(12) LEVEL III

AD-E300 442

DNA 4531F-2

AD A 064 135

AERODYNAMIC ACCOUNTING TECHNIQUE FOR DETERMINING EFFECTS OF NUCLEAR DAMAGE TO AIRCRAFT

Volume II – Program User Guide

General Dynamics Corporation
Fort Worth Division
P.O. Box 748, Grants Lane
Fort Worth, Texas 76101

DDC FILE COPY

28 February 1978

Final Report for Period 15 March 1977–30 January 1978

CONTRACT No. DNA 001-77-C-0075

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

THIS WORK SPONSORED BY THE DEFENSE NUCLEAR AGENCY UNDER RDT&E RMSS CODE B342077464 N99QAXAE50602 H2590D.

Prepared for
Director
DEFENSE NUCLEAR AGENCY
Washington, D. C. 20305

79 01

Destroy this report when it is no longer needed. Do not return to sender.

PLEASE NOTIFY THE DEFENSE NUCLEAR AGENCY, ATTN: TISI, WASHINGTON, D.C. 20305, IF YOUR ADDRESS IS INCORRECT, IF YOU WISH TO BE DELETED FROM THE DISTRIBUTION LIST, OR IF THE ADDRESSEE IS NO LONGER EMPLOYED BY YOUR ORGANIZATION.

DISCLAIMER NOTICE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: DNA 4531F-2

2. GOVT ACCESSION NO.:

3. RECIPIENT'S CATALOG NUMBER:

4. TITLE (and Subtitle): AERODYNAMIC ACCOUNTING TECHNIQUE FOR DETERMINING EFFECTS OF NUCLEAR DAMAGE TO AIRCRAFT. Volume II. Program User Guide.

5. TYPE OF REPORT & PERIOD COVERED: Final Report for Period 15 Mar 77—30 Jan 78

6. PERFORMING ORG. REPORT NUMBER:

7. AUTHOR(s): Roy Schemensky, George Howell

8. CONTRACT OR GRANT NUMBER(s): DNA 001-77-C-0075

9. PERFORMING ORGANIZATION NAME AND ADDRESS:
General Dynamics Corporation
Fort Worth Division, P.O. Box 748, Grants Lane
Fort Worth, Texas 76101

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: Subtask N99QAXAE506-02

11. CONTROLLING OFFICE NAME AND ADDRESS:
Director
Defense Nuclear Agency
Washington, D.C. 20305

12. REPORT DATE: 28 February 1978

13. NUMBER OF PAGES: 252

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office):

15. SECURITY CLASS (of this report): UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT (of this Report):
Approved for public release; distribution unlimited.
DNA, SBIE 4531F-2, AD-E300 442

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

18. SUPPLEMENTARY NOTES:
This work sponsored by the Defense Nuclear Agency under RDT&E RMSS Code B342077464 N99QAXAE50602 H2590D.

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

Drag Prediction	Roughness	Computer Program
Lift Prediction	Blast	Nuclear Damage
Aircraft	Overpressure	
Damage	Thermal Effects	

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

This report (Volume II, Program User Guide) describes the computer program developed for the aerodynamic evaluation of aircraft, including the effects of roughness and damage. The computer code has the capability of assessing the aerodynamic effects of damage, such as rough, bent, and burnt skins, boundary-layer thickness effects, and loss of radomes, panels, doors and covers. Also, the computer code has the capability of analyzing changes in drag due to lift and trim caused by asymmetric loss of parts of the wing or trim surfaces. → over

402709 LB

20. ABSTRACT (Continued)

The computer program is coded in the FORTRAN Extended Version IV language to operate on a CDC 6600 or CYBER 172 computer. The computer code structure is explained and a computer code input utilization is presented along with sample input and output with a FORTRAN source deck listing. Details of the methods, equations, and substantiating data for this computer code are contained in Volume I, Empirical Methods.

TABLE OF CONTENTS
Volume II

		Page
Section 1.	GENERAL DESCRIPTION	3
Section 2.	AAT COMPUTER CODE UTILIZATION	5
Section 3.	SAMPLE PROBLEMS	39
Section 4.	PROGRAM DESCRIPTION	91
	4.1 Program Structure	91
	4.2 Subroutine Descriptions	91
	4.3 Common Block Descriptions	111
	4.4 Program Listing	126

1. GENERAL DESCRIPTION

The Aerodynamic Accounting Technique Computer Code was developed to provide a computerized, systematic method to evaluate the aerodynamic effects of nuclear damage to an aircraft. In addition, it can be used to evaluate the aerodynamic characteristics of an undamaged configuration or any type damage that can be modeled into one of the fourteen damage modes provided by the program.

The computer program is coded in the Fortran Extended Version IV language to operate on the CDC 6600 computer at Wright-Patterson Air Force Base. It is also operational on General Dynamics' CDC CYBER 172 computer as procedure R7F. Four primary overlays are used by the program to keep the central memory core requirements below 54,000 bytes. The main program controls the calling of the four overlay programs: XINPT, GEOM, SURVEY, and NUCDAM. Figure 1.1 shows the program overlay structure and the arrangement of the 57 subprograms and subroutines that comprise the AAT computer code. Program XINPT controls the reading of input data and the printout of data that will be used in the problem. Program GEOM computes the geometric parameters that are required in the calculations. Program SURVEY controls the calculation of aerodynamic characteristics of the undamaged aircraft, and program NUCDAM determines the aerodynamic effects that result from the damage. The third primary overlay, SURVEY, calls four secondary overlay programs VGEOM, MCRIT, AEROA, and AEROB. These programs and subprograms call the appropriate subroutines necessary to make the computations.

The program utilization describes the inputs that are required. Aerodynamic characteristics of an undamaged configuration can be estimated by entering only the basic geometric parameters and the survey conditions. Aircraft damage evaluation is optional. The damage input data are contained in a separate data block and the effects of damage are determined on an incremental basis and integrated into the aerodynamics of the undamaged configuration.

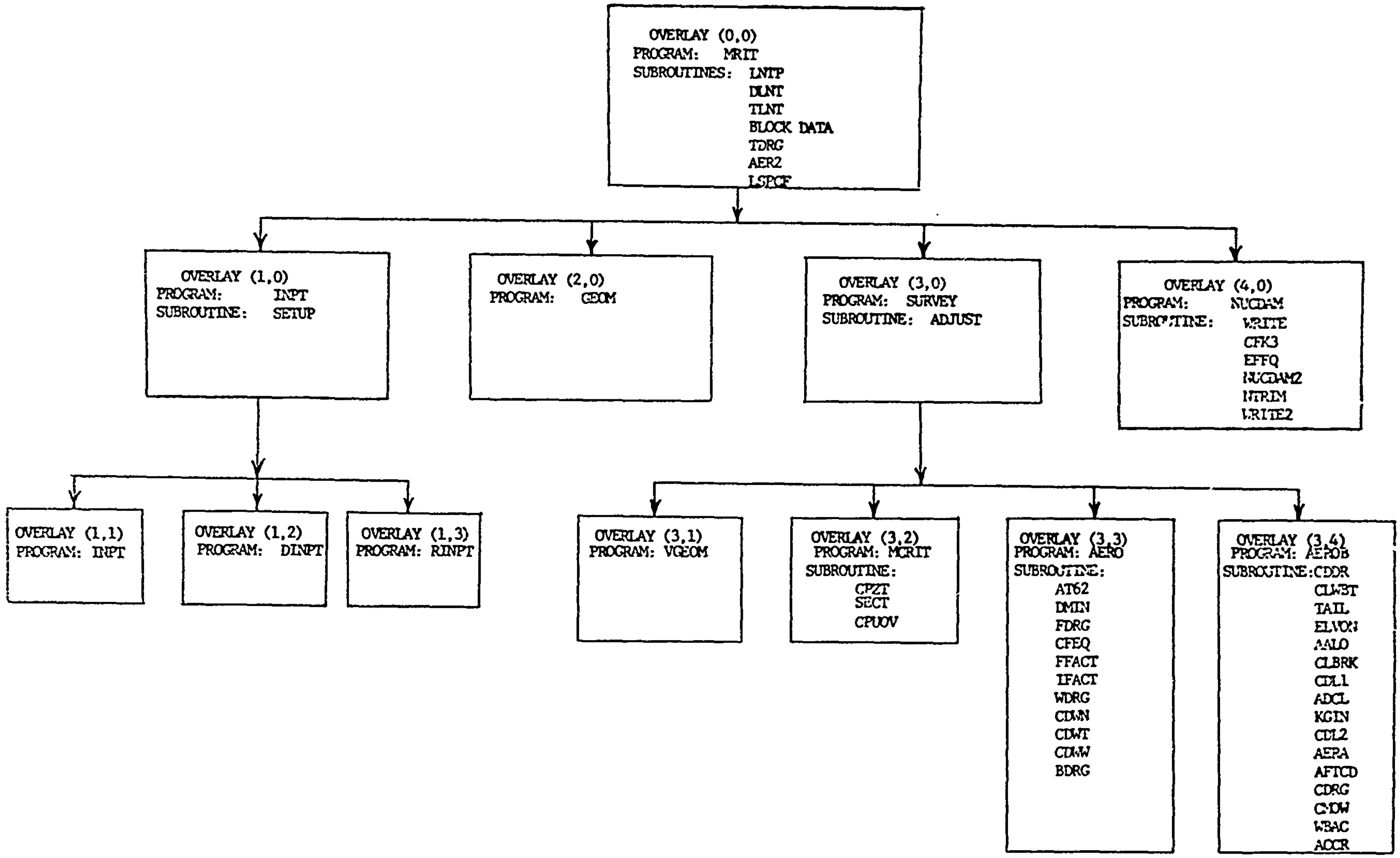


FIGURE 1-1 PROGRAM OVERLAY STRUCTURE

2. AAT COMPUTER CODE UTILIZATION

The first step in preparing program input is to determine how to best represent the configuration with circular bodies and surfaces. Components such as the fuselage, canopy, stores, external fairings, and nacelles are represented as bodies, while the wing, horizontal tail, vertical tail, pylons, and ventral fins are represented as surfaces. A straight-wing planform is represented by one panel, and a cranked- or complex-wing planform is approximated with two panels. Up to seven bodies and seven surfaces may be used to represent a configuration. Figures 1 and 2 define some of the geometric parameters required to define the individual bodies and surfaces.

Input requirements for the Aerodynamic Accounting Technique computer code are partially determined by the options selected by the user. The input format is divided into 12 blocks to assist the user in determining his input requirements. The user may decide which blocks are necessary to complete his problem while not concerning himself with the remaining blocks. The user determines which data blocks are required from the following descriptions:

1. General Information. This block contains information that the AAT program needs to read the remainder of the input data. Indicators, such as the number of bodies and surfaces, are included, which tell the program which blocks of data will follow. Certain reference data and geometric dimensions are also included. This block is always required.

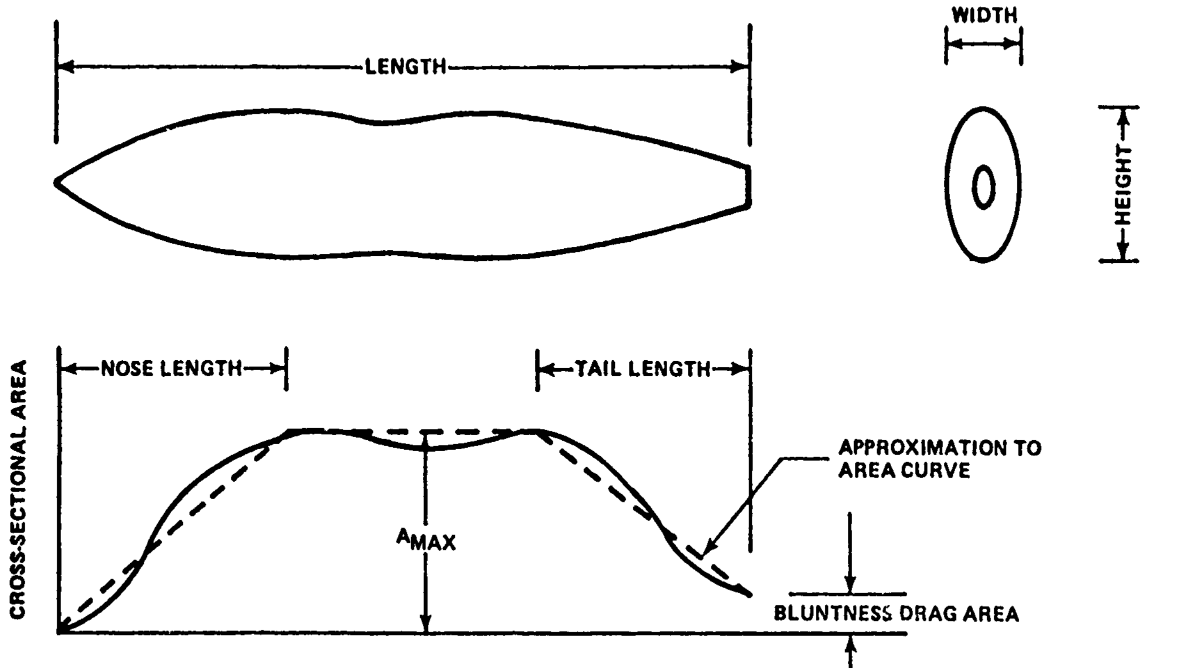
2. Body Geometry. The geometric parameters of each component which is represented as a body is contained in this block. A maximum of seven bodies may be used.

3. Surface Geometry. The geometry of each surface is represented in this block. A maximum of seven panels may be used, with either one or two panels for the main wing.

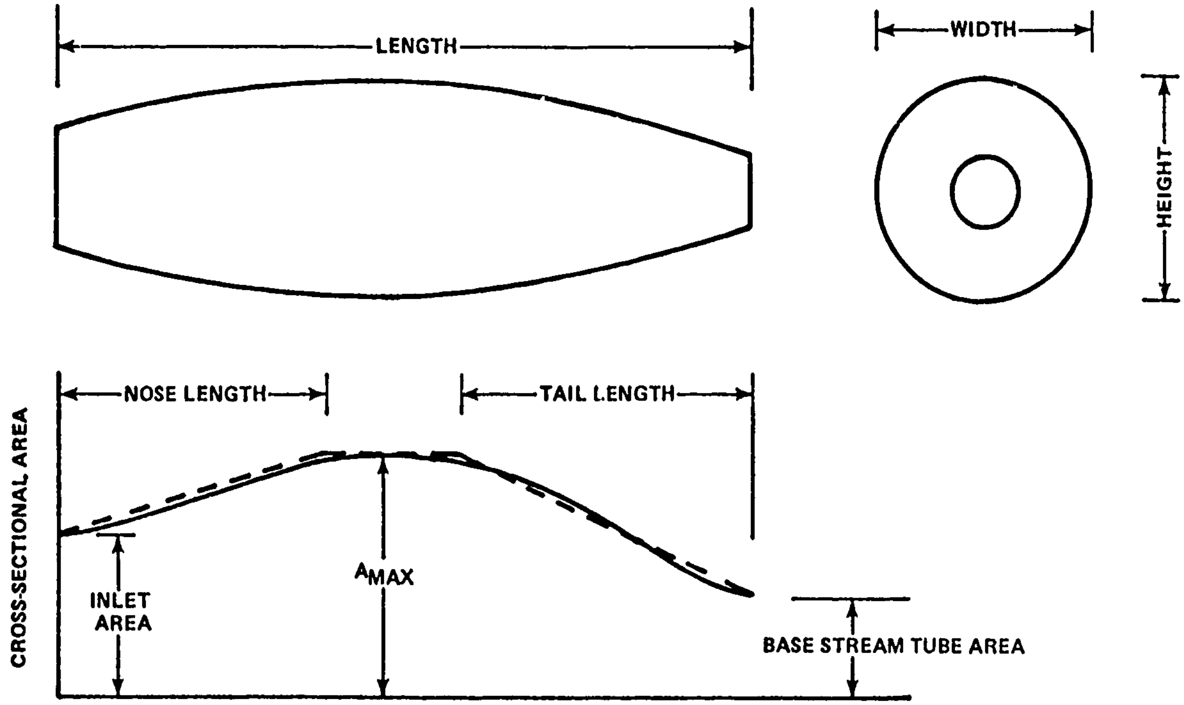
4. Arbitrary Airfoil. This block is used to describe the camber and thickness distribution of an arbitrary airfoil section if one of the standard sections contained internally is not sufficient.

5. Variable Sweep. Aircraft with movable wing panels such as the F-111 and B-1, may be evaluated at any selected sweep angle. This block of data is required to describe the wing movement.

6. Survey Conditions. This block of data contains the Mach number and Reynolds number conditions at which the analysis will be conducted.

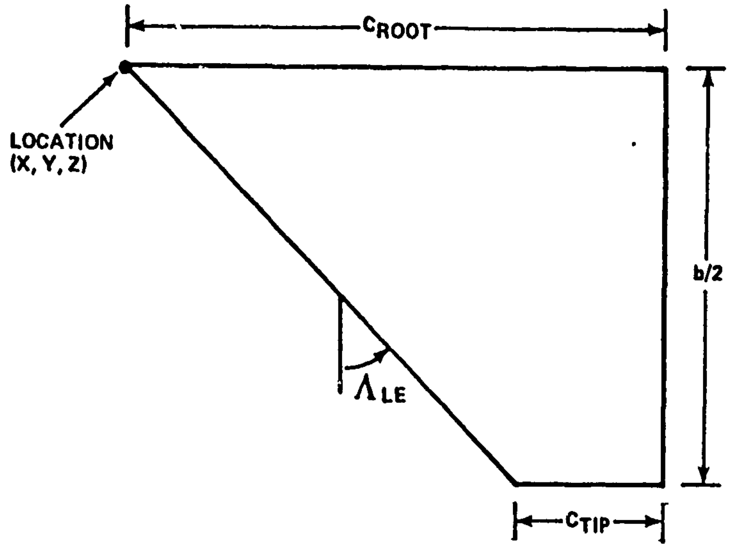


(a) Closed Body

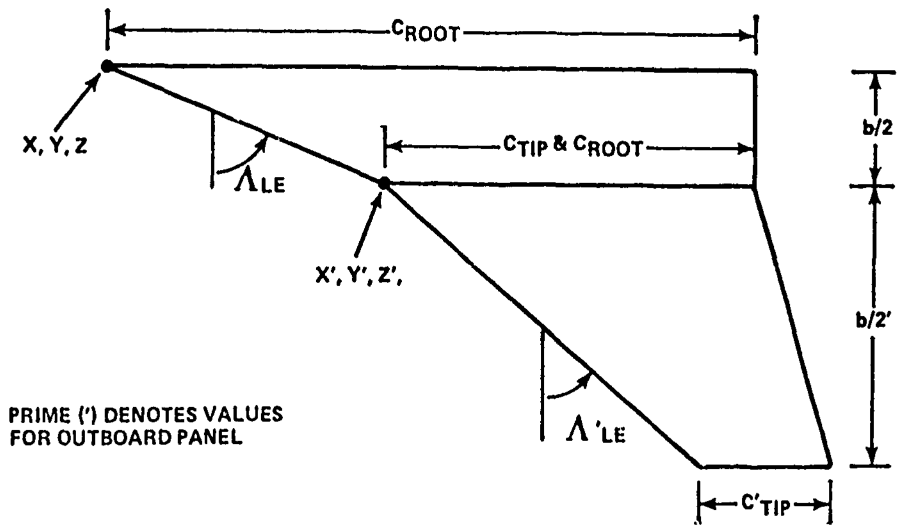


(b) Nacelle

Figure 2-1 Geometry for Closed Bodies and Nacelles



(a) Single-Panel Wing or Other Surface (Exposed Area)



(b) Two-Panel Wing (Exposed Area)

Figure 2-2 Surface Geometry

7. Adjustment Factors. A table of adders and multipliers can be specified in this block to correlate the AAT predictions with known test levels. At least one card is always required.

