CCL5(J)=TA*(R+A2/R)*CCL3(J)/8.0
      IF (J.EQ.1.OR.J.EQ.MSF) GO TO 4
      TC=(R2+A2)*TB/(R2-A2)
      CCL5(J)=CCL5(J)+TOVPI*((R-A2/R)**2)*ALOG(TC+SQRT(TC*TC-1.0))
    4 CONTINUE
      RETURN
C
C     CRUCIFORM EMPENNAGE
C
   10 CAPRSQ=(SMS+A4/SMS)/4.0
      CTNGAM=A2/(2.0*CAPRSQ)
      TNGAM=ACOS(CTNGAM)
      STNGAM=SIN(TNGAM)
      TTNGAM=STNGAM/CTNGAM
      AK1=STNGAM
      CALL CEL1 (AKK1,ATNGAM,IER)
      R2=RFIN(1)**2
      R4=R2*R2
      SM4=SMS*SMS
      TNUTHE=ACOS(SMS*(A4+R4)/(M2*(A4+SM4)))
      CCL5(1)=4.0*TOVPI*CAPRSQ*(-CTNGAM*ALOG(CTNGAM)+0.5*AKK1*SIN(2.0*
     1TNUTHE))
      CK=SQRT(1.0-AK1*AK1)
      CKCK=CK*CK
      CALL CEL2 (EPI02K,AK1,1.0,CKCK,IER)
      DO 11 J=2,MSF
      R2=RFIN(J)**2
      R4=R2*R2
      CTNTHE=SMS*(A4+R4)/(R2*(A4+SM4))
      TNTHE=ACOS(CTNTHE)
      STNTHE=SIN(TNTHE)
      TTNTHE=TAN(TNTHE)
      ARG1=STNTHE/STNGAM
      ARG2=TTNTHE/TTNGAM
      ARG3=ASIN(ARG1)
      CCL5(J)=4.0*TOVPI*CAPRSQ*(CTNTHE+0.5*ALOG((1.0+ARG1)/(1.0-ARG1))
     1-CTNGAM+0.5*ALOG((1.0+ARG2)/(1.0-ARG2)))
      TANA1=TAN(ARG3)
      CALL EL1I (FAK,TANA1,CK)
      CALL EL2I (EAK,TANA1,CK,1.0,CKCK)
      CCL5(J)=CCL5(J)+2.0*TOVPI*CAPRSQ*(AKK1*SIN(2.0*TNTHE)-2.0*STNGAM
     1*COS(ARG3)*(AKK1*EAK-EPI02K*FAK))
   11 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SFORCE
C
C     SUBROUTINE TO CALCULATE AERODYNAMIC FORCES AND MOMENTS ON BODY
C
      DIMENSION DUMMY(12)
C
      COMMON /CFORCE/ CDC,CLMBY,CLMCP,CLMBB,CLNBY,CLNCF,CLNSB,CNRY,CNLP,
     1CNBB,CNX(40),CYBY,CYCP,CYBB,CYX(40),DC(3,3),DELR,DX,RDMDX(81),
     2ERAD(81),ESTRMX,UT(81),VAR(12),VSTORE,VT(81),WT(81),XMUM
      COMMON /FLOW/ALPHCP,GAMP,FMACH,RHO,VINF,BETA,BETABU,FMCHSU
      COMMON /IFORCE/ NOAMP,NEJSTR,NEMP,NGAM,NMSEG,NMSEGU,NROLL
      COMMON/WGEOM/XBWOC,ZBWO,DUMA(63)
      COMMON/NUFLOW/FNUMCH,BETANU,INUMCH,IBMTH,XTS,XTN
C
      IF (NGAM) 20,20,21
   20 CALL DIRCOS(VAR,DC)
      GO TO 22
   21 DUMMY(10)=VAR(10)-ATAN(VAR(2)/(VINF*COS(ALPHCP)+VAR(1)))
      DUMMY(11)=VAR(11)-ATAN(VAR(3)/(VINF*COS(ALPHCP)+VAR(1)))
      DUMMY(12)=VAR(12)
      CALL DIRCOS(DUMMY,DC)
   22 CONTINUE
      YY=VAR(8)
      ZZ=VAR(9)-ZBWO
      CALL NUMACH(YY,ZZ)
      DO 1 J=1,NMSEG
      UT(J)=0.0
      VT(J)=0.0
    1 WT(J)=0.0
      XX=DX
      ISMTH=1
      DO 10 N=1,NMSEG
      XX=XX-DX
C
C     LOCATE POINT IN FUSELAGE SYSTEM
C
      XXX=XX+XMUM
      CALL STTOIN (XXX,0.0,0.0,0.0,XI,ETA,ZETA,DC)
```


Figure 1.- Continued.

```
      XB=VAR(7)+XI
      YB=VAR(8)+ETA
      ZB=VAR(9)+ZETA
C
C     CALCULATE PERTURBATION VELOCITY FIELD AND RESOLVE
C         INTO STORE COORDINATE SYSTEM
C
      CALL RESVEL(XB,YB,ZB,URES,VRES,WRES)
      CALL INTOST(URES,VRES,WRES,UT(N),VT(N),WT(N),UC)
   10 CONTINUE
C
C     CALCULATE STORE FREE-STREAM VELOCITY AND COMPONENTS
C
      IF (NGAM) 50,50,55
   55 VXI=VINF*COS(ALFACR)
      VETA=0.0
      VZETA=VINF*SIN(ALFACR)
      GO TO 57
   50 VXI=VINF*COS(ALFACR)+VAR(1)
      VETA=VAR(2)
      VZETA=VINF*SIN(ALFACR)+VAR(3)
   57 VSTORE=SQRT(VXI**2+VETA**2+VZETA**2)
      CALL INTOST (VXI,VETA,VZETA,VX,VY,VZ,DC)
C
C     ADD FREE-STREAM COMPONENTS TO PERTURBATION VELOCITIES
C     RESULTING VELOCITIES ARE IN -X, +Y, AND -Z DIRECTIONS
C
      VRATIO=VINF/VSTORE
      TA=VX/VSTORE
      TB=VY/VSTORE
      TC=VZ/VSTORE
      DO 60 N=1,NMBEG
      UT(N)=-VRATIO*UT(N)+TA
      VT(N)=VRATIO*VT(N)-TB
   60 WT(N)=-VRATIO*WT(N)+TC
C
C     ADD IN DAMPING TERMS
C
      IF (NDAMP.EQ.0) GO TO 70
      XX=-DX
      DO 65 N=1,NMBEG
      XX=XX+DX
      DUM=(XX-XMOM)/VSTORE
      VT(N)=VT(N)+VAR(6)*DUM
   65 WT(N)=WT(N)+VAR(5)*DUM
C
C     CALCULATE FORCES AND MOMENTS
C
   70 CONA=2.0/(ESTRMX**2)
      CONB=CONA*CDC/3.1415927
      XSTOR=-DX
      CNBY=0.0
      CYBY=0.0
      CLMBY=0.0
      CLNBY=0.0
C
C     BUOYANCY FORCES AND MOMENTS
C
      TA=DELX/(2.0*ESTRMX)
      DO 71 N=2,NMBEG,2
      NN=N/2
      XSTOR=XSTOR+DELX
      DUM=XMOM-XSTOR
      CONC=CONA*ERAD(N)**2
      COND=DUM*TA
      DCN=CONC*(WT(N+1)-WT(N-1))/DELX
      DCY=CONC*(VT(N+1)-VT(N-1))/DELX
      CNBY=CNBY+DELX*DCN
      CYBY=CYBY+DELX*DCY
      CNX(NN)=DCN
      CYX(NN)=DCY
      CLMBY=CLMBY+COND*DCN
      CLNBY=CLNBY+COND*DCY
   71 CONTINUE
C
C     SLENDER BODY THEORY FORCES AND MOMENTS
C
      CNSB=0.0
      CYSB=0.0
      CLMSB=0.0
      CLNSB=0.0
      XSTOR=-DX
      DO 72 N=2,NMBEGO,2
      XSTOR=XSTOR+DELX
      NN=N/2
      CONC=2.0*CONA*ERAD(N)*EDRDX(N)
      COND=(XMOM-XSTOR)*TA
      DCN=CNX(NN)+CONC*WT(N)
      DCY=CYX(NN)+CONC*VT(N)
      CNSB=CNSB+DELX*DCN
      CYSB=CYSB+DELX*DCY
      CNX(NN)=CNX(NN)+DCN
      CYX(NN)=CYX(NN)+DCY
      CLMSB=CLMSB+COND*DCN
      CLNSB=CLNSB+COND*DCY
   72 CONTINUE
C
C     VISCOUS CROSSFLOW FORCES AND MOMENTS
C
      CNCF=0.0
      CYCF=0.0
      CLMCF=0.0
      CLNCF=0.0
      IF (NMBEG.EQ.NMBEGO) GO TO 80
      NI=NMBEGO+1
      DO 73 N=NI,NMBEG,2
      VC=SQRT(WT(N)**2+VT(N)**2)
      XSTOR=XSTOR+DELX
      NN=N/2
      CONC=CONB*ERAD(N)*VC
      COND=(XMOM-XBTOR)/(2.0*ESTRMX)*DELX
      DCN=CONC*WT(N)
      DCY=CONC*VT(N)
      CNCF=CNCF+DELX*DCN
      CYCF=CYCF+DELX*DCY
      CNX(NN)=CNX(NN)+DCN
      CYX(NN)=CYX(NN)+DCY
      CLMCF=CLMCF+COND*DCN
      CLNCF=CLNCF+COND*DCY
   73 CONTINUE
   80 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SHAPE (X,NS,XE,C,R,DRDX)
C
C     SUBROUTINE TO CALCULATE LOCAL BODY RADIUS AND SURFACE SLOPE
C
      DIMENSION XE(7),C(7,7)
      DO 1 K=1,NS
      XL=XE(K)
      JBK
      IF (X.LE.XL) GO TO 2
    1 CONTINUE
    2 R=C(J,1)+X*C(J,5)+X*X*C(J,6)
      ARG=X*X*C(J,2)+X*C(J,3)+C(J,4)
      DRDX=C(J,5)+2.0*X*C(J,6)
      IF (ARG.LE.0.0) RETURN
      R=R+SQRT(ARG)*C(J,7)
      DRDX=DRDX+(2.0*X*C(J,2)+C(J,3))/(2.0*SQRT(ARG))*C(J,7)
      RETURN
      END
```

(t) Page 20.

Figure 1.- Continued.

```
      SUBROUTINE SIMSUN(N,F,DX,SUM)                                     SPA25  1
C     SIMPSON RULE SUBROUTINE                                           SPA25  2
      DIMENSION F(101)                                                  SPA25  3
      SUM=F(1)+F(N)                                                     SPA25  4
      DO 1 I=2,N,2                                                      SPA25  5
    1 SUM=SUM+4.0*F(I)                                                  SPA25  6
      M=N-2                                                             SPA25  7
      DO 2 I=3,M,2                                                      SPA25  8
    2 SUM=SUM+2.0*F(I)                                                  SPA25  9
      SUM=DX*SUM/3.0                                                    SPA25 10
      RETURN                                                            SPA25 11
      END                                                               SPA25 12
```

```
      SUBROUTINE SOURCE(TX)                                             SPA26  1
C                                                                       SPA26  2
C SUBROUTINE TO CALCULATE THE STRENGTH OF A LINEAR LINE SOURCE OF       SPA26  3
C UNIT SLOPE WITH ORIGIN AT TX(J).                                      SPA26  4
C                                                                       SPA26  5
      COMMON/SRCE/XFIELD,RFIELD,BSQ,PI2,      U,V,VT                    SPA26  6
      COMMON /FLOW/DUM1(5),BETA,DUM2(2)                                 SPA26  7
C                                                                       SPA26  8
      XL=XFIELD-TX                                                      SPA26  9
      BR=BETA*RFIELD                                                    SPA26 10
      IF(XL,LE,BR) GO TO 10                                             SPA26 11
      XL=XL/BR                                                          SPA26 12
      D23= SQRT(XL*XL-1.)                                               SPA26 13
      U=-ALOG(XL+D23)                                                   SPA26 14
      V=BETA*D23                                                        SPA26 15
      RETURN                                                            SPA26 16
C                                                                       SPA26 17
C FIELD POINT IS AHEAD OF MACH CONE FROM SOURCE ORIGIN,                 SPA26 18
C                                                                       SPA26 19
   10 U=0.                                                              SPA26 20
      V=0.                                                              SPA26 21
      RETURN                                                            SPA26 22
      END                                                               SPA26 23
```

```
      SUBROUTINE SOUTPT                                                 SPA27  1
C                                                                       SPA27  2
C     SUBROUTINE TO OUTPUT FORCE AND MOMENT DATA AND TRAJECTORY DATA    SPA27  3
C                                                                       SPA27  4
      COMMON /CFORCE/ CDC,CLMBY,CLMCP,CLMBB,CLNBY,CLNCP,CLNSB,CNBY,CNCP,SPA27  5
     1CNSB,CNP(40),CYBY,CYCP,CYBB,CYP(40),DC(3,3),DELM,DX,PDROX(81),    SPA27  6
     2ERAD(81),EBTRMX,VT(81),VAR(12),VSTORE,VT(81),MT(81),XMOM          SPA27  7
      COMMON /COUTPT/ DVAR(12),ESTLGC,EXBT(81),TIME                     SPA27  8
      COMMON /FFORCE/ CNEM,CLMEM,CYEM,CLNEM,CLLEM                       SPA27  9
      COMMON /IFORCE/ NDAMP,NEJBTR,NEMP,NGAM,NMSEG,NMSEGO,NROLL         SPA27 10
      COMMON /OUTINI/ XNOSEI,YNOSEI,ZNOSEI,XCGI,YCGI,ZCGI,XBASEI,YBASEI,SPA27 11
                                                                        SPA27 12
     1ZBASEI                                                            SPA27 13
      COMMON /THRUST/ NTPOLY,TEND(5),TC(5,6),FTHRUS,NTHRUS              SPA27 14
C                                                                       SPA27 15
  701 FORMAT(1H1,5X,6HTIME =,F8,5,4H SECONDS)                           SPA27 16
  702 FORMAT(/5X,29HFORCE AND MOMENT COEFFICIENTS/25X,2HCN,8X,2HCY,7X,  SPA27 17
     13HCLM,7X,3HCLN,7X,3HCLL/7X,6HBUOYANCY,5X,4F10,5/7X,12HBLENDER BODYSPA27 18
     2,1X,4F10,5/7X,9HCROSSFLOW,4X,4F10,5/7X,9HEMPENNAGE,4X,5F10,5/7X, SPA27 19
     35HTOTAL,8X,5F10,5)                                                SPA27 20
  703 FORMAT(/5X,31HLOAD AND VELOCITY DISTRIBUTIONS/14X,5HX, FT,6X,5HX/LSPA27 21
     1,5X,6HDCN/DX,4X,6HDCY/DX,5X,4HU/VS,6X,4HV/VS,6X,4HW/VS)           SPA27 22
  704 FORMAT(10X,9F10,5)                                                SPA27 23
  705 FORMAT(/5X,67HLOCATION OF STORE IN FUSELAGE COORDINATE SYSTEM, DIMSPA27 24
     1ENSIONS OF FEET/17X,25HRELATIVE TO FUSELAGE NOSE,8X,28HRELATIVE TOSPA27 25
     2 INITIAL POSITION/20X,2HXF,8X,2HYF,8X,2HZF,11X,6HDEL XF,4X,6HDEL YSPA27 26
     3F,4X,6HDEL ZF)                                                    SPA27 27
  706 FORMAT(/5X,81HTRANSLATIONAL VELOCITIES AND ACCELERATIONS OF STORE SPA27 28
     1IN FUSELAGE COORDINATE SYSTEM/10X,29HRELATIVE TO FUSELAGE MOTION/ SPA27 29
     215X,3HDXF,7X,3HDYF,7X,3HDZF,6X,4HD2XF,6X,4HD2YF,6X,4HD2ZF)        SPA27 30
  707 FORMAT(/5X,75HROTATIONAL VELOCITIES AND ACCELERATIONS OF STORE IN SPA27 31
     1STORE COORDINATE SYSTEM/10X,1HP,9X,1HQ,9X,1HR,7X,6HPDOT,6X,6HQDOT,SPA27 32
     26X,6HRDOT)                                                        SPA27 33
  708 FORMAT(/5X,91HSTORE ANGULAR ORIENTATION IN FUSELAGE COORDINATE SYSSPA27 34
     1TEM AND RATES OF CHANGE OF THESE ANGLES/10X,50HANGLES IN DEGREES, SPA27 35
     2RATES OF CHANGE IN RADIANS PER SECOND/15X,3HPSI,6X,5HTHETA,6X,3HPHSPA27 36
     3I,6X,6HDPSI,5X,6HDTHETA,5X,6HDPHI)                                SPA27 37
  709 FORMAT(10X,4HNOSE,1X,3F10,5,5X,3F10,5)                            SPA27 38
  710 FORMAT(10X,4HXMOM,1X,3F10,5,5X,3F10,5)                            SPA27 39
  711 FORMAT(10X,4HBASE,1X,3F10,5,5X,3F10,5)                            SPA27 40
  712 FORMAT(/5X,15HSTORE THRUST IS,F11,3,7H POUNDS)                    SPA27 41
C                                                                       SPA27 42
C     OUTPUT TIME                                                       SPA27 43
C                                                                       SPA27 44
      WRITE (6,701) TIME                                                SPA27 45
C                                                                       SPA27 46
C     OUTPUT FORCES AND MOMENTS                                         SPA27 47
C                                                                       SPA27 48
      CNT=CNBY+CNSB+CNCP+CNEM                                           SPA27 49
      CYT=CYBY+CYBB+CYCP+CYEM                                           SPA27 50
      CLMT=CLMBY+CLMBB+CLMCP+CLMEM                                      SPA27 51
      CLNT=CLNBY+CLNSB+CLNCP+CLNEM                                      SPA27 52
      CLLT=CLLEM                                                        SPA27 53
      WRITE (6,702) CNBY,CYBY,CLMBY,CLNBY,CNSB,CYSB,CLMBB,CLNSB,CNCP,  SPA27 54
     1CYCP,CLMCP,CLNCP,CNEM,CYEM,CLMEM,CLNEM,CLLEM,CNT,CYT,CLMT,CLNT,  SPA27 55
     2CLLT                                                              SPA27 56
C                                                                       SPA27 57
C     OUTPUT STORE THRUST                                               SPA27 58
C                                                                       SPA27 59
      IF (NTHRUS,EQ,0) GO TO 2                                          SPA27 60
      WRITE (6,712) FTHRUS                                              SPA27 61
    2 CONTINUE                                                          SPA27 62
C                                                                       SPA27 63
C     OUTPUT LOAD AND VELOCITY DISTRIBUTIONS                            SPA27 64
C                                                                       SPA27 65
      WRITE (6,703)                                                     SPA27 66
      DO 1 J=2,NMSEG,2                                                  SPA27 67
      K=J/2                                                             SPA27 68
      XL=EXBT(J)/ESTLGC                                                 SPA27 69
    1 WRITE (6,704) EXBT(J),XL,CNB(K),CYX(K),VT(J),VT(J),MT(J)          SPA27 70
C                                                                       SPA27 71
C     CALCULATE AND OUTPUT STORE LOCATION (NOSE, MOMENT CENTER,         SPA27 72
C         AND BASE) IN FUSELAGE COORDINATE SYSTEM                       SPA27 73
C                                                                       SPA27 74
      XXX=XMOM                                                          SPA27 75
      CALL STTOIN (XXX,0,0,0,0,XI,ETA,ZETA,DC)                          SPA27 76
      XNOSE=VAR(7)+XI                                                   SPA27 77
      YNOSE=VAR(8)+ETA                                                  SPA27 78
      ZNOSE=VAR(9)+ZETA                                                 SPA27 79
      DX=XNOSE-XNOSEI                                                   SPA27 80
      DY=YNOSE-YNOSEI                                                   SPA27 81
      DZ=ZNOSE-ZNOSEI                                                   SPA27 82
      XXX=-(ESTLGC-XMOM)                                                SPA27 83
      CALL STTOIN (XXX,0,0,0,0,XI,ETA,ZETA,DC)                          SPA27 84
      XBASE=VAR(7)+XI                                                   SPA27 85
      YBASE=VAR(8)+ETA                                                  SPA27 86
      ZBASE=VAR(9)+ZETA                                                 SPA27 87
      DXB=XBASE-XBASEI                                                  SPA27 88
      DYB=YBASE-YBASEI                                                  SPA27 89
      DZB=ZBASE-ZBASEI                                                  SPA27 90
      DXCG=VAR(7)-XCGI                                                  SPA27 91
      DYCG=VAR(8)-YCGI                                                  SPA27 92
      DZCG=VAR(9)-ZCGI                                                  SPA27 93
      WRITE (6,705)                                                     SPA27 94
      WRITE (6,709) XNOSE,YNOSE,ZNOSE,DXN,DYN,DZN                       SPA27 95
      WRITE (6,710) VAR(7),VAR(8),VAR(9),DXCG,DYCG,DZCG                 SPA27 96
```


(u) Page 21.

Figure 1.- Continued.

```
      WRITE (6,711) XBASE,YBASE,ZBASE,DXB,DYB,DZB                       SPA27 96
C                                                                       SPA27 97
C     OUTPUT STORE VELOCITIES                                           SPA27 98
C                                                                       SPA27 99
      WRITE (6,706)                                                     SPA27100
      WRITE (6,704) VAR(1),VAR(2),VAR(3),DVAR(1),DVAR(2),DVAR(3)        SPA27101
C                                                                       SPA27102
C     OUTPUT STORE ACCELERATIONS                                        SPA27103
C                                                                       SPA27104
      WRITE (6,707)                                                     SPA27105
      WRITE (6,704) VAR(4),VAR(5),VAR(6),DVAR(4),DVAR(5),DVAR(6)        SPA27106
C                                                                       SPA27107
C     OUTPUT STORE ANGULAR ORIENTATION                                  SPA27108
C                                                                       SPA27109
      PSI=57.29578*VAR(10)                                              SPA27110
      THA=57.29578*VAR(11)                                              SPA27111
      PHI=57.29578*VAR(12)                                              SPA27112
      WRITE (6,708)                                                     SPA27113
      WRITE (6,704) PSI,THA,PHI,DVAR(10),DVAR(11),DVAR(12)              SPA27114
      RETURN                                                            SPA27115
      END                                                               SPA27116
```

```
      SUBROUTINE STORID                                                 SPA28  1
C                                                                       SPA28  2
C     SUBROUTINE TO INPUT AND OUTPUT STORE DATA                         SPA28  3
C         LINE SOURCE AND DOUBLET DISTRIBUTIONS ARE CALCULATED          SPA28  4
C         USING SUBROUTINE BDYGEN                                       SPA28  5
C                                                                       SPA28  6
C     ****                                                              SPA28  7
C     PRESENT ROUTINE LIMITED TO ONE STORE                              SPA28  8
C     ****                                                              SPA28  9
C                                                                       SPA28 10
      DIMENSION T(100),TX(101),TC(100)                                  SPA28 11
      DIMENSION NSHAPE(10),XSNC(10),YSN(10),ZSN(10),SIC(10)             SPA28 12
      DIMENSION SXNDJ(7),SCOFJ(7,7)                                     SPA28 13
C                                                                       SPA28 14
      COMMON /CONFIG/NFU,NPY,NSTRS,LVS=P                                SPA28 15
      COMMON /CONSTS/PI,P12,DTOR,RTOD,FOURPI                            SPA28 16
      COMMON /FLOW/ALFACR,GAMP,FMACH,RHO,VINF,BETA,BETASQ,FMCHSQ        SPA28 17
      COMMON/INDEX/NC=,MS=,MS=P,NPANLS,NCBB,NBDCR1,NBDCR2,NBD,NBIP,MP,  SPA28 18
     1   NCP,MSP,N1P,N2,N2P,NPTOT                                       SPA28 19
      COMMON /SSOR/SXL(101,1),SSS(100,1),SDB(100,1),NSSOR(1)            SPA28 20
      COMMON /STGEOM/SRMAX(1),SLTHC(1),XHBOC(1),YHBO(1),ZHBO(1),        SPA28 21
     1       NSPOLY(1),SXEND(7,1),SCOEF(7,7,1),XBSU(1),YBSO(1),ZBSO(1), SPA28 22
     2       SIBCR(1),NUMSTR(1),SSIBCR(1),CSIBCR(1)                     SPA28 23
      COMMON /=GEOM/XH=OC,ZBHO,CR=,SLPHLE,SLPATF,PSI=LE(20),PSI=TE(20),  SPA28 24
     1       Y(20)                                                      SPA28 25
      COMMON/=PGEOM/XH=(200),XB(200),XLF(200),XLB(200),YHC(200),        SPA28 26
     1   YLE(200),XCPT(200),YCPT(200),ZCPT(200),S=PPLE(200),S=PPTE(200) SPA28 27
C                                                                       SPA28 28
  701 FORMAT (2I5,7F10.0)                                               SPA28 29
  702 FORMAT(1H1,10X,18HSTORE INPUT DATA)                               SPA28 30
  703 FORMAT(/40X,56HMAXIMUM   STORE LOCATION RELATIVE TO,6X,9HINCIDENCESPA28 31
     1/15X,20HSTORE SHAPE   LENGTH,5X,6HRADIUS,4X,29HLOCAL WING CHORD LESPA28 32
     2ADING EDGE,5X,9HANGLE/17X,2HNO,8X,2HFT,9X,2HFT,8X,5HX, FT,       SPA28 33
     3 8X,5HY, FT,8X,5HZ, FT,6X,5HDEG)                                  SPA28 34
  704 FORMAT(15X,2I5,6F11.5)                                            SPA28 35
  707 FORMAT(///5X,48H***** SHAPE POLYNOMIALS NOT INPUT FOR ALL STORES) SPA28 36
  712 FORMAT (8F10.5)                                                   SPA28 37
  713 FORMAT (///15X,37HPOLYNOMIALS SPECIFYING STORE SHAPE NO.I3/       SPA28 38
     1      18X,26HX/L UP END OF EACH SECTION/21X,7HSECTION,5X,3HX/L)   SPA28 39
  714 FORMAT (23X,I2,3X,7F10.5)                                         SPA28 40
  715 FORMAT(18X,51HCOEFFICIENTS OF POLYNOMIALS DESCRIBING EACH SECTION/SPA28 41
     121X,7HSECTION,5X,2HC1,8X,2HC2,8X,2HC3,8X,2HC4,8X,2HC5,8X,2HC6,8X, SPA28 42
     22HC7)                                                             SPA28 43
  716 FORMAT(//15X,11HSTORE SHAPEI3,91H AS CALCULATED FROM THE INPUT POLYSPA28 44
     1NOMIALS, ORIGINS OF SOURCES AND DOUBLETS REPRESENTING THE/20X,    SPA28 45
     253HSTORE, AND VALUES OF THE SOURCE AND DOUBLET CONSTANTS)         SPA28 46
```

```
C                                                                       SPA28 47
C                                                                       SPA28 48
C     READ IN AND WRITE STORE GEOMETRY                                  SPA28 49
C                                                                       SPA28 50
C                                                                       SPA28 51
      DO 1 J=1,NSTRS                                                    SPA28 52
      READ (5,701) NUMSTR(J),NSHAPE(J),SLTHC(J),SRMAX(J),XBNC(J),      SPA28 53
     1      YSN(J),ZSN(J),SIC(J)                                        SPA28 54
    1 CONTINUE                                                          SPA28 55
      WRITE (6,702)                                                     SPA28 56
      WRITE (6,703)                                                     SPA28 57
      DO 2 J=1,NSTRS                                                    SPA28 58
      WRITE (6,704) NUMSTR(J),NSHAPE(J),SLTHC(J),SRMAX(J),XBNC(J),      SPA28 59
     1      YSN(J),ZSN(J),SIC(J)                                        SPA28 60
    2 CONTINUE                                                          SPA28 61
      NCOUNT=0                                                          SPA28 62
C                                                                       SPA28 63
C     READ IN AND PRINT POLYNOMIALS DESCRIBING STORE SHAPE              SPA28 64
C                                                                       SPA28 65
      READ (5,701) NSHPT                                                SPA28 66
      DO 3 J=1,NSHPT                                                    SPA28 67
      READ (5,701) MSHAPE,MSOR                                          SPA28 68
      READ(5,701) NSPOLJ                                                SPA28 69
      READ(5,712)(SXNDJ(K),K=1,NSPOLJ)                                  SPA28 70
      DO 50 K=1,NSPOLJ                                                  SPA28 71
   50 READ (5,712) (SCOFJ(K,L),L=1,7)                                   SPA28 72
      WRITE (6,713) MSHAPE                                              SPA28 73
      DO 51 K=1,NSPOLJ                                                  SPA28 74
   51 WRITE (6,714) K,SXNDJ(K)                                          SPA28 75
      WRITE (6,715)                                                     SPA28 76
      DO 52 K=1,NSPOLJ                                                  SPA28 77
   52 WRITE (6,714) K,(SCOFJ(K,L),L=1,7)                                SPA28 78
C                                                                       SPA28 79
      DO 4 K=1,NSTRS                                                    SPA28 80
      IF (MSHAPE .NE. NSHAPE(K)) GO TO 4                                SPA28 81
C                                                                       SPA28 82
C     CALCULATE LINE SOURCE AND DOUBLET DISTRIBUTION                    SPA28 83
C                                                                       SPA28 84
      SIBCR(K)=SIC(K)*DTOR                                              SPA28 85
      ALFAB=ALFACR+SIBCR(K)                                             SPA28 86
      NXBODY = MSOR+1                                                   SPA28 87
      WRITE (6,716) MSHAPE                                              SPA28 88
      CALL BDYGEN(NXBODY,SRMAX(K),SLTHC(K),NSPOLJ,SXNDJ,SCOFJ,T,TC,     SPA28 89
     1     TX,ALFAB)                                                    SPA28 90
C                                                                       SPA28 91
C     SET SOURCE AND DOUBLET DISTRIBUTION AND SHAPE DATA FOR            SPA28 92
C     STORE K EQUAL TO JTH INPUT DISTRIBUTION AND SHAPE                 SPA28 93
C                                                                       SPA28 94
      NSSOR(K)=MSOR                                                     SPA28 95
      NSPOLY(K)=NSPOLJ                                                  SPA28 96
      DO 55 I=1,NSPOLJ                                                  SPA28 97
      SXEND(I,K)=SXNDJ(I)                                               SPA28 98
      DO 56 IC=1,7                                                      SPA28 99
   56 SCOEF(I,IC,K)=SCOFJ(I,IC)                                         SPA28100
   55 CONTINUE                                                          SPA28101
      DO 5 L=1,MSOR                                                     SPA28102
      SXL(L,K)=TX(L)                                                    SPA28103
      SSS(L,K)=T(L)                                                     SPA28104
    5 SDB(L,K)=TC(L)                                                    SPA28105
      NCOUNT=NCOUNT+1                                                   SPA28106
    4 CONTINUE                                                          SPA28107
    3 CONTINUE                                                          SPA28108
C                                                                       SPA28109
C     WRITE ERROR MSG IF DATA IS MISSING FOR SOME STORE                 SPA28110
C                                                                       SPA28111
      IF (NCOUNT .EQ. NSTRS) GO TO 10                                   SPA28112
      WRITE (6,707)                                                     SPA28113
      STOP                                                              SPA28114
C                                                                       SPA28115
C     LOCATE STORES IN WING COORDINATE SYSTEM                           SPA28116
C                                                                       SPA28117
   10 CONTINUE                                                          SPA28118
```

(v) Page 22.

Figure 1.- Continued.

```
      DO 11 J=1,NSTRS                                                   SPA28119
      YY=YSN(J)                                                         SPA28120
      IF (YY .GT. 0.0) YY=-YY                                           SPA28121
      I=1                                                               SPA28122
      JLE=1                                                             SPA28123
      IF(YY.GE.Y(1)) GO TO 13                                           SPA28124
   12 I=I+1                                                             SPA28125
      JLE=NCH*(I-1) +1                                                  SPA28126
      IF(YY=Y(I)) 12,13,14                                              SPA28127
   13 X=BUC(J)+XSNC(J)*XRF(JLE)                                         SPA28128
      Y=BU(J)=YSN(J)                                                    SPA28129
      Z=BU(J)=ZSN(J)                                                    SPA28130
      GO TO 11                                                          SPA28131
   14 JLE=JLE+NCH                                                       SPA28132
      FRAC=(YY-Y(I-1))/(Y(I)-Y(I-1))                                    SPA28133
      X=BUC(J)+XSNC(J)*XRF(JLEM)+FRAC*(XRF(JLE)-XRF(JLEM))              SPA28134
      Y=BU(J)=YSN(J)                                                    SPA28135
      Z=BU(J)=ZSN(J)                                                    SPA28136
   11 CONTINUE                                                          SPA28137
      RETURN                                                            SPA28138
      END                                                               SPA28139
```

```
      SUBROUTINE STTOIN (X,Y,Z,X1,ETA,ZETA,DC)                          SPA29  1
C     SUBROUTINE TO TRANSFORM FROM STORE TO INERTIAL SYSTEM             SPA29  2
C                                                                       SPA29  3
      DIMENSION DC(3,3)                                                 SPA29  4
      X1=X*DC(1,1)+Y*DC(1,2)+Z*DC(1,3)                                  SPA29  5
      ETA=X*DC(2,1)+Y*DC(2,2)+Z*DC(2,3)                                 SPA29  6
      ZETA=X*DC(3,1)+Y*DC(3,2)+Z*DC(3,3)                                SPA29  7
      RETURN                                                            SPA29  8
      END                                                               SPA29  9
```

```
      SUBROUTINE SWNGIN                                                 SPA30  1
C                                                                       SPA30  2
C     SUBROUTINE TO INPUT WING CONSTANT U-VELOCITY PANEL                SPA30  3
C       DATA AND TWIST/CAMBER DISTRIBUTION IF ANY                       SPA30  4
C                                                                       SPA30  5
      COMMON/CAMBER/ALPHAL(200)                                         SPA30  6
      COMMON /FSGEOM/FRMAX,SSPAN,   FLTHC,BUOYPL                        SPA30  7
      COMMON/INDEX/NCW,MSW,MSWP,NPANLS,NCWS,NBDCR1,NBDCR2,NBD,NBIP,MP,  SPA30  8
     1   NCP,MSP,N1P,N2,N2P,NPTOT                                       SPA30  9
      COMMON /WGEOM/XBAR=DC,ZBAR=0,CRW,SLPWLE,SLPWTE,PSIWLE(20),PSIWTE(20),SPA30 10
     1       Y(20)                                                      SPA30 11
  701 FORMAT(10I5)                                                      SPA30 12
  706 FORMAT(8F10.0)                                                    SPA30 13
 7020 FORMAT(15X,25HLENGTH OF WING ROOT CHORD/25X,5HCRW =,F10.5,5H FEET/SPA30 14
     115X,13HWING SEMISPAN/25X,7HSSPAN =,F10.5,5H FEET)                 SPA30 15
 7040 FORMAT(I5,3F10.5)                                                 SPA30 16
 7050 FORMAT(/10X,I4,  50H U-VELOCITY PANELS ARE TO BE LAID OUT ON EACH SPA30 17
     1WING PANEL/20X,I3,20H CHORDWISE ROWS WITH,I3,12H IN EACH ROW/)    SPA30 18
 7060 FORMAT(15X,58HSPANWISE LOCATIONS OF PANEL SIDE EDGES AND SWEEP ANGSPA30 19
     1LES OF/20X,25HWING SECTION TO THE RIGHT                           SPA30 20
     2                    //26X,20HSPANWISE LE SWEEP  TE SWEEP/        SPA30 21
     3  20X,1HI,5X,9HLOCATION/28X,6HFEET,5X,7HDEGREES,5X,7HDEGREES)     SPA30 22
 7070 FORMAT(18X,I3,4X,6F10.5)                                          SPA30 23
C                                                                       SPA30 24
      READ(5,706) CRW,SSPAN                                             SPA30 25
      WRITE(6,7020) CRW,SSPAN                                           SPA30 26
      READ(5,701) NCW,MSW                                               SPA30 27
      NPANLS=NC=MSW                                                     SPA30 28
      MSWP=MSW+1                                                        SPA30 29
      DO 11 J=1,MSWP                                                    SPA30 30
   11 READ(5,7040) I,Y(I),PSIWLE(I),PSIWTE(I)                           SPA30 31
      WRITE(6,7050) NPANLS,MSW,NCW                                      SPA30 32
      WRITE(6,7060)                                                     SPA30 33
      DO 13 J=1,MSWP                                                    SPA30 34
   13 WRITE(6,7070) J,Y(J),PSIWLE(J),PSIWTE(J)                          SPA30 35
C                                                                       SPA30 36
C     INPUT TWIST AND CAMBER DISTRIBUTION, IF ANY.                      SPA30 37
C                                                                       SPA30 38
      DO 15 J=1,NPANLS                                                  SPA30 39
   15 ALPHAL(J)=0.0                                                     SPA30 40
      READ (5,701) NTAC,NUNI                                            SPA30 41
      IF (NTAC .EQ. 0) GO TO 19                                         SPA30 42
      IF (NUNI .NE. 0) GO TO 17                                         SPA30 43
      MN=0                                                              SPA30 44
      DO 16 JN=1,NPANLS,NCW                                             SPA30 45
      MN=MN+NCW                                                         SPA30 46
   16 READ (5,706) (ALPHAL(J),J=JN,MN)                                  SPA30 47
      GO TO 19                                                          SPA30 48
   17 READ (5,706) (ALPHAL(J),J=1,NCW)                                  SPA30 49
      DO 18 J=2,MSW                                                     SPA30 50
      JJ=(J-1)*NCW                                                      SPA30 51
      DO 18 K=1,NCW                                                     SPA30 52
      KK=JJ+K                                                           SPA30 53
   18 ALPHAL(KK)=ALPHAL(K)                                              SPA30 54
   19 CONTINUE                                                          SPA30 55
      RETURN                                                            SPA30 56
      END                                                               SPA30 57
```

```
      SUBROUTINE THKLYT                                                 SPA31  1
C                                                                       SPA31  2
C     SUBROUTINE TO LAY OUT WING AND PYLON THICKNESS PANELS             SPA31  3
C                                                                       SPA31  4
      COMMON /CONFIG/NFU,NPY,NSTRS,LVSWP                                SPA31  5
      COMMON /CONSTS/PI,PI2,DTOR,RTOD,FOURPI                            SPA31  6
      COMMON/INDEX/NCW,MSW,MSWP,NPANLS,NCWS,NBDCR1,NBDCR2,NBD,NBIP,MP,  SPA31  7
     1   NCP,MSP,N1P,N2,N2P,NPTOT                                       SPA31  8
      COMMON /PYGEOM/Z(20),XPLE,YPL,CRP,MP,PSIPLE,IP,BLLE,BLTE,PSLPUP,  SPA31  9
     1       CENTER                                                     SPA31 10
      COMMON /THKDAT/NCWS,NCPS,MS,MPS  ,NTHP,XRFT(400),XRBT(400),       SPA31 11
     1   XLFT(400),XLBT(400),YRCT(400),YLCT(400),ZRCT(400),ZLCT(400),   SPA31 12
     2   THETAL(400),THETPL(200),BLLBT(400),BLTET(400),DZDX(400)        SPA31 13
      COMMON /WGEOM/XBAR=DC,ZBAR=0,CRW,SLPWLE,SLPWTE,PSIWLE(20),PSIWTE(20),SPA31 14
     1       Y(20)                                                      SPA31 15
C                                                                       SPA31 16
C     LAY OUT THICKNESS PANELS ON LEFT WING PANEL                       SPA31 17
C                                                                       SPA31 18
      SLPDIF=SLPWLE-SLPWTE                                              SPA31 19
      CSIDEP=CRW                                                        SPA31 20
      WLEX=0.0                                                          SPA31 21
      ANCP=NCWS                                                         SPA31 22
C                                                                       SPA31 23
C     LOOP OVER CHORDWISE ROWS OF THICKNESS PANELS                      SPA31 24
C               I IS THE SPAN INDEX                                     SPA31 25
C                                                                       SPA31 26
      DO 140 I=2,MSWP                                                   SPA31 27
      IM1=I-1                                                           SPA31 28
      IF(LVSWP.EQ.0) GO TO 40                                           SPA31 29
      SLPWLE=TAN(PSIWLE(I)*DTOR)                                        SPA31 30
      SLPWTE=TAN(PSIWTE(I)*DTOR)                                        SPA31 31
      SLPDIF=SLPWLE-SLPWTE                                              SPA31 32
   40 CSIDE=CSIDEP+(Y(I)-Y(IM1))*SLPDIF                                 SPA31 33
      IF(I.EQ.2) GO TO 45                                               SPA31 34
      JLE=(I-1)*NCWS+1                                                  SPA31 35
      WLEX=XLFT(JLE)                                                    SPA31 36
   45 CONTINUE                                                          SPA31 37
C                                                                       SPA31 38
C     LOOP OVER THICKNESS PANELS IN THIS CHORDWISE ROW                  SPA31 39
C                                                                       SPA31 40
      DO 150 K=1,NCWS                                                   SPA31 41
```

(w) Page 23.

Figure 1.- Continued.

```
      J=(I-2)*NC+3+K
      AK=K
      AKM=K-1
C
C     SLOPES OF THICKNESS PANEL LEADING AND TRAILING EDGES
C
      SLLET(J)=SLPHLE-AKM*SLPDIF/ANC
      SLTET(J)=SLPHLE-AK*SLPDIF/ANC
C
C     CORNER POINT COORDINATES
C
      XRPT(J)=AKM*CSIDEP/ANC+PLEX
      XRBT(J)=XRPT(J)+CSIDEP/ANC
      XLPT(J)=AKM*CSIDE/ANC+(Y(I)-Y(IM))*SLPHLE+PLEX
      XLBT(J)=XLPT(J)+CSIDE/ANC
      YRCT(J)=Y(IM)
      YLCT(J)=Y(I)
      ZRCT(J)=0.0
      ZLCT(J)=0.0
  130 CONTINUE
      CSIDEP=CSIDE
  140 CONTINUE
C
C     LAY OUT THICKNESS PANELS ON PYLON IF PRESENT
C
      IF(NPY.EQ.0) RETURN
      CSIDEP=CRP
      PLEX=PLE
      INDX=(IP-2)*NC+S+1
      IF(IP.GT.1) PLEX=PLE+XLPT(INDX)
      ANCP=NCPS
C
C     LOOP OVER CHORDWISE ROWS OF PYLON THICKNESS PANELS
C
      NSPP=NSP+1
      DO 300 I=2,NSPP
      CSIDE=CRP+(Z(1)-Z(I))*PSLPDF
      IM=I-1
      IF(I.EQ.2) GO TO 201
      JLE=(I-3)*NCPS+1+NS
      PLEX=XLPT(JLE)
  201 CONTINUE
C
C     LOOP OVER PYLON THICKNESS PANELS IN THIS CHORDWISE ROW
C
      DO 250 K=1,NCPS
      J=(I-2)*NCPS+K+NS
      AKM=K-1
      AK=K
C
C     THICKNESS PANEL LEADING AND TRAILING EDGE SLOPES
C
      SLLET(J)=SLLE+AKM*PSLPDF/ANCP
      SLTET(J)=SLLE+AK*PSLPDF/ANCP
C
C     CORNER POINT COORDINATES
C
      XRPT(J)=AKM*CSIDEP/ANCP+PLEX
      XRBT(J)=XRPT(J)+CSIDEP/ANCP
      XLPT(J)=AKM*CSIDE/ANCP+(Z(IM)-Z(I))*SLLE+PLEX
      XLBT(J)=XLPT(J)+CSIDE/ANCP
      YLCT(J)=YPL
      YRCT(J)=YPL
      ZRCT(J)=Z(IM)
      ZLCT(J)=Z(I)
  250 CONTINUE
      CSIDEP=CSIDE
  300 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE THRCUT                                                  SPA32  1
C                                                                        SPA32  2
C     SUBROUTINE TO OUTPUT THICKNESS SURFACE SLOPES                      SPA32  3
C                                                                        SPA32  4
      COMMON /CONFIG/NPO,NPY,NSTNS,LVS=P                                 SPA32  5
      COMMON /CONSTS/PI,P12,DTOR,RTOD,FOURPI                             SPA32  6
      COMMON /THRDAT/NCWS,NCPS,MS,MPS  ,NTMP,ARPT(400),XRPT(400),       SPA32  7
     1   XLBT(400),XLBT(400),YRCT(400),YLCT(400),ZRCT(400),ZLCT(400),   SPA32  8
     2   THETAL(400),THETPL(200),SLLET(400),SLTET(400),DZDX(400)        SPA32  9
C                                                                        SPA32 10
  720 FORMAT(1H1,20X,58HINPUT VALUES OF THE LOCAL SURFACE SLOPE OF THE TSPA32 11
     1HICKNESS/22X52HDISTRIBUTION, FOR EACH CHORDWISE ROW THE FIRST VALUSPA32 12
     2E/27X,41HIS FOR THE PANEL NEAREST THE LEADING EDGE)               SPA32 13
  721 FORMAT(/10X,19HWING THICKNESS DATA)                              SPA32 14
  722 FORMAT(17X,5HROW=,37X,6HSLOPES)                                  SPA32 15
  723 FORMAT(15X,15,8F10.5,9(/20X,8F10.5))                             SPA32 16
  724 FORMAT(/10X,20HPYLON THICKNESS DATA)                             SPA32 17
C                                                                        SPA32 18
C     OUTPUT WING SLOPES                                                 SPA32 19
C                                                                        SPA32 20
      WRITE (6,720)                                                      SPA32 21
      WRITE (6,721)                                                      SPA32 22
      WRITE (6,722)                                                      SPA32 23
      MN=0                                                               SPA32 24
      DO 32 JN=1,MS,NC=S                                                 SPA32 25
      MN=MN+NC=S                                                         SPA32 26
      I=MN/NC=S                                                          SPA32 27
   32 WRITE (6,723) (I,(THETAL(J),J=JN,MN))                              SPA32 28
C                                                                        SPA32 29
C     OUTPUT PYLON SLOPES, IF PRESENT                                    SPA32 30
C                                                                        SPA32 31
      IF (NPY .EQ. 0) GO TO 38                                           SPA32 32
      WRITE (6,724)                                                      SPA32 33
      WRITE (6,722)                                                      SPA32 34
      MN=0                                                               SPA32 35
      DO 35 JN=1,MPS,NCPS                                                SPA32 36
      MN=MN+NCPS                                                         SPA32 37
      I=MN/NCPS                                                          SPA32 38
   35 WRITE (6,723) (I,(THETPL(J),J=JN,MN))                              SPA32 39
C                                                                        SPA32 40
C     DIVIDE INPUT SLOPES BY PI AND STORE IN DZDX ARRAY                  SPA32 41
C                                                                        SPA32 42
      DO 36 I=1,MPS                                                      SPA32 43
      IP=MS+I                                                            SPA32 44
   36 DZDX(IP)=THETPL(I)/PI                                              SPA32 45
   38 CONTINUE                                                           SPA32 46
      DO 37 I=1,MS                                                       SPA32 47
   37 DZDX(I)=THETAL(I)/PI                                               SPA32 48
      RETURN                                                             SPA32 49
      END                                                                SPA32 50
```

```
      SUBROUTINE THRCAL (T)                                              SPA33  1
C                                                                        SPA33  2
C     SUBROUTINE TO CALCULATE THRUST AT TIME=T                           SPA33  3
C                                                                        SPA33  4
      COMMON /THRUST/ NTPOLY,TEND(5),TC(5,6),FTHRUS,NTHRUS              SPA33  5
C                                                                        SPA33  6
      DO 1 K=1,NTPOLY                                                    SPA33  7
      TE=TEND(K)                                                         SPA33  8
      J=K                                                                SPA33  9
      IF (T.LE.TE) GO TO 2                                               SPA33 10
    1 CONTINUE                                                           SPA33 11
  701 FORMAT(//5X,60H***** TIME GREATER THAN END OF SPECIFIED THRUST TIMSPA33 12
     1E HISTORY)                                                         SPA33 13
      WRITE (6,701)                                                      SPA33 14
      STOP                                                               SPA33 15
    2 FTHRUS=TC(J,1)+T*(TC(J,2)+T*(TC(J,3)+T*(TC(J,4)+T*(TC(J,5)+T*TC(J,SPA33 16
     1 6)))))                                                            SPA33 17
      RETURN                                                             SPA33 18
      END                                                                SPA33 19
```


(x) Page 24.

Figure 1.- Continued.

```
      SUBROUTINE VELWD(XX,YY,ZZ)                                        SPA34   1
C                                                                       SPA34   2
C     CALCULATES PERTURBATION VELOCITIES DUE TO BODY INTERFERENCE        SPA34   3
C     PANELS USING A SUPERPOSITION SCHEME OVER FOUR CORNER SOLUTIONS     SPA34   4
C     VELOCITIES UP,VP,WP ARE RETURNED IN THE WING REFERENCE FRAME       SPA34   5
C                                                                       SPA34   6
C     IF  ITRIP, AN INFLUENCE COEFFICIENT RATHER THAN A VELOCITY IS     SPA34   7
C     CALCULATED                                                        SPA34   8
C                                                                       SPA34   9
      COMMON/BPGEOM/                    SNT2(100),CST2(100),ZLC(200),   SPA34  10
     1   ZRC(200),TMTI(100),SANG,CANG                                   SPA34  11
      COMMON/ICVEL/UP,VP,WP,II,IF,          DELTP(200)                  SPA34  12
      COMMON/INDEX/NCH,MSW,MSWP,NPANLS,NCHB,NBOCH1,NBOCH2,NBD,NWIP,MP,  SPA34  13
     1   NCP,MSP,N1P,N2,N2P,NPTOT                                       SPA34  14
      COMMON/WPGEOM/XWF(200),XRD(200),XLF(200),XLB(200),YWC(200),       SPA34  15
     1   YLC(200),XCPT(200),YCPT(200),ZCPT(200),SWPPLE(200),BWPPTE(200) SPA34  16
      COMMON /VELAMG/X,Y,Z,U,V,W,EM,TLWNC,TIPY,PYPNL                    SPA34  17
C                                                                       SPA34  18
      LOGICAL       PYPNL,FELTA,FELTAI,FELTB,FELTBI,FELT                SPA34  19
C                                                                       SPA34  20
C     THERE ARE TWO COORDINATE SYSTEMS IN USE IN THIS SUBROUTINE --     SPA34  21
C                                                                       SPA34  22
C     SYSTEM 1 = WING COORDINATE SYSTEM                                 SPA34  23
C                X,U POSITIVE FORWARD                                   SPA34  24
C                Y,V POSITIVE TO RIGHT                                  SPA34  25
C                Z,W POSITIVE DOWN                                      SPA34  26
C                                                                       SPA34  27
C     SYSTEM 2 = ORIGIN AT VERTEX OF INFLUENCING SEMI-INFINITE TRIANGLE SPA34  28
C     (REQUIRED BY SUBROUTINE VELO)                                     SPA34  29
C                X,U POSITIVE BACK                                      SPA34  30
C                Y,V POSITIVE CLOCKWISE LOOKING FORWARD, IN PLANE OF    SPA34  31
C                   INFLUENCING PANEL                                   SPA34  32
C                Z,W POSITIVE OUTWARD NORMAL TO INFLUENCING PANEL       SPA34  33
C                                                                       SPA34  34
C     VELOCITIES OR INFLUENCE COEFFICIENTS UP,VP,WP ARE IN SYSTEM 1     SPA34  35
C                                                                       SPA34  36
      PYPNL=.$ALSE.                                                     SPA34  37
      UP=0.0                                                            SPA34  38
      VP=0.0                                                            SPA34  39
      WP=0.0                                                            SPA34  40
      XI=XX                                                             SPA34  41
      YI=YY                                                             SPA34  42
      ZI=ZZ                                                             SPA34  43
      EM=0.0                                                            SPA34  44
C                                                                       SPA34  45
C     I IS THE INDEX OF THE INFLUENCING PANEL                           SPA34  46
C                                                                       SPA34  47
      DO 100 I=II,IP                                                    SPA34  48
      K=I+N2                                                            SPA34  49
      CS=CST2(K)                                                        SPA34  50
      SN=SNT2(K)                                                        SPA34  51
      FELTB=.FALSE.                                                     SPA34  52
      TU=0.0                                                            SPA34  53
      TV=0.0                                                            SPA34  54
      TW=0.0                                                            SPA34  55
      IF(XI.GE.XLF(I)) GO TO 100                                        SPA34  56
      IF((TMTI(K).GT.90..AND.TMTI(K).LE.180.)) GO TO 15                 SPA34  57
C                                                                       SPA34  58
C     ********************* CORNER 1 ************************************SPA34  59
C                                                                       SPA34  60
C                                                                       SPA34  61
C     EQUATIONS FOR PANELS IN LOWER LEFT QUADRANT                       SPA34  62
C                                                                       SPA34  63
C                                                                       SPA34  64
C     X IS IN SYSTEM 2                                                  SPA34  65
C                                                                       SPA34  66
      X=XLF(I)-XI                                                       SPA34  67
      Y=BYI-YLC(I)                                                      SPA34  68
      Z=WZI-ZLC(I)                                                      SPA34  69
C                                                                       SPA34  70
C     TRANSFORM COORDINATES YW,ZW IN SYSTEM 1 TO Y,Z IN                 SPA34  71
C     SYSTEM 2 FOR PANEL NUMBER I                                       SPA34  72
C                                                                       SPA34  73
      Y=YW*CS+ZW*SN                                                     SPA34  74
      Z=YW*SN-ZW*CS                                                     SPA34  75
      CALL VELO(FELTA)                                                  SPA34  76
      IF(.NOT.FELTA) GO TO 11                                           SPA34  77
C                                                                       SPA34  78
C     CHANGE SIGN OF U TO RETURN IT TO SYSTEM 1                         SPA34  79
C                                                                       SPA34  80
      U=-U                                                              SPA34  81
      V=V                                                               SPA34  82
      W=W                                                               SPA34  83
C                                                                       SPA34  84
C     ROTATE V,W IN SYSTEM 2 INTO V,W IN SYSTEM 1                       SPA34  85
C                                                                       SPA34  86
      V=V*CS-W*SN                                                       SPA34  87
      W=V*SN+W*CS                                                       SPA34  88
      TU=TU-U                                                           SPA34  89
      TV=TV-V                                                           SPA34  90
      TW=TW+W                                                           SPA34  91
      FELTB=.TRUE.                                                      SPA34  92
C                                                                       SPA34  93
C     INFLUENCE OF CORNER 1 ON MATCHING PANEL OF RIGHT WING             SPA34  94
C                                                                       SPA34  95
   11 Y=W-(YI+YLC(I))                                                   SPA34  96
C                                                                       SPA34  97
C     TRANSFORM YW,ZW IN SYSTEM 1 TO Y,Z, IN SYSTEM 2                   SPA34  98
C                                                                       SPA34  99
      Y=YW*CS+ZW*SN                                                     SPA34100
      Z=YW*SN-ZW*CS                                                     SPA34101
      CALL VELO(FELTAI)                                                 SPA34102
      IF(.NOT.FELTAI) GO TO 10                                          SPA34103
C                                                                       SPA34104
C     RETURN U TO SYSTEM 1                                              SPA34105
C                                                                       SPA34106
      U=-U                                                              SPA34107
      V=V                                                               SPA34108
      W=W                                                               SPA34109
C                                                                       SPA34110
C     TRANSFORM VB,WB IN SYSTEM 2 TO V,W IN SYSTEM 1                    SPA34111
C                                                                       SPA34112
      V=VB*CS+WB*SN                                                     SPA34113
      W=VB*SN-WB*CS                                                     SPA34114
      TU=TU-U                                                           SPA34115
      TV=TV-V                                                           SPA34116
      TW=TW+W                                                           SPA34117
      FELTB=.TRUE.                                                      SPA34118
      GO TO 10                                                          SPA34119
C                                                                       SPA34120
C     CORNER 1 EQUATIONS FOR PANELS IN UPPER LEFT QUADRANT              SPA34121
C                                                                       SPA34122
   15 X=XLF(I)-XI                                                       SPA34123
      Y=WYI-YLC(I)                                                      SPA34124
      Z=WZI-ZLC(I)                                                      SPA34125
      Y=YW*CS+ZW*SN                                                     SPA34126
      Z=YW*SN-ZW*CS                                                     SPA34127
      YW=Y                                                              SPA34128
      CALL VELO(FELTA)                                                  SPA34129
      IF(.NOT.FELTA) GO TO 16                                           SPA34130
      U=-U                                                              SPA34131
      VB=V                                                              SPA34132
      WB=W                                                              SPA34133
      V=VB*CS+WB*SN                                                     SPA34134
      W=VB*SN-WB*CS                                                     SPA34135
      TU=TU-U                                                           SPA34136
      TV=TV-V                                                           SPA34137
      TW=TW+W                                                           SPA34138
      FELTB=.TRUE.                                                      SPA34139
   16 Y=W-(YLC(I)+YI)                                                   SPA34140
      Y=YW*CS+ZW*SN                                                     SPA34141
      Z=YW*SN-ZW*CS                                                     SPA34142
      YW=Y                                                              SPA34143
      CALL VELO(FELTAI)                                                 SPA34144
```


(y) Page 25.

Figure 1.- Continued.

```
      IF(.NOT.FELTAI) GO TO 17                                          SPA34145
      U=-U                                                              SPA34146
      V=-V                                                              SPA34147
      A=A                                                               SPA34148
      VSV=C$*A-B*SN                                                     SPA34149
      W=V*B+SN*A+B*CS                                                   SPA34150
      TU=TU+U                                                           SPA34151
      TV=TV+V                                                           SPA34152
      TW=TW+W                                                           SPA34153
      FELTA=.TRUE.                                                      SPA34154
      GO TO 19                                                          SPA34155
C                                                                       SPA34156
C     ********** CORNER 2 ************************************************SPA34157
C     EQUATIONS FOR PANELS IN LOWER LEFT QUADRANT                       SPA34158
C                                                                       SPA34159
      XBXLB(I)=XI                                                       SPA34160
   10 YW=YI-YLC(I)                                                      SPA34161
      ZW=ZI-ZLC(I)                                                      SPA34162
      YBY=C$*Z+A*SN                                                     SPA34163
      ZBY=-SN*Z+A*CS                                                    SPA34164
      CALL VELO(FELTBI)                                                 SPA34165
      IF(.NOT. FELTBI) GO TO 12                                         SPA34166
      U=-U                                                              SPA34167
      V=V                                                               SPA34168
      W=W                                                               SPA34169
      VSV=C$*A-B*SN                                                     SPA34170
      W=V*B+SN*A+B*CS                                                   SPA34171
      TU=TU+U                                                           SPA34172
      TV=TV+V                                                           SPA34173
      TW=TW+W                                                           SPA34174
      FELTA=.TRUE.                                                      SPA34175
   12 YW=-(YI+YLC(I))                                                   SPA34176
      YBY=C$*Z+A*SN                                                     SPA34177
      ZBY=-SN*Z+A*CS                                                    SPA34178
      CALL VELO(FELTBI)                                                 SPA34179
      IF(.NOT.FELTBI) GO TO 13                                          SPA34180
      U=-U                                                              SPA34181
      V=V                                                               SPA34182
      W=W                                                               SPA34183
      VSV=C$*A-B*SN                                                     SPA34184
      W=V*B+SN*A+B*CS                                                   SPA34185
      TU=TU+U                                                           SPA34186
      TV=TV+V                                                           SPA34187
      TW=TW+W                                                           SPA34188
      FELTA=.TRUE.                                                      SPA34189
   13 IF(.NOT.FELT) GO TO 100                                           SPA34190
      GO TO 20                                                          SPA34191
C                                                                       SPA34192
C     CORNER 2 EQUATIONS FOR PANELS IN THE UPPER LEFT QUADRANT          SPA34193
C                                                                       SPA34194
C                                                                       SPA34195
   17 XBXRF(I)=XI                                                       SPA34196
      YW=YI-YRC(I)                                                      SPA34197
      ZW=ZI-ZRC(I)                                                      SPA34198
      YBY=C$*Z+B*SN                                                     SPA34199
      ZBY=-SN*Z+A*CS                                                    SPA34200
      Y=-Y                                                              SPA34201
      CALL VELO(FELTB)                                                  SPA34202
      IF(.NOT.FELTB) GO TO 24                                           SPA34203
      U=-U                                                              SPA34204
      V=-V                                                              SPA34205
      W=W                                                               SPA34206
      VSV=C$*A+B*SN                                                     SPA34207
      W=V*B+SN*A+B*CS                                                   SPA34208
      TU=TU+U                                                           SPA34209
      TV=TV+V                                                           SPA34210
      TW=TW+W                                                           SPA34211
      FELTA=.TRUE.                                                      SPA34212
   24 YW=-(YRC(I)+YI)                                                   SPA34213
      YBY=C$*Z+A*SN                                                     SPA34214
      ZBY=-SN*Z+A*CS                                                    SPA34215
      Y=-Y                                                              SPA34216
      CALL VELO(FELTAI)                                                 SPA34217
      IF(.NOT.FELTBI) GO TO 25                                          SPA34218
      U=-U                                                              SPA34219
      V=-V                                                              SPA34220
      A=A                                                               SPA34221
      VSV=C$*A+B*SN                                                     SPA34222
      W=V*B+SN*A+B*CS                                                   SPA34223
      TU=TU+U                                                           SPA34224
      TV=TV+V                                                           SPA34225
      TW=TW+W                                                           SPA34226
      FELTA=.TRUE.                                                      SPA34227
   25 IF(.NOT.FELT) GO TO 100                                           SPA34228
C                                                                       SPA34229
C     ********** CORNER 3 ************************************************SPA34230
C                                                                       SPA34231
   20 IF(XI.GE.XLB(I)) GO TO 40                                         SPA34232
      IF((THTI(K).GT.90..AND.THTI(K).LE.180)) GO TO 19                  SPA34233
C                                                                       SPA34234
C     CORNER 3 EQUATIONS FOR PANELS IN THE LOWER LEFT QUADRANT          SPA34235
C                                                                       SPA34236
      XBXLB(I)=XI                                                       SPA34237
      YW=YI-YLC(I)                                                      SPA34238
      ZW=ZI-ZLC(I)                                                      SPA34239
      YBY=C$*Z+A*SN                                                     SPA34240
      ZBY=-SN*Z+A*CS                                                    SPA34241
      CALL VELO(FELTA)                                                  SPA34242
      IF(.NOT.FELTA) GO TO 23                                           SPA34243
      U=-U                                                              SPA34244
      V=-V                                                              SPA34245
      W=W                                                               SPA34246
      VSV=C$*A+B*SN                                                     SPA34247
      W=V*B+SN*A+B*CS                                                   SPA34248
      TU=TU+U                                                           SPA34249
      TV=TV+V                                                           SPA34250
      TW=TW+W                                                           SPA34251
   23 YW=-(YI+YLC(I))                                                   SPA34252
      YBY=C$*Z+A*SN                                                     SPA34253
      ZBY=-SN*Z+A*CS                                                    SPA34254
      CALL VELO(FELTAI)                                                 SPA34255
      IF(.NOT.FELTAI) GO TO 30                                          SPA34256
      U=-U                                                              SPA34257
      V=-V                                                              SPA34258
      W=W                                                               SPA34259
      VSV=C$*A+B*SN                                                     SPA34260
      W=V*B+SN*A+B*CS                                                   SPA34261
      TU=TU+U                                                           SPA34262
      TV=TV+V                                                           SPA34263
      TW=TW+W                                                           SPA34264
      GO TO 30                                                          SPA34265
C                                                                       SPA34266
C     CORNER 3 EQUATIONS FOR PANELS IN UPPER LEFT QUADRANT              SPA34267
C                                                                       SPA34268
   19 XBXLB(I)=XI                                                       SPA34269
      YW=YI-YLC(I)                                                      SPA34270
      ZW=ZI-ZLC(I)                                                      SPA34271
      YBY=C$*Z+A*SN                                                     SPA34272
      ZBY=-SN*Z+A*CS                                                    SPA34273
      Y=-Y                                                              SPA34274
      CALL VELO(FELTA)                                                  SPA34275
      IF(.NOT.FELTA) GO TO 26                                           SPA34276
      U=-U                                                              SPA34277
      V=-V                                                              SPA34278
      W=W                                                               SPA34279
      VSV=C$*A+B*SN                                                     SPA34280
      W=V*B+SN*A+B*CS                                                   SPA34281
      TU=TU+U                                                           SPA34282
      TV=TV+V                                                           SPA34283
      TW=TW+W                                                           SPA34284
   26 YW=-(YLC(I)+YI)                                                   SPA34285
      YBY=C$*Z+A*SN                                                     SPA34286
      ZBY=-SN*Z+A*CS                                                    SPA34287
      Y=-Y                                                              SPA34288
      CALL VELO(FELTAI)
```


(z) Page 26.

Figure 1.- Continued.

```
      IF(.NOT.,FELTAI) GO TO 21                                          SPA34289
      U=-U                                                               SPA34290
      V=-V                                                               SPA34291
      W=W                                                                SPA34292
      V=V*C$+W*S$                                                        SPA34293
      W=V*S$-W*C$                                                        SPA34294
      TU=TU+U                                                            SPA34295
      TV=TV+V                                                            SPA34296
      TW=TW+W                                                            SPA34297
      GO TO 21                                                           SPA34298
C                                                                        SPA34299
C     ********************  CORNER 4  ************************************SPA34300
C     CORNER 4 EQUATIONS FOR PANELS IN THE LOWER LEFT QUADRANT           SPA34301
C                                                                        SPA34302
   30 X=XRB(I)-XI                                                        SPA34303
      Y=-YI-YRC(I)                                                       SPA34304
      Z=-ZI-ZRC(I)                                                       SPA34305
      XT=-C$*Z-*S$                                                       SPA34306
      ZT=-S$-Z-*C$                                                       SPA34307
      CALL VELU(FELTB)                                                   SPA34308
      IF(.NOT.,FELTB) GO TO 35                                           SPA34309
      U=-U                                                               SPA34310
      V=V                                                                SPA34311
      W=W                                                                SPA34312
      V=V*C$+W*S$                                                        SPA34313
      W=V*S$-W*C$                                                        SPA34314
      TU=TU+U                                                            SPA34315
      TV=TV+V                                                            SPA34316
      TW=TW+W                                                            SPA34317
   35 Y=-(YI+YRC(I))                                                     SPA34318
      XT=-C$*Z-*S$                                                       SPA34319
      ZT=-S$-Z-*C$                                                       SPA34320
      CALL VELU(FELTBI)                                                  SPA34321
      IF(.NOT.,FELTBI) GO TO 40                                          SPA34322
      U=-U                                                               SPA34323
      V=V                                                                SPA34324
      W=W                                                                SPA34325
      V=V*C$+W*S$                                                        SPA34326
      W=V*S$-W*C$                                                        SPA34327
      TU=TU+U                                                            SPA34328
      TV=TV+V                                                            SPA34329
      TW=TW+W                                                            SPA34330
      GO TO 40                                                           SPA34331
C                                                                        SPA34332
C     CORNER 4 EQUATIONS FOR PANELS IN UPPER LEFT QUADRANT               SPA34333
C                                                                        SPA34334
   21 X=XRB(I)-XI                                                        SPA34335
      Y=-YI-YRC(I)                                                       SPA34336
      Z=-ZI-ZRC(I)                                                       SPA34337
      XT=-C$*Z-*S$                                                       SPA34338
      ZT=-S$-Z-*C$                                                       SPA34339
      Y=-Y                                                               SPA34340
      CALL VELU(FELTB)                                                   SPA34341
      IF(.NOT.,FELTB) GO TO 28                                           SPA34342
      U=-U                                                               SPA34343
      V=-V                                                               SPA34344
      W=W                                                                SPA34345
      V=V*C$+W*S$                                                        SPA34346
      W=V*S$-W*C$                                                        SPA34347
      TU=TU+U                                                            SPA34348
      TV=TV+V                                                            SPA34349
      TW=TW+W                                                            SPA34350
   28 Y=-(YI+YRC(I))                                                     SPA34351
      XT=-C$*Z+*S$                                                       SPA34352
      ZT=-S$-Z-*C$                                                       SPA34353
      Y=-Y                                                               SPA34354
      CALL VELU(FELTBI)                                                  SPA34355
      IF(.NOT.,FELTBI) GO TO 40                                          SPA34356
      U=-U                                                               SPA34357
      V=-V                                                               SPA34358
      W=W                                                                SPA34359
      V=V*C$+W*S$                                                        SPA34360
      W=V*S$-W*C$                                                        SPA34361
      TU=TU+U                                                            SPA34362
      TV=TV+V                                                            SPA34363
      TW=TW+W                                                            SPA34364
   40 IF(I1.EQ.IF) GO TO 101                                             SPA34365
C                                                                        SPA34366
C     CALCULATION OF PERTURBATION VELOCITIES                             SPA34367
C                                                                        SPA34368
      UP=UP+TU+DELTP(I)                                                  SPA34369
      VP=VP+TV+DELTP(I)                                                  SPA34370
      WP=WP+TW+DELTP(I)                                                  SPA34371
  100 CONTINUE                                                           SPA34372
      RETURN                                                             SPA34373
C                                                                        SPA34374
C     RETURN INFLUENCE COEFFICIENTS IF I1=IF                             SPA34375
C                                                                        SPA34376
  101 UP=TU                                                              SPA34377
      VP=TV                                                              SPA34378
      WP=TW                                                              SPA34379
      RETURN                                                             SPA34380
      END                                                                SPA34381
```

```
      SUBROUTINE VELCAL(T,TC,TX,N,XP,Y,ZP,U1,V1,W1)                      SPA35  1
C                                                                        SPA35  2
C SUBROUTINE TO CALCULATE THE VELOCITIES FOR THE FIELD POINT X,Y,Z DUE   SPA35  3
C TO THE BODY SINGULARITIES.                                             SPA35  4
C                                                                        SPA35  5
C     XP,Y,ZP = FIELD POINT COORDINATES WHERE VELOCITY                   SPA35  6
C               IS TO BE CALCULATED                                      SPA35  7
C     XP,Y,ZP IN BODY SYSTEM                                             SPA35  8
C          XP = POSITIVE FORWARD                                         SPA35  9
C          Y  = POSITIVE RIGHT                                           SPA35 10
C          ZP = POSITIVE DOWN                                            SPA35 11
C     U1,V1,W1 = VELOCITIES ACCORDING TO THE FOLLOWING SIGN CONVENTION   SPA35 12
C           U1 = POSITIVE FORWARD                                        SPA35 13
C           V1 = POSITIVE TO RIGHT                                       SPA35 14
C           W1 = POSITIVE DOWN                                           SPA35 15
C                                                                        SPA35 16
      DIMENSION T(100),TC(100),TX(101)                                   SPA35 17
C                                                                        SPA35 18
      COMMON/BRCE/XFIELD,RFIELD,BBB,T2,     U,V,VT                       SPA35 19
      COMMON/FLOW/ DUM1(5),BETA,BETASQ,DUM2                              SPA35 20
C                                                                        SPA35 21
C     TRANSFORM FIELD POINT COORDINATES TO VELCAL SYSTEM                 SPA35 22
C                                                                        SPA35 23
      X=-XP                                                              SPA35 24
      Z=-ZP                                                              SPA35 25
      XFIELD=X                                                           SPA35 26
      RFIELD=SQRT(Y*Y+Z*Z)                                               SPA35 27
      THETA=ATAN2(Y,Z)                                                   SPA35 28
C                                                                        SPA35 29
C CALCULATION OF AXIAL, RADIAL AND TANGENTIAL PERTURBATION VELOCITIES    SPA35 30
C                                                                        SPA35 31
      US=0.                                                              SPA35 32
      VS=0.                                                              SPA35 33
      UD=0.                                                              SPA35 34
      VD=0.                                                              SPA35 35
      VTD=0.                                                             SPA35 36
      DO 110 J=1,N                                                       SPA35 37
      X1=XFIELD-TX(J)                                                    SPA35 38
      RR=BETA*RFIELD                                                     SPA35 39
      IF(X1.LE.RR) GO TO 115                                             SPA35 40
      XL=X1/BR                                                           SPA35 41
      D23=SQRT(XL*XL-1.)                                                 SPA35 42
      ACOSH=ALOG(XL+D23)                                                 SPA35 43
      U=ACOSH                                                            SPA35 44
      V=BETA*D23                                                         SPA35 45
```


Figure 1.- Continued.