8. Expansion Provisions. This data block is provided for future expansion.

9. Damage Mode Indicators. Provisions are included for fourteen modes of nuclear damage in the AAT program. Seven modes are for body damage and seven for surface damage. This block, which requires three cards, allows the user to select only those modes that are desired.

10. Body Damage Parameters. This block of data contains parameters that define nuclear damage to the bodies. It is required only if a body damage mode is indicated in Block number 9.

11. Surface Damage Parameters. This block describes nuclear damage to the surfaces and is similar in format to Block 10. It is required only if a surface damage mode is indicated in Block 9.

12. End of Problem. Describes how one may end the problem or change the input and run new problems.

These twelve blocks of data represent a wide range of options for defining the geometry and the manner in which the data are handled. Most aircraft analyses will not require all blocks.

It should be noted that there is an option that allows the program to compute the values of certain parameters that are shown as input. These items are indicated with an asterisk (*). If a zero or blank is input, the program will use other geometric parameters to compute these values. All dimensions are input in either inches, feet or meters, except for the roughness factor, which is always in inches. All angles are in degrees. The utilization specifies one of the following formats for each item of input:

"I" Format — Input must be in integer form and right adjusted in the specified field.

"F" Format — Input must include a decimal and may be anywhere within the specified field.

"A" Format — Input may include any alphanumeric characters.

Data Block No. 1, General Information

Data Block No. 1 requires from 5 to 7 cards, depending on the options selected.

CARD 1.1 - Title. Enter any alphanumeric characters in columns 1 through 60 to identify each problem. These characters will be printed out at the top of each page of output for the undamaged configuration.

CARD 1.2 - Print-out options. This card should normally be left blank; however, certain seldom-used data and subroutine dumps will be printed if KPRINT(I)=1. The subroutine dumps are normally for diagnostic purposes and can be used only in conjunction with the programming statements. If KPRINT(31)=1, the aerodynamic data for both the damaged and undamaged configurations will be printed at constant values of angle of attack.

COLUMN	SYMBOL	PRINT-OUT
1	KPRINT(1)	Airfoil ordinates and pressure distribution
11	(11)	Dump Subroutine AER2
12	(12)	GEOM
13	(13)	AALO
14	(14)	CDL2
15	(15)	BDRG
16	(16)	CLBRK
17	(17)	AERA
18	(18)	WBAC
19	(19)	CFEQ
20	(20)	CDWW
21	(21)	TAIL
22	(22)	CDL1
23	(23)	CDDR
24	(24)	ADJUST
25	(25)	CMOW
26	(26)	CPZT
27	(27)	SSET
28	(28)	NUCDAM
29	(29)	NUCDAM2
30	(30)	NTRIM
31	(31)	WRITE2

CARD 1.3 - Configuration Definition (I5 Format).

Column	Symbol	Definition
5	NPODS	Number of different types of bodies. NPODS ≤ 7.
10	NPNLS	Number of surfaces used to represent the main wing. Must be 1 or 2.
15	NHT	Number of lifting surfaces other than main wing, i.e., horizontal tail or canard. Program assumes that these surfaces are symmetric with respect to configuration centerline.
20	NVT	Number of non-lifting surface types, i.e., vertical tail, ventral fin, or pylon. For example, a twin vertical tail should be entered as 1 surface type; symmetry will be indicated in Data Block No. 3. NPNLS+NHT+NVT ≤ 7.
25	ISWP	Variable-sweep indicator. 0 for fixed wing 1 for variable-sweep wing
30	IREF	Reference angle-of-attack indicator. 0 reference to wing root-chord plane 1 reference to fuselage centerline
35	IWNG	WING-definition indicator. 0 omit card 1.6 1 enter reference planform area, taper ratio, and leading-edge sweep on Card 1.6 See note number 1 at end of input instructions for Data Block No. 1.
39-40	NAFO	Number of stations at which thickness and camber are to be defined for an arbitrary airfoil, ≤ 30. If NAFO > 0, Data Block No. 4 must be input.
45	METER	Unit-system indicator 0 use units of feet 1 use units of meters 2 use units of inches, except for reference area, planform area, wetted areas, Reynolds number, and altitude which are in feet.
50	ITRIM	Trim indicator. 0 trim configuration 1 do not trim

CARD 1.4 - "F" Format.

Column	Symbol	Definition
1-10	SREF	Reference area.
11-20	CMAC*	Reference aerodynamic chord.
21-30	XMAC*	Location for leading-edge of CMAC. Does not change with sweep.
31-40	ZCG	Height of moment reference point relative to wing root-chord plane.
41-50	UPFUS	Upsweep angle of the aft fuselage (deg). For use with transport-type aircraft.
51-60	AOB	Ratio of width to height of the aft fuselage in the upswept region.

CARD 1.5 - (F10 Format).

Column	Symbol	Definition
1-10	ROUGHK	Average surface roughness height for friction drag, (inches).
11-20	FMISC	Miscellaneous drag factor as a percentage of total friction and form drag.
21-30	TWIST	Wing twist between exposed root chord and tip chord. Negative for washout (wing-tip leading edge down).
31-40	CONCL	Wing conical camber design lift coefficient.

CARD 1.6 (F10 format). This card is required only if IWNG=1 on Card 1.3.

Column	Symbol	Definition
1-10	SPLAN	Wing theoretical planform area.
11-20	TAPR	Reference wing theoretical taper ratio.
21-30	SWP	Reference wing leading-edge sweep.

*An asterisk beside a symbol indicates that the value of that item will be computed internally if a blank or zero is input.

CARD 1.7 (F10 Format) Elevon Definition. Required if NHT=0 and ITRIM=0. This card defines the elevon, which will be used to trim the configuration if there is no horizontal tail or canard.

Column	Symbol	Definition
1-10	CFOC	Elevon chord length, $\Delta c/c$
11-20	EI	Elevon inboard edge, expressed as a fraction of semi-span.
21-30	EO	Elevon outboard edge, expressed as a fraction of semi-span.

Notes pertaining to the input for Data Block No. 1 are as follows:

1. Card 1.6 is used to input the wing planform area, taper ratio, and leading-edge sweep when the user desires to override the internal calculations that normally compute these values based on other wing parameters. Equations used to make these calculations are presented in Section 2.1.3 of Volume I. Card 1.6 should not be required for a single-panel wing or for a two-panel wing with a moderate amount of crank. It is recommended, however, when two panels are used to represent a wing with an extreme amount of crank or a strake/wing planform where the strake is highly swept.

2. The input refeience area is arbitrary and does not have to be related to the theoretical wing planform area.

Data Block No. 2, Body Geometry

Thirteen cards are used to input the name and 12 parameters for each of the bodies. The name that is input on Card 2.1 will be used in the printout to identify the various components. Data for each body are listed vertically in the input. The main body (fuselage) should be listed to the extreme left, (Column 1-10), followed next by other bodies with zero inlet area. Nacelle data should be listed last. Up to seven columns for seven different body types can be input.

Card	Format	Symbol	Definition
2.1	7A10	BNAME(I)	Body I name, I=1, NPODS
2.2	7F10	BOD(I,1)	Length
2.3	↓	(I,2)	Width
2.4	↓	(I,3)	Height
2.5	↓	(I,4)*	Wetted area (total for all bodies of type I).
2.6	↓	(I,5)	Interference factor
2.7	↓	(I,6)	Number of bodies of type I
2.8	↓	(I,7)*	Maximum cross-sectional area
2.9	↓	(I,8)	Base streamtube area
2.10	↓	(I,9)	Nose length
2.11	↓	(I,10)	Boattail length
2.12	↓	(I,11)	Base area for bluntness drag
2.13	↓	(I,12)	Inlet area.

There are several points that should be mentioned pertaining to Data Block No. 2.

1. The name assigned to each body will be used in the printout of results.

2. The following is presented as a guide for determining the interference factor (BOD(I,5)):

=1.0 for nacelles and external stores mounted out of the local velocity field of the wing and fuselage.

=1.25 for external stores mounted symmetrically on the wing tip.

=1.3 for nacelles and external stores if mounted in moderate proximity of the wing.

=1.5 for nacelles and external stores mounted flush to the wing.

(2., continued)

The same variation of the interference factor applies in the case of a nacelle or external store strut-mounted to or flush-mounted on the fuselage.

3. The length, width, height, cross-sectional area, etc. listed for each body type describes a single body. The total number of bodies of each type must also be specified even though they may be paired symmetrically with respect to the centerline. One exception is that the wetted area represents the total for all bodies of a particular type.

4. Bodies must be defined for I=1 to NPODS. The main body must be first, closed bodies second, and bodies with inlet area last.

5. When fewer than seven bodies are defined, columns to the right of the last field may be conveniently used to identify the parameter listed on that card.

6. The interference factor for the fuselage that is entered on Card 2.6 is overridden by internal calculations for wing/body interference.

Data Block No. 3, Surface Geometry

Thirteen cards are used to input the name and 12 parameters for each of the surfaces. The format for the surface data is similar to that for the bodies.

Card	Format	Symbol	Definition
3.1	7A10	SNAME(I)	Surface name I, I=1, NPNLS+NHT+NVT
3.2	7(A3,7X)	SUR(I,1)	Airfoil section
3.3	7F10	(I,2)	2-D design lift coefficient
3.4	↓	(I,3)	Thickness ratio, $(t/c)_{RMS}$
3.5	↓	(I,4)	Leading-edge sweep
3.6	↓	(I,7)*	Total surface(s) wetted area
3.7	↓	(I,8)	Exposed root chord
3.8	↓	(I,9)	Tip chord
3.9	↓	(I,10)	Exposed semi-span
3.10	↓	(I,11)	X-station at LE of exposed root chord
3.11	↓	(I,12)	Y-station at LE of exposed root chord
3.12	↓	(I,13)	Z-station at LE of exposed root chord
3.13	↓	(I,14)	Incidence with respect to main body

Several points that should be noted regarding Data Block No. 3 are:

1. The name assigned to each surface will be used in the printout of results.

2. Surfaces must be input from left to right in the following order: NPNLS main wing panels, NHT lifting surfaces, and finally, NVT non-lifting surfaces.

3. Twenty options are available for defining the airfoil section type:

INPUT	AIRFOIL TYPE*	INPUT	AIRFOIL TYPE*
63-	63-YXX(6 Series)	-62	00XX-62(4 Digit)
64-	64-YXX "	-63	00XX-63 "
65-	65-YXX "	-64	00XX-64 "
66-	66-YXX "	-65	00XX-65 "
63A	63AYXX(6A Series)	-33	00XX-33 "
64A	64AYXX "	-34	00XX-34 "
65A	65AYXX "	-35	00XX-35 "
BIC	Biconvex	-93	00XX-93 "
INPUT	Arbitrary Airfoil	-94	00XX-94 "
		-95	00XX-95 "

*Y= Section Camber in Tenths
XX= Section t/c in % Z/C

(3.,continued)

Any of these sections may be input to the program with the identification shown above. The characters must, however, be left-adjusted in the data field.

An arbitrary section may be input for any surface by entering "INPUT". This will be used for both wing panels if NPNLS=2. Use of "INPUT" requires that Data Block No. 4 be included and that NAFO $>$ 0 on Card 1.4. Only one arbitrary section can be defined, but this section can be used for any of the surfaces by showing "INPUT" on Card 3.2.

4. Symmetry is controlled in the following manner: NPNLS and NHT surfaces are automatically assumed to have counterparts on the opposite side of the fuselage. The NVT non-lifting surfaces are assumed to be single surfaces if y=0 (SUR(I,12)). If y$>$ 0, an identical surface is assumed to be on the opposite side. Irregardless of symmetry, note that the wetted area must be the total value, not a single, one-sided value.

Data Block No. 4, Arbitrary Airfoil

Data Block No. 4 must follow immediately after Data Block No. 3 if NAFO > 0 on Card 1.4,

CARD 4.1 - (F10 format)

Column	Symbol	Definition
1-10	RLE	Leading-edge radius divided by chord (RLE/CHORD), at reference t/c (TOCR).
11-20	TOCR	Reference t/c of AFT (see Card 4.4).
21-30	CLDR	Reference design lift coefficient of AFC (see Card 4.3).
31-40	TECH	Technology Factor (See Sections 5.1 and 6.2 of Volume I) = 0.0 Conventional airfoil = 1.0 Advanced supercritical wing section

CARD 4..2

Format	Symbol	Definition
7F10	AFX(I)	Chordwise stations at which the camber and thickness data are to be entered. Enter as x/c. (I=1,NAFO)

Card 4.2 should be repeated until NAFO (Card 1.4) ordinates have been entered.

CARD 4.3

Format	Symbol	Definition
7F10	AFC(I)	Camber of the arbitrary airfoil section. Defined by Z/C where Z is the distance from the chord line to the camber line. (I =1, NAFO)

Card 4.3 should be repeated in the same format as Card 4.2 until NAFO values of AFC(I) have been input. The camber ordinates must correspond with the chordwise stations on Card 4.2.

CARD 4.4

Format	Symbol	Definition
7F10	AFT(1)	Thickness of the arbitrary airfoil section. Defined by Z/C, where Z is the distance from the chord line to the upper or lower surface on an uncambered airfoil. (I=1,NAFO)

Card 4.4 should be repeated in the same format as Card 4.2 until NAFO values of AFT(I) have been input. The thickness ordinates must correspond with the chordwise stations on Card 4.2.

Data Block No. 5., Variable Sweep

Data Block No. 5 is required only if the variable-sweep option is indicated by ISWP=1 on Card 1.4. This Block contains only one card. Use of the variable-sweep option does not change the reference area or the reference moment center, which are defined in Data Block No. 1.

CARD 5.1 - (F10 format)

Column	Symbol	Definition
1-10	XPIVOT	X-location of wing pivot
11-20	YPIVOT	Y-location of wing pivot
21-30	AFTSW	Maximum aft sweep
31-40	AFTCB*	Mean aerodynamic chord of movable panel in aft-sweep position.
41-50	AFTOC*	Thickness ratio (t/c) of movable panel in aft-sweep position.
51-60	AFTAW*	Wetted area of movable panel in aft-sweep position.

Data Block No. 6, Survey Conditions

This data block is always required, and will consist of from 2 to 22 cards, dependent upon the number of survey conditions specified.

CARD 6.1 - (I5 format)

Column	Symbol	Definition
4-5	NSURV	Number of surveys (sets of conditions) at which the problem is to be run (NSURV $\leq$ 20).
9-10	NCLAS	Number of evenly spaced C_L values for which data will be computed (NCLAS $\leq$ 21).

CARD 6.2 - (F10 format)

Column	Symbol	Definition
1-10	FMSURV(I)	Mach number
11-20	ALT(I)	Altitude
21-30	CG(I)	Position for trim or moment reference. Measured as a fraction of the mean aerodynamic chord (CMAC) relative to leading-edge of MAC.
31-40	SWPV(I)	Leading-edge sweep.
41-50	CLLO(I)	Low C_L
51-60	CLHI(I)	High C_L

Card 6.2 is repeated until NSURV conditions have been specified. Several items that may prove useful to the user are:

1. The Reynolds number may be specified instead of altitude by entering the negative value of RN/ft divided by 10^6. For example, if RN/ft=3.0 x 16^6, enter -3.0 in Column 11-20.

2. Model-scale predictions can be obtained even though full-scale geometric data are loaded in the program. Enter the negative of the RN/ft multiplied by model scale divided by 10^6. For example, if RN/ft=3.0 x 10^6 and model scale desired is 1/15, enter -0.2 in Columns 11-20 of Card 6.2.

3. The low C_L and high C_L should be defined in conjunction with NCLAS so that the calculated values of C_L will be rounded off to convenient numbers. For example,

INPUT	Data Calculated at
NCLAS=21	C_L =0.00
CLLO=0.0	C_L =0.05
CLHI=1.0	C_L =0.10
	C_L =0.15
	.
	.
	.
	C_L =1.00

4. If low C_L and high C_L are left blank, the program will automatically set CLLO=0.0 and CLHI=1.0.

Data Block No. 7, Adjustment Factors

Data Block No. 7 must always be entered in the input; it may, however, contain only one card which indicates that no adjustments will be made.

CARD 7.1

Write "ADJUST" beginning in Column 1 to indicate if adjustment factors are to be applied to some of the aerodynamic parameters predicted by the program. If no adjustment factors are to be read in, write "NO ADJUSTMENTS" beginning in Column 1 and continue with the next data block.

The adjustment options allow certain predicted items in the computer procedure to be adjusted to match a desired value. Thus, the predictions can be adjusted to match wind tunnel data, for instance, so that perturbations in geometry for trade studies can be predicted from a firm baseline. An aerodynamic parameter of interest (A_{PRED}) can be adjusted to match an experimental value (A_{EXP}) by the equation

$$A_{EXP} = (A_{PRED}) \cdot YM + YA$$

where YM is an input correlation multiplier and YA is an adder. Each is a function of Mach number or lift coefficient.

CARD 7.2 - (I format)

Set a given IVAL indicator equal to 1 to identify it as being an aerodynamic parameter to be adjusted.

Column	Symbol	Definition
1	IVAL(1)	Adjust C_{M_O} as a function of Mach number.
2	IVAL(2)	Adjust $C_{D_{MISC}}$ as a function of Mach number.
3	IVAL(3)	Adjust α_{LO} as a function of Mach number.
4	IVAL(4)	Adjust M_{CR} as a function of lift coefficient.
5	IVAL(5)	Adjust C_{D_L} as a function of lift coefficient.
24-25	NXVAR	Number of Mach values in the table of Mach function adjust factors ($\leq$ 15).

(CARD 7.2 - (I format), continued)

Column	Symbol	Definition
29-30	NADJ	Number of parameters to be adjusted as a function of Mach number ($\leq$ 3).
34-35	NXCL	Number of C_L values in the table of lift function adjust factors ($\leq$ 15).
39-40	NADJ2	Number of parameters to be adjusted as a function of lift coefficient ($\leq$ 2).

CARD 7.3 - (7F10.0 format) Required if NADJ $>$ 0.

Column	Symbol	Definition
1-10	X(1)	Mach numbers for the table of Mach function adjust factors.
11-20	X(2)	↓
21-30	X(3)	↓
31-40	X(4)	↓
41-50	X(5)	↓
51-60	X(6)	↓
61-70	X(7)	↓

Repeat Card 7.3 until NXVAR values of X are read in.

CARD 7.4 (6F10.0 format) (NADJ $>$ 0)

Column	Symbol	Definition
1-10	YM(J=1, I=1)	Multiplier factor.
11-20	YA(J=1, I=1)	Adder factor.
21-30	YM(J=2, I=1)	Multiplier factor.
31-40	YA(J=2, I=1)	Adder factor.
41-50	YM(J=3, I=1)	Multiplier factor.
51-60	YA(J=3, I=1)	Adder factor.

Repeat Card 7.4 until NXVAR values are read in (J=NXVAR) for each aerodynamic parameter to be adjusted (I=1 to NADJ).