```
      US=US+T(J)*U                                                      SPA35 46
      VS=VS+T(J)*V                                                      SPA35 47
      U=V                                                               SPA35 48
      XX=XL*.023                                                        SPA35 49
      VR=.5*BETABU*(ACOSH*XX)                                           SPA35 50
      VT=.5*BETABU*(ACOSH*XX)                                           SPA35 51
      UD=UD+U*TC(J)                                                     SPA35 52
      VD=VD+V*TC(J)                                                     SPA35 53
  110 VTD=VTD+VT*TC(J)                                                  SPA35 54
C                                                                       SPA35 55
C TRANSFORMATION OF VELOCITIES INTO ORIGINAL COORDINATE SYSTEM...I.E.... SPA35 56
C U1,V1,W1 RATHER THAN U,VR,VTHETA                                      SPA35 57
C                                                                       SPA35 58
  115 CONTINUE                                                          SPA35 59
      COSTH=COS(THETA)                                                  SPA35 60
      U1=US*UD*COSTH                                                    SPA35 61
      VR=VD*COSTH*VS                                                    SPA35 62
      SINTH=SIN(THETA)                                                  SPA35 63
      VTD=VTD*SINTH                                                     SPA35 64
      V1=VR*SINTH+VTD*COSTH                                             SPA35 65
      W1=VTD*SINTH+VR*COSTH                                             SPA35 66
      RETURN                                                            SPA35 67
      END                                                               SPA35 68
```

```
      SUBROUTINE VELO(R3LT)                                             SPA36  1
C                                                                       SPA36  2
C     THIS SUBROUTINE CALCULATES THE INFLUENCE OF A                     SPA36  3
C     SEMI-INFINITE TRIANGLE UNDER CONSTANT U DIFFERENCE, U*/VINF       SPA36  4
C                                                                       SPA36  5
      LOGICAL PYPNL,R3LT,INSIDE                                         SPA36  6
      EQUIVALENCE(EM,EML)                                               SPA36  7
C                                                                       SPA36  8
      COMMON/NUFLOW/FNUMCH,BETANU,INUMCH,IBMTH,XTS,XTW                  SPA36  9
      COMMON /CONSTS/PI,PI2,DTOR,RTOD,FOURPI                            SPA36 10
      COMMON /FLOW/ALFACR,GAMF,FMACH,RHO,VINF,BETAUL,BTSQUL,FMCHSQ      SPA36 11
      COMMON/ICVEL/UP,VP,WP,II,IF,          DELTP(200)                  SPA36 12
      COMMON /VELARG/X,YS,ZS,U,V,W,EM,TLRNC,TIPY,PYPNL                  SPA36 13
C                                                                       SPA36 14
C     FUNCTION STATEMENTS FOR COMPUTING F1,FQ,FS,F7                     SPA36 15
C                                                                       SPA36 16
      FN1(X)=ATAN2(Z*ROOTA1,EML*YZSQ*X*Y)                               SPA36 17
      FN07(Y)=(Y/YZSQ)*ROOTA1                                           SPA36 18
      FN5(X)=ALOG((X*ROOTA1)/(BETA*SQRT(YZSQ)))                         SPA36 19
C                                                                       SPA36 20
      IF (INUMCH.EQ.1.AND..NOT.PYPNL) GO TO 2                           SPA36 21
      BETA=BETAUL                                                       SPA36 22
      BTSQ=BTSQUL                                                       SPA36 23
      GO TO 3                                                           SPA36 24
    2 BETA=BETANU                                                       SPA36 25
      BTSQ=BETA*BETA                                                    SPA36 26
    3 CONTINUE                                                          SPA36 27
C                                                                       SPA36 28
      FELT=.TRUE.                                                       SPA36 29
      IF(X.GE.TLRNC) GO TO 5                                            SPA36 30
C                                                                       SPA36 31
C     SET VELOCITIES EQUAL TO ZERO FOR WHATEVER REASON                  SPA36 32
C                                                                       SPA36 33
    1 U=0.                                                              SPA36 34
      V=0.                                                              SPA36 35
      W=0.                                                              SPA36 36
      FELT=.FALSE.                                                      SPA36 37
      RETURN                                                            SPA36 38
C                                                                       SPA36 39
C     CHECK IF INFLUENCING PANEL IS ON THE PYLON.                       SPA36 40
C     IF SO, Y=ZS AND Z=-YS                                             SPA36 41
C                                                                       SPA36 42
    5 IF(PYPNL) GO TO 10                                                SPA36 43
      Y=YS                                                              SPA36 44
      Z=ZS                                                              SPA36 45
      GO TO 15                                                          SPA36 46
   10 Y=ZS                                                              SPA36 47
      Z=-YS                                                             SPA36 48
   15 YSQ=Y*Y                                                           SPA36 49
      ZSQ=Z*Z                                                           SPA36 50
      YZSQ=YSQ+ZSQ                                                      SPA36 51
      EMLSQ=EML*EML                                                     SPA36 52
      ARG1=X*X-BTSQ*YZSQ                                                SPA36 53
      INSIDE=ARG1.GT.TLRNC                                              SPA36 54
C                                                                       SPA36 55
C     CHECK FOR UNSWEPT LEADING EDGE                                    SPA36 56
C                                                                       SPA36 57
      IF(ABS(EML).LT.TLRNC) GO TO 40                                    SPA36 58
      YYEDGE=X/EML                                                      SPA36 59
C                                                                       SPA36 60
C     CHECK WHETHER LEADING EDGE IS SUBSONIC,SONIC, OR SUPERSONIC.      SPA36 61
C                                                                       SPA36 62
      STEST=BTSQ-EMLSQ                                                  SPA36 63
      IF(ABS(STEST).LT.TLRNC) GO TO 20                                  SPA36 64
      IF(STEST.GT.0) GO TO 30                                           SPA36 65
C                                                                       SPA36 66
C*************** SUBSONIC LEADING EDGE ********************************SPA36 67
C                                                                       SPA36 68
C     DETERMINE WHETHER POINT LIES INSIDE MACH CONE FROM ORIGIN.        SPA36 69
C     IF IT LIES OUTSIDE, SET PERTURBATION VELOCITIES TO ZERO.          SPA36 70
C                                                                       SPA36 71
      IF(.NOT.INSIDE) GO TO 1                                           SPA36 72
C                                                                       SPA36 73
C     INSIDE CONE FROM ORIGIN.                                          SPA36 74
C                                                                       SPA36 75
C     TEST FOR POSSIBLE SINGULARITIES IN FUNCTIONS.                     SPA36 76
C                                                                       SPA36 77
      IF(ABS(Y).LT.TLRNC.AND.ABS(Z).LT.TLRNC) GO TO 1                   SPA36 78
      IF(ABS(YYEDGE).LT.TLRNC.AND.ABS(Z).LT.TLRNC) GO TO 1              SPA36 79
      ROOTA1=SQRT(ARG1)                                                 SPA36 80
      T1=X*EML-BTSQ*Y                                                   SPA36 81
      T2=BTSQ*((Y*EML-X)**2+ZSQ*(EMLSQ-BTSQ))                           SPA36 82
      F2=(EML/SQRT(EMLSQ-BTSQ))*ALOG((T1+SQRT(T1*T1-T2))/SQRT(T2))      SPA36 83
      F1=FN1(E)                                                         SPA36 84
      FQ=FN07(Y)                                                        SPA36 85
      F5=FN5(E)                                                         SPA36 86
      F7=FN07(Z)                                                        SPA36 87
      GO TO 100                                                         SPA36 88
C                                                                       SPA36 89
C*************** SONIC LEADING EDGE ***********************************SPA36 90
C                                                                       SPA36 91
C     DETERMINE WHETHER POINT LIES INSIDE MACH CONE FROM ORIGIN.        SPA36 92
C     IF OUTSIDE, SET PERTURBATION VELOCITIES TO ZERO.                  SPA36 93
C                                                                       SPA36 94
   20 IF(.NOT.INSIDE) GO TO 1                                           SPA36 95
      IF(ABS(Y).LT.TLRNC.AND.ABS(Z).LT.TLRNC) GO TO 25                  SPA36 96
      IF(ABS(YYEDGE).LT.TLRNC.AND.ABS(Z).LT.TLRNC) GO TO 25             SPA36 97
      ROOTA1=SQRT(ARG1)                                                 SPA36 98
      F2=ROOTA1/(X-BETA*Y)                                              SPA36 99
      F1=FN1(E)                                                         SPA36100
      FQ=FN07(Y)                                                        SPA36101
      F5=FN5(E)                                                         SPA36102
      F7=FN07(Z)                                                        SPA36103
      GO TO 100                                                         SPA36104
C                                                                       SPA36105
C     SINGULAR OR INDETERMINATE VELOCITIES                              SPA36106
C         AT X=AXIS OR LEADING EDGE                                     SPA36107
C         SET U, V, AND W TO ZERO                                       SPA36108
C                                                                       SPA36109
   25 GO TO 1                                                           SPA36110
C                                                                       SPA36111
C*************** SUPERSONIC LEADING EDGE ******************************SPA36112
C                                                                       SPA36113
C     DETERMINE WHETHER POINT LIES INSIDE MACH CONE FROM ORIGIN.        SPA36114
C     IF OUTSIDE, THERE IS ONE MORE CHECK TO MAKE.                      SPA36115
C                                                                       SPA36116
```

(bb) Page 28.

Figure 1.- Continued.

```
   30 IF(INSIDE) GO TO 35                                                SPA36117
C                                                                        SPA36118
C     POINT IS OUTSIDE MACH CONE FROM ORIGIN,                            SPA36119
C     DETERMINE IF IT IS INSIDE CONE FROM LEADING EDGE                   SPA36120
C     IF OUTSIDE, SET PERTURBATION VELOCITIES TO ZERO.                   SPA36121
C                                                                        SPA36122
      IF(Y.LT.0.0.OR.YYEDGE.GE.0.0) GO TO 1                              SPA36123
      XLE=Y*EML                                                          SPA36124
      XTRNSP=X-XLE                                                       SPA36125
      RBTB=ABS(Z)*BETA                                                   SPA36126
      IF((RBTB.LT.TLRNC.AND.X.LE.XLE).OR.XTRNSP.LE.RBTB) GO TO 1         SPA36127
C                                                                        SPA36128
C     INSIDE MACH CONE FROM LEADING EDGE                                 SPA36129
C                                                                        SPA36130
      F1=PI                                                              SPA36131
      IF(Z.LT.0.0) F1=-PI                                                SPA36132
      F2=PI*EML/SQRT(BTSQ-EMLSQ)                                         SPA36133
      F4=0.                                                              SPA36134
      F5=0.                                                              SPA36135
      F7=0.                                                              SPA36136
      GO TO 100                                                          SPA36137
C                                                                        SPA36138
C     INSIDE MACH CONE FROM ORIGIN                                       SPA36139
C                                                                        SPA36140
   35 CONTINUE                                                           SPA36141
      IF(ABS(Y).LT.TLRNC.AND.ABS(Z).LT.TLRNC) GO TO 38                   SPA36142
      IF(ABS(YYEDGE).LT.TLRNC.AND.ABS(Z).LT.TLRNC) GO TO 3A              SPA36143
      ROOTA1=SQRT(ANG1)                                                  SPA36144
      T3=SQRT(RTSQ-EMLSQ)                                                SPA36145
      F2=(EML/T3)*ATAN2(T3*ROOTA1,X*EML-RTSQ*Y)                          SPA36146
      F1=FN1(X)                                                          SPA36147
      F4=FN47(Y)                                                         SPA36148
      F5=FN5(Z)                                                          SPA36149
      F7=FN47(Z)                                                         SPA36150
      GO TO 100                                                          SPA36151
C                                                                        SPA36152
C     SINGULAR OR INDETERMINATE VELOCITIES                               SPA36153
C         AT X=AXIS OR LEADING EDGE                                      SPA36154
C         SET U, V, AND W TO ZERO                                        SPA36155
C                                                                        SPA36156
   38 GO TO 1                                                            SPA36157
C                                                                        SPA36158
C************** UNSWEPT LEADING EDGE ********************************** SPA36159
C                                                                        SPA36160
C     DETERMINE WHETHER POINT LIES INSIDE MACH CONE FROM ORIGIN,         SPA36161
C     IF OUTSIDE, THERE IS ONE MORE CHECK TO MAKE                        SPA36162
C                                                                        SPA36163
   40 IF(INSIDE) GO TO 45                                                SPA36164
C                                                                        SPA36165
C     POINT IS OUTSIDE MACH CONE FROM ORIGIN,                            SPA36166
C     DETERMINE IF IT IS INSIDE CONE FROM LEADING EDGE                   SPA36167
C     IF OUTSIDE, SET PERTURBATION VELOCITIES TO ZERO.                   SPA36168
C                                                                        SPA36169
      IF(Y.LT.0.0) GO TO 1                                               SPA36170
      RBTB=ABS(Z)*BETA                                                   SPA36171
      IF(X.LE.RBTB) GO TO 1                                              SPA36172
C                                                                        SPA36173
C     INSIDE MACH CONE FROM LEADING EDGE                                 SPA36174
C                                                                        SPA36175
      F1=PI                                                              SPA36176
      IF(Z.LT.0.0) F1=-PI                                                SPA36177
      F2=-BETA*PI                                                        SPA36178
      F4=0.                                                              SPA36179
      F5=0.                                                              SPA36180
      F7=0.                                                              SPA36181
      GO TO 46                                                           SPA36182
C                                                                        SPA36183
C     INSIDE MACH CONE FROM ORIGIN                                       SPA36184
C                                                                        SPA36185
   45 CONTINUE                                                           SPA36186
      ROOTA1=SQRT(ANG1)                                                  SPA36187
      IF(ABS(Y).LT.TLRNC.AND.ABS(Z).LT.TLRNC) GO TO 1                    SPA36188
      F1=ATAN2(Z*ROOTA1,-X*Y)                                            SPA36189
      F2=-BETA*ATAN2(ROOTA1,-BETA*Y)                                     SPA36190
      F4=FN47(Y)                                                         SPA36191
      F5=FN5(Z)                                                          SPA36192
      F7=FN47(Z)                                                         SPA36193
C                                                                        SPA36194
C     VELOCITIES FOR UNSWEPT LE CASE                                     SPA36195
C                                                                        SPA36196
   46 U=F1                                                               SPA36197
      V=F7                                                               SPA36198
      W=F2-F4                                                            SPA36199
      GO TO 105                                                          SPA36200
C                                                                        SPA36201
C************** CALCULATE PERTURBATION VELOCITIES U,V,W ***************SPA36202
C                                                                        SPA36203
  100 U=F1                                                               SPA36204
      V=-EML*F1+F7                                                       SPA36205
      W=F1*((1.0-BTSQ/EMLSQ)*F2-F5)-F4                                   SPA36206
C                                                                        SPA36207
C     TRANSFORM V AND W FOR PYLON PANELS                                 SPA36208
C                                                                        SPA36209
  105 IF(.NOT. PYPNL) GO TO 110                                          SPA36210
      TEMP=V                                                             SPA36211
      V=W                                                                SPA36212
      W=TEMP                                                             SPA36213
  110 CONTINUE                                                           SPA36214
      RETURN                                                             SPA36215
      END                                                                SPA36216
```

```
      SUBROUTINE VELOTH(FELT)                                            SPA37  1
C                                                                        SPA37  2
C     THIS SUBROUTINE CALCULATES THE INFLUENCE OF A SEMI-INFINITE        SPA37  3
C     TRIANGLE ASSOCIATED WITH A THICKNESS SOURCE PANEL                  SPA37  4
C                                                                        SPA37  5
      LOGICAL PYPNL,INSIDE,FELT                                          SPA37  6
      COMMON /CONSTS/PI,PI2,DTOR,RTOD,FOURPI                             SPA37  7
      COMMON /FLOW/ALFACR,GAMF,FMACH,RHO,VINF,BETASQ,BTSQOL,FMCHSQ       SPA37  8
      COMMON /THVARG/XTH,YTH,ZTH,UTH,VTH,WTH,EML,PRT                     SPA37  9
      COMMON /VELARG/DUM(7),TLRNC,TIPY,PYPNL                             SPA37 10
      COMMON/NUFLOW/FNUMCH,BETANU,INUMCH,ISWTH,XT3,XT4                   SPA37 11
C                                                                        SPA37 12
      EQUIVALENCE(XTH,X)                                                 SPA37 13
C                                                                        SPA37 14
      IF (INUMCH.EQ.1.AND..NOT.PYPNL) GO TO 6                            SPA37 15
      BETA=BETASQL                                                       SPA37 16
      BTSQ=BTSQOL                                                        SPA37 17
      GO TO 7                                                            SPA37 18
    6 BETA=BETANU                                                        SPA37 19
      BTSQ=BETA*BETA                                                     SPA37 20
    7 CONTINUE                                                           SPA37 21
      FELT=.TRUE.                                                        SPA37 22
      IF(X.GE.TLRNC) GO TO 5                                             SPA37 23
C                                                                        SPA37 24
C     SET VELOCITIES TO ZERO FOR WHATEVER REASON                         SPA37 25
C                                                                        SPA37 26
    1 UTH=0.0                                                            SPA37 27
      VTH=0.0                                                            SPA37 28
      WTH=0.0                                                            SPA37 29
      FELT=.FALSE.                                                       SPA37 30
      RETURN                                                             SPA37 31
C                                                                        SPA37 32
C     CHECK IF THE INFLUENCING PANEL IS ON THE PYLON                     SPA37 33
C     IF SO, Y=ZTH AND Z=YTH                                             SPA37 34
C                                                                        SPA37 35
    5 IF(PYPNL) GO TO 4                                                  SPA37 36
      Y=YTH                                                              SPA37 37
      Z=ZTH                                                              SPA37 38
      GO TO 5                                                            SPA37 39
```

(cc) Page 29.

Figure 1.- Continued.

```
    4 YBZTH
      ZBBYTH
    5 YSUBYBY
      ZSUBZBZ
      EMLSQBEML*EML
      AMGIBY*SBRTSQ*(YSQ+ZSQ)
      INSIDEBAMG1 .GT. TLRNC
C
C     CHECK FOR SPECIAL CASE OF UNSWEPT LEADING EDGE
C
      IF(ABS(EML),LT,TLRNC) GO TO 70
      YEDGEBX/EML
      ARGYBY-YEDGE
C
C     CHECK WHETHER LEADING EDGE IS SUBSONIC, SONIC, OR SUPERSONIC
C
      STESTBRTSQ-EMLSQ
      IF(ABS(RTEST),LT,TLRNC) GO TO 20
      IF(STEST,GT,0,0) GO TO 30
C
C********** SUBSONIC LEADING EDGE **********
C
C     DETERMINE IF POINT LIES INSIDE OR OUTSIDE MACH CONE FROM ORIGIN
C     IF IT LIES OUTSIDE, SET PERTURBATION VELOCITIES TO ZERO
C
      IF(.NOT.INSIDE) GO TO 1
C
C     POINT LIES INSIDE MACH CONE FROM ORIGIN
C
      RADBSQRT(ARG1)
      T1BX*EML-BTSQ*Y
      T2BBETA*SQRT(((Y*EML-X)*(Y*EML+X)+ZSQ*(EMLSQ-BTSQ))
C
C     TEST FOR POSSIBLE SINGULARITY IN F1 OR F2
C
      IF(ABS(ARGY),LT,TLRNC,AND,ABS(Z),LT,TLRNC) GO TO 1
      F2B(EML/SQRT(EMLSQ-BTSQ))*ALOG((T1+SQRT(T1*T1-T2*T2))/T2)
      IF(ABS(Y),LT,TLRNC,AND,ABS(Z),LT,TLRNC) GO TO 13
      F1BATAN2(Z*RAD,EML*(YSQ+ZSQ)-Y*X)
      F5BALOG((X+RAD)/(BETA*SQRT(YSQ+ZSQ)) )
      GO TO 100
C
C     CASE FOR Y AND Z BOTH SMALL
C
   13 F1B0.0
      VTHB0.0
      GO TO 101
C
C********** SONIC LEADING EDGE **********
C
C     DETERMINE WHETHER POINT LIES INSIDE MACH CONE FROM ORIGIN
C     IF OUTSIDE, SET PERTURBATION VELOCITIES TO ZERO
C
   20 IF(.NOT.INSIDE) GO TO 1
      RADBSQRT(ARG1)
      F2BRAD/(X-BETA*Y)
C
C     F1 AND B5 SAME AS IN SUBSONIC LEADING EDGE
C
      IF (ABS(Y) .LT. TLRNC .AND. ABS(Z) .LT. TLRNC) GO TO 21
      F5BALOG((X+RAD)/(BETA*SQRT(YSQ+ZSQ)))
      IF(ABS(ARGY),LT,TLRNC,AND,ABS(Z),LT,TLRNC) GO TO 22
      F1BATAN2(Z*RAD,EML*(YSQ+ZSQ)-Y*X)
      GO TO 100
C
C     Y AND Z BOTH SMALL
C
   21 F1B0.
      VTH B 0.
      GO TO 101
C
C     Z SMALL AND Y CLOSE TO LEADING EDGE
C
      22 F1B0.
         GO TO 100
C
C********** SUPERSONIC LEADING EDGE **********
C
C     DETERMINE WHETHER POINT LIES INSIDE MACH CONE FROM ORIGIN
C     IF OUTSIDE, THERE IS ONE MORE CHECK TO BE MADE
C
   30 IF(INSIDE) GO TO 31
C
C     POINT IS OUTSIDE MACH CONE FROM ORIGIN
C     DETERMINE IF IT IS INSIDE MACH CONE FROM LEADING EDGE
C     IF OUTSIDE, SET PERTURBATION VELOCITIES TO ZERO
C
      IF(Y .LT. 0.0 .OR. Y .GE. YEDGE) GO TO 1
      XLEBY*EML
      RSTBBABS(Z)*BETA
      XTHNSPBBXLE
      IF((RSTH,LT,TLRNC,AND,X,LE,XLE),OR,(THNSP,LE,RSTH) GO TO 1
C
C     POINT IS INSIDE MACH CONE FROM LEADING EDGE
C
      F2B PI*EML/SQRT(RTSQ-EMLSQ)
      F1BPI
      IF(Z,LT,0,0) F1B-PI
      F5B0.0
      GO TO 100
C
C     POINT IS INSIDE MACH CONE FROM ORIGIN
C
   31 T3BSQRT(RTSQ-EMLSQ)
      RADBSQRT(ARG1)
      F2B(EML/T3)*ATAN2(RAD*T3,X*EML-BTSQ*Y)
C
C     USE F1 AND B5 AS IN SUBSONIC LEADING EDGE CASE
C
      IF (ABS(Y) .LT. TLRNC .AND. ABS(Z) .LT. TLRNC) GO TO 32
      F5BALOG((X+RAD)/(BETA*SQRT(YSQ+ZSQ)))
      IF(ABS(ARGY),LT,TLRNC,AND,ABS(Z),LT,TLRNC) GO TO 33
      F1BATAN2(Z*RAD,EML*(YSQ+ZSQ)-Y*X)
      GO TO 100
C
C     Y AND Z BOTH SMALL
C
   32 F1B0.
      VTHB0.
      GO TO 101
C
C     Z SMALL AND Y CLOSE TO LEADING EDGE
C
   33 F1B0.
      GO TO 100
C
C********** SPECIAL CASE FOR UNSWEPT LEADING EDGE **********
C
C     DETERMINE WHETHER POINT LIES INSIDE MACH CONE FROM ORIGIN
C     IF OUTSIDE, THERE IS ONE MORE CHECK TO BE MADE
C
   70 IF(INSIDE) GO TO 72
C
C     POINT IS OUTSIDE MACH CONE FROM ORIGIN
C     DETERMINE IF IT IS INSIDE MACH CONE FROM LEADING EDGE
C     IF OUTSIDE, SET PERTURBATION VELOCITIES TO ZERO
C
      IF(Y .LT. 0.) GO TO 1
      RSTBBAMS(Z)*BETA
      XTESTBX-RSTH
      IF((RSTH .LT. TLRNC .AND. X.LE.0.),OR.XTEST.LE.TLRNC)GO TO 1
C
C     POINT BETWEEN MACH CONE FROM ORIGIN AND LEADING EDGE
C
      VTHB-PI/BETA
```

(dd) Page 30.

Figure 1.- Continued.

```
      F1=PI                                                              SPA37185
      IF(Z.LT.0.0) F1=-PI                                                SPA37186
      VTH=0.0                                                            SPA37187
      WTH=F1                                                             SPA37188
      GO TO 102                                                          SPA37189
C                                                                        SPA37190
C     POINT IS INSIDE MACH CONE FROM ORIGIN                              SPA37191
C                                                                        SPA37192
   72 RAD=SQRT(ARG1)                                                     SPA37193
      UTH=-ATAN2(RAD,-BETA*Y)/BETA                                       SPA37194
      F2=0.0                                                             SPA37195
      IF(ABS(Z) .LT. TLRNC .AND. ABS(Y) .LT. TLRNC)GO TO 73              SPA37196
      F1=ATAN2(Z*RAD,-Y*X)                                               SPA37197
      F2=ALOG((X+RAD)/(BETA*SQRT(Y*Y+Z*Z)))                              SPA37198
      VTH=-F2                                                            SPA37199
      WTH=F1                                                             SPA37200
      GO TO 102                                                          SPA37201
   73 VTH=0                                                              SPA37202
      WTH=0                                                              SPA37203
      GO TO 102                                                          SPA37204
C                                                                        SPA37205
C     COMPUTE PERTURBATION VELOCITIES                                    SPA37206
C                                                                        SPA37207
  100 VTH=F2-F5                                                          SPA37208
  101 UTH=F2/EML                                                         SPA37209
      WTH=F1                                                             SPA37210
  102 IF (.NOT. PYPNL) GO TO 103                                         SPA37211
      TEMP=VTH                                                           SPA37212
      VTH=-WTH                                                           SPA37213
      WTH=TEMP                                                           SPA37214
  103 CONTINUE                                                           SPA37215
      RETURN                                                             SPA37216
      END                                                                SPA37217
```

```
      SUBROUTINE VELPP(XX,YY,ZZ)                                         SPA38  1
C                                                                        SPA38  2
C     THIS SUBROUTINE CALCULATES PERTURBATION VELOCITIES INDUCED BY      SPA38  3
C     THE PYLON CONSTANT U-VELOCITY PANELS                               SPA38  4
C     SUPERPOSITION OF 4 CORNER SOLUTIONS IS USED                        SPA38  5
C     VELOCITIES UP,VP,WP ARE RETURNED IN WING REFERENCE FRAME.          SPA38  6
C                                                                        SPA38  7
C     IF II=IP, INFLUENCE COEFFICIENTS RATHER THAN VELOCITIES ARE        SPA38  8
C        CALCULATED                                                      SPA38  9
C                                                                        SPA38 10
      COMMON/BPGEOM/                  SNT2(100),CST2(100),ZLC(200),      SPA38 11
     1   ZRC(200),THTI(100),SANG=,CANG=                                  SPA38 12
      COMMON/ICVEL/UP,VP,WP,II,IF,          DELTP(200)                 SPA38 13
      COMMON/INDEX/NC=,MB=,MS=P,NPANLS,NC=B,NBDCR1,NBDCR2,NBD,NBIP,MP,   SPA38 14
     1   NCP,MSP,N1P,N2,N2P,NPTOT                                        SPA38 15
      COMMON /PYGEOM/DUM(20),XPLE,YPL,CHP,HP,PBIPLE,IP,BLLE,BLTE,PSLPDF, SPA38 16
     1      CENTER                                                       SPA38 17
      COMMON/=PGEOM/XHP(200),XHB(200),XLF(200),XLB(200),YHC(200),        SPA38 18
     1   YLC(200),XCPT(200),YCPT(200),ZCPT(200),S=PPLE(200),S=PPTE(200)  SPA38 19
      COMMON /VELANG/X,Y,Z,U,V,W,EM,TLRNC,TIPY,PYPNL                     SPA38 20
C                                                                        SPA38 21
      LOGICAL         PYPNL,FELTA,FPLTAI,FBLTB,FELTBI,FELT,CENTER        SPA38 22
C                                                                        SPA38 23
      PYPNL=.TRUE.                                                       SPA38 24
      UP=0.0                                                             SPA38 25
      VP=0.0                                                             SPA38 26
      WP=0.0                                                             SPA38 27
      XI=XX                                                              SPA38 28
      ZI=ZZ                                                              SPA38 29
      YDIR=YY-YPL                                                        SPA38 30
      YIMG=-(YY+YPL)                                                     SPA38 31
C                                                                        SPA38 32
C     I IS INDEX OF INFLUENCING PANEL                                    SPA38 33
C                                                                        SPA38 34
      DO 100 I=II,IP                                                     SPA38 35
      FELT=.FALSE.                                                       SPA38 36
      U=0.0                                                              SPA38 37
      V=0.0                                                              SPA38 38
      W=0.0                                                              SPA38 39
      EM1=S=PPLE(I)                                                      SPA38 40
      EM2=S=PPTE(I)                                                      SPA38 41
C                                                                        SPA38 42
C**********.********** CORNER 1 **********************************SPA38 43
C                                                                        SPA38 44
      IF (EM1 .LT. 0.0) GO TO 15                                         SPA38 45
C                                                                        SPA38 46
C     CORNER 1 FOR EM1 .GE. 0 === PANEL LEADING EDGE SWEPT BACK          SPA38 47
C                                                                        SPA38 48
      EM=EM1                                                             SPA38 49
      IF(XI.GE.XRF(I)) GO TO 100                                         SPA38 50
      X=XI+XLP(I)                                                        SPA38 51
      Y=YDIR                                                             SPA38 52
      Z=ZI-ZLC(I)                                                        SPA38 53
      CALL VELO(FELTA)                                                   SPA38 54
      IF(.NOT.FELTA) GO TO 50                                            SPA38 55
      U=U=U                                                              SPA38 56
      V=V=V                                                              SPA38 57
      W=W=W                                                              SPA38 58
      FELT=.TRUE.                                                        SPA38 59
   50 IF(CENTER) GO TO 10                                                SPA38 60
      Y=YIMG                                                             SPA38 61
      CALL VELO(FPLTAI)                                                  SPA38 62
      IF(.NOT.FELTAI) GO TO 10                                           SPA38 63
      U=U=U                                                              SPA38 64
      V=V=V                                                              SPA38 65
      W=W=W                                                              SPA38 66
      FELT=.TRUE.                                                        SPA38 67
      GO TO 10                                                           SPA38 68
C                                                                        SPA38 69
C     CORNER 1 FOR EM1 .LT. 0 === PANEL LEADING EDGE SWEPT FORWARD       SPA38 70
C                                                                        SPA38 71
   15 CONTINUE                                                           SPA38 72
      IF(XI.GE.XLF(I)) GO TO 100                                         SPA38 73
      EM=-EM1                                                            SPA38 74
      X=XI+XLF(I)                                                        SPA38 75
      Y=YDIR                                                             SPA38 76
      Z=ZI+ZLC(I)                                                        SPA38 77
      CALL VELO(FELTA)                                                   SPA38 78
      IF(.NOT.FELTA) GO TO 52                                            SPA38 79
      U=U+U                                                              SPA38 80
      V=V+V                                                              SPA38 81
      W=W+W                                                              SPA38 82
      FELT=.TRUE.                                                        SPA38 83
   52 IF(CENTER) GO TO 17                                                SPA38 84
      Y=YIMG                                                             SPA38 85
      CALL VELO(FPLTAI)                                                  SPA38 86
      IF(.NOT.FELTAI) GO TO 17                                           SPA38 87
      U=U+U                                                              SPA38 88
      V=V+V                                                              SPA38 89
      W=W+W                                                              SPA38 90
      FELT=.TRUE.                                                        SPA38 91
      GO TO 17                                                           SPA38 92
C                                                                        SPA38 93
C**********.********** CORNER 2 **********************************SPA38 94
C                                                                        SPA38 95
C     CORNER 2 FOR EM1 .GE. 0                                            SPA38 96
C                                                                        SPA38 97
   10 CONTINUE                                                           SPA38 98
      X=XI+XRF(I)                                                        SPA38 99
      Y=YDIR                                                             SPA38100
      Z=ZI-ZRC(I)                                                        SPA38101
      CALL VELO(FPLTB)                                                   SPA38102
      IF(.NOT.FELTB) GO TO 12                                            SPA38103
      U=U+U                                                              SPA38104
      V=V+V                                                              SPA38105
      W=W+W                                                              SPA38106
```


(ee) Page 31.

Figure 1.- Continued.

```
      FELTB=.TRUE.
   12 IF(CENTER) GO TO 13
      Y=YIMG
      CALL VELO(FELTBI)
      IF(.NOT.FELTBI)GO TO 13
      TU=TU+U
      TV=TV+V
      TW=TW+W
      FELTB=.TRUE.
   13 IF(.NOT.FELT) GO TO 100
      GO TO 20
C
C     CORNER 2 FOR EM1 .LT. 0
C
   17 CONTINUE
      X=XI+XRB(I)
      Y=YDIR
      Z=Z +ZRC(I)
      CALL VELO(FELTB)
      IF(.NOT.FELTB) GO TO 18
      TU=TU+U
      TV=TV+V
      TW=TW+W
      FELTB=.TRUE.
   18 IF(CENTER) GO TO 16
      Y=YIMG
      CALL VELO(FELTBI)
      IF(.NOT.FELTBI) GO TO 16
      TU=TU+U
      TV=TV+V
      TW=TW+W
      FELTB=.TRUE.
   16 IF(.NOT.FELT) GO TO 100
C
C********************  CORNER 3  ********************************************
C
   20 IF (EM2 .LT. 0.0) GO TO 19
C
C     CORNER 3 FOR EM2 .GE. 0 --- PANEL TRAILING EDGE SWEPT BACK
C
      IF(XI.GE.XRB(I)) GO TO 40
      EM=EM2
      X=XI+XLB(I)
      Y=YDIR
      Z=ZI-ZLC(I)
      CALL VELO(FELTA)
      IF(.NOT.FELTA) GO TO 51
      TU=TU+U
      TV=TV+V
      TW=TW+W
   51 IF(CENTER) GO TO 30
      Y=YIMG
      CALL VELO(FELTAI)
      IF(.NOT.FELTAI) GO TO 30
      TU=TU+U
      TV=TV+V
      TW=TW+W
      GO TO 30
C
C     CORNER 3 FOR EM2 .LT. 0 --- PANEL TRAILING EDGE SWEPT FORWARD
C
   19 EM=-EM2
      IF(XI.GE.XLB(I)) GO TO 40
      X=XI+XLB(I)
      Y=YDIR
      Z=ZI+ZLC(I)
      CALL VELO(FELTA)
      IF(.NOT.FELTA) GO TO 53
      TU=TU+U
      TV=TV+V
      TW=TW+W
   53 IF(CENTER) GO TO 21
      Y=YIMG
      CALL VELO(FELTAI)
      IF(.NOT.FELTAI) GO TO 21
      TU=TU+U
      TV=TV+V
      TW=TW+W
      GO TO 21
C
C********************  CORNER 4  ********************************************
C
C     CORNER 4 FOR EM2 .GE. 0
C
   30 CONTINUE
      X=XI+XRB(I)
      Y=YDIR
      Z=ZI-ZRC(I)
      CALL VELO(FELTB)
      IF(.NOT.FELTB) GO TO 35
      TU=TU+U
      TV=TV+V
      TW=TW+W
   35 IF(CENTER) GO TO 40
      Y=YIMG
      CALL VELO(FELTBI)
      IF(.NOT.FELTBI) GO TO 40
      TU=TU+U
      TV=TV+V
      TW=TW+W
      GO TO 40
C
C     CORNER 4 FOR EM2 .LT. 0
C
   21 CONTINUE
      X=XI+XRB(I)
      Y=YDIR
      Z=ZI+ZRC(I)
      CALL VELO(FELTB)
      IF(.NOT.FELTB) GO TO 25
      TU=TU+U
      TV=TV+V
      TW=TW+W
   25 IF(CENTER) GO TO 40
      Y=YIMG
      CALL VELO(FELTBI)
      IF(.NOT.FELTBI) GO TO 40
      TU=TU+U
      TV=TV+V
      TW=TW+W
C
C     SIGN CHANGES NECESSARY TO ACCOMODATE USE OF SUBROUTINE
C     VELO WITH DIFFERENT COORDINATE SYSTEM
C
   40 CONTINUE
      TU=-TU
      TW=-TW
      IF(II.EQ.IP) GO TO 101
C
C     CALCULATION OF PERTURBATION VELOCITIES
C
      UP=UP+TU*DELTP(I)
      VP=VP+TV*DELTP(I)
      WP=WP+TW*DELTP(I)
  100 CONTINUE
      RETURN
C
C     RETURN INFLUENCE COEFFICIENTS IF II=IP
C
  101 UP=TU
      VP=TV
      WP=TW
      RETURN
      END
```

(ff) Page 32.

Figure 1.- Continued.

```
      SUBROUTINE VELPTM(XX,YY,ZZ)                                       SPA39  1
C                                                                       SPA39  2
C     THIS SUBROUTINE CALCULATES PERTURBATION VELOCITIES INDUCED BY     SPA39  3
C     THE PYLON THICKNESS PANELS.                                       SPA39  4
C     SUPERPOSITION OF 4 CORNER SOLUTIONS IS USED                       SPA39  5
C     VELPTM RETURNS VELOCITIES UP,VP,WP IN WING REFERENCE FRAME.       SPA39  6
C                                                                       SPA39  7
      COMMON /ICVEL/UP,VP,WP,II,IF,        DELTM(200)                   SPA39  8
      COMMON /PYGEOM/DUM(20),XPLE,YPL,CNP,WP,PSIPLE,IP,SLLE,SLTE,PSLPDF,SPA39  9
     1       CENTER                                                     SPA39 10
      COMMON /TMKDAT/NC=3,NCPS,MS,MPS  ,NTMP,XRFT(400),ZRBT(400),       SPA39 11
     1    XLFT(400),XLBT(400),YRCT(400),YLCT(400),ZRCT(400),ZLCT(400),  SPA39 12
     2    THETAL(400),THETPL(200),SLLFT(400),SLTET(400),DZDX(400)       SPA39 13
      COMMON /THWANG/XTM,YTM,ZTM,UTM,VTM,WTM,EML,MMT                    SPA39 14
      COMMON /VELARG/DUMY(9),PYPNL                                      SPA39 15
C                                                                       SPA39 16
      EQUIVALENCE (X,XTM),(Y,YTM),(Z,ZTM),(U,UTM),(V,VTM),(W,WTM)       SPA39 17
C                                                                       SPA39 18
      LOGICAL FELTA,FELTAI,FELTB,FELTBI,FELT,        PYPNL,CENTER       SPA39 19
C                                                                       SPA39 20
      PYPNL=.TRUE.                                                      SPA39 21
      UP=0.0                                                            SPA39 22
      VP=0.0                                                            SPA39 23
      WP=0.0                                                            SPA39 24
      XI=XX                                                             SPA39 25
      ZI=ZZ                                                             SPA39 26
      YDIR=YY-YPL                                                       SPA39 27
      YIMG=-(YY+YPL)                                                    SPA39 28
C                                                                       SPA39 29
C     I IS INDEX OF INFLUENCING PANEL                                   SPA39 30
C                                                                       SPA39 31
      DO 100 I=II,IF                                                    SPA39 32
      FELT=.FALSE.                                                      SPA39 33
      U=0.0                                                             SPA39 34
      V=0.0                                                             SPA39 35
      W=0.0                                                             SPA39 36
      EM1=SLLFT(I)                                                      SPA39 37
      EM2=SLTET(I)                                                      SPA39 38
C                                                                       SPA39 39
C**********************  CORNER 1  ***********************************SPA39 40
C                                                                       SPA39 41
      IF (EM1 .LT. 0.0) GO TO 15                                        SPA39 42
C                                                                       SPA39 43
C     CORNER 1 FOR EM1.GE. 0 === PANEL LEADING EDGE SWEPT BACK          SPA39 44
C                                                                       SPA39 45
      IF(XI.GE.XRFT(I))GO TO 100                                        SPA39 46
      EML=EM1                                                           SPA39 47
      X=XI-XLFT(I)                                                      SPA39 48
      Y=YDIR                                                            SPA39 49
      Z=ZI-ZLFT(I)                                                      SPA39 50
      CALL VELOTM(FELTA)                                                SPA39 51
      IF(.NOT.FELTA) GO TO 11                                           SPA39 52
      U=U+U                                                             SPA39 53
      V=V+V                                                             SPA39 54
      W=W+W                                                             SPA39 55
      FELT=.TRUE.                                                       SPA39 56
   11 IF(CENTER) GO TO 10                                               SPA39 57
      Y=YIMG                                                            SPA39 58
      CALL VELOTM(FELTAI)                                               SPA39 59
      IF(.NOT.FELTAI) GO TO 10                                          SPA39 60
      U=U+U                                                             SPA39 61
      V=V+V                                                             SPA39 62
      W=W+W                                                             SPA39 63
      FELT=.TRUE.                                                       SPA39 64
      GO TO 10                                                          SPA39 65
C                                                                       SPA39 66
C     CORNER 1 FOR EM1.LT. 0 === PANEL LEADING EDGE SWEPT FORWARD       SPA39 67
C                                                                       SPA39 68
   15 CONTINUE                                                          SPA39 69
      IF(XI.GE.XLFT(I)) GO TO 100                                       SPA39 70
      EML=EM1                                                           SPA39 71
      X=XI-XLFT(I)                                                      SPA39 72
      Y=YDIR                                                            SPA39 73
      Z=-ZI+ZLCT(I)                                                     SPA39 74
      CALL VELOTM(FELTA)                                                SPA39 75
      IF(.NOT.FELTA) GO TO 16                                           SPA39 76
      U=U+U                                                             SPA39 77
      V=V+V                                                             SPA39 78
      W=W+W                                                             SPA39 79
      FELT=.TRUE.                                                       SPA39 80
   16 IF(CENTER) GO TO 17                                               SPA39 81
      Y=YIMG                                                            SPA39 82
      CALL VELOTM(FELTAI)                                               SPA39 83
      IF(.NOT.FELTAI) GO TO 17                                          SPA39 84
      U=U+U                                                             SPA39 85
      V=V+V                                                             SPA39 86
      W=W+W                                                             SPA39 87
      FELT=.TRUE.                                                       SPA39 88
      GO TO 17                                                          SPA39 89
C                                                                       SPA39 90
C**********************  CORNER 2  ***********************************SPA39 91
C                                                                       SPA39 92
C     CORNER 2 FOR EM1 .GE. 0                                           SPA39 93
C                                                                       SPA39 94
   10 CONTINUE                                                          SPA39 95
      X=XI-XRFT(I)                                                      SPA39 96
      Y=YDIR                                                            SPA39 97
      Z=ZI-ZRCT(I)                                                      SPA39 98
      CALL VELOTM(FELTB)                                                SPA39 99
      IF(.NOT.FELTB) GO TO 12                                           SPA39100
      U=U+U                                                             SPA39101
      V=V+V                                                             SPA39102
      W=W+W                                                             SPA39103
      FELT=.TRUE.                                                       SPA39104
   12 IF(CENTER) GO TO 13                                               SPA39105
      Y=YIMG                                                            SPA39106
      CALL VELOTM(FELTBI)                                               SPA39107
      IF(.NOT.FELTBI) GO TO 13                                          SPA39108
      U=U+U                                                             SPA39109
      V=V+V                                                             SPA39110
      W=W+W                                                             SPA39111
      FELT=.TRUE.                                                       SPA39112
   13 IF(.NOT.FELT) GO TO 100                                           SPA39113
      GO TO 20                                                          SPA39114
C                                                                       SPA39115
C     CORNER 2 FOR EM1 .LT. 0                                           SPA39116
C                                                                       SPA39117
   17 CONTINUE                                                          SPA39118
      X=XI-XRFT(I)                                                      SPA39119
      Y=YDIR                                                            SPA39120
      Z=ZI-ZRCT(I)                                                      SPA39121
      CALL VELOTM(FELTB)                                                SPA39122
      IF(.NOT.FELTB) GO TO 24                                           SPA39123
      U=U+U                                                             SPA39124
      V=V+V                                                             SPA39125
      W=W+W                                                             SPA39126
      FELT=.TRUE.                                                       SPA39127
   24 IF(CENTER) GO TO 25                                               SPA39128
      Y=YIMG                                                            SPA39129
      CALL VELOTM(FELTBI)                                               SPA39130
      IF(.NOT.FELTBI) GO TO 25                                          SPA39131
      U=U+U                                                             SPA39132
      V=V+V                                                             SPA39133
      W=W+W                                                             SPA39134
      FELT=.TRUE.                                                       SPA39135
   25 IF(.NOT.FELT) GO TO 100                                           SPA39136
C                                                                       SPA39137
C**********************  CORNER 3  ***********************************SPA39138
C                                                                       SPA39139
   20 IF (EM2 .LT. 0.0) GO TO 19                                        SPA39140
C                                                                       SPA39141
C     CORNER 3 FOR EM2 .GE.0 === PANEL TRAILING EDGE SWEPT BACK         SPA39142
C                                                                       SPA39143
      IF(XI.GE.XRHT(I)) GO TO 40                                        SPA39144
```

(gg) Page 33.

Figure 1.- Continued.

```
      EML=EM2                                                           SPA39145
      X=XI+ALBT(I)                                                      SPA39146
      Y=YDIR                                                            SPA39147
      Z=ZI-ZLCT(I)                                                      SPA39148
      CALL VELUTH(FELTA)                                                SPA39149
      IF(.NOT.FELTA) GO TO 23                                           SPA39150
      TU=TU+U                                                           SPA39151
      TV=TV+V                                                           SPA39152
      TW=TW+W                                                           SPA39153
   23 IF(CENTER) GO TO 30                                               SPA39154
      Y=YIMG                                                            SPA39155
      CALL VELUTH(FELTAI)                                               SPA39156
      IF(.NOT.FELTAI) GO TO 30                                          SPA39157
      TU=TU+U                                                           SPA39158
      TV=TV+V                                                           SPA39159
      TW=TW+W                                                           SPA39160
      GO TO 30                                                          SPA39161
C                                                                       SPA39162
C     CORNER 3 FOR EM2 .LT.0 *** PANEL TRAILING EDGE SWEPT FORWARD      SPA39163
C                                                                       SPA39164
   19 EML=EM2                                                           SPA39165
      IF(XI.GE.XLBT(I)) GO TO 40                                        SPA39166
      X=XI+XLBT(I)                                                      SPA39167
      Y=YDIR                                                            SPA39168
      Z=ZI+ZLCT(I)                                                      SPA39169
      CALL VELUTH(FELTA)                                                SPA39170
      IF(.NOT.FELTA) GO TO 26                                           SPA39171
      TU=TU+U                                                           SPA39172
      TV=TV+V                                                           SPA39173
      TW=TW+W                                                           SPA39174
   26 IF(CENTER) GO TO 21                                               SPA39175
      Y=YIMG                                                            SPA39176
      CALL VELUTH(FELTAI)                                               SPA39177
      IF(.NOT.FELTAI) GO TO 21                                          SPA39178
      TU=TU+U                                                           SPA39179
      TV=TV+V                                                           SPA39180
      TW=TW+W                                                           SPA39181
      GO TO 21                                                          SPA39182
C                                                                       SPA39183
C*********************  CORNER 4  ***************************************SPA39184
C                                                                       SPA39185
C     CORNER 4 FOR EM2 .GE. 0                                           SPA39186
C                                                                       SPA39187
   30 CONTINUE                                                          SPA39188
      X=XI+XRBT(I)                                                      SPA39189
      Y=YDIR                                                            SPA39190
      Z=ZI+ZRCT(I)                                                      SPA39191
      CALL VELOTH(FELTB)                                                SPA39192
      IF(.NOT.FELTB) GO TO 35                                           SPA39193
      TU=TU+U                                                           SPA39194
      TV=TV+V                                                           SPA39195
      TW=TW+W                                                           SPA39196
   35 IF(CENTER) GO TO 40                                               SPA39197
      Y=YIMG                                                            SPA39198
      CALL VELUTH(FELTBI)                                               SPA39199
      IF(.NOT.FELTBI) GO TO 40                                          SPA39200
      TU=TU+U                                                           SPA39201
      TV=TV+V                                                           SPA39202
      TW=TW+W                                                           SPA39203
      GO TO 40                                                          SPA39204
C                                                                       SPA39205
C     CORNER 4 FOR EM2 .LT. 0                                           SPA39206
C                                                                       SPA39207
   21 CONTINUE                                                          SPA39208
      X=XI+XRBT(I)                                                      SPA39209
      Y=YDIR                                                            SPA39210
      Z=ZI+ZRCT(I)                                                      SPA39211
      CALL VELOTH(FELTB)                                                SPA39212
      IF(.NOT.FELTB) GO TO 28                                           SPA39213
      TU=TU+U                                                           SPA39214
      TV=TV+V                                                           SPA39215
      TW=TW+W                                                           SPA39216
   28 IF(CENTER) GO TO 40                                               SPA39217
      Y=YIMG                                                            SPA39218
      CALL VELUTH(FELTBI)                                               SPA39219
      IF(.NOT.FELTBI) GO TO 40                                          SPA39220
      TU=TU+U                                                           SPA39221
      TV=TV+V                                                           SPA39222
      TW=TW+W                                                           SPA39223
C                                                                       SPA39224
C     SIGN CHANGES NECESSARY TO ACCOMMODATE USE OF SUBROUTINE           SPA39225
C     VELUTH WITH DIFFERENT COORDINATE SYSTEM                           SPA39226
C                                                                       SPA39227
   40 CONTINUE                                                          SPA39228
      TU=-TU                                                            SPA39229
      TW=-TW                                                            SPA39230
C                                                                       SPA39231
C     CALCULATION OF PERTURBATION VELOCITIES                            SPA39232
C                                                                       SPA39233
      UP=UP+TU*DZDX(I)                                                  SPA39234
      VP=VP+TV*DZDX(I)                                                  SPA39235
      WP=WP+TW*DZDX(I)                                                  SPA39236
  100 CONTINUE                                                          SPA39237
      RETURN                                                            SPA39238
      END                                                               SPA39239
```

```
      SUBROUTINE VELWP(XX,YY,ZZ)                                        SPA40  1
C                                                                       SPA40  2
C     THIS SUBROUTINE CALCULATES PERTURBATION VELOCITIES INDUCED BY     SPA40  3
C     THE WING CONSTANT U-VELOCITY PANELS USING A SUPERPOSITION SCHEME  SPA40  4
C     SUPERPOSITION OF 4 CORNER SOLUTIONS IS USED                       SPA40  5
C     VELWP  RETURNS VELOCITIES UP,VP,WP IN WING REFERENCE FRAME        SPA40  6
C                                                                       SPA40  7
C     IF II=IF, INFLUENCE COEFFICIENTS RATHER THAN VELOCITIES ARE       SPA40  8
C     CALCULATED                                                        SPA40  9
C                                                                       SPA40 10
      COMMON/ICVEL/UP,VP,WP,II,IF,          DELTP(200)                  SPA40 11
      COMMON /WGEOM/XRW,UC,ZRW,CHW,SLPWLE,SLPWTE,PSIWLE(20),PSIWTE(20), SPA40 12
     1             DUM(20)                                              SPA40 13
      COMMON/WPGEOM/XRF(200),XRB(200),XLF(200),XLB(200),YRC(200),       SPA40 14
     1   YLC(200),XCPT(200),YCPT(200),ZCPT(400),SWPPLE(200),SWPPTE(200) SPA40 15
      COMMON /VELARG/X,Y,Z,U,V,W,EM,TLRNC,TIPY,PYPNL                    SPA40 16
C                                                                       SPA40 17
      LOGICAL       PYPNL,FELTA,FELTAI,FELTB,FELTBI,FELT                SPA40 18
C                                                                       SPA40 19
      PYPNL=.FALSE.                                                     SPA40 20
      UP=0.0                                                            SPA40 21
      VP=0.0                                                            SPA40 22
      WP=0.0                                                            SPA40 23
      XI=XX                                                             SPA40 24
      YI=YY                                                             SPA40 25
      ZI=ZZ                                                             SPA40 26
C                                                                       SPA40 27
C     I IS INDEX OF INFLUENCING PANEL                                   SPA40 28
C                                                                       SPA40 29
      DO 100 I=II,IF                                                    SPA40 30
      FELT=.FALSE.                                                      SPA40 31
      TU=0.0                                                            SPA40 32
      TV=0.0                                                            SPA40 33
      TW=0.0                                                            SPA40 34
      EM1=SWPPLE(I)                                                     SPA40 35
      EM2=SWPPTE(I)                                                     SPA40 36
C                                                                       SPA40 37
C*********************  CORNER 1  ***************************************SPA40 38
C                                                                       SPA40 39
      IF (EM1 .LT. 0.0) GO TO 15                                        SPA40 40
C                                                                       SPA40 41
C     CORNER 1 FOR EM1 .GE. 0 ***PANEL LEADING EDGE SWEPT BACK          SPA40 42
C                                                                       SPA40 43
      EML=EM1                                                           SPA40 44
```

(hh) Page 34.

Figure 1.- Continued.

```
      IF(XI,GE,XRB(I)) GO TO 100                                        SPA40 45
      XB=XI+XLB(I)                                                      SPA40 46
      YB=YI+YLC(I)                                                      SPA40 47
      CALL VELU(FELTA)                                                  SPA40 48
      IF(.NOT.FELTA) GO TO 50                                           SPA40 49
      TU=TU+U                                                           SPA40 50
      TV=TV+V                                                           SPA40 51
      TW=TW+W                                                           SPA40 52
      FELT=.TRUE.                                                       SPA40 53
   50 YBYI+YLC(I)                                                       SPA40 54
      CALL VPLU(FELTAI)                                                 SPA40 55
      IF(.NOT.FELTAI) GO TO 10                                          SPA40 56
      TU=TU+U                                                           SPA40 57
      TV=TV+V                                                           SPA40 58
      TW=TW+W                                                           SPA40 59
      FELT=.TRUE.                                                       SPA40 60
      GO TO 10                                                          SPA40 61
C                                                                       SPA40 62
C     CORNER 1 FOR EM1 .LT. 0 ***PANEL LEADING EDGE SWEPT FORWARD       SPA40 63
C                                                                       SPA40 64
   15 CONTINUE                                                          SPA40 65
      IF(XI,GE,XLB(I)) GO TO 100                                        SPA40 66
      EM=EM1                                                            SPA40 67
      XB=XI+XLB(I)                                                      SPA40 68
      YB=YI+YLC(I)                                                      SPA40 69
      CALL VELU(FELTA)                                                  SPA40 70
      IF(.NOT.FELTA) GO TO 52                                           SPA40 71
      TU=TU+U                                                           SPA40 72
      TV=TV+V                                                           SPA40 73
      TW=TW+W                                                           SPA40 74
      FELT=.TRUE.                                                       SPA40 75
   52 YB=(YI+YLC(I))                                                    SPA40 76
      CALL VPLU(FELTAI)                                                 SPA40 77
      IF(.NOT.FELTAI) GO TO 17                                          SPA40 78
      TU=TU+U                                                           SPA40 79
      TV=TV+V                                                           SPA40 80
      TW=TW+W                                                           SPA40 81
      FELT=.TRUE.                                                       SPA40 82
      GO TO 17                                                          SPA40 83
C                                                                       SPA40 84
C********************  CORNER 2  ***************************************SPA40 85
C                                                                       SPA40 86
C     CORNER 2 FOR EM1 .GE. 0                                           SPA40 87
C                                                                       SPA40 88
   10 CONTINUE                                                          SPA40 89
      XB=XI+XRB(I)                                                      SPA40 90
      YB=YI+YRC(I)                                                      SPA40 91
      CALL VELU(FELTB)                                                  SPA40 92
      IF(.NOT.FELTB) GO TO 12                                           SPA40 93
      TU=TU+U                                                           SPA40 94
      TV=TV+V                                                           SPA40 95
      TW=TW+W                                                           SPA40 96
      FELT=.TRUE.                                                       SPA40 97
   12 YBYI+YRC(I)                                                       SPA40 98
      CALL VPLO(FELTBI)                                                 SPA40 99
      IF(.NOT.FELTBI)GO TO 13                                           SPA40100
      TU=TU+U                                                           SPA40101
      TV=TV+V                                                           SPA40102
      TW=TW+W                                                           SPA40103
      FELT=.TRUE.                                                       SPA40104
   13 IF(.NOT.FELT) GO TO 100                                           SPA40105
      GO TO 20                                                          SPA40106
C                                                                       SPA40107
C     CORNER 2 FOR EM1 .LT. 0                                           SPA40108
C                                                                       SPA40109
   17 CONTINUE                                                          SPA40110
      XB=XI+XRB(I)                                                      SPA40111
      YB=YI+YRC(I)                                                      SPA40112
      CALL VELO(FELTB)                                                  SPA40113
      IF(.NOT.FELTB) GO TO 18                                           SPA40114
      TU=TU+U                                                           SPA40115
      TV=TV+V                                                           SPA40116
      TW=TW+W                                                           SPA40117
      FELT=.TRUE.                                                       SPA40118
   18 YB=(YI+YRC(I))                                                    SPA40119
      CALL VPLO(FELTBI)                                                 SPA40120
      IF(.NOT.FELTBI) GO TO 16                                          SPA40121
      TU=TU+U                                                           SPA40122
      TV=TV+V                                                           SPA40123
      TW=TW+W                                                           SPA40124
      FELT=.TRUE.                                                       SPA40125
   16 IF(.NOT.FELT) GO TO 100                                           SPA40126
C                                                                       SPA40127
C********************  CORNER 3  ***************************************SPA40128
C                                                                       SPA40129
   20 IF (EM2 .LT. 0.0) GO TO 19                                        SPA40130
C                                                                       SPA40131
C     CORNER 3 FOR EM2 .GE. 0 *** PANEL TRAILING EDGE SWEPT BACK        SPA40132
C                                                                       SPA40133
      IF(XI,GE,XRB(I)) GO TO 40                                         SPA40134
      EM=EM2                                                            SPA40135
      XB=XI+XLB(I)                                                      SPA40136
      YB=YI+YLC(I)                                                      SPA40137
      CALL VELU(FELTA)                                                  SPA40138
      IF(.NOT.FELTA) GO TO 51                                           SPA40139
      TU=TU+U                                                           SPA40140
      TV=TV+V                                                           SPA40141
      TW=TW+W                                                           SPA40142
   51 YBYI+YLC(I)                                                       SPA40143
      CALL VPLU(FELTAI)                                                 SPA40144
      IF(.NOT.FELTAI) GO TO 30                                          SPA40145
      TU=TU+U                                                           SPA40146
      TV=TV+V                                                           SPA40147
      TW=TW+W                                                           SPA40148
      GO TO 30                                                          SPA40149
C                                                                       SPA40150
C     CORNER 3 FOR EM2 .LT. 0 *** PANEL TRAILING EDGE SWEPT FORWARD     SPA40151
C                                                                       SPA40152
   19 EM=EM2                                                            SPA40153
      IF(XI,GE,XLB(I)) GO TO 40                                         SPA40154
      XB=XI+XLB(I)                                                      SPA40155
      YB=YI+YLC(I)                                                      SPA40156
      CALL VPLO(FELTA)                                                  SPA40157
      IF(.NOT.FELTA) GO TO 53                                           SPA40158
      TU=TU+U                                                           SPA40159
      TV=TV+V                                                           SPA40160
      TW=TW+W                                                           SPA40161
   53 YB=(YI+YLC(I))                                                    SPA40162
      CALL VPLU(FELTAI)                                                 SPA40163
      IF(.NOT.FELTAI) GO TO 21                                          SPA40164
      TU=TU+U                                                           SPA40165
      TV=TV+V                                                           SPA40166
      TW=TW+W                                                           SPA40167
      GO TO 21                                                          SPA40168
C                                                                       SPA40169
C********************  CORNER 4  ***************************************SPA40170
C                                                                       SPA40171
C     CORNER 4 FOR EM2 .GE. 0                                           SPA40172
C                                                                       SPA40173
   30 CONTINUE                                                          SPA40174
      XB=XI+XRB(I)                                                      SPA40175
      YB=YI+YRC(I)                                                      SPA40176
      CALL VELU(FELTB)                                                  SPA40177
      IF(.NOT.FELTB) GO TO 35                                           SPA40178
      TU=TU+U                                                           SPA40179
      TV=TV+V                                                           SPA40180
      TW=TW+W                                                           SPA40181
   35 YBYI+YRC(I)                                                       SPA40182
      CALL VPLO(FELTBI)                                                 SPA40183
      IF(.NOT.FELTBI) GO TO 40                                          SPA40184
      TU=TU+U                                                           SPA40185
      TV=TV+V                                                           SPA40186
      TW=TW+W                                                           SPA40187
      GO TO 40                                                          SPA40188
```


(ii) Page 35.

Figure 1.- Continued.

```
C
C     CORNER 4 FOR EM2 ,LT, 0
C
   21 CONTINUE
      X=XI+XRB(I)
      Y=YI+YRC(I)
      CALL VELU(FELTB)
      IF(.NOT.FELTB) GO TO 25
      TU=TU+U
      TV=TV+V
      TW=TW+W
   25 Y=-(YI+YRC(I))
      CALL VELU(FELTBI)
      IF(.NOT.FELTBI) GO TO 40
      TU=TU+U
      TV=TV-V
      TW=TW+W
C
C     SIGN CHANGES NECESSARY TO ACCOUNDATE USE OF SUBROUTINE
C     VELU WITH DIFFERENT COORDINATE SYSTEM
C
   40 CONTINUE
      TU=TU
      TW=TW
      IF(II.EQ.IF) GO TO 101
C
C     CALCULATION OF PERTURBATION VELOCITIES
C
      UP=UP+TU*DELTP(I)
      VP=VP+TV*DELTP(I)
      WP=WP+TW*DELTP(I)
  100 CONTINUE
      RETURN
C
C     RETURN INFLUENCE FUNCTIONS IF II=IF
C
  101 UP=TU
      VP=TV
      WP=TW
      RETURN
      END
```

```
      SUBROUTINE VELWTH(XX,YY,ZZ)
C
C     THIS SUBROUTINE CALCULATES PERTURBATION VELOCITIES INDUCED BY
C     THE WING THICKNESS PANELS,
C     SUPERPOSITION OF 4 CORNER SOLUTIONS IS USED
C     VELWTH RETURNS VELOCITIES UP,VP,WP IN WING REFERENCE FRAME
C
      COMMON /ICVEL/UP,VP,WP,II,IF,        DELTP(200)
      COMMON /THKDAT/NCWB,NCPB,MS,MPS  ,NTHP,XRFT(400),XRBT(400),
     1   XLFT(400),XLBT(400),YRCT(400),YLCT(400),ZRCT(400),ZLCT(400),
     2   THETAL(400),THETPL(200),BLLFT(400),BLTET(400),DZDX(400)
      COMMON /THVARG/XTH,YTH,ZTH,UTH,VTH,WTH,EML,PRT
      COMMON /VELARG/DUMY(9),PYPNL
C
      EQUIVALENCE (X,XTH),(Y,YTH),(Z,ZTH),(U,UTH),(V,VTH),(W,WTH)
C
      LOGICAL      FELTA,FELTAI,FELTB,FELTBI,FELT,        PYPNL
C
      PYPNL=.FALSE.
      UP=0.0
      VP=0.0
      WP=0.0
      XI=XX
      YI=YY
      ZI=ZZ
C
```

```
C     I IS INDEX OF INFLUENCING PANEL
C
      DO 100 I=II,IF
      FELT=.FALSE.
      TU=0.0
      TV=0.0
      TW=0.0
      EM1=BLLBT(I)
      EM2=BLTBT(I)
C
C*********************  CORNER 1  ********************************
C
      IF (EM1 .LT. 0.0) GO TO 15
C
C     CORNER 1 FOR EM1 .GE. 0 ---PANEL LEADING EDGE SWEPT BACK
C
      IF(XI.GE.XRFT(I))GO TO 100
      EML=EM1
      X=XI+XLFT(I)
      Y=YI+YLCT(I)
      CALL VELOTH(FELTA)
      IF(.NOT.FELTA) GO TO 11
      TU=TU+U
      TV=TV+V
      TW=TW+W
      FELT=.TRUE.
   11 Y=YI+YLCT(I)
      CALL VELOTH(FELTAI)
      IF(.NOT.FELTAI) GO TO 10
      TU=TU+U
      TV=TV-V
      TW=TW+W
      FELT=.TRUE.
      GO TO 10
C
C     CORNER 1 FOR EM1 .LT. 0 ---PANEL LEADING EDGE SWEPT FORWARD
C
   15 CONTINUE
      IF(XI.GE.XLFT(I)) GO TO 100
      EML=EM1
      X=XI+XLFT(I)
      Y=YI+YLCT(I)
      CALL VELOTH(FELTA)
      IF(.NOT.FELTA) GO TO 16
      TU=TU+U
      TV=TV+V
      TW=TW+W
      FELT=.TRUE.
   16 Y=-(YI+YLCT(I))
      CALL VELOTH(FELTAI)
      IF(.NOT.FELTAI) GO TO 17
      TU=TU+U
      TV=TV-V
      TW=TW+W
      FELT=.TRUE.
      GO TO 19
C
C*********************  CORNER 2  ********************************
C
C     CORNER 2 FOR EM1 .GE. 0
C
   10 CONTINUE
      X=XI+XRFT(I)
      Y=YI+YRCT(I)
      CALL VELOTH(FELTB)
      IF(.NOT.FELTB) GO TO 12
      TU=TU+U
      TV=TV-V
      TW=TW+W
      FELT=.TRUE.
   12 Y=YI+YRCT(I)
      CALL VELOTH(FELTBI)
```

SPA40189 SPA40190 SPA40191 SPA40192 SPA40193 SPA40194 SPA40195 SPA40196 SPA40197 SPA40198 SPA40199 SPA40200 SPA40201 SPA40202 SPA40203 SPA40204 SPA40205 SPA40206 SPA40207 SPA40208 SPA40209 SPA40210 SPA40211 SPA40212 SPA40213 SPA40214 SPA40215 SPA40216 SPA40217 SPA40218 SPA40219 SPA40220 SPA40221 SPA40222 SPA40223 SPA40224 SPA40225 SPA40226 SPA40227 SPA40228 SPA40229

SPA41 1 SPA41 2 SPA41 3 SPA41 4 SPA41 5 SPA41 6 SPA41 7 SPA41 8 SPA41 9 SPA41 10 SPA41 11 SPA41 12 SPA41 13 SPA41 14 SPA41 15 SPA41 16 SPA41 17 SPA41 18 SPA41 19 SPA41 20 SPA41 21 SPA41 22 SPA41 23 SPA41 24 SPA41 25 SPA41 26

SPA41 27 SPA41 28 SPA41 29 SPA41 30 SPA41 31 SPA41 32 SPA41 33 SPA41 34 SPA41 35 SPA41 36 SPA41 37 SPA41 38 SPA41 39 SPA41 40 SPA41 41 SPA41 42 SPA41 43 SPA41 44 SPA41 45 SPA41 46 SPA41 47 SPA41 48 SPA41 49 SPA41 50 SPA41 51 SPA41 52 SPA41 53 SPA41 54 SPA41 55 SPA41 56 SPA41 57 SPA41 58 SPA41 59 SPA41 60 SPA41 61 SPA41 62 SPA41 63 SPA41 64 SPA41 65 SPA41 66 SPA41 67 SPA41 68 SPA41 69 SPA41 70 SPA41 71 SPA41 72 SPA41 73 SPA41 74 SPA41 75 SPA41 76 SPA41 77 SPA41 78 SPA41 79 SPA41 80 SPA41 81 SPA41 82 SPA41 83 SPA41 84 SPA41 85 SPA41 86 SPA41 87 SPA41 88 SPA41 89 SPA41 90 SPA41 91 SPA41 92 SPA41 93 SPA41 94 SPA41 95 SPA41 96 SPA41 97 SPA41 98

(jj) Page 36.

Figure 1.- Continued.

```
      IF(.NOT.FELTBI) GO TO 13                                          SPAW1 99
      TU=TU+U                                                           SPAW1100
      TV=TV+V                                                           SPAW1101
      TW=TW+W                                                           SPAW1102
      FELTB=.TRUE.                                                      SPAW1103
   13 IF(.NOT.FELT) GO TO 100                                           SPAW1104
      GO TO 20                                                          SPAW1105
C                                                                       SPAW1106
C     CORNER 2 FOR EM1 .LT. 0                                           SPAW1107
C                                                                       SPAW1108
   17 CONTINUE                                                          SPAW1109
      XB=XI+XRFT(I)                                                     SPAW1110
      YB=YI+YRCT(I)                                                     SPAW1111
      CALL VELOTH(FELTB)                                                SPAW1112
      IF(.NOT.FELTB) GO TO 24                                           SPAW1113
      TU=TU+U                                                           SPAW1114
      TV=TV+V                                                           SPAW1115
      TW=TW+W                                                           SPAW1116
      FELTB=.TRUE.                                                      SPAW1117
   24 YB=(YI+YRCT(I))                                                   SPAW1118
      CALL VELOTH(FELTBI)                                               SPAW1119
      IF(.NOT.FELTBI) GO TO 25                                          SPAW1120
      TU=TU+U                                                           SPAW1121
      TV=TV+V                                                           SPAW1122
      TW=TW+W                                                           SPAW1123
      FELTB=.TRUE.                                                      SPAW1124
   25 IF(.NOT.FELT) GO TO 100                                           SPAW1125
C                                                                       SPAW1126
C*********************  CORNER 3  ***********************************SPAW1127
C                                                                       SPAW1128
   20 IF (EM2 .LT. 0.0) GO TO 19                                        SPAW1129
C                                                                       SPAW1130
C     CORNER 3 FOR EM2 .GE. 0 ---PANEL TRAILING EDGE SWEPT BACK         SPAW1131
C                                                                       SPAW1132
      IF(XI.GE.XRBT(I)) GO TO 40                                        SPAW1133
      EMLE=EM2                                                          SPAW1134
      XB=XI+XLBT(I)                                                     SPAW1135
      YB=YI+YLCT(I)                                                     SPAW1136
      CALL VELOTH(FELTA)                                                SPAW1137
      IF(.NOT.FELTA) GO TO 23                                           SPAW1138
      TU=TU+U                                                           SPAW1139
      TV=TV+V                                                           SPAW1140
      TW=TW+W                                                           SPAW1141
   23 YB=YI+YLCT(I)                                                     SPAW1142
      CALL VELOTH(FELTAI)                                               SPAW1143
      IF(.NOT.FELTAI) GO TO 30                                          SPAW1144
      TU=TU+U                                                           SPAW1145
      TV=TV+V                                                           SPAW1146
      TW=TW+W                                                           SPAW1147
      GO TO 30                                                          SPAW1148
C                                                                       SPAW1149
C     CORNER 3 FOR EM2 .LT. 0 ---PANEL TRAILING EDGE SWEPT FORWARD      SPAW1150
C                                                                       SPAW1151
   19 EMLE=EM2                                                          SPAW1152
      IF(XI.GE.XLHT(I)) GO TO 40                                        SPAW1153
      XB=XI+XLBT(I)                                                     SPAW1154
      YB=YI+YLCT(I)                                                     SPAW1155
      CALL VELOTH(FELTA)                                                SPAW1156
      IF(.NOT.FELTA) GO TO 26                                           SPAW1157
      TU=TU+U                                                           SPAW1158
      TV=TV+V                                                           SPAW1159
      TW=TW+W                                                           SPAW1160
   26 YB=(YI+YLCT(I))                                                   SPAW1161
      CALL VELOTH(FELTAI)                                               SPAW1162
      IF(.NOT.FELTAI) GO TO 21                                          SPAW1163
      TU=TU+U                                                           SPAW1164
      TV=TV+V                                                           SPAW1165
      TW=TW+W                                                           SPAW1166
      GO TO 21                                                          SPAW1167
C                                                                       SPAW1168
C*********************  CORNER 4  ***********************************SPAW1169
C                                                                       SPAW1170
C     CORNER 4 FOR EM2 .GE. 0                                           SPAW1171
C                                                                       SPAW1172
   30 CONTINUE                                                          SPAW1173
      XB=XI+XRHT(I)                                                     SPAW1174
      YB=YI+YRCT(I)                                                     SPAW1175
      CALL VELOTH(FELTB)                                                SPAW1176
      IF(.NOT.FELTB) GO TO 35                                           SPAW1177
      TU=TU+U                                                           SPAW1178
      TV=TV+V                                                           SPAW1179
      TW=TW+W                                                           SPAW1180
   35 YB=YI+YRCT(I)                                                     SPAW1181
      CALL VELOTH(FELTBI)                                               SPAW1182
      IF(.NOT.FELTBI) GO TO 40                                          SPAW1183
      TU=TU+U                                                           SPAW1184
      TV=TV+V                                                           SPAW1185
      TW=TW+W                                                           SPAW1186
      GO TO 40                                                          SPAW1187
C                                                                       SPAW1188
C     CORNER 4 FOR EM2 .LT. 0                                           SPAW1189
C                                                                       SPAW1190
   21 CONTINUE                                                          SPAW1191
      XB=XI+XRHT(I)                                                     SPAW1192
      YB=YI+YRCT(I)                                                     SPAW1193
      CALL VELOTH(FELTB)                                                SPAW1194
      IF(.NOT.FELTB) GO TO 28                                           SPAW1195
      TU=TU+U                                                           SPAW1196
      TV=TV+V                                                           SPAW1197
      TW=TW+W                                                           SPAW1198
   28 YB=(YI+YRCT(I))                                                   SPAW1199
      CALL VELOTH(FELTBI)                                               SPAW1200
      IF(.NOT.FELTBI) GO TO 40                                          SPAW1201
      TU=TU+U                                                           SPAW1202
      TV=TV+V                                                           SPAW1203
      TW=TW+W                                                           SPAW1204
C                                                                       SPAW1205
C     SIGN CHANGES NECESSARY TO ACCOMMODATE USE OF SUBROUTINE           SPAW1206
C     VELOTH WITH DIFFERENT COORDINATE SYSTEM                           SPAW1207
C                                                                       SPAW1208
   40 CONTINUE                                                          SPAW1209
      TU=-TU                                                            SPAW1210
      TW=-TW                                                            SPAW1211
C                                                                       SPAW1212
C     CALCULATION OF PERTURBATION VELOCITIES                            SPAW1213
C                                                                       SPAW1214
      UP=UP+TU*DZDX(I)                                                  SPAW1215
      VP=VP+TV*DZDX(I)                                                  SPAW1216
      WP=WP+TW*DZDX(I)                                                  SPAW1217
  100 CONTINUE                                                          SPAW1218
      RETURN                                                            SPAW1219
      END                                                               SPAW1220
```

```
      SUBROUTINE WITHIN                                                 SPAW2  1
C                                                                       SPAW2  2
C     SUBROUTINE TO INPUT WING THICKNESS DATA                           SPAW2  3
C                                                                       SPAW2  4
      COMMON/INDEX/NC,MS,MSP,NPANLS,NCH,NBDER1,NBDER2,NBD,NBIP,MP,      SPAW2  5
     1   NCP,MSP,N1P,N2,N2P,NPTOT                                       SPAW2  6
      COMMON /THKDAT/NC=B,NCPS,MS,MPS  ,NTHP,XRFT(400),XRHT(400),       SPAW2  7
     1   XLFT(400),XLBT(400),YRCT(400),YLCT(400),ZRCT(400),ZLCT(400),   SPAW2  8
     2   THETAL(400),THETPL(200),SLLPT(400),SLTET(400),DZDX(400)        SPAW2  9
C                                                                       SPAW2 10
  701 FORMAT (10I5)                                                     SPAW2 11
  702 FORMAT (8F10.0)                                                   SPAW2 12
  703 FORMAT(//10X14,55H THICKNESS PANELS ARE TO BE LAID OUT ON EACH WINSPAW2 13
     1G PANEL/20X,I3,20H CHORDWISE ROWS WITH,13,12H IN EACH ROW/21X,43HISPAW2 14
     2HE CHORDWISE ROWS COINCIDE WITH THOSE USED/21X,                   SPAW2 15
     3 25HFOR THE U-VELOCITY PANELS//)                                  SPAW2 16
  704 FORMAT(//5X,90H***** SLOPE OF WING THICKNESS ENVELOPE AT LEADING ESPAW2 17
```

(kk) Page 37.

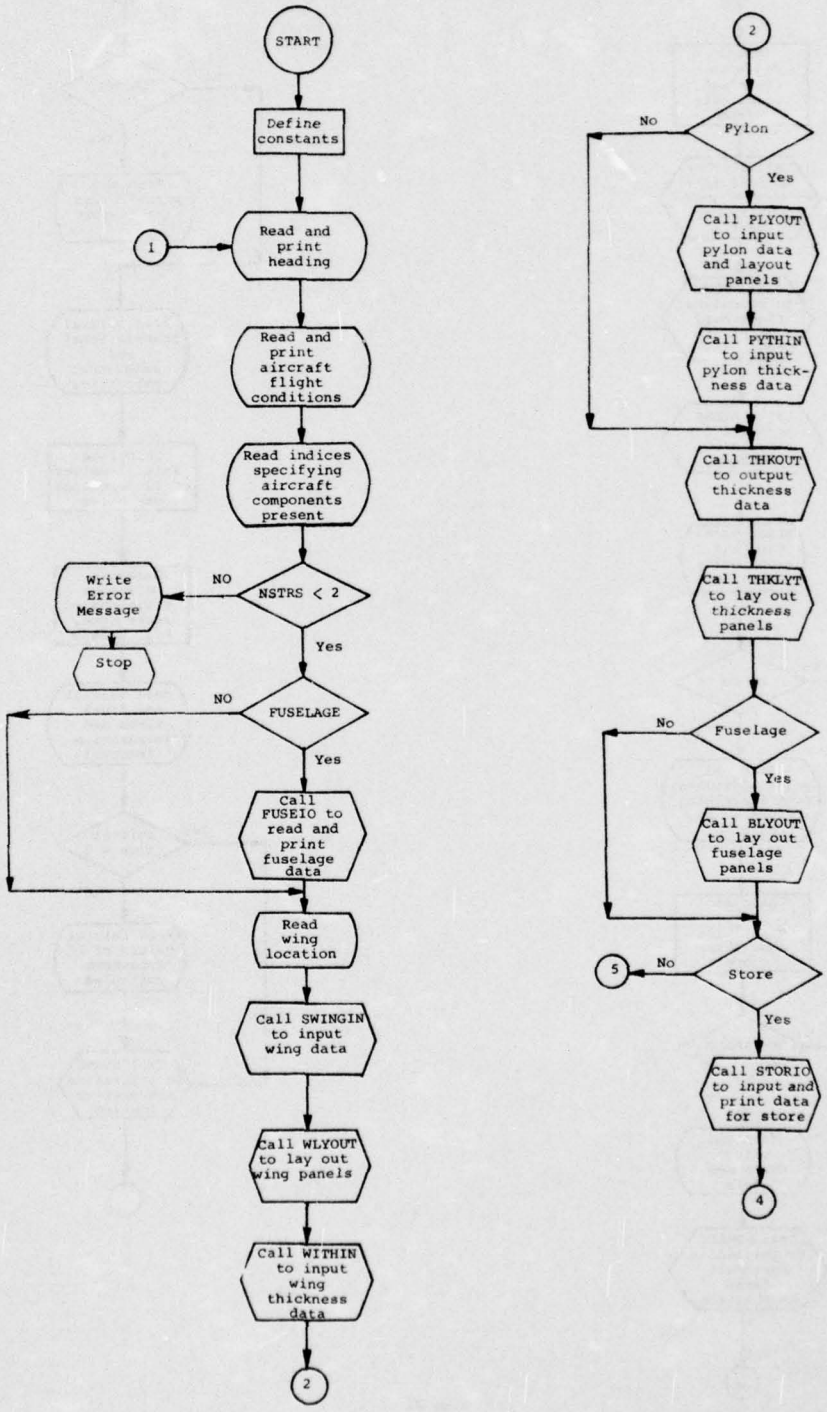
Figure 1.- Continued.