CARD 7.5 - (7F10 format) Required if NADJ2 > 0.

Column	Symbol	Definition
1-10	XCL(1)	C_L values for the table of lift function adjust factors.
11-20	XCL(2)	↓
21-30	XCL(3)	↓
31-40	XCL(4)	↓
41-50	XCL(5)	↓
51-60	XCL(6)	↓
61-70	XCL(7)	↓

Repeat Card 7.5 until NXCL values of XCL are read in.

CARD 7.6 - (6F10.0 format) NADJ2 0.

Column	Symbol	Definition
1-10	YM(J=1, I=NXVAR+1)	Multiplier factor.
11-20	YA(J=1, I=NXVAR+1)	Adder factor.
21-30	YM(J=2, I=NXVAR+1)	Multiplier factor.
31-40	YA(J=2, I=NXVAR+1)	Adder factor.
41-50	YM(J=3, I=NXVAR+1)	Multiplier factor.
51-60	YA(J=3, I=NXVAR+1)	Adder factor.

Repeat Card 7.6 until NXCL values are read in (J=NXCL).

Data Block No. 8, Reserved for Expansion

Data Block No. 8 represents a section of the input common block that has been reserved for future expansion. The user should ignore this block.

Data Block No. 9, Damage Mode Indicators

If no damage parameters are to be loaded, skip the remaining input and go to "End of Problem" instructions in Data Block No. 12. If damage parameters are to be loaded for either a body or a surface, complete the following three cards.

CARD 9.1

Enter "DAMAGE CASES FOLLOW" beginning in Column 1.

CARD 9.2

Enter any alphanumeric title in Columns 1 through 60. This title will be printed at the top of each page of output related to damage calculations.

CARD 9.3 - (I format)

This card allows the user to select the modes of damage that will be used. Set IDAM(I)=1 if mode I input are to be entered in Blocks 10 and 11.

Column	Symbol	Definition
1	IDAM(1)	Mode 1, Roughness on bodies
2	(2)	2, Fwd-facing steps on bodies
3	(3)	3, Aft-facing steps on bodies
4	(4)	4, Holes in bodies
5	(5)	5, Surface waviness on bodies
6	(6)	6, Protuberances on bodies
7	(7)	7, Nose bluntness on bodies
11	IDAM(11)	Mode 11, Roughness on surfaces
12	(12)	12, Fwd-facing steps on surfaces
13	(13)	13, Aft-facing steps on surfaces
14	(14)	14, Holes in surfaces

(CARD 9.3 - I format), continued

Column	Symbol	Definition
15	IDAM(15)	Mode 15, Surface waviness on surfaces
16	↓ (16)	↓ 16, Protuberances on surfaces
17	↓ (17)	↓ 17, Missing parts of surfaces

Data Block No. 10, Body Damage

This block is required if one or more of the indicators IDAM(1) through IDAM(7) equals 1, otherwise, it can be omitted. Several cards are required to define the body damage for each mode selected. The input format is similar to that in Block No. 2. If only part of the bodies are damaged, the columns for the undamaged bodies are to be left blank. The space to the right of the last field of data may be used to identify the damage parameter on that card.

CARD 10.1

Enter the body names in 10-column fields and in the same order as on Card 2.1

Body roughness, required if IDAM(1)=1.

Card	Format	Symbol	Definition
10.2	7F10	DBOD(I,1)	Roughness factor on body I *before* damage is incurred.
10.3	↓	(I,2)	Roughness factor on body I *after* damage is incurred.
10.4	↓	(I,3)	x/1 where damage roughness starts
10.5	↓	(I,4)	x/1 where damage roughness ends
10.6	↓	(I,5)	Fraction of area affected by increased roughness (includes area forward and aft of damaged region).

Table 1 is presented as an aid in selecting the roughness factors after damage has been incurred.

Fwd-facing steps, required if IDAM(2)=1

Card	Format	Symbol	Definition
10.7	7F10	DBOD(I,11)	Number of fwd-facing steps on body I.
10.8	↓	(I,12)	Width of each step.
10.9	↓	(I,13)	Height of each step.
10.10	↓	(I,14)	x/1 at first step.
10.11	↓	(I,15)	x/1 at last step.

Aft-facing steps, required if IDAM(3)=1.

Card	Format	Symbol	Definition
10.12	7F10	DBOD(I,21)	Number of aft-facing steps on body I.
10.13	↓	(I,22)	Width of each step.
10.14	↓	(I,23)	Height of each step.
10.15	↓	(I,24)	x/l at first step.
10.16	↓	(I,25)	x/l at last step.

Holes, required if IDAM(4)=1.

Card	Format	Symbol	Definition
10.17	7F10	DBOD(I,31)	Total no. of holes in body I.
10.18	↓	(I,32)	Number of these holes over wing.
10.19	↓	(I,33)	Length of each hole.
10.20	↓	(I,34)	Width of each hole.
10.21	↓	(I,35)	Depth of each hole.
10.22	↓	(I,36)	x/l where first hole starts.
10.23	↓	(I,37)	x/l where last hole starts.
10.24	↓	(I,38)	Type hole 1.0 Missing Panel 2.0 Caved-in Panel

Waviness, required if IDAM(5)=1.

Card	Format	Symbol	Definition
10.25	7F10	DBOD(I,41)	Total no. of waves in body I.
10.26	↓	(I,42)	No. of these waves over wing.
10.27	↓	(I,43)	Length of each wave.
10.28	↓	(I,44)	Width of each wave.
10.29	↓	(I,45)	Amplitude of each wave.
10.30	↓	(I,46)	x/l where first wave starts.
10.31	↓	(I,47)	x/l where last wave starts.

Protuberances, required if IDAM(6)=1.

Card	Format	Symbol	Definition
10.32	7F10	DBOD(I,51)	No. of protuberances on body I.
10.33	↓	(I,52)	Height of each protuberance.
10.34	↓	(I,53)	Parasite area of each protuberance, Δf.
10.35	↓	(I,54)	x/1 of first protuberance.
10.34	↓	(I,55)	x/1 of last protuberance.

Bluntness, required if IDAM(7)=1.

Card	Format	Symbol	Definition
10.37	7F10	DBOD(I,61)	Frontal area of blunted body at point of damage. For example, a fuselage with the radome missing might have a cross-section area of 10 sq ft at the point where the radome is detached.
10.38	↓	(I,62)	1/d at point of damage. d is the body diameter at most forward body section that is not damaged by the bluntness. 1 is the remaining nose length forward of the point where d is measured. For example, 1/d=0.0 for a nose section that is flat and 1/d=0.5 for a hemi-spherical section.

All input parameters describe damage to only one body of type I. If all bodies of type I have sustained the damage described, a negative sign on the first parameter in each mode of damage will indicate this. For example, if there is increased roughness on only one of four wing-mounted pods, DBOD(I,1) through DBOD(I,5) are all positive values. If all four pods are damaged, this is indicated by entering DBOD(I,1) as a negative number.

The parasite area for a protuberance is the drag divided by the dynamic pressure (D/q) in freestream air flow. Effects of local boundary layer are computed. Units for Δf must be consistent with those selected on Card 1.3.

TABLE 1 TYPICAL ROUGHNESS VALUES

Surface or Condition of Surface	Roughness k ~ Inch
Average aircraft wing or tail surface	.0006
Average fuselage, nacelle surface	.0012
Alunimum skin with blistered paint	.0012
Fiberglass/Enamel with blistered paint	.0025
Fiberglass/Thick Coatings and Graphite with blistered paint	.0030
Broken Skin	.01
Exposed Honeycomb	.1

Data Block No. 11, Surface Damage

This block is required if one or more of the indicators IDAM(11) through IDAM(17) equals 1, otherwise, it can be omitted. The format is identical to that in Data Block No. 10.

CARD 11.1

Enter the surface names in 10-column fields and in the same order as on Card 3.1

Surface Roughness, required if IDAM(11)=1

Card	Symbol	Definition
11.2	DSUR(I,1)	Roughness factor on surface I before damage is incurred.
11.3	(I,2)	Roughness factor on surface I after damage is incurred.
11.4	(I,3)	x/c where damage roughness starts.
11.5	(I,4)	x/c where damage roughness ends.
11.6	(I,5)	Fraction of area affected by increased roughness (includes area forward and aft of damaged region).
11.7	(I,6)	0.0 for Lower Surface. 1.0 for Upper Surface.

Table 1 is presented as an aid in selecting the roughness factors after damage has been incurred.

Fwd-facing steps, required if IDAM(12)=1.

Card	Symbol	Definition
11.8	DSUR(I,11)	Number of fwd-facing steps on lower surface.
11.9	(I,12)	Number of fwd-facing steps on upper surface.
11.10	(I,13)	Width of each step.
11.11	(I,14)	Height of each step.
11.12	(I,15)	x/c at first step.
11.13	(I,16)	x/c at last step.

Aft-facing steps, required if IDAM(13)=1.

Card	Symbol	Definition
11.14	DSUR(I,21)	Number of aft-facing steps on lower surface.
11.15	(I,22)	Number of aft-facing steps on upper surface.
11.16	(I,23)	Width of each step.
11.17	(I,24)	Height of each step.
11.18	(I,25)	x/c at first step.
11.19	(I,26)	x/c at last step.

Holes, required if IDAM(14)=1.

Card	Symbol	Definition
11.20	DSUR(I,31)	Number of holes in lower surface.
11.21	(I,32)	Number of holes in upper surface.
11.22	(I,33)	Length of each hole.
11.23	(I,34)	Width of each hole.
11.24	(I,35)	Depth of each hole.
11.25	(I,36)	x/c where first hole starts.
11.26	(I,37)	x/c where last hole starts.
11.27	(I,38)	Type Hole 1.0 Missing Panel 2.0 Caved-in Panel
11.28	(I,39)	Porosity Factor (must have holes on both upper & lower surface)

Waviness, required if IDAM(15)=1.

Card	Symbol	Definition
11.29	DSUR(I,41)	Number of waves in lower surface.
11.30	(I,42)	Number of waves in upper surface.
11.31	(I,43)	Length of each wave.
11.32	(I,44)	Width of each wave.

Card	Symbol	Denifition
11.33	DSUR(I,45)	Amplitude of each wave.
11.34	↓ (I,46)	x/c where first wave starts.
11.35	↓ (I,47)	x/c where last wave starts.

Protuberances, required if IDAM(16)=1

Card	Symbol	Definition
11.36	DSUR(I,51)	Number of protuberances on Surface I.
11.37	↓ (I,52)	Height of each protuberance.
11.38	↓ (I,53)	Parasite area of each protuberance, Δf.
11.39	↓ (I,54)	x/c of first protuberance.
11.40	↓ (I,55)	x/c of last protuberance.

Missing Wing Parts, required if IDAM(17)=1

CARD 11.41

Column	Symbol	Definition
1-10	DWING(1)	$\Delta c/c$ fraction of wing chord that is missing from leading edge.
11-20	↓ (2)	η_i, inboard edge of missing leading edge, expressed as a fraction of semispan.
21-30	↓ (3)	$\Delta\eta$, span of missing leading edge, expressed as a fraction of semispan.
31-40	↓ (4)	z/t ratio of leading edge to maximum thickness. z represents the wing thickness at the point where the missing leading-edge section is detached. t is the wing maximum thickness.

Column	Symbol	Definition
41-50	DWING(5)	Corner sharpness factor for leading edge. Enter 1.0 if corners on leading edge are rounded. Enter 2.0 if corners on leading edge are sharp.
51-60	(6)	Indicates that leading edge is missing on: 1.0, left side only 2.0, right side only 3.0, both sides

CARD 11.42

Column	Symbol	Definition
1-10	DWING(7)	$\Delta c/c$, fraction of wing chord that is missing from trailing edge.
11-20	(8)	η_i, inboard edge of missing trailing edge expressed as a fraction of semispan.
21-30	(9)	$\Delta\eta$, span of missing trailing edge, expressed as a fraction of semispan.
31-40	(10)	Indicates that trailing edge is missing on: 1.0, left side only 2.0, right side only 3.0, both sides

CARD 11.43

Column	Symbol	Definition
1-10	DWING(11)	$\Delta\eta$, fraction of semispan that is missing from the left wing tip.
11-20	(12)	$\Delta\eta$, right wing tip.
21-30	(13)	Moment arm of device that trims rolling moment. Entered as a fraction of wing semispan.

CARD 11.44

Column	Symbol	Definition
1-10	DWING(14)	The first NHT surface requires input for the left side (Column 1-10) and the right side (Column 11-20). Remaining surfaces must be in the same order as those on Card 3.1. If NHT=0, list NVT surfaces in order beginning with Columns 1-10.
11-20	(15)	
21-30	(16)	
31-40	(17)	
41-50	(18)	
51-60	(19)	
61-70	(20)	

There are several points that should be noted regarding the surface damage parameters:

1. All input parameters describe damage to only one surface (such as the right wing). If the damage is symmetric (left wing also damaged), then a negative sign should be used on the first parameter in each mode of damage. The first parameter will be zero in some cases. When this happens, a negative sign on the second parameter will indicate symmetric damage.

2. Porosity factor specifies the fraction of hole area that allows air to flow from the lower to the upper surface. The hole area is the smaller of the area of upper surface.holes and lower surface holes.

3. If IDAM(17)=1, Cards 11.42 through 11.45 must all be entered. A zero or blank should be entered for the parameters that are not applicable to the damage case being described.

4. The parasite area for a protuberance is the drag divided by the dynamic pressure (D/q) in freestream air flow. Effects of local boundary layer are computed. Units for Δf must be consistent with those selected on Card 1.3.

Data Block No. 12, End of Problem

Several options are available when the end of the initial problem is attained. The user may either end the problem, add more damage cases, or change the basic input for an additional run.

CARD 12.1

Option 1. Input "END" in columns 1 through 3 and the problem will be terminated.

Option 2. Input "DAMAGE CASES FOLLOW" beginning in Column 1 and continue input with Card 9.2 to run additional damage cases.

Option 3. "CHANGE INPUT" beginning in Column 1 allows the user to change any variable in the basic input (Data Block No. 1 through 7). The variable-location is its position in the input common block. This location is given in Volume II, Section 4 of the AAT Computer Code Final Report. The remaining cards in this block must be input.

CARD 12.2

Enter any alphanumeric characters in Columns 1 through 60 to identify the new problem. This title will replace the one originally entered on Card 1.1 when the new problem is run.

CARD 12.3

Column	Format	Symbol	Definition
4-5	I2	M	Number of integer variables to be changed.
9-10	I2	N	Number of floating point variables to be changed.

CARD 12.4 (Req'd if M > 0)

Column	Format	Symbol	Definition
1-5	I5	MO(1)	Location of 1st integer variable.
6-10	I5	IA(1)	New value of 1st integer variable.

Column	Format	Symbol	Definition
11-15	I5	MO(2)	Location of 2nd integer variable.
16-20	I5	IA(2)	New value of 2nd integer variable.
21-25	I5	MO(3)	Location of 3rd integer variable.
26-30	I5	IA(3)	New value of 3rd integer variable.

This pattern is repeated for 6 variables per card, until M pairs of variables have been loaded.

CARD 12.5 - (Req'd if N > 0)

Column	Format	Symbol	Definition
1-5	I5	NO(1)	Location of 1st floating point variable.
11-20	F10	AA(1)	New value of 1st floating point variable.
21-25	I5	NO(2)	Location of 2nd floating point variable.
31-40	F10	AA(2)	New value of 2nd floating point variable.
41-45	I5	NO(3)	Location of 3rd floating point variable.
51-60	F10	AA(3)	New value of 3rd floating point variable.

Card 12.5 is repeated until N pairs of variables have been loaded.

CARD 12.6

Input "END" in columns 1 through 3 to terminate problem or omit this card and go to Card 12.1 to run additional problems.

3. SAMPLE PROBLEMS

This section contains the computer printout for the following sample problems:

1. F-16A
2. FB-111
3. C-141 with damage

The output is self-explanatory, except that in the C-141 problem it should be noted that the damage imposed is strictly hypothetical and is not an attempt to simulate an actual damage case. Damage inputs are entered for each of the fourteen AAT damage modes to demonstrate the flexibility of the program.

The F-16A and FB-111 problems presented in this section provided the estimates that are compared with data in Section 9, Volume I of this report.

```
F-16 WIND TUNNEL MODEL WITH TIP MISSILES AND LAUNCHERS REMOVED          CARD 1.1
                                                                        CARD 1.2
    3    1    1    2                                                    CARD 1.3
300.      11.31667  22.76     0.0                                       CARD 1.4
0.00      0.0       -3.0                                                CARD 1.5
FUSELAGE  CANOPY    NACELLE                                             CARD 2.1
42.07     11.675    33.557         LENGTH (FT.)                         CARD 2.2
4.833     1.975     4.9            WIDTH                                CARD 2.3
3.333     1.975     2.167          HEIGHT                               CARD 2.4
532.06    35.12     237.42         AWET                                 CARD 2.5
1.0       1.3       1.0            INTF.                                CARD 2.6
1.        1.        1.             NC.                                  CARD 2.7
15.972    2.917     8.75           MAX.AREA (FT*FT)                     CARD 2.8
8.058     0.0       1.917          BASE(EXIT)                           CARD 2.9
18.75     3.5       5.25           L.NOSE                               CARD 2.10
7.917     5.333     20.417         L.TAIL                               CARD 2.11
0.        0.        0.             BASE DRAG AREA                       CARD 2.12
0.        0.        5.741          INLET AREA (FT*FT)                   CARD 2.13
WING      HORIZ.    VERT.     VENTRAL                                   CARD 3.1
64A       BICONVEX  BICONVEX  BICONVEX        SECTION                   CARD 3.2
0.2       0.        0.        0.              CAMBER                    CARD 3.3
0.04      0.053     0.045     0.019           THICKNESS                 CARD 3.4
40.       40.       47.5      45.             L.E.SWEEP                 CARD 3.5
344.89    98.       117.02    22.2            AWET                      CARD 3.6
14.5445   6.683     9.052     4.708           CR (FT.)                  CARD 3.7
3.702     2.004     3.958     3.519           CT                        CARD 3.8
12.917    5.642     8.417     1.618           B/2                       CARD 3.9
19.534    28.326    34.082    32.275          XLE                       CARD 3.10
2.083     3.458     0.0       1.417           YLE                       CARD 3.11
0.0       0.0       2.125     -2.133          ZLE (W.L.91)              CARD 3.12
0.0                                           INCIDENCE                 CARD 3.13
    4    21                                                             CARD 6.1
0.85      -0.16667  0.35                0.0        1.0                  CARD 6.2
0.90      -0.16667  0.35                0.0        1.0                  CARD 6.2
1.2       -0.16667  0.35                0.0        1.0                  CARD 6.2
1.6       -0.13333  0.35                0.0        1.0                  CARD 6.2
NO ADJUSTMENTS                                                          CARD 7.1
END                                                                     CARD 12.1
```

EMPIRICALLY BASED COMPUTER PROGRAM

TO PREDICT THE AERODYNAMIC CHARACTERISTICS

OF AIRCRAFT

INCLUDING

ROUGHNESS, PROTUBERANCES, AND

DAMAGE CALCULATIONS

F-16 WIND TUNNEL MODEL WITH TIP MISSILES AND LAUNCHERS REMOV

PROBLEM INPUT PARAMETERS

NPODS =	3	SREF =	300.00	ROUGHK=	0.00060
NPNLS =	1	CMAC =	11.317	FMISC =	0.00
NHT =	1	XMAC =	22.760	TWIST =	-3.000
NVT =	2	XCG =	0.000	CONCL =	0.600
ISWP =	0	UPFUS =	0.000		
IREF =	0	ACE =	0.000		
IWNG =	0				
NAFC =	0				
METER =	0				
ITRIM =	0				