```
      IDGE NOT POSITIVE FOR SOME CHORDWISE ROW)                          SPA442 18
C                                                                        SPA442 19
      READ (5,701) NC=S,NUMIS                                            SPA442 20
      MSP=NC=S+MS=                                                       SPA442 21
      WRITE (6,703) MS,MSP,NC=S                                          SPA442 22
      IF (NUMIS .NE. 0) GO TO 5                                          SPA442 23
      MN=0                                                               SPA442 24
      DO 1 JN=1,MS,NC=S                                                  SPA442 25
      MN=MN+NC=S                                                         SPA442 26
    1 READ (5,702) (THETAL(J),J=JN,MN)                                   SPA442 27
      GO TO 10                                                           SPA442 28
    5 READ (5,702) (THETAL(J),J=1,NC=S)                                  SPA442 29
      DO 9 J=2,MS=                                                       SPA442 30
      JJ=(J-1)*NC=S                                                      SPA442 31
      DO 9 K=1,NC=S                                                      SPA442 32
      KK=JJ+K                                                            SPA442 33
    9 THETAL(KK)=THETAL(K)                                               SPA442 34
C                                                                        SPA442 35
C     CHECK FOR ZERO OR NEGATIVE LEADING EDGE SLOPE                      SPA442 36
C                                                                        SPA442 37
   10 DO 15 JLE=1,MS,NC=S                                                SPA442 38
   15 IF (THETAL(JLE),LE.0.0) GO TO 20                                   SPA442 39
      RETURN                                                             SPA442 40
   20 WRITE (6,704)                                                      SPA442 41
      STOP                                                               SPA442 42
      END                                                                SPA442 43
```

```
      SUBROUTINE =LYOUT                                                  SPA443  1
C                                                                        SPA443  2
C     SUBROUTINE TO LAYOUT CONSTANT D=VELOCITY PANELS ON THE =ING        SPA443  3
C                                                                        SPA443  4
      COMMON/BPGEOM/                    SNTB(100),CST2(100),ZLC(200),    SPA443  5
     1   ZRC(200),TMTI(100),SANGW,EANG=                                  SPA443  6
      COMMON /CONFIG/NFU,NPY,NSTRS,LVS=P                                 SPA443  7
      COMMON /CONSTS/PI,PI2,DTOR,RTOD,FOURPI                             SPA443  8
      COMMON/INDEX/NCW,MSW,MSWP,NPANLS,NCWS,NSDCW1,NSDCW2,NBD,NBIP,MP,   SPA443  9
     1   NCP,MSP,N1P,N2,N2P,NPTOT                                        SPA443 10
      COMMON /WGEOM/XBWUC,ZBWU,CRW,SLPWLE,SLPWTE,PSIWLE(20),PSIWTE(20),  SPA443 11
     1  Y(80)                                                           SPA443 12
      COMMON/WPGEOM/XRF(200),XRB(200),XLF(200),XLB(200),YRC(200),        SPA443 13
     1   YLC(200),XCPT(200),YCPT(200),ZCPT(200),SWPPLE(200),SWPPTE(200)  SPA443 14
C                                                                        SPA443 15
      DATA FAC/0,95/                                                     SPA443 16
C                                                                        SPA443 17
C     LAY OUT ELEMENTAL PANELS ON LEFT =ING PANEL                        SPA443 18
C                                                                        SPA443 19
      LVS=P=0                                                            SPA443 20
      DO 10 I=3,MSWP                                                     SPA443 21
      IM=I-1                                                             SPA443 22
   10 IF(PSI=LE(I),NE,PSI=LE(IM),OR,PSI=TP(I),NE,PSI=TE(IM)) LVSWP=1     SPA443 23
      SLP=LE=TAN(PSI=LE(MSWP)*DTOR)                                      SPA443 24
      SLP=TE=TAN(PSI=TE(MSWP)*DTOR)                                      SPA443 25
      SLPDIF=SLP=LE-SLP=TE                                               SPA443 26
      CSIDEP=CRW                                                         SPA443 27
      =LEX=0,0                                                           SPA443 28
      ANC=WNC=                                                           SPA443 29
C                                                                        SPA443 30
C     LOOP OVER =ING CHORDWISE ROWS                                      SPA443 31
C                                                                        SPA443 32
      DO 140 I=2,MSWP                                                    SPA443 33
      IM=I-1                                                             SPA443 34
      IF(LVSWP,EQ,0) GO TO 40                                            SPA443 35
      SLP=LE=TAN(PSI=LE(I)*DTOR)                                         SPA443 36
      SLP=TE=TAN(PSI=TE(I)*DTOR)                                         SPA443 37
      SLPDIF=SLP=LE-SLP=TE                                               SPA443 38
   40 CSIDE=CSIDEP+(Y(I)-Y(IM))*SLPDIF                                   SPA443 39
      IF(I,EQ,2) GO TO 45                                                SPA443 40
      JLE=(I-3)*NC=+1                                                    SPA443 41
      =LEX=XLF(JLE)                                                      SPA443 42
```

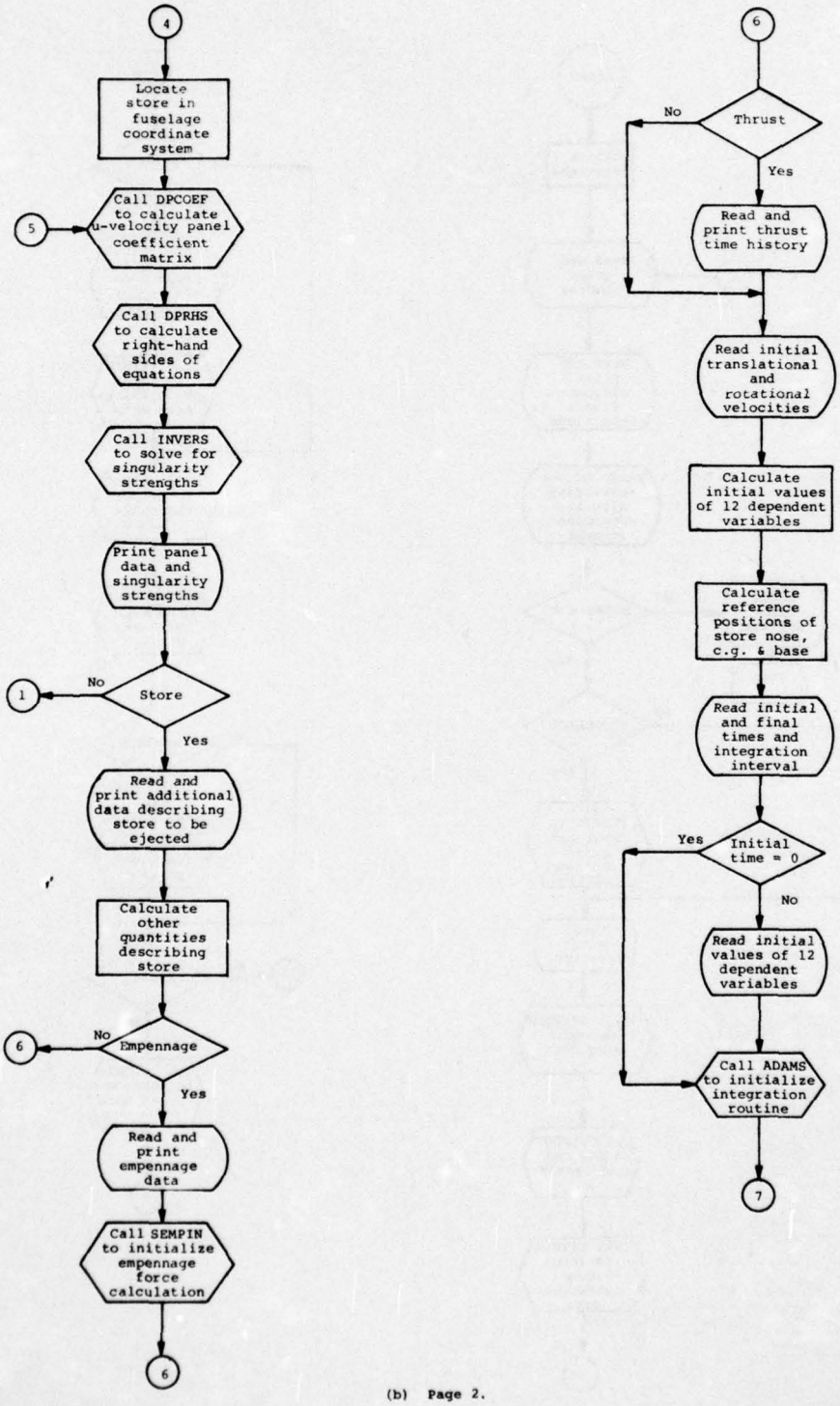
```
   45 CONTINUE                                                           SPA443 43
C                                                                        SPA443 44
C     LOOP OVER ELEMENTAL PANELS IN THIS CHORDWISE ROW                   SPA443 45
C                                                                        SPA443 46
      DO 130 K=1,NC=                                                     SPA443 47
      J=(I-2)*NC=+K                                                      SPA443 48
      AK=K                                                               SPA443 49
      AKM=K-1                                                            SPA443 50
      S=PPLE(J)=SLP=LE-AKM*SLPDIF/ANC=                                   SPA443 51
      S=PPTE(J)=SLP=LE-AK*SLPDIF/ANC=                                    SPA443 52
C                                                                        SPA443 53
C     CORNER POINT COORDINATES                                           SPA443 54
C                                                                        SPA443 55
      XRF(J)=-AKM*CSIDEP/ANC=+=LEX                                       SPA443 56
      XRB(J)=XRF(J)-CSIDEP/ANC=                                          SPA443 57
      XLF(J)= -AKM*CSIDE/ANC=+(Y(I)-Y(IM))*SLP=LE+=LEX                   SPA443 58
      XLB(J)=XLF(J)-CSIDE/ANC=                                           SPA443 59
      YRC(J)=Y(IM)                                                       SPA443 60
      YLC(J)=Y(I)                                                        SPA443 61
      ZRC(J)=0,0                                                         SPA443 62
      ZLC(J)=0,0                                                         SPA443 63
C                                                                        SPA443 64
C     Y=COORDINATES OF CONTROL POINT AND PANEL CENTROID                  SPA443 65
C                                                                        SPA443 66
      A1=XRB(J)-XRF(J)                                                   SPA443 67
      A2=XLB(J)-XLF(J)                                                   SPA443 68
      H=Y(IM)-Y(I)                                                       SPA443 69
      YBAR=(2,0*A1+A2)*H/(3,0*(A1+A2))                                   SPA443 70
      YCPT(J)=Y(IM)+YBAR                                                 SPA443 71
C                                                                        SPA443 72
C     X=COORDINATES OF CONTROL POINT                                     SPA443 73
C                                                                        SPA443 74
      XPCLE=XRF(J)+YBAR*SWPPLE(J)                                        SPA443 75
      XPCTE=XRB(J)+YBAR*SWPPTE(J)                                        SPA443 76
      PNLC=XPCLE-XPCTE                                                   SPA443 77
      XCPT(J)=XPCLE+FAC*PNLC                                             SPA443 78
C                                                                        SPA443 79
C     Z=COORDINATES OF CONTROL POINT                                     SPA443 80
C                                                                        SPA443 81
      ZCPT(J)=0,                                                         SPA443 82
  130 CONTINUE                                                           SPA443 83
      CSIDEP=CSIDE                                                       SPA443 84
  140 CONTINUE                                                           SPA443 85
      RETURN                                                             SPA443 86
      END                                                                SPA443 87
```


Figure 1.- Concluded.



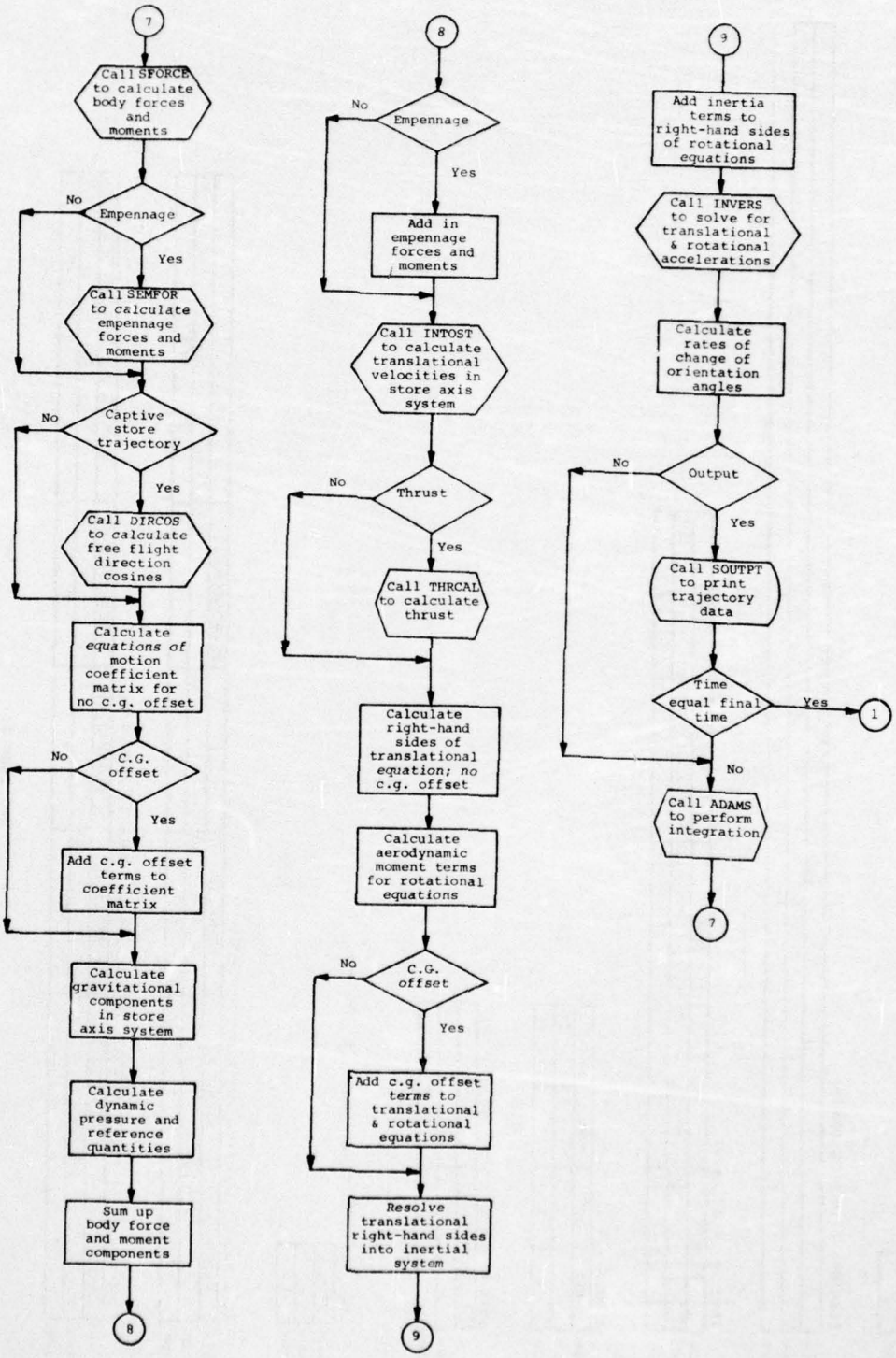
(a) Page 1.

Figure 2.- General flow chart of trajectory program.



(b) Page 2.

Figure 2.- Continued.



(c) Page 3.

Figure 2.- Concluded.

Item No. 1 (1 card)

Variable	NCARDS
Card Column	5
Format Type	I

Item No. 2 (NCARDS cards)

Variable	HEAD
Card Column	1 ... 80
Format Type	A

Item No. 3 (1 card)

Variable	ALFAC	GAMF	FMACH	RHO	VINF
Card Column	10	20	30	40	50
Format Type	F	F	F	F	F

Item No. 4 (1 card)

Variable	NFU	NPY	NDMY	NSTRS
Card Column	5	10	15	20
Format Type	I	I	I	I

Item No. 5 (1 card, omit if NFU = 0)

Variable	FLTHC	FRMAX
Card Column	10	20
Format Type	F	F

Item No. 6 (1 card. Omit if NFU = 0)

Variable	NFPOLY
Card Column	5
Format Type	I

Item No. 7 (1 card. Omit if NFU = 0)

Variable	FXEND(1)	FXEND(2)	FXEND(3)	...	FXEND(NFPOLY)		
Card Column	10	20	30	40	50	60	70
Format Type	F	F	F	F	F	F	F

Item No. 8 (J = 1 to NFPOLY; NFPOLY cards. Omit if NFU = 0)

Variable	FCOEF(J,1)	FCOEF(J,2)	FCOEF(J,3)	FCOEF(J,4)	FCOEF(J,5)	FCOEF(J,6)	FCOEF(J,7)
Card Column	10	20	30	40	50	60	70
Format Type	F	F	F	F	F	F	F

(a) Page 1.

Figure 3.- Trajectory program input format.

Item No. 9 (1 card. Omit if NFU = 0)

Variable	NCWB	NBDCR1	NBDCR2	NFSOR
Card Column	5	10	15	20
Format Type	I	I	I	I

Item No. 10 (1 card. Omit if NFU = 0)

Variable	BODYPL
Card Column	10
Format Type	F

Item No. 11 (1 card)

Variable	XBWOC	ZBWO
Card Column	10	20
Format Type	F	F

Item No. 12 (1 card)

Variable	CRW	SSPAN
Card Column	10	20
Format Type	F	F

Item No. 13 (1 card)

Variable	NCW	MSW
Card Column	5	10
Format Type	I	I

Item No. 14 (I = 1 to MSW+1; MSW+1 cards)

Variable	I	Y(I)	PSIWLE(I)	PSIWTE(I)
Card Column	5	15	25	35
Format Type	I	F	F	F

(b) Page 2.

Figure 3.- Continued.

Item No. 15 (1 card)

Variable	NTAC	NUNI
Card Column	5	10
Format Type	I	I

Item No. 16 (NCW values, eight to a card. Omit if NTAC = 0. One set of cards if NUNI = 1; MSW sets if NUNI = 0.)

Variable	ALPHAL(1)	ALPHAL(2)	...	ALPHAL(NCW)				
Card Column	10	20	30	40	50	60	70	80
Format Type	F	F	F	F	F	F	F	F

Item No. 17 (1 card)

Variable	NCWS	NUNIS
Card Column	5	10
Format Type	I	I

Item No. 18 (NCWS values, eight to a card. One set of cards if NUNIS = 1; MSW sets if NUNIS = 0.)

Variable	THETAL(1)	THETAL(2)	...	THETAL(NCWS)				
Card Column	10	20	30	40	50	60	70	80
Format Type	F	F	F	F	F	F	F	F

Item No. 19 (1 card. Omit if NPY = 0.)

Variable	IP	PSIPLE	PSIPTE	CRP	HP	XPLE
Card Column	5	15	25	35	45	55
Format Type	I	F	F	F	F	F

Item No. 20 (1 card. Omit if NPY = 0.)

Variable	NCP	MSP
Card Column	5	10
Format Type	I	I

(c) Page 3.

Figure 3.- Continued.

Item No. 21 (K = 1 to MSP+1; MSP+1 cards. Omit if NPY = 0.)

Variable	K	Z(K)
Card Column	5	15
Format Type	I	F

Item No. 22 (1 card. Omit if NPY = 0.)

Variable	NCPS	NUNIP
Card Column	5	10
Format Type	I	I

Item No. 23 (NCPS values, eight to a card. Omit if NPY = 0. One set of cards if NUNIP = 1; MSP sets if NUNIP = 0.)

Variable	THETPL(1)	THETPL(2)	...	THETPL(NCPS)				
Card Column	10	20	30	40	50	60	70	80
Format Type	F	F	F	F	F	F	F	F

Note: If NSTRS = 0 this completes the input data.

Item No. 24 (1 card)

Variable	NUMSTR(1)	NSHAPE(1)	SLTSC(1)	SRMAX(1)	XSNC(1)	YSN(1)	ZSN(1)	SIC(1)
Card Column	5	10	20	30	40	50	60	70
Format Type	I	I	F	F	F	F	F	F

Item No. 25 (1 card)

Variable	NSHPT
Card Column	5
Format Type	I

Item No. 26 (1 card)

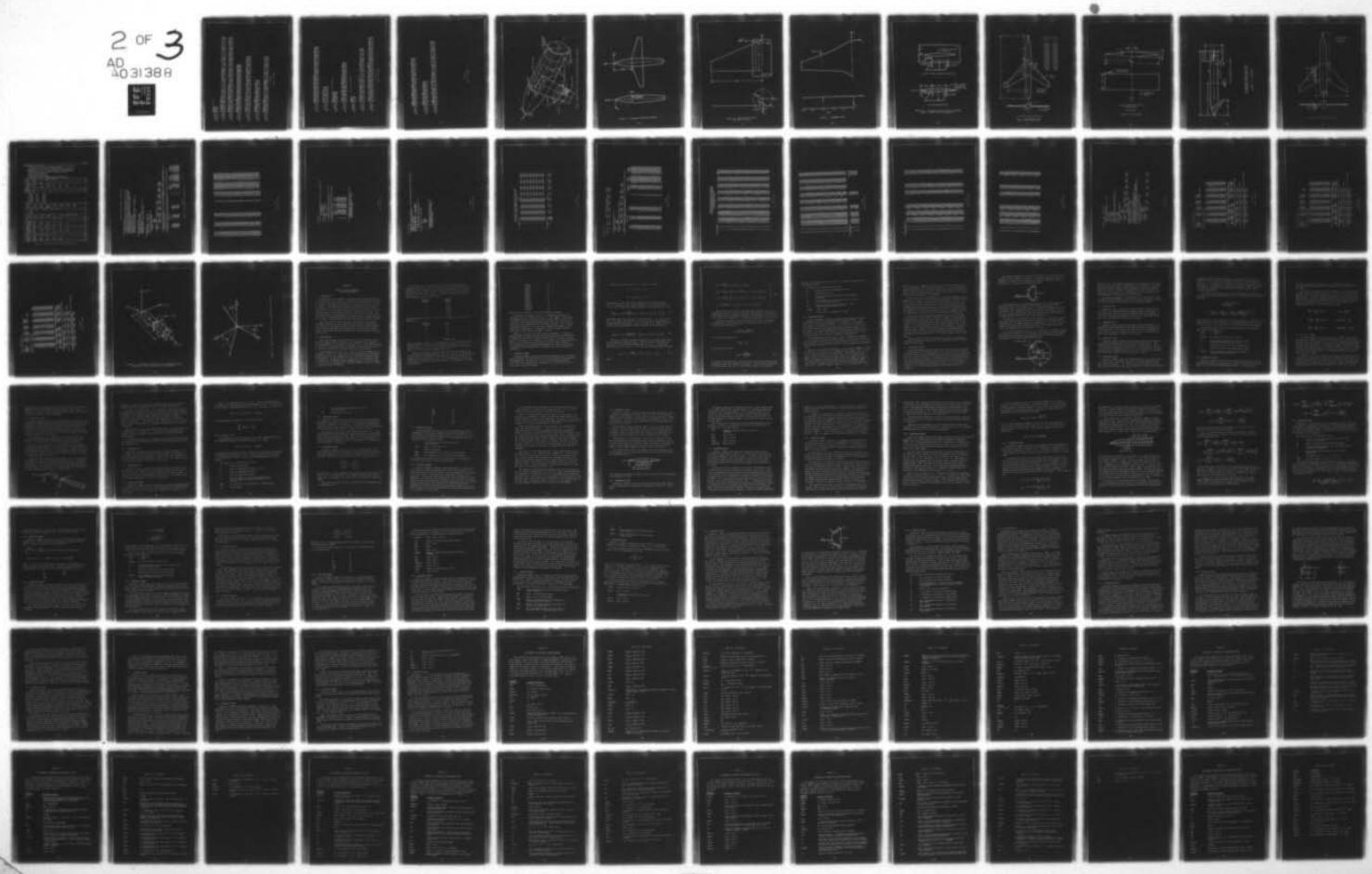
Variable	MSHAPE	MSOR
Card Column	5	10
Format Type	I	I

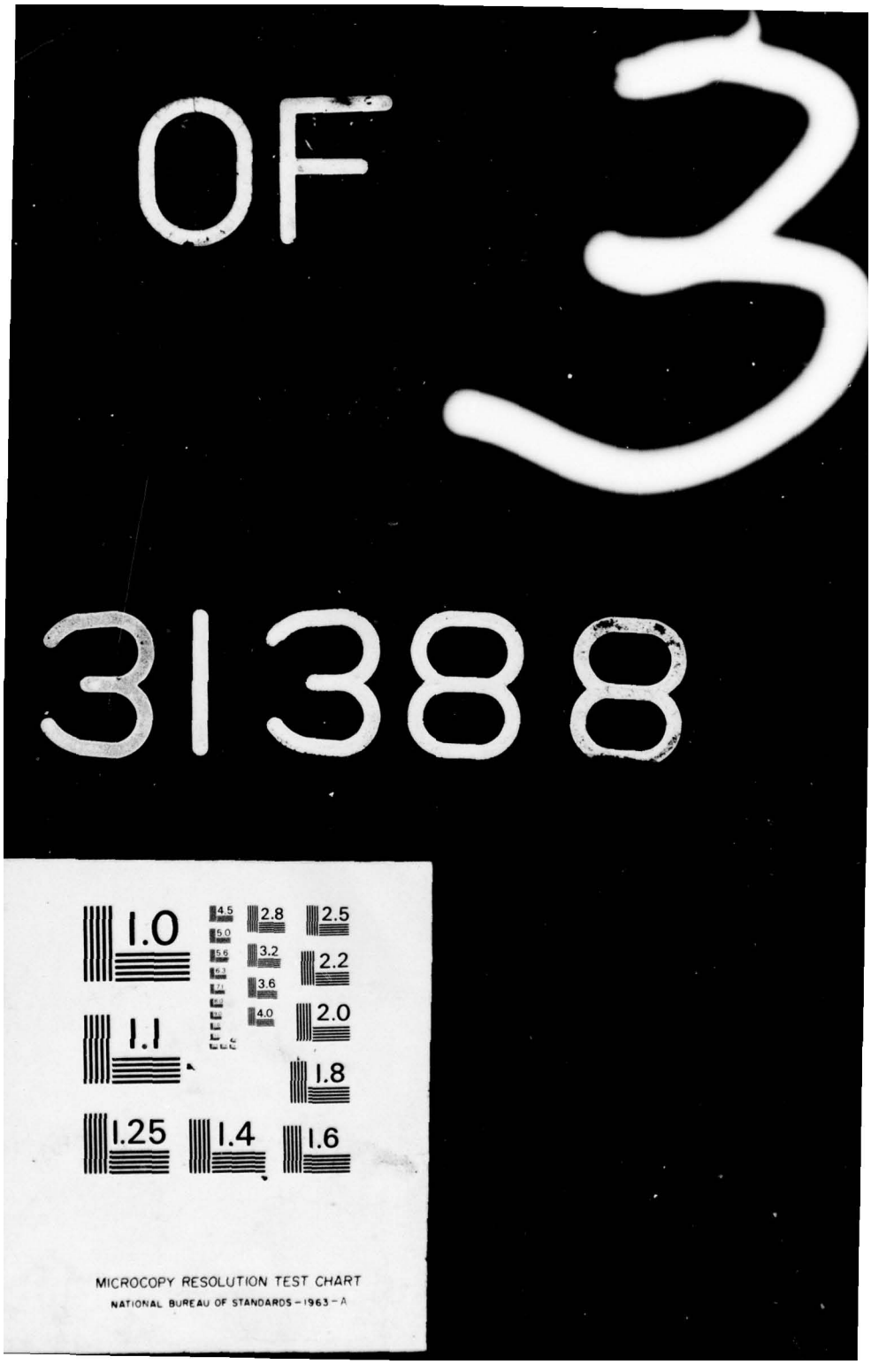
(d) Page 4.

Figure 3.- Continued.

AD-A031 388 NIELSEN ENGINEERING AND RESEARCH INC MOUNTAIN VIEW CALIF F/G 1/3
PREDICTION OF SUPERSONIC STORE SEPARATION CHARACTERISTICS. VOLU--ETC(U)
MAY 76 F K GOODWIN, M M KEIRSTEAD F33615-75-C-3053
UNCLASSIFIED NEAR-TR-106 AFFDL-TR-76-41-VOL-2 NL

2 OF 3
AD A031388




OF 3
31388
1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
6.3
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS - 1963 - A

Item No. 27 (1 card)

Variable	NSPOLJ
Card Column	5
Format Type	I

Item No. 28 (1 card)

Variable	SXNDJ(1)	SXNDJ(2)	. . . .	SXNDJ(NSPOLJ)			
Card Column	10	20	30	40	50	60	70
Format Type	F	F	F	F	F	F	F

Item No. 29 (J = 1, NSPOLJ; NSPOLJ cards)

Variable	SCOFJ(J,1)	SCOFJ(J,2)	SCOFJ(J,3)	SCOFJ(J,4)	SCOFJ(J,5)	SCOFJ(J,6)	SCOFJ(J,7)
Card Column	10	20	30	40	50	60	70
Format Type	F	F	F	F	F	F	F

Item No. 30 (1 card)

Variable	NEJECT	NSEG	NSEGXO	NGAM	NPOLY	NROLL	NEMP	NDAMP	NTHRUS
Card Column	5	10	15	20	25	30	35	40	45
Format Type	I	I	I	I	I	I	I	I	I

Item No. 31 (1 card)

Variable	SMASS	FIXX	FIYY	FIZZ	FIYZ	FIXZ	FIXY
Card Column	10	20	30	40	50	60	70
Format Type	F	F	F	F	F	F	F

Item No. 32 (1 card)

Variable	XMOM	XBAR	YBAR	ZBAR
Card Column	10	20	30	40
Format Type	F	F	F	F

Item No. 33 (1 card)

Variable	XEND(1)	XEND(2)	...	XEND(NPOLY)			
Card Column	10	20	30	40	50	60	70
Format Type	F	F	F	F	F	F	F

(e) Page 5.

Figure 3.- Continued.

Item No. 34 (J = 1 to NPOLY; NPOLY cards)

Variable	COEF(J,1)	COEF(J,2)	COEF(J,3)	COEF(J,4)	COEF(J,5)	COEF(J,6)	COEF(J,7)
Card Column	10	20	30	40	50	60	70
Format Type	F	F	F	F	F	F	F

Item No. 35 (1 card)

Variable	CA	CDC
Card Column	10	20
Format Type	F	F

Item No. 36 (1 card. Omit if NEMP = 0)

Variable	IPLNR	MSF
Card Column	5	10
Format Type	I	I

Item No. 37 (1 card. Omit if NEMP = 0)

Variable	XTAIL	RADAV	FINSS	PHIROL	CLALPH
Card Column	10	20	30	40	50
Format Type	F	F	F	F	F

Item No. 38 (1 card. Omit if NTHRUS = 0)

Variable	NTPOLY
Card Column	5
Format Type	I

Item No. 39 (1 card. Omit if NTHRUS = 0)

Variable	TEND(1)	TEND(2)	. . . .	TEND(NTPOLY)	
Card Column	10	20	30	40	50
Format Type	F	F	F	F	F

Item No. 40 (J = 1, NTPOLY; NTPOLY cards. Omit if NTHRUS = 0)

Variable	TC(J,1)	TC(J,2)	TC(J,3)	TC(J,4)	TC(J,5)	TC(J,6)
Card Column	10	20	30	40	50	60
Format Type	F	F	F	F	F	F

(f) Page 6.

Figure 3.- Continued.

Item No. 41 (1 card)

Variable	VXZERO	VYZERO	VZZERO	VAR(4)	VAR(5)	VAR(6)
Card Column	10	20	30	40	50	60
Format Type	F	F	F	F	F	F

Item No. 42 (1 card)

Variable	DTIME	TIMEI	TIMEF
Card Column	10	20	30
Format Type	F	F	F

Item No. 43 (2 cards. Omit if TIMEI = 0)

Variable	VAR(1)	VAR(2)	...	VAR(12)				
Card Column	10	20	30	40	50	60	70	80
Format Type	F	F	F	F	F	F	F	F

(g) Page 7.

Figure 3.- Concluded.

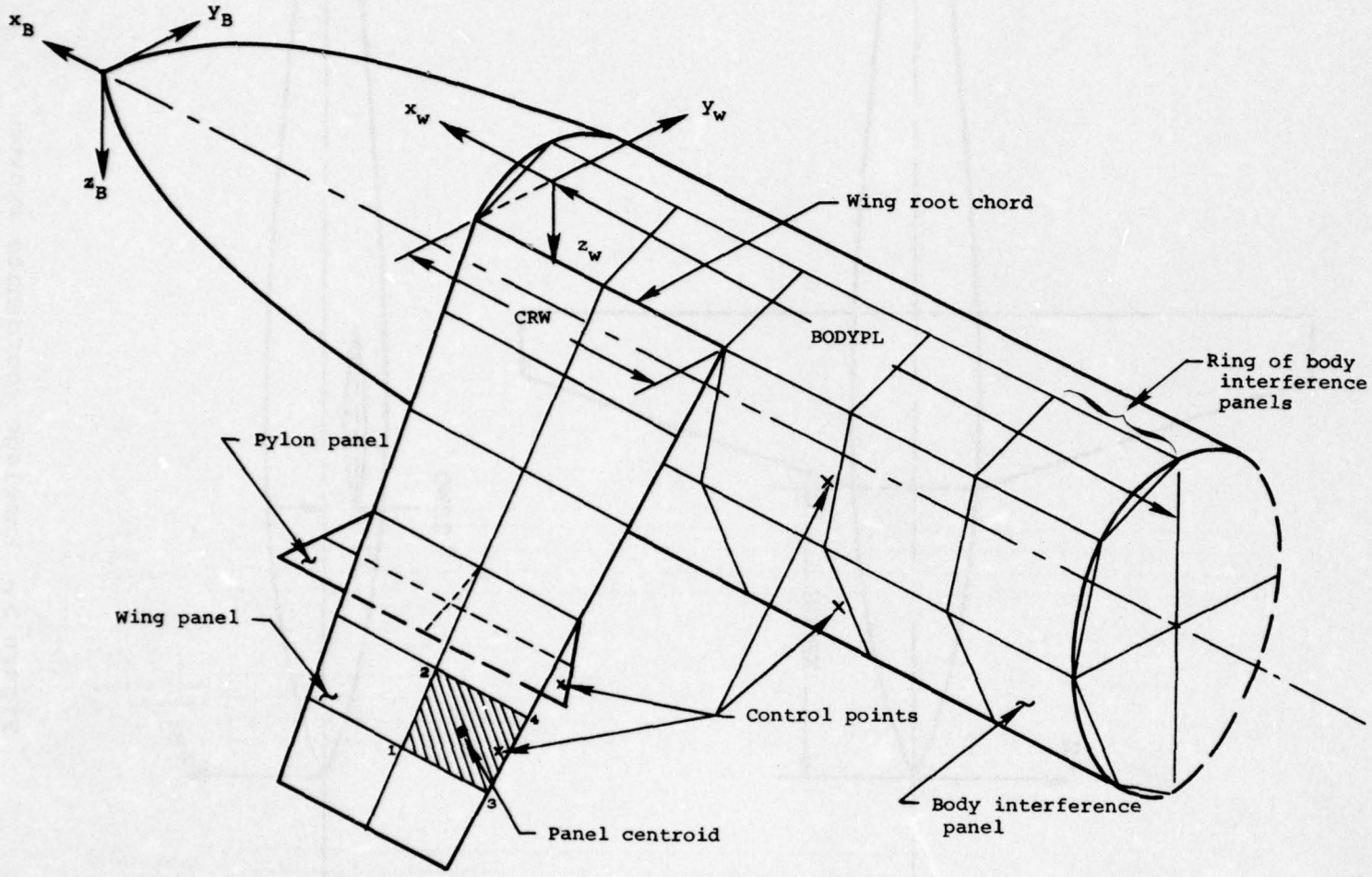


Figure 4.- Simplified layout of panels for wing-pylon and fuselage combination.

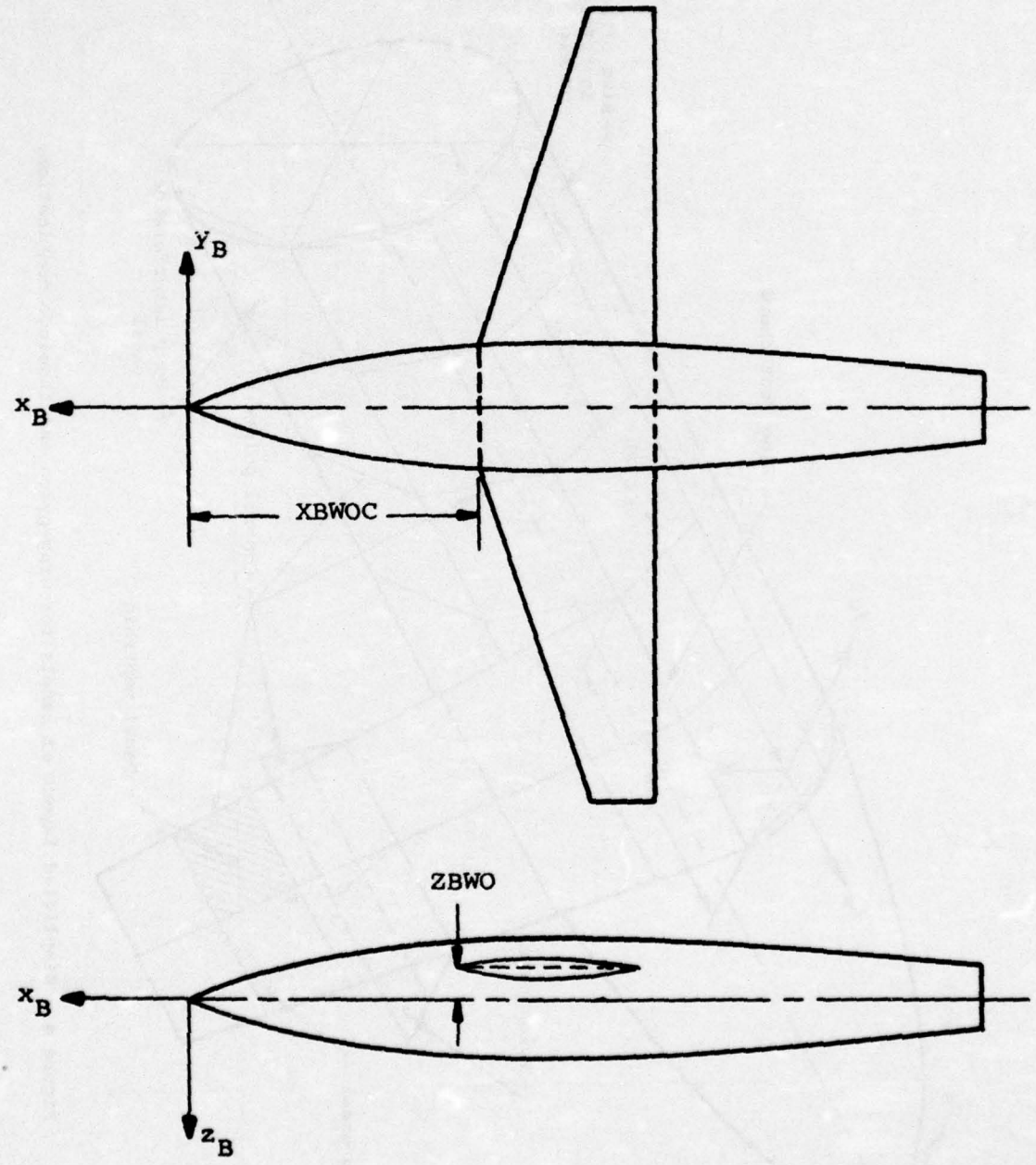


Figure 5.- Fuselage coordinate system.

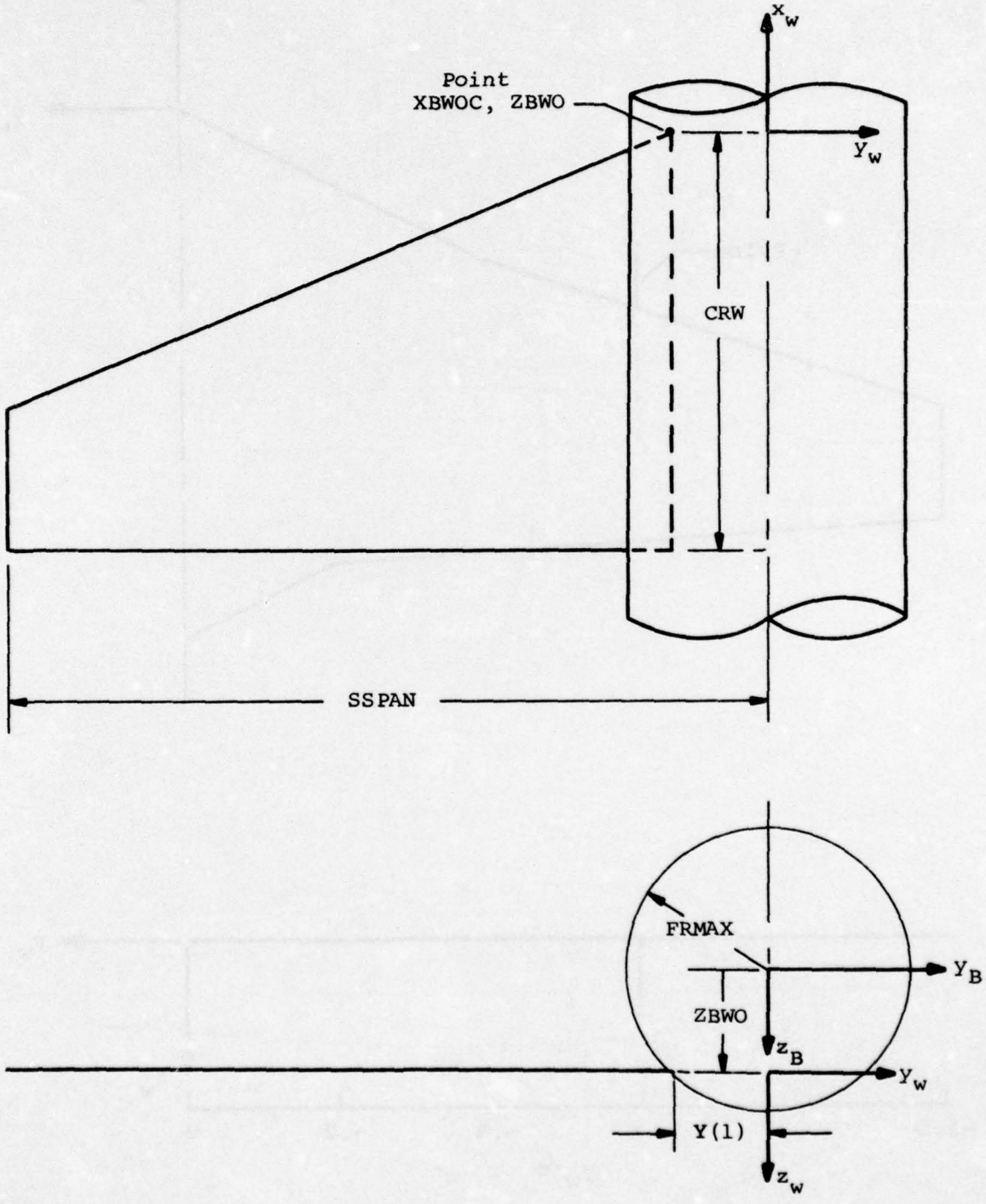


Figure 6.- Definition of certain wing input variables.

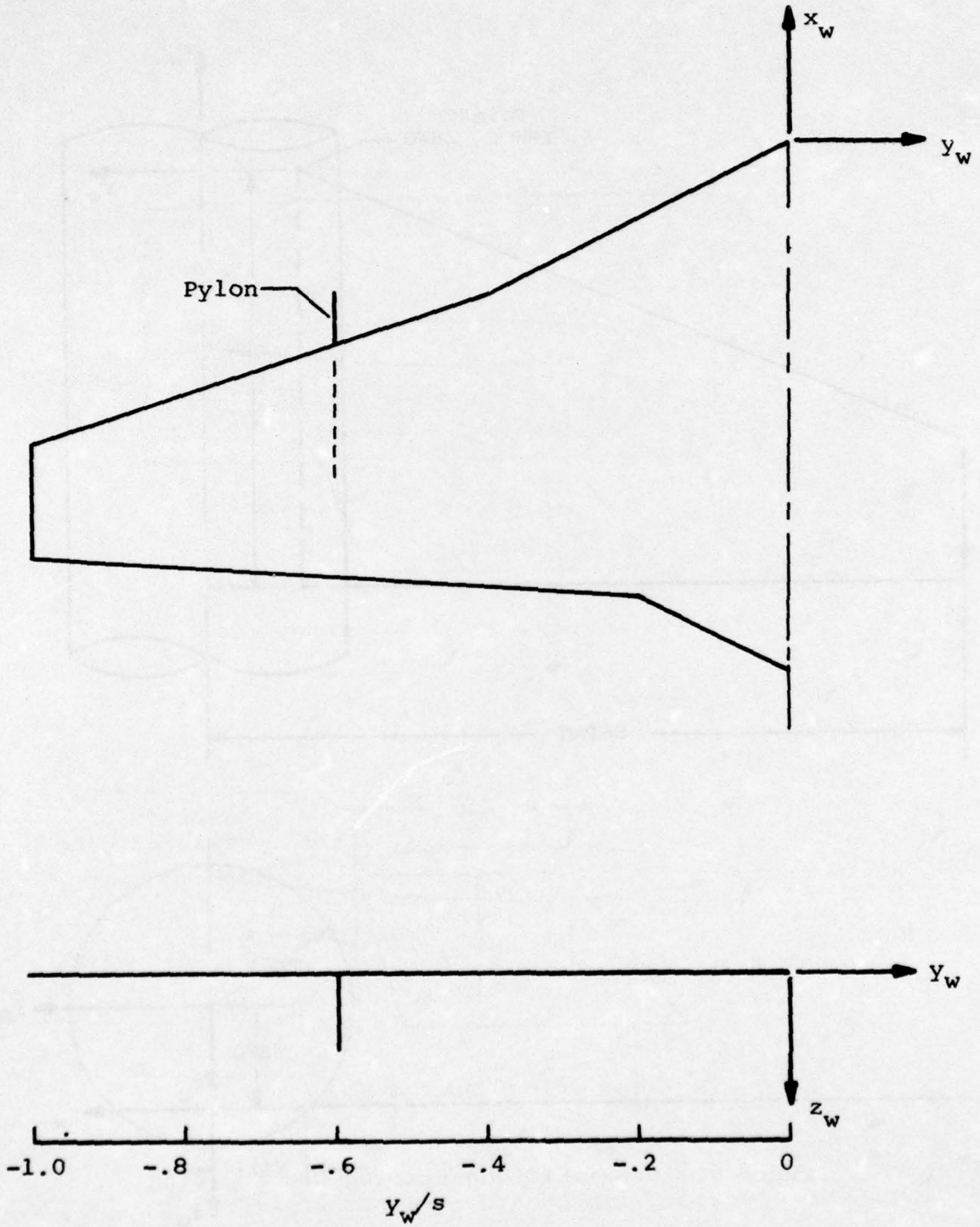
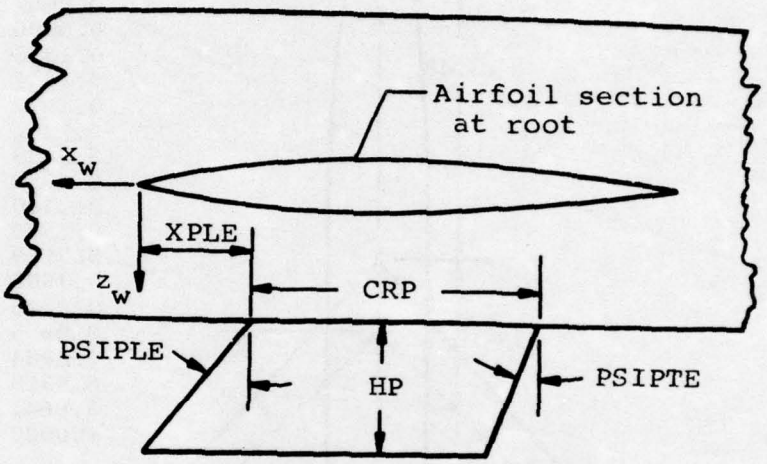
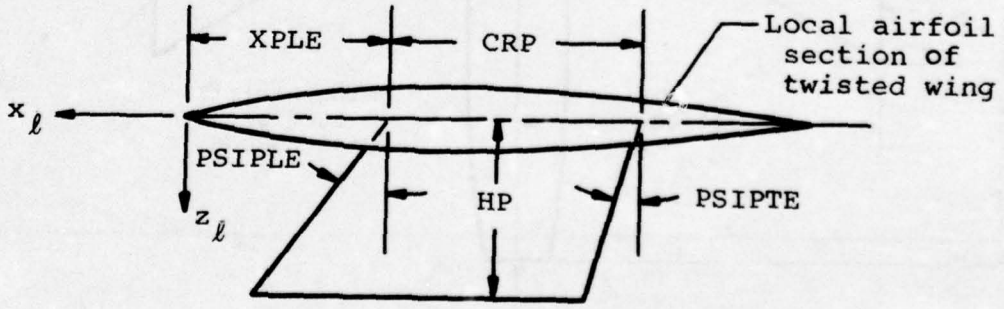


Figure 7 .- Example wing.

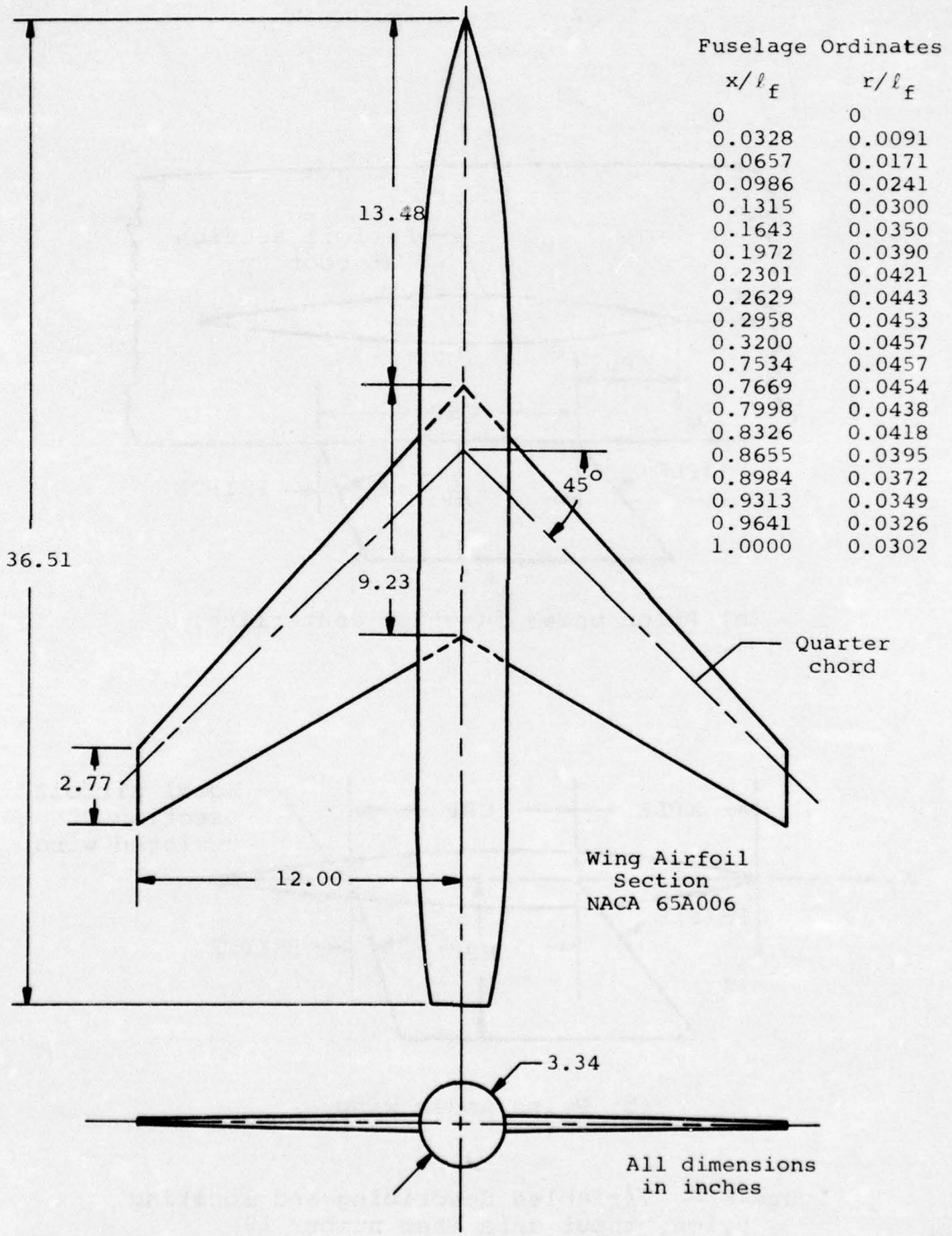


(a) Pylon under fuselage centerline.



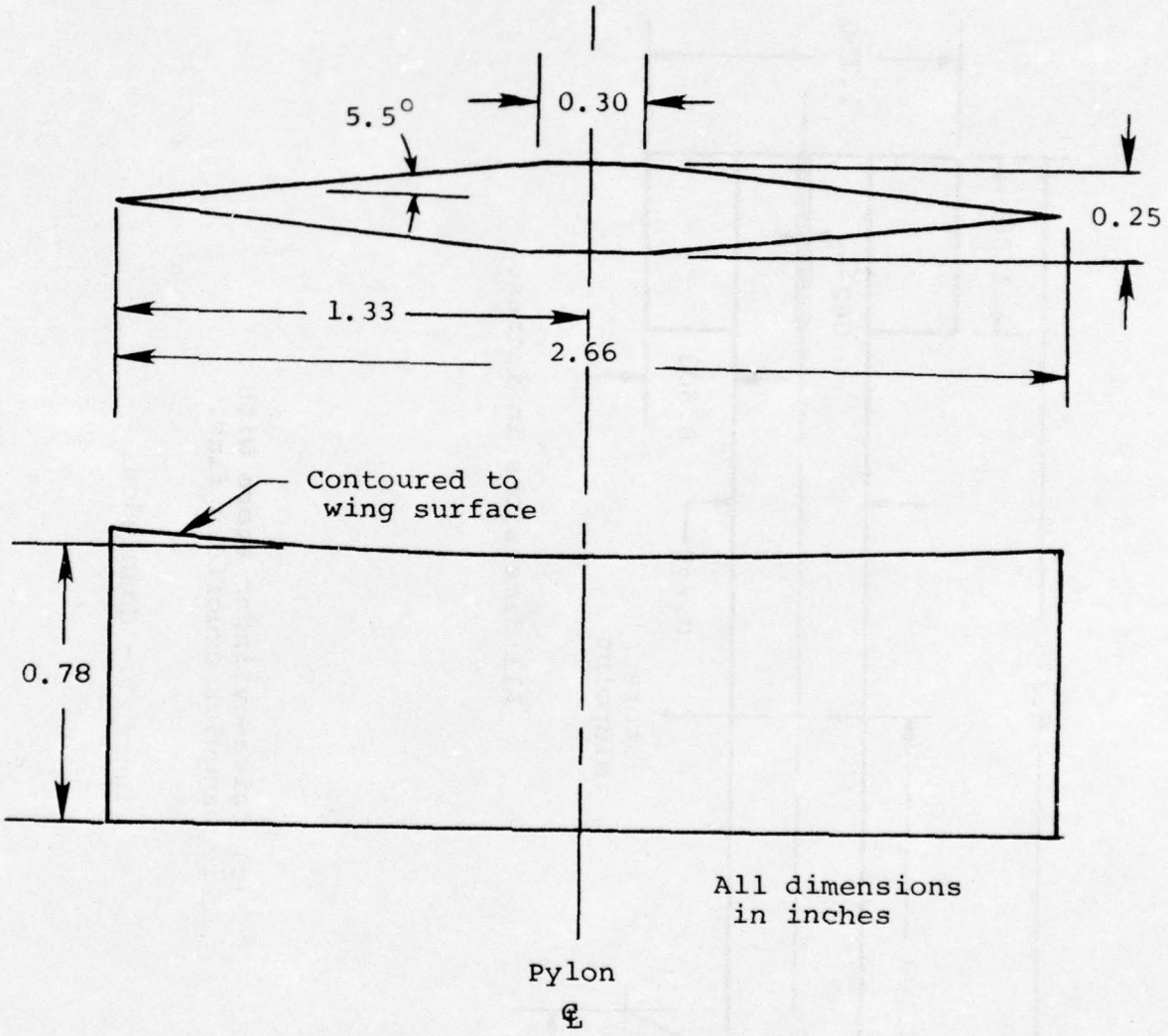
(b) Pylon under wing.

Figure 8.- Variables describing and locating pylon, input data item number 19.



(a) Wing-fuselage combination.

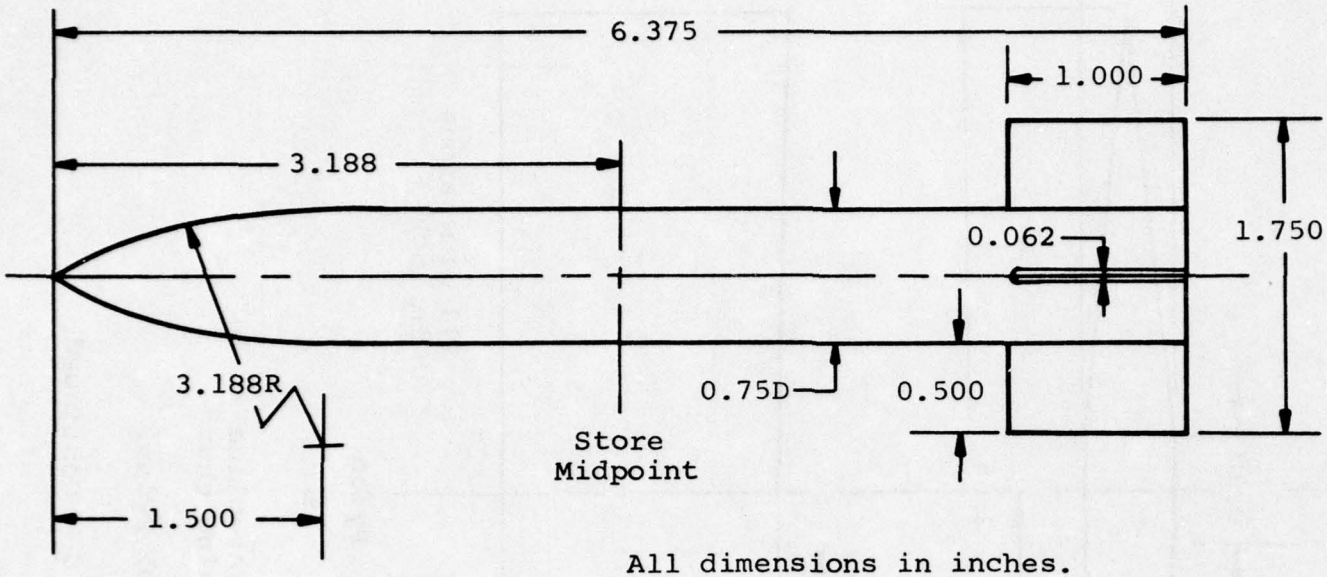
Figure 9.- Wind-tunnel models used in sample calculation.



Pylon centerline at 40°
of wing chord

(b) Pylon.

Figure 9.- Continued.



All dimensions in inches.

(c) Ogive-cylinder store with rectangular cruciform fins.

Figure 9.- Concluded.

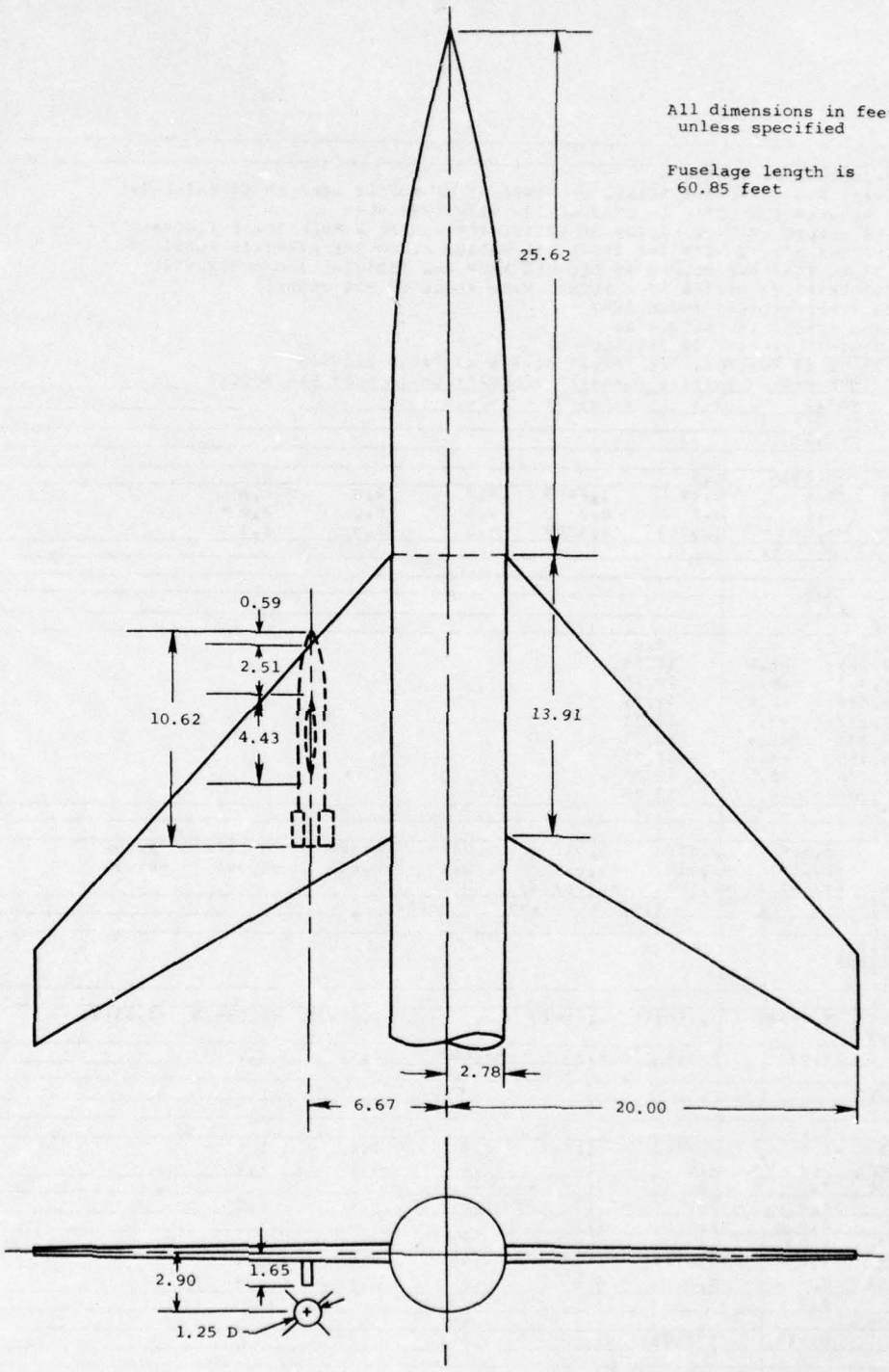


Figure 10.- Configuration for sample trajectory.

```
                                                                                                                    Item No.
12                                                                                                                      1
SAMPLE TRAJECTORY CALCULATION
    AIRCRAFT AND STORE ARE SCALED UP VERSIONS OF MODELS USED IN EXPERIMENTAL
        PROGRAM CONDUCTED IN CONJUNCTION WITH THIS WORK
    MODELS SCALED UP BY A FACTOR OF 20 TO APPROXIMATE A FULL SCALE AIRCRAFT
    TRAJECTORY STARTS WITH THE STORE ONE RADIUS BELOW THE CARRIAGE POSITION
    THE STORE FINS ARE ROLLED 45 DEGREES FROM THE VERTICAL AND HORIZONTAL                                           2
    THE AIRCRAFT IS DIVING AT A FLIGHT PATH ANGLE OF -30 DEGREES
    IT IS A FREE-FLIGHT TRAJECTORY
    ROLLING MOMENT IS CALCULATED
    AERODYNAMIC DAMPING IS INCLUDED
    THE STORE IS POWERED, THE THRUST STARTS AT T=0.0 SECONDS
    THE STORE HAS AN INITIAL DOWNWARD VELOCITY OF 10 FEET PER SECOND
2.0       -30.0       1.5       0.0008907  1492.0                                                                 3
  1   1     0     1                                                                                               4
60.85       2.78                                                                                                  5
  3                                                                                                               6
0.32        0.7534    1.0                                                                                         7
-1.0975     -1.0      0.64      1.2045     0.0       0.0       1.0
 0.0457      0.0      0.0       0.0        0.0       0.0       0.0                                               8
-1.9237     -1.0      1.5068    3.3109     0.0       0.0       1.0
12    4     4    52                                                                                               9
13.91                                                                                                           10
-25.62      0.0                                                                                                 11
13.91       20.0                                                                                                12
12    8                                                                                                         13
  1  -2.78      0.0       0.0
  2  -4.725     48.6      30.75
  3  -6.667     48.6      30.75
  4  -8.888     48.6      30.75
  5 -11.112     48.6      30.75                                                                                   14
  6 -13.333     48.6      30.75
  7 -15.555     48.6      30.75
  8 -17.778     48.6      30.75
  9 -20.000     48.6      30.75
  0                                                                                                             15
20    1                                                                                                         17
 0.209      0.109     0.077     0.057     0.043     0.029     0.017     0.006
-0.005     -0.014    -0.024    -0.034    -0.044    -0.053    -0.061    -0.066                                   18
-0.067     -0.067    -0.067    -0.067
  3   0.0       0.0       4.43      1.65      -2.51                                                             19
  4   2                                                                                                         20
  1   0.0
  2   0.825                                                                                                     21
  5   1.65
  9   1                                                                                                         22
 0.09629   0.09629   0.09629   0.09629   0.0       -0.09629  -0.09629  -0.09629                                  23
-0.09629
 10   2  10.625    0.625     0.59      -6.667    2.90      0.0                                                   24
  1                                                                                                             25
  2   30                                                                                                        26
  2                                                                                                             27
 0.2353     1.0                                                                                                28
-0.4413     -1.0      0.4706    0.1947    0.0       0.0       1.0                                               29
 0.05882    0.0       0.0       0.0       0.0       0.0       0.0
 10   20   20    0     2    1     1    1     1                                                                   30
15.53       16.0      400.0     400.0     0.0       0.0       0.0                                               31
-7.4        0.0       0.0       0.0                                                                             32
 0.2353     1.0                                                                                                33
-0.4413     -1.0      0.4706    0.1947    0.0       0.0       1.0                                               34
 0.05882    0.0       0.0       0.0       0.0       0.0       0.0
 0.65       0.0                                                                                                35
  0    5                                                                                                       36
-9.558      0.625     1.45833   45.0      3.491                                                               37
  2                                                                                                             38
 0.2        1.0                                                                                                39
1500.0      0.0       0.0       0.0       0.0       0.0                                                       40
1875.0      -1875.0   0.0       0.0       0.0       0.0
 0.0        0.0       10.0      0.0       0.0       0.0                                                       41
 0.025      0.0       0.4                                                                                      42
```

Figure 11.- Input data deck for sample trajectory calculation.

SUPERSONIC SIX-DEGREE-OF-FREEDOM TRAJECTORY PROGRAM

SAMPLE TRAJECTORY CALCULATION
AIRCRAFT AND STORE ARE SCALED UP VERSIONS OF MODELS USED IN EXPERIMENTAL
PROGRAM CONDUCTED IN CONJUNCTION WITH THIS WORK
MODELS SCALED UP BY A FACTOR OF 20 TO APPROXIMATE A FULL SCALE AIRCRAFT
TRAJECTORY STARTS WITH THE STORE ONE RADIUS BELOW THE CARRIAGE POSITION
THE STORE FINS ARE ROLLED 45 DEGREES FROM THE VERTICAL AND HORIZONTAL
THE AIRCRAFT IS DIVING AT A FLIGHT PATH ANGLE OF -30 DEGREES
IT IS A FREE-FLIGHT TRAJECTORY
ROLLING MOMENT IS CALCULATED
AERODYNAMIC DAMPING IS INCLUDED
THE STORE IS POWERED, THE THRUST STARTS AT T=0.0 SECONDS
THE STORE HAS AN INITIAL DOWNWARD VELOCITY OF 10 FEET PER SECOND

AIRCRAFT FLIGHT CONDITIONS
ANGLE OF ATTACK = 2.00 DEGREES
FLIGHT PATH ANGLE =-30.00 DEGREES
MACH NUMBER = 1.50
FREE STREAM MASS DENSITY = .0008907 SLUGS PER CUBIC FOOT
FREE STREAM VELOCITY = 1492.00 FEET PER SECOND

FUSELAGE INPUT DATA
FUSELAGE LENGTH = 60.85000 FEET
MAXIMUM RADIUS = 2.78000 FEET

POLYNOMIALS SPECIFYING FUSELAGE SHAPE
X/L OF END OF EACH SECTION

SECTION	X/L
1	.32000
2	.75340
3	1.00000

COEFFICIENTS OF POLYNOMIALS DESCRIBING EACH SECTION

SECTION	C1	C2	C3	C4	C5	C6	C7
1	-1.09750	-1.00000	.64000	1.20450	0.00000	0.00000	1.00000
2	.04570	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
3	-1.92370	-1.00000	1.50680	3.31090	0.00000	0.00000	1.00000

12 RINGS OF U-VELOCITY PANELS ARE TO BE LAID OUT ON THE FUSELAGE FROM XW= 0 FEET TO XW= -13.91000 FEET
THERE ARE 8 PANELS IN EACH RING ON THE LEFT HALF OF THE FUSELAGE, 4 ABOVE THE WING AND 4 BELOW THE WING
TOTAL NUMBER OF PANELS IS 96

FUSELAGE SHAPE AS CALCULATED FROM THE INPUT POLYNOMIALS, ORIGINS OF SOURCES AND DOUBLETS REPRESENTING THE FUSELAGE, AND VALUES OF THE SOURCE AND DOUBLET CONSTANTS

BODY SHAPE				SINGULARITY ORIGIN	SOURCE CONSTANTS	DOUBLET CONSTANTS
X, FT	R, FT	DR/DX	N	X(N), FT	K(N)	KD(N)
0.00000	-.00017	.29157	0	.00019	7.39679E-02	4.46090E-03
1.17019	.32995	.27270	1	.80129	-1.35779E-02	-6.65444E-04
2.34038	.63816	.25410	2	1.62690	-9.44987E-03	-2.94702E-04
3.51058	.92474	.23574	3	2.47669	-9.91233E-03	-5.21890E-04
4.68077	1.18997	.21761	4	3.35034	-9.07424E-03	-4.30348E-04
5.85096	1.43410	.19969	5	4.24759	-8.58021E-03	-4.54834E-04
7.02115	1.65735	.18192	6	5.16818	-8.02313E-03	-4.44529E-04

(a) Page 1.

Figure 12.- Trajectory program output for sample case.

8.19135	1.85993	.16434	7	6.11188	-7.49093E-03	-4.43262E-04
9.36154	2.04202	.14690	8	7.07849	-6.95794E-03	-4.38422E-04
10.53173	2.20379	.12959	9	8.06782	-6.42446E-03	-4.32927E-04
11.70192	2.34537	.11240	10	9.07972	-5.88518E-03	-4.25552E-04
12.87212	2.46689	.09531	11	10.11405	-5.33609E-03	-4.16071E-04
14.04231	2.56845	.07829	12	11.17069	-4.77280E-03	-4.03915E-04
15.21250	2.65015	.06135	13	12.24954	-4.19089E-03	-3.88484E-04
16.38269	2.71205	.04445	14	13.35053	-3.58568E-03	-3.69048E-04
17.55288	2.75420	.02760	15	14.47359	-2.95229E-03	-3.44739E-04
18.72308	2.77665	.01077	16	15.61869	-2.28550E-03	-3.14521E-04
19.89327	2.78085	0.00000	17	16.78419	1.93781E-02	-7.97311E-05
21.06346	2.78085	0.00000	18	17.95438	4.95587E-03	3.65240E-04
22.23365	2.78085	0.00000	19	19.12457	3.24829E-03	3.83268E-04
23.40385	2.78085	0.00000	20	20.29477	1.92565E-03	3.87240E-04
24.57404	2.78085	0.00000	21	21.46496	1.31540E-03	3.42929E-04
25.74423	2.78085	0.00000	22	22.63515	9.09252E-04	2.88421E-04
26.91442	2.78085	0.00000	23	23.80534	6.52339E-04	2.30360E-04
28.08462	2.78085	0.00000	24	24.97554	4.77028E-04	1.76141E-04
29.25481	2.78085	0.00000	25	26.14573	3.55373E-04	1.28657E-04
30.42500	2.78085	0.00000	26	27.31592	2.68651E-04	8.92176E-05
31.59519	2.78085	0.00000	27	28.48611	2.05736E-04	5.78498E-05
32.76533	2.78085	0.00000	28	29.65631	1.59347E-04	3.39141E-05
33.93558	2.78085	0.00000	29	30.82650	1.24676E-04	1.64167E-05
35.10577	2.78085	0.00000	30	31.99669	9.84494E-05	4.24052E-06
36.27596	2.78085	0.00000	31	33.16688	7.83970E-05	-3.71543E-06
37.44615	2.78085	0.00000	32	34.33707	6.29162E-05	-8.45400E-06
38.61635	2.78085	0.00000	33	35.50727	5.08585E-05	-1.08392E-05
39.78654	2.78085	0.00000	34	36.67746	4.13905E-05	-1.15828E-05
40.95673	2.78085	0.00000	35	37.84765	3.38997E-05	-1.12485E-05
42.12692	2.78085	0.00000	36	39.01784	2.79314E-05	-1.02650E-05
43.29712	2.78085	0.00000	37	40.18804	2.31447E-05	-8.94837E-06
44.46731	2.78085	0.00000	38	41.35823	1.92818E-05	-7.51646E-06
45.63750	2.78085	0.00000	39	42.52842	-4.40215E-05	-3.51644E-05
46.80769	2.77659	-.00804	40	43.70337	-9.81787E-03	-2.51974E-04
47.97788	2.76147	-.01781	41	44.89047	-1.81966E-03	-3.16028E-04
49.14808	2.73492	-.02758	42	46.09035	-1.71653E-03	-2.93819E-04
50.31827	2.69692	-.03736	43	47.30302	-6.57163E-04	-2.51972E-04
51.48846	2.64748	-.04715	44	48.52849	-7.16925E-05	-1.94202E-04
52.65865	2.58657	-.05695	45	49.76678	4.65034E-04	-1.29312E-04
53.82885	2.51417	-.06678	46	51.01791	9.33168E-04	-6.04166E-05
54.99904	2.43028	-.07662	47	52.28190	1.37706E-03	1.01154E-05
56.16923	2.33486	-.08648	48	53.55878	1.80696E-03	8.08870E-05
57.33942	2.22788	-.09637	49	54.84858	2.23283E-03	1.50817E-04
58.50962	2.10931	-.10628	50	56.15133	2.66003E-03	2.18912E-04
59.67981	1.97912	-.11623	51	57.46708	3.09192E-03	2.84006E-04
60.85000	1.83728	-.12621	52	58.79586	0.	0.

(b) Page 2.

Figure 12.- Continued.

WING INPUT DATA

LOCATION OF WING ROOT CHORD LEADING EDGE RELATIVE TO FUSELAGE NOSE
XF = -25.62000 FEET
ZF = 0.00000 FEET
LENGTH OF WING ROOT CHORD
CR = 13.91000 FEET
WING SEMISPAN
SSPAN = 20.00000 FEET

96 U-VELOCITY PANELS ARE TO BE LAID OUT ON EACH WING PANEL
8 CHORDWISE ROWS WITH 12 IN EACH ROW

SPANWISE LOCATIONS OF PANEL SIDE EDGES AND SWEEP ANGLES OF WING SECTION TO THE RIGHT

I	SPANWISE LOCATION FEET	LE SWEEP DEGREES	TE SWEEP DEGREES
1	-2.78000	0.00000	0.00000
2	-4.72500	48.60000	30.75000
3	-6.66700	48.60000	30.75000
4	-8.88800	48.60000	30.75000
5	-11.11200	48.60000	30.75000
6	-13.33300	48.60000	30.75000
7	-15.55500	48.60000	30.75000
8	-17.77800	48.60000	30.75000
9	-20.00000	48.60000	30.75000

160 THICKNESS PANELS ARE TO BE LAID OUT ON EACH WING PANEL
8 CHORDWISE ROWS WITH 20 IN EACH ROW
THE CHORDWISE ROWS COINCIDE WITH THOSE USED FOR THE U-VELOCITY PANELS

(c) Page 3.

Figure 12.- Continued.