DIMENSIONS ARE IN FEET

		FUSELAGE	CANOPY	NACELLE
LENGTH	=	42.078	11.675	23.557
WIDTH	=	4.833	1.575	4.500
HEIGTH	=	3.333	1.575	2.167
WETTED AREA	=	532.060	35.120	237.420
INTER.FACTOR	=	1.000	1.300	1.000
NO.OF THIS TYPE	=	1.000	1.000	1.000
MAXIMUM AREA	=	15.972	2.517	8.750
BASE AREA	=	8.056	0.000	1.917
NOSE LENGTH	=	18.758	3.500	5.250
BOATTAIL LENGTH	=	7.917	5.333	20.417
BASE DRAG AREA	=	0.000	0.000	0.000
INLET AREA	=	0.000	0.000	5.741

	WING	HORIZ.	VERT.	VENTRAL
SECTION =	64A	BIC	BIC	BIC
CAMBER (2-D) =	.200	0.000	0.000	0.000
THICKNESS RATIO =	.040	.053	.045	.019
L.E. SWEEP =	40.000	40.000	47.500	45.000
WETTED AREA =	344.890	58.000	117.020	32.200
ROOT CHORD =	14.545	6.623	9.052	4.708
TIP CHORD =	3.702	2.004	3.556	3.519
SPAN OF PANEL =	12.917	5.642	8.417	1.618
X- LE ROOT CHORD=	19.534	28.226	34.063	32.375
Y- LE ROOT CHORD=	2.083	3.458	0.000	1.417
Z- LE ROOT CHORD=	0.000	0.000	2.125	-2.133
INCIDENCE =	0.000	0.000	0.000	0.000

4 DRAG POLARS TO BE GENERATED AT THE FOLLOWING CONDITIONS

MACH NO.	ALTITUDE	C.G.	SWEEP	FROM CL	TO CL
.85	-.167	.25	40.0	0.00	1.00
.90	-.167	.25	40.0	0.00	1.00
1.20	-.167	.25	40.0	0.00	1.00
1.60	-.133	.25	40.0	0.00	1.00

NO ADJUSTMENTS *****

CALCULATED BODY PARAMETERS

		FUSELAGE	CANOPY	NACELLE
FINENESS RATIO	=	10.482	5.911	10.258
WETTED AREA	=	532.080	35.120	231.420
MAXIMUM AREA	=	15.972	2.917	6.750

CALCULATED SURFACE PARAMETERS

		WING	HORIZ.	VERT.	VENTRAL
WETTED AREA	=	344.890	98.000	127.020	32.200
TAPER RATIO	=	.255	.300	.437	.747
PLANFORM AREA	=	225.690	49.012	54.753	13.211
ASPECT RATIO	=	2.832	2.598	2.588	.787
CHARACT. LENGTH	=	10.198	4.764	6.838	4.142
SWP C/4	=	32.180	22.284	43.229	39.224
SWP C/2	=	22.753	22.998	38.263	32.316
SWP TE	=	-.017	.561	25.925	14.850
SWP MAX T/C	=	27.100	22.998	38.263	32.316
X/C AT MAX T/C	=	.290	.500	.500	.500

TOTAL AIRPLANE PARAMETERS

```
WING SEMI-SPAN                          ,BO2    =     15.000
BODY DIA/ WING SPAN                     ,DOB    =       .129
WING ROOT CHORD AT CENTERLINE           ,CR     =     16.253
THEO. WING PLANFORM AREA                ,SPLAN  =    299.924
X-STA OF ROOT CHORD (CR) T.E.           ,XCRTE  =     34.079
EQUIV. WING ASPECT RATIO                ,AR     =      3.001
EQUIV. WING TAPER  RATIO                ,TAPR   =       .227
EQUIV. WING LEADING EDGE SWEEP          ,SWPLE  =     40.000
EQUIV. WING QUARTER-CHORD SWEEP         ,SWPQC  =     32.180
EQUIV. WING MID-CHORD      SWEEP        ,SWPMC  =     22.753
EQUIV. WING TRAILING-EDGE SWEEP         ,SWPTE  =      -.017
EQUIV. WING MAX.THICKNESS SWEEP         ,SWPT   =     27.100
EQUIV. WING EXPOSED AREA                ,SEXW   =    235.690
EQUIV. WING EXPOSED MAC                 ,CBAR   =     10.198
EQUIV. WING THICKNESS RATIO             ,TOCW   =       .040
EQUIV. WING 2-D DESIGN CL               ,CLD    =       .200
EQUIV. WING EXPOSED ASPECT RATIO        ,ARXP   =      2.632
CHORD LENGTH AT WING ROOT               ,CRX    =     14.545
CHORD LENGTH AT WING TIP                ,CTX    =      3.702
X-STA. AT L.E. OF WING ROOT             ,XRX    =     19.534
X-STA. AT L.E. OF WING TIP              ,XTX    =     30.373
Y-STA. AT L.E. OF WING ROOT             ,YRX    =      2.063
Y-STA. AT L.E. OF WING TIP              ,YTX    =     15.000
X-STA. AT L.E. OF WING MAC              ,XB     =     22.758
Y-STA. AT L.E. OF WING MAC              ,YB     =      5.926
DELTA-X  C/4 WING TO C/4 TAIL           ,XH     =     18.499
INCLINATION LINE BET. WING AND TAIL,OMEGA =      0.000
   ENTER SURVEY

       6A SERIES AIRFOIL SECTION
```

MACH CRITICAL TABLE

CL	MACH CRITICAL
0.0000	.9024
.1000	.8922
.2000	.8794
.3000	.8664
.4000	.8525
.5000	.8350
.6000	.8151
.7000	.7854
.8000	.7441
.9000	.6898
1.0000	.6243

F-16 WIND TUNNEL MODEL WITH TIP MISSILES AND LAUNCHERS FENCE
MACH NO. = .850   RN/FT = 1.6667E+05   L.E. SWEEP ANGLE = 40.00 DEG.   TRIMMED CONDITION

CL	TOTAL CD	CM	ALPHA	CD LIFT	CD R+AFT	CL AT CM=0	CD AT CM=0
-.019	.02168	-1.75130	-.58	.00073	0.00000	0.00000	.02120
.034	.02088	-1.47213	.11	.00005	0.00000	.05000	.02061
.087	.02093	-1.19315	.80	.00016	0.00000	.10000	.02074
.140	.02180	-.91426	1.48	.00113	0.00000	.15000	.02170
.193	.02352	-.63533	2.17	.00290	0.00000	.20000	.02347
.246	.02607	-.35625	2.86	.00549	0.00000	.25000	.02605
.299	.02946	-.07691	3.54	.00850	0.00000	.30000	.02946
.352	.03359	.20283	4.23	.01312	0.00000	.35000	.03368
.405	.03875	.48307	4.92	.01816	0.00000	.40000	.03872
.458	.04472	.76417	5.62	.02402	.00007	.45000	.04460
.512	.05160	1.04769	6.45	.03074	.00042	.50000	.05166
.565	.06054	1.32990	7.42	.03896	.00123	.55000	.06675
.617	.07579	1.52566	8.54	.05251	.00246	.60000	.07554
.668	.09668	1.58251	9.79	.07104	.00540	.65000	.09660
.716	.12377	1.46926	11.19	.09454	.00843	.70000	.12354
.763	.15829	1.15259	12.72	.12282	.01467	.75000	.15826
.807	.19970	.60064	14.42	.15648	.02263	.80000	.19968
.847	.25068	-.22683	16.24	.19491	.03539	.85000	.25687
.885	.30774	-1.37187	18.21	.23833	.04855	.90000	.30744
.923	.37260	-2.43916	25.32	.28171	.06442	.95000	.37170
.972	.44150	-2.51054	27.57	.34007	.08029	1.00000	.44093

CLA = .07285 PER DEG.   ALO = -.57503 DEG.   CLCB = .53503
K = .16362   DELCL = .06694   CLMAX = .93466
CMO = -.02517*   CM/CL = .07630*   (*TAIL OFF)

DRAG BREAKDOWN -----

		FUSELAGE	CANOPY	NACELLE	
FRICTION	= .00821	.00531	.00044	.00246	
FORM	= .00063	.00042	.00013	.00008	
INTERF	=-.00021	-.00038	.00017	0.00000	
WAVE	=0.00000	0.00000	0.00000	0.00000	
BASE	=0.00000	0.00000	0.00000	0.00000	
		WING	HORIZ.	VERT.	VENTRAL
FRICTION	= .00792	.00429	.00144	.00160	.00049
FORM	= .00046	.00027	.00009	.00009	.00001
INTERF	= .00220	.00109	.00048	.00053	.00015
WAVE	=0.00000	0.00000	0.00000	0.00000	0.00000
CAMBER	= .00135				
DRAG RISE	=0.00090				
MISC	=0.00000				

CDMIN = .02056

F-16 WIND TUNNEL MODEL WITH TIP MISSILES AND LAUNCHERS REMOV
MACH NO. = .900   RN/FT = 1.6667E+05   L.E. SWEEP ANGLE = 40.00 DEG.   TRIMMED CONDITION

CL	TOTAL CD	CM	ALPHA	CD LIFT	CD RMAFT	CL AT CM=0	CD AT CM=0
-.022	.02198	-1.92064	-.58	.00067	0.00000	0.00000	.02152
.032	.02125	-1.60979	.08	.00003	.00001	.05000	.02491
.085	.02143	-1.29914	.74	.00020	.00012	.10000	.02416
.139	.02260	-.98880	1.40	.00117	.00041	.15000	.02444
.192	.02475	-.67807	2.06	.00299	.00085	.20000	.02466
.246	.02751	-.36744	2.72	.00553	.00148	.25000	.02787
.299	.03209	-.05661	3.38	.00891	.00228	.30000	.03204
.353	.03726	.25451	4.04	.01309	.00342	.35000	.03727
.406	.04349	.56604	4.70	.01607	.00456	.40000	.04249
.460	.05103	.87937	5.37	.02286	.00632	.45000	.05134
.514	.05970	1.20295	6.16	.03045	.00837	.50000	.05968
.568	.07001	1.54191	7.08	.03785	.01125	.55000	.06595
.621	.08553	1.84402	8.13	.05001	.01456	.60000	.06543
.673	.10874	2.03192	9.22	.06766	.02010	.65000	.10883
.724	.13816	2.07421	10.63	.09071	.02653	.70000	.13810
.772	.17640	1.94070	12.09	.11912	.03646	.75000	.17644
.818	.22009	1.59861	13.67	.15290	.04645	.80000	.22624
.862	.27221	1.01238	15.38	.19206	.05961	.85000	.27253
.902	.33027	.14261	17.23	.23659	.07277	.90000	.33022
.938	.39660	-1.05930	19.21	.28650	.08864	.95000	.39660
.991	.46771	-.80447	26.32	.34177	.10452	1.00000	.46715

CLA = .07574 PER DEG.   ALO = -.58392 DEG.   CLCB = .55311
K = .16054   DELCL = .06447   CLMAX = .95532
CM0 = -.02783*   CM/CL = .07449*   (*TAIL OFF)

DRAG BREAKDOWN -----		FUSELAGE	CANOPY	NACELLE	
FRICTION	= .00817	.00528	.00043	.00245	
FORM	= .00062	.00041	.00013	.00008	
INTERF	=-.00021	-.00038	.00017	0.00000	
WAVE	=0.00000	0.00000	0.00000	0.00000	
BASE	=0.00000	0.00000	0.00000	0.00000	
		WING	HORIZ.	VERT.	VENTRAL
FRICTION	= .00788	.00437	.00143	.00160	.00048
FORM	= .00046	.00027	.00009	.00009	.00001
INTERF	= .00263	.00128	.00056	.00062	.00018
WAVE	=0.00000	0.00000	0.00000	0.00000	0.00000
CAMBER	= .00130				
DRAG RISE	=0.00000				
MISC	=0.00000				

CDMIN = .02086

F-16 WIND TUNNEL MODEL WITH TIP MISSILES AND LAUNCHERS REMOV
MACH NO. = 1.200   RN/FT = 1.6667E+05   L.E. SWEEP ANGLE = 40.00 DEG.   TRIMMED CONDITION

CL	TOTAL CD	CM	ALPHA	CD LIFT	CD RMAFT	CL AT CM=0	CD AT CM=0
0.000	.03929	0.00000	0.00	.00003	0.00000	0.00000	.03938
.047	.03962	-.23427	.65	.00026	0.00000	.05000	.03961
.095	.04099	-.46866	1.30	.00161	0.00000	.10000	.04054
.142	.04348	-.70328	1.95	.00404	0.00000	.15000	.04227
.190	.04709	-.93827	2.61	.00756	0.00000	.20000	.04690
.237	.05182	-1.17373	3.26	.01219	0.00000	.25000	.05152
.285	.05768	-1.40979	3.91	.01791	0.00000	.30000	.05725
.332	.06486	-1.64657	4.56	.02473	0.00000	.35000	.06407
.380	.07276	-1.88420	5.21	.03265	0.00000	.40000	.07199
.427	.08199	-2.12279	5.86	.04167	0.00000	.45000	.08101
.474	.09224	-2.36247	6.51	.05179	0.00000	.50000	.09112
.522	.10381	-2.60336	7.17	.06300	0.00000	.55000	.10234
.569	.11641	-2.84559	7.82	.07531	0.00000	.60000	.11465
.617	.13071	-3.08930	8.47	.08930	0.00000	.65000	.12863
.664	.14757	-3.33460	9.12	.10582	0.00000	.70000	.14516
.711	.16572	-3.58163	9.77	.12360	0.00000	.75000	.16293
.759	.18514	-3.83052	10.42	.14262	0.00000	.80000	.18195
.806	.20584	-4.08141	11.07	.16289	0.00000	.85000	.20223
.853	.22782	-4.33444	11.73	.18442	0.00000	.90000	.22375
.900	.25108	-4.58975	12.38	.20719	0.00000	.95000	.24653
.948	.27583	-4.84747	13.03	.23122	0.00000	1.00000	.27055

CLA = .07676 PER DEG.   ALC = 0.00000 DEG.   CLCB = .63496
K = .21564   DELCL = .01442   CLMAX = 1.22258
CMC = 0.00000*   CM/CL = -.09207*   (*TAIL OFF)

DRAG BREAKDOWN -----

	TOTAL	FUSELAGE	CANOPY	NACELLE	
FRICTION	.00787	.00509	.00042	.00236	
FORM	0.00000	0.00000	0.00000	0.00000	
INTERF	0.00000	0.00000	0.00000	0.00000	
WAVE	.01282	.00720	.00250	.00212	
BASE	0.00000	0.00000	0.00000	0.00000	
		WING	HORIZ.	VERT.	VENTRAL
FRICTION	.00768	.00422	.00138	.00154	.00047
FORM	0.00000	0.00000	0.00000	0.00000	0.00000
INTERF	0.00000	0.00000	0.00000	0.00000	0.00060
WAVE	.01099	.00759	.00213	.00123	.00005
CAMBER	.00004				
DRAG RISE	0.00000				
MISC	0.00000				

CDMIN = .03923

F-16 WIND TUNNEL MODEL WITH TIP MISSILES AND LAUNCHERS REMOV
MACH NO. = 1.600   RN/FT = 1.3333E+05   L.E. SWEEP ANGLE = 40.00 DEG.   TRIPPED CONDITION

CL	TOTAL CD	CM	ALPHA	CD LIFT	CD RNAFT	CL AT CM=0	CD AT CM=0
0.000	.03543	0.00000	0.00	0.00000	0.00000	0.00000	.03543
.043	.03618	-.76343	.83	.00072	0.00000	.05000	.03615
.087	.03849	-1.52717	1.67	.00288	0.00000	.10000	.03832
.130	.04211	-2.29156	2.50	.00649	0.00000	.15000	.04192
.174	.04731	-3.05692	3.34	.01153	0.00000	.20000	.04697
.217	.05359	-3.82257	4.17	.01802	0.00000	.25000	.05346
.260	.06215	-4.59183	5.01	.02595	0.00000	.30000	.06139
.304	.07180	-5.36204	5.84	.03532	0.00000	.35000	.07076
.347	.08294	-6.13453	6.68	.04614	0.00000	.40000	.08157
.390	.09557	-6.90964	7.51	.05839	0.00000	.45000	.09383
.434	.10989	-7.68771	8.35	.07209	0.00000	.50000	.10752
.477	.12530	-8.46908	9.18	.08723	0.00000	.55000	.12266
.520	.14240	-9.25411	10.02	.10381	0.00000	.60000	.13924
.563	.16107	-10.04317	10.85	.12190	0.00000	.65000	.15724
.607	.18244	-10.83661	11.69	.14265	0.00000	.70000	.17668
.650	.20652	-11.63481	12.52	.16644	0.00000	.75000	.20186
.693	.23451	-12.43816	13.36	.19328	0.00000	.80000	.22871
.736	.26520	-13.24706	14.19	.22216	0.00000	.85000	.25659
.779	.29900	-14.06190	15.03	.25608	0.00000	.90000	.29152
.822	.33551	-14.88211	15.86	.29205	0.00000	.95000	.32749
.865	.37594	-15.71111	16.70	.33107	0.00000	1.00000	.36650

CLA = .05990 PER DEG.   ALO = 0.00000 DEG.   CLID = 1.00000
K = .28836   DELCL = 0.00000   CLMAX = 1.29635
CMO = 0.00000*   CM/CL = -.12794*   (*TAIL OFF)

DRAG BREAKDOWN -----

		FUSELAGE	CANOPY	NACELLE
FRICTION	= .00767	.00496	.00041	.00230
FORM	=0.00000	0.00000	0.00000	0.00000
INTERF	=0.00000	0.00000	0.00000	0.00000
WAVE	= .01035	.00563	.00309	.00163
BASE	=0.00000	0.00000	0.00000	0.00000

		WING	HORIZ.	VERT.	VENTRAL
FRICTION	= .00749	.00414	.00136	.00152	.00046
FORM	=0.00000	0.00000	0.00000	0.00000	0.00000
INTERF	=0.00000	0.00000	0.00000	0.00000	0.00000
WAVE	= .00993	.00662	.00192	.00136	.00004
CAMBER	=0.00000				
DRAG RISE	=0.00000				
MISC	=0.00000				