```
PYLON INPUT DATA

    LEADING EDGE OF PYLON ROOT CHORD IS AT X=  -2,51000 FEET (MEASURED FROM LOCAL WING LEADING EDGE)
    SPANWISE LOCATION IS Y=  -6,66700 FEET
    ROOTCHORD =   4,03000 FEET
    HEIGHT =   1,65000 FEET
    LEADING EDGE SWEEP ANGLE =   0,00000 DEGREES
    TRAILING EDGE SWEEP ANGLE =   0,00000 DEGREES

      8 U-VELOCITY PANELS ARE TO BE LAID OUT ON THE PYLON
          2 CHORDWISE ROWS WITH  4 IN EACH ROW

    SPANWISE LOCATIONS OF PANEL SIDE EDGES
          K          Z
                   FEET
          1        0,00000
          2         ,82500
          3        1,65000


     18 THICKNESS PANELS ARE TO BE LAID OUT ON THE PYLON
          2 CHORDWISE ROWS WITH  9 IN EACH ROW
         THE CHORDWISE ROWS COINCIDE WITH THOSE USED
         FOR THE U-VELOCITY PANELS
```

(d) Page 4.

Figure 12.- Continued.

INPUT VALUES OF THE LOCAL SURFACE SLOPE OF THE THICKNESS
DISTRIBUTION, FOR EACH CHORDWISE ROW THE FIRST VALUE
IS FOR THE PANEL NEAREST THE LEADING EDGE

WING THICKNESS DATA

ROW	SLOPES							
1	.20900	.10900	.07700	.05700	.04300	.02900	.01700	.00600
	-.00500	-.01400	-.02400	-.03400	-.04400	-.05300	-.06100	-.06600
	-.06700	-.06700	-.06700	-.06700				
2	.20900	.10900	.07700	.05700	.04300	.02900	.01700	.00600
	-.00500	-.01400	-.02400	-.03400	-.04400	-.05300	-.06100	-.06600
	-.06700	-.06700	-.06700	-.06700				
3	.20900	.10900	.07700	.05700	.04300	.02900	.01700	.00600
	-.00500	-.01400	-.02400	-.03400	-.04400	-.05300	-.06100	-.06600
	-.06700	-.06700	-.06700	-.06700				
4	.20900	.10900	.07700	.05700	.04300	.02900	.01700	.00600
	-.00500	-.01400	-.02400	-.03400	-.04400	-.05300	-.06100	-.06600
	-.06700	-.06700	-.06700	-.06700				
5	.20900	.10900	.07700	.05700	.04300	.02900	.01700	.00600
	-.00500	-.01400	-.02400	-.03400	-.04400	-.05300	-.06100	-.06600
	-.06700	-.06700	-.06700	-.06700				
6	.20900	.10900	.07700	.05700	.04300	.02900	.01700	.00600
	-.00500	-.01400	-.02400	-.03400	-.04400	-.05300	-.06100	-.06600
	-.06700	-.06700	-.06700	-.06700				
7	.20900	.10900	.07700	.05700	.04300	.02900	.01700	.00600
	-.00500	-.01400	-.02400	-.03400	-.04400	-.05300	-.06100	-.06600
	-.06700	-.06700	-.06700	-.06700				
8	.20900	.10900	.07700	.05700	.04300	.02900	.01700	.00600
	-.00500	-.01400	-.02400	-.03400	-.04400	-.05300	-.06100	-.06600
	-.06700	-.06700	-.06700	-.06700				

PYLON THICKNESS DATA

ROW	SLOPES							
1	.09629	.09629	.09629	.09629	0.00000	-.09629	-.09629	-.09629
	-.09629							
2	.09629	.09629	.09629	.09629	0.00000	-.09629	-.09629	-.09629
	-.09629							

(e) Page 5.

Figure 12.- Continued.

STORE INPUT DATA

STORE NO	SHAPE NO	LENGTH FT	MAXIMUM RADIUS FT	STORE LOCATION RELATIVE TO LOCAL WING CHORD LEADING EDGE X, FT	Y, FT	Z, FT	INCIDENCE ANGLE DEG
10	2	10,62500	,62500	,59000	-6,66700	2,90000	0,00000

POLYNOMIALS SPECIFYING STORE SHAPE NO 2

X/L OF END OF EACH SECTION

SECTION	X/L
1	,23530
2	1,00000

COEFFICIENTS OF POLYNOMIALS DESCRIBING EACH SECTION

SECTION	C1	C2	C3	C4	C5	C6	C7
1	-,44130	-1,00000	,47060	,19470	0,00000	0,00000	1,00000
2	,05882	0,00000	0,00000	0,00000	0,00000	0,00000	0,00000

STORE SHAPE 2 AS CALCULATED FROM THE INPUT POLYNOMIALS, ORIGINS OF SOURCES AND DOUBLETS REPRESENTING THE STORE, AND VALUES OF THE SOURCE AND DOUBLET CONSTANTS

BODY SHAPE X, FT	R, FT	DR/DX	N	SINGULARITY ORIGIN X(N), FT	SOURCE CONSTANTS K(N)	DOUBLET CONSTANTS KD(N)
0,00000	-,00655	,53326	0	,00061	2,22348E-01	1,12251E-02
,35417	,17177	,44149	1	,16212	-9,38550E-02	-2,77962E-03
,70833	,31317	,35820	2	,35820	-5,61048E-02	-1,85146E-03
1,06250	,42622	,28105	3	,58597	-4,92991E-02	-2,08928E-03
1,41667	,51276	,20828	4	,84338	-3,96957E-02	-2,06210E-03
1,77083	,57411	,13856	5	1,12896	-3,26082E-02	-2,16494E-03
2,12500	,61114	,07077	6	1,44173	-2,55567E-02	-2,26491E-03
2,47917	,62435	,00393	7	1,78112	1,90142E-02	2,26161E-03
2,83333	,62496	0,00000	8	2,13460	4,02755E-02	-7,95620E-05
3,18750	,62496	0,00000	9	2,48877	3,22692E-03	8,21709E-04
3,54167	,62496	0,00000	10	2,84294	5,33093E-03	8,92266E-04
3,89583	,62496	0,00000	11	3,19710	2,18199E-03	8,33331E-04
4,25000	,62496	0,00000	12	3,55127	1,55488E-03	6,69918E-04
4,60417	,62496	0,00000	13	3,90544	9,46609E-04	4,97912E-04
4,95833	,62496	0,00000	14	4,25960	6,39481E-04	3,41629E-04
5,31250	,62496	0,00000	15	4,61377	4,32584E-04	2,16013E-04
5,66667	,62496	0,00000	16	4,96794	3,01772E-04	1,22525E-04
6,02083	,62496	0,00000	17	5,32210	2,13968E-04	5,79447E-05
6,37500	,62496	0,00000	18	5,67627	1,54426E-04	1,66581E-05
6,72917	,62496	0,00000	19	6,03044	1,13107E-04	-7,23356E-06
7,08333	,62496	0,00000	20	6,38460	8,39857E-05	-1,90303E-05
7,43750	,62496	0,00000	21	6,73877	6,31417E-05	-2,30202E-05
7,79167	,62496	0,00000	22	7,09294	4,80195E-05	-2,24196E-05
8,14583	,62496	0,00000	23	7,44710	3,69102E-05	-1,94792E-05
8,50000	,62496	0,00000	24	7,80127	2,86546E-05	-1,56647E-05
8,85417	,62496	0,00000	25	8,15544	2,24537E-05	-1,18494E-05
9,20833	,62496	0,00000	26	8,50960	1,77495E-05	-8,48987E-06
9,56250	,62496	0,00000	27	8,86377	1,41472E-05	-5,76904E-06
9,91667	,62496	0,00000	28	9,21794	1,13643E-05	-3,70488E-06
10,27083	,62496	0,00000	29	9,57210	9,19641E-06	-2,22647E-06
10,62500	,62496	0,00000	30	9,92627	0,	0,

(f) Page 6.

Figure 12.- Continued.

U-VELOCITY PANEL CONTROL POINT COORDINATES,
INTERFERENCE VELOCITIES INDUCED AT THESE POINTS,
INPUT TWIST AND CAMBER SLOPE AT THESE POINTS,
AND PANEL SINGULARITY STRENGTHS
(WING COORDINATE SYSTEM)

WING CONTROL POINTS

ROW	PANEL	X, FT	Y, FT	Z, FT	U/VINF	V/VINF	W/VINF	ALPHAL	UPLUS/VINF
1	1	2.17664	3.76520	0.00000	-.01284	.00158	-.02072	0.00000	.05651
1	2	3.29152	3.76520	0.00000	-.01135	.00132	-.02047	0.00000	.03697
1	3	4.40641	3.76520	0.00000	-.01010	.00111	-.02025	0.00000	.02817
1	4	5.52130	3.76520	0.00000	-.00903	.00094	-.02005	0.00000	.02556
1	5	6.63618	3.76520	0.00000	-.00812	.00081	-.01987	0.00000	.02348
1	6	7.75107	3.76520	0.00000	-.00733	.00070	-.01973	0.00000	.03410
1	7	8.86596	3.76520	0.00000	.04365	.04206	-.06140	0.00000	.08088
1	8	9.98084	3.76520	0.00000	-.01468	-.00309	-.01730	0.00000	.03028
1	9	11.09573	3.76520	0.00000	.00546	.02147	-.02220	0.00000	.00596
1	10	12.21062	3.76520	0.00000	-.04099	-.03045	-.01615	0.00000	.00153
1	11	13.32550	3.76520	0.00000	-.04321	-.04031	-.00670	0.00000	-.02611
1	12	14.44039	3.76520	0.00000	-.02438	-.02519	-.01185	0.00000	-.00395
2	1	4.29925	5.70974	0.00000	-.01080	.00296	-.00932	0.00000	.07027
2	2	5.32674	5.70974	0.00000	-.00968	.00253	-.00917	0.00000	.04184
2	3	6.35423	5.70974	0.00000	-.00873	.00219	-.00904	0.00000	.03181
2	4	7.38172	5.70974	0.00000	.05374	.02378	-.07532	0.00000	.10056
2	5	8.40921	5.70974	0.00000	.05184	.05724	-.08938	0.00000	.05342
2	6	9.43670	5.70974	0.00000	.00052	.03486	-.02213	0.00000	.00449
2	7	10.46419	5.70974	0.00000	-.08465	-.06275	.00704	0.00000	-.00610
2	8	11.49168	5.70974	0.00000	-.06632	-.05340	.03927	0.00000	-.03204
2	9	12.51917	5.70974	0.00000	.00991	.01033	.00694	0.00000	.02209
2	10	13.54666	5.70974	0.00000	.01132	.00880	-.01656	0.00000	.03752
2	11	14.57415	5.70974	0.00000	-.00122	.00158	-.00841	0.00000	.01742
2	12	15.60164	5.70974	0.00000	-.00311	.00044	-.00751	0.00000	.00381
3	1	6.57796	7.79727	0.00000	-.00937	.00360	-.00518	0.00000	.08320
3	2	7.51162	7.79727	0.00000	.04911	-.02076	-.06659	0.00000	.11676
3	3	8.44529	7.79727	0.00000	.05356	-.05510	-.05101	0.00000	.05754
3	4	9.37895	7.79727	0.00000	-.00213	-.02866	-.01139	0.00000	.03229
3	5	10.31262	7.79727	0.00000	-.04191	.02016	-.00363	0.00000	.02504
3	6	11.24629	7.79727	0.00000	-.07458	.06198	.03640	0.00000	-.02268
3	7	12.17995	7.79727	0.00000	-.04007	.04235	.02266	0.00000	-.00103
3	8	13.11362	7.79727	0.00000	.01519	-.01105	-.00824	0.00000	.03474
3	9	14.04728	7.79727	0.00000	.00245	-.00189	-.00671	0.00000	.02761
3	10	14.98095	7.79727	0.00000	-.00231	.00096	-.00425	0.00000	.02850
3	11	15.91461	7.79727	0.00000	-.00346	.00161	-.00372	0.00000	.02187
3	12	16.84828	7.79727	0.00000	-.00377	.00179	-.00356	0.00000	.01723
4	1	9.00664	10.02220	0.00000	.04135	-.03882	-.04032	0.00000	.14096
4	2	9.84031	10.02220	0.00000	.01440	-.01830	-.02289	0.00000	.08569
4	3	10.67397	10.02220	0.00000	.01937	-.03358	.00400	0.00000	.02666
4	4	11.50764	10.02220	0.00000	.00487	-.01718	-.00528	0.00000	.04312
4	5	12.34131	10.02220	0.00000	.00151	-.01305	-.00562	0.00000	.03729
4	6	13.17497	10.02220	0.00000	-.07239	.07041	.00804	0.00000	.00551
4	7	14.00864	10.02220	0.00000	-.03557	.03522	.00574	0.00000	.00399
4	8	14.84230	10.02220	0.00000	-.02445	.02607	.00318	0.00000	.01384
4	9	15.67597	10.02220	0.00000	.00874	-.01083	-.00647	0.00000	.03331
4	10	16.50964	10.02220	0.00000	-.00129	-.00082	-.00352	0.00000	.02815
4	11	17.34330	10.02220	0.00000	-.00303	.00093	-.00308	0.00000	.02935
4	12	18.17697	10.02220	0.00000	-.00351	.00147	-.00294	0.00000	.02261
5	1	11.43590	12.24764	0.00000	.03424	-.03891	-.02485	0.00000	.15950
5	2	[illegible].16954	12.24764	0.00000	-.00059	-.00520	-.00746	0.00000	.07903

(g) Page 7.

Figure 12.- Continued.

5	3	-12.90318	-12.24764	0.00000	-.02239	.01713	.00416	0.00000	.03338
5	4	-13.63683	-12.24764	0.00000	.00872	-.01702	-.00359	0.00000	.04522
5	5	-14.37047	-12.24764	0.00000	.00201	-.00984	-.00312	0.00000	.04262
5	6	-15.10411	-12.24764	0.00000	.00070	-.00850	-.00321	0.00000	.03111
5	7	-15.83776	-12.24764	0.00000	-.06056	.06152	.00804	0.00000	.00631
5	8	-16.57140	-12.24764	0.00000	-.02651	.02631	.00175	0.00000	.01335
5	9	-17.30504	-12.24764	0.00000	-.01986	.02025	.00076	0.00000	.01973
5	10	-18.03869	-12.24764	0.00000	.00746	-.01035	-.00403	0.00000	.03349
5	11	-18.77233	-12.24764	0.00000	-.00111	-.00122	-.00249	0.00000	.02941
5	12	-19.50597	-12.24764	0.00000	-.00271	.00053	-.00223	0.00000	.03105
6	1	-13.86518	-14.47312	0.00000	.02228	-.02785	-.01377	0.00000	.17099
6	2	-14.49880	-14.47312	0.00000	-.00210	-.00269	-.00448	0.00000	.08626
6	3	-15.13242	-14.47312	0.00000	-.01907	.01553	.00233	0.00000	.04559
6	4	-15.76604	-14.47312	0.00000	.02994	-.03836	-.00472	0.00000	.05113
6	5	-16.39966	-14.47312	0.00000	.00378	-.00972	-.00221	0.00000	.04468
6	6	-17.03328	-14.47312	0.00000	.00121	-.00698	-.00211	0.00000	.04133
6	7	-17.66690	-14.47312	0.00000	-.04265	.04199	.00118	0.00000	.02687
6	8	-18.30052	-14.47312	0.00000	-.05738	.05949	.00602	0.00000	.01185
6	9	-18.93414	-14.47312	0.00000	-.02409	.02392	.00095	0.00000	.01794
6	10	-19.56776	-14.47312	0.00000	-.01837	.01833	.00031	0.00000	.02060
6	11	-20.20138	-14.47312	0.00000	.02768	-.03255	-.00519	0.00000	.03498
6	12	-20.83500	-14.47312	0.00000	.00083	-.00317	-.00203	0.00000	.03249
7	1	-16.29719	-16.70109	0.00000	.01891	-.02420	-.00966	0.00000	.18733
7	2	-16.83067	-16.70109	0.00000	-.00106	-.00298	-.00357	0.00000	.09730
7	3	-17.36416	-16.70109	0.00000	-.01573	.01299	.00109	0.00000	.06073
7	4	-17.89764	-16.70109	0.00000	-.01321	.01064	.00048	0.00000	.05022
7	5	-18.43112	-16.70109	0.00000	.01104	-.01652	-.00223	0.00000	.05078
7	6	-18.96461	-16.70109	0.00000	.00272	-.00742	-.00159	0.00000	.04709
7	7	-19.49809	-16.70109	0.00000	.00108	-.00564	-.00156	0.00000	.04348
7	8	-20.03157	-16.70109	0.00000	.00038	-.00490	-.00156	0.00000	.03520
7	9	-20.56506	-16.70109	0.00000	-.01653	.01456	.00004	0.00000	.02379
7	10	-21.09854	-16.70109	0.00000	-.02667	.02686	.00102	0.00000	.01866
7	11	-21.63202	-16.70109	0.00000	-.01923	.01914	.00034	0.00000	.01806
7	12	-22.16551	-16.70109	0.00000	-.01583	.01577	.00004	0.00000	.02131
8	1	-18.73185	-18.93150	0.00000	.01694	-.02179	-.00731	0.00000	.20909
8	2	-19.16509	-18.93150	0.00000	.00113	-.00474	-.00329	0.00000	.11362
8	3	-19.59833	-18.93150	0.00000	-.01255	.01021	.00026	0.00000	.07776
8	4	-20.03157	-18.93150	0.00000	-.01708	.01538	.00153	0.00000	.05824
8	5	-20.46480	-18.93150	0.00000	-.01139	.00938	.00016	0.00000	.05313
8	6	-20.89804	-18.93150	0.00000	.01067	-.01538	-.00176	0.00000	.04979
8	7	-21.33128	-18.93150	0.00000	.00315	-.00711	-.00127	0.00000	.03418
8	8	-21.76452	-18.93150	0.00000	.00133	-.00512	-.00120	0.00000	.02329
8	9	-22.19776	-18.93150	0.00000	.00061	-.00434	-.00120	0.00000	.01420
8	10	-22.63099	-18.93150	0.00000	-.04285	.04423	.00085	0.00000	.00195
8	11	-23.06423	-18.93150	0.00000	-.01475	.01325	-.00005	0.00000	-.00511
8	12	-23.49747	-18.93150	0.00000	-.02634	.02684	.00087	0.00000	-.01105

PYLON CONTROL POINTS

ROW	PANEL	X, FT	Y, FT	Z, FT	V/VINF	UPLUB/VINF
1	1	-7.97106	-6.66700	.41250	.03493	-.01072
1	2	-9.07856	-6.66700	.41250	.04513	-.00531
1	3	-10.18606	-6.66700	.41250	.03422	-.00891
1	4	-11.29356	-6.66700	.41250	.02849	-.00083
2	1	-7.97106	-6.66700	1.23750	.01438	-.00114
2	2	-9.07856	-6.66700	1.23750	.02634	-.00699
2	3	-10.18606	-6.66700	1.23750	.03511	-.02524
2	4	-11.29356	-6.66700	1.23750	.01508	-.00482

(h) Page 8.

Figure 12.- Continued.

FUSELAGE CONTROL POINTS

RING	PANEL	X, FT	Y, FT	Z, FT	V/VINF	W/VINF	UPLUS/VINF
1	1	-1.10121	-2.67419	-.53193	.02958	-.05500	-.00894
1	2	-1.10121	-2.26707	-1.51481	0.00000	0.00000	-.00124
1	3	-1.10121	-1.51481	-2.26707	0.00000	0.00000	-.00017
1	4	-1.10121	-.53193	-2.67419	0.00000	0.00000	-.00003
1	5	-1.10121	-.53193	2.67419	0.00000	0.00000	-.00006
1	6	-1.10121	-1.51481	2.26707	0.00000	0.00000	-.00038
1	7	-1.10121	-2.26707	1.51481	0.00000	0.00000	-.00277
1	8	-1.10121	-2.67419	.53193	.02958	.05500	-.02001
2	1	-2.26038	-2.67419	-.53193	.04431	-.03515	-.00527
2	2	-2.26038	-2.26707	-1.51481	.01472	-.03168	-.00161
2	3	-2.26038	-1.51481	-2.26707	0.00000	0.00000	-.00112
2	4	-2.26038	-.53193	-2.67419	0.00000	0.00000	-.00033
2	5	-2.26038	-.53193	2.67419	0.00000	0.00000	-.00083
2	6	-2.26038	-1.51481	2.26707	0.00000	0.00000	-.00310
2	7	-2.26038	-2.26707	1.51481	.01472	.03168	-.00791
2	8	-2.26038	-2.67419	.53193	.04431	.03515	-.04022
3	1	-3.41954	-2.67419	-.53193	.04244	-.02448	-.00005
3	2	-3.41954	-2.26707	-1.51481	.02218	-.02658	-.00648
3	3	-3.41954	-1.51481	-2.26707	.01412	-.02194	.00157
3	4	-3.41954	-.53193	-2.67419	0.00000	0.00000	-.00131
3	5	-3.41954	-.53193	2.67419	0.00000	0.00000	-.00433
3	6	-3.41954	-1.51481	2.26707	.01412	.02194	-.00201
3	7	-3.41954	-2.26707	1.51481	.02218	.02658	-.02600
3	8	-3.41954	-2.67419	.53193	.04244	.02448	-.02748
4	1	-4.57871	-2.67419	-.53193	.03647	-.01562	.00478
4	2	-4.57871	-2.26707	-1.51481	.02361	-.02089	-.00220
4	3	-4.57871	-1.51481	-2.26707	.01722	-.01914	-.00574
4	4	-4.57871	-.53193	-2.67419	.01596	-.01721	.00314
4	5	-4.57871	-.53193	2.67419	.01596	.01721	-.00156
4	6	-4.57871	-1.51481	2.26707	.01722	.01914	-.01835
4	7	-4.57871	-2.26707	1.51481	.02361	.02089	-.02127
4	8	-4.57871	-2.67419	.53193	.03647	.01562	-.02074
5	1	-5.73788	-2.67419	-.53193	.02852	-.00709	.00842
5	2	-5.73788	-2.26707	-1.51481	.02204	-.01463	-.00142
5	3	-5.73788	-1.51481	-2.26707	.00723	-.02105	.00029
5	4	-5.73788	-.53193	-2.67419	.00133	-.02408	-.00032
5	5	-5.73788	-.53193	2.67419	.00133	.02408	-.00760
5	6	-5.73788	-1.51481	2.26707	.00723	.02105	-.01273
5	7	-5.73788	-2.26707	1.51481	.02204	.01463	-.02384
5	8	-5.73788	-2.67419	.53193	.02852	.00709	-.01582
6	1	-6.89704	-2.67419	-.53193	.00542	-.00148	.01893
6	2	-6.89704	-2.26707	-1.51481	.00469	-.01243	.01465
6	3	-6.89704	-1.51481	-2.26707	.00211	-.01852	-.00126
6	4	-6.89704	-.53193	-2.67419	.00043	-.02063	-.00293
6	5	-6.89704	-.53193	2.67419	.00043	.02063	-.01429
6	6	-6.89704	-1.51481	2.26707	.00211	.01852	-.01641
6	7	-6.89704	-2.26707	1.51481	.00469	.01243	-.00661
6	8	-6.89704	-2.67419	.53193	.00542	.00148	-.00618
7	1	-8.05621	-2.67419	-.53193	-.00463	.00529	.03168
7	2	-8.05621	-2.26707	-1.51481	.00041	-.00678	.00809
7	3	-8.05621	-1.51481	-2.26707	.00010	-.01334	.00935
7	4	-8.05621	-.53193	-2.67419	.00012	-.01669	-.00339
7	5	-8.05621	-.53193	2.67419	.00012	.01669	-.01929
7	6	-8.05621	-1.51481	2.26707	.00010	.01334	-.01080
7	7	-8.05621	-2.26707	1.51481	.00041	.00678	-.00957
7	8	-8.05621	-2.67419	.53193	-.00463	-.00529	-.01211

(i) Page 9.

Figure 12.- Continued.

8	1	-9.21538	-2.67419	-.53193	-.01454	.01313	.02999
8	2	-9.21538	-2.26707	-1.51481	-.00401	-.00055	.02386
8	3	-9.21538	-1.51481	-2.26707	-.00166	-.00805	.00429
8	4	-9.21538	-.53193	-2.67419	-.00047	-.01154	.00444
8	5	-9.21538	-.53193	2.67419	-.00047	.01154	-.01714
8	6	-9.21538	-1.51481	2.26707	-.00166	.00805	-.02161
8	7	-9.21538	-2.26707	1.51481	.05046	-.01672	-.05261
8	8	-9.21538	-2.67419	.53193	.03283	-.04133	-.05874
9	1	-10.37454	-2.67419	-.53193	.00926	-.00973	-.00380
9	2	-10.37454	-2.26707	-1.51481	-.00835	.00625	.01469
9	3	-10.37454	-1.51481	-2.26707	-.00359	-.00268	.01442
9	4	-10.37454	-.53193	-2.67419	-.00118	-.00640	.00970
9	5	-10.37454	-.53193	2.67419	-.00118	.00640	-.02453
9	6	-10.37454	-1.51481	2.26707	.04073	-.00324	-.04687
9	7	-10.37454	-2.26707	1.51481	.00087	-.01021	-.01597
9	8	-10.37454	-2.67419	.53193	-.01569	-.02563	.00552
10	1	-11.53371	-2.67419	-.53193	-.01200	.02536	.01296
10	2	-11.53371	-2.26707	-1.51481	.01403	-.01505	.00663
10	3	-11.53371	-1.51481	-2.26707	-.00593	.00334	.00916
10	4	-11.53371	-.53193	-2.67419	-.00161	-.00143	.01457
10	5	-11.53371	-.53193	2.67419	.03581	-.00036	-.04071
10	6	-11.53371	-1.51481	2.26707	-.01891	-.00281	-.00434
10	7	-11.53371	-2.26707	1.51481	-.02138	-.01075	.00128
10	8	-11.53371	-2.67419	.53193	.02340	-.02780	-.02240
11	1	-12.69288	-2.67419	-.53193	-.02407	.02309	.00570
11	2	-12.69288	-2.26707	-1.51481	.00099	.01225	.00630
11	3	-12.69288	-1.51481	-2.26707	.01708	-.01632	.01241
11	4	-12.69288	-.53193	-2.67419	-.00241	.00449	-.00482
11	5	-12.69288	-.53193	2.67419	-.01652	-.00480	-.01936
11	6	-12.69288	-1.51481	2.26707	-.01423	-.00935	-.02583
11	7	-12.69288	-2.26707	1.51481	.00099	-.01391	-.01782
11	8	-12.69288	-2.67419	.53193	-.02294	-.02653	.04015
12	1	-13.85204	-2.67419	-.53193	-.06787	.03751	.05075
12	2	-13.85204	-2.26707	-1.51481	-.00846	.01632	.00917
12	3	-13.85204	-1.51481	-2.26707	.00490	.00868	-.01379
12	4	-13.85204	-.53193	-2.67419	.02083	-.01177	-.00396
12	5	-13.85204	-.53193	2.67419	-.00884	-.01076	-.01615
12	6	-13.85204	-1.51481	2.26707	.00505	-.00963	-.01508
12	7	-13.85204	-2.26707	1.51481	-.06118	-.02249	.03694
12	8	-13.85204	-2.67419	.53193	-.15399	-.02560	.10017

(j) Page 10.

Figure 12.- Continued.

```
STORE NUMBER   10 IS THE STORE EJECTED

     ADDITIONAL INPUT FOR THIS STORE
          STORE MASS =  15,53000 SLUGS
          MOMENTS AND PRODUCTS OF INERTIA, SLUG - SQ FT
             IXX =  16,00000
             IYY = 400,00000
             IZZ = 400,00000
             IYZ =   0,00000
             IXZ =   0,00000
             IXY =   0,00000
          STORE MOMENT CENTER IS -7,40000 FEET BEHIND NOSE
          STORE CENTER OF GRAVITY OFFSET FROM MOMENT CENTER, FEET
             XBAR =  0,00000
             YBAR =  0,00000
             ZBAR =  0,00000
          POLYNOMIALS SPECIFYING STORE SHAPE
             X/L OF END OF EACH SECTION
                SECTION      X/L
                   1        ,23530
                   2       1,00000
             COEFFICIENTS OF POLYNOMIALS DESCRIBING EACH SECTION
                SECTION      C1          C2          C3          C4          C5          C6          C7
                   1       -,44130   -1,00000     ,47060     ,19470    0,00000    0,00000    1,00000
                   2        ,05882    0,00000    0,00000    0,00000    0,00000    0,00000    0,00000
          SEPARATION ASSUMED  10,62500 FEET FROM NOSE
          AXIAL-FORCE COEFFICIENT IS     ,65000
          CROSSFLOW-DRAG COEFFICIENT IS   0,00000

          THIS STORE HAS A CRUCIFORM EMPENNAGE
          THE EMPENNAGE FORCES ACT -9,55800 FEET BEHIND NOSE
          THE AVERAGE BODY RADIUS IN THE EMPENNAGE REGION IS    ,62500 FEET
          THE TAIL FIN SEMISPAN MEASURED FROM THE BODY AXIS IS  1,45833 FEET
          THE FINS ARE INITIALLY ROLLED 45,00 DEGREES FROM THE VERTICAL AND HORIZONTAL
          THE FIN LIFT-CURVE SLOPE IS  3,49100 PER RADIAN

          POLYNOMIALS SPECIFYING STORE THRUST TIME HISTORY
             TIME AT END OF EACH POLYNOMIAL
                POLYNOMIAL   TIME, SEC
                   1          ,20000
                   2         1,00000
             COEFFICIENTS OF POLYNOMIALS
                POLYNOMIAL      C1          C2          C3          C4          C5          C6
                   1          1500,000       0,000       0,000       0,000       0,000       0,000
                   2          1875,000   -1875,000       0,000       0,000       0,000       0,000
```

(k) Page 11.

Figure 12.- Continued.

TIME = 0.00000 SECONDS

FORCE AND MOMENT COEFFICIENTS

	CN	CY	CLM	CLN	CLL
BUOYANCY	.07250	.11988	-.25975	.31491	
SLENDER BODY	.09273	.04426	-.13025	-.06018	
CROSSFLOW	0.00000	0.00000	0.00000	0.00000	
EMPENNAGE	.43352	.05682	-.74843	-.09809	-.02310
TOTAL	.59875	.21696	-1.13843	.15664	-.02310

STORE THRUST IS 1500.000 POUNDS

LOAD AND VELOCITY DISTRIBUTIONS

X, FT	X/L	DCN/DX	DCY/DX	U/VS	V/VS	W/VS
.26563	.02500	.02934	-.00100	1.01139	.00043	.04791
.79688	.07500	-.08973	-.20354	1.00346	-.00601	.04412
1.32813	.12500	-.05269	-.00477	.90980	-.07967	-.01963
1.85938	.17500	-.04784	.05886	.91896	-.06260	-.02507
2.39063	.22500	-.02361	.09412	.92761	-.04802	-.02861
2.92188	.27500	-.00733	.08440	.93579	-.03573	-.03041
3.45313	.32500	.00419	.06923	.94349	-.02555	-.03060
3.98438	.37500	.01461	.05546	.95075	-.01729	-.02933
4.51563	.42500	-.12069	.04333	.95325	-.01077	-.03168
5.04688	.47500	-.06162	.03293	.94304	-.00572	-.04936
5.57813	.52500	-.02121	.02625	.94370	-.00193	-.05425
6.10938	.57500	.03181	.02311	.94639	.00141	-.05556
6.64063	.62500	.29236	.01672	.99423	.00401	-.02652
7.17188	.67500	.33799	.01839	1.00891	.00632	.01543
7.70313	.72500	.19874	.01377	1.03063	.00867	.04679
8.23438	.77500	.08349	-.00207	1.04457	.00936	.07029
8.76563	.82500	-.10553	-.00693	1.04159	.00861	.07424
9.29688	.87500	-.11120	-.00845	1.01285	.00761	.04623
9.82813	.92500	-.03192	-.00653	1.00501	.00652	.03871
10.35938	.97500	-.00813	-.00183	1.00254	.00597	.03640

LOCATION OF STORE IN FUSELAGE COORDINATE SYSTEM, DIMENSIONS OF FEET

	RELATIVE TO FUSELAGE NOSE			RELATIVE TO INITIAL POSITION		
	XF	YF	ZF	DEL XF	DEL YF	DEL ZF
NOSE	-29.43894	-6.66700	2.90000	.00000	0.00000	0.00000
XMOM	-36.83894	-6.66700	2.90000	0.00000	0.00000	0.00000
BASE	-40.06394	-6.66700	2.90000	0.00000	0.00000	0.00000

TRANSLATIONAL VELOCITIES AND ACCELERATIONS OF STORE IN FUSELAGE COORDINATE SYSTEM RELATIVE TO FUSELAGE MOTION

DXF	DYF	DZF	D2XF	D2YF	D2ZF
0.00000	0.00000	10.00000	60.74562	17.00534	-18.52163

ROTATIONAL VELOCITIES AND ACCELERATIONS OF STORE IN STORE COORDINATE SYSTEM

P	Q	R	PDOT	QDOT	RDOT
0.00000	0.00000	0.00000	-2.19689	-4.33040	.59583

STORE ANGULAR ORIENTATION IN FUSELAGE COORDINATE SYSTEM AND RATES OF CHANGE OF THESE ANGLES ANGLES IN DEGREES, RATES OF CHANGE IN RADIANS PER SECOND

PSI	THETA	PHI	DPSI	DTHETA	DPHI
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

(*l*) Page 12.

Figure 12.- Continued.

TIME = .20000 SECONDS

FORCE AND MOMENT COEFFICIENTS

	CN	CY	CLM	CLN	CLL
BUOYANCY	-.06191	-.00857	-.51928	-.20215	
SLENDER BODY	-.16233	-.04355	-1.02266	-.37721	
CROSSFLOW	0.00000	0.00000	0.00000	0.00000	
EMPENNAGE	-.89438	-.16287	1.54405	.28117	.00219
TOTAL	-1.11862	-.21499	.00211	-.29818	.00219

STORE THRUST IS 1500.000 POUNDS

LOAD AND VELOCITY DISTRIBUTIONS

X, FT	X/L	DCN/DX	DCY/DX	U/VS	V/VS	W/VS
.20563	.02500	-.03056	-.01049	1.01451	-.01656	-.04859
.79688	.07500	-.05998	-.02140	1.01396	-.01673	-.04896
1.32813	.12500	-.05958	-.02213	1.01253	-.01760	-.04984
1.85938	.17500	-.04069	-.01636	1.01194	-.01785	-.05027
2.39063	.22500	-.01062	-.00529	1.01085	-.01850	-.05098
2.92188	.27500	-.00417	-.00321	1.01030	-.01875	-.05141
3.45313	.32500	-.00904	-.00983	1.00941	-.01928	-.05204
3.98438	.37500	-.29197	-.30549	1.00578	-.02156	-.05439
4.51563	.42500	-.06617	.04143	.94574	-.05814	-.09641
5.04688	.47500	-.05053	.03776	.94220	-.05289	-.10414
5.57813	.52500	-.03605	.03426	.94034	-.04810	-.10987
6.10938	.57500	-.02272	.03091	.94001	-.04378	-.11376
6.64063	.62500	-.01056	.02772	.94106	-.03989	-.11595
7.17188	.67500	.00045	.02469	.94334	-.03641	-.11660
7.70313	.72500	.01030	.02182	.94671	-.03332	-.11587
8.23438	.77500	-.08057	.02141	.94296	-.03045	-.12134
8.76563	.82500	-.02687	.01833	.93843	-.02782	-.12798
9.29688	.87500	-.00593	.01771	.93909	-.02546	-.12991
9.82813	.92500	.15530	.01171	.94624	-.02326	-.12575
10.35938	.97500	.21766	.00834	.99966	-.02214	-.09628

LOCATION OF STORE IN FUSELAGE COORDINATE SYSTEM, DIMENSIONS OF FEET

	RELATIVE TO FUSELAGE NOSE			RELATIVE TO INITIAL POSITION		
	XF	YF	ZF	DEL XF	DEL YF	DEL ZF
NOSE	-28.20105	-0.81485	5.81553	1.23780	-.14785	2.91553
XMOM	-35.56068	-0.66748	5.05791	1.27825	-.00048	2.15791
BASE	-38.76810	-0.60325	4.72773	1.29584	.06375	1.82773

TRANSLATIONAL VELOCITIES AND ACCELERATIONS OF STORE IN FUSELAGE COORDINATE SYSTEM
RELATIVE TO FUSELAGE MOTION

DXF	DYF	DZF	D2XF	D2YF	D2ZF
12.98350	-.57574	15.23975	64.02201	-14.87566	110.06979

ROTATIONAL VELOCITIES AND ACCELERATIONS OF STORE IN STORE COORDINATE SYSTEM

P	Q	R	PDOT	QDOT	RDOT
-.04042	-.90265	-.24448	.21163	.01767	-1.18940

STORE ANGULAR ORIENTATION IN FUSELAGE COORDINATE SYSTEM AND RATES OF CHANGE OF THESE ANGLES
ANGLES IN DEGREES, RATES OF CHANGE IN RADIANS PER SECOND

PSI	THETA	PHI	DPSI	DTHETA	DPHI
-1.14721	-5.87628	-.20646	-.24250	-.90353	-.01559

(m) Page 13.

Figure 12.- Continued.

TIME = .40000 SECONDS

FORCE AND MOMENT COEFFICIENTS

	CN	CY	CLM	CLN	CLL
BUOYANCY	.03035	.02446	.11146	.08679	
SLENDER BODY	-.33540	-.12739	-1.72577	-.67645	
CROSSFLOW	0.00000	0.00000	0.00000	0.00000	
EMPENNAGE	-1.15409	-.43945	1.99242	.75176	.00015
TOTAL	-1.45914	-.53838	.37812	.16210	.00015

STORE THRUST IS 1125.000 POUNDS

LOAD AND VELOCITY DISTRIBUTIONS

X, FT	X/L	DCN/DX	DCY/DX	U/VS	V/VS	W/VS
.26563	.02500	-.11373	-.04756	.96013	-.07654	-.18257
.79688	.07500	-.21218	-.08724	.96320	-.07557	-.18123
1.32813	.12500	-.19891	-.08031	.96644	-.07449	-.17976
1.85938	.17500	-.11804	-.04452	.97021	-.07314	-.17797
2.39063	.22500	-.00940	.00110	.97313	-.07212	-.17661
2.92188	.27500	.01176	.00906	.97724	-.07054	-.17455
3.45313	.32500	.01551	.01219	.98092	-.06909	-.17270
3.98438	.37500	.01340	.01064	.98435	-.06771	-.17095
4.51563	.42500	.01471	.01186	.98824	-.06608	-.16892
5.04688	.47500	.01486	.01215	.99147	-.06470	-.16721
5.57813	.52500	.01379	.01144	.99537	-.06296	-.16510
6.10938	.57500	.01521	.01280	.99876	-.06140	-.16324
6.64063	.62500	.01348	.01155	1.00224	-.05975	-.16129
7.17188	.67500	-.01312	-.01076	1.00139	-.06003	-.16166
7.70313	.72500	-.00217	-.00148	1.00006	-.06054	-.16232
8.23438	.77500	-.00599	-.00473	.99869	-.06110	-.16303
8.76563	.82500	-.00364	-.00272	.99803	-.06132	-.16335
9.29688	.87500	-.00348	-.00259	.99677	-.06184	-.16401
9.82813	.92500	-.00367	-.00277	.99622	-.06203	-.16431
10.35938	.97500	-.00262	-.00186	.99517	-.06247	-.16488

LOCATION OF STORE IN FUSELAGE COORDINATE SYSTEM, DIMENSIONS OF FEET

	RELATIVE TO FUSELAGE NOSE			RELATIVE TO INITIAL POSITION		
	XF	YF	ZF	DEL XF	DEL YF	DEL ZF
NOSE	-24.80190	-7.87210	12.59828	4.63703	-1.20510	9.69828
XMOM	-31.98907	-7.38477	10.90499	4.84987	-.71777	8.00499
BASE	-35.12131	-7.17238	10.16903	4.94263	-.50538	7.26703

TRANSLATIONAL VELOCITIES AND ACCELERATIONS OF STORE IN FUSELAGE COORDINATE SYSTEM RELATIVE TO FUSELAGE MOTION

DXF	DYF	DZF	D2XF	D2YF	D2ZF
21.38756	-7.48757	43.36490	22.75717	-40.56573	139.27963

ROTATIONAL VELOCITIES AND ACCELERATIONS OF STORE IN STORE COORDINATE SYSTEM

P	Q	R	PDOT	QDOT	RDOT
-.04648	-.39263	-.15702	.01439	1.48022	.61834

STORE ANGULAR ORIENTATION IN FUSELAGE COORDINATE SYSTEM AND RATES OF CHANGE OF THESE ANGLES ANGLES IN DEGREES, RATES OF CHANGE IN RADIANS PER SECOND

PSI	THETA	PHI	DPSI	DTHETA	DPHI
-3.87905	-13.22779	-.21302	-.15980	-.39321	-.00991

(n) Page 14.

Figure 12.- Concluded.

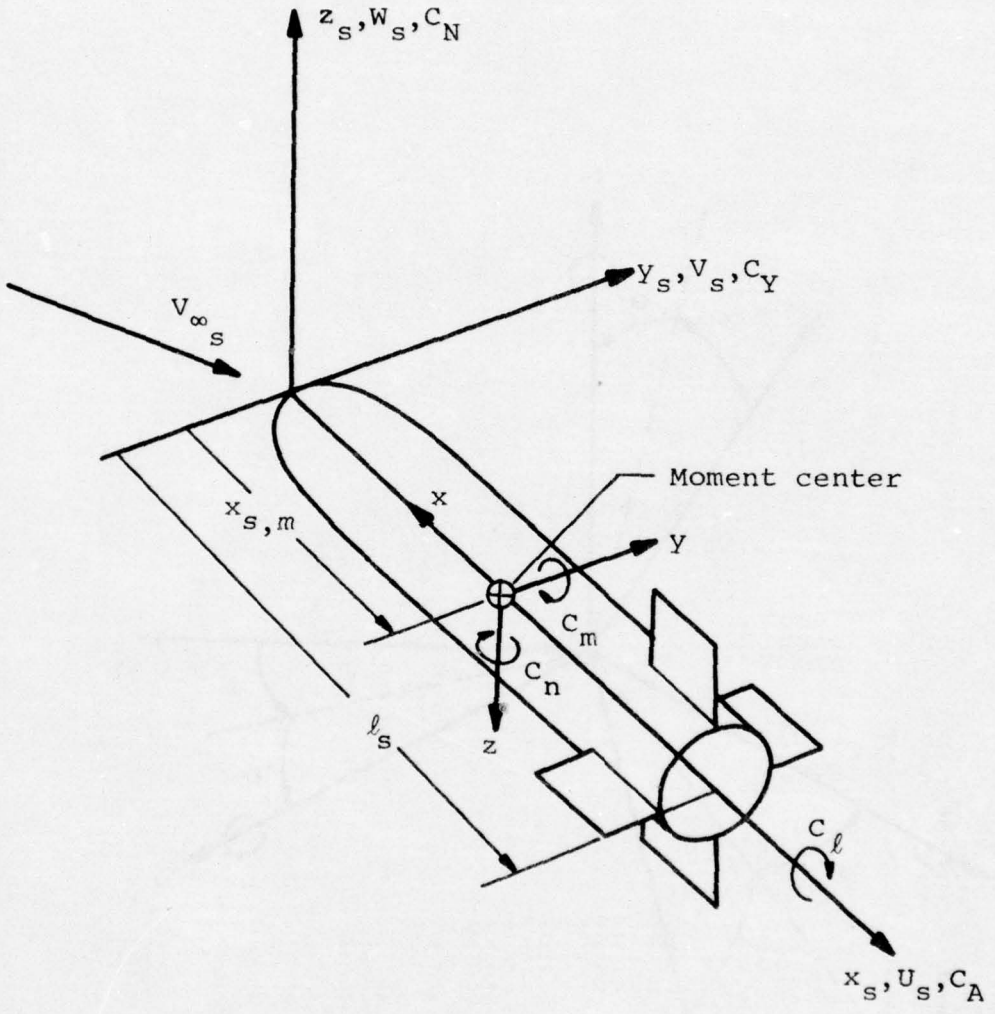


Figure 13.- Coordinate systems fixed in separated store and used in force and moment calculation.

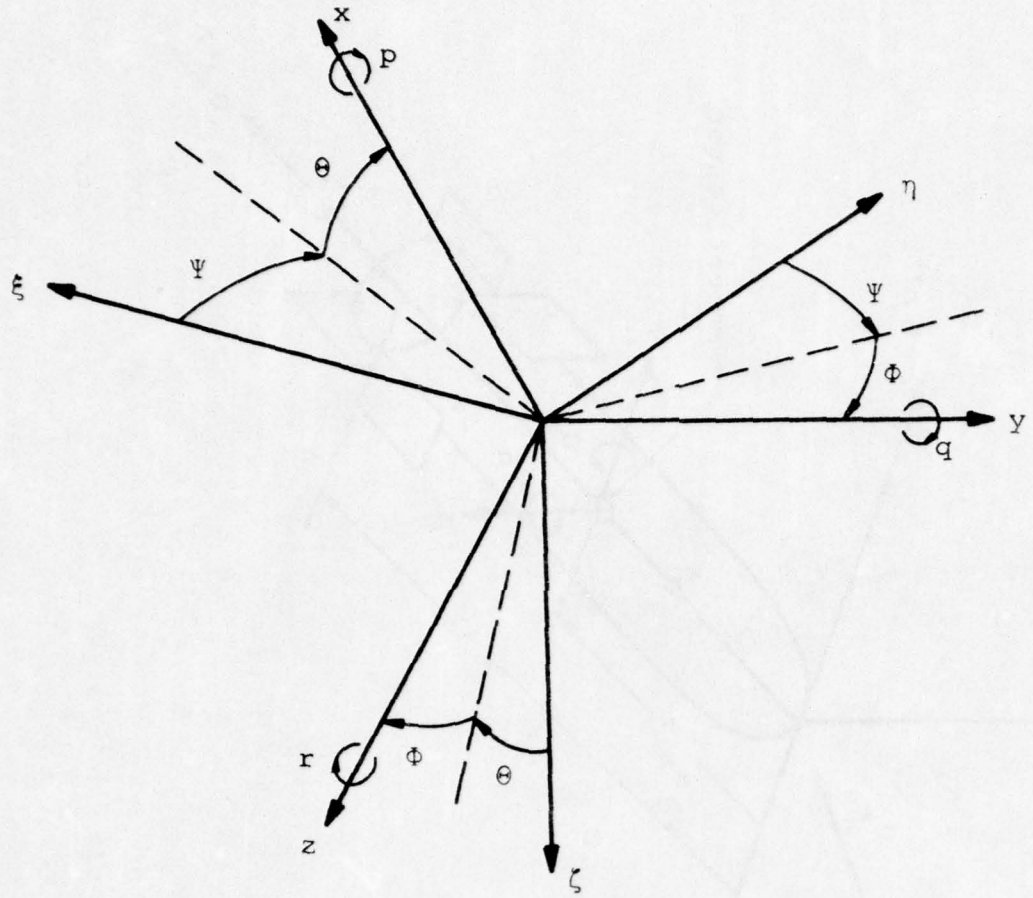


Figure 14.- Coordinate systems used in trajectory calculation.

APPENDIX I

DETAILS OF STORE SEPARATION TRAJECTORY PROGRAM

I-1. INTRODUCTION

The purpose of this appendix is to provide more detailed information on the store separation trajectory program which was described in section 2. A listing of the program was presented in figure 1 and a general flow chart in figure 2. This appendix will present a cross-reference chart showing the subroutines and the common statements contained in each, a schematic description of the subroutine calling sequence, more detailed flow charts, and tables equating the program notation to the algebraic notation. The program consists of a main program and 42 subroutines. The main program will first be described and then the subroutines will be described in alphabetical order. The subroutines and their functions were listed in Table I.

The cross-reference chart showing the subroutines and the common statements contained in each is presented in figure I-1. Across the top of the chart are the subroutine names including the main program SSTORE. Down the side of the page are the common names. The last one is blank common. The schematic showing the subroutine calling sequence is presented in figure I-2.

I-2. MAIN PROGRAM

The first two pages of the flow chart which was presented in figure 2 are quite complete and will therefore not be expanded in this appendix. Using those pages, the description in section 3.1, the input data discussion in section 3.2, and the dictionary of notation in Table I-1, there should be sufficient information to understand the flow of the program.

The last page of the flow chart of figure 2 has been expanded and is presented in figure I-3. This portion of the program begins with FORTRAN statement 62 of the main program (fig. 1(c)), and is the integration loop of the program. The first step in the loop is to call subroutine SFORCE to calculate the aerodynamic forces and moments acting on the store body. If the store has an empennage then subroutine SEMFOR is called to calculate the empennage forces and moments.

The next portion of the program solves for the translational and rotational accelerations. The set of six simultaneous equations which are solved are given in Appendix II of reference 4, equations (II-16) through (II-18) and (II-41) through (II-43). The coefficient matrix and right-hand side are stored in the FVN array. The first subscript of FVN is the equation number. The correspondence is

Subscript Value	Equation Number
1	II-16
2	II-17
3	II-18
4	II-41
5	II-42
6	II-43

The second subscript of FVN is the term in the equation. Here the correspondence is

Subscript Value	Term
1	$\ddot{\xi}$
2	$\ddot{\eta}$
3	$\ddot{\zeta}$
4	$\dot{p}$
5	$\dot{q}$
6	$\dot{r}$
7	Right-Hand Side

Thus, for example, FVN(3,5) is the coefficient of $\dot{q}$ in equation (II-18). Certain parts of this section of the program are bypassed if the store center of gravity, c.g., lies on the store moment center. If this is the case, $\bar{x}$,$\bar{y}$ and $\bar{z}$ are zero and NASYM equals zero.

After calculating the coefficient matrix and right-hand side of the set of equations, subroutine INVERS is called to solve for the accelerations and the values are transferred to the DVAR array. The next steps of the program put the values of $\dot{\xi}$,$\dot{\eta}$, and $\dot{\zeta}$ into the DVAR array and calculate the values of $\dot{\Psi}$,$\dot{\Theta}$, and $\dot{\Phi}$, which are also put in the DVAR array. The DVAR array contains

DVAR(1)	$\ddot{\xi}$
DVAR(2)	$\ddot{\eta}$
DVAR(3)	$\ddot{\zeta}$
DVAR(4)	$\dot{p}$
DVAR(5)	$\dot{q}$
DVAR(6)	$\dot{r}$
DVAR(7)	$\dot{\xi}$
DVAR(8)	$\dot{\eta}$
DVAR(9)	$\dot{\zeta}$
DVAR(10)	$\dot{\psi}$
DVAR(11)	$\dot{\theta}$
DVAR(12)	$\dot{\phi}$

These are the derivatives of the twelve dependent variables.

The program next checks to see if the integration procedure has reached the end of an integration step. If it has NOUT = 1 and subroutine SOUTPT is called to print the output. Next a check is made to see if the end of the trajectory has been reached, that is, is the current value of the time equal to the final time which was input. If it is then the program transfers to the first input statement and attempts to read another set of input data. If the end has not been reached, then the integration routine, subroutine ADAMS, is called.

NDIFEQ is a control index used by subroutine ADAMS. If NDIFEQ = 1 upon returning to the main program,an error condition has been encountered in ADAMS, the calculation is to be terminated, and the next set of input data is to be read. If $2 \leq$ NDIFEQ ≤ 7, the program is at some intermediate point in the integration from one point to the next. When NDIFEQ > 7, the integration of one step has been completed and NOUT is set equal to 1 so that the output subroutine will be called after the derivatives are calculated.

I-3. SUBROUTINE ADAMS

Subroutine ADAMS is the subroutine which integrates the set of differential equations. The subroutine will not be described in detail, however, an examination of the flow chart (fig. I-4) and the program listing (fig. 1(d)) will indicate how it functions. All returns from the subroutine are to the main program.

Consider the following set of n differential equations:

$$\dot{y}_1 = f_1(t, y_1, y_2, \ldots, y_n)$$
$$\vdots$$
$$\dot{y}_n = f_n(t, y_1, y_2, \ldots, y_n)$$

This subroutine uses a fourth-order Adams predictor-corrector method (ref. 10) to solve the above set of equations. To find the value of y_i at the $(J + 4)^{th}$ step, the following formula is used to predict the value

$$y^{(p)}_{i,j+4} = y_{i,j+3} + \frac{h}{24}\left(55\dot{y}_{i,j+3} - 59\dot{y}_{i,j+2} + 37\dot{y}_{i,j+1} - 9\dot{y}_{i,j}\right) \qquad \text{(I-1)}$$

This assumes that all of the $\dot{y}_i$'s are known for the j, (j + 1), (j + 2), and (j + 3) steps. The quantity h is the interval in the independent variable between these points. After the values of the $y^{(p)}_{i,j+4}$ have been found, the following equation is used to obtain the corrected values:

$$y_{i,j+4} = y_{i,j+3} + \frac{h}{24}\left(9\dot{y}^{(p)}_{i,j+4} + 19\dot{y}_{i,j+3} - 5\dot{y}_{i,j+2} + \dot{y}_{i,j+1}\right) \qquad \text{(I-2)}$$

The use of the above equations requires that four evenly spaced values of the dependent variables be known. These are found in this subroutine by means of a fourth-order Runge-Kutta method (ref. 10). To find the values of the y_i's at the $(j + 1)^{th}$ step, the following equation is used:

$$y_{i,j+1} = y_{i,j} + \frac{1}{6}\left(k_{i,1} + 2k_{i,2} + 2k_{i,3} + k_{i,4}\right) \qquad \text{(I-3)}$$

where

$$
\left.
\begin{aligned}
k_{i,1} &= hf_i\left(t_j,\ x_{1,j},\ x_{2,j},\ \ldots,\ x_{n,j}\right) \\
k_{i,2} &= hf_i\left(t_j + \tfrac{1}{2}h,\ x_{1,j} + \tfrac{1}{2}k_{1,1},\ \ldots,\ x_{n,j} + \tfrac{1}{2}k_{n,1}\right) \\
k_{i,3} &= hf_i\left(t_j + \tfrac{1}{2}h,\ x_{1,j} + \tfrac{1}{2}k_{1,2},\ \ldots,\ x_{n,j} + \tfrac{1}{2}k_{n,2}\right) \\
k_{i,4} &= hf_i\left(t_j + h,\ x_{1,j} + k_{1,3},\ \ldots,\ x_{n,j} + k_{n,3}\right)
\end{aligned}
\right\} \qquad \text{(I-4)}
$$

Thus, given initial values of the dependent variables, the y_i's, the independent variable, t, and the integration interval size, h, the differential equations are integrated three steps using equations (I-3) and (I-4). At this point, the integration is continued using equations (I-1) and (I-2).

A discussion of both the Adams and Runge-Kutta methods is presented in reference 10. From this reference, the truncation error, Δy, at a given step can be shown to be

$$\Delta y = \left(\frac{y_{i,j+4} - y^{(p)}_{i,j+4}}{14.2}\right) \qquad \text{(I-5)}$$

so that the absolute error is

$$\Delta y_{ABS} = |\Delta y|$$

and the relative error is

$$\Delta y_{REL} = \frac{\Delta y_{ABS}}{\left|y_{i,j+4}\right|} \qquad \text{(I-6)}$$

At the end of each integration step, error tests could be performed and the integration interval, h, adjusted accordingly. This procedure is described in the flow chart (fig. I-4). In the present version of the program,

however, a fixed interval size is used and no attempt is made to satisfy error specifications.

The quantities in the parameter list are:

H	current value of the integration interval
DS	integration interval
Y	array containing current values of the dependent variables
DY	array containing current values of the derivatives of the dependent variables
NEQ	number of equations being integrated; routine dimensioned for a maximum of 12
NDIFEQ	control index
S	current value of independent variable

I-4. SUBROUTINE BDYGEN

Subroutine BDYGEN calculates the line source and doublet strengths using control points on the surface of the fuselage and store. The method used is described in Appendix I of reference 1. A listing of the subroutine is presented in figure 1(e), a flow chart in figure I-5, and a table equating the algebraic and program notation in Table I-2 of this report. The variables appearing in the subroutine parameter list are included in this table. The coordinate system associated with the subroutine is shown in the sketches of Appendix I of reference 1.

At the beginning of the subroutine a test is performed to determine if, at the base of the body, the radial distance to the Mach cone emanating from the body nose is less than the maximum radius of the body. If so, an error message is printed out (see section 4 of this report) and the program stops.

Next, as the flow chart indicates, N is set equal to NXBODY - 1 and the body axis is divided into N segments of equal length. The x locations of the body definition points, XBODY(J), are determined at these equally spaced axis points and, inside a short DO loop, subroutine SHAPE is called to calculate the radius of the body, RBODY, and the surface slope, RPBODY, at each body definition point. Subroutine SHAPE requires that the shape definition quantities be made dimensionless by body length. Subroutine BDYGEN accounts for this before and after the

calls to subroutine SHAPE. Next, the control points are located effectively midway between the body definition points. Subroutine SHAPE calculates the body radius, RF, and the surface slope, DRDX, at each control point. Finally, the axis points, TX, which are the origins of the conical line sources and doublets, are determined.

The next section of the subroutine is a routine for dropping control points that are located outside the Mach cone emanating from the body nose. The routine is a loop in which the first control point is tested. If it is found to be outside the Mach cone, the second body definition point is dropped and a new first control point is determined as described above. The remaining control points and conical line source and doublet origins are redefined so that the second point of each set is dropped. The loop is repeated until a control point is found within the nose Mach cone.

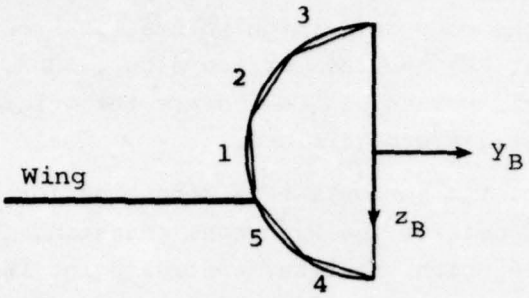
The remainder of the subroutine calculates the source and doublet strengths at the control points. Subroutine SOURCE is called and the source strength at the first control point is calculated using equation (I-14) and at the remaining control points using equation (I-17) of reference 1. One should note that the I^{th} source strength, T(I), is the constant K_{I-1} in the algebraic notation of the equations. Similarly, using subroutine DOUBLT, the doublet strengths are calculated using equation (I-27) for the first control point and equation (I-28) for the remaining control points. Here, the I^{th} doublet strength, TC(I), equals $K_{d,I-1}$.

After calculating the source and doublet strengths, the subroutine prints the body definition point data, the singularity origins, and the singularity strengths.

I-5. SUBROUTINE BLYOUT

Subroutine BLYOUT calculates quantities needed to define the constant u-velocity panels on the fuselage and to locate the panel control points. The corner and control point coordinates calculated for the body panels are stored in arrays in locations which follow the same quantities calculated for the wing and pylon constant u-velocity panels. All coordinates are in the wing coordinate system which is shown in figure 6 of this report. A listing of the subroutine is presented in figure 1(f), a flow chart in figure I-6, and a table equating algebraic and program notations in Table I-3.

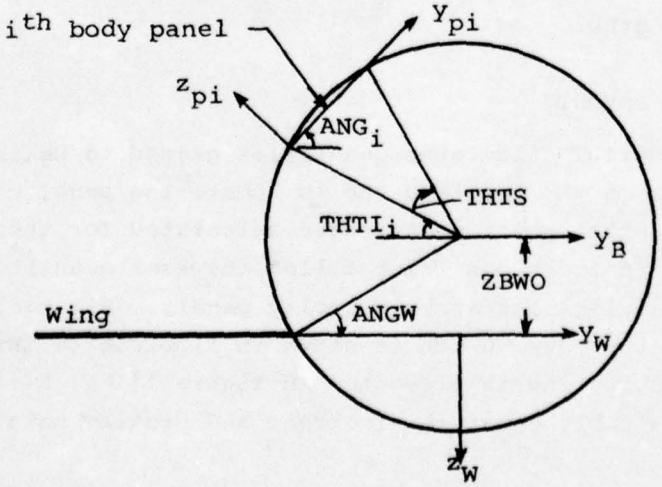
The fuselage interference panels are laid out on the left-half of the fuselage surface, as shown in figure 4. The panel numbering convention employed in a typical ring containing five panels, three above the wing and two below, is shown in the following sketch.



At the beginning of the subroutine quantities DX, NBD, NBIP, and ANGW, associated with the geometry of the entire panel layout, are calculated. Next, the variable ANGW is tested to determine if the wing is tangent to the fuselage, either at the top or at the bottom. As the flow chart shows, this tangency condition determines the initialization procedure performed before the main loop calculations.

The main part of the subroutine is a double DO loop with the outer loop index running over the rows above or below the wing and the inner loop index over the NCWB panels in a lengthwise row. If the wing is not tangent, this double loop is performed first for the panels above the wing, then repeated for the panels below the wing.

For a given lengthwise row the y,z coordinates of the panel corners and the control points are first obtained in the panel coordinate system. The following sketch shows the coordinate system and certain quantities associated with a panel located above the wing.



Within the inner DO loop the coordinates are transformed into the wing coordinate system and stored in appropriate array locations. Polar angle, THTI and quantities SNT2 and CST2, necessary for panel-wing transformations, are saved in arrays for subsequent use by other routines. The corner and control point x_w coordinates are calculated and stored; no transformations are necessary for these quantities.

The convention employed in labeling the corner coordinates for the fuselage interference panels is that the right side of the panel is located in clockwise rotation from the left side, when looking forward.

I-6. SUBROUTINE CEL1

Subroutine CEL1 calculates the complete elliptic integral of the first kind. This subroutine has been taken directly from reference 11. For a description of the routine that reference should be consulted. A listing of the routine is presented in figure 1(g). The comment cards give a brief explanation of the use of the routine.

I-7. SUBROUTINE CEL2

Subroutine CEL2 calculates the complete elliptic integral of the second kind. This subroutine has been taken directly from reference 11. For a description of the routine that reference should be consulted. A listing of the routine is presented in figure 1(g). The comment cards give a brief explanation of the use of the routine.

I-8. SUBROUTINE DIRCOS

Subroutine DIRCOS computes the direction cosines which relate the store body coordinate system to the inertial coordinate system. The direction cosines are given by equation (28) of reference 4. A listing of the routine is presented in figure 1(h). The three angles Ψ,θ, and Φ are brought into the subroutine in A(10), A(11), and A(12), respectively, and the direction cosines are returned in the D array.

I-9. SUBROUTINE DOUBLT

Subroutine DOUBLT calculates coefficients used in the determination of the line doublet strengths. They occur as terms in equations (I-27) and (I-28) of reference 1. The relation of the coefficients to perturbation velocities, $u_{B,d}/V_\infty$ and $v_{B,d}/V_\infty$, induced by a number of line doublets

distributed along the body centerline is shown in the first two of equation (I-30) in which the coefficients occur as the multipliers of $K_{d,n} \cos \theta$. A listing of the subroutine is presented in figure 1(h) of this report. The subroutine is called by subroutine BDYGEN and section I-4 should be referred to for further details concerning the doublet strength calculations.

For a specified control point, x_B, r_B, and singularity origin ξ, the subroutine calculates quantities U and V according to the following equations:

$$U = \beta \sqrt{\left(\frac{x_B - \xi}{\beta r_B}\right)^2 - 1}$$

$$V = -\frac{\beta^2}{2}\left[\cosh^{-1}\left(\frac{x_B - \xi}{\beta r_B}\right) + \left(\frac{x_B - \xi}{\beta r_B}\right)\sqrt{\left(\frac{x_B - \xi}{\beta r_B}\right)^2 - 1}\right]$$

At the beginning of the subroutine a test is performed to determine if the control point is ahead of the Mach cone from the doublet origin. If so, U and V are set to zero and control is returned to the calling program.

The following table of definitions contains most of the variable names used in the subroutine:

BETA	$\beta = \sqrt{M_\infty^2 - 1}$
RFIELD	r_B; radius of body at control point
TX	ξ; location on body axis of doublet origin, positive measured from tip of nose
U	coefficient defined by first equation above
V	coefficient defined by second equation above
XFIELD	x_B; x location of control point, positive measured from tip of nose

I-10. SUBROUTINE DPCOEF

Subroutine DPCOEF calculates the coefficient matrix of the set of simultaneous boundary condition equations which are to be solved for the constant u-velocity panel singularity strengths. A listing of the

subroutine is presented in figure 1(h), a flow chart in figure I-7, and a table equating the algebraic and program notation in Table I-4 of this report.

The elements of the matrix are the aerodynamic influence coefficients described in section 3.3.4 of reference 1. They occur indirectly in the summation terms in the left-hand side of equations (9), (10), and (11) of that reference. The actual coefficients which the subroutine calculates, FVN(ν,n), are related to the summation terms through the panel strengths, u_{+n}/V_∞, by the following equations in which n is the index of the influencing panel and ν is the control point index:

$$\frac{w_{w_{\nu,n}}}{V_\infty} = \frac{u_{+n}}{\pi V_\infty} \quad FVN(\nu,n) \qquad \nu = 1,2,\ldots,NPANLS,$$

$$\frac{-v_{w_{\nu,n}}}{V_\infty} = \frac{u_{+n}}{\pi V_\infty} \quad FVN(\nu,n) \qquad \nu = N1P,N1P+1,\ldots,N2,$$

$$\frac{v_{N_{\nu,n}}}{V_\infty} = \frac{u_{+n}}{\pi V_\infty} \quad FVN(\nu,n) \qquad \nu = N2P,N2P+1,\ldots,NPTOT,$$

for n = 1,2,...,NPTOT.

The subroutine consists of three double DO loops. The first loop uses subroutine VELWP to calculate the influence of the wing panels at the wing, pylon, and fuselage control points. The second loop is bypassed if there is no pylon, NPY = 0. If a pylon is present, the influence coefficients for the pylon panels are calculated by means of subroutine VELPP. The third double loop is bypassed if there is no fuselage, NFU = 0. Otherwise, the influence coefficients for the fuselage panels are calculated using subroutine VELBD.

Within each outer loop are three inner loops in series, which fix the control point location on the wing, pylon, or fuselage, respectively. On the flow chart, only the first occurrence of the first inner loop is shown in detail, which includes the call to subroutine VELWP. The remaining inner loops have the same logical structure and are not shown. If the

control point is on a fuselage panel, the influence coefficient is the component normal to the body surface at the control point. This is obtained by a rotation through the panel orientation angle using quantities calculated previously in subroutine BLYOUT.

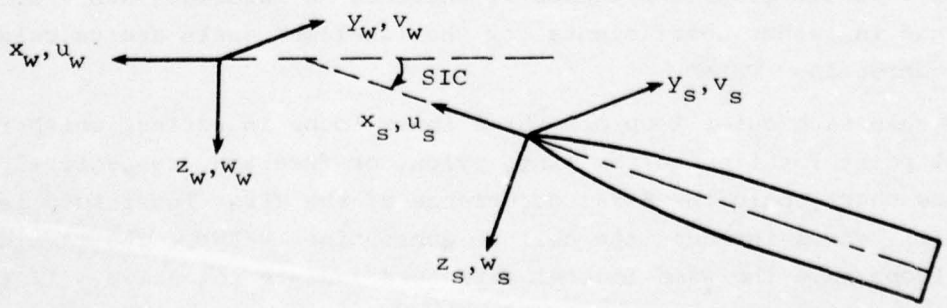
I-11. SUBROUTINE DPRHS

Subroutine DPRHS calculates the right-hand-side vector of the set of simultaneous boundary condition equations which are to be solved in order to determine the constant u-velocity panel singularity strengths. The boundary conditions are specified in equations (9), (10), and (11) of reference 1. A listing of the subroutine is presented in figure 1(i), a flow chart in figure I-8, and a table equating the algebraic and program notation in Table I-5 of this report.

The major portion of the subroutine is devoted to evaluating the externally induced perturbation velocities, $^{u}w_{i,v}/V_{\infty}$, $^{v}w_{i,v}/V_{\infty}$, and $^{w}w_{i,v}/V_{\infty}$, at all of the wing, pylon, and fuselage control points.

The first section calculates velocities induced at wing and pylon control points by the fuselage line sources, sinks, and line doublets if a fuselage is present (NFU $\neq$ 0). A control point is located in the fuselage coordinate system (see figure 5) and subroutine VELCAL is called to calculate the velocities at this point. These velocities are summed in the UEI, VEI, and WEI arrays.

The perturbation velocities induced by the store are calculated next in a double DO loop. (The subroutine has been written to accommodate multiple stores although in the present program the number of stores is limited to one.) The velocities induced at the fuselage control points are included in these calculations. A control point is first located in the store coordinate system shown in the sketch below.



Subroutine VELCAL then computes the perturbation velocities which are resolved into the wing system and added to the appropriate array.

If a pylon is present, NPY $\neq$ 0, its thickness distribution produces perturbation velocities at the wing control points. Subroutine VELPTH is called for the purpose of calculating these velocities which are added to the UEI, VEI, and WEI arrays. Similarly, subroutine VELWTH is called to calculate the wing thickness perturbation velocities at the pylon control points. Following this, each subroutine is called a second time to compute velocities at the fuselage control points if a fuselage is present.

The last three loops in the subroutine calculate the right-hand sides of equations (9), (10), and (11). If the pylon is located below the fuselage centerline, CENTER = TRUE, VEI is set to zero for all pylon control points.

It should be noted that for a fuselage control point the perturbation velocity normal to the fuselage surface at the control point is used for the right-hand side.

I-12. SUBROUTINE ELI1

Subroutine ELI1 calculates the general elliptic integral of the first kind. This subroutine has been taken directly from reference 11. For a description of the routine that reference should be consulted. A listing of the routine is presented in figure 1(j). The comment cards give a brief explanation of the use of the routine.

I-13. SUBROUTINE ELI2

Subroutine ELI2 calculates the general elliptic integral of the second kind. This subroutine has been taken directly from reference 11. For a description of the routine that reference should be consulted. A listing of the routine is presented is figure 1(k). The comment cards give a brief explanation of the use of the routine.

I-14. SUBROUTINE FLSQFY

Subroutine FLSQFY performs a weighted least square orthogonal polynomial curve fit of a set of x,y data. The routine is based on reference 12. A listing of the subroutine is presented in figure 1(ℓ) of this report.

Given a set of points (x_i,y_i), $i = 1,2,\ldots,M$ and corresponding positive weights w_i, there exist orthogonal polynomials $P_r(x)$ of degree r, $r = 0,1,\ldots,M-1$, and coefficients $C_0,C_1,\ldots,C_N$ such that, for any $k \leq N$,

$$Q(x) = C_0 + C_1P_1(x) + \ldots + C_kP_k(x)$$

is that polynomial of degree k for which the scalar

$$\sum_{i=1}^{M} w_i\left[Q(x_i) - y_i\right]^2$$

has its minimum value.

The subroutine calculates an array A of $N+1$ coefficients of the least squares polynomial in standard form $\left[A(i + 1) = a_i\right]$:

$$Q(x) = a_0 + a_1x + \ldots + a_Nx^N$$

Best results are obtained from the subroutine if a preliminary coordinate transformation is performed on the x data points so that the origin lies in the middle of the x data. This is performed in the calling program.

The quantities in the parameter list are

M	number of data points
N	degree of polynomial desired; $0 < N < M$
X	array of independent variable
Y	array of dependent variable
W	array of positive weights in correspondence to the x array; for an unweighted fit, $w_i = 1$ for all i
MN1	row dimension of scratch array S1; $MN1 \geq M+N+1$
S1	scratch array

A	array of computed coefficients in order of increasing degree
IER	error indicator (not used)

I-15. SUBROUTINE FUSEIO

Subroutine FUSEIO reads and prints the input data which describe the fuselage and calculates the line source and doublet distributions as described in Appendix I of reference 1. A listing of the subroutine is presented in figure 1(m), a flow chart in figure I-9, and a table equating the algebraic and program notation in Table I-6 of this report.

The subroutine first reads in and prints input items 5, 6, 7, and 8 which consist of the fuselage length, the maximum radius, and the polynomials specifying the fuselage shape. Next, the data used to lay out the body interference panels, input items 9 and 10, are read and printed. Finally, subroutine BDYGEN is called for the purpose of calculating the source and doublet distributions.

I-16. SUBROUTINE INTOST

Subroutine INTOST (see fig. 1(m) for a listing) takes a vector with components specified in the inertial ξ,η,ζ coordinate system directions and transforms it into a vector with components in the store x,y,z coordinate system directions, see figure 14 of this report. That is,

$$\begin{bmatrix} s_x \\ s_y \\ s_z \end{bmatrix} = [A]' \begin{bmatrix} s_\xi \\ s_\eta \\ s_\zeta \end{bmatrix}$$

The matrix [A]' is the transpose of the direction cosine matrix given by equation (II-2) of reference 4. The transpose is equal to the inverse since [A] is orthogonal. The matrix [A] was calculated in subroutine DIRCOS.

In terms of the above notation, the quantities in the parameter list of the subroutine are

XI	s_ξ
ETA	s_η
ZETA	s_ζ
X	s_x
Y	s_y
Z	s_z
DC	[A]

I-17. SUBROUTINE INVERS

Subroutine INVERS (see fig. 1(m) for a listing) solves a set of simultaneous linear algebraic equations by Gaussian elimination. This routine comes from reference 13 which contains a flow chart. Not all of the options shown in that flow chart are in the present routine. The quantities in the parameter list are

A	array containing coefficient matrix and right-hand sides
NSYS	number of right-hand sides
N	number of equations
NMAX	first dimension of A array in calling program
MMAX	second dimension of A array in calling program

If the coefficient matrix is found to be singular an error message is printed out (see section 4) and the program stops.

I-18. SUBROUTINE NUMACH

Subroutine NUMACH determines a local Mach number at a given field point lateral and vertical position based on the first occurrence of an induced downward flow velocity behind the free stream Mach cone associated with wing thickness. The method used is described in section 3.4 of reference 1. The routine also calculates least-squares polynomials which approximate the velocities at the specified y_w, z_w location. The velocities are based on the local Mach number and represent the sum of velocities induced at each point of the traverse by the wing-fuselage aircraft components. A listing of the program is presented in figure 1(n), a flow chart in figure I-10, and a table equating the algebraic and program notation in Table I-7 of this report.

At the beginning of the subroutine a test is performed to determine if the specified z_w location (Z) is below the wing. If not, an error message is printed (see section 4) and the program stops.

Next, the subroutine calculates the local wing chord, CHRD, at the specified y_w location (Y). Quantities previously determined by subroutine WLYOUT are used in this calculation. As shown in the flow chart, if the field point is below the fuselage, CHRD equals the wing root-chord; outboard of the wing tip CHRD equals the tip-chord.

After initializing certain control variables, the subroutine next begins the calculation of an x traverse aft of $x_w = 0$. At each point the velocity due to wing thickness is calculated (subroutine VELWTH). The point at which wing thickness first induces a downward velocity is determined using an x-interval size, DELX1, equal to one-tenth CHRD. The search is refined twice, each time backing up one step and repeating the calculations using DELX1 equal to one-third the previous interval size. Once the point in the traverse is isolated (XT3,Y,Z) at which the downward velocity is first felt, the local Mach number, M_ℓ, specified by equations (27) and (28) of reference 1 is computed from the velocity components.

The search for XT3 described above is next repeated with velocity calculations based on M_ℓ (specified when INUMCH = 1). The new value of XT3 defines the beginning of the range of influence associated with the wing thickness panels. The subroutine next determines the end of this range by searching for the point in the traverse (XT4,Y,Z) at which a 50 percent decrease in the magnitude of the downward velocity due to wing thickness occurs. The traverse begins at seven-tenths CHRD aft of XT3. The search is very similar to that for XT3.

The remainder of the subroutine calculates the flow field, based on M_ℓ, at a series of NCW points in the XT3, XT4 range. Contributions from all singularities associated with the wing and fuselage are included in the calculations. The flow field is then smoothed by fitting each velocity component with a least-squares cubic polynomial over the data points in the XT3, XT4 range. Subroutine FLSQFY is called for this purpose. The returned polynomial coefficients are stored in the array WSCOE.

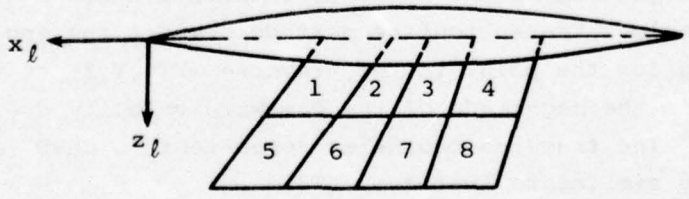
I-19. SUBROUTINE PLYOUT

Subroutine PLYOUT reads in data which describe the geometric characteristics of the pylon and calculates quantities which specify the pylon constant u-velocity panels. The pylon input variables are shown in figure 8 of this report. A listing of the subroutine is presented in figure 1(P), a flow chart in figure I-11, and a table defining the program notation in Table I-8 of this report.

The first part of the subroutine reads in and prints input items 19, 20, and 21. The total number, MP, of pylon constant u-velocity panels is calculated. Next, the pylon leading and trailing edge sweeps and their difference, PSLPDF, are calculated. The variables PLEX and CSIDE are initialized for the calculation of the first chordwise row of panels.

The remainder of the subroutine is a double DO loop within which panel leading-edge and trailing-edge *sweeps*, *corner* coordinates, and control point coordinates are calculated. These quantities for the pylon are stored in arrays following the same quantities calculated by subroutine WLYOUT for the NPANLS constant u-velocity panels on the wing.

The following sketch shows the numbering convention associated with the pylon panels for a typical pylon-under-wing layout with four chordwise and two spanwise panels.



In the subroutine notation, the panel corners labeled left are those farther from the pylon root chord.

I-20. SUBROUTINE PYTHIN

Subroutine PYTHIN reads in the pylon thickness data, input item 22 and 23. A listing of the subroutine is presented in figure 1(p) of this report.

The number of panels in a chordwise row, NCPS, is first read in along with an index, NUNIP, which indicates whether the thickness distribution is similar at all spanwise stations. If it is similar, NUNIP = 1, the values of $\tan \theta_p$ are read in for the first row and then the values of $\tan \theta_p$ for the other rows are set equal to those of the first row. If the distribution is not similar, NUNIP = 0, the values of $\tan \theta_p$ for all rows are read in.

The following table contains most of the variable names used in the subroutine. Section 3.2.1 should be referred to for the definition of a variable that is an input quantity.

MPS	number of thickness panels on the pylon; MSP*NCPS
MSP	input, item 20
NCPS	input, item 22
NUNIP	input, item 22
THETPL(I)	input, item 23

I-21. SUBROUTINE RESVEL

Subroutine RESVEL, along with other subroutines which it calls, calculates the perturbation velocities at a specified field point due to all aircraft components other than the store. A listing of the subroutine is presented in figure 1(q), a flow chart in figure I-12, and a table equating the algebraic and program notation in Table I-9 of this report. The variables appearing in the subroutine parameter list are defined in this table.