CDMIN = .03543

```
 F8-111                                                                   CARD 1.1
                                                                          CARD 1.2
   2     2    1     2    1     0    1              0                      CARD 1.3
550.0    8.8        40.1        0.0                                       CARD 1.4
0.00073  15.0       -5.6                                                  CARD 1.5
550.0    0.25       16.0                                                  CARD 1.6
FUSELAGE NACELLE                                  BODY NAME               CARD 2.1
70.8     33.5                                     LENGTH                  CARD 2.2
11.37    3.55                                     WIDTH                   CARD 2.3
5.5      3.55                                     HEIGHT                  CARD 2.4
1020.0   443.0                                    A WET                   CARD 2.5
1.2      1.5                                      INTERF. FACTOR          CARD 2.6
1.0      2.0                                      NO. OF BODIES           CARD 2.7
37.15    5.55                                     MAX CROSS-SEC.A         CARD 2.8
15.97    7.47                                     BASE STR.TUBE A         CARD 2.9
26.42    3.58                                     NOSE LENGTH             CARD 2.10
22.75    28.08                                    B.TAIL LENGTH           CARD 2.11
0.0      2.6                                      BASE DRAG A             CARD 2.12
         3.51                                     INLET AREA/SIDE         CARD 2.13
GLOVE    VAR.WING  H.T.      V.T.      STRAKE     AIRFOIL NAME            CARD 3.1
643      64A       BIC       BIC       BIC        SECTION                 CARD 3.2
0.0      0.2       0.3       0.0       0.0        CAMBER                  CARD 3.3
0.055    0.106     0.035     0.032     0.02       T/C                     CARD 3.4
70.0     16.0      57.5      55.0      40.0       LE SWP                  CARD 3.5
313.0    800.0     334.0     223.0     50.0       A WET                   CARD 3.6
25.58    10.58     15.01     17.75     14.63      ROOT CHORD              CARD 3.7
10.58    3.13      2.47      7.31      13.29      TIP CHORD               CARD 3.8
5.5667   26.9333   8.98      8.9       1.44       EXP.SEMI-SPAN           CARD 3.9
23.32    28.42     56.17     53.83     50.83      XLE AT ROOT             CARD 3.10
2.55     8.0667    5.67      0.0       4.42       YLE AT ROOT             CARD 3.11
0.42     0.0       0.0       1.25      -4.17      ZLE AT ROOT             CARD 3.12
1.0      1.0       1.0       0.0       0.0        INCIDENCE               CARD 3.13
40.64    5.86      72.5      16.58     0.045      626.0                   CARD 5.1
   8   21                                                                 CARD 6.1
0.8      30000.0   0.45      16.0      0.0        1.0                     CARD 6.2
0.8      30000.0   0.45      26.0      0.0        1.0                     CARD 6.2
0.80     30000.0   0.45      35.0      0.0        1.0                     CARD 6.2
0.80     30000.0   0.45      50.0      0.0        1.0                     CARD 6.2
0.8      30000.0   0.45      72.5      0.0        1.0                     CARD 6.2
1.2      30000.0   0.45      72.5      0.0        1.0                     CARD 6.2
1.6      30000.0   0.45      72.5      0.0        1.0                     CARD 6.2
2.0      30000.0   0.45      72.5      0.0        1.0                     CARD 6.2
NO ADJUSTMENTS                                                            CARD 7.1
END                                                                       CARD 12.1
```

EMPIRICALLY BASED COMPUTER PROGRAM

TO PREDICT THE AERODYNAMIC CHARACTERISTICS

OF AIRCRAFT

INCLUDING

ROUGHNESS, PROTUBERANCES, AND

DAMAGE CALCULATIONS

FB-111

MODELER INPUT PARAMETERS

NPODS =	2	SREF =	550.00	ROUGHK=	.66073
NPNLS =	2	CMAC =	8.000	FMISC =	15.60
NHT =	1	XMAC =	40.000	TWIST =	-5.600
NVT =	2	ZCG =	0.000	CONCL =	6.000
ISWP =	1	UFFUS =	0.000		
IREF =	0	WCE =	0.000		
IWNG =	1				
NAFC =	0				
METER =	0				
ITRIP =	0				

DIMENSIONS ARE IN FEET

REFERENCE WING FOR TWO PANEL CASE

SPLAN = 550.000   TAPR = .2500   SWP = 16.000

		FUSELAGE	NACELLE
LENGTH	=	70.800	23.500
WIDTH	=	11.370	3.550
HEIGTH	=	5.900	3.550
WETTED AREA	=	1320.000	443.000
INTER.FACTOR	=	1.200	1.500
NO.OF THIS TYPE	=	1.000	2.000
MAXIMUM AREA	=	37.150	5.550
BASE AREA	=	15.970	7.470
NOSE LENGTH	=	26.420	3.500
BOATTAIL LENGTH	=	22.750	28.000
BASE DRAG AREA	=	0.000	2.000
INLET AREA	=	0.000	3.510

		GLOVE	VAR.WING	H.T.	V.T.	STRAKE
SECTION	=	64A	64A	BIC	BIC	BIC
CAMBER (2-D)	=	0.000	.200	0.000	0.000	0.000
THICKNESS RATIO	=	.055	.106	.025	.032	.020
L.E. SWEEP	=	70.000	16.000	57.500	55.000	40.000
WETTED AREA	=	313.000	800.000	334.000	223.000	50.600
ROOT CHORD	=	25.580	10.580	15.010	17.790	14.630
TIP CHORD	=	10.580	3.120	2.470	7.210	13.290
SPAN OF PANEL	=	5.567	26.933	8.580	8.500	1.440
X- LE ROOT CHORD	=	23.330	36.420	66.170	53.830	50.830
Y- LE ROOT CHORD	=	2.500	6.667	5.670	0.000	4.420
Z- LE ROOT CHORD	=	.420	0.000	0.000	1.250	-4.170
INCIDENCE	=	1.000	1.000	1.000	0.000	0.000

8 DRAG POLARS TO BE GENERATED AT THE FOLLOWING CONDITIONS

MACH NO.	ALTITUDE	C.G.	SWEEP	FROM CL	TO CL
.80	30000.000	.45	16.0	0.00	1.00
.80	30000.000	.45	26.0	0.00	1.00
.80	30000.000	.45	35.0	0.00	1.00
.80	30000.000	.45	50.0	0.00	1.00
.80	30000.000	.45	72.5	0.00	1.00
1.20	30000.000	.45	72.5	0.00	1.00
1.60	30000.000	.45	72.5	0.00	1.00
2.00	30000.000	.45	72.5	0.00	1.00

NO ADJUSTMENTS *****

CALCULATED BODY PARAMETERS

		FUSELAGE	NACELLE
FINENESS RATIO	=	8.644	9.437
WETTED AREA	=	1520.000	443.000
MAXIMUM AREA	=	37.150	5.550

CALCULATED SURFACE PARAMETERS

		GLOVE	VAR.WING	H.T.	V.T.	STRAKE
WETTED AREA	=	313.000	800.000	334.000	223.000	50.000
TAPER RATIO	=	.414	.256	.185	.411	.508
PLANFORM AREA	=	201.292	369.256	156.970	111.695	40.205
ASPECT RATIO	=	.616	7.858	2.055	1.418	.206
CHARACT. LENGTH	=	19.118	7.530	10.240	13.280	13.971
SWP C/4	=	64.257	12.276	50.673	48.587	21.235
SWP C/2	=	54.466	8.443	41.071	40.010	20.497
SWP TE	=	3.027	.581	9.829	14.070	-5.226
SWP MAX T/C	=	59.484	10.141	41.071	40.010	20.497
X/C AT MAX T/C	=	.390	.390	.500	.500	.510

TOTAL AIRPLANE PARAMETERS

```
WING SEMI-SPAN                         ,BO2    =     35.000
BODY DIA/ WING SPAN                    ,DOB    =       .071
WING ROOT CHORD AT CENTERLINE          ,CR     =     32.316
THEO. WING PLANFORM AREA               ,SPLAN  =    550.000
X-STA OF ROOT CHORD (CR) T.E.          ,XCRTE  =     38.038
EQUIV. WING ASPECT RATIO               ,AR     =      8.909
EQUIV. WING TAPER  RATIO               ,TAPR   =       .250
EQUIV. WING LEADING EDGE SWEEP         ,SWPLE  =     49.112
EQUIV. WING QUARTER-CHORD SWEEP        ,SWPQC  =     38.221
EQUIV. WING MID-CHORD      SWEEP       ,SWPMC  =     32.304
EQUIV. WING TRAILING-EDGE SWEEP        ,SWPTE  =      1.445
EQUIV. WING MAX.THICKNESS SWEEP        ,SWPT   =     35.291
EQUIV. WING EXPOSED AREA               ,SEXP   =    570.547
EQUIV. WING EXPOSED MAC                ,CBAR   =     11.618
EQUIV. WING THICKNESS RATIO            ,TOCW   =       .092
EQUIV. WING 2-D DESIGN CL              ,CLD    =       .161
EQUIV. WING EXPOSED ASPECT RATIO       ,ARXR   =      7.465
CHORD LENGTH AT WING ROOT              ,CRX    =     25.580
CHORD LENGTH AT WING TIP               ,CTX    =      3.130
X-STA. AT L.E. OF WING ROOT            ,XRX    =      8.009
X-STA. AT L.E. OF WING TIP             ,XTX    =     46.143
Y-STA. AT L.E. OF WING ROOT            ,YRX    =      2.500
Y-STA. AT L.E. OF WING TIP             ,YTX    =     35.000
X-STA. AT L.E. OF WING MAC             ,XB     =     21.890
Y-STA. AT L.E. OF WING MAC             ,YB     =     14.000
DELTA-X  C/4 WING TO C/4 TAIL          ,XH     =     23.992
INCLINATION LINE BET. WING AND TAIL,COEA =     0.000
   ENTER SURVEY

        6A SERIES AIRFOIL SECTION
```

MACH CRITICAL TABLE

CL	MACH CRITICAL
0.0000	.8109
.1000	.7982
.2000	.7859
.3000	.7728
.4000	.7606
.5000	.7472
.6000	.7324
.7000	.7161
.8000	.6974
.9000	.6759
1.0000	.6521

FB-111
MACH NO. = .800 ALTITUDE = 30000. FT. L.E. SWEEP ANGLE = 16.00 DEG. TRIMMED CONDITION

CL	TOTAL CD	CM	ALPHA	CD LIFT	CD RAFT	CL AT CM =0	CD AT CM=0
.016	.02274	.92203	.35	.00005	0.00000	0.00000	.02241
.066	.02287	.93901	.75	.00001	0.00000	.05000	.02226
.116	.02323	.95613	1.14	.00020	.00001	.10000	.02257
.167	.02397	.97244	1.53	.00062	.00016	.15000	.02314
.217	.02513	.99099	1.92	.00126	.00049	.20000	.02412
.267	.02667	1.00824	2.32	.00213	.00098	.25000	.02547
.318	.02860	1.02703	2.71	.00322	.00162	.30000	.02722
.368	.03094	1.04561	3.10	.00454	.00245	.35000	.02936
.419	.03368	1.06465	3.50	.00609	.00344	.40000	.03189
.469	.03692	1.08418	3.89	.00786	.00470	.45000	.03492
.519	.04060	1.10427	4.28	.00985	.00616	.50000	.03836
.569	.04490	1.12496	4.67	.01208	.00801	.55000	.04246
.620	.04968	1.14631	5.07	.01453	.01011	.60000	.04700
.670	.05517	1.16837	5.46	.01720	.01268	.65000	.05425
.720	.06119	1.19119	5.85	.02010	.01555	.70000	.05801
.771	.06825	1.21482	6.25	.02323	.01921	.75000	.06480
.821	.07593	1.23933	6.64	.02658	.02326	.80000	.07221
.872	.08515	1.26476	7.03	.03036	.02835	.85000	.08414
.918	.09599	1.23308	7.68	.03613	.03400	.90000	.09249
.950	.11271	-.02616	9.15	.05013	.04031	.95000	.11481
.982	.12670	-1.06554	10.55	.07205	.04682	1.00000	.14103

CLA = .12728 PER DEG. ALC = .35238 DEG. CLCB = .88036
K = .04518 DELCL = .03298 CLMAX = 1.56026
CMG = .05001* CM/CL = .32210* (*TAIL OFF)

DRAG BREAKDOWN -----		FUSELAGE	NACELLE
FRICTION	= .00528	.00255	.00172
FORM	= .00047	.00041	.00006
INTERF	= .00096	.00066	.00090
WAVE	=0.00000	0.00000	0.00000
BASE	= .00114	0.00000	.00114

		GLOVE	VAR.WING	H.T.	V.T.	STRAKE
FRICTION	= .00814	.00133	.00399	.00158	.00101	.00022
FORM	= .00092	.00012	.00070	.00007	.00004	.00001
INTERF	= .00252	.00041	.00133	.00044	.00028	.00006
WAVE	=0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
CAMBER	= .00019					
DRAG RISE	=0.00000					
MISC	= .00274					

CDMIN = .02226

64 SERIES AIRFOIL SECTION

MACH CRITICAL TABLE

CL	MACH CRITICAL
0.0000	.8160
.1000	.8031
.2000	.7899
.3000	.7776
.4000	.7651
.5000	.7513
.6000	.7359
.7000	.7189
.8000	.6992
.9000	.6764
1.0000	.6513

FB-111
MACH NO. = .800   ALTITUDE = 30000. FT.   L.E. SWEEP ANGLE = 26.60 DEG.   TRIMMED CONDITION

CL	TOTAL CD	CM	ALPHA	CD LIFT	CD R*AFT	CL AT CM =0	CD AT CM=0
.012	.02233	.68982	.30	.00005	0.00000	0.00000	.02211
.059	.02220	.49738	.79	.00002	0.00000	.05000	.02208
.105	.02247	.30504	1.21	.00023	0.00040	.10000	.02229
.152	.02286	.11281	1.62	.00069	.00003	.15000	.02278
.199	.02361	-.07926	2.04	.00129	.00022	.20000	.02367
.245	.02473	-.27115	2.45	.00233	.00056	.25000	.02456
.292	.02615	-.46284	2.87	.00352	.00106	.30000	.02664
.339	.02758	-.65430	3.29	.00496	.00173	.35000	.02875
.385	.03016	-.84550	3.70	.00663	.00257	.40000	.03127
.432	.03281	-1.03641	4.12	.00855	.00368	.45000	.03420
.479	.03586	-1.22701	4.53	.01072	.00499	.50000	.03777
.526	.03950	-1.41726	4.95	.01313	.00670	.55000	.04189
.572	.04359	-1.60715	5.37	.01578	.00865	.60000	.04650
.619	.04837	-1.79663	5.78	.01867	.01111	.65000	.05185
.666	.05366	-1.98569	6.20	.02181	.01387	.70000	.05775
.710	.06025	-2.31907	6.77	.02598	.01744	.75000	.06548
.742	.07377	-3.28457	8.12	.03680	.02141	.80000	.08228
.774	.09547	-4.41472	9.41	.05923	.02652	.85000	.10782
.807	.12486	-5.42525	10.65	.08727	.03216	.90000	.14456
.839	.16216	-6.42808	11.85	.12251	.03852	.95000	.18349
.872	.20665	-7.43296	13.01	.16616	.04488	1.00000	.23410

CLA = .12023 PER DEG.   ALC = .37519 DEG.   CLCB = .74201
K = .04876   DELCL = .03117   CLMAX = 1.53188
CMO = .04108*   DM/CL = .15216*   (*TAIL OFF)

DRAG BREAKDOWN -----		FUSELAGE	NACELLE			
FRICTION	= .00528	.00255	.00173			
FORM	= .00047	.00041	.00006			
INTERF	= .00096	.00006	.00090			
WAVE	=0.00000	0.00000	0.00000			
BASE	= .00114	0.00000	.00114			
		GLOVE	VAR.WING	H.T.	V.T.	STRAKE
FRICTION	= .00799	.00133	.00384	.00158	.00101	.00022
FORM	= .00088	.00012	.00064	.00007	.00004	.00001
INTERF	= .00246	.00041	.00127	.00044	.00028	.00006
WAVE	=0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
CAMBER	= .00018					
DRAG RISE	=0.00000					
MISC	= .00271					

CDMIN = .02208

6A SERIES AIRFOIL SECTION

MACH CRITICAL TABLE

CL	MACH CRITICAL
0.0000	.8220
.1000	.8095
.2000	.7971
.3000	.7847
.4000	.7717
.5000	.7574
.6000	.7409
.7000	.7225
.8000	.7013
.9000	.6763
1.0000	.6490

FB-111

MACH NO. = .800 ALTITUDE = 30000. FT. L.E. SWEEP ANGLE = 35.00 DEG. TRIMMED CONDITION

CL	TOTAL CD	CM	ALPHA	CD LIFT	CD RMAFT	CL AT CM =0	CD AT CM=0
.006	.02124	.22122	.42	.00005	0.00000	0.00000	.02175
.049	.02171	-.05199	.88	.00002	0.00000	.05000	.02173
.093	.02178	-.42439	1.33	.00027	0.00000	.10000	.02198
.136	.02207	-.79706	1.79	.00080	0.00000	.15000	.02250
.180	.02257	-1.16971	2.24	.00160	.00002	.20000	.02332
.223	.02341	-1.54231	2.69	.00268	.00016	.25000	.02455
.267	.02461	-1.91483	3.15	.00404	.00046	.30000	.02620
.311	.02618	-2.28729	3.60	.00568	.00093	.35000	.02821
.354	.02812	-2.65967	4.05	.00759	.00153	.40000	.03066
.398	.03054	-3.03198	4.51	.00978	.00246	.45000	.03394
.441	.03335	-3.40421	4.96	.01224	.00359	.50000	.03749
.485	.03678	-3.77636	5.41	.01498	.00505	.55000	.04174
.528	.04068	-4.14842	5.87	.01800	.00682	.60000	.04652
.572	.04522	-4.52039	6.32	.02130	.00912	.65000	.05213
.613	.05519	-5.63952	7.64	.03173	.01176	.70000	.06519
.634	.07129	-6.70995	8.89	.04861	.01520	.75000	.08551
.667	.09908	-7.73508	10.10	.07190	.01908	.80000	.11468
.700	.12999	-8.72408	11.28	.10156	.02422	.85000	.14756
.733	.16277	-9.68343	12.38	.13767	.02993	.90000	.18531
.767	.20676	-10.61810	13.47	.18016	.03635	.95000	.22821
.801	.25687	-11.53193	14.54	.22905	.04276	1.00000	.29251

CLA = .11021 PER DEG. ALO = .42447 DEG. CLCB = .65065

K = .05539 DELCL = .02590 CLMAX = 1.57321

CMO = .02829* CM/CL = .01672* (*TAIL OFF)

DRAG BREAKDOWN -----

		FUSELAGE	NACELLE
FRICTION	= .00528	.00355	.00173
FORM	= .00047	.00041	.00006
INTERF	= .00096	.00006	.00090
WAVE	=0.00000	0.00000	0.00000
BASE	= .00114	0.00000	.00114

		GLOVE	VAR.WING	H.T.	V.T.	STRAKE
FRICTION	= .00782	.00133	.00367	.00158	.00101	.00022
FORM	= .00050	.00012	.00057	.00027	.00004	.00001
INTERF	= .00229	.00041	.00120	.00044	.00028	.00008
WAVE	=0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
CAMBER	= .00019					
DRAG RISE	=0.00000					
MISC	= .00266					

CDMIN = .02170

6A SERIES AIRFOIL SECTION

MACH CRITICAL TABLE

CL	MACH CRITICAL
0.0000	.8383
.1000	.8266
.2000	.8146
.3000	.8018
.4000	.7873
.5000	.7715
.6000	.7515
.7000	.7284
.8000	.7015
.9000	.6706
1.0000	.6353