The subroutine first transforms the field point coordinates into the wing coordinate system and initializes the velocity sums to zero. Next, a test is performed to determine whether the wing-fuselage velocities are to be calculated using a least-squares polynomial approximation, described in section I-18, or in accordance with equations in section 3 of reference 1. If ISMTH = 0, the program calculates and sums velocities using the latter methods. First, if a fuselage is present, NFU = 1, velocities induced by the source and doublet distributions are calculated using subroutine VELCAL and velocities due to the fuselage interference panels using subroutine VELBD. Next, the wing constant u-velocity panel and wing thickness panel velocities are calculated using subroutines VELWP and VELWTH, respectively.

Finally, if a pylon is present, NPY = 1, the pylon velocities are calculated by calling VELPP for constant u-velocity panels and VELPTH for thickness panels.

If ISMTH = 1, a second test is performed to determine whether the x_w coordinate of the field point is within the range of x, at a specified y,z location, for which a smoothed velocity field has been previously calculated by subroutine NUMACH (see section I-18). This range is referred to in the flow chart as the XT3, XT4 range since these quantities are the limiting x_w values of the range. If the field point x_w coordinate is inside this range, wing-fuselage induced velocities are calculated using the least-squares polynomials. Pylon velocities are calculated as described above for ISMTH = 0. If the field point x_w coordinate is outside the XT3, XT4 range, all velocities are calculated as described for ISMTH = 0.

I-22. SUBROUTINE SEMFOR

Subroutine SEMFOR calculates the empennage forces and moments by the method described in section 5.3 and Appendix I of reference 4. A listing of the subroutine is presented in figure 1(q), a flow chart in figure I-13, and a table equating the algebraic and program notation in Table I-10 of this report.

An examination of the flow chart shows that the first steps in the routine are to locate the point at which the empennage forces act relative to the store moment center and to set JMAX equal to 2 or 4 depending on whether the empennage is planar or cruciform.

The next part of the subroutine calculates the perturbation velocity field at the MSF control points on each of the JMAX fins. After the velocities are calculated by subroutine RESVEL they are resolved into the store-body coordinate system by subroutine INTOST. The free-stream components are then calculated, resolved into the store-body coordinate system, and added to the perturbation velocities. The resultant velocities are made dimensionless by the store free-stream velocity and the pitch and yaw damping terms are added if aerodynamic damping is being included. From these velocities the components normal to the fin surfaces are determined. Positive directions are shown in figure 10 of reference 4.

The remainder of the routine calculates the empennage forces and moments. First W_o and V_o shown in figure 10 of reference 4 are determined from the velocity field used in the store body force and moment

calculation. Then, the normal force and the side force, if the empennage is cruciform, are calculated using equations (I-13) and (I-18) of reference 4. The spanwise integrations are performed using Simpson's rule. It is to be noted that in the present program all four fins are assumed to have the same span, $s_h = s_v$. The pitching moment and yawing moment are calculated using equations (I-21) and (I-22).

These forces and moments are in the fin coordinate system. They are resolved into the body coordinate system using equations (58) through (61) of reference 4.

Finally, if rolling moment is to be calculated this is done using equation (I-30) or (I-52) of reference 4. Equation (I-30) is for a planar empennage and (I-52) is for a cruciform empennage.

I-23. SUBROUTINE SEMPIN

Subroutine SEMPIN initializes certain quantities which will be used repeatedly in the empennage force and moment calculation, subroutine SEMFOR. The equations programmed are given in Appendix I of reference 4. A listing of the subroutine is presented in figure 1(r), a flow chart in figure I-14, and a table equating the algebraic and program notation in Table I-11 of this report.

The flow chart indicates that the first calculation performed is to determine the radial distance outward from the body axis to the MSF fin control points. The first point is at the body-fin juncture, $r_f = a$ (see fig. 10, ref. 4), and the last is at $r_f = s_h = s_v$. The others are equally spaced in between these two points. Next, a check is made to determine that XTAIL was input as a negative quantity and then the angular orientation of the fins in the store-body coordinate system is determined. Referring to figure 10 of reference 4, these angles are measured in the clockwise direction from the z_s axis.

JMAX is next set equal to 2 or 4 depending on whether the empennage is planar or cruciform and then the y_s and z_s coordinates of the control points on all of the fins are determined. Next, certain constants are calculated and then the values of $(cc_\ell)_3$ are calculated at the control points. They are the same for all panels since $s_h = s_v$ (see eqs. (I-14) and (I-19), ref. 4).

If rolling moment is not to be calculated, NROLL = 0, control is returned to the calling program. If rolling moment is to be calculated, and the empennage is planar, IPLNR = 1, $(cc_\ell)_5$ given by equation (I-29) of reference 4 is calculated at the control points. Note that in the program the following substitution is made

$$\cosh^{-1}(x) = \ell n(x + \sqrt{x^2 - 1})$$

For a cruciform empennage, IPLNR = 0, equation (I-51) of reference 4 is used for the first control point where $y_f = a$. For the other control points equation (I-38) is used. The following substitution is made in the program

$$\tanh^{-1}(x) = \frac{1}{2}\ \ell n\ \frac{(1 + x)}{(1 - x)}$$

I-24. SUBROUTINE SFORCE

Subroutine SFORCE calculates the store-body forces and moments by the methods described in section 5.2 of reference 4. A listing of the subroutine is presented in figure 1(s), a flow chart in figure I-15, and a table equating the algebraic and program notation in Table I-12 of this report.

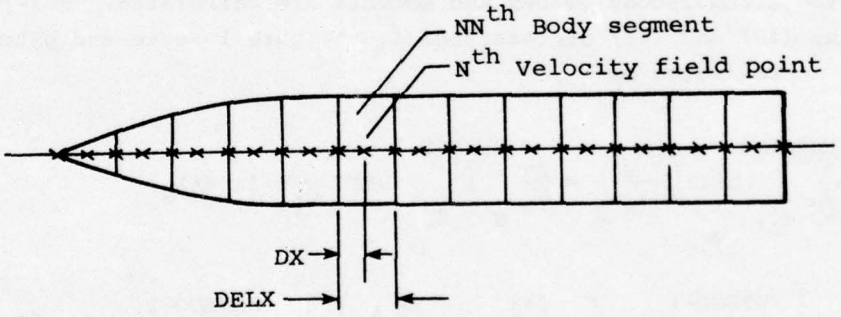
An examination of the flow chart shows that the first step in the routine is to check to see if the trajectory being calculated is to simulate a wind-tunnel captive-store trajectory. If it is, NGAM = 1, then the store attitude relative to the parent aircraft is changed. In wind-tunnel captive-store testing the yaw and pitch angles of the store are changed, only while measuring the aerodynamic forces and moments, to account for translational motion relative to the aircraft. The new angles are

$$\Psi_{cs} = \Psi_{ff} - \tan^{-1}\left(\frac{\dot{\eta}}{V_\infty \cos\ \alpha_f + \dot{\xi}}\right)$$

$$\Theta_{cs} = \Theta_{ff} + \tan^{-1}\left(\frac{\dot{\zeta}}{V_\infty \cos\ \alpha_f + \dot{\xi}}\right)$$

The subscript cs refers to captive store and ff refers to free flight. The direction cosines between this new body coordinate system and the inertial or fuselage system are calculated in DIRCOS. If the trajectory is a free-flight case, the direction cosines between the true body coordinate system and the inertial system are calculated.

In the next section of the program subroutine NUMACH is called (see section I-18). Following this, the program calculates the velocity field given by equation (40) of reference 4 at specified points along the separated store longitudinal axis. The store is removed from the flow field during this calculation. One of the input parameters was NSEG which is the number of equal length segments the body is to be broken into. These segments are of length DELX as shown in the following sketch.



The velocities are calculated at the midpoint of each segment as well as at the two ends, the points indicated by x in the sketch. There are thus NHSEG = 2*NSEG + 1 points. The perturbation velocity field is first calculated for all NHSEG points. These velocities are the u,v, and w components given by equation (37) of reference 4. After the velocities are calculated by subroutine RESVEL, they are resolved into the store-body coordinate system by subroutine INTOST. The free-stream velocity components are next determined, resolved into the store-body coordinate system and added to the perturbation velocities given by equation (38). At this time the velocities are made dimensionless by V_{∞_s}. If damping is being included the damping terms in equation (40) are added.

The routine next calculates the forces and moments. The buoyancy forces and moments are first determined. From equations (42) and (44) of reference 4 the normal force and the pitching moment can be approximated by

$$(C_N)_{BY} = \sum_{N=2,4..}^{NHSEG-1} DELX \left(\frac{dC_N}{dx_s}\right)_N = \frac{2\pi}{S_R} \sum_{N=2,4..}^{NHSEG-1} DELX \left[a_N^2 \frac{\left(W^*_{s_{N+1}} - W^*_{s_{N-1}}\right)}{DELX}\right]$$

$$(C_m)_{BY} = \frac{1}{\ell_R} \sum_{N=2,4..}^{NHSEG-1} DELX \left[(x_{s,m} - x_s)\left(\frac{dC_N}{dx_s}\right)_N\right]$$

These are the equations which are programmed. Similar equations can be written for the side force and yawing moment.

Next the slender-body forces and moments are calculated. Referring to equations (46) and (48) of reference 4, the normal force and pitching moment are approximated by

$$(C_N)_{SB} = \sum_{N=2,4..}^{NHSEGO-1} DELX\left(\frac{dC_N}{dx_s}\right)_N = \frac{2\pi}{S_R} \sum_{N=2,4..}^{NHSEGO-1} DELX \frac{d}{dx_s} (a^2 W^*_s)_N$$

$$= \frac{2\pi}{S_R} \left\{ \sum_{N=2,4...}^{NHSEGO-1} DELX \left[a_N^2 \frac{\left(W^*_{s_{N+1}} - W^*_{s_{N-1}}\right)}{DELX}\right] + \sum_{N=2,4...}^{NHSEGO-1} DELX \left(a_N \frac{da_N}{dx_s} W^*_{s_N}\right) \right\}$$

$$(C_m)_{SB} = \frac{1}{\ell_R} \sum_{N=2,4...}^{NHSEGO-1} DELX \left[(x_{s,m} - x_s)\left(\frac{dC_N}{dx_s}\right)_N\right]$$

Here the summations are terminated at the assumed separation location as specified by NSEGXO of item 30 of the input data since NHSEGO = 2*NSEGXO+1. Note that the quantity in the first summation of $(C_N)_{SB}$ is the same as that in $(C_N)_{BY}$. Use has been made of this fact in the program. Similar expressions can be written for $(C_Y)_{SB}$ and $(C_n)_{SB}$.

If separation is assumed to occur ahead of the base of the store, NHSEGO < NHSEG, the viscous crossflow forces and moments in this region are calculated using equations (51) through (54) of reference 4. The normal force and pitching moment are approximated by

$$(C_N)_{CF} = \sum_{N=NHSEGO+1,+3..}^{NHSEG-1} DELX \left(\frac{dC_N}{dx_s}\right)_N = \frac{2c_{d_c}}{S_R} \sum_{N=NHSEGO+1,+3..}^{NHSEG-1} DELX \left(a_N V^*_{c_N} W^*_{s_N}\right)$$

$$(C_m)_{CF} = \frac{1}{\ell_R} \sum_{N=NHSEGO+1,+3..}^{NHSEG-1} DELX \left[(x_{s,m} - x_s) \left(\frac{dC_N}{dx_s}\right)_N \right]$$

I-25. SUBROUTINE SHAPE

The purpose of this subroutine is to calculate the body radius and surface slope at a specified axial station. The body shape is specified by a series of polynomials of the form of equation (2) of this report. A flow chart of subroutine SHAPE is presented in figure I-16 and a listing of the subroutine in figure 1(t).

The quantities in the parameter list are:

X	value of x/ℓ at which radius and surface slope are to be calculated
NS	number of polynomials describing body shape
XE	array containing values of x/ℓ for the end points of the NS polynomials
C	array containing the coefficients of the NS polynomials
R	calculated value of r/ℓ at $x/\ell = X$
DRDX	calculated value of dr/dx at $x/\ell = X$

The calculation performed by this subroutine consists of two steps. The first step is to determine which of the NS polynomials describes the shape at the value of X where the radius and surface slope are required. Once this is determined, the appropriate set of coefficients is used in equation (2) to determine r/ℓ. The value of dr/dx is found by differentiating equation (2).

$$\frac{dr}{dx} = \frac{C_7}{2}\left[\frac{2C_2\frac{x}{\ell} + C_3}{\sqrt{C_2\left(\frac{x}{\ell}\right)^2 + C_3\frac{x}{\ell} + C_4}}\right] + C_5 + 2C_6\frac{x}{\ell}$$

It should be noted that r/ℓ and dr/dx are calculated using the coefficients of the NS^{th} polynomial if x/ℓ is greater than XE(NS).

I-26. SUBROUTINE SIMSON

Subroutine SIMSON calculates the value of a definite integral using Simpson's rule. This can be found in any elementary numerical analysis book, for example, reference 14. As programmed here

$$I = \int_{x_o}^{x_o+m\Delta x} f(x)dx \cong \frac{\Delta x}{3}\Big\{f(x_o) + 4f(x_o+\Delta x) + 2f(x_o+2\Delta x) + 4f(x_o+3\Delta x)$$

$$+ 2f(x_o+4\Delta x) + \ldots + 4f[x_o+(m-1)\Delta x] + f(x_o+m\Delta x)\Big\}$$

where m must be an even number and 4 or greater. The subroutine is listed in figure 1(u) of this report. Referring to the listing and the above equation, the quantities in the subroutine parameter list are

N	m + 1
F	f(x)
DX	Δx
SUM	I

I-27. SUBROUTINE SOURCE

Subroutine SOURCE calculates coefficients used in the determination of the line source strengths. They occur as terms in equations (I-14) and (I-17) of reference 1. The relation of the coefficients to perturbation velocities, $u_{B,a}/V_\infty$ and $v_{B,a}/V_\infty$, induced by a number of line sources distributed along the body centerline is shown in the first two of equation (I-30) in which the coefficients occur as the multipliers of K_n. A listing of the subroutine is presented in figure 1(u) of this report. The subroutine is called by subroutine BDYGEN and section I-4 should be referred to for further details concerning the source strength calculations.

For a specified control point, x_B, r_B, and singularity origin, ξ, the subroutine calculates quantities U and V according to the following equations:

$$U = - \cosh^{-1}\left(\frac{x_B - \xi}{\beta r_B}\right)$$

$$V = \beta\sqrt{\left(\frac{x_B - \xi}{\beta r_B}\right)^2 - 1}$$

At the beginning of the subroutine a test is performed to determine if the control point is ahead of the Mach cone from the source origin. If so, U and V are set to zero and control is returned to the calling program.

The following table of definitions contains most of the variable names used in the subroutine:

BETA	$\beta = \sqrt{M_\infty^2 - 1}$
RFIELD	r_B; radius of body at control point
TX	ξ; location on body axis of source origin; positive measured from tip of nose
U	coefficient defined by first equation above
V	coefficient defined by second equation above
XFIELD	x_B; x location of control point, positive measured from tip of nose

I-28. SUBROUTINE SOUTPT

Subroutine SOUTPT prints the output at the end of each integration step. A listing of the subroutine is presented in figure 1(u), a flow chart in figure I-17, and a table equating the algebraic and program notation in Table I-13 of this report.

The current value of the time is first printed and then the force and moment components calculated in subroutines SFORCE and SEMFOR are summed up. The components and totals are then printed. If store thrust has been calculated (NTHRUS not equal to zero) the thrust force is printed. Next the normal-force and side-force distributions and the velocity field along the store centerline are printed. The x_s locations are the midpoints of the body segments.

The next section of the subroutine locates the store nose, moment center, and base in the fuselage or inertial system. These points are

located relative to the fuselage nose and also relative to where they would be had the store remained in the $t = 0$ position on the aircraft. These positions are printed.

The remainder of the subroutine prints the store moment center translational velocities and accelerations, the store rotational velocities and accelerations, and the store angular orientation and rates of change of these angles.

I-29. SUBROUTINE STORIO

Subroutine STORIO reads and prints the data which describe and locate the store and calculates the store source and doublet distribution. Although the number of stores in the present version of the program is restricted to one, subroutine STORIO has been written in a general form that will, with minor changes, accommodate multiple stores. A listing of the program is presented in figure 1(v), a flow chart in figure I-18, and a table equating the algebraic and program notation in Table I-14 of this report.

The subroutine first reads in and prints data for each store, including a shape number, NSHAPE(J). The next section of the program is a double DO loop. The outer loop reads in and prints polynomials specifying a particular store shape. In the inner loop this shape is assigned to the store with matching shape number, and subroutine BDYGEN is called to calculate the source and doublet distributions for this store. At the end of the double loop a check is made to see if the index NCOUNT is equal to the number of stores, NSTRS. If it is not, data are missing for some store. This causes the message "shape polynomials not input for all stores" to be printed and the program stops. The last part of the subroutine locates all the stores in the wing coordinate system. The wing coordinate system is shown in figure 6.

I-30. SUBROUTINE STTOIN

Subroutine STTOIN (see fig. 1(w) for a listing) takes a vector with components specified in the store x,y,z coordinate system directions and transforms it into a vector with components in the inertial ξ,η,ζ coordinate system directions, see figure 14 of this report. That is,

$$\begin{bmatrix} s_\xi \\ s_\eta \\ s_\zeta \end{bmatrix} = [A] \begin{bmatrix} s_x \\ s_y \\ s_z \end{bmatrix}$$

The matrix [A] is given by equation (II-2) of reference 4 and was calculated in subroutine DIRCOS.

In terms of the above notation, the quantities in the parameter list of the subroutine are

X	s_x
Y	s_y
Z	s_z
XI	s_ξ
ETA	s_η
ZETA	s_ζ
DC	[A]

I-31. SUBROUTINE SWNGIN

Subroutine SWNGIN reads in data required to describe the geometric characteristics of the wing and to lay out the wing constant u-velocity panels. A listing of the subroutine is presented in figure 1(w) of this report.

The first part of the subroutine reads in and prints input items 12, 13, and 14, which consist of wing geometry data and quantities used to locate the trapezoidal-shaped elemental panels. Next, the wing twist and camber distribution, if any, is read, input items 15 and 16. Two indices, NTAC and NUNI, are first input. If NTAC = 0 there is no twist and camber. The index NUNI indicates whether the twist and camber distribution is similar at all spanwise stations. If it is similar, NUNI = 1, the values of $\tan \alpha_\ell$ are read in for the first row and then the values of $\tan \alpha_\ell$ for the other rows are set equal to those of the first row. If the distribution is not similar, NUNI = 0, the values of $\tan \alpha_\ell$ for all rows are read in.

The following table contains definitions of most of the variable names used in the subroutine. Section 3.2.1 of this report should be referred to for the definition of a variable that is an input quantity.

ALPHAL(I)	$\tan \alpha_\ell$; input, item 16
CRW	length of wing root chord; input, item 12
MSW	input, item 13
MSWP	MSW + 1
NCW	input, item 13
NPANLS	number of constant u-velocity panels on wing; NCW*MSW
NTAC	input, item 15
NUNI	input, item 15
PSIWLE(I)	input, item 14
PSIWTE(I)	input, item 14
SSPAN	semispan of wing; input, item 12
Y(I)	input, item 14

I-32. SUBROUTINE THKLYT

Subroutine THKLYT calculates quantities which characterize the wing and pylon thickness source panels. These quantities are stored in arrays in each of which the wing panel variables precede the pylon variables. All panel coordinates are expressed in the wing coordinate system which is shown in figure 6. A listing of the subroutine is presented in figure $1(_w)$, a flow chart in figure I-19, and a table equating the algebraic and program notation in Table I-15 of this report.

The first part of the subroutine calculates the layout of the wing panels. After the variables CSIDEP, SLPDIF, and WLEX are initialized for the first chordwise row of panels, the remaining calculations are performed within a double DO loop. The outer loop index, I, specifies the chordwise row and the inner loop index, K, the panel location in the I^{th} row. Within the inner loop the panel leading-edge and trailing-edge slopes and the corner coordinates are calculated and stored. The panels are numbered consecutively in chordwise rows beginning with panel number

one of row one adjacent to the wing root-chord at the leading edge. The sequence proceeds in increasing numbers to the trailing edge. Then back to the leading edge for the second chordwise row. The process continues until the last panel, numbered MS, is located adjacent to the wing tip at the trailing edge.

If a pylon is present (NPY = 1), the remainder of the subroutine calculates the panel leading-edge and trailing-edge slopes and the corner coordinates for the pylon source panels. The procedure used differs very little from the wing panel calculations. The value of PSLPDF, previously calculated by subroutine PLYOUT, is constant for all chordwise rows because no breaks in leading-edge or trailing-edge sweep angles may occur on the pylon. Also, the initial value of PLEX depends upon the pylon location; the index IP is tested for this purpose. The numbering convention associated with the pylon thickness panels is the same as that used for pylon constant u-velocity panels. It is described in section I-19. In describing corner coordinates for both wing and pylon panels, the corners closer to the respective root-chord are designated right corners; those farther from the root-chord are designated left corners.

I-33. SUBROUTINE THKOUT

Subroutine THKOUT prints the slopes of the wing and pylon thickness distributions which were read in as items 18 and 23 of the input data, see section 3.2.1. After the thickness slopes are printed, they are divided by PI and saved in a combined array, DZDX, for subsequent velocity calculations. A listing is presented in figure 1(x) of this report.

The following list defines most of the variables used in the subroutine:

DZDX	array containing wing and pylon thickness slopes after dividing by π
MPS	number of pylon thickness panels
MS	number of wing thickness panels
NCPS	number of thickness panels in a chordwise row on the pylon; input, item 22
NCWS	number of thickness panels in a chordwise row on the wing; input, item 17
NPY	index indicating whether a pylon is (NPY = 1) or is not (NPY = 0) present; input, item 4

PI — π

THETAL — array containing wing thickness slopes; input, item 18

THETPL — array containing pylon thickness slopes; input, item 23

I-34. SUBROUTINE THRCAL

Subroutine THRCAL calculates the store thrust at a given time. A listing of the subroutine in presented in figure 1(x) of this report.

The thrust force acts along the store longitudinal axis and is specified by a series of polynomials of the form

$$F_T = \sum_{n=1}^{6} a_n t^{n-1}$$

where F_T is the thrust in pounds at time t. The time history is specified by one of NTPOLY polynomials each of which is applicable for a range of t. The subroutine first determines which polynomial should be used for the given time value t. Once this is determined, the appropriate set of coefficients, a_n, is used in the above equation to calculate F_T. If t is greater than the end of the specified thrust time history, an error message is printed (see section 4) and the program stops.

The following table of definitions contains most of the variable names used in the subroutine. Section 3.2.1 should be consulted for the definition of a variable which is an input item.

FTHRUS — F_T; store thrust at time t

NTPOLY — input, item 38

T — t; time at which thrust force is to be calculated

TC(J,I) — input, item 40

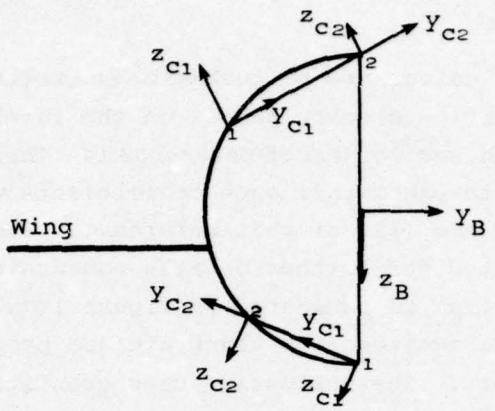
TEND(J) — input, item 39

I-35. SUBROUTINE VELBD

Subroutine VELBD calculates perturbation velocities at a given field point due to the constant u-velocity panels on the fuselage, according to the methods described in section 3.3 of reference 1. The subroutine is also used to obtain the single-panel influence coefficients which occur indirectly in equations (9), (10), and (11) of that reference. Section I-10 of this report should be consulted for further details concerning these coefficients. A listing of the subroutine is presented in figure 1(y), a flow chart in figure I-20, and a table equating the algebraic and program notation in Table I-16 of this report. The subroutine uses quantities calculated in subroutine BLYOUT which is described in section I-5. The wing coordinate system is shown in figure 6.

At the beginning of the subroutine the logical variable PYPNL is set equal to FALSE, the velocity totals UP, VP, and WP are initialized to zero, and the panel leading-edge and trailing-edge slopes, EM, are defined as zero. The remainder of the subroutine consists of a DO loop in which the influence functions for the four corners of the I^{th} panel are calculated and superposed. The superposition scheme is described in section 3.3.4 of reference 1 and that reference should be consulted for further details. In the corner numbering convention for fuselage panels corner one is the left front, corner two the right front, corner three the left rear, and corner four the right rear panel corner. The right corners are clockwise from the left when viewed from the rear.

At the beginning of the DO loop the corner influence function totals TU, TV, and TW are initialized to zero. Next, a test is performed to determine if the point at which velocities are to be calculated lies ahead of the panel leading edge; if so, all calculations for the I^{th} panel are skipped. If the point is not ahead of the leading edge, a test is performed next to determine in which fuselage quadrant the panel lies. As the flow chart indicates, two similar but distinct transformation and superposition procedures are followed, depending upon the panel location. In each procedure, the field point is first located relative to corner one in a corner coordinate system which is shown in the sketch below. Each panel corner has an associated coordinate system. The x_c axis, not shown in the sketch, is positive to the rear.



If corner one is in the upper left quadrant the sign of the y_c coordinate of the field point is reversed and subroutine VELO is called to calculate the influence of the corner on the image of the point with respect to the x_c,z_c plane. The sign of V, which is the returned influence function in the y_c direction, is then reversed. The functions U,V,W are resolved back into the wing system and superposed in the same manner as that used for wing panels with positive sweep (see fig. 4, ref. 1). Panels in the lower left quadrant are treated in the same manner as wing panels with negative sweep. No further coordinate change is necessary and subroutine VELO returns U,V, and W which are resolved into the wing system and superposed.

Next, in each procedure, the influence of the mirror image of corner one with respect to the aircraft vertical plane of symmetry is calculated. This is accomplished by the equivalent method of calculating the direct influence of corner one on the field point image. Following the call to subroutine VELO the sign of V is reversed in the superposition.

Corner one calculations are repeated in a similar manner for corners two, three, and four, but only corner one is detailed in the flow chart. After superposition of the four corner influence functions for the I^{th} panel is completed, a test is performed to determine if perturbation velocities are to be calculated. If II = IF, the influence coefficients for the I^{th} fuselage panel and the given control point are returned by the subroutine. If II $\neq$ IF, the coefficients are multiplied by the strength of the I^{th} panel, divided by π, to obtain the perturbation velocities induced by that panel at the given field point. These velocities are computed and summed for all fuselage panels.

I-36. SUBROUTINE VELCAL

Subroutine VELCAL calculates perturbation velocities at a given field point due to the fuselage and store source and doublet distributions, according to equation (I-32) of reference 1. A listing of the subroutine is presented in figure 1(aa) of this report. The fuselage coordinate system is shown in figure 5. The store coordinate system is shown in the sketch of section I-11.

The coordinates of the field point are given as formal parameters in the appropriate body coordinate system. The subroutine first transforms these coordinates into the VELCAL system by changing the sign of X and Z, and then into the polar coordinates XFIELD, RFIELD, and THETA.

The major part of the program consists of a DO loop within which the axial, radial, and tangential velocities due to the N sources and doublets are calculated and summed. A test is made to determine whether the field point is ahead of the Mach cone from the I^{th} source origin, TX(I). If so, all remaining perturbation velocities are equal to zero and no further calculations are performed within the loop. At the end of the subroutine, the velocities are resolved back into the directions of the body coordinate system.

The variables in the subroutine parameter list are:

T	array containing the source strengths
TC	array containing the doublet strengths
TX	array containing the x locations of the origins of the singularities; positive, measured aft from tip of nose
N	number of line sources and doublets
XP	x coordinate of field point in body system
Y	y coordinate of field point in body system
ZP	z coordinate of field point in body system
U1	u/V_∞ perturbation velocity at field point; body system
V1	v/V_∞ perturbation velocity at field point; body system
W1	w/V_∞ perturbation velocity at field point; body system

I-37. SUBROUTINE VELO

Subroutine VELO calculates the aerodynamic influence functions of a semi-infinite triangle associated with a constant u-velocity panel, as described in section II-2.1 of Appendix II of reference 1. The influence functions relate the panel singularity strength to the perturbation velocities induced by the traingle at a given point. They occur as the coefficients of $1/\pi(u_+/V_\infty)$ in equations (II-4) and (II-12) of reference 1. A listing of the subroutine is presented in figure 1(bb), a flow chart in figure I-21, and a table equating the algebraic and program notation in Table I-17 of this report. The coordinate system used by the subroutine is shown in figure 3 of reference 1.

At the beginning of the subroutine the quantity BETA is set equal to BETANU or to BETAOL depending on the values of INUMCH and PYPNL (section I-18 of this report should be referred to for further details concerning this choice). Next, the logical variable FELT is initialized to TRUE and a test is performed to determine if the field point is located ahead of the influencing triangle ($X \leq 0$). If so, the influence functions U,V,W are set to zero, FELT is set to FALSE, and control is returned to the calling program.

Next, the variable PYPNL is tested and, if the triangle is on the pylon, a transformation is performed which rotates the triangle into the VELO x,y plane. After the calculation of the logical variable INSIDE and other frequently used quantities, the remainder of the subroutine consists of four major sections in which the influence function terms, F1, F2, F4, F5, and F7, are calculated. Each section corresponds to a condition of the slope, EML, of the leading edge associated with the semi-infinite triangle. The subroutine requires that $EML \geq 0$ and this is accounted for in the VELO calling programs. The four leading-edge conditions are described fully, with accompanying sketches in section II-2.1 of reference 1. All equation numbers mentioned in the following paragraphs are from section II-2.1 of reference 1.

The first section of the subroutine corresponds to a subsonic leading edge, $BTSQ < EMLSQ$. Equation (II-5) is used if the point is inside the Mach cone from the origin, INSIDE = TRUE. If not, U,V, and W are set to zero. In this section as in the remaining ones, discontinuities in some of the equations may occur for certain field point locations. In such cases the affected influence function is set to zero. The quantities

YYEDGE and TLRNC, as well as Y and Z, are used to test the singularity locations.

If BETASQ = EMLSQ, the leading edge is a sonic leading edge. The equations used are the same as for the subsonic case except for the function F2, which is given by equation (II-7). If the point lies outside the Mach cone from the origin, the influence functions equal zero.

The third section of the subroutine is used if the triangle leading edge is supersonic, BTSQ > EMLSQ. Equations (II-5) and (II-9) calculate the terms of the influence functions if INSIDE = TRUE. If not, a second test is performed and equation (II-11) is used if the point is inside the Mach cone whose origin is on the leading edge at the field point y location. Otherwise the functions U,V,W are set to zero.

The fourth section of the subroutine is executed if the leading edge is unswept, EML = 0. For this special case the perturbation velocity equations are given by (II-12). If INSIDE = TRUE, the influence function terms are given by (II-5) and (II-13). Outside the Mach cone from the origin but inside the cone from the leading edge equation (II-14) is used. Otherwise, U,V,W are set to zero.

The last part of the subroutine calculates the function U,V,W from the component terms, in the case of a leading edge with positive sweep, using equation (II-4). If the triangle is located on the pylon, V and W are rotated back into pylon orientation.

I-38. SUBROUTINE VELOTH

Subroutine VELOTH calculates the aerodynamic influence functions of a semi-infinite triangle associated with a wing or pylon thickness panel, as described in section II-2.2 of Appendix II of reference 1. The influence functions relate the panel source strength to the perturbation velocities induced by the triangle at a given point. They occur as the coefficients of $1/\pi(\tan\theta)$ in equations (II-15) and (II-16) of reference 1. A listing of the subroutine is presented in figure 1(cc) of this report. The subroutine is very similar in logic to subroutine VELO which is described in detail in section I-37 and represented by a flow chart in figure I-21.

In subroutine VELOTH, three component terms, F1, F2, and F5, need to be calculated in order to determine the influence functions UTH,VTH, and WTH. Referring to section II-2.1 and II-2.2 of reference 1, function F1, F2, and F5 are specified in equation (II-5) for the case of a subsonic

leading edge, in equations (II-5) and II-7) for a sonic leading edge, and in equations (II-5), (II-9), and (II-11) for a supersonic leading edge. For the special case of an unswept leading edge (EML = 0), the general perturbation velocity equations are given by (II-16). If the given point lies inside the Mach cone from the origin of the triangle, function F1 is given by equation (II-13), function F5 by equation (II-5), and function F2 by equation (II-17). If the point lies outside the Mach cone from the origin but inside the Mach cone from the triangle leading edge at the field point y location, functions F1, F2, and F5 are given by equation (II-16). In all leading edge cases, the function F1, F2, and F5 are singular for certain field point locations. When this occurs, the affected influence function is set to zero.

A table equating the algebraic and program notation for subroutine VELO is presented in Table I-17 of this report. Almost all of the notation used in subroutine VELOTH is defined in this table. One should note that the influence functions, U,V,W and the point coordinates, YS,ZS, in VELO are named UTH,VTH,WTH,YTH, and ZTH, respectively in subroutine VELOTH.

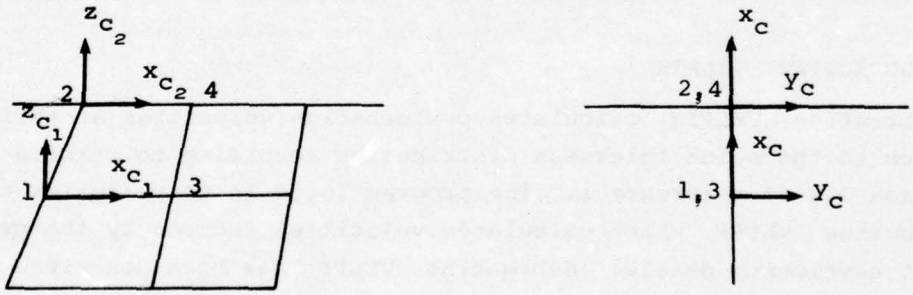
I-39. SUBROUTINE VELPP

Subroutine VELPP calculates perturbation velocities at a field point due to the constant u-velocity panels on the pylon according to methods described in section 3.3 of reference 1. The subroutine is also used to obtain the single panel influence coefficients which form the coefficients of $1/\pi(u_+/V_\infty)$ in equation (12) of reference 1. A listing of the subroutine is presented in figure 1(ee), a flow chart in figure I-22, and a table equating the algebraic and program notation in Table I-18 of this report. The quantities in the subroutine parameter list are contained in this table. The wing coordinate system is shown in figure 6.

At the beginning of the subroutine, the logical variable PYPNL is set equal to TRUE, indicating to subroutine VELO that calculations are to be performed for a pylon panel. The quantities YDIR and YIMG are calculated and the velocity totals, UP,VP,WP, are initialized to zero. The remainder of the subroutine consists of a DO loop in which the influence functions for the four corners of the I^{th} panel are calculated and superposed. The superposition scheme is described in section 3.3.4 of reference 1 and that reference should be consulted for further details.

The corner numbering convention for pylon panels associates corners one and two with the leading edge left and right corners, respectively; corners three and four with the trailing edge left and right corners, respectively. The left corners are those farther from the pylon root chord.

At the beginning of the DO loop the influence function totals are initialized to zero and the leading-edge slope, EM1, and trailing-edge slope, EM2, are defined. Next, a test of the sign of EM1 is performed and two distinct transformation and superposition procedures are followed depending on the results of this test, as the flow chart indicates. In each procedure a test is performed first to determine if the field point lies ahead of the most forward leading-edge corner; if so, all calculations for the I^{th} panel are skipped. If the point is not ahead of the leading edge, the point is located relative to corner one in a corner coordinate system which is illustrated in the sketch below. Each panel corner has an associated coordinate system.



For a panel with swept back leading edge, EM1 ≥ 0 the superposition scheme is the same as for a wing panel with positive sweep (see figure 4, reference 1). The field point z_c coordinate is reversed and subroutine VELO is called to calculate the influence of corner one on the image of the point with respect to the x_c, y_c plane. The sign of W is then reversed. The functions U,V,W are resolved back into the wing system and superposed. If the panel leading edge is swept forward, superposition is the same as for wing panels with negative sweep. The sign of EM1 is reversed and subroutine VELO returns U,V,W which are resolved into the wing system and superposed. It should be noted that the subroutine code combines superposition and transformation steps and that the final sign changes of U and W are made at the end of the DO loop.

The next calculations are omitted if the pylon is located under the fuselage centerline. Otherwise, the influence of the mirror image of corner one with respect to the vertical plane of symmetry is calculated. This is accomplished by the equivalent method of calculating the direct influence of corner one on the image of the field point and then reversing the sign of V.

Corner one calculations are repeated in a similar manner for corners two, three, and four, but only corner one is described in detail in the flow chart. After superposition of the four corner influence functions for the I^{th} panel is completed, a test is performed to determine if perturbation velocities are to be calculated. If II = IF, the influence coefficient for the I^{th} pylon panel and the given control point is returned by the subroutine. If II $\neq$ IF, the coefficients are multiplied by the strength of the I^{th} panel, divided by π, to obtain the perturbation velocities induced by that panel at the given field point. These velocities are computed and summed for all pylon constant u-velocity panels.

I-40. SUBROUTINE VELPTH

Subroutine VELPTH calculates perturbation velocities at a field point due to the pylon thickness distribution according to methods described in section 3.3 of reference 1. The program logic is very similar to that of subroutine VELPP which calculates velocities induced by the pylon constant u-velocity panels. Subroutine VELPP has been described in detail in section I-39 and is represented by a flow chart in figure I-22 of this report. Only those details, therefore, in which subroutine VELPTH differs from subroutine VELPP are included in this description. A listing of the subroutine is presented in figure 1(gg) and a table equating the algebraic and program notation in Table I-19. The quantities in the subroutine parameter list are contained in this table.

The coordinates of the field point are given as formal parameters in the wing coordinate system. The point is located relative to each of the four panel corners using the same transformation scheme as in VELPP. However, the corner coordinate arrays which define the pylon thickness panels and which have been previously calculated by subroutine THKLYT (see section I-32) are used in the transformations. Subroutine VELOTH is called to calculate the corner influence functions, U,V,W, which are then superposed in the same manner as in VELPP.

Finally, the subroutine calculates perturbation velocities only and is not used to obtain single panel influence coefficients. Thus, after the calculation of the influence coefficients for the I^{th} panel is completed, no test is performed to determine if II = IF. The influence coefficients are multiplied by DZDX(I) to obtain perturbation velocities induced by the I^{th} panel at the given point. These velocities are calculated and summed for all pylon thickness panels.

I-41. SUBROUTINE VELWP

Subroutine VELWP calculates perturbation velocities induced at a given field point by the constant u-velocity panels on the wing according to methods described in section 3.3 of reference 1. The subroutine is also used to obtain single panel influence coefficients which occur in equation (12) of reference 1 as the coefficients of $1/\pi(u_+/V_\infty)$. A listing of the subroutine is presented in figure 1(hh), a flow chart in figure I-23, and a table equating the algebraic and program notation in Table I-20 of this report. The quantities in the subroutine parameter list are contained in this table. The wing coordinate system is shown in figure 6.

At the beginning of the subroutine, the logical variable PYPNL is set equal to FALSE indicating to subroutine VELO that calculations are for a horizontal rather than a pylon panel. After the velocity totals, UP,VP,WP, are initialized to zero, the remainder of the subroutine consists of a DO loop within which the influence functions for the four corners of the I^{th} panel are calculated and superposed. The superposition scheme is described in section 3.3.4 of reference 1 and that reference should be consulted for further details. The corner numbering convention for wing panels is shown in figure 4 of reference 1.

At the beginning of the DO loop the influence totals, TU,TV,and TW are initialized to zero and the leading-edge slope, EM1, and trailing-edge slope, EM2, are defined. Next, a test of the sign of EM1 is performed and two distinct transformation and superposition procedures are followed depending on the results of this test, as the flow chart indicates. In each procedure the point is first located relative to corner one in the corner coordinate system described in section I-39 of this report. The z_c location of the field point is constant for all wing panels and is defined outside the DO loop. If EM1 ≥ 0, the sign of the y_c coordinate of the point is reversed and subroutine VELO is called to calculate

the influence of corner one on the image of the point with respect to the x_c, z_c plane. The sign of V is then reversed. The functions U,V,W are resolved back into the wing system and superposed. If EM1 < 0, the sign of EM1 is reversed. After the call to VELO, the functions U,V,W are resolved back into the wing system and superposed. In the subroutine code transformation and superposition steps are combined and the final sign change of U and W are made at the end of the DO loop.

Next, in each procedure, the influence of the image of corner one with respect to the aircraft vertical plane of symmetry is calculated. This is accomplished by the equivalent method of calculating the direct influence of corner one on the field point image and then reversing the sign of V.

Corner one calculations are repeated in a similar manner for corners two, three, and four, but only corner one is detailed in the flow chart. After superposition of the four corner influence functions for the I^{th} panel is completed, a test is performed to determine if perturbation velocities are to be calculated. If II = IF, the influence coefficient for the II^{th} panel and the given control point is returned by the subroutine. If II ≠ IF, the coefficients are multiplied by the strength of the I^{th} panel, divided by π, to obtain the perturbation velocities induced by that panel at the given field point. These velocities are computed and summed for all pylon constant u-velocity panels.

I-42. SUBROUTINE VELWTH

Subroutine VELWTH calculates perturbation velocities at a given field point due to the wing thickness distribution, according to methods described in section 3.3 of reference 1. The program logic is very similar to that of the corresponding subroutine VELWP, which calculates velocities induced by the wing constant u-velocity panels. Subroutine VELWP has been described in detail in section I-41 and is represented by a flow chart in figure I-23 of this report. Only those details, therefore, in which subroutine VELWTH differs from subroutine VELWP are included in this description. A listing of the subroutine is presented in figure 1(jj) and a table equating the algebraic and program notation in Table I-21. The quantities in the subroutine parameter list are contained in this table.

The coordinates of the field point are given as formal parameters in the wing coordinate system. The point is located relative to each of the four corners of the trapezoidal shaped thickness panels using the same transformation schemes as in VELWP. However, the corner coordinate arrays which define the wing thickness panels and which have been previously calculated by subroutine THKLYT (see section I-32) are used in the transformations. Subroutine VELOTH is called to calculate the corner influence functions, U,V,W, which are then superposed in the same manner as in VELWP.

Finally, the subroutine calculates perturbation velocities only and is not used to obtain single panel influence coefficients. Thus, after the calculation of the influence coefficients for the I^{th} panel is completed, no test is performed to determine if II = IF. The influence coefficients are multiplied by DZDX(I) to obtain the perturbation velocities induced by the I^{th} panel at the given field point. These velocities are calculated and summed for all wing thickness panels.

I-43. SUBROUTINE WITHIN

Subroutine WITHIN reads in the wing thickness data, input items 17 and 18. A listing of the subroutine is presented in figure 1(kk) of this report.

The number of panels in a chordwise row, NCWS, is first read in along with an index, NUNIS, which indicates whether the thickness distribution is similar at all spanwise stations. If it is similar, NUNIS = 1, the values of $\tan\theta$ are read in for the first row of panels and then the values of $\tan\theta$ for the other rows are set equal to those of the first row. If the thickness distribution is not similar, NUNIP = 0, the values of θ for all rows are read in.

After the thickness slopes are read in, an input error check is performed. If any of the $\tan\theta$ values is zero or negative at the wing leading edge, an error message is printed (see section 4) and the program halts.

The following table contains definitions of most of the variable names used in the subroutine. Section 3.2.1 should be referred to for the definition of a variable that is an input quantity.

JLE	index of first wing thickness panel in a chordwise row
MS	number of thickness panels on wing; MSW*NCWS
MSW	input, item 13
NCWS	input, item 17
NUNIS	input, item 17
THETAL(I)	input, item 18

I-44. SUBROUTINE WLYOUT

Subroutine WLYOUT calculates quantities which characterize the constant u-velocity panels on the wing. An example of a wing trapezoidal panel is shown in figure 2 of reference 1. Similar routines PLYOUT and BLYOUT calculate these quantities for the pylon and fuselage panels, respectively. The arrangement of variables in any single coordinate array is wing panels first, pylon panels second, and fuselage panels last. All coordinates are in the wing coordinate system which is shown in figure 6. A listing of the subroutine is presented in figure 1(ll), a flow chart in figure I-24, and a table equating the algebraic and program notation in Table I-22 of this report.

As indicated by the flow chart, the first part of the subroutine tests whether the wing leading-edge and trailing-edge sweep angles are constant at all spanwise Y-stations. If breaks in sweep occur, the indicator LVSWP is set equal to one; otherwise, LVSWP = 0 and the quantity SLPDIF, the difference between leading-edge and trailing-edge slopes for a chordwise row, is computed outside the main DO loop.

After initializing the quantities CSIDEP and WLEX, the remainder of the subroutine consists of a double DO loop. The outer loop index, I, controls the chordwise row; the inner loop index, K, specifies the panel location in the I^{th} row. Within the inner loop, the panel leading-edge and trailing-edge slopes, the corner coordinates, and the control point coordinates are calculated and stored. The convention used in numbering the NPANLS wing constant u-velocity panels is the same as that used for the wing thickness panels. Section I-32 should be referred to for details. In the subroutine notation, the right-hand side of the panel is the one closer to the root-chord.

TABLE I-1

DICTIONARY OF NOTATION IN MAIN PROGRAM

The following list presents most of the variable names used in the main program. Those which appear in common statements or parameter lists of subroutines but are not used in the main program are not included. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 4. If the symbol or equation number is from another reference, the reference number is given. Where a variable is an input quantity, it is so identified and section 3.2.1 of this report should be referred to for the definition. The x_B, y_B, z_B coordinate system is shown in figure 5 of this report.

PROGRAM NOTATION	ALGEBRAIC NOTATION
ACCG	g, 32.174 feet/second
ALFAC	α_f, degrees; input, item 3
ALFACR	α_f, radians
ALPHAL(N)	tan α_ℓ; input, item 16
BETA	$\beta = \sqrt{M_\infty^2 - 1}$
BETASQ	β^2
CA	C_A; input, item 35
CAPG	$\cos(\alpha_f + \gamma_f)$
CDC	c_{d_C}; input, item 35
CENTER	logical variable which is TRUE for a pylon under the fuselage centerline
CLALPH	lift-curve slope of tail panels; input, item 37
CLLEM	$(C_\ell)_E$, equations (62) and (I-30) or equations (63) and (I-52)
CLMBY	$(C_m)_{BY}$, equation (44)
CLMCF	$(C_m)_{CF}$, equation (53)
CLMEM	$(C_m)_E$, equation (60)

Table I-1.- Continued.

CLMSB	$(C_m)_{SB}$, equation (48)
CLNBY	$(C_n)_{BY}$, equation (45)
CLNCF	$(C_n)_{CF}$, equation (54)
CLNEM	$(C_n)_E$, equation (61)
CLNSB	$(C_n)_{SB}$, equation (49)
CNBY	$(C_N)_{BY}$, equation (42)
CNCF	$(C_N)_{CF}$, equation (51)
CNEM	$(C_N)_E$, equation (58)
CNORM	total C_N
CNSB	$(C_N)_{SB}$, equation (46)
COEF(I,J)	coefficients of polynomials specifying separated store shape; input, item 34
CPITCH	total C_m
CROLL	total C_ℓ
CSIBCR(N)	cos[SIBCR(N)]
CSIDE	total C_Y
CYAW	total C_n
CYBY	$(C_Y)_{BY}$, equation (43)
CYCF	$(C_Y)_{CF}$, equation (52)
CYEM	$(C_Y)_E$, equation (59)
CYSB	$(C_Y)_{SB}$, equation (47)
DC(I,J)	[A], equation (28)
DDTIME	input value of integration interval; set equal to DTIME of item 42

Table I-1.- Continued.

DELTP(N)	$1/\pi(u_+/V_\infty)$; equation 15, reference 1
DELX	length of body segment used in force calculation
DTIME	current value of integration interval
DTOR	degrees to radians conversion factor, 1/57.29578
DVAR(N) N = 1,2,...12	$\ddot{\xi},\ddot{\eta},\ddot{\zeta},\dot{p},\dot{q},\dot{r},\dot{\xi},\dot{\eta},\dot{\zeta},\dot{\Psi},\dot{\theta},\dot{\Phi}$ respectively
EDRDX(I)	da/dx_s at the midpoint of the i^{th} segment of the separated store
ERAD(I)	a at the midpoint of the i^{th} segment of the separated store
ESTLGC	ℓ_s of separated store
ESTRMX	a_{max} of separated store
EXST(I)	x_s location of the i^{th} half segment on the separated store
FINSS	tail fin semispan; input, item 37
FIXX	I_{xx}; input, item 31
FIXY	I_{xy}; input, item 31
FIXZ	I_{xz}; input, item 31
FIYY	I_{yy}; input, item 31
FIYZ	I_{yz}; input, item 31
FIZZ	I_{zz}; input, item 31
FLTHC	fuselage length; input, item 5
FMACH	M_∞; input, item 3
FMCHSQ	M_∞^2
FTHRUS	thrust force, F_T, reference 1
GAMF	fuselage flight path angle, γ_f; input, item 3
GXX,GYY,GZZ	g_x,g_y,g_z, equation (67)
IPLNR	empennage type; input, item 36

Table I-1.- Continued.

MP	number of u-velocity panels on pylon, MP = NCP*MSP
MPS	number of pylon thickness panels, MPS = NCPS*MSP
MS	number of wing thickness panels, MS = NCWS*MSW
MSF	input, item 36
MSP	input, item 20
MSW	input, item 13
NBD	NBD = NBDCR1 + NBDCR2 which were input in item 9
NBIP	number of u-velocity panels on fuselage; NBIP = NBD*NCWB
NCP	input, item 20
NCPS	input, item 22
NCW	input, item 13
NCWB	input, item 9
NCWS	input, item 17
NDAMP	input, item 30
NDIFEQ	control index used in subroutine ADAMS
NEJECT	number of store being separated; input, item 30
NEJSTR	subscript associated with separated store; $1 \leq$ NEJSTR $\leq$ NSTRS
NEMP	input, item 30
NEQ	number of differential equations being integrated by subroutine ADAMS; NEQ = 12
NFPOLY	input, item 6
NFU	input, item 4
NGAM	input, item 30
NHSEG	number of points along store axis where velocity field is to be calculated; NHSEG = 2*NSEG + 1

Table I-1.- Continued.

NHSEGO	number of points along store axis ahead of separation point where velocity field is to be calculated; NHSEGO = 2*NSEGXO + 1
NPANLS	number of u-velocity panels on left wing panel; NCW*MSW
NPOLY	input, item 30
NPTOT	NPANLS + MP + NBIP
NPTOTP	NPTOT + 1
NPY	input, item 4
NROLL	input, item 30
NSEG	input, item 30
NSEGXO	input, item 30
NSTRS	input, item 4
NTHP	MS + MPS
NTHRUS	input, item 30
NTPOLY	input, item 38
NUMSTR(I)	number associated with i^{th} store; input, item 24
N1P	NPANLS + 1
N2	NPANLS + MP
N2P	N2 + 1
PHIROL	input, item 37
QSREF	$q_{\infty_s} S_R$
QSREFL	$q_{\infty_s} S_R \ell_R$
QSTORE	q_{∞_s}, equation (69)
RADAV	input, item 37
REFL	ℓ_R, equation (72)
RHO	ρ_∞; input, item 3

Table I-1.- Continued.

SAPG	$\sin(\alpha_f + \gamma_f)$
SIBCR(I)	incidence angle of i^{th} store relative to fuselage axis, radians; SIC(I)*DTOR
SIC(I)	incidence angle of the i^{th} store; input, item 24
SLTHC(I)	length of i^{th} store; input, item 24
SMASS	m, mass of separated store; input, item 31
SREF	S_R, equation (70)
SRMAX(I)	maximum radius of i^{th} store; input, item 24
SSIBCR(I)	sin[SIBCR(I)]
SSPAN	input, item 12
TC(I,J)	input, item 40
TEND(I)	input, item 39
TIME	t, current value of time
TIMEF	final time; input, item 42
TIMEI	initial time; input, item 42
TIPY	exposed wing span
UU	$\dot{x}_o$
VAR(N), N = 1,2,...12	$\dot{\xi},\dot{\eta},\dot{\zeta},p,q,r,\xi\ \eta\ \zeta\ \Psi\ \Theta\ \Phi$ respectively
VINF	V_∞; input, item 3
VSTORE	V_{∞_s}, equation (41)
VV	$\dot{y}_o$
VXZERO	input, item 41
VYZERO	input, item 41
VZZERO	input, item 41
WW	$\dot{z}_o$

Table I-1. Concluded.

XBAR	$\bar{x}$; input, item 32
XBASEI	ξ coordinate of separated store base at t = 0
XBSO(I)	x_B coordinate of tip of nose of i^{th} store
XBWOC	$x_{B_{item\ 11}}$ coordinate of wing root-chord leading edge; input,
XCGI	ξ coordinate of separated store moment center at t = 0
XEND(N)	end points of polynomials specifying separated store shape; input, item 33
XMOM	$x_{s,m}$; input, item 32
XNOSEI	ξ coordinate of tip of separated store nose at t = 0
XTAIL	input, item 37
XWSOC(I)	x coordinate of tip of nose of i^{th} store measured from wing root-chord leading edge
YBAR	$\bar{y}$; input,item 32
YBASEI	η coordinate of separated store base at t = 0
YBSO(I)	y_B coordinate of i^{th} store
YCGI	η coordinate of separated store moment center at t = 0
YNOSEI	η coordinate of tip of separated store nose at t = 0
YPL	y_B coordinate of pylon root-chord leading edge
YWSO(I)	y location of i^{th} store in wing coordinate system
ZBAR	$\bar{z}$; input, item 32
ZBASEI	ζ coordinate of separated store base at t = 0
ZBSO(I)	z_B coordinate of tip of nose of i^{th} store
ZBWO	$z_{B_{item\ 11}}$ coordinate of wing root-chord leading edge; input,
ZCGI	ζ coordinate of separated store moment center at t = 0
ZNOSEI	ζ coordinate of tip of separated store nose at t = 0
ZWSO(I)	z location of tip of i^{th} store nose measured relative to wing root-chord leading edge

TABLE I-2

DICTIONARY OF NOTATION IN SUBROUTINE BDYGEN

The following list presents most of the variable names used in subroutine BDYGEN. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 1. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition.

PROGRAM NOTATION	ALGEBRAIC NOTATION
A(J)	coefficient of $K_{d,J-1}$ in equation (I-28), or of K_{J-1} in equation (I-17), for a given control point.
ALPHA	angle measured between the body centerline and the free-stream direction
BETA	$\sqrt{M_\infty^2 - 1}$
DEL	LBODY/N
DRDX(I)	slope of body surface at I^{th} control point
LBODY	length of body for which source and doublet distribution is to be calculated
MACH	M_∞; input, item 3
N	NXBODY - 1
NSEG	number of polynomials used to specify body shape
NXBODY	number of body definition points
RADIUS	maximum radius of body
RBODY(I)	radius of body at I^{th} body definition point
RF(I)	radius of body at I^{th} control point
RFIELD	radius of body at a control point
ROL	body radius at a specified axis point, made dimensionless by body length
RPBODY(I)	slope of body surface at I^{th} body definition point

Table I-2.- Concluded.

SCOEF	array containing coefficients of polynomials used to specify body shape
SLOPE	slope of body surface at a control point
SUM	quantity represented by summation expression in right-hand side of the first of equations (I-17) and (I-28)
T	array containing source strengths; T(J) is the quantity K_{J-1} calculated by equations (I-14) and (I-17)
TC	array containing doublet strengths; TC(J) is the quantity $K_{d,J-1}$ calculated by equations (I-27) and (I-28)
TX	array containing x locations of origins of conical line sources and doublets; positive, measured from tip of nose
U	quantities from equations (I-14), (I-17), (I-27), and (I-28), calculated by subroutines SOURCE and DOUBLT
V	terms from equations (I-14), (I-17), (I-27), and (I-28), calculated by subroutines SOURCE and DOUBLT
XBODY(I)	x location of I^{th} body definition point; positive back, measured from tip of nose
XEND	array containing x locations of end points of polynomial sections specifying body shape, made dimensionless by body length
XF(I)	x location of I^{th} control point; positive, measured from tip of nose
XFIELD	x location of a control point; positive, measured from tip of nose
XOL	x location of a specified axis point, made dimensionless by body length

TABLE I-3

DICTIONARY OF NOTATION IN SUBROUTINE BLYOUT

The following list presents most of the variable names used in subroutine BLYOUT. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 1. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition. The x_W, y_W, z_W coordinate system is shown in figure 6.

PROGRAM NOTATION	ALGEBRAIC NOTATION
ANG	fuselage panel orientation angle relative to x_B, y_B plane; negative measured in counterclockwise rotation from positive y_B direction
ANGW	$\sin^{-1}$(ZBWO/FRMAX)
BODYPL	input, item 10
CPY	y coordinate of a fuselage control point in panel coordinate system (see section I-5)
CS	cos(THT)
CST2(J)	cos(ANG_J); ANG_J is the value of ANG associated with J^{th} panel
DX	panel length; BODYPL/NCWB
FAC	0.95; constant used to locate XCPT(J) at 95 percent of chord containing panel centroid
FRMAX	input, item 5
IL,IU	initial and final values, respectively,of loop index over ring of body interference panels
ITIMES	control constant: ITIMES = 1, panel is above non-tangent wing; ITIMES = 2, wing is tangent, or panel is below wing
NBD	number of panels in a ring of body interference panels; NBDCR1 + NBDCR2
NBDCR1	input, item 9
NBDCR2	input, item 9

Table I-3.- Continued.

NBIP	total number of body interference panels; NCWB*NBD
NCWB	input, item 9
N2	number of constant u-velocity panels on wing and pylon
PI	π
PI2	$\pi/2$
RTOD	radians to degrees conversion factor; $180/\pi$
SN	sin(THT)
SNT2(J)	$\sin(ANG_J)$; ANG_J is value of ANG associated with J^{th} body panel
THT	polar angle in cross-sectional plane, defining the origin of panel coordinate system relative to fuselage center; positive in clockwise rotation from negative y_B axis
THTI(J)	$(\pi - THT_J)$*RTOD; THT_J is value of THT associated with J^{th} panel
THTS	central angle in cross-sectional plane, defined by radial lines drawn from center of fuselage to panel edges (see section I-5)
WBIP	y location of right edge of panel, in panel coordinate system (see section I-5)
XCPT(I)	x_W coordinate of I^{th} control point
XFBIP	x_W location of front edge of panels in a ring of body interference panels
XLB(I)	x_W coordinate of left rear corner of I^{th} constant u-velocity panel
XLF(I)	x_W coordinate of left front corner of I^{th} constant u-velocity panel
XRB(I)	x_W coordinate of right rear corner of I^{th} constant u-velocity panel
XRF(I)	x_W coordinate of right front corner of I^{th} constant u-velocity panel
YCPT(I)	y_W coordinate of I^{th} control point
YLC(I)	y_W coordinate of left edge of I^{th} constant u-velocity panel

Table I-3.- Concluded.

YRC(I)	y_W coordinate of right edge of I^{th} constant u-velocity panel
ZBWO	input, item 11
ZCPT(I)	z_W coordinate of I^{th} control point
ZLC(I)	z_W coordinate of left edge of I^{th} constant u-velocity panel
ZRC(I)	z_W coordinate of right edge of I^{th} constant u-velocity panel

TABLE I-4

DICTIONARY OF NOTATION IN SUBROUTINE DPCOEF

The following list presents most of the variable names used in subroutine DPCOEF. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 1. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition.

PROGRAM NOTATION	ALGEBRAIC NOTATION
CST2(I)	cosine of orientation angle of I^{th} fuselage panel (see section I-5)
FVN(J,K)	aerodynamic influence coefficient representing the influence of the K^{th} panel at the J^{th} control point
II,IF	index of the influencing panel; II = IF
NFU	input, item 4; NFU = 0, no fuselage
NPANLS	number of constant u-velocity panels on left wing panel.
NPTOT	total number of constant u-velocity panels
NPY	input, item 4; NPY = 0, no pylon
N1P	NPANLS + 1
N2	number of constant u-velocity panels on wing and pylon
N2P	N2 + 1
SNT2(I)	sine of orientation angle of I^{th} fuselage panel (see section I-5)
VP	influence coefficient associated with function F_v of equation (12)
WP	influence coefficient associated with function F_w of equation (12)
WW	influence coefficient normal to the body surface at a specified fuselage control point
XCPT(I)	x_w coordinate of I^{th} control point
YCPT(I)	y_w coordinate of I^{th} control point
ZCPT(I)	z_w coordinate of I^{th} control point

TABLE I-5

DICTIONARY OF NOTATION IN SUBROUTINE DPRHS

The following list presents most of the variable names used in subroutine DPRHS. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 1. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition. The store coordinate system is shown in the sketch in section I-11, the fuselage coordinate system in figure 5, and the wing coordinate system in figure 6.

PROGRAM NOTATION	ALGEBRAIC NOTATION
ALFACR	α_f, radians; α_f in degrees was input in item 3
ALPHAL(J)	$\tan(\alpha_\ell)$; input, item 16
CENTER	logical variable, TRUE, if pylon is located under the fuselage centerline
CIR	array containing the right-hand sides of equations (9), (10), and (11)
CSIBCR(K)	cos[SIC(K)]; SIC(K) was input, item 24
CST2(I)	cosine of the orientation angle of I^{th} fuselage interference panel
FDS	array containing the strengths of the fuselage doublet distribution
FSS	array containing the strengths of the fuselage source distribution
FXL	array containing the x positions of the fuselage singularity origins; positive, measured from tip of nose
II,IF	initial and final values, respectively, of thickness panel index
MS	number of wing thickness panels
NFSOR	input, item 9
NFU	input, item 4; NFU = 0, no fuselage
NPANLS	number of constant u-velocity panels on wing
NPTOT	number of constant u-velocity panels on wing, pylon, and fuselage

Table I-5.- Continued.

NPY	input, item 4; NPY = 0, no pylon
NSSOR(K)	MSOR associated with K^{th} store; MSOR was input in item 26
NSTRS	input, item 4; NSTRS limited to one in present program
NTHP	number of thickness panels on wing and pylon
N1P	NPANLS + 1
N2	number of constant u-velocity panels on wing and pylon
N2P	N2 + 1
SDS	array containing the strengths of the doublet distribution for the K^{th} store
SLTHC(K)	input, item 24
SNT2(I)	sine of the orientation angle of the I^{th} fuselage interference panel
SSIBCR(K)	sin[SIC(K)]; SIC(K) was input, item 24
SSS	array containing the strengths of the source distribution for the K^{th} store
SXL	array containing the x positions of the singularity origins for the K^{th} store; positive, measured from tip of store nose
UEI(I)	$u_{W_{i,\nu}}/V_\infty$ at $\nu = I^{th}$ control point
UP	u/V_∞ perturbation velocity at a control point due to pylon or wing thickness
UQ	u/V_∞ perturbation velocity at a control point due to fuselage or store body singularities
US	store induced u/V_∞ perturbation velocity, not corrected for store incidence
VEI(I)	$v_{W_{i,\nu}}/V_\infty$ at the $\nu = I^{th}$ control point
VP	v/V_∞ perturbation velocity at a control point due to wing or pylon thickness
VQ	v/V_∞ perturbation velocity at a control point due to fuselage or store body singularities

Table I-5.- Concluded.

WEI(I)	$w_{W_i, \nu}/V_\infty$ at the $\nu = I^{th}$ control point
WP	w/V_∞ perturbation velocity at a control point due to wing or pylon thickness
WQ	w/V_∞ perturbation velocity at a control point due to fuselage or store body singularities
WS	store induced w/V_∞ perturbation velocity, not corrected for store incidence
WW	perturbation velocity normal to fuselage surface at a fuselage control point
XBWOC	input, item 11
XCPT(I)	x_W coordinate of I^{th} control point
XVC	x_B coordinate of a control point
XVCS	x coordinate of a control point in store coordinate system
XWSOC(K)	x_W coordinate of tip of nose of K^{th} store
YCPT(I)	y_W coordinate of I^{th} control point
YVCS	y coordinate of a control point in the store coordinate system
YWSO(K)	y_W coordinate of tip of nose of K^{th} store
ZBWO	input, item 11
ZCPT(I)	z_W coordinate of I^{th} control point
ZVC	z_B coordinate of a control point
ZVCS	z coordinate of a control point in the store coordinate system
ZWSO(K)	z_W coordinate of tip of nose of K^{th} store

TABLE I-6

DICTIONARY OF NOTATION IN SUBROUTINE FUSEIO

The following list presents most of the variable names used in subroutine FUSEIO. Where possible a variable name is identified by a symbol in the list of symbols. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition.

PROGRAM NOTATION	ALGEBRAIC NOTATION
ALFACR	α_f, radians; α_f in degrees was input in item 3
BODYPL	input, item 10
FCOEF	input, item 8
FDS	array containing the strengths of the fuselage doublet distribution
FLTHC	input, item 5
FRMAX	input, item 5
FSS	array containing the strengths of the fuselage source distribution
FXEND	input, item 7
FXL	array containing the x positions of the fuselage sources; positive, measured from tip of nose
NBD	number of constant u-velocity panels in a ring on fuselage; NBDCR1 + NBDCR2
NBDCR1	input, item 9
NBDCR2	input, item 9
NCWB	input, item 9
NFPOLY	input, item 6
NFSOR	input, item 9
NXBODY	NFSOR + 1

TABLE I-7

DICTIONARY OF NOTATION IN SUBROUTINE NUMACH

The following list presents most of the variable names used in subroutine NUMACH. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 1. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition. The x_W, y_W, z_W coordinate system is shown in figure 6.

PROGRAM NOTATION	ALGEBRAIC NOTATION
ANGNU1	ν_1, degrees, equation (26)
ANGNU2	ν_2, degrees, equation (25)
BETA	$(M_\infty^2 - 1)^{1/2}$
BETANU	$(M_\ell^2 - 1)^{1/2}$
CHRD	length of local wing chord at field point y_W location
DELTW	change in downward velocity induced by wing thickness at points in XT4 traverse
DELX	(XT4 - XT3)/20
DELX1	x interval size used in velocity traverse in determining XT3 and XT4
FDS	array containing fuselage doublet strengths
FNUMCH	M_ℓ, equation (28)
FSS	array containing fuselage source strengths
FXL	array containing x locations of fuselage singularity origins; positive, measured from tip of nose
IF	final index value of panels included in sum of induced velocities representing a particular aircraft component
II	initial index value of panels included in sum of induced velocities representing a particular aircraft component
INUMCH	control constant; INUMCH = 0, velocity calculations performed using M_∞; INUMCH = 1, velocity calculations performed using M_ℓ
MS	number of thickness panels on the wing

Table I-7.- Continued.

MSWP	MSW + 1; MSW was input item 13
NCW	input, item 13
NFSOR	input, item 9
NFU	input, item 4; NFU = 0, no fuselage
NORDWS	3; degree of least squares polynomials used to smooth velocities
NPANLS	number of wing constant u-velocity panels
NPTOT	number of constant u-velocity panels on wing, pylon, and fuselage
NPTSSM	number of velocity data points included in least squares fit
N2P	N2 + 1; N2 is the number of constant u-velocity panels on wing and pylon
S	scratch array required by subroutine FLSQFY
TANNU	tan $\Delta\nu$, equation (23)
TNU	$\tan^{-1}$(TANNU), degrees
UP	u/V_∞ perturbation velocity at a point due to an aircraft component
UTU	sum of u/V_∞ perturbation velocities at a point due to wing-fuselage aircraft components
VEL	array containing the three velocity component vectors to be fit by least squares polynomial
VP	v/V_∞ perturbation velocity at a point due to an aircraft component
VTV	sum of v/V_∞ perturbation velocities at a point due to wing-fuselage aircraft components
W	array of positive weights used in calculating least squares fit; $w_i = 1$, $i = 1,\ldots,$NPTSSM
WLEX	x_W coordinate of wing leading edge at field point y_W location
WP	w/V_∞ perturbation velocity at a point due to an aircraft component
WSAVE	test value of w/V_∞ velocity at a traverse point due to wing thickness; used in determining XT3 and XT4

Table I-7.- Continued.

WSCOE	array containing three column vectors of least squares polynomial coefficients, in row order of increasing degree
WTEX	$x_{W_{y_W}}$ coordinate of wing trailing edge at field point location
WTW	sum of w/V_∞ perturbation velocity at a point due to wing-fuselage aircraft components
X	x_W coordinate of point at which velocities due to wing thickness are calculated in determining XT3 and XT4
XBWOC	input, item 11
XLB(I)	x_W coordinate of left rear corner of I^{th} constant u-velocity panel
XLF(I)	x_W coordinate of left front corner of I^{th} constant u-velocity panel
XRB(I)	x_W coordinate of right rear corner of I^{th} constant u-velocity panel
XRF(I)	x_W coordinate of right front corner of I^{th} constant u-velocity panel
XSAVE	x_W location of traverse point at which perturbation velocity WSAVE occurs
XSHFTW	midpoint of XT3, XT4 range based on local Mach number
XSM	vector of x coordinates of data points to be fit by least squares polynomial
XT3	x_W location of point on traverse at which first non-zero w/V_∞ velocity due to wing thickness is calculated
XT4	x_W location of point on traverse at which magnitude of decrease in w/V_∞ velocity due to wing thickness is at least 50 percent; XT4 $\leq$ XT3 - 0.7*CHRD
Y	y_W location of field point at which flow field is to be calculated
YLC(I)	y_W coordinate of left edge of I^{th} constant u-velocity panel
YNU	p, equation (27)

Table I-7.- Concluded.

Z	z_W location of field point at which flow field is to be calculated
ZB	z in fuselage coordinate system
ZBWO	input, item 11

TABLE I-8

DICTIONARY OF NOTATION IN SUBROUTINE PLYOUT

The following list presents most of the variable names used in subroutine PLYOUT. Where possible a variable name is identified by a symbol in the list of symbols. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition. The x_W, y_W, z_W coordinate system is shown in figure 6. If not specified, the panels referred to in this dictionary are constant u-velocity panels located on the pylon.

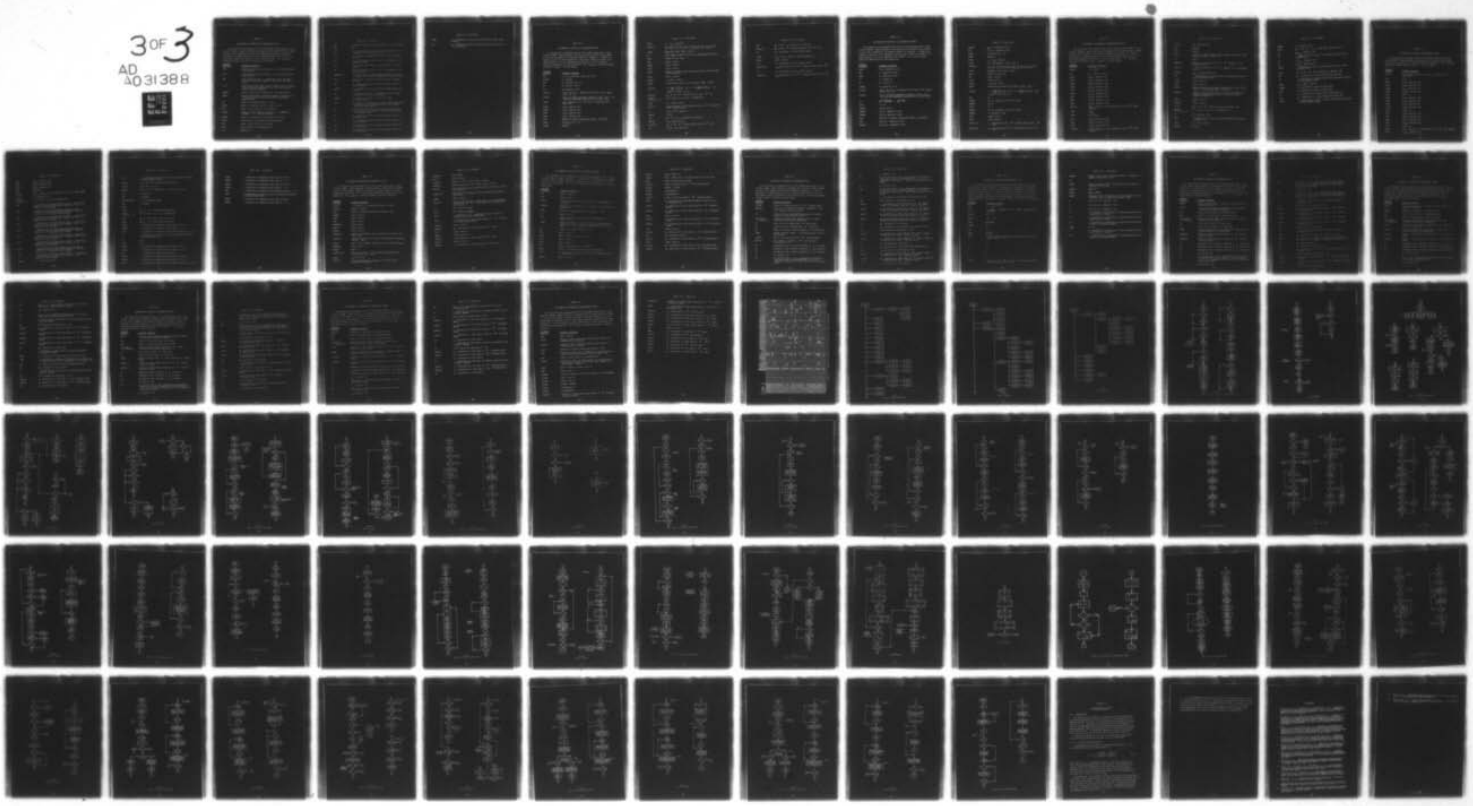
PROGRAM NOTATION	ALGEBRAIC NOTATION
CRP	length of pylon root-chord; input, item 19
CSIDE	length of local pylon chord along outboard side of chordwise row
CSIDEP	length of local pylon chord along inboard side of chordwise row
DTOR	degrees to radians conversion factor; $\pi/180$
FAC	FAC = 0.95; used to locate control point at 95 percent of chord containing panel centroid
HP	pylon height; input, item 19
IP	input, item 19
JLE	array index of first pylon panel in a chordwise row
KMAX	MSP + 1
MP	number of constant u-velocity panels on pylon
MSP	input, item 20
NCP	input, item 20
NCW	number of panels in a chordwise row on wing; input, item 13
NPANLS	number of constant u-velocity panels on wing
PLEX	XLF(JLE)
PSIPLE	sweep angle of pylon leading edge; input, item 19
PSIPTE	sweep angle of pylon trailing edge; input, item 19

Table I-8.- Concluded.

PSLPDF	SLLE-SLTE
SLLE	tan(PSIPLE)
SLTE	tan(PSIPTE)
SWIPPLE(I)	leading-edge slope of I^{th} panel
SWPPTE(I)	trailing-edge slope of I^{th} panel
XCPT(I)	x_W coordinate of control point on I^{th} panel
XLB(I)	x_W coordinate of left rear corner of I^{th} panel
XLF(I)	x_W coordinate of left front corner of I^{th} panel
XPLE	input, item 19
XRB(I)	x_W coordinate of right rear corner of I^{th} panel
XRF(I)	x_W coordinate of right front corner of I^{th} panel
Y(IP)	y_W location of pylon
YCPT(I)	y_W coordinate of control point on I^{th} panel
YLC(I)	y_W location of left edge of I^{th} panel
YPL	Y(IP), IP $\geq$ 1; 0, IP = 0
YRC(I)	y_W location of right edge of I^{th} panel
Z(K)	input, item 21
ZBAR	z_W location of panel centroid
ZCPT(I)	z_W coordinate of control point on I^{th} panel
ZLC(I)	z_W coordinate of left edge of I^{th} panel
ZRC(I)	z_W coordinate of right edge of I^{th} panel

AD-A031 388 NIELSEN ENGINEERING AND RESEARCH INC MOUNTAIN VIEW CALIF F/G 1/3
PREDICTION OF SUPERSONIC STORE SEPARATION CHARACTERISTICS. VOLU--ETC(U)
MAY 76 F K GOODWIN, M M KEIRSTEAD F33615-75-C-3053
UNCLASSIFIED NEAR-TR-106 AFFDL-TR-76-41-VOL-2 NL

3 OF 3
AD A031388



END

DATE FILMED
12-76

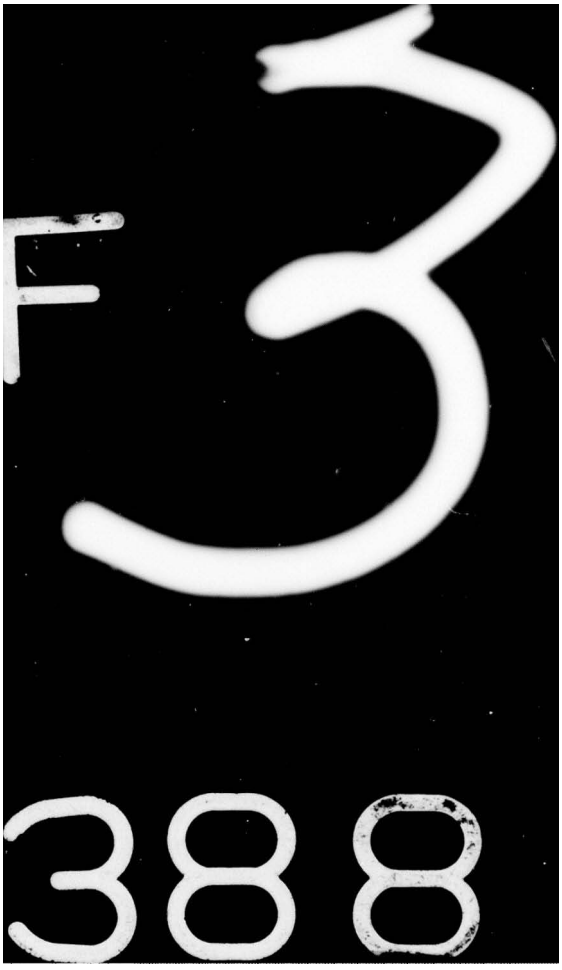

F
3
388

TABLE I-9

DICTIONARY OF NOTATION IN SUBROUTINE RESVEL

The following list presents most of the variable names used in subroutine RESVEL. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 1. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition. The x_W, y_W, z_W coordinate system is shown in figure 6 of this report.

PROGRAM NOTATION	ALGEBRAIC NOTATION
FDS	array containing the strengths of the fuselage doublet distribution
FSS	array containing the strengths of the fuselage source distribution
FXL	array containing the x positions of the fuselage singularity origins; positive, measured from tip of nose
II,IF	initial and final index values, respectively, of panels which induce perturbation velocities representing a given aircraft component
INUMCH	control index used in subroutines VELO and VELOTH
ISMTH	control index used to determine whether velocities are to be calculated using least-squares polynomial approximation
MS	number of thickness panels on wing
NFSOR	number of fuselage sources; input, item 9
NFU	input, item 4; NFU = 0, no fuselage
NORDWS	degree of least squares polynomial; $1 \leq$ NORDWS ≤ 5 (NORDWS = 3 in present program)
NPANLS	number of constant u-velocity panels on wing
NPTOT	total number of constant u-velocity panels
NPY	input, item 4; NPY = 0, no pylon
NTHP	total number of thickness panels
N1P	NPANLS + 1

Table I-9.- Continued.

N2	number of constant u-velocity panels on wing and pylon
N2P	N2 + 1
UP	u/V_∞ perturbation velocity at field point due to a component of the parent aircraft
UTU	sum of u/V_∞ perturbation velocities due to parent aircraft
VP	v/V_∞ perturbation velocity due to a component of the parent aircraft
VTV	sum of v/V_∞ perturbation velocities due to parent aircraft
WP	w/V_∞ perturbation velocity due to a component of the parent aircraft
WSCOE(I,J)	I^{th} coefficient in least squares polynomial used to calculate J^{th} velocity component; $1 \leq I \leq 6$, $J = 1,2,3$
WTW	sum of w/V_∞ perturbation velocities due to parent aircraft
XB	$x_{B_{be}}$ coordinate of point at which velocity field is to be calculated
XBWOC	$x_{B_{item\ 11}}$ coordinate of wing root-chord leading edge; input
XP	x location, relative to XSHFTW, of point at which velocity field is to be calculated
XSHFTW	$x_{W_{1/2(XT3 + XT4)}}$ location of center of XT3, XT4 range;
XT3	$x_{W_{ity}}$ location of beginning of region within which velocity field is to be calculated using least squares polynomial approximation
XT4	$x_{W_{field}}$ location of end of region within which velocity field is to be calculated using least square polynomial approximation
XW	$x_{W_{be}}$ coordinate of point at which velocity field is to be calculated
YB	$y_{B_{be}}$ coordinate of point at which velocity field is to be calculated
YW	$y_{W_{be}}$ coordinate of point at which velocity field is to be calculated
ZB	$z_{B_{be}}$ coordinate of point at which velocity field is to be calculated

Table I-9.- Concluded.

ZBWO	$z_{B_{item\ 11}}$ coordinate of wing root-chord leading edge; input
ZW	$z_{W_{be\ calculated}}$ coordinate of point at which velocity field is to

Table I-10

DICTIONARY OF NOTATION IN SUBROUTINE SEMFOR

The following list presents most of the variable names used in subroutine SEMFOR. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 4. Symbols in figures 8 and 10 of that reference are also used. Where a variable is an input quantity, it is so identified and section 3.2.1 of this report should be referred to for the definition. The x_B, y_B, z_B coordinate system is shown in figure 5 of this report.

PROGRAM NOTATION	ALGEBRAIC NOTATION
ABU	$\alpha_u + \beta_u$, equations (I-28) and (I-37)
ALFACR	α_f, radians
ANG	ϕ_f, radians (fig. 10)
AS	α_s, equation (I-16)
BS	β_s, equation (I-20)
CCL3(K)	$(cc_\ell)_3$ and $(cc_\ell)_4$, equations (I-14) and (I-19); equal since $s_h = s_v$
CCL5(K)	$(cc_\ell)_5$ if planar empennage, equation (I-29); $(cc_\ell)_6$ if cruciform empennage, equations (I-38) and (I-51)
CLLEM	$(C_\ell)_E$, equations (62) and (I-30) or equations (63) and (I-52)
CLMEM	$(C_m)_E$, equation (60)
CLNEM	$(C_n)_E$, equation (61)
CNEM	$(C_N)_E$, equation (58)
CYEM	$(C_Y)_E$, equation (59)
DTOR	degrees to radians conversion factor, 1/57.29578
FCONA	$(dC_L/d\alpha)_H/\pi(s_h - a)^2$
FCONB	FCONA/ℓ_R

Table I-10.- Continued.

FCONC	$\ell_f - x_{s,m}$, figure 8
FROLE(K)	$\phi_f + 90°$ for fin 1, $\phi_f + 270°$ for fin 2, $\phi_f + 180°$ for fin 3, ϕ_f for fin 4; radians; see figure 10
IPLNR	empennage type; input, item 36
IT	body segment number at which the empennage forces act
JMAX	number of tail fins
MSF	input, item 36
NDAMP	input, item 30
NGAM	input, item 30
NHSEG	number of points along store axis where velocity field was calculated
NROLL	input, item 30
PHIROL	ϕ_f; input, item 37
RADAV	average body radius in fin region; input, item 37
RFIN(K)	$a + \left(\frac{K - 1}{MSF - 1}\right)(s_h - a)$ or $a + \left(\frac{K - 1}{MSF - 1}\right)(s_v - a)$; figure 10 with $s_h \approx s_v$
UTL(K,J)	U_s^* at the k^{th} control point on the j^{th} fin
VAR(N) N = 1,2,...12	$\dot{\xi}, \dot{\eta}, \dot{\zeta}, p, q, r, \xi, \eta, \zeta, \Psi, \Theta, \Phi$, respectively
VINF	V_∞; input, item 3
VN(K,J)	velocity normal to the surface of the j^{th} fin at the k^{th} control point
VO	V_o/V_{∞_s}, see figure 10
VRATIO	V_∞/V_{∞_s}
VSO	V_s^* in the y_s direction of figure 10
VSTORE	V_{∞_s}, equation (41)
VTL(K,J)	V_s^* at the k^{th} control point on the j^{th} fin
WO	W_o/V_{∞_s}, see figure 10

Table I-10.- Concluded.

WSO	W_s^* in the z_s direction of figure 10
WTL(K,J)	W_s^* at the k^{th} control point on the j^{th} fin
XB	x_B coordinate of a fin control point
XMOM	$x_{s,m}$; input, item 32 , positive quantity
XTAIL	input, item 37
YB	y_B coordinate of a fin control point