FB-111
MACH NO. = .800 ALTITUDE = 30000. FT. L.E. SWEEP ANGLE = 50.00 DEG. TRIMMED CONDITION

CL	TOTAL CD	CM	ALPHA	CD LIFT	CD RMAFT	CL AT DM =0	CD AT CM=0
.000	.02103	.01056	.53	.00006	0.00000	0.00000	.02102
.039	.02082	-.61982	1.09	.00024	0.00000	.05000	.02101
.078	.02091	-1.25024	1.64	.00041	0.00000	.10000	.02138
.118	.02131	-1.88074	2.20	.00117	0.00000	.15000	.02214
.157	.02202	-2.51137	2.76	.00232	0.00000	.20000	.02328
.196	.02303	-3.14218	3.31	.00385	0.00000	.25000	.02482
.235	.02424	-3.77323	3.87	.00577	0.00000	.30000	.02674
.274	.02601	-4.40455	4.42	.00808	.00005	.35000	.02905
.313	.02815	-5.03620	4.98	.01078	.00026	.40000	.03200
.352	.03081	-5.66824	5.53	.01386	.00068	.45000	.03551
.384	.03524	-6.70684	6.62	.01969	.00141	.50000	.04397
.414	.04304	-7.86254	7.87	.02920	.00239	.55000	.05455
.446	.05393	-8.96450	9.05	.04175	.00379	.60000	.06651
.477	.06822	-10.02285	10.18	.05735	.00591	.65000	.08413
.510	.08977	-11.04408	11.26	.07600	.00827	.70000	.10524
.543	.10708	-12.03254	12.31	.09770	.01167	.75000	.13034
.576	.13181	-12.99121	13.33	.12244	.01566	.80000	.15907
.610	.16070	-13.92218	14.32	.15024	.02095	.85000	.19215
.644	.19306	-14.82663	15.29	.18108	.02686	.90000	.22891
.679	.22921	-15.70611	16.24	.21498	.03370	.95000	.26964
.715	.26823	-16.56055	17.16	.25192	.04054	1.00000	.31242

CLA = .09000 PER DEG. ALO = .52360 DEG. CLDB = .46506
K = .07748 DELCL = .02708 CLMAX = 1.60436
CMO = .01930* CM/CL = -.16262* (*TAIL OFF)

DRAG BREAKDOWN -----		FUSELAGE	NACELLE			
FRICTION	= .00578	.00355	.00273			
FORM	= .00047	.00041	.00006			
INTERF	= .00096	.00006	.00090			
WAVE	=0.00000	0.00000	0.00000			
BASE	= .00114	0.00000	.00114			
		GLOVE	VAR.WING	H.T.	V.T.	STRAKE
FRICTION	= .00746	.00133	.00331	.00158	.00101	.00022
FORM	= .00065	.00012	.00042	.00007	.00004	.00001
INTERF	= .00225	.00041	.00106	.00044	.00028	.00006
WAVE	=0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
CAMBER	= .00020					
DRAG RISE	=0.00000					
MISC	= .00256					

CDMIN = .02097

6A SERIES AIRFOIL SECTION

MACH CRITICAL TABLE

CL	MACH CRITICAL
0.0000	.8669
.1000	.8557
.2000	.8438
.3000	.8312
.4000	.8124
.5000	.7888
.6000	.7569
.7000	.7154
.8000	.6588
.9000	.6041
1.0000	.5497

FB-111
MACH NO. = .800 ALTITUDE = 30000. FT. L.E. SWEEP ANGLE = 72.50 DEG. TRIMMED CONDITION

CL	TOTAL CD	CM	ALPHA	CD LIFT	CD R+AFT	CL AT CM =0	CD AT CM=0
-.010	.01924	-.58710	.72	.00008	0.00000	0.00000	.01992
.031	.01969	-1.10695	1.63	.00016	0.00000	.05000	.01999
.073	.02035	-1.62318	2.55	.00120	0.00000	.10000	.02103
.114	.02183	-2.13471	3.45	.00320	0.00000	.15000	.02303
.155	.02430	-2.64043	4.35	.00633	0.00000	.20000	.02616
.197	.02863	-3.13672	5.23	.01265	0.00000	.25000	.03248
.239	.03739	-3.61642	6.11	.02113	0.00000	.30000	.04096
.281	.04702	-4.08361	6.94	.03176	0.00000	.35000	.05159
.324	.05868	-4.49855	7.78	.04454	0.00000	.40000	.06437
.367	.07235	-4.92654	8.62	.05947	0.00000	.45000	.07930
.409	.08812	-5.35617	9.50	.07655	.00042	.50000	.09651
.452	.10623	-5.78992	10.42	.09576	.00073	.55000	.11634
.495	.12697	-6.22729	11.38	.11717	.00183	.60000	.13883
.537	.15058	-6.67403	12.39	.14071	.00402	.65000	.16456
.579	.17553	-7.13107	13.47	.16640	.00736	.70000	.19329
.622	.20755	-7.58714	14.59	.19424	.01257	.75000	.22664
.664	.24121	-8.03056	15.75	.22423	.01922	.80000	.26328
.707	.27652	-8.46571	16.96	.25636	.02568	.85000	.30188
.749	.31361	-8.89916	18.22	.29067	.03214	.90000	.34264
.792	.35241	-9.33267	19.58	.32712	.03858	.95000	.38553
.834	.39278	-9.78492	21.03	.36572	.04501	1.00000	.43056

CLA = .05422 PER DEG. ALC = .71616 DEG. CLCR = .19516
K = .19198 DELCL = .02089 CLMAX = 1.14049
CMO = .00035* CM/CL = .05549* (*TAIL OFF)

DRAG BREAKDOWN -----		FUSELAGE	NACELLE			
FRICTION	= .00528	.00355	.00173			
FORM	= .00047	.00041	.00006			
INTERF	= .00096	.00006	.00090			
WAVE	=0.00000	0.00000	0.00000			
BASE	= .00114	0.00000	.00114			
		GLOVE	VAR.WING	H.T.	V.T.	STRAKE
FRICTION	= .00988	.00133	.00273	.00158	.00101	.00022
FORM	= .00042	.00012	.00019	.00007	.00004	.00001
INTERF	= .00202	.00041	.00083	.00044	.00028	.00006
WAVE	=0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
CAMBER	= .00026					
DRAG RISE	=0.00000					
MISC	= .00240					

CDMIN = .01983

FB-111
MACH NO. = 1.200   ALTITUDE = 30000. FT.   L.E. SWEEP ANGLE = 72.50 DEG.   TRIMMED CONDITION

CL	TOTAL CD	CM	ALPHA	CD LIFT	CD R*AFT	CL AT CM =0	CD AT CM=0
-.004	.03878	-.20314	.21	.00003	0.00000	0.00000	.03875
.037	.03905	-.69275	1.07	.00043	0.00000	.05000	.03919
.078	.04061	-1.18437	1.93	.00224	0.00000	.10000	.04099
.119	.04346	-1.67502	2.79	.00546	0.00000	.15000	.04422
.160	.04758	-2.16572	3.65	.01009	0.00000	.20000	.04885
.200	.05259	-2.65648	4.51	.01614	0.00000	.25000	.05489
.241	.05969	-3.14732	5.37	.02360	0.00000	.30000	.06235
.282	.06767	-3.63826	6.24	.03247	0.00000	.35000	.07122
.323	.07693	-4.12930	7.10	.04275	0.00000	.40000	.08150
.364	.08748	-4.62047	7.96	.05444	0.00000	.45000	.09320
.404	.09931	-5.11178	8.82	.06755	0.00000	.50000	.10630
.445	.11472	-5.60325	9.68	.08437	0.00000	.55000	.12312
.486	.13185	-6.09489	10.54	.10303	0.00000	.60000	.14176
.527	.15050	-6.58673	11.41	.12334	0.00000	.65000	.16209
.568	.17068	-7.07877	12.27	.14531	0.00000	.70000	.18406
.609	.19238	-7.57105	13.13	.16893	0.00000	.75000	.20769
.649	.21561	-8.06356	13.99	.19421	0.00000	.80000	.23296
.690	.24037	-8.55635	14.85	.22114	0.00000	.85000	.25989
.731	.26665	-9.04941	15.71	.24972	0.00000	.90000	.28848
.772	.29446	-9.54278	16.58	.27996	0.00000	.95000	.31872
.812	.32379	-10.03648	17.44	.31186	0.00000	1.00000	.35061

CLA = .05803 PER DEG.   ALO = .20531 DEG.   CLDB = .50400
K = .28244   DELCL = .01095   CLMAX = 1.45677
CMO = 0.00000*   CM/CL = -.02589*   (*TAIL OFF)

DRAG BREAKDOWN -----		FUSELAGE	NACELLE
FRICTION	= .00487	.00314	.00153
FORM	=0.00000	0.00000	0.00000
INTERF	=0.00000	0.00000	0.00000
WAVE	= .01725	.00705	.01019
BASE	= .00210	0.00000	.00210

		GLOVE	VAR.WING	H.T.	V.T.	STRAKE
FRICTION	= .00611	.00119	.00243	.00140	.00090	.00020
FORM	=0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
INTERF	=0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
WAVE	= .00657	.00542	0.00000	.00098	.00052	.00005
CAMBER	= .00004					
DRAG RISE	=0.00000					
MISC	= .00162					

CDMIN = .03875

NOTE. WAVE DRAG FOR VAR.WING IS INCLUDED IN WAVE DRAG OF GLOVE

FB-111
MACH NO. = 1.600 ALTITUDE = 30000. FT. L.E. SWEEP ANGLE = 72.50 DEG. TRIMMED CONDITION

CL	TOTAL CD	CM	ALPHA	CD LIFT	CD RWAFT	CL AT CM =0	CD AT CM=0
-.003	.03423	-.19770	.20	.00000	0.00000	0.00000	.03434
.037	.03450	-.76340	1.08	.00068	0.00000	.05000	.03502
.078	.03685	-1.32916	1.96	.00287	0.00000	.10000	.03721
.119	.04020	-1.89505	2.85	.00656	0.00000	.15000	.04092
.160	.04494	-2.46111	3.73	.01180	0.00000	.20000	.04614
.200	.05107	-3.02741	4.61	.01853	0.00000	.25000	.05287
.241	.05868	-3.59400	5.50	.02685	0.00000	.30000	.06119
.282	.06877	-4.16095	6.38	.03779	0.00000	.35000	.07413
.322	.08044	-4.72832	7.26	.05042	0.00000	.40000	.08476
.363	.09368	-5.29616	8.15	.06474	0.00000	.45000	.09906
.404	.10849	-5.86453	9.03	.08077	0.00000	.50000	.11511
.444	.12488	-6.43351	9.91	.09848	0.00000	.55000	.13282
.485	.14284	-7.00314	10.80	.11789	0.00000	.60000	.15223
.525	.16238	-7.57351	11.68	.13900	0.00000	.65000	.17334
.566	.18349	-8.14467	12.56	.16181	0.00000	.70000	.19615
.607	.20618	-8.71669	13.44	.18630	0.00000	.75000	.22064
.647	.23044	-9.28965	14.33	.21250	0.00000	.80000	.24684
.688	.25628	-9.86362	15.21	.24039	0.00000	.85000	.27473
.728	.28370	-10.43865	16.09	.26997	0.00000	.90000	.26421
.769	.31268	-11.01486	16.98	.30126	0.00000	.95000	.33560
.809	.34225	-11.59230	17.86	.33423	0.00000	1.00000	.36657

CLA = .05662 PER DEG. ALC = .19715 DEG. CLID = .29630
K = .30261 DELCL = .00256 CLMAX = 1.42361
CMO = 0.00000* CM/CL = -.08058* (*TAIL OFF)

DRAG BREAKDOWN -----		FUSELAGE	NACELLE			
FRICTION = .00416		.00281	.00135			
FORM =0.00000		0.00000	0.00000			
INTERF =0.00000		0.00000	0.00000			
WAVE = .01351		.00667	.00744			
BASE = .00210		0.00000	.00210			
		GLOVE	VAR.WING	H.T.	V.T.	STRAKE
FRICTION = .00530		.00103	.00211	.00121	.00078	.00017
FORM =0.00000		0.00000	0.00000	0.00000	0.00000	0.00000
INTERF =0.00000		0.00000	0.00000	0.00000	0.00000	0.00000
WAVE = .00785		.00618	0.00000	.00100	.00062	.00005
CAMBER = .00000						
DRAG RISE=0.00000						
MISC = .00142						

CDMIN = .03434

NOTE. WAVE DRAG FOR VAR.WING IS INCLUDED IN WAVE DRAG OF GLOVE

FB-111
MACH NO. = 2.000 ALTITUDE = 30000. FT. L.E. SWEEP ANGLE = 72.50 DEG. TRIMMED CONDITION

CL	TOTAL CD	CM	ALPHA	CD LIFT	CD R+AFT	CL AT CM =0	CD AT CM=0
-.003	.03078	-.23769	.20	0.00000	0.00000	0.00000	.03079
.037	.03145	-.92633	1.15	.00092	0.00000	.05000	.03161
.077	.03361	-1.61486	2.09	.00328	0.00000	.10000	.03407
.118	.03726	-2.30356	3.04	.00738	0.00000	.15000	.03817
.158	.04239	-2.99250	3.99	.01313	0.00000	.20000	.04391
.199	.04901	-3.68178	4.94	.02051	0.00000	.25000	.05130
.239	.05711	-4.37147	5.89	.02954	0.00000	.30000	.06033
.279	.06775	-5.06166	6.84	.04125	0.00000	.35000	.07204
.320	.08169	-5.75242	7.79	.05642	0.00000	.40000	.08721
.360	.09854	-6.44386	8.74	.07505	0.00000	.45000	.10584
.401	.11949	-7.13604	9.69	.09715	0.00000	.50000	.12794
.441	.14335	-7.82507	10.63	.12271	0.00000	.55000	.15350
.481	.17052	-8.52303	11.58	.15173	0.00000	.60000	.18252
.522	.20100	-9.21201	12.53	.18421	0.00000	.65000	.21500
.562	.23479	-9.91411	13.48	.22016	0.00000	.70000	.25095
.602	.27188	-10.61144	14.43	.25957	0.00000	.75000	.29036
.642	.31229	-11.31008	15.38	.30244	0.00000	.80000	.33323
.683	.35800	-12.01015	16.33	.34877	0.00000	.85000	.37956
.723	.40302	-12.71175	17.28	.39856	0.00000	.90000	.42935
.763	.45335	-13.41459	18.22	.45182	0.00000	.95000	.48261
.803	.50659	-14.12000	19.17	.50454	0.00000	1.00000	.53533

CLA = .05279 PER DEG. ALO = .19717 DEG. CLDS = 1.00000
K = .32813 DELCL = 0.00000 CLMAX = 1.22014
CMO = 0.00000* CM/CL = -.68330* (*TAIL OFF)

```
DRAG BREAKDOWN -----     FUSELAGE      NACELLE
  FRICTION = .00370        .00250        .00120
  FORM     =0.00000       0.00000       0.00000
  INTERF   =0.00000       0.00000       0.00000
  WAVE     = .01156        .00554        .00632
  BASE     = .00188       0.00000        .00188
                         GLOVE         VAR.WING      H.T.        V.T.        STRAKE
  FRICTION = .00472        .00052        .00188      .00108      .00069      .00015
  FORM     =0.00000       0.00000       0.00000     0.00000     0.00000     0.00000
  INTERF   =0.00000       0.00000       0.00000     0.00000     0.00000     0.00000
  WAVE     = .00737        .00568       0.00000      .00112      .00053      .00024
  CAMBER   =0.00000
  DRAG RISE=0.00000
  MISC     = .00126

    CDMIN = .03079
```