YTAIL(K,J)	$y_{s_{fin}}$ coordinate of the k^{th} control point on the j^{th}
ZB	z_B coordinate of a fin control point
ZTAIL(K,J)	$z_{s_{fin}}$ coordinate of the k^{th} control point on the j^{th}

TABLE I-11

DICTIONARY OF NOTATION IN SUBROUTINE SEMPIN

The following list presents most of the variable names used in subroutine SEMPIN. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 4. Symbols in figures 8 and 10 of that reference are also used. Where a variable is an input quantity, it is so identified and section 3.2.1 of this report should be referred to for the definition.

PROGRAM NOTATION	ALGEBRAIC NOTATION
AKK1	$K(k_1)$, equation (I-43)
AK1	k_1, equation (I-42)
ARG3	A_1, equation (I-44)
A2	a^2, figure 10
A4	a^4, figure 10
CAPRSQ	R^2, equation (I-39)
CCL3(K)	$(cc_\ell)_3$ and $(cc_\ell)_4$, equations (I-14) and (I-19); equal since $s_h = s_v$
CCL5(K)	$(cc_\ell)_5$ if planar empennage, equation (I-29); $(cc_\ell)_6$ if cruciform empennage, equations (I-38) and (I-51)
CK	$\sqrt{1 - AK1*AK1} = \sqrt{1 - k_1^2}$
CKCK	$1 - k_1^2$
CLALPH	input, item 37
CTWGAM	$\cos 2\gamma$, equation (I-41)
CTWTHE	$\cos 2\theta$, equation (I-40)
DTOR	degrees to radians conversion factor, 1/57.29578
EAK	$E(A_1,k_1)$, equation (I-49)
EPIO2K	$E(\pi/2,k_1)$, equation (I-48)

Table I-11.- Concluded.

ESTRMX	a_{max} of separated store
FAK	$F(A_1,k_1)$, equation (I-47)
FCONA	$(dC_L/d\alpha)_H/\pi(s_h - a)^2$
FCONB	FCONA/ℓ_R
FCONC	$\ell_f - x_{s,m}$, figure 8
FINSS	tail fin semispan; input, item 37
FROLE(K)	ϕ_f + 90° for fin 1, ϕ_f + 270° for fin 2, ϕ_f + 180° for fin 3, ϕ_f for fin 4; radians; see figure 10
IPLNR	empennage type; input, item 36
JMAX	number of tail fins
MSF	input, item 36
NROLL	input, item 30
PHIROL	ϕ_f; input, item 37
RADAV	average body radius in fin region; input, item 37
RFIN(K)	$a + \left(\frac{K - 1}{MSF - 1}\right)(s_h - a)$ or $a + \left(\frac{K - 1}{MSF - 1}\right)(s_v - a)$; figure 10 with $s_h = s_v$
SHS	s_h^2
STWGAM	sin 2γ, equations (I-41) and (I-42)
STWTHE	sin 2θ
TOVPI	$2/\pi$
TTWGAM	tan 2γ, equations (I-41) and (I-42)
TWOTHE	2θ, equation (I-40)
XMOM	input, item 32
XTAIL	input, item 37
YTAIL(K,J)	y_s coordinate of the k^{th} control point on the j^{th} fin
ZTAIL(K,J)	z_s coordinate of the k^{th} control point on the j^{th} fin

TABLE I-12

DICTIONARY OF NOTATION IN SUBROUTINE SFORCE

The following list presents most of the variable names used in subroutine SFORCE. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 4. Where a variable is an input quantity, it is so identified and section 3.2.1 of this report should be referred to for the definition. The x_B, y_B, z_B coordinate system is shown in figure 5 of this report.

PROGRAM NOTATION	ALGEBRAIC NOTATION
ALFACR	α_f, radians
CDC	c_{d_c} ; input, item 35
CLMBY	$(C_m)_{BY}$, equation (44)
CLMCF	$(C_m)_{CF}$, equation (53)
CLMSB	$(C_m)_{SB}$, equation (48)
CLNBY	$(C_n)_{BY}$, equation (45)
CLNCF	$(C_n)_{CF}$, equation (54)
CLNSB	$(C_n)_{SB}$, equation (49)
CNBY	$(C_N)_{BY}$, equation (42)
CNCF	$(C_N)_{CF}$, equation (51)
CNSB	$(C_N)_{SB}$, equation (46)
CNX(NN)	total dC_N/dx_s at the midpoint of the NN^{th} body segment
CONA	$2/a^2_{max}$
CONB	$2c_{d_c}/\pi a^2_{max}$
CYBY	$(C_Y)_{BY}$, equation (43)
CYCF	$(C_Y)_{CF}$, equation (52)
CYSB	$(C_Y)_{SB}$, equation (47)
CYX(NN)	total dC_Y/dx_s at the midpoint of the NN^{th} body segment

Table I-12.- Continued.

DC(I,J)	[A], equation (28)
DCN	dC_N/dx_s
DCY	dC_Y/dx_s
DELX	length of a body segment; store length divided by NSEG which was input in item 30
DX	DELX/2
EDRDX(I)	da/dx_s at the midpoint of the i^{th} segment of the separated store
ERAD(I)	a at the midpoint of the i^{th} segment of the separated store
ESTRMX	a_{max} of separated store
ETA	η coordinate of point on store axis measured relative to store moment center
NDAMP	input, item 30
NGAM	input, item 30
NHSEG	number of points along store axis where velocity field is to be calculated; NHSEG = 2*NSEG + 1
NHSEGO	number of points along store axis ahead of separation point where velocity field is to be calculated; NHSEGO = 2*NSEGXO + 1
NSEG	input, item 30
NSEGXO	input, item 30
UT(N)	U_s^* at the N^{th} point along body, equation (40)
VAR(N), N = 1,2,...12	$\dot{\xi},\dot{\eta},\dot{\zeta},p,q,r,\xi,\eta,\zeta,\Psi,\Theta,\Phi$ respectively
VC	V_c^*, equation (55)
VETA	η component of the store free-stream velocity vector, equation (32)
VINF	V_∞; input, item 3
VRATIO	V_∞/V_{∞_s}

Table I-12.- Concluded.

VSTORE	V_{∞_s}, equation (41)
VT(N)	V_s^* at the N^{th} point along body, equation (40)
VX	$U_{\infty_{s,x_s}}$ equation (35)
VXI	ξ component of the store free-stream velocity vector, equation (32)
VY	$-V_{\infty_{s,y_s}}$ equation (35)
VZ	$W_{\infty_{s,z_s}}$ equation (35)
VZETA	ζ component of the store free-stream velocity vector, equation (32)
WT(N)	W_s^* at the N^{th} point along body, equation (40)
XB	x_B coordinate of a point along store body axis
XI	ξ coordinate of a point on store axis measured relative to store moment center
XMOM	$x_{s,m}$; input, item 32
XSTOR	x_s coordinate of point on store axis
XXX	x coordinate of point on store axis
YB	y_B coordinate of a point along store body axis
ZB	z_B coordinate of a point along store body axis
ZETA	ζ coordinate of a point on store axis measured relative to store moment center

TABLE I-13

DICTIONARY OF NOTATION IN SUBROUTINE SOUTPT

The following list presents most of the variable names used in subroutine SOUTPT. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 4. The x_B, y_B, z_B coordinate system is shown in figure 5 of this report.

PROGRAM NOTATION	ALGEBRAIC NOTATION
CLLEM	$(C_\ell)_E$, equations (62) and (I-30) or equations (63) and (I-52)
CLLT	total C_ℓ
CLMBY	$(C_m)_{BY}$, equation (44)
CLMCF	$(C_m)_{CF}$, equation (53)
CLMEM	$(C_m)_E$, equation (60)
CLMSB	$(C_m)_{SB}$, equation (48)
CLMT	total C_m
CLNBY	$(C_n)_{BY}$, equation (45)
CLNCF	$(C_n)_{CF}$, equation (54)
CLNEM	$(C_n)_E$, equation (61)
CLNSB	$(C_n)_{SB}$, equation (49)
CLNT	total C_n
CNBY	$(C_N)_{BY}$, equation (42)
CNCF	$(C_N)_{CF}$, equation (51)
CNEM	$(C_N)_E$, equation (58)
CNSB	$(C_N)_{SB}$, equation (46)
CNT	total C_N
CNX(K)	total dC_N/dx_s at the midpoint of the K^{th} body segment
CYBY	$(C_Y)_{BY}$, equation (43)

Table I-13.- Continued.

CYCF	$(C_Y)_{CF}$, equation (52)
CYEM	$(C_Y)_E$, equation (59)
CYSB	$(C_Y)_{SB}$, equation (47)
CYT	total C_Y
CYX(K)	total dC_Y/dx_s at the midpoint of the K^{th} body segment
DC(I,J)	[A], equation (28)
DVAR(N), N = 1,2,...12	$\ddot{\xi},\ddot{\eta},\ddot{\zeta},\dot{p},\dot{q},\dot{r},\dot{\xi},\dot{\eta},\dot{\zeta},\dot{\psi},\dot{\theta},\dot{\phi}$ respectively
DXB	ξ coordinate of the store base at time t relative to where it would have been had it remained in the t = 0 position on the aircraft
DXCG	ξ coordinate of the store moment center at time t relative to where it would have been had it remained in the t = 0 position on the aircraft
DXN	ξ coordinate of the store nose at time t relative to where it would have been had it remained in the t = 0 position on the aircraft
DYB	η coordinate of the store base at time t relative to where it would have been had it remained in the t = 0 position on the aircraft
DYCG	η coordinate of the store moment center at time t relative to where it would have been had it remained in the t = 0 position on the aircraft
DYN	η coordinate of the store nose at time t relative to where it would have been had it remained in the t = 0 position on the aircraft
DZB	ζ coordinate of the store base at time t relative to where it would have been had it remained in the t = 0 position on the aircraft
DZCG	ζ coordinate of the store moment center at time t relative to where it would have been had it remained in the t = 0 position on the aircraft
DZN	ζ coordinate of the store nose at time t relative to where it would have been had it remained in the t = 0 position on the aircraft
ESTLGC	ℓ_s of separated store

Table I-13.- Continued.

ETA	η coordinate of point on store axis measured relative to store moment center
EXST(J)	x_s at the J^{th} point along store axis
FTHRUS	store thrust force
NTHRUS	input, item 30; NTHRUS = 0, no thrust calculated
PHI	Φ, degrees
PSI	Ψ, degrees
SLTHC(NEJSTR)	ℓ_s of separated store
THA	Θ, degrees
TIME	t
UT(J)	U_s^* at J^{th} point along store axis
VAR(N), N = 1,2,...12	$\dot{\xi},\dot{\eta},\dot{\zeta},p,q,r,\xi,\eta,\zeta,\Psi,\Theta,\Phi$ respectively
VT(J)	V_s^* at J^{th} point along store axis
WT(J)	W_s^* at J^{th} point along store axis
XBASE	ξ coordinate of separated store base at time t
XBASEI	ξ coordinate of separated store base at t = 0
XCGI	ξ coordinate of separated store moment center at t = 0
XI	ξ coordinate of point on store axis measured relative to store moment center
XL	x_s/ℓ_s
XMOM	$x_{s,m}$
XNOSE	ξ coordinate of separated store nose at time t
XNOSEI	ξ coordinate of separated store nose at t = 0
XXX	x coordinate of point on store axis
YBASE	η coordinate of separated store base at time t
YBASEI	η coordinate of separated store base at t = 0
YCGI	η coordinate of separated store moment center at t = 0

Table I-13.- Concluded.

YNOSE	η coordinate of separated store nose at time t
YNOSEI	η coordinate of separated store nose at t = 0
ZBASE	ζ coordinate of separated store base at time t
ZBASEI	ζ coordinate of separated store base at t = 0
ZCGI	ζ coordinate of separated store moment center at t = 0
ZETA	ζ coordinate of point on store axis measured relative to store moment center
ZNOSE	ζ coordinate of separated store nose at time t
ZNOSEI	ζ coordinate of separated store nose at t = 0

TABLE I-14

DICTIONARY OF NOTATION IN SUBROUTINE STORIO

The following list presents most of the variable names used in subroutine STORIO. Where possible a variable name is identified by a symbol in the list of symbols. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition. The x_W, y_W, z_W coordinate system is shown in figure 6.

PROGRAM NOTATION	ALGEBRAIC NOTATION
ALFACR	α_f, radians; α_f in degrees was input in item 3
ALFAS(K)	ALFACR + SIBCR(K)
DTOR	degrees to radians conversion constant; $\pi/180$
MSHAPE	input, item 26
MSOR	input, item 26
NCW	input, item 13
NSHAPE(K)	input, item 24
NSHPT	input, item 25
NSPOLJ	input, item 27
NSPOLY(K)	number of polynomials used to specify the shape of the K^{th} store
NSSOR(K)	number of sources in the source distribution representing the K^{th} store
NSTRS	input, item 4; NSTRS limited to one in present program
NUMSTR(K)	input, item 24
NXBODY	MSOR + 1
SCOEF(I,J,K)	J^{th} coefficient of I^{th} polynomial describing the shape of the K^{th} store
SCOFJ	input, item 29
SDS(L,K)	array containing the strengths of the K^{th} store doublet distribution

Table I-14.- Concluded.

SIBCR(K)	SIC(J)*DTOR
SIC(K)	input, item 24
SLTHC(K)	length of J^{th} store; input, item 24
SRMAX(K)	maximum radius of J^{th} store; input, item 24
SSS(L,K)	array containing the strengths of the K^{th} store source distribution
SXEND(I,K)	end points of polynomial sections specifying shape of K^{th} store
SXNDJ	input, item 28
SXL(L,K)	array containing the x positions of the singularity origins for the K^{th} store; positive, measured from tip of nose
T(J)	J^{th} source strength
TC(J)	J^{th} doublet strength
TX(J)	x location of J^{th} singularity origin; positive, measured from tip of body nose
XRF(J)	x_W coordinate of right front corner of J^{th} constant u-velocity wing panel
XSNC(J)	input, item 24
XWSOC(J)	x_W coordinate of tip of nose of J^{th} store
Y(I)	input, item 14
YSN(J)	input, item 24
YWSO(J)	y_W coordinate of tip of nose of J^{th} store
ZSN(J)	input, item 24
ZWSO(J)	z_W coordinate of tip of nose of J^{th} store

TABLE I-15

DICTIONARY OF NOTATION IN SUBROUTINE THKLYT

The following list presents most of the variable names used in subroutine THKLYT. Where possible a variable name is identified by a symbol in the list of symbols. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition.

PROGRAM NOTATION	ALGEBRAIC NOTATION
CRP	input, item 19
CRW	input, item 12
CSIDE	length of local chord on left side of a chordwise row of panels
CSIDEP	length of local chord on right side of a chordwise row of panels
DTOR	degrees to radians conversion constant; $\pi/180$
IP	input, item 19
JLE	index of array location of the first panel in a chordwise row
LVSWP	control variable; if wing leading-edge and trailing-edge sweep angles constant at all spanwise stations, LVSWP = 0; if not constant, LVSWP = 1
MS	number of thickness panels on wing
MSP	input, item 20
MSWP	MSW + 1; MSW was input, item 13
NCPS	input, item 22
NCWS	input, item 17
NPY	input, item 4; NPY = 0, no pylon
NTHP	number of thickness panels on wing and pylon
PLEX	x_{W_a} coordinate of pylon leading edge on right side of a chordwise row
PSIWLE	input, item 19

Table I-15.- Concluded.

PSIWTE	input, item 19
PSLPDF	SLLE - SLTE; SLTE is slope of pylon trailing edge
SLLE	slope of pylon leading edge
SLLET(I)	slope of leading edge of I^{th} thickness panel
SLPDIF	SLPWLE - SLPWTE
SLPWLE	tan[PSIWLE(I)]
SLPWTE	tan[PSIWTE(I)]
SLTET(I)	slope of trailing edge of I^{th} thickness panel
WLEX	x_{W_a} coordinate of wing leading edge on right side of chordwise row
XLBT(I)	$x_{W_{panel}}$ coordinate of left rear corner of I^{th} thickness
XLFT(I)	$x_{W_{panel}}$ coordinate of left front corner of I^{th} thickness
XPLE	input, item 19
XRBT(I)	$x_{W_{panel}}$ coordinate of right rear corner of I^{th} thickness
XRFT(I)	$x_{W_{panel}}$ coordinate of right front corner of I^{th} thickness
Y(K)	input, item 14
YLCT(I)	y_W coordinate of left side of I^{th} thickness panel
YPL	y_W location of pylon
YRCT(I)	y_W coordinate of right side of I^{th} thickness panel
Z(K)	input, item 21
ZLCT(I)	z_W coordinate of left side of I^{th} thickness panel
ZRCT(I)	z_W coordinate of right side of I^{th} thickness panel

TABLE I-16

DICTIONARY OF NOTATION IN SUBROUTINE VELBD

The following list presents most of the variable names used in subroutine VELBD. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 1. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition. The wing coordinate system is shown in figure 6 of this report.

PROGRAM NOTATION	ALGEBRAIC NOTATION
CST2(K)	cosine of orientation angle of K^{th} fuselage panel
DELTP(I)	$1/\pi(u_{+_I}/V_\infty)$; singularity strength, divided by π, of I^{th} constant u-velocity panel
EM	slope of leading edge and trailing edge of fuselage constant u-velocity panels; EM = 0
FELT, FELTα α = A,AI,B,BI	logical variables; TRUE whenever non-zero corner influence functions are calculated
IF	final value of fuselage panel DO loop index
II	initial value of fuselage panel DO loop index
N2	number of constant u-velocity panels on wing and pylon
PYPNL	logical variable; FALSE if influencing panel is not a pylon panel
SNT2(K)	sine of orientation angle of K^{th} fuselage panel
THTI(K)	polar angle in degrees of K^{th} fuselage panel (see section I-5)
TU	superposed corner influence functions in x_W direction
TV	superposed corner influence functions in y_W direction
TW	superposed corner influence functions in z_W direction
U	F_u, equation (12)
UP	if II $\neq$ IF, sum of u/V_∞ perturbation velocities at field point due to fuselage panels, if II = IF, influence coefficient of II^{th} panel and given control point

Table I-16.- Concluded.

V	F_v, equation (12)
VP	if II ≠ IF, sum of v/V_∞ perturbation velocities at field point due to fuselage panels; if II = IF, influence coefficient of II^{th} panel and given control point
W	F_w, equation (12)
WP	if II ≠ IF, sum of w/V_∞ perturbation velocities at field point due to fuselage panels; if II = IF, influence coefficient of II^{th} panel and given control point
X	x_c coordinate of field point (see section I-35)
XI	local subroutine equivalent of XX
XLB(I)	x_W coordinate of left rear corner of I^{th} panel; right is clockwise from left, looking forward
XLF(I)	x_W coordinate of left front corner of I^{th} panel; right is clockwise from left, looking forward
XRB(I)	x_W coordinate of right rear corner of I^{th} panel; right is clockwise from left, looking forward
XRF(I)	x_W coordinate of right front corner of I^{th} panel; right is clockwise from left, looking forward
XX	x_W coordinate of field point
Y	y_c coordinate of field point (see section I-35)
YI	local subroutine equivalent of YY
YLC(I)	y_W coordinate of left edge of I^{th} panel; right is clockwise from left, looking forward
YRC(I)	y_W coordinate of right edge of I^{th} panel; right is clockwise from left, looking forward
YY	y_W coordinate of field point
Z	z_c coordinate of field point (see section I-35)
ZI	local subroutine equivalent of ZZ
ZLC(I)	z_W coordinate of left edge of I^{th} panel; right is clockwise from left, looking forward
ZRC(I)	z_W coordinate of right edge of I^{th} panel; right is clockwise from left, looking forward
ZZ	z_W coordinate of field point

TABLE I-17

DICTIONARY OF NOTATION IN SUBROUTINE VELO

The following list presents most of the variable names used in subroutine VELO. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 1. Symbols defined in section II-2.1 of that reference are also used. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition. The coordinate system associated with a semi-infinite triangle is shown in figure 3 of reference 1.

PROGRAM NOTATION	ALGEBRAIC NOTATION
ARG1	$x^2 - \beta^2(y^2 + z^2)$
BETA	β; BETANU, if INUMCH = 1 and PYPNL = FALSE; BETAOL otherwise
BETANU	$(M_\ell^2 - 1)^{1/2}$
BETAOL	$(M_\infty^2 - 1)^{1/2}$
BTSQ	BETA*BETA
BTSQOL	$M_\infty^2 - 1$
EML	$M_{\ell e}$
EMLSQ	EML*EML
FELT	logical variable, FALSE when returning from the subroutine with U = V = W = 0
F1	F_1
F2	F_2
F4	F_4
F5	F_5
F7	F_7
INSIDE	logical variable; TRUE if point is located inside Mach cone from origin of triangle

Table I-17.- Concluded.

INUMCH	control index used to determine choice of BETANU or BETAOL (see section I-18)
PI	π
PYPNL	logical variable, TRUE if semi-infinite triangle is located on the pylon
ROOTA1	$(ARG1)^{1/2}$
STEST	BTSQ - EMLSQ
TLRNC	tolerance factor related to the wing semispan; used to represent zero in subroutine logical tests
U	$F_u(x,y,z,\beta,\Psi)$, equation (12)
V	$F_v(x,y,z,\beta,\Psi)$, equation (12)
W	$F_w(x,y,z,\beta,\Psi)$, equation (12)
X	x coordinate of field point in system associated with semi-infinite triangle
Y	y coordinate of field point in system associated with semi-infinite triangle
YS	y coordinate of field point in system associated with corner of influencing panel
YZSQ	$Y^2 + Z^2$
Z	z coordinate of field point in system associated with semi-infinite triangle
ZS	z coordinate of field point in system associated with corner of influencing panel

TABLE I-18

DICTIONARY OF NOTATION IN SUBROUTINE VELPP

The following list presents most of the variable names used in subroutine VELPP. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 1. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition. The x_W, y_W, z_W coordinate system is shown in figure 6.

PROGRAM NOTATION	ALGEBRAIC NOTATION
CENTER	logical variable; TRUE if pylon is located under the fuselage centerline
DELTP	$u_+/\pi V_\infty$; array containing constant u-velocity panel singularity strengths, divided by π
EM1	slope of leading edge of influencing panel
EM2	slope of trailing edge of influencing panel
FELT, FELTα α = A, AI, B, BI	logical variables; TRUE whenever non-zero influence functions are calculated
IF	final value of panel DO loop index
II	initial value of panel DO loop index
PYPNL	logical variable required by subroutine VELO; TRUE if influencing panel is a pylon panel
SWPPLE(I)	tangent of leading-edge sweep angle of I^{th} constant u-velocity panel
SWPPTE(I)	tangent of trailing-edge sweep angle of I^{th} constant u-velocity panel
TU	superposed corner influence functions in x_W direction
TV	superposed corner influence functions in y_W direction
TW	superposed corner influence functions in z_W direction
U	F_u, equation (12)
UP	if II $\neq$ IF, sum of u/V_∞ perturbation velocities at field point due to pylon constant u-velocity panels; if II = IF, influence coefficient of II^{th} panel and given control point
V	F_v, equation (12)

Table I-18.- Concluded.

VP	if II ≠ IF, sum of v/V_∞ perturbation velocities at field point due to pylon constant u-velocity panels; if II = IF, influence coefficient of II^{th} panel and given control point
W	F_w, equation (12)
WP	if II ≠ IF, sum of w/V_∞ perturbation velocities at field point due to pylon constant u-velocity panels; if II = IF, influence coefficient of II^{th} panel and given control point
X	x coordinate of field point in system required by subroutine VELO
XI	local subroutine equivalent of XX
XLB(I)	$x_{W_{u\text{-velocity panel}}}$ coordinate of left rear corner of I^{th} constant
XLF(I)	$x_{W_{u\text{-velocity panel}}}$ coordinate of left front corner of I^{th} constant
XRB(I)	$x_{W_{u\text{-velocity panel}}}$ coordinate of right rear corner of I^{th} constant
XRF(I)	$x_{W_{u\text{-velocity panel}}}$ coordinate of right front corner of I^{th} constant
XX	x_W coordinate of field point
Y	y coordinate of field point in system required by subroutine VELO
YDIR	y_c coordinate of field point (see section I-39)
YIMG	y_c coordinate of image of field point with respect to aircraft vertical plane of symmetry (see section I-39)
YPL	y_W location of pylon
YY	y_W coordinate of field point
Z	z coordinate of field point in system required by subroutine VELO
ZI	local subroutine equivalent of ZZ
ZLC(I)	$z_{W_{\text{panel}}}$ coordinate of left edge of I^{th} constant u-velocity
ZRC(I)	$z_{W_{\text{panel}}}$ coordinate of right edge of I^{th} constant u-velocity
ZZ	z_W coordinate of field point

TABLE I-19

DICTIONARY OF NOTATION IN SUBROUTINE VELPTH

The following list presents most of the variable names used in subroutine VELPTH. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 1. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition. The x_W, y_W, z_W coordinate system is shown in figure 6.

PROGRAM NOTATION	ALGEBRAIC NOTATION
CENTER	logical variable; TRUE if pylon is located under the fuselage centerline
DZDX(I)	THETPL(I)/π; THETPL(I) was input item 23
EM1	slope of leading edge of influencing panel
EM2	slope of trailing edge of influencing panel
FELT,FELTα α = A,AI,B,BI	logical variables; TRUE whenever non-zero corner influence functions are calculated
II,IF	initial and final values, respectively, of influencing panel DO loop index
PYPNL	logical variable used by subroutine VELOTH; TRUE if influencing panel is a pylon panel
SLLET(I)	tangent of leading-edge sweep angle of I^{th} thickness panel
SLTET(I)	tangent of trailing-edge sweep angle of I^{th} thickness panel
TU	superposed corner influence functions in x_W direction
TV	superposed corner influence functions in y_W direction
TW	superposed corner influence functions in z_W direction
U	F_{u_t}, equation (14)
UP	sum of u/V_∞ perturbation velocities at the field point due to pylon thickness panels
V	F_{v_t}, equation (14)

Table I-19.- Concluded.

VP	sum of v/V_∞ perturbation velocities at the field point due to pylon thickness panels
W	F_{w_t}, equation (14)
WP	sum of w/V_∞ perturbation velocities at the field point due to pylon thickness panels
X	x coordinate of field point in system required by subroutine VELOTH
XI	local subroutine equivalent of XX
XLBT(I)	x_W coordinate of left rear corner of I^{th} thickness panel
XLFT(I)	x_W coordinate of left front corner of I^{th} thickness panel
XRBT(I)	x_W coordinate of right rear corner of I^{th} thickness panel
XRFT(I)	x_W coordinate of right front corner of I^{th} thickness panel
XX	x_W coordinate of field point
Y	y coordinate of field point in system required by subroutine VELOTH
YDIR	y_c coordinate of field point (see section I-39)
YIMG	y_c coordinate of image of field point with respect to aircraft vertical plane of symmetry (see section I-39)
YPL	y_W location of pylon in wing coordinate system
YY	y_W coordinate of field point
Z	z coordinate of field point in system required by subroutine VELOTH
ZI	local subroutine equivalent of ZZ
ZLCT(I)	z_W coordinate of left edge of I^{th} thickness panel
ZRCT(I)	z_W coordinate of right edge of I^{th} thickness panel
ZZ	z_W coordinate of field point

TABLE I-20

DICTIONARY OF NOTATION IN SUBROUTINE VELWP

The following list presents most of the variable names used in subroutine VELWP. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 1. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition. The x_W, y_W, z_W coordinate system is shown in figure 6.

PROGRAM NOTATION	ALGEBRAIC NOTATION
DELTP	$u_+/(\pi V_\infty)$; array containing constant u-velocity panel singularity strengths, divided by π
EM1	slope of influencing panel leading edge
EM2	slope of influencing panel trailing edge
FELT, FELTα α = A,AI,B,BI	logical variable; TRUE whenever non-zero corner influence functions are calculated
IF	final value of panel DO loop index
II	initial value of panel DO loop index
PYPNL	logical variable; TRUE if influencing panel is a pylon panel
SWPPLE(I)	tangent of leading-edge sweep angle of I^{th} constant u-velocity panel
SWPPTE(I)	tangent of trailing-edge sweep angle of I^{th} constant u-velocity panel
TU	superposed corner influence in x_W direction
TV	superposed corner influence in y_W direction
TW	superposed corner influence in z_W direction
U	F_u, equation (12)
UP	if II $\neq$ IF, sum of u/V_∞ perturbation velocities at field point due to wing constant u-velocity panels; if II = IF, influence coefficient for II^{th} panel and given control point
V	F_v, equation (12)

Table I-20.- Concluded.

VP	if II ≠ IF, sum of v/V_∞ perturbation velocities at field point due to wing constant u-velocity panels; if II = IF, influence coefficient for II^{th} panel and given control point
W	F_w, equation (12)
WP	if II ≠ IF, sum of w/V_∞ perturbation velocities at field point due to wing constant u-velocity panels; if II = IF, influence coefficient for II^{th} panel and given control point
X	x coordinate of field point in system required by subroutine VELO
XI	local subroutine equivalent of XX
XLB(I)	x_w coordinate of left rear corner of I^{th} panel, viewed from the rear
XLF(I)	x_w coordinate of left front corner of I^{th} constant u-velocity panel
XRB(I)	x_w coordinate of right rear corner of I^{th} constant u-velocity panel
XRF(I)	x_w coordinate of right front corner of I^{th} constant u-velocity panel
XX	x_w coordinate of field point
Y	y coordinate of field point in system required by subroutine VELO
YI	local subroutine equivalent of YY
YLC(I)	y_w coordinate of left edge of I^{th} constant u-velocity panel
YRC(I)	y_w coordinate of right edge of I^{th} constant u-velocity panel
YY	y_w coordinate of field point
Z	z coordinate of field point in system required by subroutine VELO
ZZ	z_w coordinate of field point

TABLE I-21

DICTIONARY OF NOTATION IN SUBROUTINE VELWTH

The following list presents most of the variable names used in subroutine VELWTH. Where possible a variable name is identified by a symbol in the list of symbols or an equation number in reference 1. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition. The x_w, y_w, z_w coordinate system is shown in figure 6.

PROGRAM NOTATION	ALGEBRAIC NOTATION
DZDX(I)	THETAL(I)/π; THETAL(I) was input, item 23
EM1	slope of leading edge of influencing panel
EM2	slope of trailing edge of influencing panel
FELT,FELTα α = A,AI,B,BI	logical variable; TRUE whenever non-zero corner influence functions are calculated
II,IF	initial and final values, respectively, of influencing panel DO loop index
PYPNL	logical variable used by subroutine VELOTH; FALSE if influencing panel is a wing panel
SLLET(I)	tangent of leading-edge sweep angle of I^{th} thickness panel
SLTET(I)	tangent of trailing-edge sweep angle of I^{th} thickness panel
TU	superposed corner influence functions in x_w direction
TV	superposed corner influence functions in y_w direction
TW	superposed corner influence functions in z_w direction
U	F_{u_t}, equation (14)
UP	sum of u/V_∞ perturbation velocities due to wing thickness panels
V	F_{v_t}, equation (14)
VP	sum of v/V_∞ perturbation velocities due to wing thickness panels
W	F_{w_t}, equation (14)

Table I-21.- Concluded.

WP	sum of w/V_∞ perturbation velocities due to wing thickness panels
X	x coordinate of field point in system required by subroutine VELOTH
XI	local subroutine equivalent of XX
XLBT(I)	x_w coordinate of left rear corner of I^{th} thickness panel
XLFT(I)	x_w coordinate of left front corner of I^{th} thickness panel
XRBT(I)	x_w coordinate of right rear corner of I^{th} thickness panel
XRFT(I)	x_w coordinate of right front corner of I^{th} thickness panel
XX	x_w coordinate of field point
Y	y coordinate of field point in system required by subroutine VELOTH
YI	local subroutine equivalent of YY
YLCT(I)	y_w coordinate of left edge of I^{th} thickness panel
YRCT(I)	y_w coordinate of right edge of I^{th} thickness panel
YY	y_w coordinate of field point
Z	z coordinate of field point in system required by subroutine VELOTH
ZLCT(I)	z_w coordinate of left edge of I^{th} thickness panel
ZRCT(I)	z_w coordinate of right edge of I^{th} thickness panel
ZZ	z_w coordinate of field point

TABLE I-22

DICTIONARY OF NOTATION IN SUBROUTINE WLYOUT

The following list presents most of the variable names used in subroutine WLYOUT. Where possible a variable name is identified by a symbol in the list of symbols. Where a variable is an input quantity it is so identified and section 3.2.1 of this report should be referred to for the definition. If not specified, panels referred to in this dictionary are constant u-velocity panels. The x_w, y_w, z_w coordinate system is shown in figure 6.

PROGRAM NOTATION	ALGEBRAIC NOTATION
CRW	input, item 12
CSIDE	length of local wing chord along outboard edge of a chordwise row of panels
CSIDEP	length of local wing chord along inboard edge of a chordwise row of panels
DTOR	degrees to radians conversion constant; $\pi/180$
FAC	constant used to locate XCPT(J) at 95 percent of the chord containing the panel centroid; FAC = 0.95
JLE	index of front panel in a chordwise row
LVSWP	control variable; if leading-edge and trailing-edge sweep angles constant for entire wing, LVSWP = 0; if not constant, LVSWP = 1
MSWP	MSW + 1; MSW is input, item 13
NCW	input, item 13
NPANLS	number of constant u-velocity panels on wing; NCW*MSW
PSIWLE	input, item 14
PSIWTE	input, item 14
SLPDIF	SLPWLE - SLPWTE
SLPWLE	tan[PSIWLE(I)]
SLPWTE	tan[PSIWTE(I)]
SWPPLE(I)	tangent of leading-edge sweep angle of I^{th} constant u-velocity panel

Table I-22.- Concluded.

SWPPTE(I)	tangent of trailing-edge sweep angle of I^{th} constant u-velocity panel
WLEX	x_w coordinate of wing leading edge on right edge of a chordwise row
XCPT(I)	x_w coordinate of I^{th} control point
XLB(I)	x_w coordinate of left rear corner of I^{th} panel
XLF(I)	x_w coordinate of left front corner of I^{th} panel
XRB(I)	x_w coordinate of right rear corner of I^{th} panel
XRF(I)	x_w coordinate of right front corner of I^{th} panel
Y(I)	input, item 14
YBAR	y_w coordinate of centroid of I^{th} panel
YCPT(I)	y_w coordinate of I^{th} control point
YLC(I)	y_w coordinate of left edge of I^{th} panel
YRC(I)	y_w coordinate of right edge of I^{th} panel
ZCPT(I)	z_w coordinate of I^{th} control point
ZLC(I)	z_w coordinate of left edge of I^{th} panel
ZRC(I)	z_w coordinate of right edge of I^{th} panel

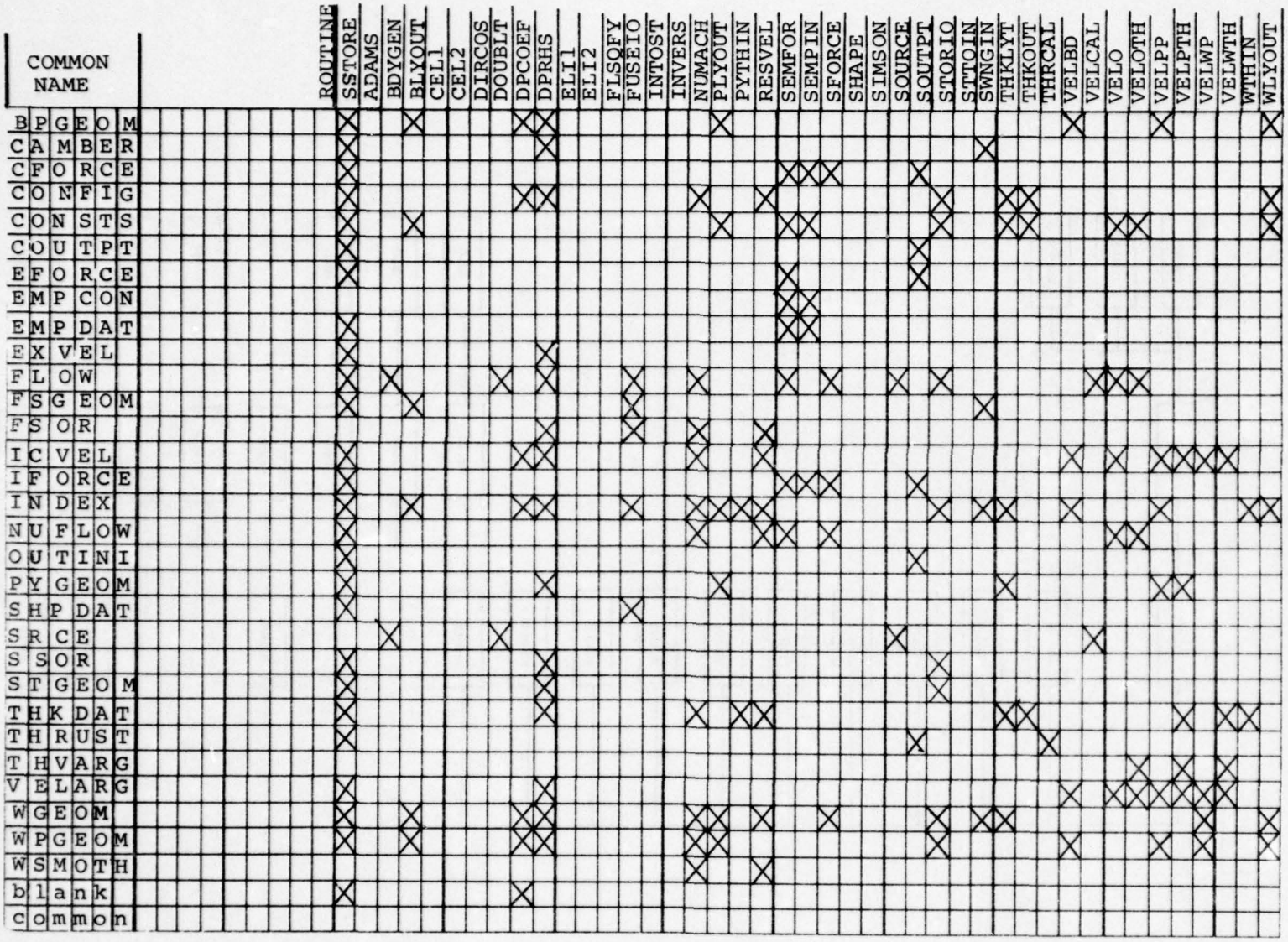
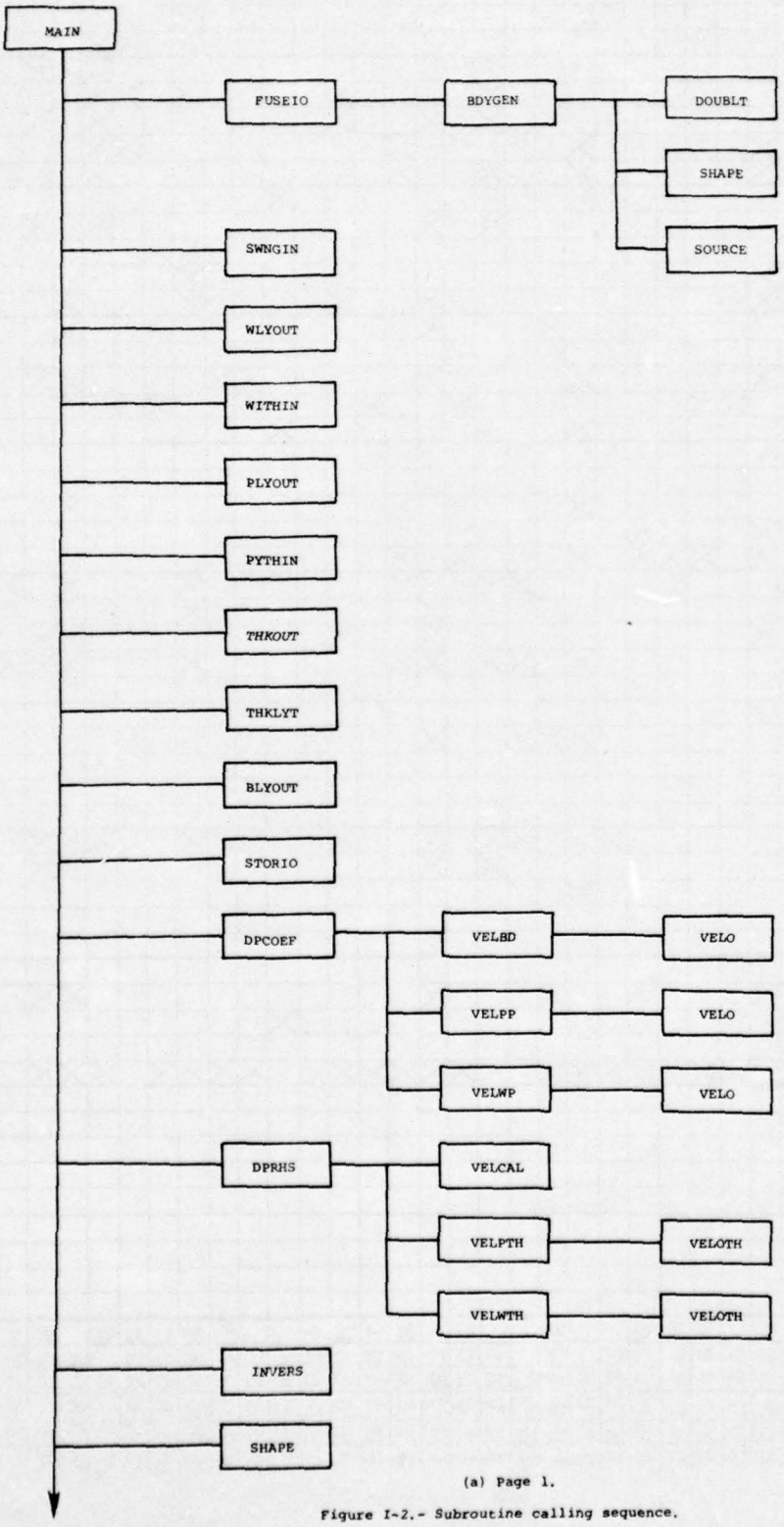
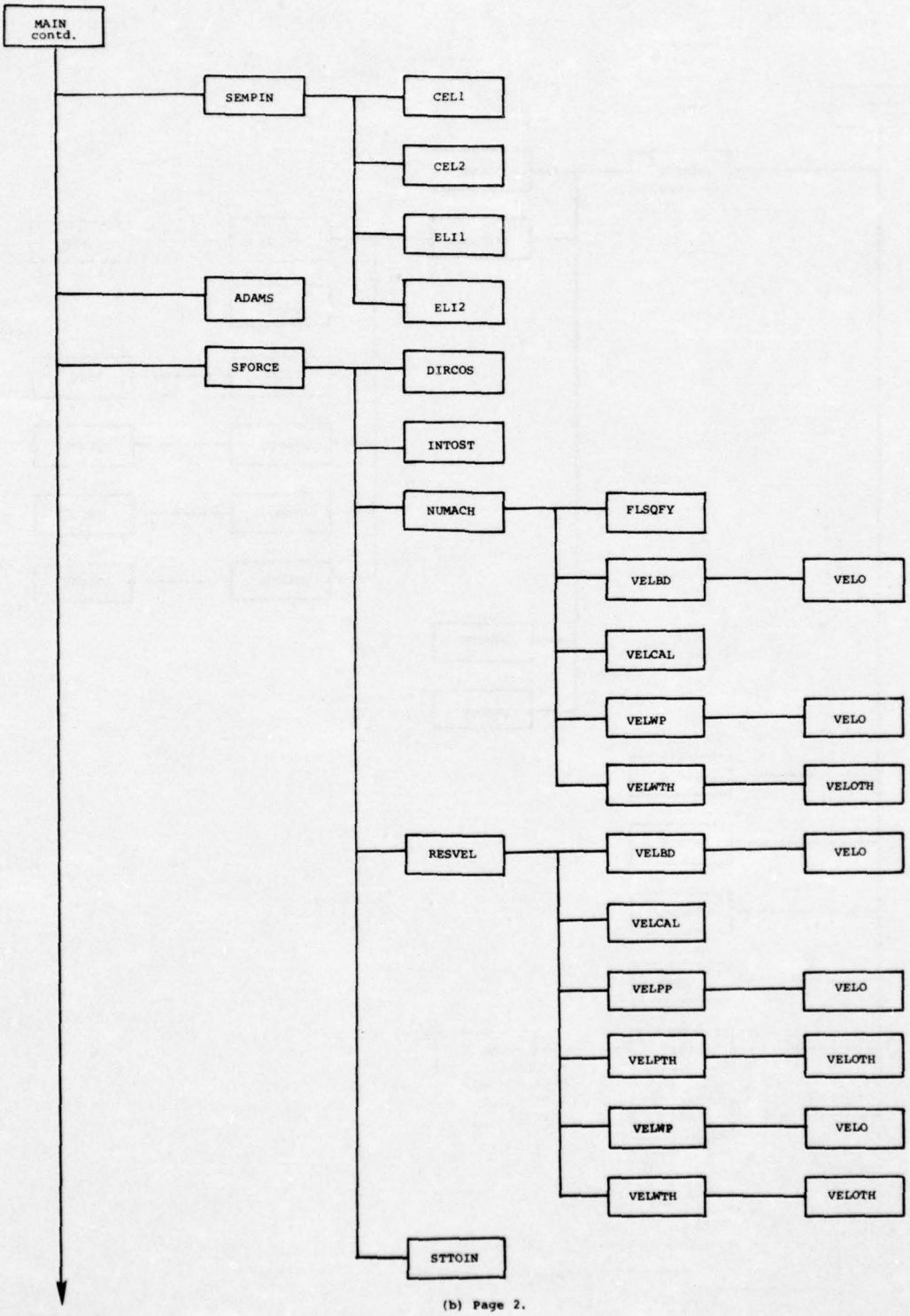


Figure I-1.- Common statements and the routines in which they appear.



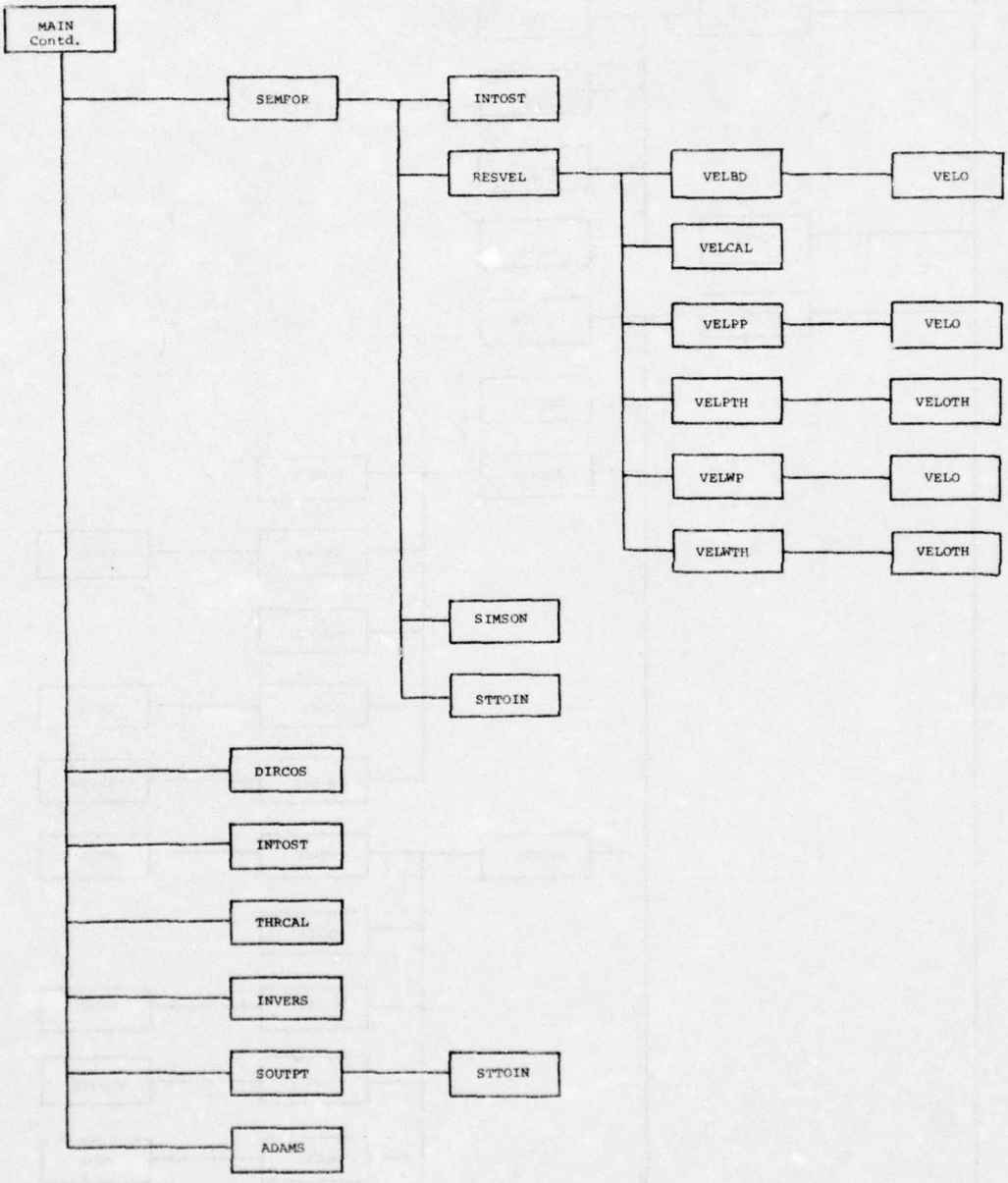
(a) Page 1.

Figure I-2.- Subroutine calling sequence.



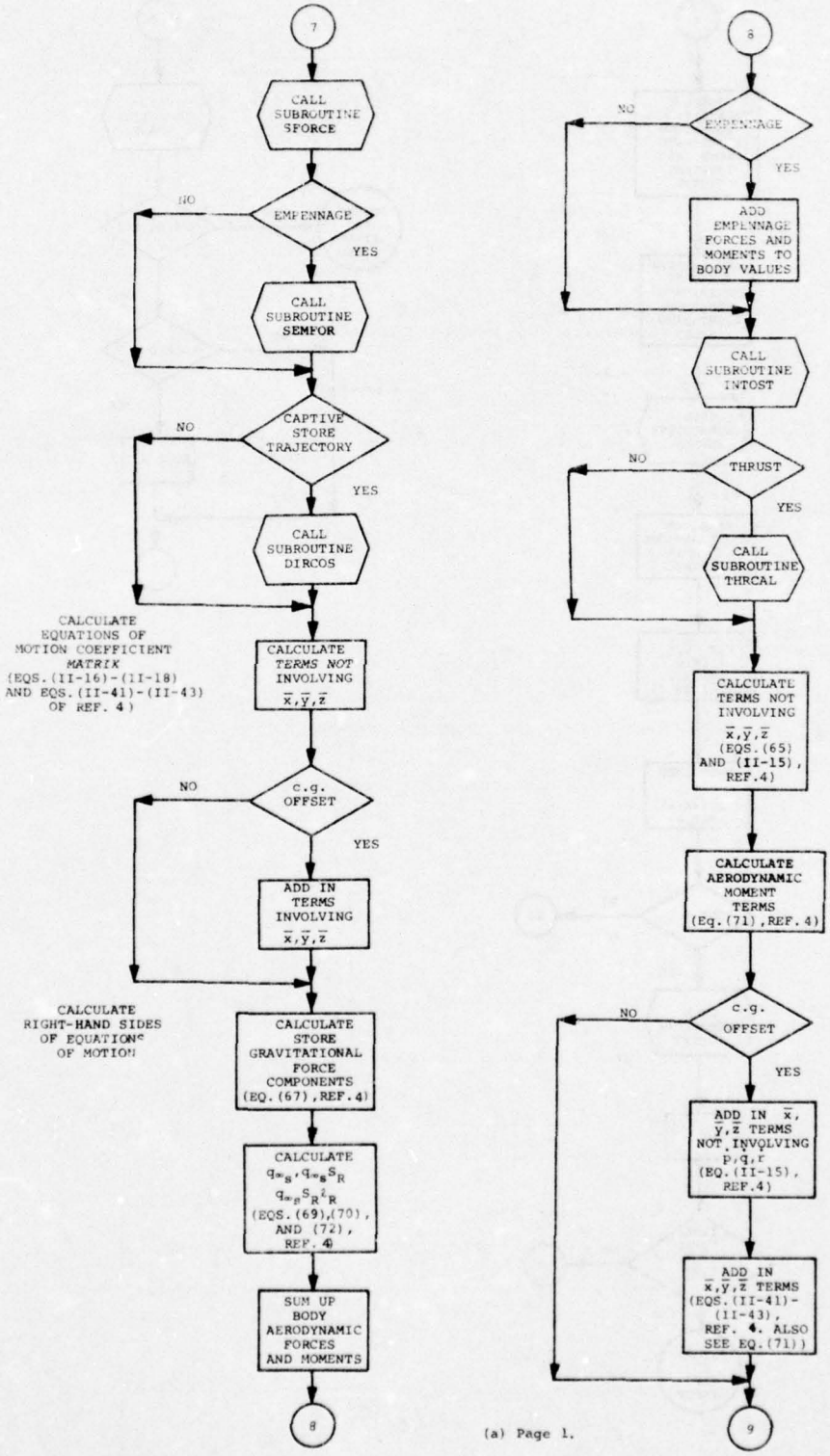
(b) Page 2.

Figure I-2.- Continued.



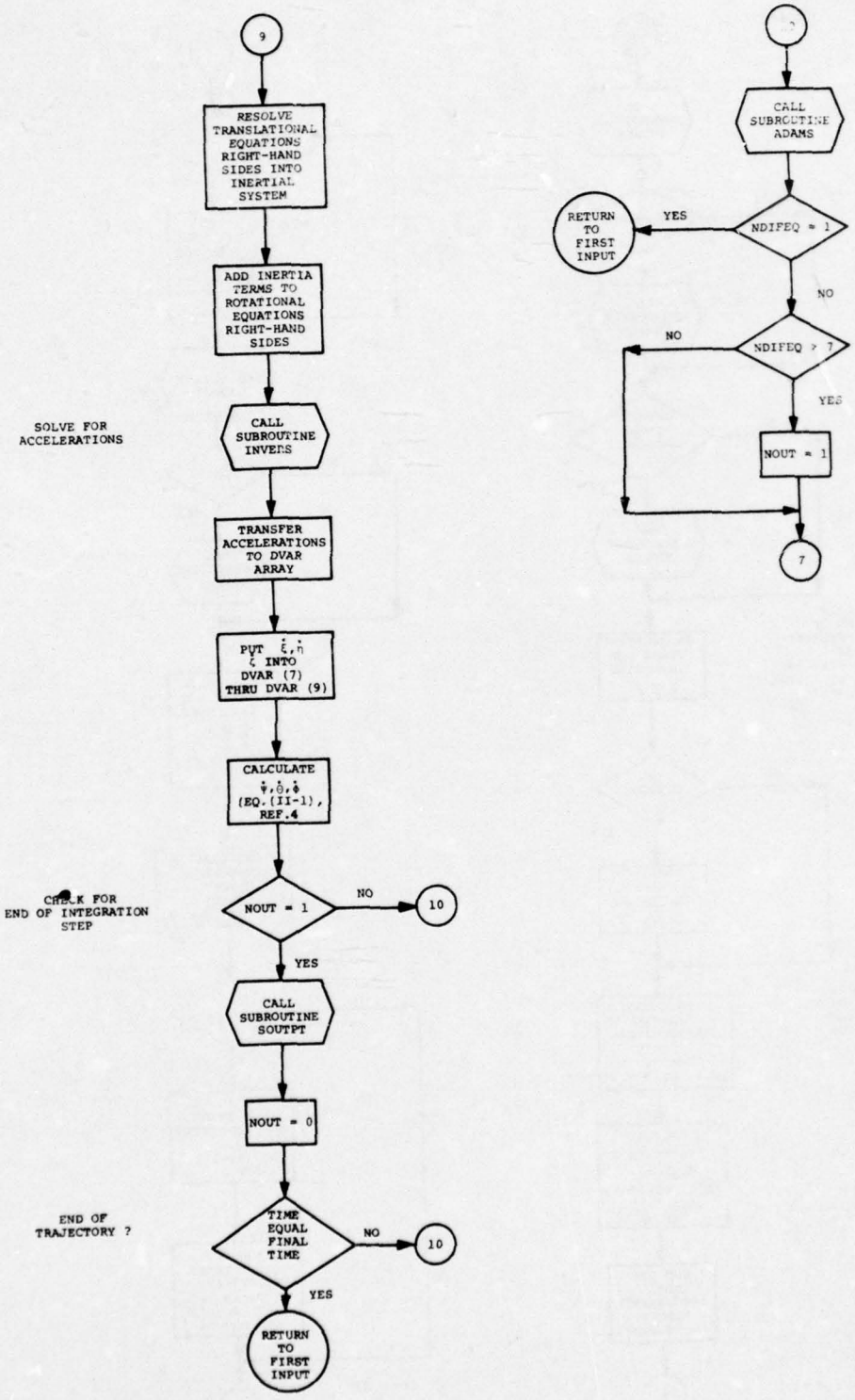
(c) Page 3.

Figure I-2.- Concluded.



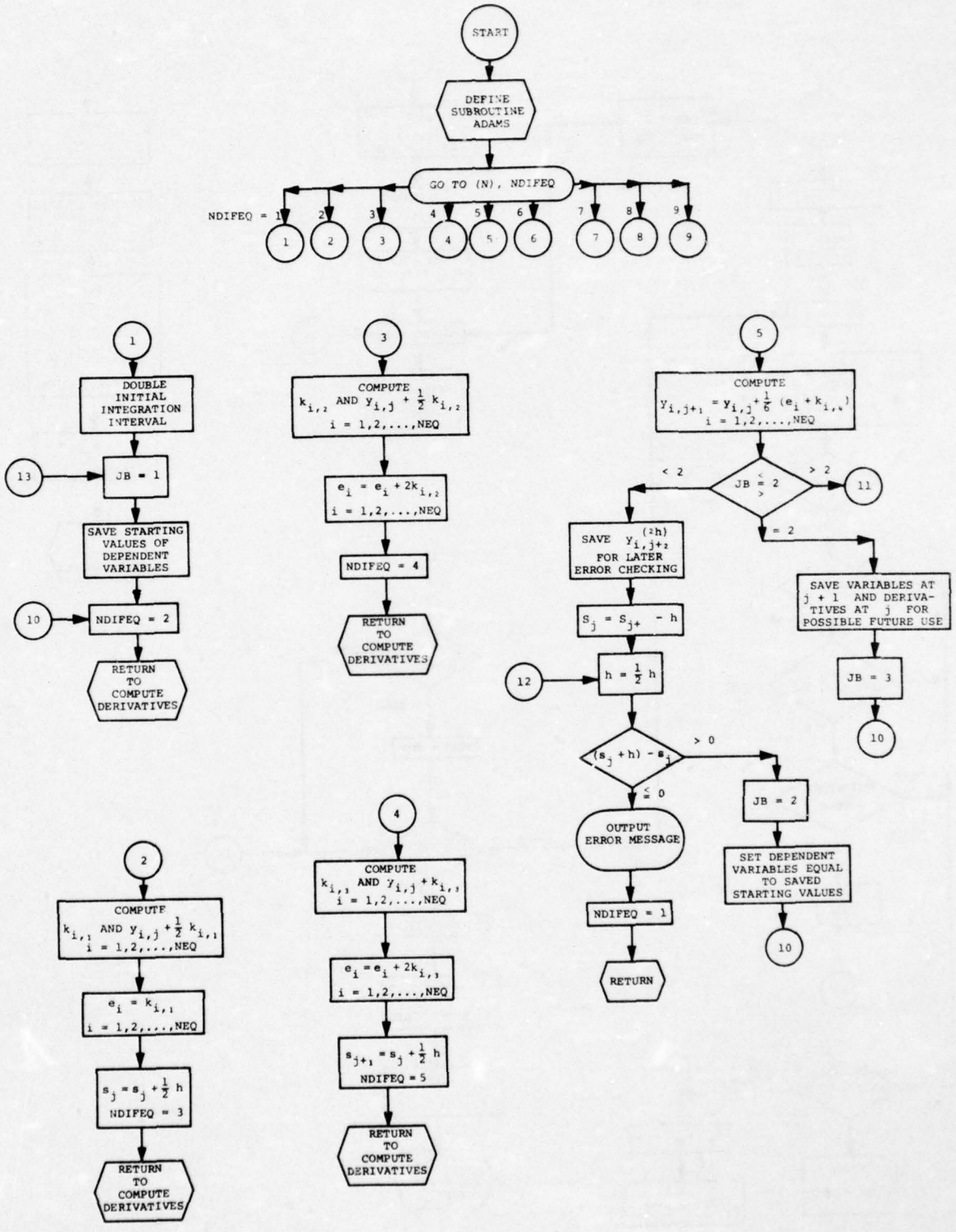
(a) Page 1.

Figure I-3.- Flow chart of integration loop of main program.



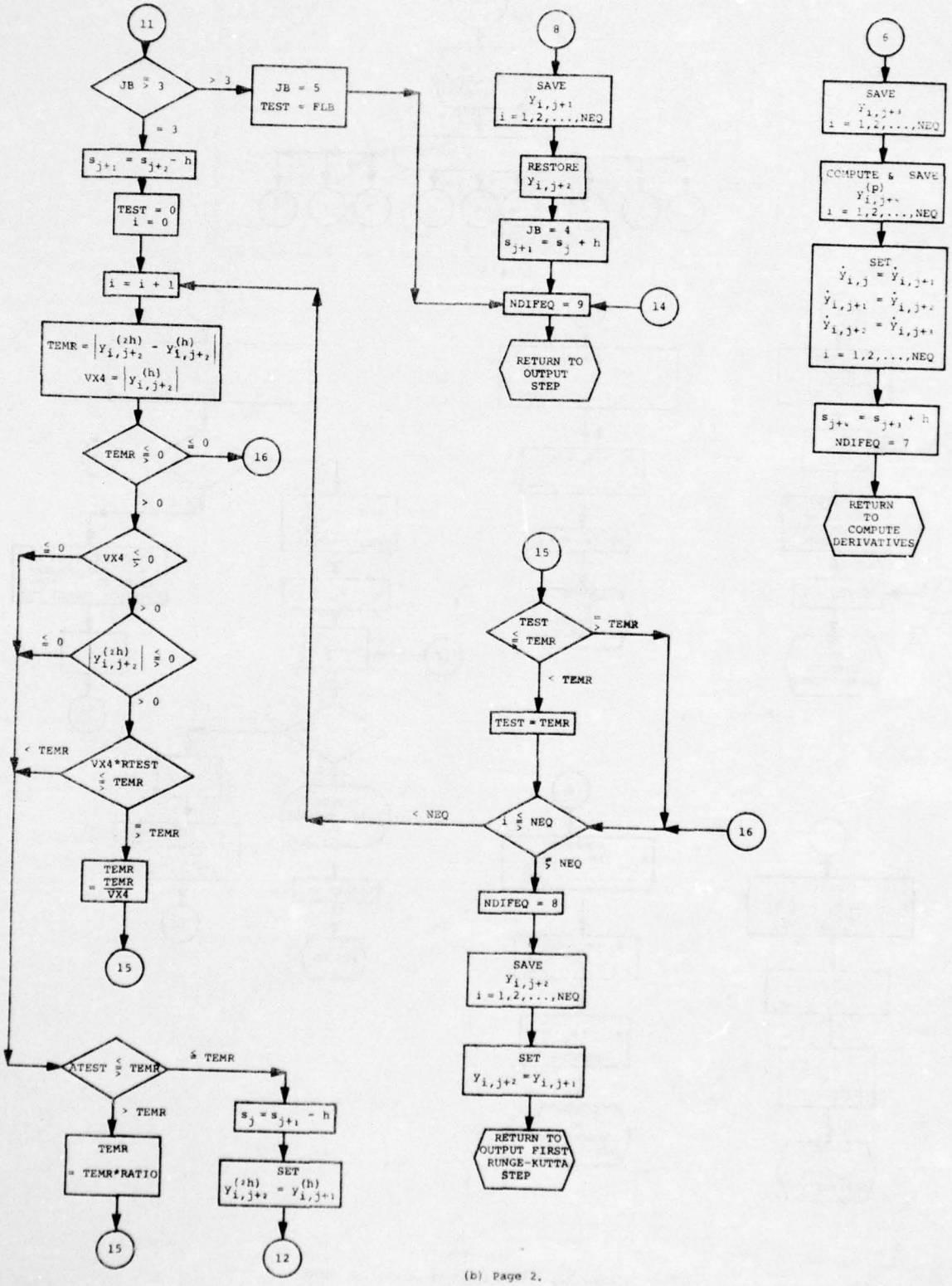
(b) Page 2.

Figure I-3.- Concluded.



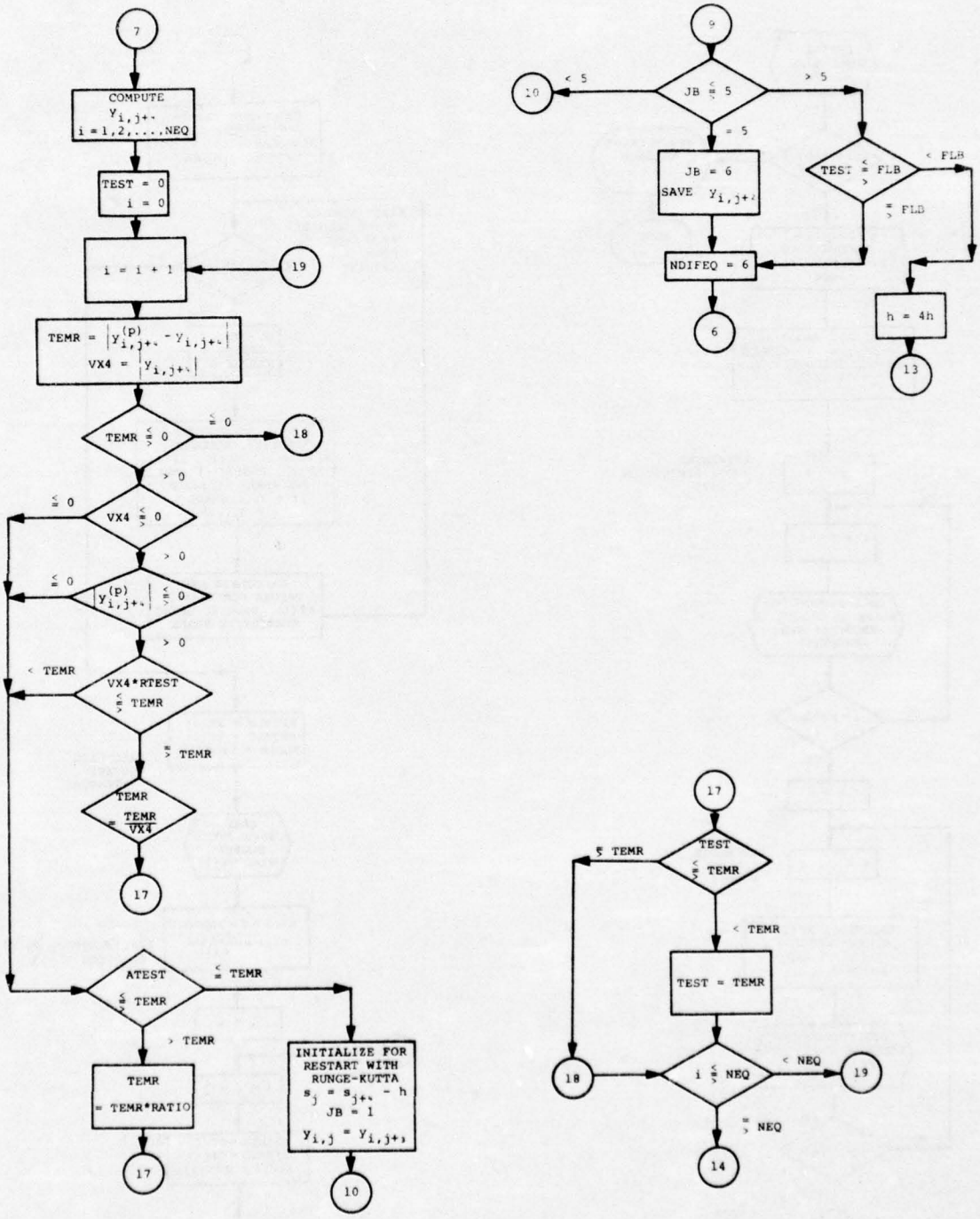
(a) Page 1.

Figure I-4.- Flow chart of subroutine ADAMS.



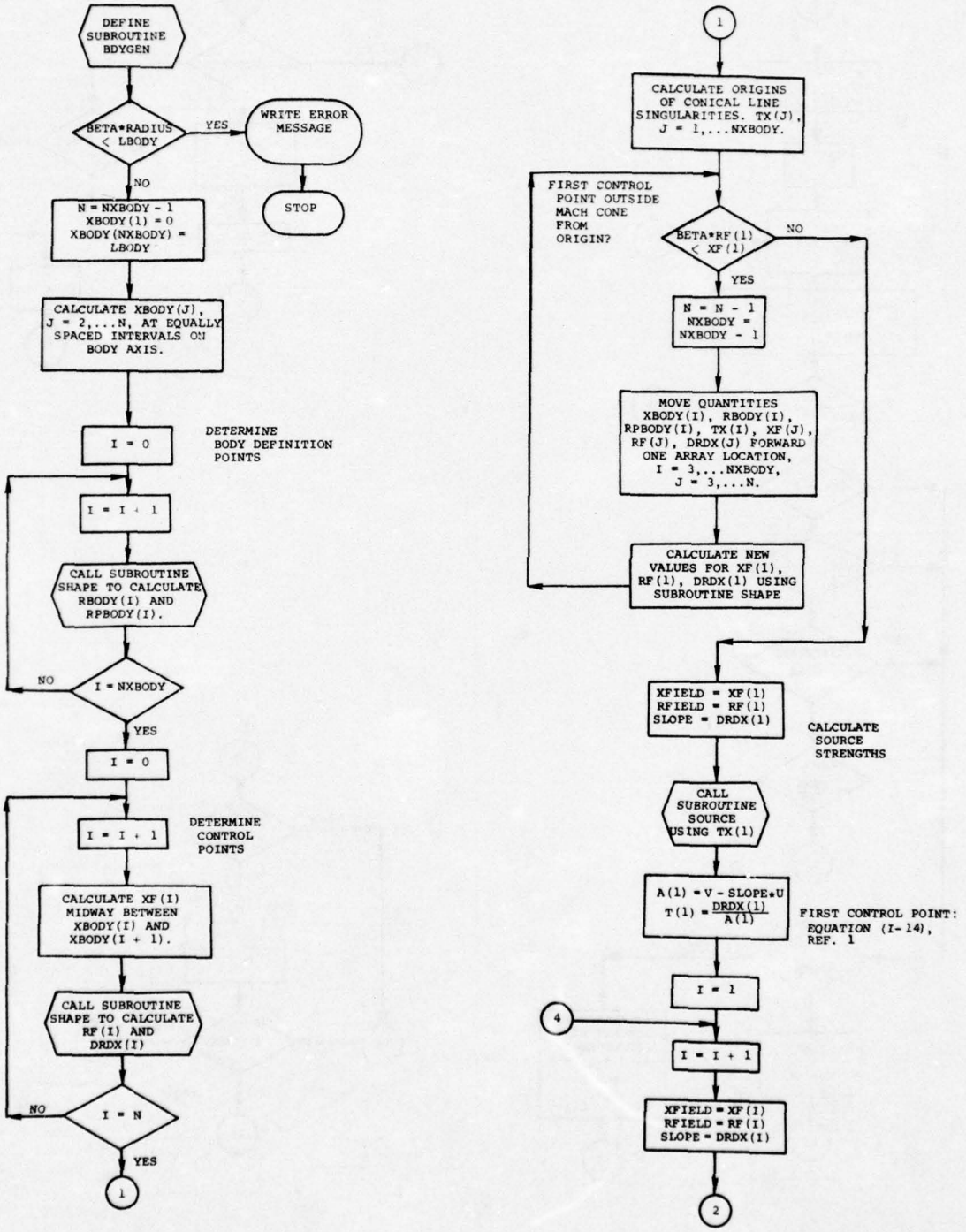
(b) Page 2.

Figure I-4.- Continued.



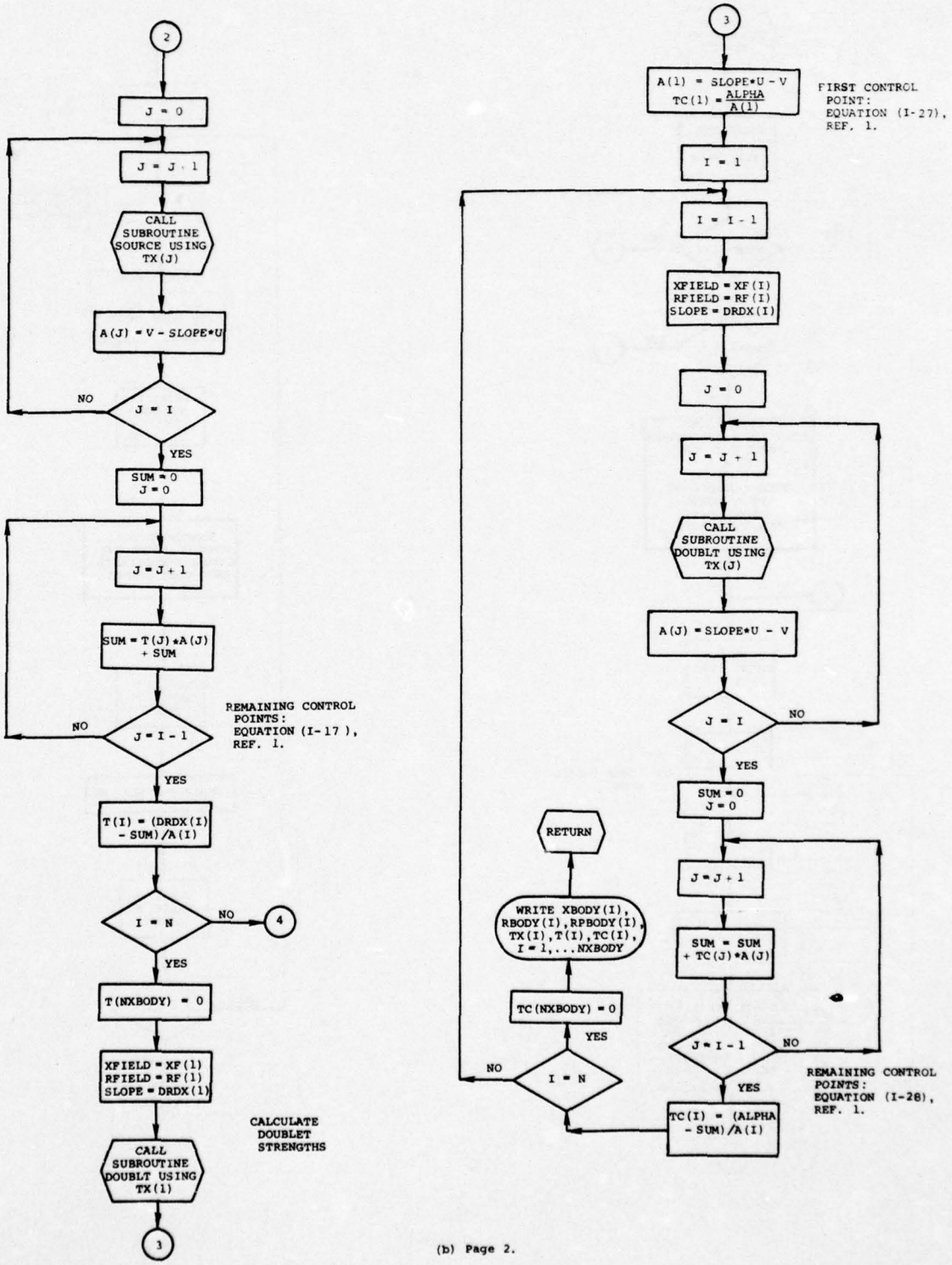
(c) Page 3.

Figure I-4. - Concluded.



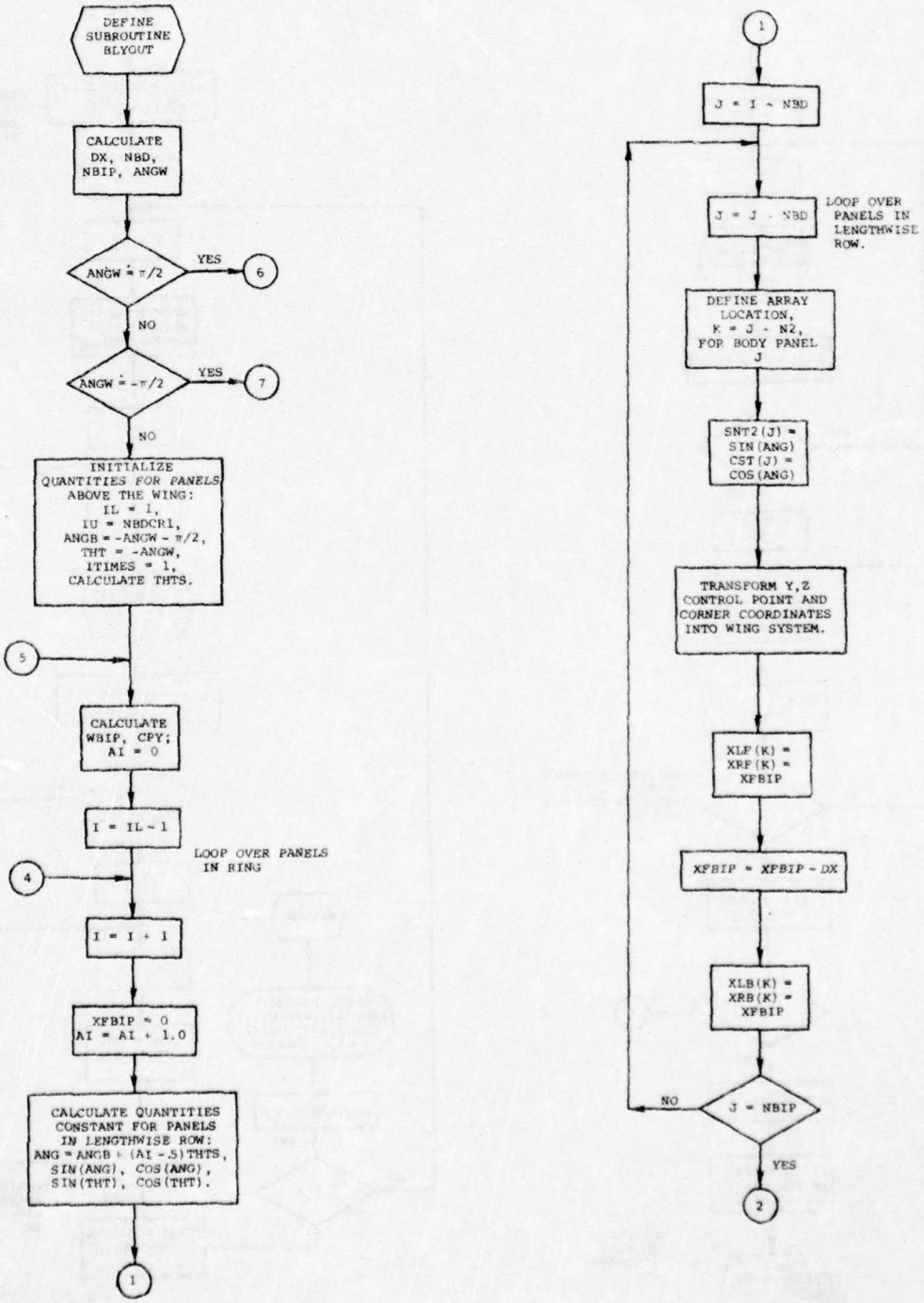
(a) Page 1.

Figure I-5.- Flow chart of su_routine BDYGEN.



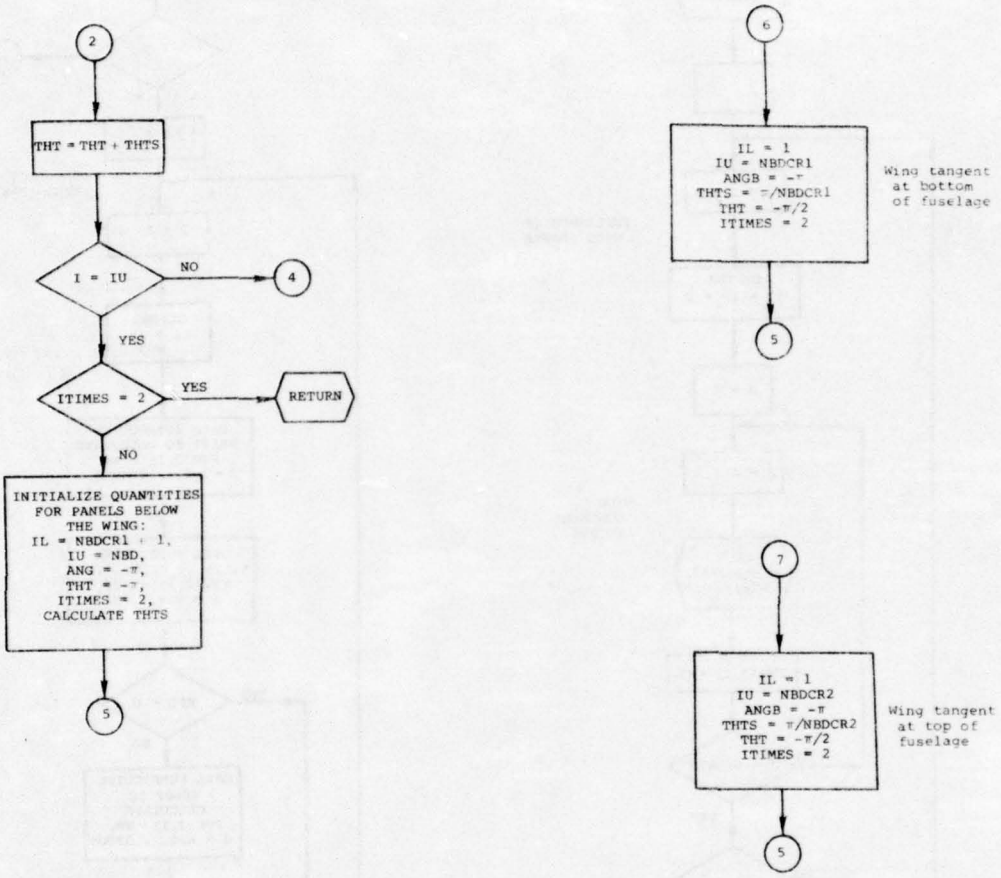
(b) Page 2.

Figure I-5.- Concluded.



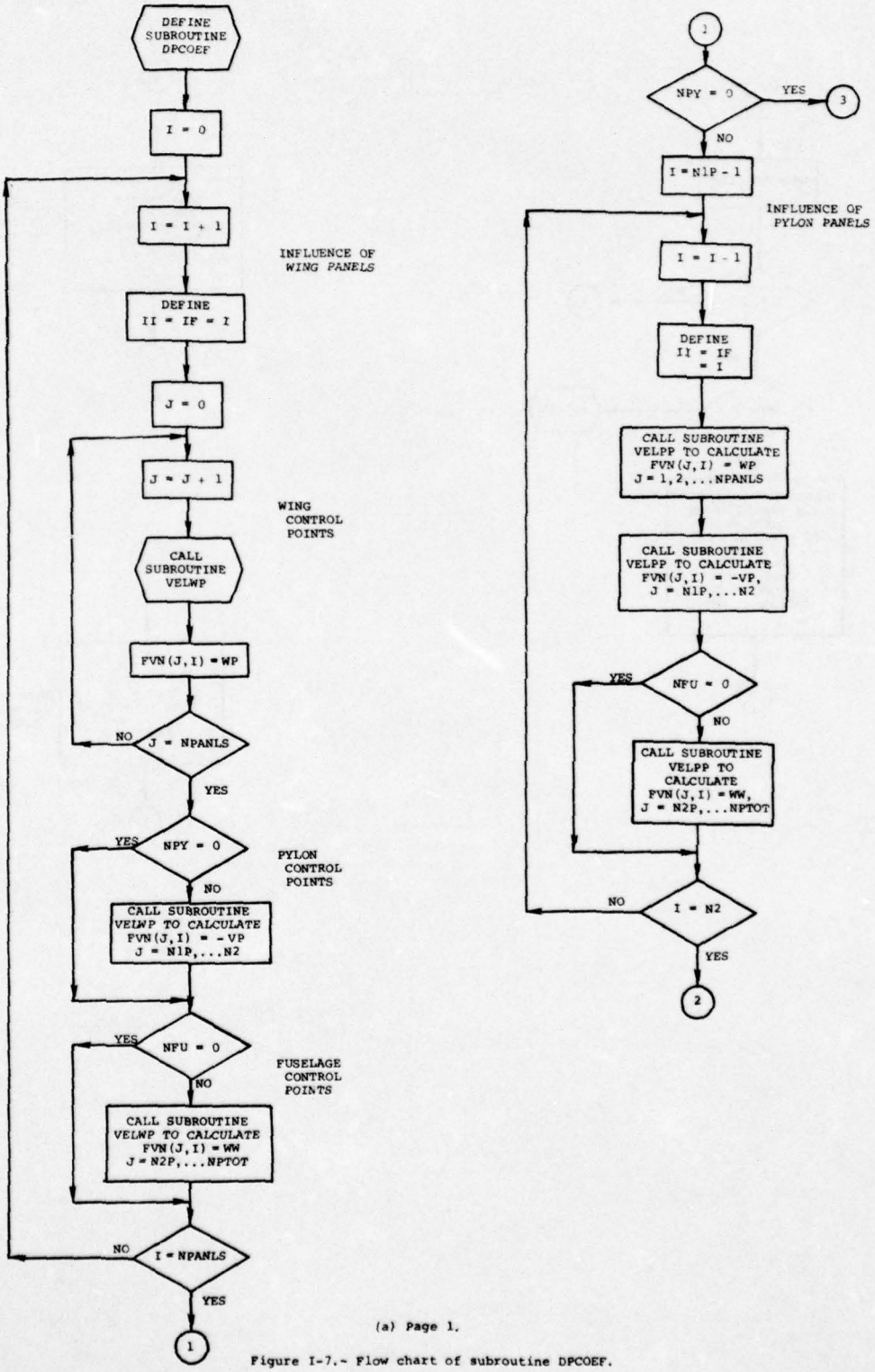
(a) Page 1.

Figure I-6.- Flow chart of subroutine BLYOUT.



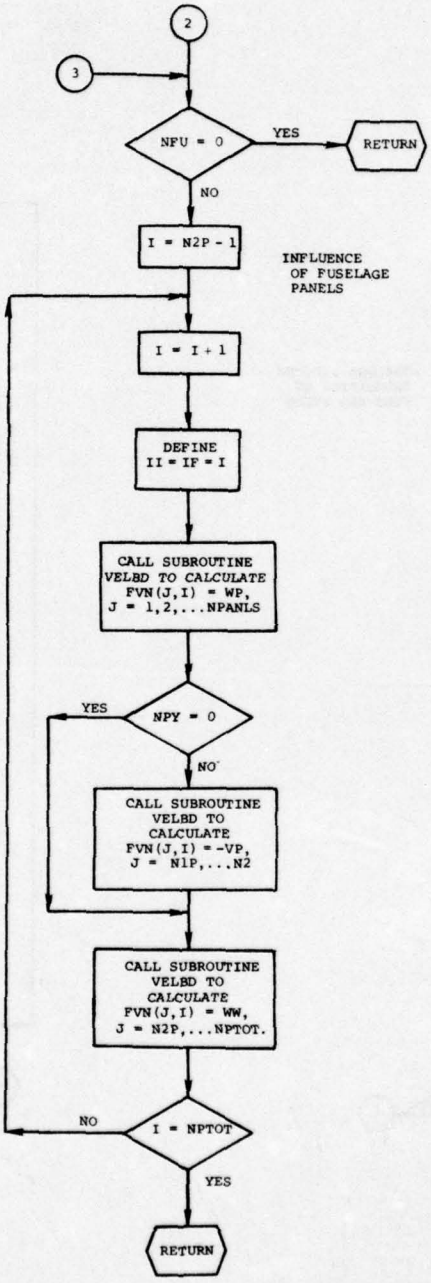
(b) Page 2.

Figure I-6.- Concluded.



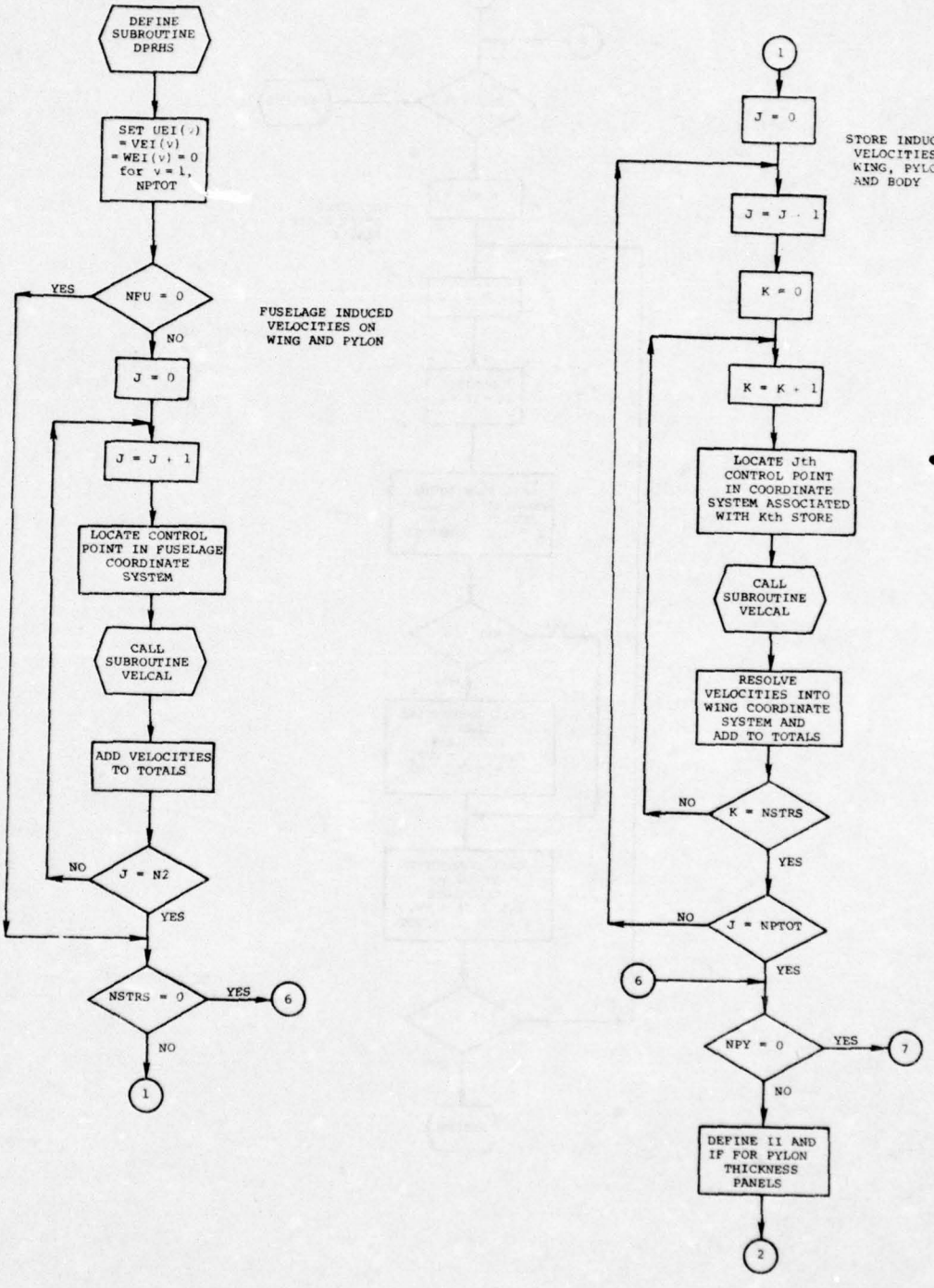
(a) Page 1.

Figure I-7.- Flow chart of subroutine DPCOEF.



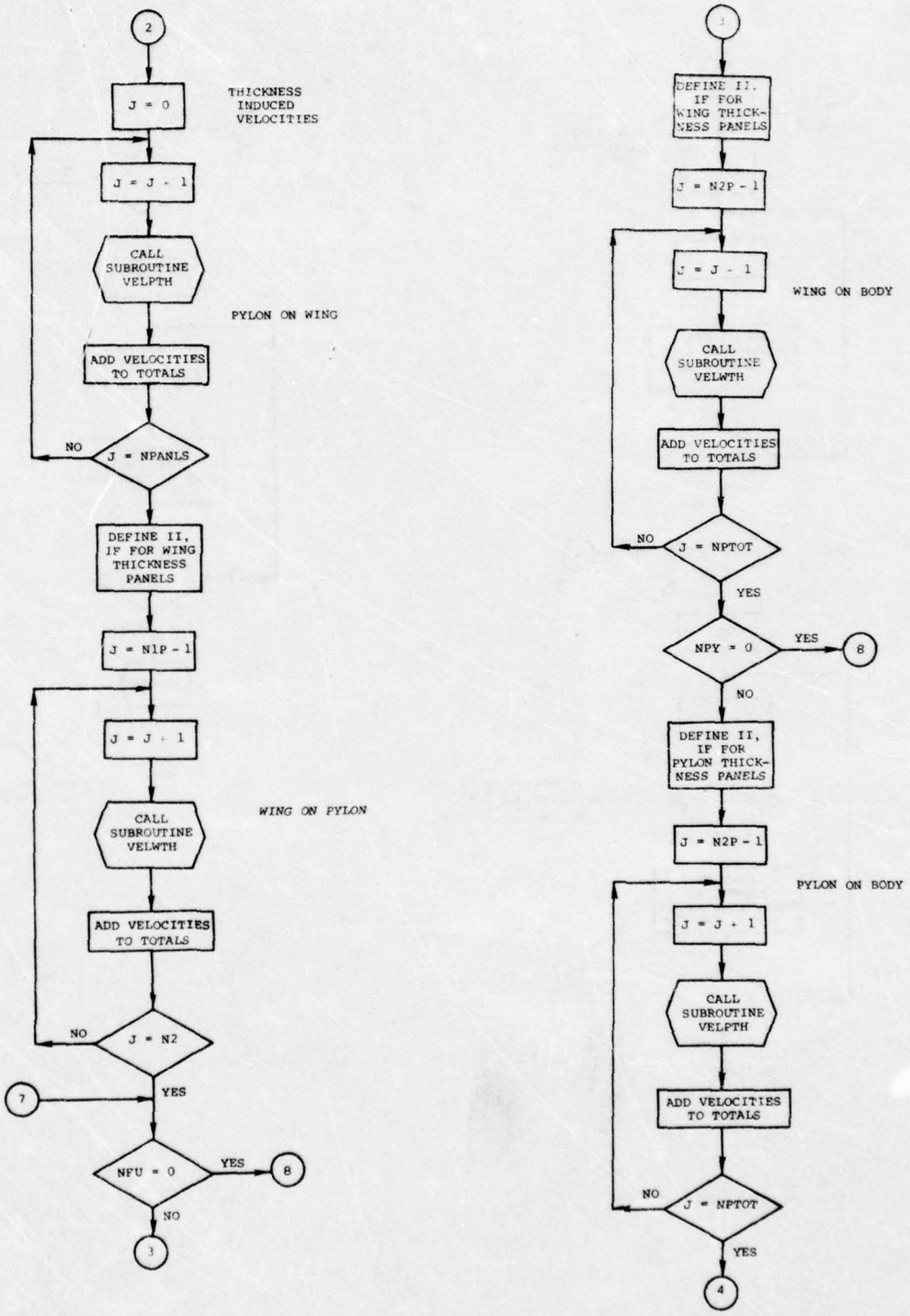
(b) Page 2.

Figure I-7.- Concluded.



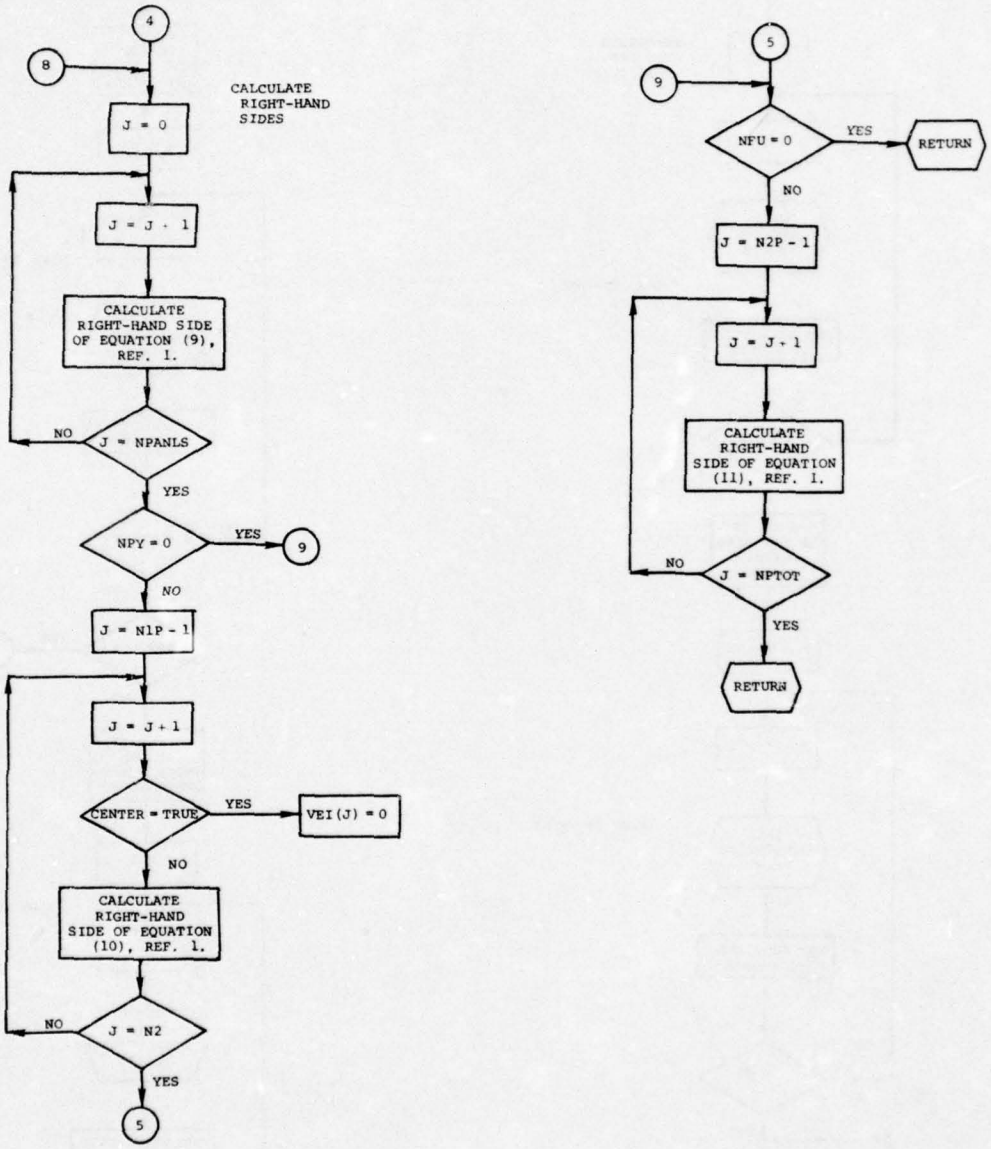
(a) Page 1.

Figure I-8.- Flow chart of subroutine DPRHS.



(b) Page 2.

Figure I-8.- Continued.



(c) Page 3.

Figure I-8.- Concluded.

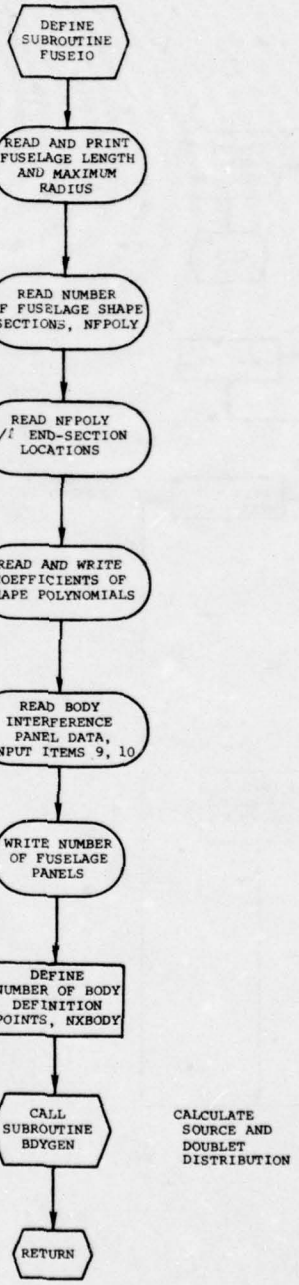
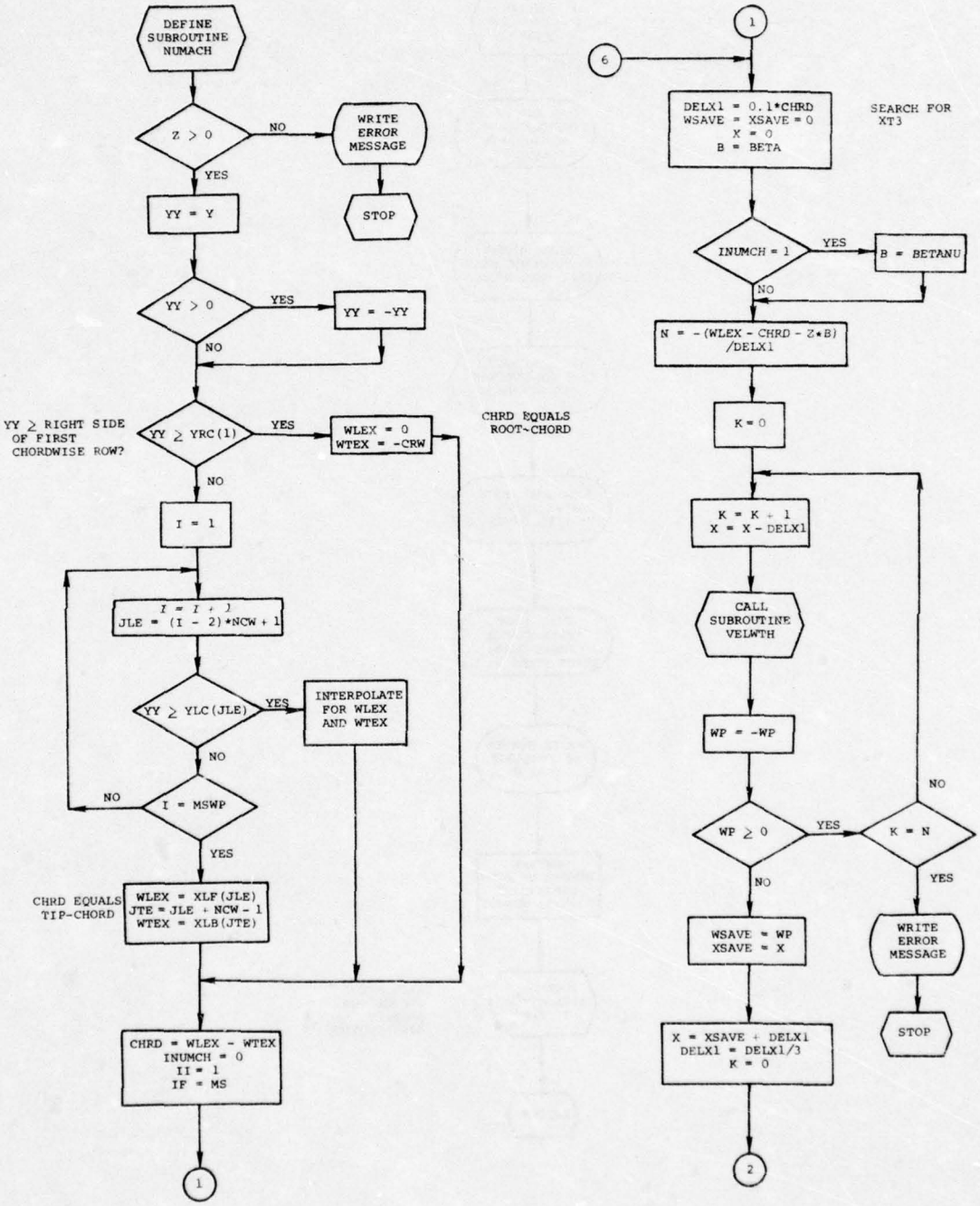
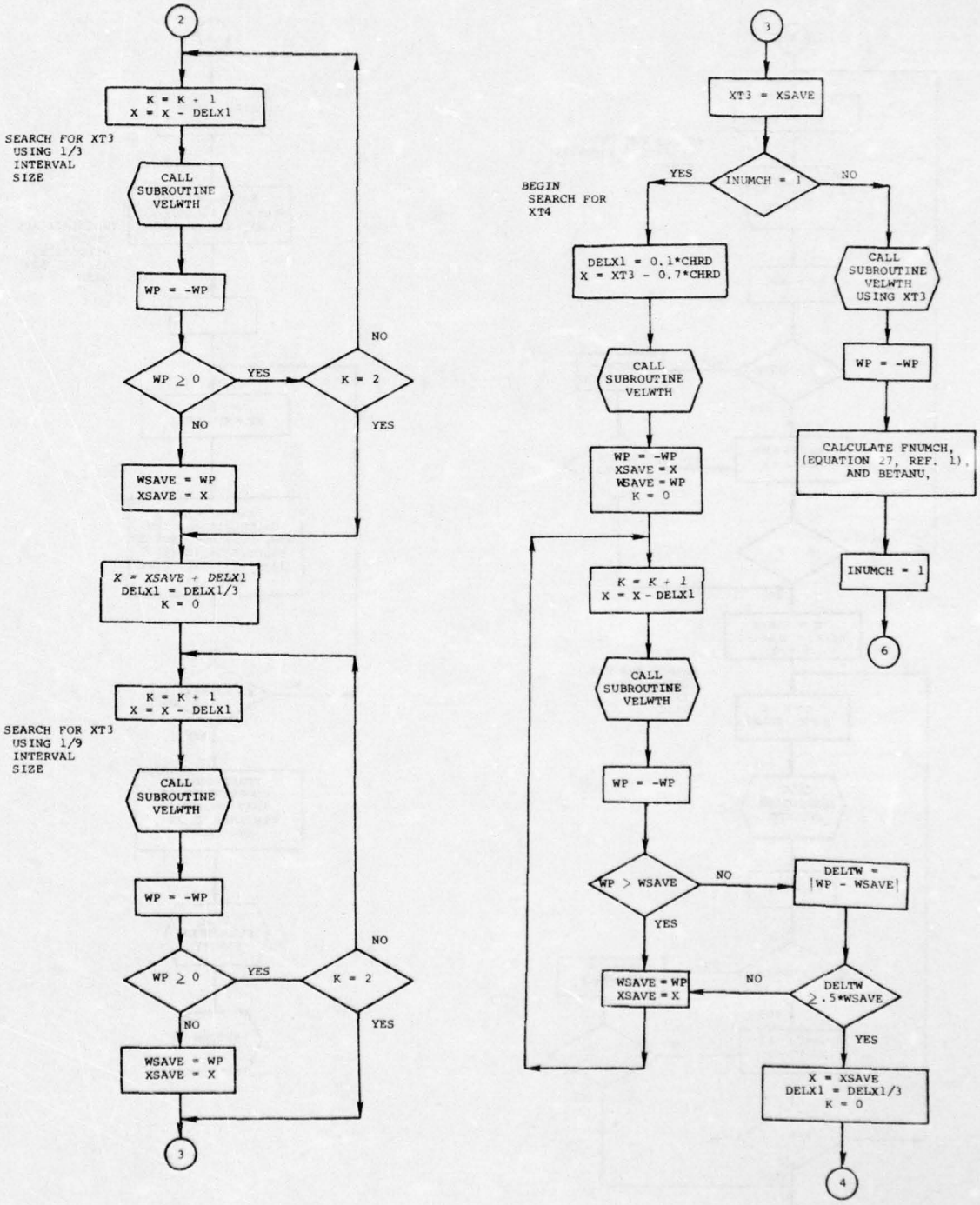


Figure I-9.- Flow chart of subroutine FUSEIO.



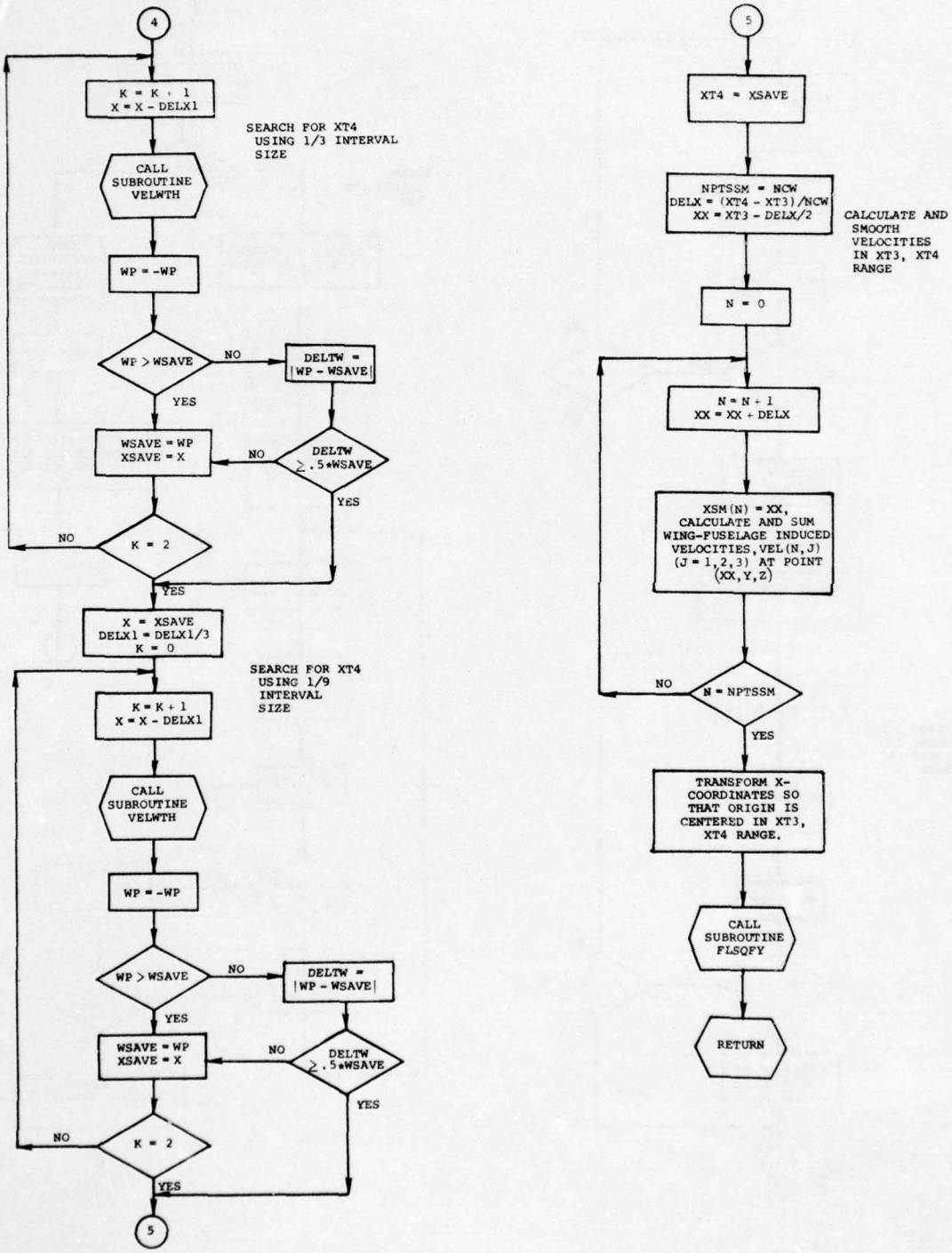
(a) Page 1.

Figure I-10.- Flow chart of subroutine NUMACH.



(b) Page 2.

Figure I-10.- Continued.



(c) Page 3.

Figure I-10.- Concluded.

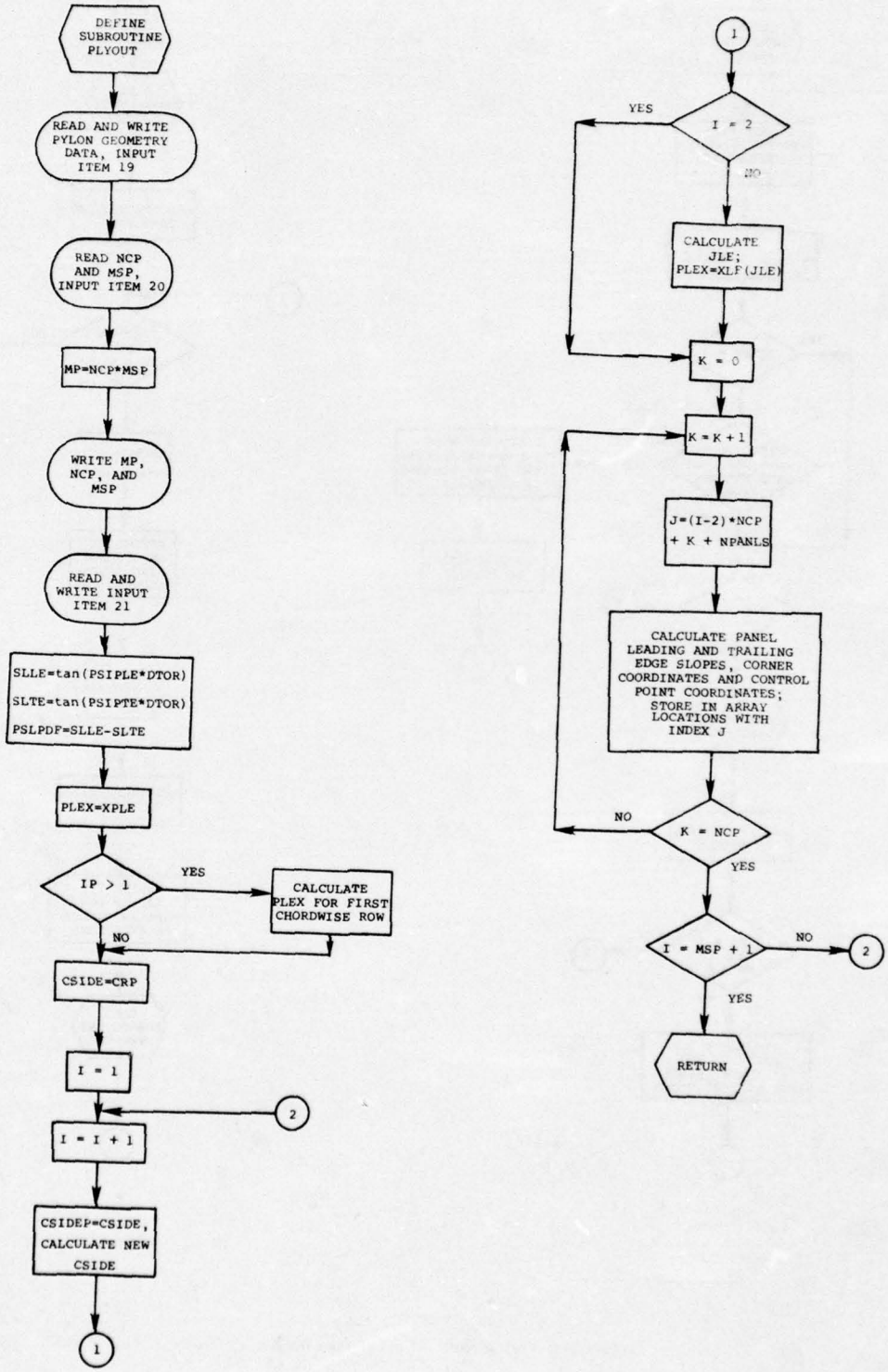
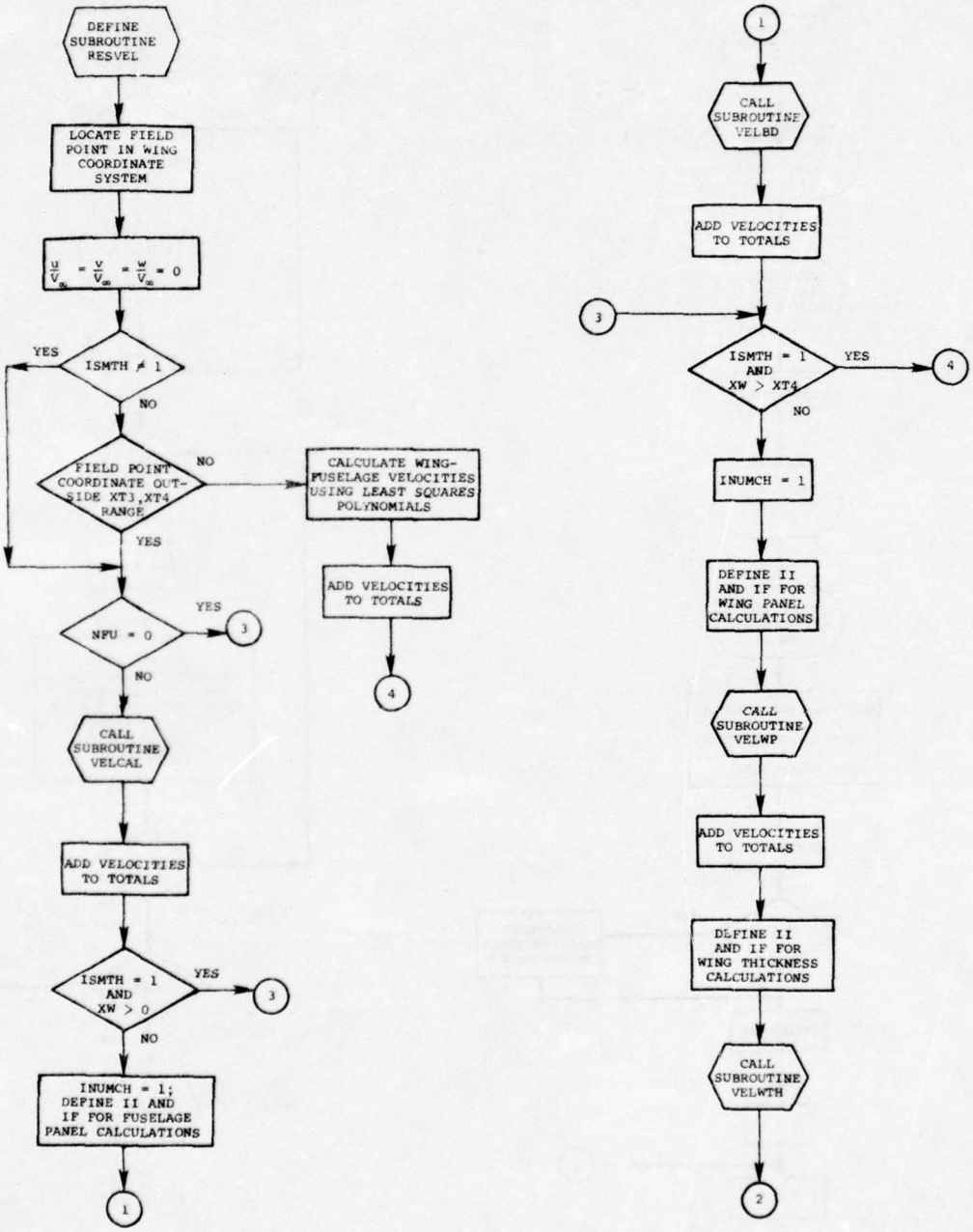
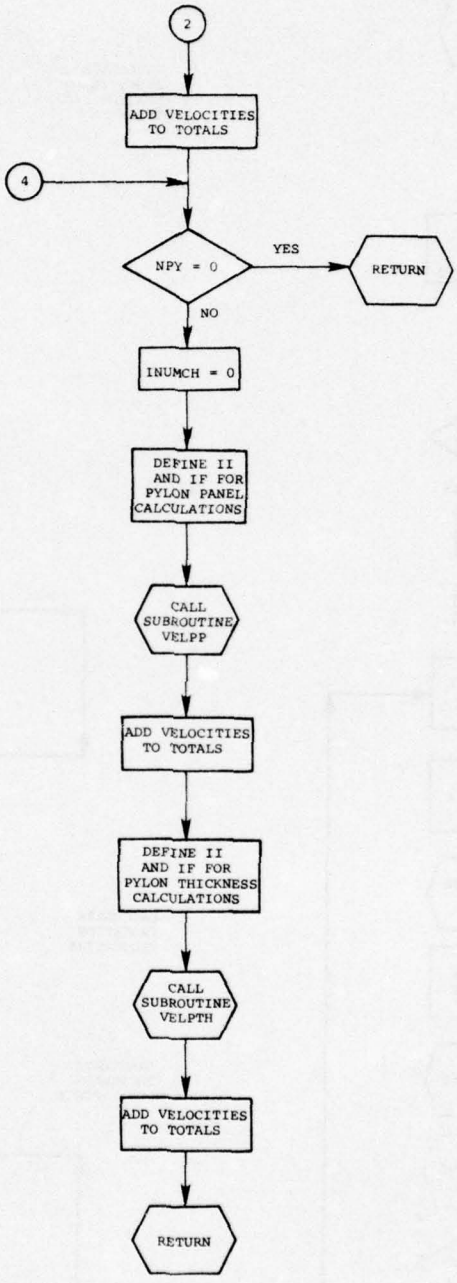


Figure I-11.- Flow chart of subroutine PLYOUT.



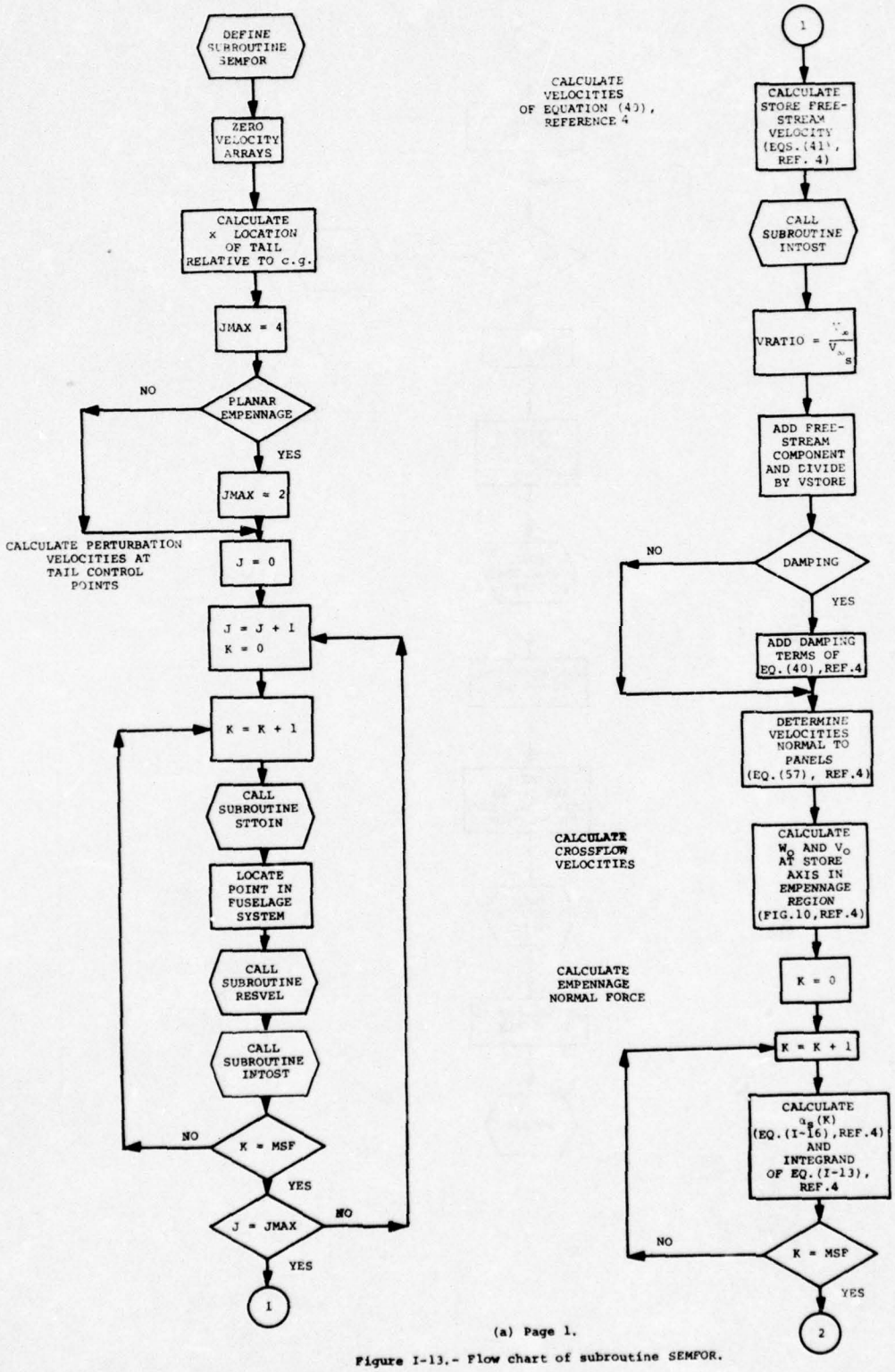
(a) Page 1.

Figure I-12.- Flow chart of subroutine RESVEL.



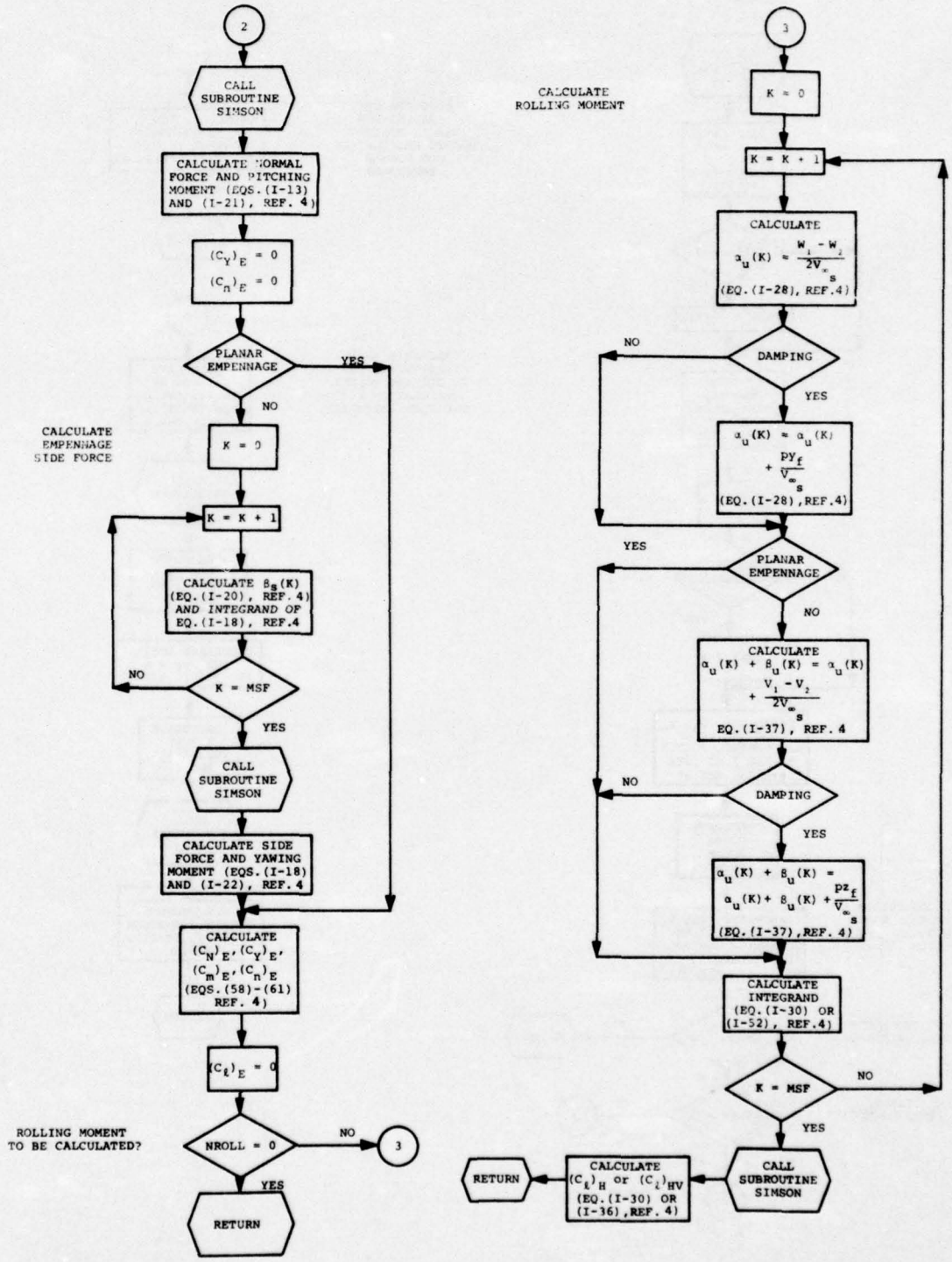
(b) Page 2.

Figure I-12.- Concluded.



(a) Page 1.

Figure I-13.- Flow chart of subroutine SEMFOR.



(b) Page 2.

Figure I-13.- Concluded.

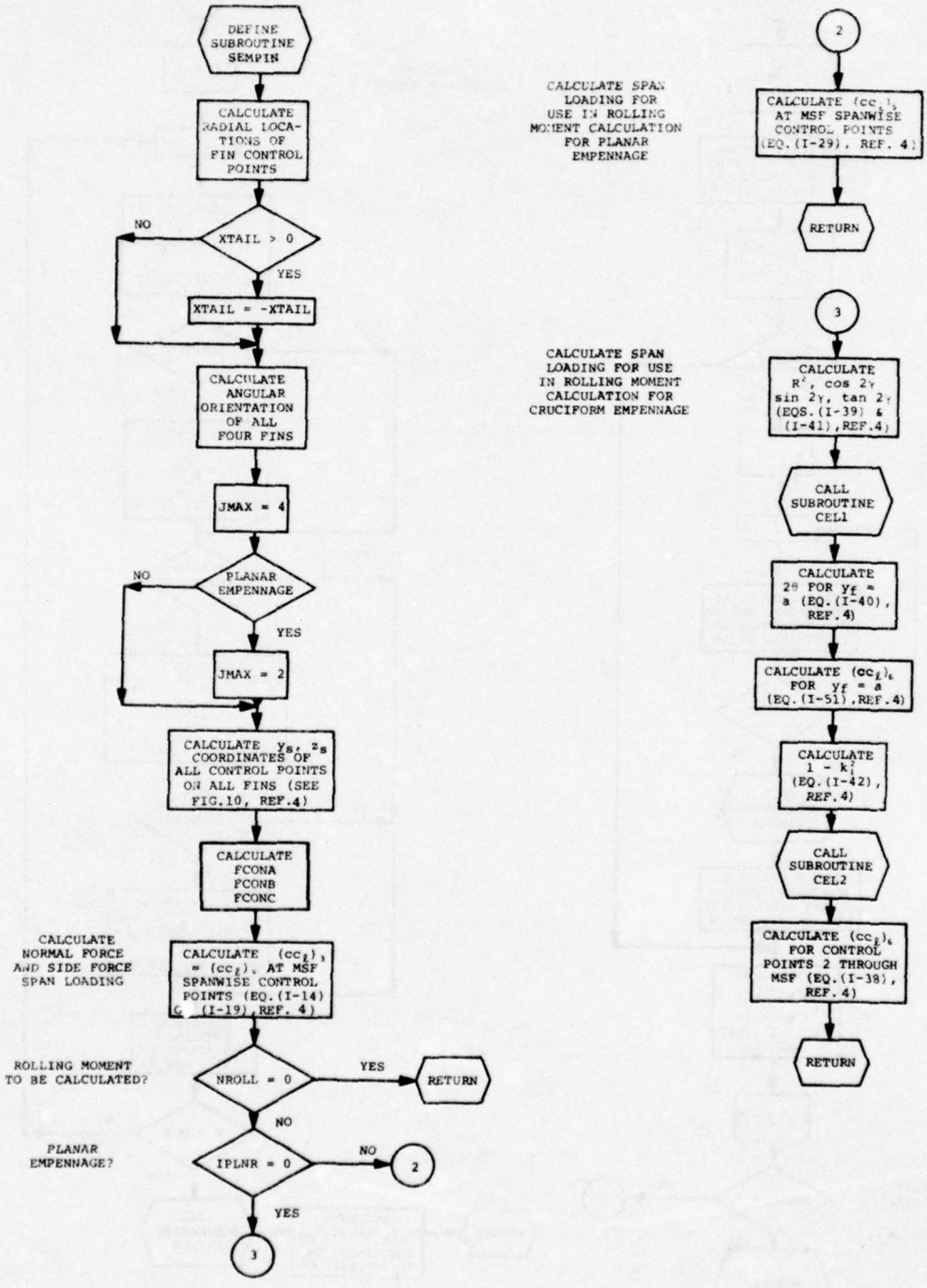
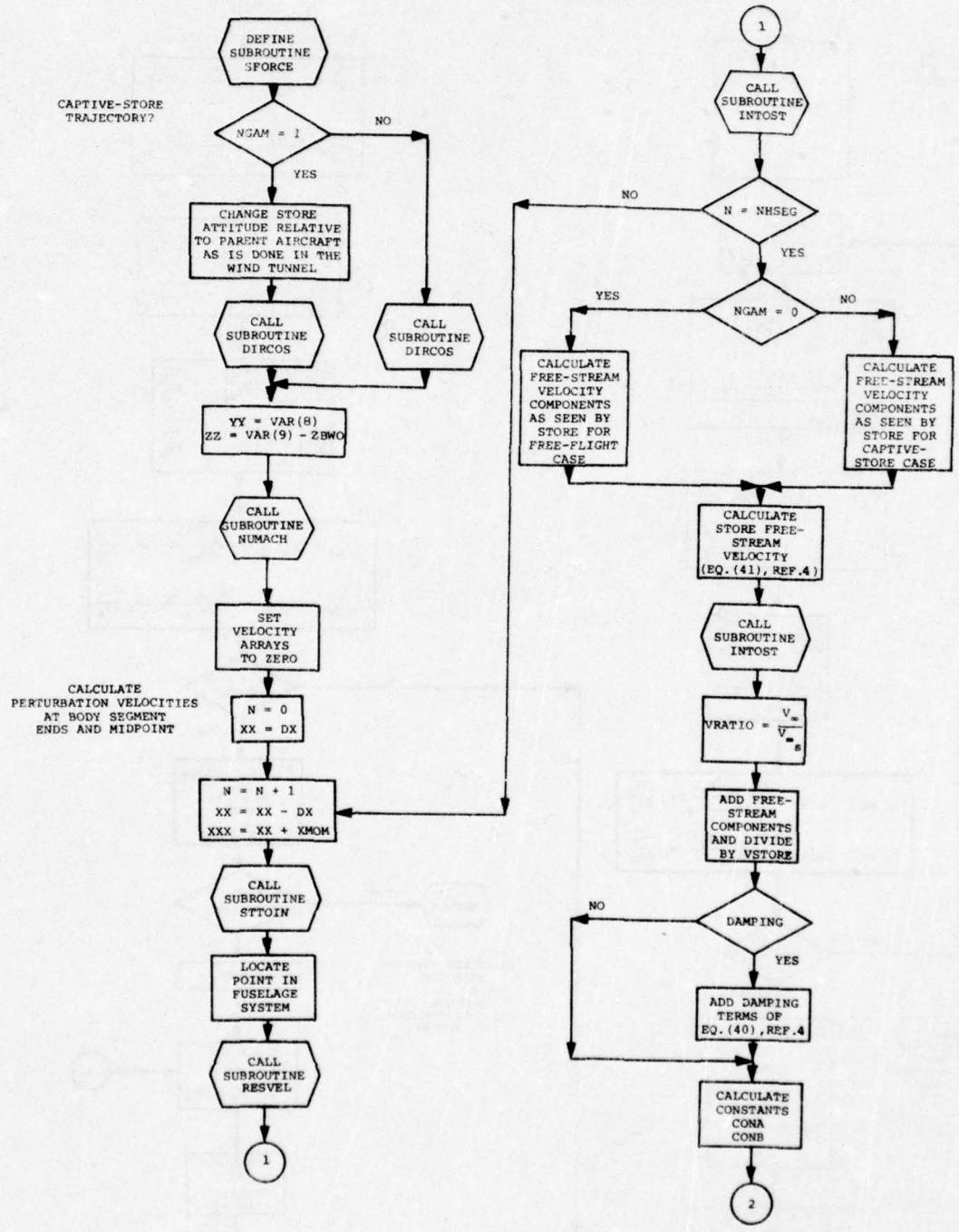
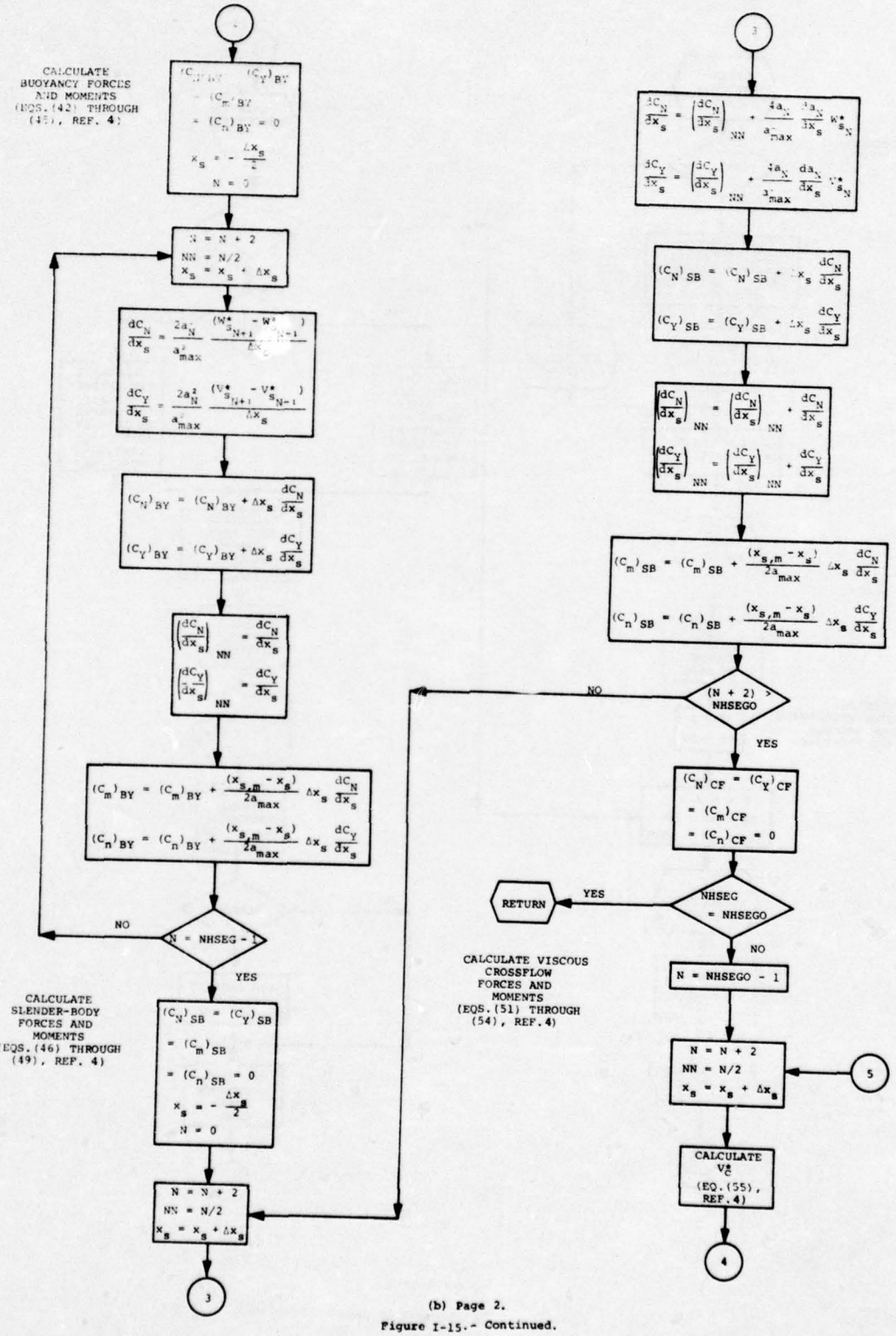


Figure I-14.- Flow chart of subroutine SEMPIN.



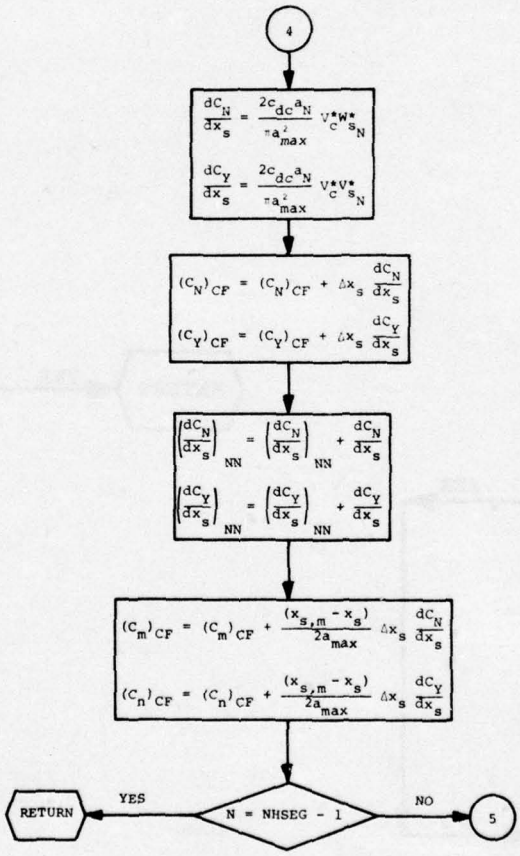
(a) Page 1.

Figure I-15.- Flow chart of subroutine SFORCE.



(b) Page 2.

Figure I-15.- Continued.



(c) Page 3.

Figure I-15.- Concluded.

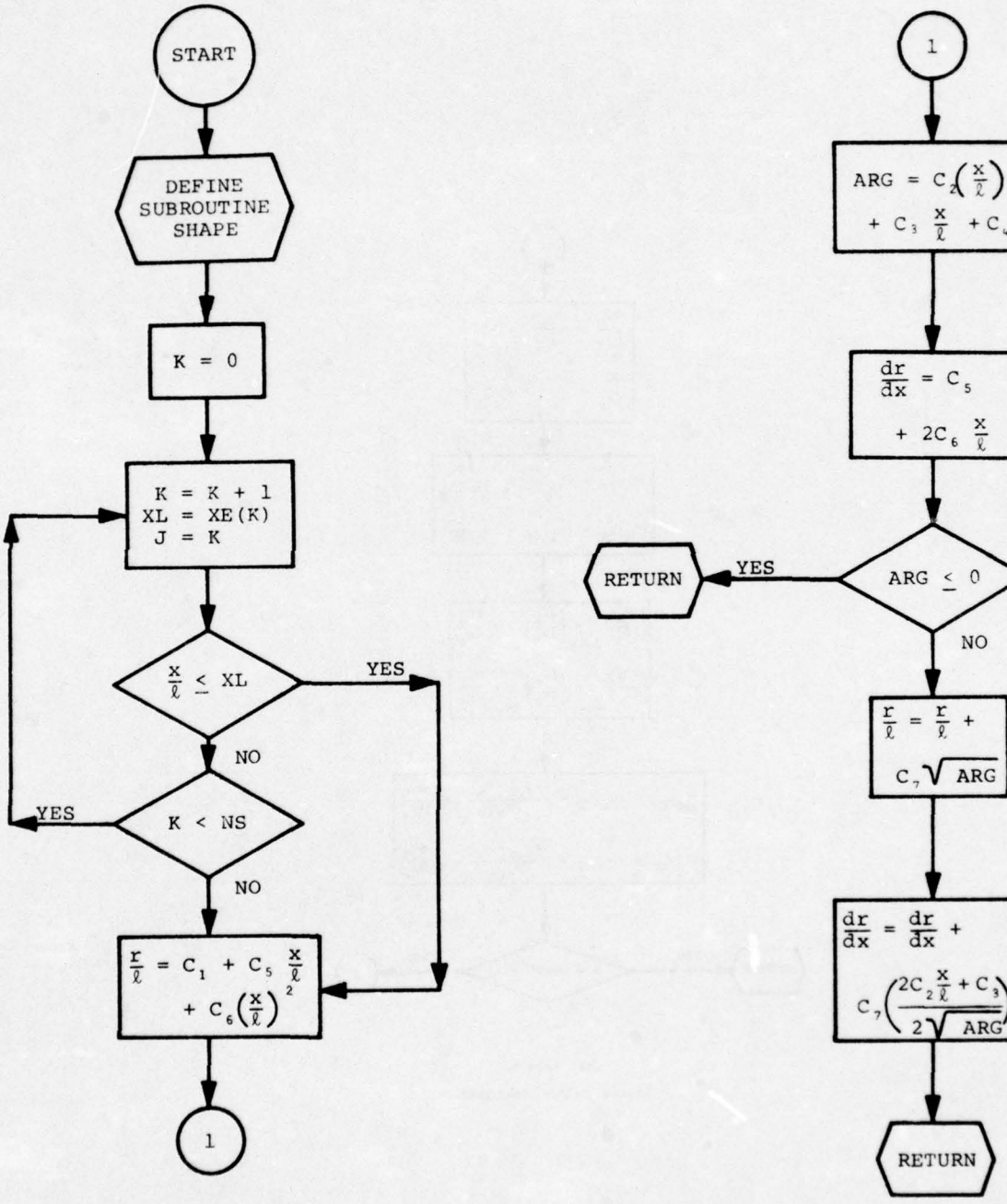


Figure I-16.-Flow chart of subroutine SHAPE.

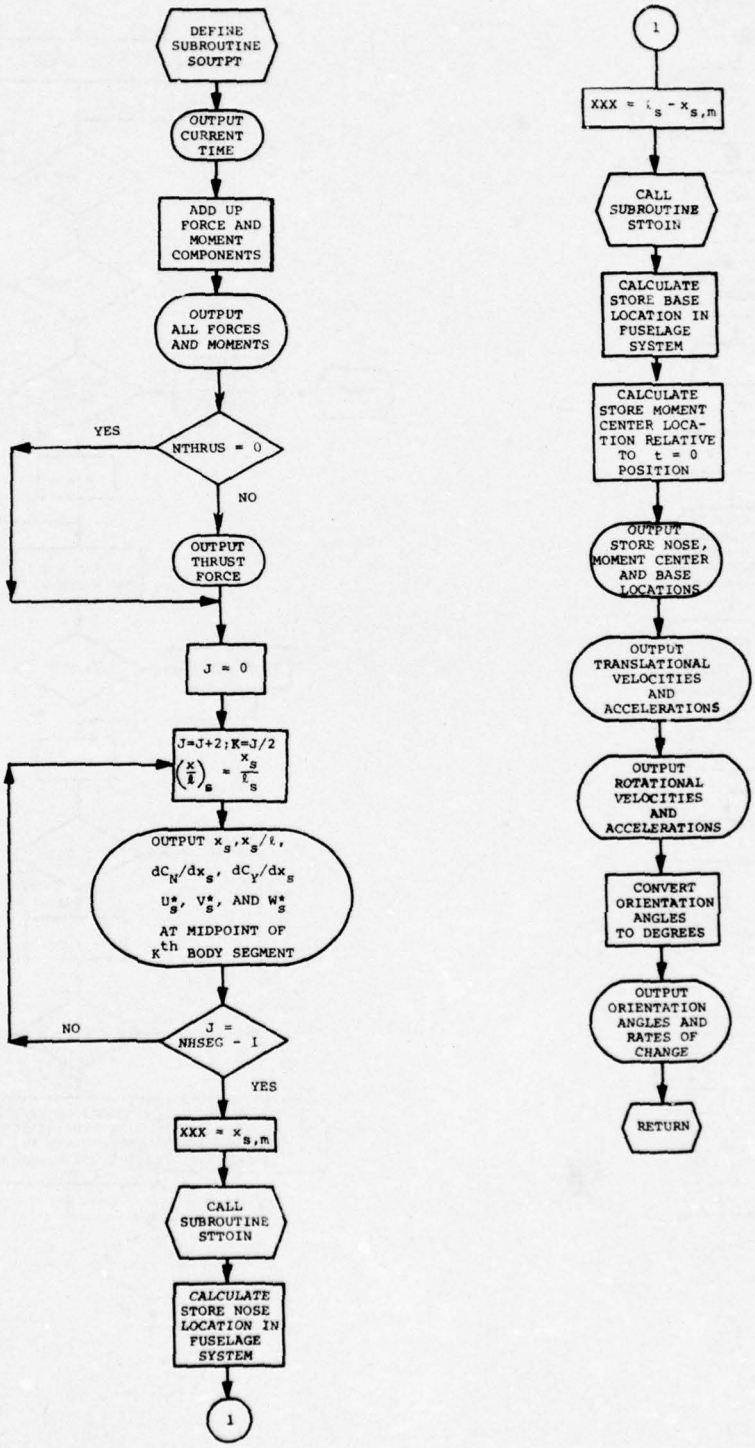


Figure I-17.- Flow chart of subroutine SOUTPT.

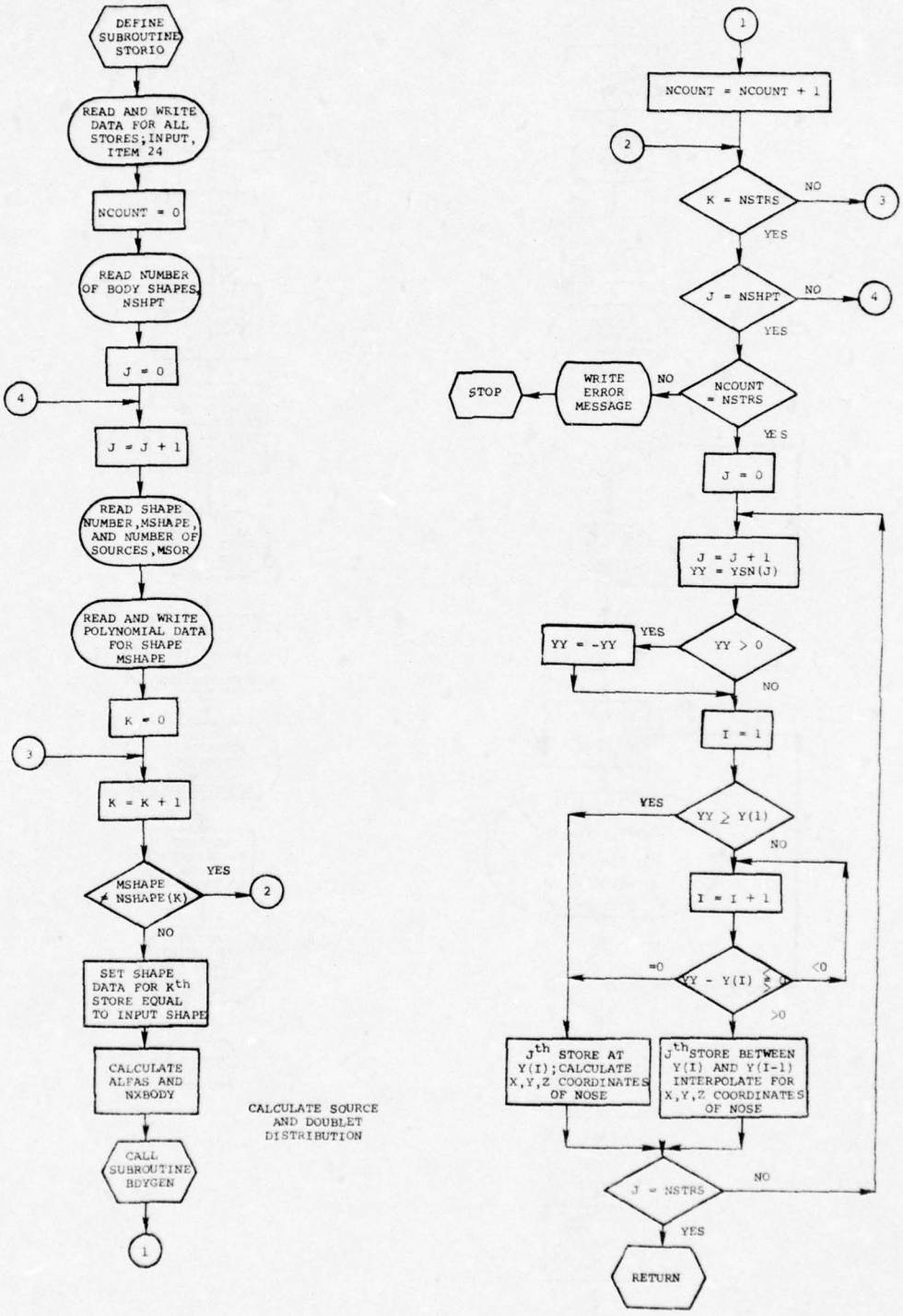
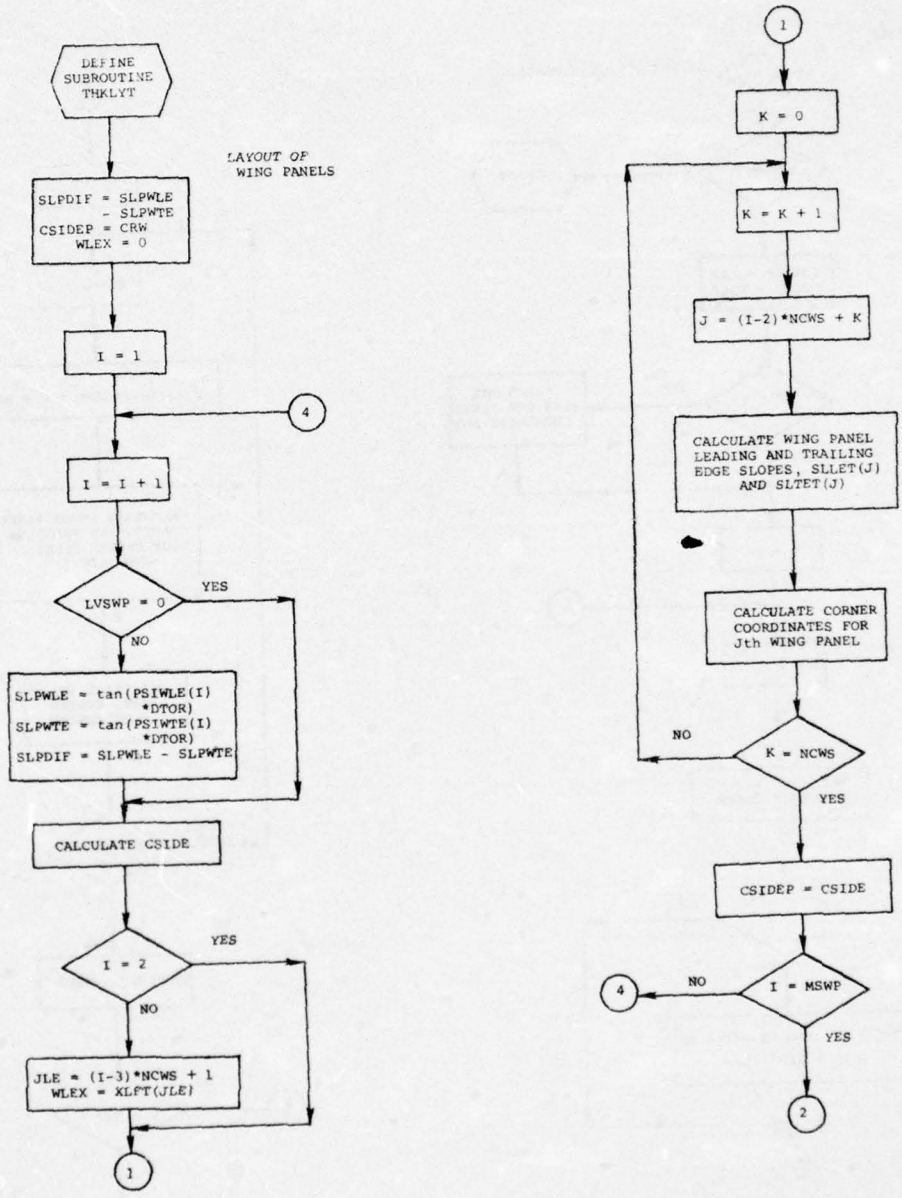
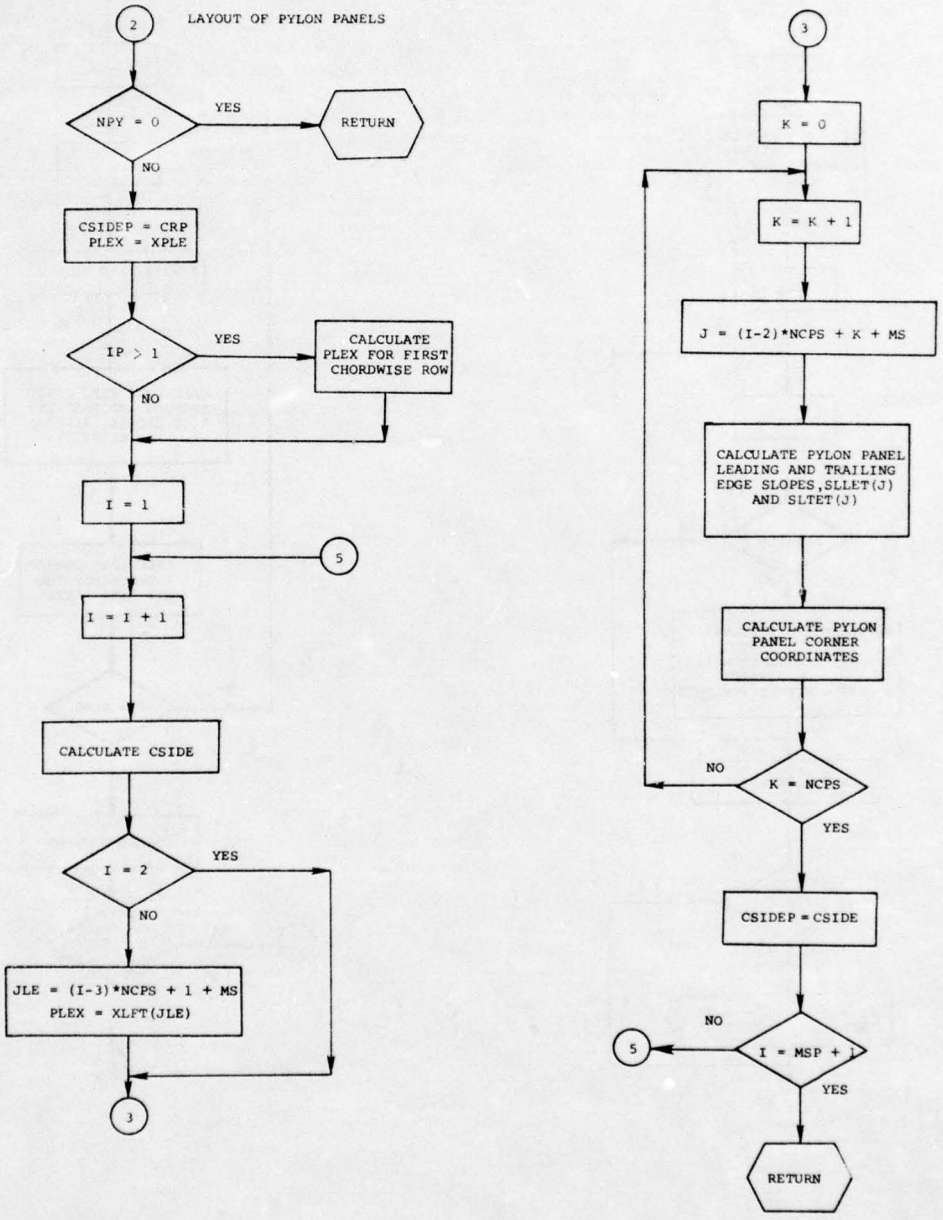


Figure I-18.- Flow chart of subroutine STORIO.



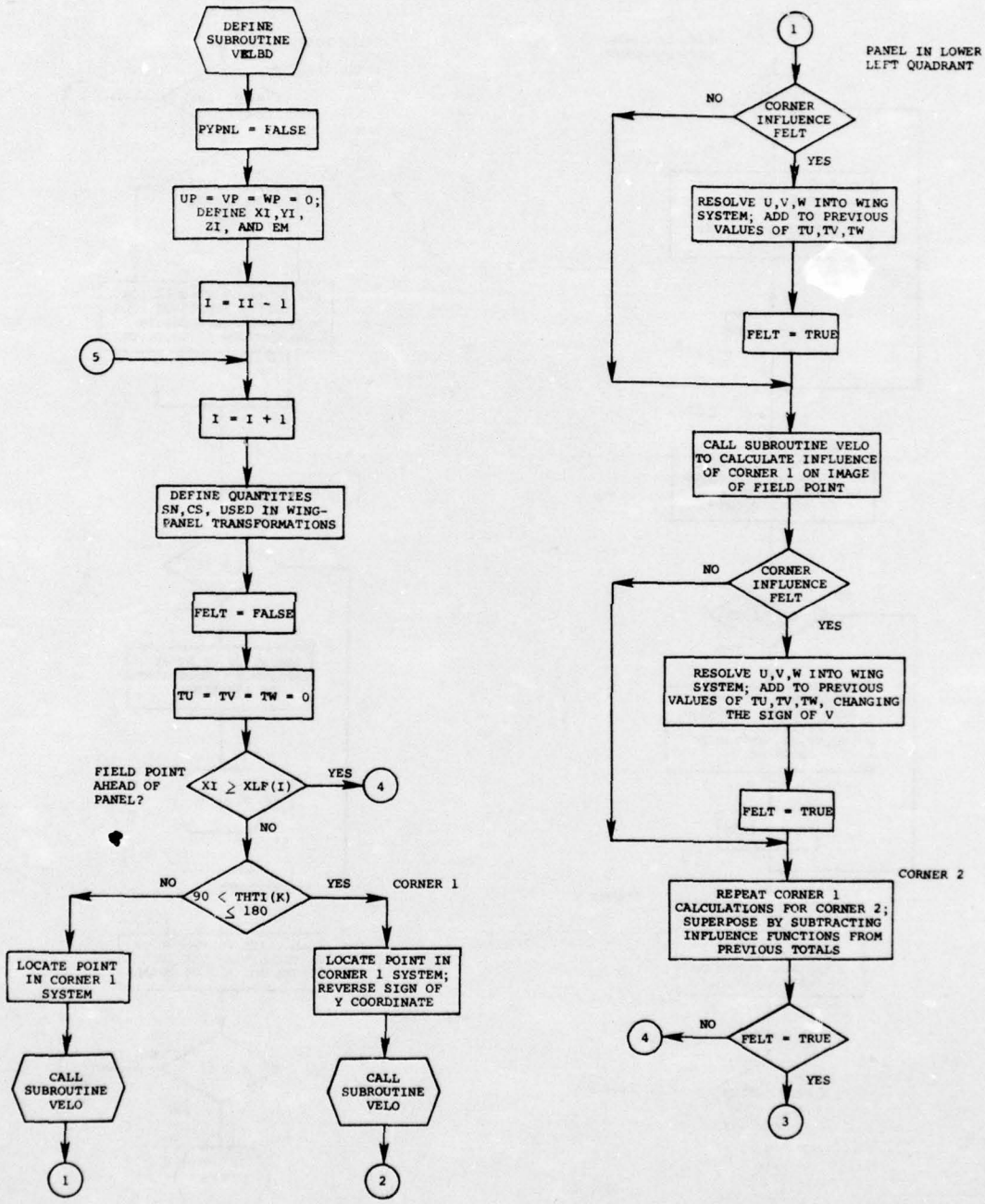
(a) Page 1.

Figure I-19.- Flow chart of subroutine THKLYT.



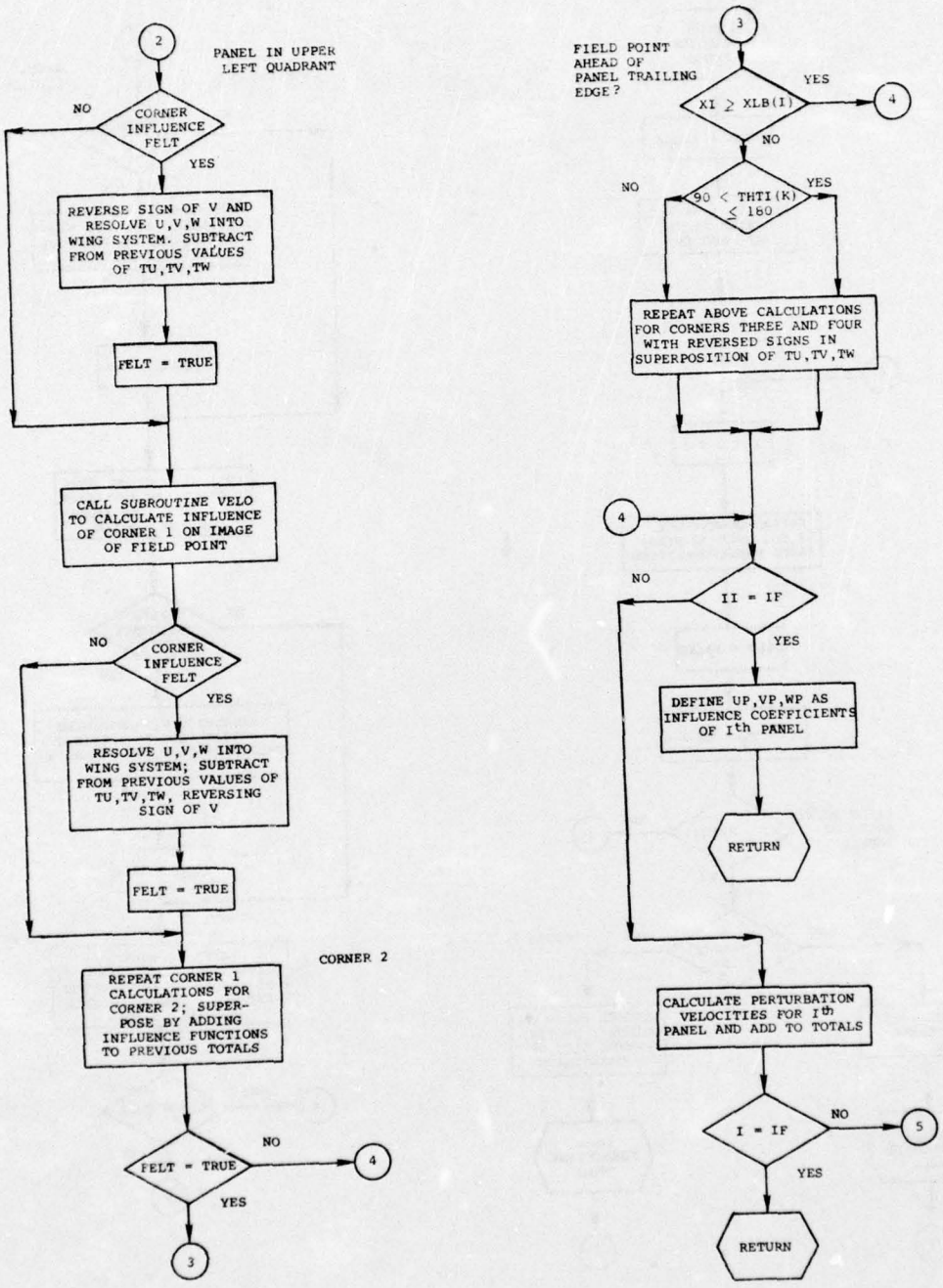
(b) Page 2.

Figure I-19.- Concluded.



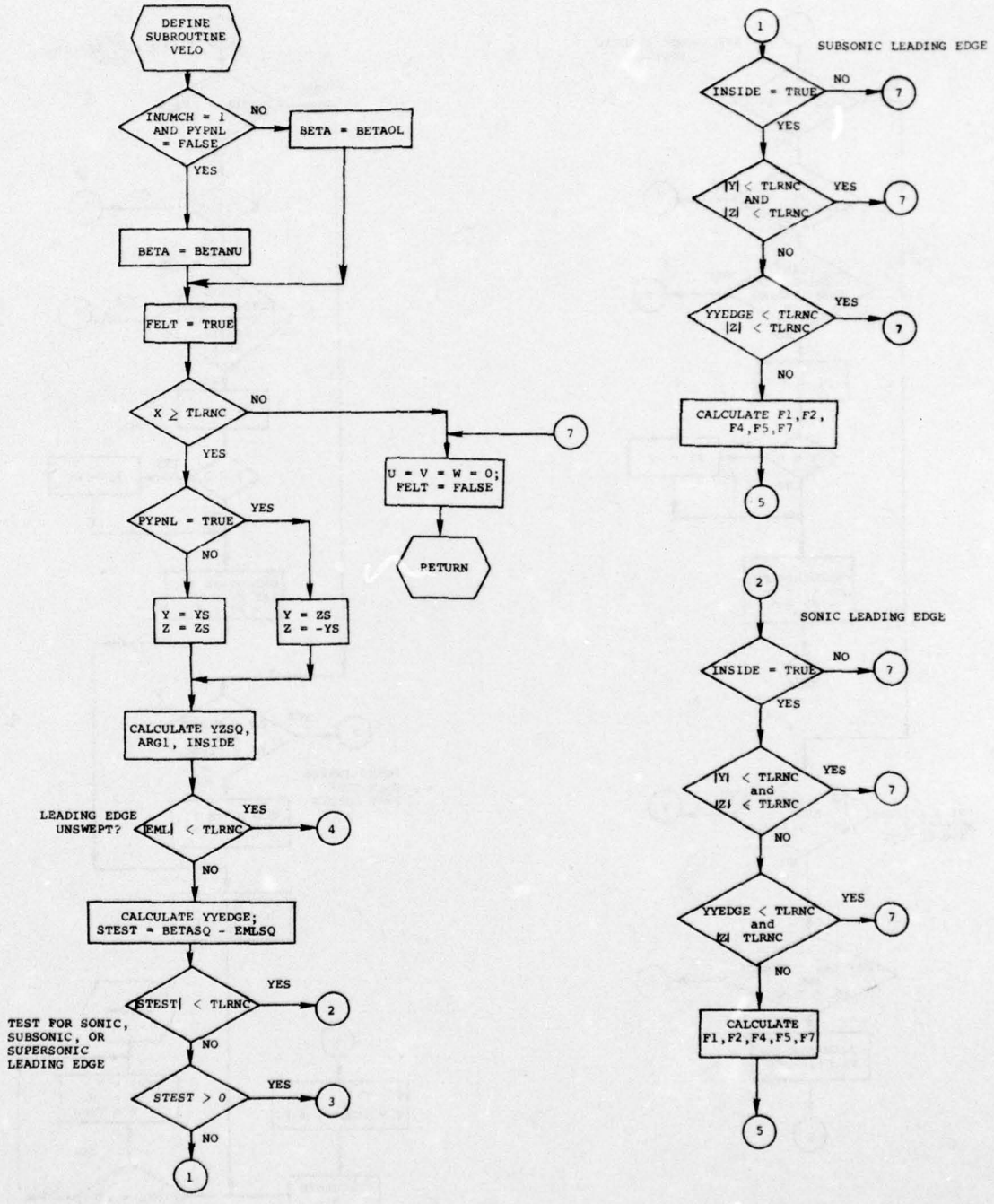
(a) Page 1.

Figure I-20.- Flow chart of subroutine VELBD.



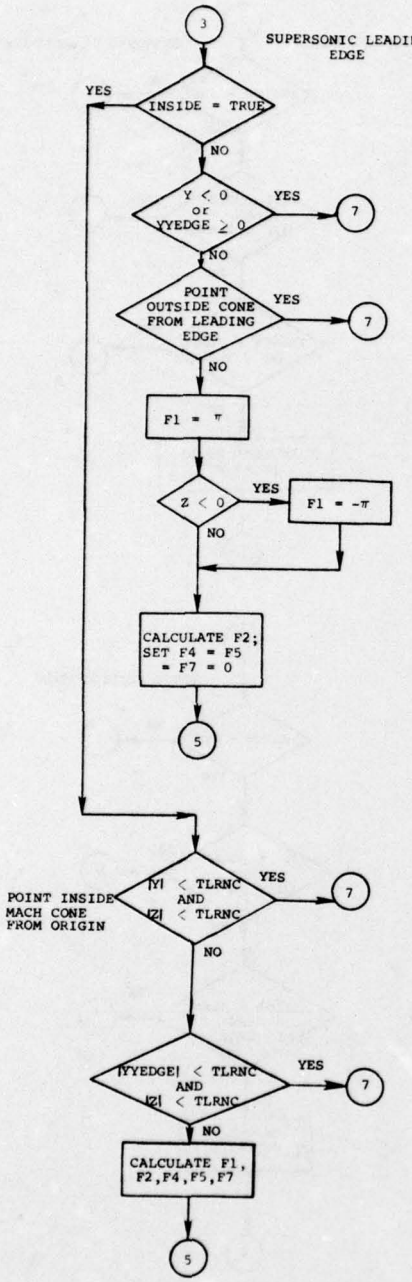
(b) Page 2.

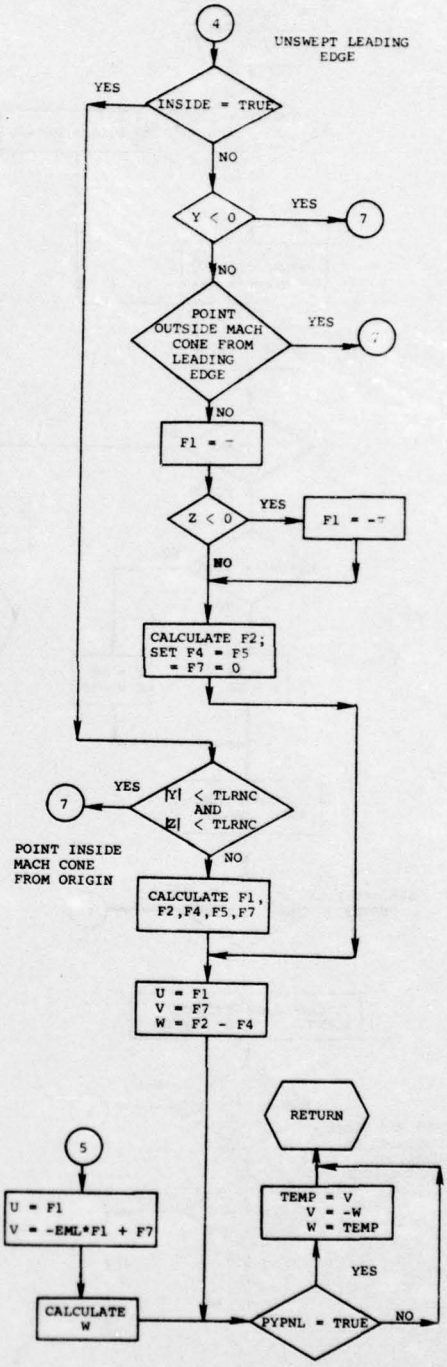
Figure I-20.- Concluded.



(a) Page 1.

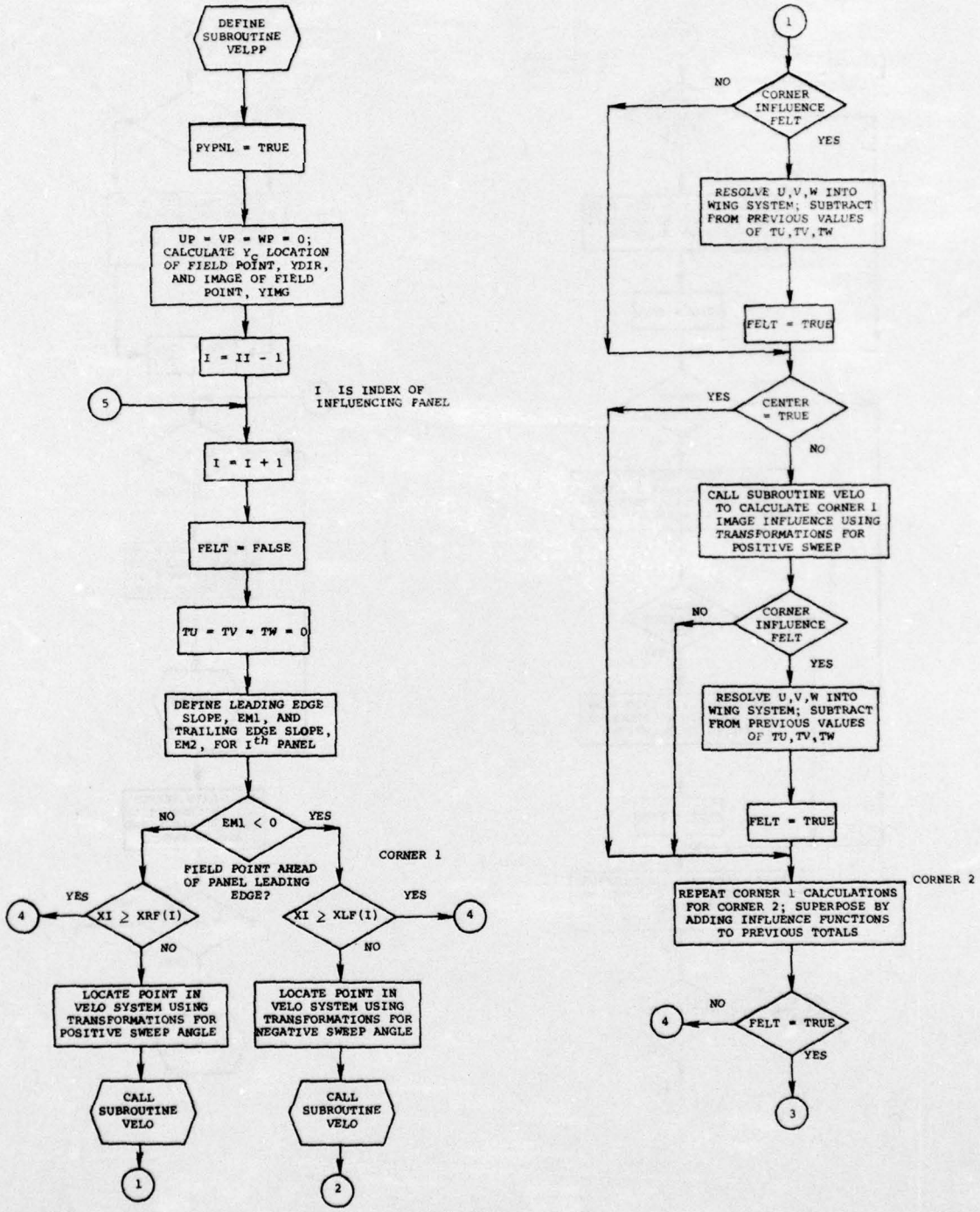
Figure I-21.- Flow chart of subroutine VELO.





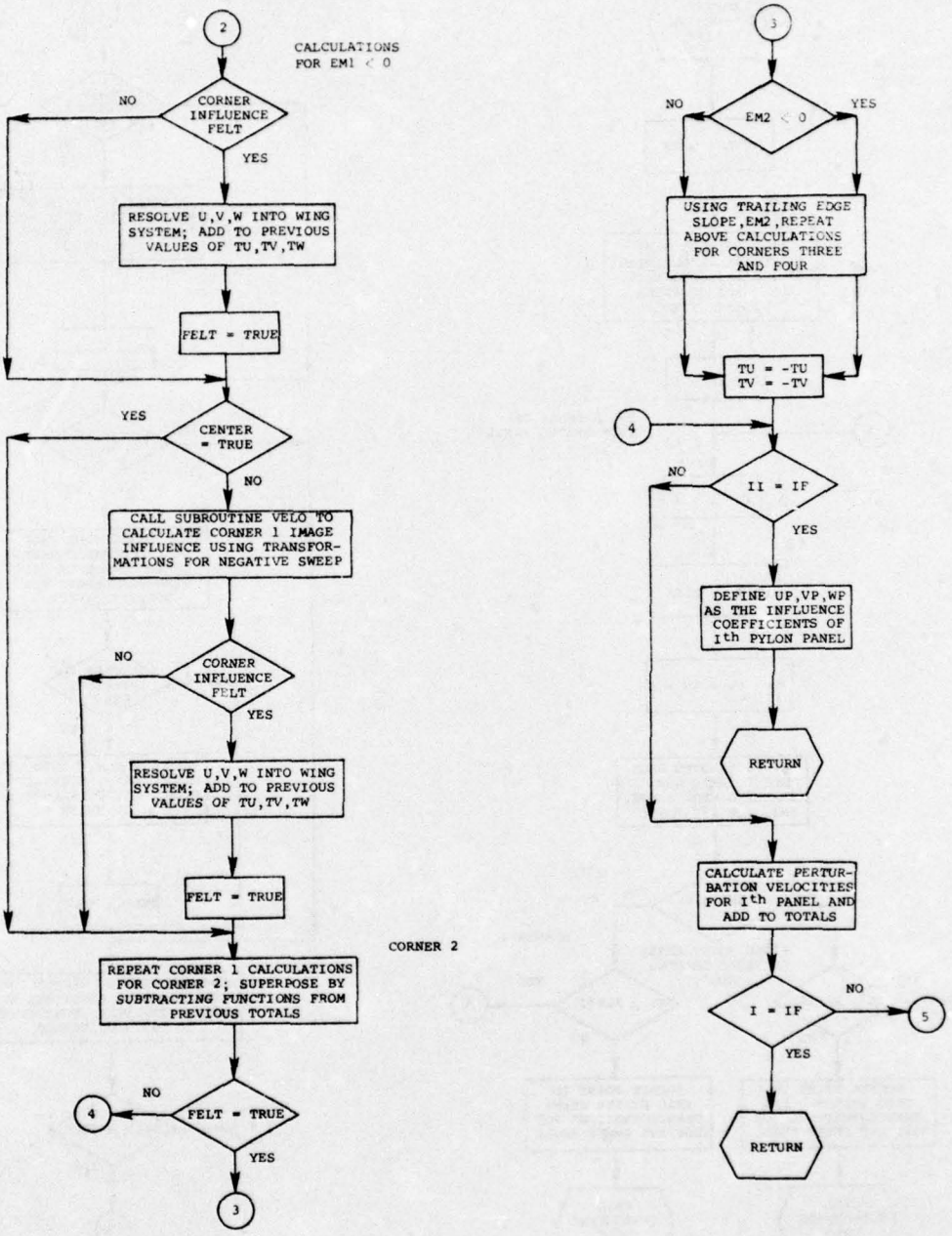
(b) Page 2.

Figure I-21.- Concluded.



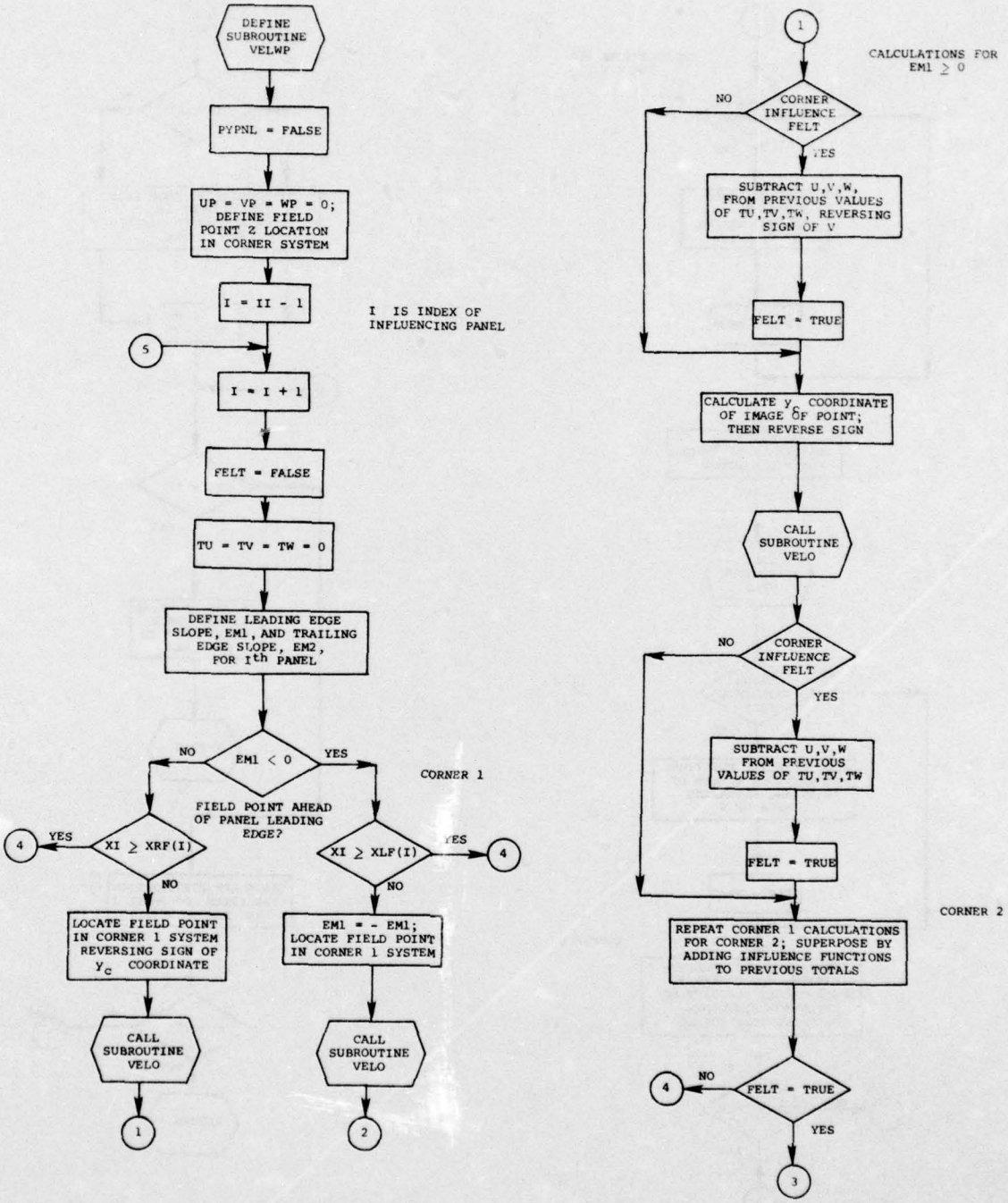
(a) Page 1.

Figure I-22.- Flow chart of subroutine VELPP.



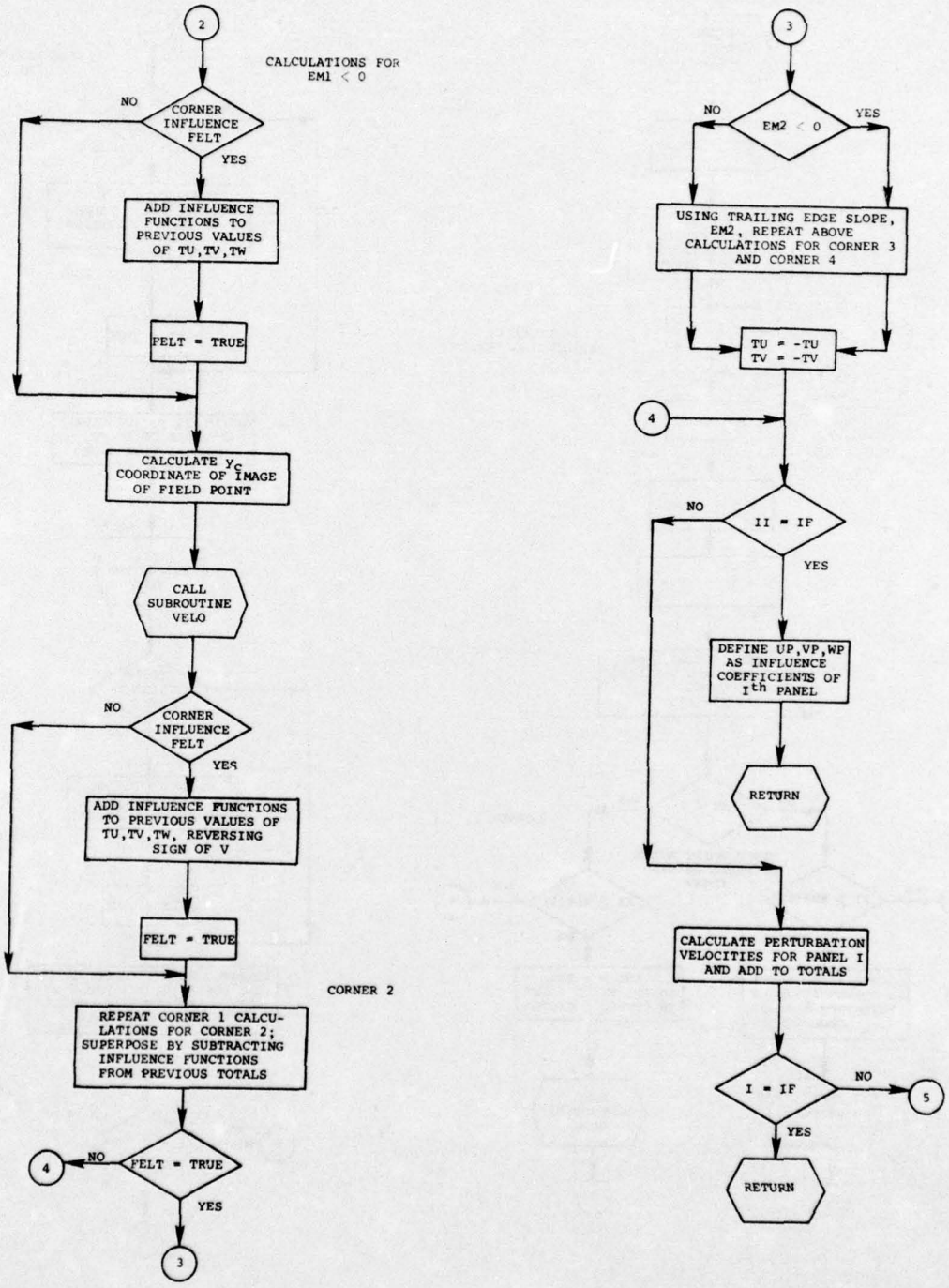
(b) Page 2.

Figure I-22.- Concluded.



(a) Page 1.

Figure I-23.- Flow chart of subroutine VELWP.



(b) Page 2.

Figure I-23.- Concluded.

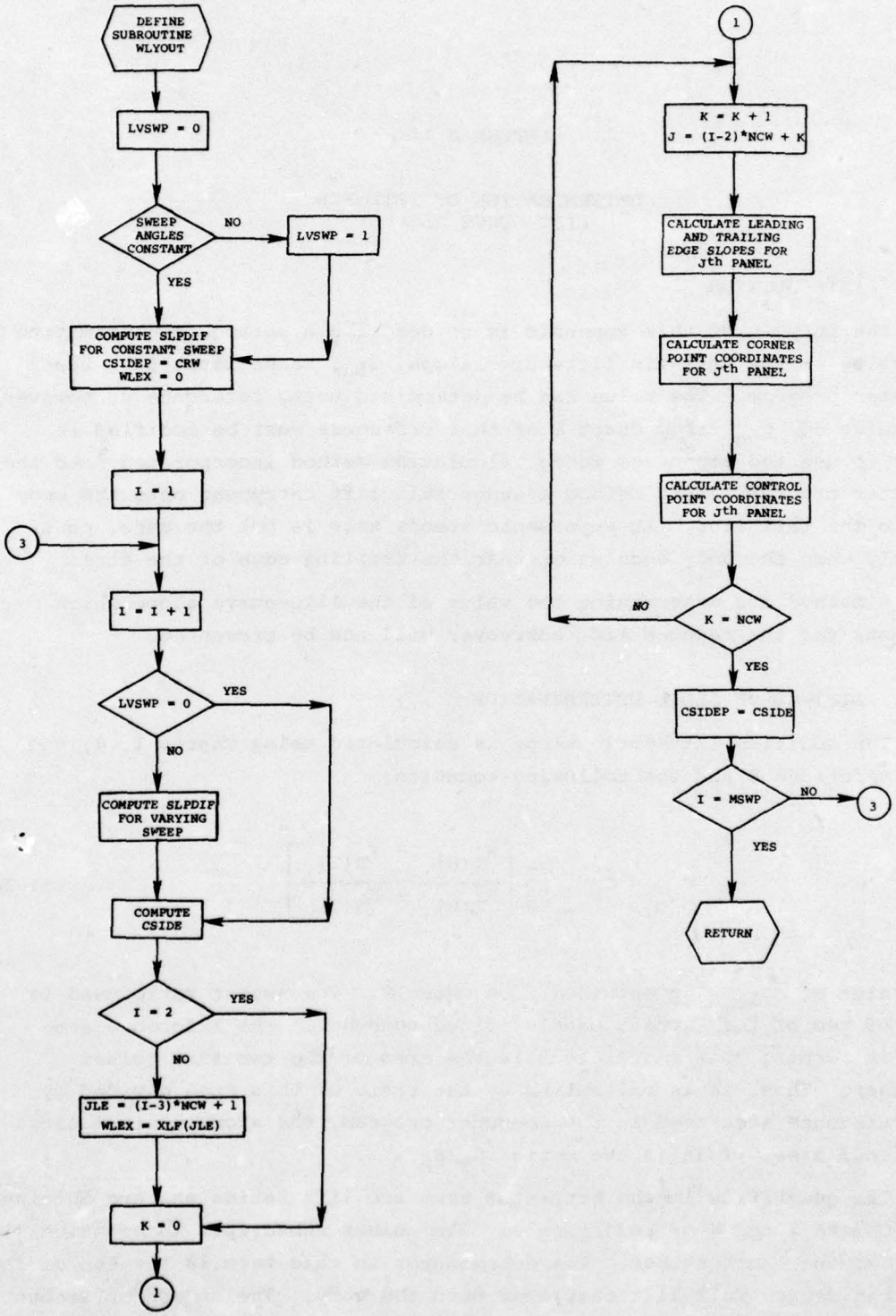


Figure I-24.- Flow chart of subroutine WLYOUT.

APPENDIX II

DETERMINATION OF TAIL-FIN LIFT-CURVE SLOPE

II-1. INTRODUCTION

The purpose of this appendix is to describe a method for estimating the value of the tail-fin lift-curve slope, C_{L_α}, to be input into the computer program. The value can be determined using reference 8; however, the value of C_{L_α} from Chart 8 of that reference must be modified in order to use the empennage force calculation method incorporated into the computer program. This method assumes full lift carryover onto the body due to the tail fins. At supersonic speeds this is not the case, particularly when the body ends at or near the trailing edge of the fins.

A method for determining the value of the lift-curve slope which accounts for the reduced lift carryover will now be presented.

II-2. LIFT-CURVE SLOPE DETERMINATION

The modified lift-curve slope is calculated using Charts 1, 4, and 8 of reference 8 and the following equation:

$$C_{L_\alpha} = C_{L_{\alpha_T}} \frac{S_T}{S_R} \left[\frac{K_{T(B)_1} + K_{B(T)_4}}{K_{T(B)_1} + K_{B(T)_1}} \right] \qquad \text{(II-1)}$$

The value of $C_{L_{\alpha_T}}$ is obtained from Chart 8. The aspect ratio used is that of two of the exposed panels joined together. The reference area used in forming this coefficient is the area of the two fins joined together. Thus, it is multiplied by the ratio of this area divided by the reference area used in the computer program, the store maximum cross-sectional area. This is the ratio S_T/S_R.

The quantities in the bracketed term are lift ratios and are obtained from Charts 1 and 4 of reference 8. The number subscripts in equation (II-1) refer to the chart number. The denominator in this term is the sum of the lift ratios for full lift carryover onto the body. The numerator accounts for only partial lift carryover onto the body. In determining the value of $K_{B(T)_4}$ from Chart 4, $(C_{L_\alpha})_W$ is $C_{L_{\alpha_T}} S_T/S_R$, the quantity multiplying the bracketed term in equation (II-1).

It is recommended that Chart 4 be used for low-aspect-ratio as well as high-aspect-ratio tail fins. Reference 8 recommends that Chart 4 only be used when the aspect ratio parameter defined there is greater than 4.0. The use of this chart in determining $K_{B(T)_4}$ for values of the aspect ratio parameter of 3.0 and 2.25 has resulted in good agreement with experimental data for the two different stores in uniform flow.

REFERENCES

1. Dillenius, M. F. E., Goodwin, F. K., and Nielsen, J. N.: Prediction of Supersonic Store Separation Characteristics. Vol. I - Theoretical Methods and Comparisons with Experiment. Technical Report AFFDL-TR-76-41, Vol. I, May 1976.

2. Dillenius, M. F. E., Goodwin, F. K., and Nielsen, J. N.: Extension of the Method for Predicting Six-Degree-of-Freedom Store Separation Trajectories at Speeds up to the Critical Speed to Include a Fuselage with Noncircular Cross Section. Vol. I - Theoretical Methods and Comparison with Experiment. Technical Report AFFDL-TR-74-130, Vol. I, Nov. 1974.

3. Goodwin, F. K. and Dillenius, M. F. E.: Extension of the Method for Predicting Six-Degree-of-Freedom Store Separation Trajectories at Speeds up to the Critical Speed to Include a Fuselage with Noncircular Cross Section. Vol. II - Users Manual for the Computer Program. Technical Report AFFDL-TR-74-130, Vol. II, Nov. 1974.

4. Goodwin, F. K., Dillenius, M. F. E., and Nielsen, J. N.: Prediction of Six-Degree-of-Freedom Store Separation Trajectories at Speeds up to the Critical Speed. Vol. I - Theoretical Methods and Comparison with Experiment. Technical Report AFFDL-TR-72-83, Vol. I, Oct. 1974.

5. Goodwin, F. K. and Dillenius, M. F. E.: Prediction of Six-Degree-of-Freedom Store Separation Trajectories at Speeds up to the Critical Speed. Vol. II - Users Manual for the Computer Program. Technical Report AFFDL-TR-72-83, Vol. II, Oct. 1974.

6. Goodwin, F. K., Nielsen, J. N., and Dillenius, M. F. E.: A Method For Predicting Three-Degree-of-Freedom Store Separation Trajectories at Speeds up to the Critical Speed. Technical Report AFFDL-TR-71-81, Nov. 1974.

7. 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.

8. Pitts, W. C., Nielsen, J. N., and Kaattari, G. E.: Lift and Center of Pressure of Wing-Body-Tail Combinations at Subsonic, Transonic, and Supersonic Speeds. NACA Report 1307, 1957.

9. Perkins, S. C., Jr., Goodwin, F. K.: Data Report for an External Store Test Program Conducted at Supersonic Speeds. Vols. I through VII. Nielsen Engineering & Research, Inc. NEAR TR 103, Dec. 1975.

10. Hildebrand, F. B.: Introduction to Numerical Analysis. McGraw-Hill Book Co., Inc., New York, NY, 1956, Chapter 6.

11. IBM System/360 Scientific Subroutine Package, Version III, Programmers Manual. IBM Corp., GH20-0205-4, August 1970, pp. 372-381.

12. Forsythe, G. E: Generation and Use of Orthogonal Polynomials for Data-Fitting with a Digital Computer. J. SIAM, Vol. V, No. 2, June 1957, pp. 74-88.

13. Conte, S. D.: Elementary Numerical Analysis. McGraw-Hill Book Co., Inc., New York, NY, 1965, pp. 174-176.

14. Scarborough, J. B.: Numerical Mathematical Analysis. Johns Hopkins Press, Baltimore, 1966, p. 137.

☆ U. S. GOVERNMENT PRINTING OFFICE: 1976 — 657-630/961