NOTE. WAVE DRAG FOR VAR.WING IS INCLUDED IN WAVE DRAG OF GLOVE

```
C-141A FLIGHT TEST ANALYSIS     NASA CR - 1558                                  CARD 1.1
                                           1                                    CARD 1.2
    4     2     1     3     0     1     1     6     0     1                     CARD 1.3
3228.     22.2        61.07      0.0        7.0        1.0                      CARD 1.4
0.0005    5.0         -5.58      0.0                                            CARD 1.5
3228.0    0.373       24.38                                                     CARD 1.6
 FUSELAGE WHEEL WELL  BULLET     NACELLE                                        CARD 2.1
132.29    33.67       24.95      16.60      LENGTH           B(I,1)             CARD 2.2
14.17     5.85        3.65       5.50       WIDTH                2              CARD 2.3
14.17     5.85        3.65       5.50       HEIGHT               3              CARD 2.4
4347.52   822.0       136.8      1045.96    AWET/TOTAL           4              CARD 2.5
1.0       1.2         1.3        1.3        INT. FACTOR          5              CARD 2.6
1.0       2.0         1.0        4.0        NO. BODIES           6              CARD 2.7
157.63    26.83       10.47      23.76      AMAX/ECCY            7              CARD 2.8
0.0       0.0         0.0        0.0        ABASE (STREAMTUBE)   8              CARD 2.9
17.00     16.83       12.50      8.25       LNOSE                9              CARD 2.10
50.83     16.83       12.08      8.33       LTAIL                10             CARD 2.11
0.0       0.0         0.0        0.0        BASE DRAG AREA       11             CARD 2.12
                                 12.5       INLET AREA           12             CARD 2.13
 WING P1   WING P2    H.TAIL     V.TAIL      OUTBD PYL INBD PYL                 CARD 3.1
-63       -63         64A        64A        64A        64A        SECTION       CARD 3.2
0.4       0.4         0.0        0.0        0.0        0.0        CLI           CARD 3.3
0.1155    0.11        0.105      0.13       0.10       0.10       THICK         CARD 3.4
23.73     25.03       29.0       40.0       73.0       73.0       SWPLE         CARD 3.5
3134.2    2932.2      893.74     819.4      223.24     214.88     AWET/TOT      CARD 3.6
33.2      20.06       14.3       23.0       14.E       14.6       CROOT         CARD 3.7
20.06     10.99       5.0        13.6       14.6       14.6       CTIP          CARD 3.8
27.1      46.4        25.17      22.72      3.24       3.17       SPAN)EXP      CARD 3.9
37.75     51.1        123.7      110.2      45.0       37.1       X             CARD 3.10
6.5       33.6        0.0        8.0        28.33      23.75      Y             CARD 3.11
0.0       0.0         23.5       2.24       -4.48      -4.48      Z             CARD 3.12
4.89      0.0         0.0                                         INCID.        CARD 3.13
     2    21                                                                    CARD 6.1
0.8       30000.      0.2                   0.0        1.0                      CARD 6.2
1.2       30000.      0.2                   0.0        1.0                      CARD 6.2
NO ADJUSTMENTS                                                                  CARD 7.1
DAMAGE CASES FOLLOW                                                             CARD 9.1
C-141A WITH NUCLEAR DAMAGE                                                      CARD 9.2
1111111   1111111                                                               CARD 9.3
FUSELAGE  WHEEL WELL BULLET   NACELLE                                           CARD 10.1
0.0002                                        DBOD(I, 1) ROUGHNESS              CARD 10.2
0.0005                                        DBOD(I, 2)                        CARD 10.3
0.1                                           DBOD(I, 3)                        CARD 10.4
0.5                                           DBOD(I, 4)                        CARD 10.5
472.0                                         DBOD(I, 5)                        CARD 10.6
                      4.0                     DBOD(I,11) FWD STEPS              CARD 10.7
                      2.0                     DBOD(I,12)                        CARD 10.8
                      0.1                     DBOD(I,13)                        CARD 10.9
                      0.4                     DBOD(I,14)                        CARD 10.10
                      0.9                     DBOD(I,15)                        CARD 10.11
          2.0                                 DBOD(I,21) AFT STEPS              CARD 10.12
          8.0                                 DBOD(I,22)                        CARD 10.13
          0.3                                 DBOD(I,23)                        CARD 10.14
          0.3                                 DBOD(I,24)                        CARD 10.15
          0.8                                 DBOD(I,25)                        CARD 10.16
3.0                              1.0          DBOD(I,31) HOLES                  CARD 10.17
2.0                              0.0          DBOD(I,32)                        CARD 10.18
4.0                              4.0          DBOD(I,33)                        CARD 10.19
2.0                              2.0          DBOD(I,34)                        CARD 10.20
6.5                              1.0          DBOD(I,35)                        CARD 10.21
```

```
0.2                           0.5        DBOD(I,36)                     CARD 10.22
0.7                           0.5        DBOD(I,37)                     CARD 10.23
1.0                           2.0        DBOD(I,38)                     CARD 10.24
4.0                           1.0        DBOD(I,41) WAVES               CARD 10.25
2.0                           0.0        DBOD(I,42)                     CARD 10.26
8.0                           6.0        DBOD(I,43)                     CARD 10.27
4.0                           5.0        DBOD(I,44)                     CARD 10.28
0.5                           0.4        DBOD(I,45)                     CARD 10.29
0.3                           0.4        DBOD(I,46)                     CARD 10.30
0.9                           0.4        DBOD(I,47)                     CARD 10.31
4.0                 2.0                  DBOD(I,51) PROTUBERANCE        CARD 10.32
0.3                 0.9                  DBOD(I,52)                     CARD 10.33
0.5                 0.75                 DBOD(I,53)                     CARD 10.34
0.4                 0.40                 DBOD(I,54)                     CARD 10.35
0.8                 0.9                  DBOD(I,55)                     CARD 10.36
12.57               8.0                  DBOD(I,61) ELLPT NOSE          CARD 10.37
0.1                 0.4                  DBOD(I,62)                     CARD 10.38
WING P1   WING P2   H. TAIL   .V. TAIL   CLTED FYL INED FYL    DSLR     CARD 11.1
0.0002                                                        (I, 1)    CARD 11.2
0.0005                                                            2)    CARD 11.3
0.1                                                               3)    CARD 11.4
0.5                                                               4)    CARD 11.5
125.                                                              5)    CARD 11.6
1.0                                                               6)    CARD 11.7
                    0.0                            2.0           11)    CARD 11.8
                    1.0                            0.0           12)    CARD 11.9
                    7.0                            2.0           13)    CARD 11.10
                    0.3                            1.0           14)    CARD 11.11
                    0.4                            0.2           15)    CARD 11.12
                    0.7                            0.8           16)    CARD 11.13
                    0.0                            2.0           21)    CARD 11.14
                    1.0                            0.0           22)    CARD 11.15
                    7.0                            2.0           23)    CARD 11.16
                    0.3                            1.0           24)    CARD 11.17
                    0.4                            0.2           25)    CARD 11.18
                    0.7                            0.8           26)    CARD 11.19
2.0                                      1.0                  (I,31)    CARD 11.20
3.0                                      0.0                  (I,32)    CARD 11.21
2.0                                      4.0                  (I,33)    CARD 11.22
2.0                                      2.0                  (I,34)    CARD 11.23
0.5                                      0.28                 (I,35)    CARD 11.24
0.2                                      0.75                 (I,36)    CARD 11.25
0.8                                      0.75                 (I,37)    CARD 11.26
1.0                                      2.0                  (I,38)    CARD 11.27
0.9                                      0.7                  (I,39)    CARD 11.28
2.0                                      2.0                  (I,41)    CARD 11.29
2.0                                      2.0                  (I,42)    CARD 11.30
3.0                                      4.0                  (I,43)    CARD 11.31
3.0                                      3.0                  (I,44)    CARD 11.32
0.4                                      0.1                  (I,45)    CARD 11.33
0.2                                      0.2                  (I,46)    CARD 11.34
0.6                                      0.6                  (I,47)    CARD 11.35
7.0                                                2.0        (I,51)    CARD 11.36
1.0                                                0.3        (I,52)    CARD 11.37
0.3                                                1.0        (I,53)    CARD 11.38
0.2                                                0.7        (I,54)    CARD 11.39
0.8                                                0.9        (I,55)    CARD 11.40
0.2       0.25      0.3       0.8        1.0       3.0                  CARD 11.41
0.21      0.22      0.23      3.0                                       CARD 11.42
0.15      0.0       0.80                                                CARD 11.43
0.05      0.05      0.2       0.3        0.4                            CARD 11.44
END                                                                     CARD 12.1
```

```
EMPIRICALLY BASED COMPUTER PROGRAM
TO PREDICT THE AERODYNAMIC CHARACTERISTICS
OF AIRCRAFT
INCLUDING
ROUGHNESS, PROTUBERANCES, AND
DAMAGE CALCULATIONS

NPRINT(28) =  1        C-141A FLIGHT TEST ANALYSIS     NASA CR - 1558
```

PROBLEM INPUT PARAMETERS

NFCCS =	4	SREF =	3228.00	ROUGHX=	.00050
NPNLS =	2	CMAC =	22.200	FMISC =	5.00
NHT =	1	XMAC =	61.070	THIST =	-5.580
NVT =	3	?CC =	0.000	CONCL =	0.040
ISWP =	0	UFFUS =	7.000		
IREF =	1	ACB =	1.000		
IWNG =	1				
NAFC =	0				
METER =	0				
ITRIM =	1				

DIMENSIONS ARE IN FEET

REFERENCE WING FOR TWO PANEL CASE

SPLAN = 3228.000 TAPR = .3730 SWP = 24.380

		FUSELAGE	WHEEL WELL	BULLET	NACELLE
LENGTH	=	132.290	33.670	24.990	16.600
WIDTH	=	14.170	5.850	3.650	5.500
HEIGTH	=	14.170	5.850	3.650	5.500
WETTED AREA	=	4347.520	822.000	136.800	1045.960
INTER.FACTOR	=	1.000	1.200	1.300	1.200
NO.OF THIS TYPE	=	1.000	2.000	1.000	4.000
MAXIMUM AREA	=	157.630	26.830	10.470	23.760
BASE AREA	=	0.000	0.000	0.000	0.000
NOSE LENGTH	=	17.000	16.820	12.500	8.250
BOATTAIL LENGTH	=	50.820	16.820	12.080	8.330
BASE DRAG AREA	=	0.000	0.000	0.000	0.000
INLET AREA	=	0.000	0.000	0.000	12.500

		WING P1	WING P2	H.TAIL	V.TAIL	OUTBD PYL	INBD PYL
SECTION	=	-63	-63	64A	64A	64A	64A
CAMBER (2-D)	=	.400	.400	0.000	0.000	0.000	0.000
THICKNESS RATIO	=	.120	.110	.105	.130	.100	.100
L.E. SWEEP	=	23.730	25.020	29.000	40.000	73.000	73.000
WETTED AREA	=	3134.200	2932.200	893.740	819.400	222.240	214.880
ROOT CHORD	=	33.200	20.060	14.300	23.000	14.600	14.600
TIP CHORD	=	20.060	10.550	5.000	13.600	14.600	14.600
SPAN OF PANEL	=	27.100	46.400	25.170	22.720	3.280	3.170
X- LE ROOT CHORD	=	37.750	51.100	123.700	110.200	45.060	37.100
Y- LE ROOT CHORD	=	6.500	33.600	0.000	0.000	38.330	23.750
Z- LE ROOT CHORD	=	0.000	0.000	23.500	2.240	-4.480	-4.480
INCIDENCE	=	4.890	0.000	0.000	0.000	0.000	0.000

2 DRAG POLARS TO BE GENERATED AT THE FOLLOWING CONDITIONS

MACH NO.	ALTITUDE	C.G.	SWEEP	FROM CL	TO CL
.80	30000.000	.20	24.4	0.00	1.00
1.20	30000.000	.20	24.4	0.00	1.00

NO ADJUSTMENTS *****

CALCULATED BODY PARAMETERS

		FUSELAGE	WHEEL WELL	BULLET	NACELLE
FINENESS RATIO	=	9.336	5.756	6.847	3.018
WETTED AREA	=	4347.520	822.000	136.860	1045.360
MAXIMUM AREA	=	157.630	26.830	10.470	23.760

CALCULATED SURFACE PARAMETERS

		WING P1	WING P2	H.TAIL	V.TAIL	OUTBD PYL	INBD PYL
WETTED AREA	=	3134.200	2932.200	893.740	819.400	222.240	214.880
TAPER RATIO	=	.664	.548	.350	.591	1.000	1.000
PLANFORM AREA	=	1443.346	1440.720	485.781	415.776	55.776	52.564
ASPECT RATIO	=	2.035	5.977	5.217	2.483	.449	.434
CHARACT. LENGTH	=	27.172	15.967	10.357	18.703	14.601	14.601
SWP C/4	=	17.668	22.689	24.794	26.341	73.000	73.000
SWP C/2	=	11.153	20.265	20.283	22.302	73.000	73.000
SWP TE	=	-2.592	15.188	10.471	23.043	73.000	73.000
SWP MAX T/C	=	16.390	22.210	22.304	24.127	73.000	73.000
X/C AT MAX T/C	=	.200	.300	.350	.390	.390	.390

TOTAL AIRPLANE PARAMETERS

```
WING SEMI-SPAN                           ,BC2   =     80.000
BODY DIA/ WING SPAN                      ,DCB   =       .081
WING ROOT CHORD AT CENTERLINE            ,CR    =     36.252
THEO. WING PLANFORM AREA                 ,SPLAN =   3228.000
X-STA OF ROOT CHORD (CR) T.E.            ,XCRTE =     72.857
EQUIV. WING ASPECT RATIO                 ,AR    =      7.931
EQUIV. WING TAPER  RATIO                 ,TAPR  =       .373
EQUIV. WING LEADING EDGE SWEEP           ,SWPLE =     24.283
EQUIV. WING QUARTER-CHORD SWEEP          ,SWPQC =     20.322
EQUIV. WING MID-CHORD      SWEEP         ,SWPMC =     16.236
EQUIV. WING TRAILING-EDGE SWEEP          ,SWPTE =      6.444
EQUIV. WING MAX.THICKNESS SWEEP          ,SWPT  =     19.508
EQUIV. WING EXPOSED AREA                 ,SEXW  =   2884.066
EQUIV. WING EXPOSED MAC                  ,CBAR  =     21.575
EQUIV. WING THICKNESS RATIO              ,TOCW  =       .115
EQUIV. WING 2-D DESIGN CL                ,CLD   =       .400
EQUIV. WING EXPOSED ASPECT RATIO         ,ARXP  =      7.493
CHORD LENGTH AT WING ROOT                ,CRX   =     33.200
CHORD LENGTH AT WING TIP                 ,CTX   =     10.990
X-STA. AT L.E. OF WING ROOT              ,XRX   =     39.452
X-STA. AT L.E. OF WING TIP               ,XTX   =     72.766
Y-STA. AT L.E. OF WING ROOT              ,YRX   =      6.500
Y-STA. AT L.E. OF WING TIP               ,YTX   =     80.000
X-STA. AT L.E. OF WING MAC               ,XB    =     51.876
Y-STA. AT L.E. OF WING MAC               ,YB    =     33.911
DELTA-X  C/4 WING TO C/4 TAIL            ,XH    =     74.869
INCLINATION LINE BET. WING AND TAIL,OMEGA =     18.293
   ENTER SURVEY

      4-DIGIT AIRFOIL SECTION
```

MACH CRITICAL TABLE

CL	MACH CRITICAL
0.0000	.8294
.1000	.8169
.2000	.8045
.3000	.7900
.4000	.7749
.5000	.7587
.6000	.7410
.7000	.7217
.8000	.7008
.9000	.6782
1.0000	.6555

FUSLAGE AFT-END UPSWEEP DRAG

WING ANGLE	DELCD
0.00000	.00189
1.00000	.00155
2.00000	.00126
3.00000	.00102
4.00000	.00081
5.00000	.00063
6.00000	.00048
7.00000	.00035
8.00000	.00024
9.00000	.00016
10.00000	.00010
11.00000	.00007
12.00000	.00006
13.00000	.00006
14.00000	.00007
15.00000	.00007
16.00000	.00007
17.00000	-.00001
18.00000	-.00007
19.00000	-.00017
20.00000	-.00029

C-141A FLIGHT TEST ANALYSIS   NASA CR - 1558
MACH NO. = .800   ALTITUDE = 30000. FT.   SWEEP ANGLE = 24.38 DEG.   TAIL DEFL. (DH) = 0.00 DEG.

CL	TOTAL CD	CM	ALPHA	CDL	CDR	CDAFT	CD(L+R+AFT)
0.000	.01829	.08021	-4.39	.00076	0.00000	.00171	.00249
.050	.01762	.07494	-3.68	.00024	0.00000	.00148	.00173
.100	.01718	.06884	-2.96	.00001	0.00000	.00128	.00129
.150	.01708	.06253	-2.24	.00008	0.00000	.00110	.00118
.200	.01729	.05602	-1.52	.00046	0.00000	.00094	.00140
.250	.01765	.04932	-.80	.00115	.00002	.00079	.00196
.300	.01858	.04244	-.09	.00214	.00028	.00066	.00308
.350	.02075	.03540	.63	.00243	.00088	.00055	.00486
.400	.02317	.02820	1.35	.00503	.00180	.00045	.00728
.450	.02633	.02086	2.07	.00693	.00315	.00036	.01044
.500	.03018	.01340	2.79	.00914	.00487	.00028	.01429
.550	.03495	.00582	3.51	.01166	.00719	.00021	.01905
.600	.04048	-.00187	4.22	.01448	.00996	.00015	.02459
.650	.04707	-.00964	4.94	.01760	.01347	.00011	.03118
.700	.05456	-.01749	5.66	.02107	.01751	.00008	.03867
.750	.06407	-.02541	6.38	.02559	.02256	.00007	.04818
.800	.07982	-.03371	7.10	.03562	.02825	.00006	.06393
.850	.10468	-.04213	7.82	.05374	.03499	.00006	.08679
.900	.13793	-.05285	8.53	.07954	.04243	.00007	.12204
.950	.17919	-.06602	9.25	.11301	.05022	.00007	.16329
1.000	.22811	-.07961	9.97	.15415	.05800	.00007	.21222

CLA = .06962 PER DEG.   ALC =-4.39481 DEG.   CLCB = .75896
K = .06102   CDLCL = .11292   CLMAX = 1.41022
CMO = -.05879*   CM/CL = .46019*   (*TAIL OFF)

DRAG BREAKDOWN -----		FUSELAGE	WHEEL WELL	FILLET	NACELLE
FRICTION	= .00371	.00223	.00053	.00009	.00075
FORM	= .00051	.00023	.00018	.00002	.00009
INTERF	= .00046	.00004	.00014	.00003	.00025
WAVE	=0.00000	0.00000	0.00000	0.00000	0.00000
BASE	=0.00000	0.00000	0.00000	0.00000	0.00000

		WING F1	WING F2	H.TAIL	V.TAIL	OUTBD PYL	INBD PYL
FRICTION	= .00579	.00209	.00211	.00069	.00058	.00016	.00016
FORM	= .00128	.00051	.00047	.00012	.00013	.00003	.00003
INTERF	= .00197	.00074	.00073	.00021	.00019	.00005	.00005
WAVE	=0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
CAMBER.	= .00150						
DRAG RISE	=0.00000						
MISC	= .00069						

CDMIN = .01589

C-141A FLIGHT TEST ANALYSIS   NASA CR - 1558
MACH NO. = 1.200   ALTITUDE = 30000. FT.   SWEEP ANGLE = 24.38 DEG.   TAIL DEFL. (CH) = 0.00 DEG.

CL	TOTAL CD	CM	ALPHA	CDL	CDP	CDAFT	CD(L+P+AFT)
0.000	.15837	.06868	-4.01	0.00000	0.00000	.00159	.00159
.050	.15861	.06710	-3.56	.00036	0.00000	.00145	.00182
.100	.15961	.06546	-3.11	.00150	0.00000	.00132	.00282
.150	.16127	.06275	-2.67	.00338	0.00000	.00120	.00458
.200	.16359	.06158	-2.22	.00601	0.00000	.00109	.00710
.250	.16717	.06015	-1.78	.00939	0.00000	.00099	.01038
.300	.17121	.05827	-1.33	.01352	0.00000	.00090	.01442
.350	.17600	.05633	-.89	.01840	0.00000	.00081	.01921
.400	.18155	.05434	-.44	.02404	0.00000	.00072	.02476
.450	.18786	.05231	.00	.03042	0.00000	.00065	.03107
.500	.19492	.05022	.45	.03756	0.00000	.00057	.03813
.550	.20274	.04810	.89	.04545	0.00000	.00051	.04595
.600	.21132	.04593	1.34	.05408	0.00000	.00045	.05453
.650	.22065	.04272	1.79	.06347	0.00000	.00039	.06386
.700	.23078	.04147	2.23	.07365	0.00000	.00034	.07399
.750	.24231	.03919	2.68	.08524	0.00000	.00029	.08553
.800	.25547	.03688	3.12	.09844	0.00000	.00024	.09868
.850	.27024	.03454	3.57	.11325	0.00000	.00020	.11345
.900	.28663	.03217	4.01	.12967	0.00000	.00017	.12984
.950	.30463	.02978	4.46	.14771	0.00000	.00014	.14785
1.000	.32426	.02736	4.90	.16736	0.00000	.00011	.16747

CLA = .11226 PER DEG.   ALC =-4.00517 DEG.   CLDP = 1.00000
K = .15023   DELCL = 0.00000   CLMAX = 1.44924
CMO = 0.00000*   CM/CL = .22818*   (*TAIL OFF)

DRAG BREAKDOWN -----	FUSELAGE	WHEEL WELL	BULLET	NACELLE		
FRICTION = .00333	.00210	.00048	.00008	.00067		
FORM =0.00000	0.00000	0.00000	0.00000	0.00010		
INTERF =0.00000	0.00000	0.00030	0.00000	0.00000		
WAVE = .04118	.02270	.00289	.00056	.01402		
BASE =0.00000	0.00000	0.00000	0.00000	0.00000		
	WING F1	WING F2	H.TAIL	V.TAIL	OUTBD PYL	INBD PYL
FRICTION = .00519	.00187	.00189	.00062	.00052	.00015	.00014
FORM =0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
INTERF =0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
WAVE = .10667	.09902	0.00000	.00568	.00694	.00002	.00002
CAMBER =0.00000						
DRAG RISE=0.00000						
MISC = .00043						

CDMIN = .15679

NOTE. WAVE DRAG FOR WING F2 IS INCLUDED IN WAVE DRAG OF WING F1

C-141A WITH NUCLEAR DAMAGE

DAMAGE PARAMETERS

IDAM(1) = 1	IDAM(11) = 1
IDAM(2) = 1	IDAM(12) = 1
IDAM(3) = 1	IDAM(13) = 1
IDAM(4) = 1	IDAM(14) = 1
IDAM(5) = 1	IDAM(15) = 1
IDAM(6) = 1	IDAM(16) = 1
IDAM(7) = 1	IDAM(17) = 1

BODY DAMAGE PARAMETERS, DBOD(I,J)

	I = 1	I = 2	I = 3	I = 4	I = 5	I = 6	I = 7
MODE 1		BODY, ROUGHNESS					
J = 1	.000200	0.000000	0.000000	0.000000			
J = 2	.000500	0.000000	0.000000	0.000000			
J = 3	.100000	0.000000	0.000000	0.000000			
J = 4	.500000	0.000000	0.000000	0.000000			
J = 5	472.000000	0.000000	0.000000	0.000000			
MODE 2		BODY,FWD-FACING STEP					
J = 11	0.000000	0.000000	4.000000	0.000000			
J = 12	0.000000	0.000000	2.000000	0.000000			
J = 13	0.000000	0.000000	.100000	0.000000			
J = 14	0.000000	0.000000	.400000	0.000000			
J = 15	0.000000	0.000000	.500000	0.000000			
MODE 3		BODY,AFT-FACING STEP					
J = 21	0.000000	2.000000	0.000000	0.000000			
J = 22	0.000000	8.000000	0.000000	0.000000			
J = 23	0.000000	.300000	0.000000	0.000000			
J = 24	0.800000	.300000	0.000000	0.000000			
J = 25	0.000000	.600000	0.000000	0.000000			
MODE 4		BODY, HOLES					
J = 31	3.000000	0.000000	0.000000	1.000000			
J = 32	2.000000	0.000000	0.000000	0.000000			
J = 33	4.000000	0.000000	0.000000	4.000000			
J = 34	2.000000	0.000000	0.000000	2.000000			
J = 35	.500000	0.000000	0.000000	1.000000			
J = 36	.200000	0.000000	0.000000	.500000			
J = 37	.700000	0.000000	0.000000	.500000			
J = 38	1.000000	0.000000	0.000000	2.000000			
MODE 5		BODY, WAVINESS					
J = 41	4.000000	0.000000	0.000000	1.000000			
J = 42	2.000000	0.000000	0.000000	0.000000			
J = 43	8.000000	0.000000	0.000000	6.000000			
J = 44	4.000000	0.000000	0.000000	5.000000			
J = 45	.500000	0.000000	0.000000	.400000			
J = 46	.300000	0.000000	0.000000	.400000			
J = 47	.530000	0.000000	0.000000	.400000			

	MODE 6		ECCY, PROTUBERANCES		
J = 51	4.000000	0.000000	2.000000	0.000000
J = 52	.350000	0.800000	.900000	0.800000
J = 53	.500000	0.800000	.750000	0.000000
J = 54	.400000	0.000000	.400000	0.000000
J = 55	.800000	0.000000	.900000	0.000000
	MODE 7		ECCY, BLUNTNESS		
J = 61	12.570000	0.000000	8.000000	0.000000
J = 62	.100000	0.000000	.400000	0.000000

C-141A WITH NUCLEAR DAMAGE

SURFACE DAMAGE PARAMETERS, DSUR(I,J)

	I = 1	I = 2	I = 3	I = 4	I = 5	I = 6	I = 7
MODE 11		SURF, ROUGHNESS					
J = 1	.000200	0.000000	0.000000	0.000000	0.000000	0.000000	
J = 2	.000500	0.000000	0.000000	0.000000	0.000000	0.000000	
J = 3	.100000	0.000000	0.000000	0.000000	0.000000	0.000000	
J = 4	.500000	0.000000	0.000000	0.000000	0.000000	0.000000	
J = 5	125.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
J = 6	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
MODE 12		SURF, FWD-FACING STEP					
J = 11	0.000000	0.000000	0.000000	0.000000	0.000000	2.000000	
J = 12	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	
J = 13	0.000000	0.000000	7.000000	0.000000	0.000000	2.000000	
J = 14	0.000000	0.000000	.300000	0.000000	0.000000	1.000000	
J = 15	0.000000	0.000000	.400000	0.000000	0.000000	.200000	
J = 16	0.000000	0.000000	.700000	0.000000	0.000000	.800000	
MODE 13		SURF, AFT-FACING STEP					
J = 21	0.000000	0.000000	0.000000	0.000000	0.000000	2.000000	
J = 22	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	
J = 23	0.000000	0.000000	7.000000	0.000000	0.000000	2.000000	
J = 24	0.000000	0.000000	.300000	0.000000	0.000000	1.000000	
J = 25	0.000000	0.000000	.400000	0.000000	0.000000	.200000	
J = 26	0.000000	0.000000	.700000	0.000000	0.000000	.800000	
MODE 14		SURF, HOLES					
J = 31	2.000000	0.000000	0.000000	0.000000	1.000000	0.000000	
J = 32	3.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
J = 33	2.000000	0.000000	0.000000	0.000000	4.000000	0.000000	
J = 34	2.000000	0.000000	0.000000	0.000000	2.000000	0.000000	
J = 35	.500000	0.000000	0.000000	0.000000	.250000	0.000000	
J = 36	.200000	0.000000	0.000000	0.000000	.750000	0.000000	
J = 37	.800000	0.000000	0.000000	0.000000	.750000	0.000000	
J = 38	1.000000	0.000000	0.000000	0.000000	2.000000	0.000000	
J = 39	.500000	0.000000	0.000000	0.000000	.700000	0.000000	
MODE 15		SURF, WAVINESS					
J = 41	2.000000	0.000000	0.000000	0.000000	2.000000	0.000000	
J = 42	2.000000	0.000000	0.000000	0.000000	2.000000	0.000000	
J = 43	3.000000	0.000000	0.000000	0.000000	4.000000	0.000000	
J = 44	3.000000	0.000000	0.000000	0.000000	3.000000	0.000000	
J = 45	.400000	0.000000	0.000000	0.000000	.100000	0.000000	
J = 46	.200000	0.000000	0.000000	0.000000	.200000	0.000000	
J = 47	.600000	0.000000	0.000000	0.000000	.600000	0.000000	
MODE 16		SURF, PROTUBERANCES					
J = 51	7.000000	0.000000	0.000000	0.000000	0.000000	2.000000	
J = 52	1.000000	0.000000	0.000000	0.000000	0.000000	.300000	
J = 53	.300000	0.000000	0.000000	0.000000	0.000000	1.000000	
J = 54	.200000	0.000000	0.000000	0.000000	0.000000	.700000	
J = 55	.800000	0.000000	0.000000	0.000000	0.000000	.900000	

```
SURFACE MISSING PART PARAMETERS, DWING(I)

DWING(1) =    .200     DWING( 7) =    .210     DWING(13) =    .800
DWING(2) =    .250     DWING( 8) =    .220     DWING(14) =    .050
DWING(3) =    .300     DWING( 9) =    .230     DWING(15) =    .050
DWING(4) =    .800     DWING(10) =   3.000     DWING(16) =    .200
DWING(5) =   1.000     DWING(11) =    .150     DWING(17) =    .300
DWING(6) =   3.000     DWING(12) =   0.000     DWING(18) =    .400
                                               DWING(19) =   0.000
                                               DWING(20) =   0.000

ENTER NLOCAP
```

C-141A WITH NUCLEAR DAMAGE

SUMMARY OF DAMAGE EFFECTS ON MINIMUM DRAG

M A C H = .80

BODY DAMAGE MODE	FUSELAGE	WHEEL WELL	BULLET	NACELLE
ROUGHNESS	-.00000	0.00000	0.00000	0.00000
FWD-FACING STEPS	0.00000	0.00000	.00009	0.00000
AFT-FACING STEPS	0.00000	.00025	0.00000	0.00000
HOLES	.00039	0.00000	0.00000	.00015
WAVINESS	.00018	0.00000	0.00000	.00004
PROTUBERANCES	.00039	0.00000	.00044	0.00000
BLUNTNESS	.00208	0.00000	.00027	0.00000
TOT.DRAG,EA.BODY =	.00304	.00025	.00080	.00019

SURFACE DAMAGE MODE	WING P1	WING P2	H.TAIL	V.TAIL	OUTBD PYL	INBD PYL
ROUGHNESS	-.00000	0.00000	0.00000	0.00000	0.00000	0.00000
FWD-FACING STEPS	0.00000	0.00000	.00036	0.00000	0.00000	.00002
AFT-FACING STEPS	0.00000	0.00000	.00015	0.00000	0.00000	.00002
HOLES	.00052	0.00000	0.00000	0.00000	.00001	0.00000
WAVINESS	.00051	0.00000	0.00000	0.00000	.00000	0.00000
PROTUBERANCES	.00056	0.00000	0.00000	0.00000	0.00000	.00054
MISSING PARTS	.00479----------		0.00000	-.00018	-.00007	-.00009
TOT.DRAG,EA.SURFACE =	.00638	-.00017	.00051	-.00018	-.00006	.00049

NET DRAG FROM EACH MODE OF DAMAGE

	BODIES	SURFACES	BODIES + SURFACES
ROUGHNESS	-.00000	-.00000	-.00000
FWD-FACING STEPS	.00009	.00028	.00046
AFT-FACING STEPS	.00025	.00017	.00042
HOLES	.00054	.00052	.00106
WAVINESS	.00022	.00052	.00073
PROTUBERANCES	.00083	.00110	.00193
BLUNTNESS	.00235	N/A	.00235
MISSING PARTS	N/A	.00428	.00428
TOTAL DAMAGE DRAG	.00427	.00697	.01124

C-141A WITH NUCLEAR DAMAGE

MACH = .800

UNTRIMMED AERODYNAMIC DATA

	UNDAMAGED			DAMAGED			INCREMENTS DUE TO NUCLEAR DAMAGE					
CL	ALPHA	CM	CD	ALPHA	CM	CD	DELTA ALPHA	DELTA CM	DELTA ROLL	DELTA CD MIN	DELTA CDL	DELTA CDTOTAL
0.000	-4.39	.0808	.01839	-6.29	-.7065	.02826	-1.90	-.7873	.0164	.01124	-.00127	.00997
.050	-3.68	.0749	.01762	-5.30	-.6361	.02881	-1.62	-.7110	.0140	.01124	-.00006	.01119
.100	-2.96	.0688	.01718	-4.30	-.5656	.02923	-1.34	-.6344	.0116	.01124	.00081	.01205
.150	-2.24	.0625	.01708	-3.30	-.4952	.02983	-1.08	-.5577	.0092	.01124	.00151	.01275
.200	-1.52	.0560	.01729	-2.30	-.4250	.03106	-.78	-.4810	.0068	.01124	.00253	.01377
.250	-.80	.0493	.01785	-1.31	-.3550	.03296	-.50	-.4043	.0043	.01124	.00388	.01511
.300	-.09	.0424	.01898	-.31	-.2852	.03573	-.23	-.3276	.0019	.01124	.00551	.01676
.350	.63	.0354	.02075	.69	-.2155	.03947	.05	-.2509	-.0005	.01124	.00747	.01872
.400	1.35	.0282	.02317	1.68	-.1460	.04417	.33	-.1742	-.0029	.01124	.00976	.02100
.450	2.07	.0209	.02633	2.68	-.0766	.04992	.61	-.0975	-.0053	.01124	.01234	.02359
.500	2.79	.0134	.03018	3.68	-.0074	.05669	.89	-.0208	-.0077	.01124	.01527	.02651
.550	3.51	.0058	.03495	4.67	.0618	.06466	1.17	.0559	-.0101	.01124	.01847	.02971
.600	4.22	-.0019	.04048	5.67	.1308	.07374	1.45	.1327	-.0125	.01124	.02202	.03326
.650	4.94	-.0096	.04707	6.67	.1997	.08415	1.72	.2094	-.0149	.01124	.02584	.03708
.700	5.66	-.0175	.05456	7.67	.2686	.09579	2.00	.2861	-.0173	.01124	.02999	.04123
.750	6.38	-.0254	.06407	8.66	.3374	.10986	2.28	.3628	-.0197	.01124	.03455	.04579
.800	7.10	-.0337	.07982	9.66	.4058	.13045	2.56	.4395	-.0221	.01124	.03939	.05063
.850	7.82	-.0431	.10468	10.66	.4737	.15571	2.84	.5168	-.0245	.01124	.03979	.05103
.900	8.53	-.0538	.13793	11.65	.5416	.18034	3.12	.5955	-.0269	.01124	.03117	.04241
.950	9.25	-.0660	.17919	12.65	.6096	.20497	3.40	.6756	-.0293	.01124	.01454	.02579
1.000	9.97	-.0798	.22811	13.65	.6776	.22960	3.68	.7574	-.0217	.01124	-.00975	.00149

C-141A WITH NUCLEAR DAMAGE

SUMMARY OF DAMAGE EFFECTS ON MINIMUM DRAG

M A C H = 1.20

BODY DAMAGE MODE	FUSELAGE	WHEEL WELL	BULLET	NACELLE
ROUGHNESS	-.00000	0.00000	0.00000	0.00000
FWD-FACING STEPS	0.00000	0.00000	.00009	0.00000
AFT-FACING STEPS	0.00000	.00044	0.00000	0.00000
HOLES	.00039	0.00000	0.00000	.00015
WAVINESS	.00381	0.00000	0.00000	.00061
PROTUBERANCES	.00040	0.00000	.00045	0.00000
BLUNTNESS	.00160	0.00000	.00103	0.00000
TOT.DRAG,EA.BODY =	.00621	.00044	.00156	.00077

SURFACE DAMAGE MODE	WING P1	WING P2	H.TAIL	V.TAIL	OLTED PYL	INED FRL
ROUGHNESS	-.00000	0.00000	0.00000	0.00000	0.00000	0.00000
FWD-FACING STEPS	0.00000	0.00000	.00036	0.00000	0.00000	.00002
AFT-FACING STEPS	0.00000	0.00000	.00025	0.00000	0.00000	.00004
HOLES	.00052	0.00000	0.00000	0.00000	.00001	0.00000
WAVINESS	.00412	0.00000	0.00000	0.00000	.00017	0.00000
PROTUBERANCES	.00057	0.00000	0.00000	0.00000	0.00000	.00055
MISSING PARTS	.00492	----------	0.00000	-.00149	-.00005	-.00006
TOT.DRAG,EA.SURFACE =	.01013	-.00009	.00061	-.00149	.00013	.00054

NET DRAG FROM EACH MODE OF DAMAGE

	BODIES	SURFACES	BODIES + SURFACES
ROUGHNESS	-.00000	-.00010	-.00000
FWD-FACING STEPS	.00009	.00037	.00046
AFT-FACING STEPS	.00044	.00029	.00072
HOLES	.00055	.00053	.00108
WAVINESS	.00442	.00428	.00871
PROTUBERANCES	.00084	.00111	.00196
BLUNTNESS	.00264	N/A	.00264
MISSING PARTS	N/A	.00323	.00323
TOTAL DAMAGE DRAG	.00898	.00982	.01879

C-141A WITH NUCLEAR DAMAGE

MACH = 1.200

U N T R I M M E D   A E R O D Y N A M I C   D A T A

CL	UNDAMAGED ALPHA	UNDAMAGED CM	UNDAMAGED CD	DAMAGED ALPHA	DAMAGED CM	DAMAGED CD	INCREMENTS DUE TO NUCLEAR DAMAGE: DELTA ALPHA	DELTA CM	DELTA ROLL	DELTA CD MIN	DELTA CDL	DELTA CDTOTAL
0.000	-4.01	.0687	.15837	-4.86	-.3977	.17607	-.85	-.4663	.0136	.01879	-.00110	.01769
.050	-3.56	.0671	.15861	-4.33	-.3691	.17764	-.77	-.4362	.0124	.01879	.00024	.01903
.100	-3.11	.0655	.15961	-3.81	-.3405	.17914	-.70	-.4060	.0112	.01879	.00074	.01953
.150	-2.67	.0637	.16137	-3.29	-.3120	.18111	-.62	-.3758	.0099	.01879	.00095	.01974
.200	-2.22	.0620	.16389	-2.77	-.2835	.18397	-.54	-.3455	.0087	.01879	.00128	.02007
.250	-1.78	.0601	.16717	-2.24	-.2551	.18771	-.46	-.3153	.0074	.01879	.00175	.02054
.300	-1.33	.0583	.17121	-1.72	-.2268	.19224	-.39	-.2850	.0062	.01879	.00234	.02113
.350	-.89	.0563	.17600	-1.20	-.1985	.19765	-.31	-.2548	.0049	.01879	.00306	.02185
.400	-.44	.0543	.18155	-.67	-.1702	.20425	-.23	-.2245	.0037	.01879	.00391	.02270
.450	.00	.0523	.18786	-.15	-.1420	.21154	-.15	-.1943	.0025	.01879	.00489	.02368
.500	.45	.0502	.19492	.37	-.1138	.21971	-.08	-.1641	.0012	.01879	.00599	.02478
.550	.89	.0481	.20274	.90	-.0857	.22876	.00	-.1338	-.0000	.01879	.00723	.02602
.600	1.34	.0459	.21132	1.42	-.0577	.23870	.08	-.1036	-.0012	.01879	.00859	.02738
.650	1.79	.0437	.22065	1.94	-.0296	.24949	.16	-.0733	-.0025	.01879	.01005	.02884
.700	2.23	.0415	.23078	2.47	-.0016	.26128	.23	-.0431	-.0038	.01879	.01171	.03050
.750	2.68	.0392	.24231	2.99	.0263	.27454	.31	-.0129	-.0050	.01879	.01344	.03223
.800	3.12	.0369	.25547	3.51	.0543	.28956	.39	.0174	-.0062	.01879	.01530	.03409
.850	3.57	.0345	.27024	4.03	.0822	.30633	.47	.0476	-.0075	.01879	.01730	.03609
.900	4.01	.0322	.28663	4.56	.1100	.32485	.54	.0779	-.0087	.01879	.01943	.03822
.950	4.46	.0298	.30463	5.08	.1379	.34476	.62	.1081	-.0100	.01879	.02133	.04012
1.000	4.90	.0274	.32426	5.60	.1657	.36456	.70	.1384	-.0112	.01879	.02151	.04030

4. PROGRAM DESCRIPTION

The AAT computer code is a long and complex program that uses many variables and subroutines. For this reason, a programmer who is not familiar with the details of the program operation may have difficulty in making modifications and additions. This program description is presented to aid the programmer in any future development work.

4.1 PROGRAM STRUCTURE

Figure 1-1 shows the program overlay structure and the positions of the 59 subprograms and subroutines that comprise the AAT computer code. The deck must be arranged as indicated by this figure; Overlay (0,0) with its program and subroutine are first, followed by Overlay (1,0), Overlay (2,0), Overlay (3,0), Overlay (3.1), Overlay (3,2), Overlay (3,3), and finally, Overlay (4,0).

The overlay structure was devised to allow a logical path through the program while keeping the required computer central memory core to a reasonable amount. Approximate 54,000 bites are required by the program.

4.2 SUBROUTINE DESCRIPTIONS

This section contains a description of each subprogram and subroutine listed in alphabetical order. The purpose of each is briefly discussed, followed by the statement required to call the subroutine and definitions for the variables used in the CALL statement.

SUBROUTINE DESCRIPTIONS
(listed in alphabetical order)

1. Subroutine AALO computes the zero-lift angle of attack, which is dependent upon camber, twist, and incidence. This subroutine is called with the statement

CALL AALO (SPEED)

where speed is the Mach number. The computed value of zero-lift angle is entered into COMMON OUTPUT.

2. Subroutine ACCR computes the aerodynamic center at low lift and at stall for single panel wings. The subroutine also obtains the lift curve slope of the wing. The low lift aerodynamic center is obtained through use of triple-interpolation of the data presented in Figures 4.1.4.2-22 and -27 in the DATCOM. Other subroutines called are: TLNT, AERA, and LNTP. This subroutine is called with the statement

CALL ACCR (SPEED, AR, SWPLE, SWPMC, TR, SPLAN, TOC, TW, FMCRO, XACR, CLAX, XACS)

where the input is

SPEED = Mach number
AR = Aspect ratio of exposed wing
SWPLE = Leading-edge sweep
SWPMC = Mid-chord sweep
TR = Taper ratio
SPAN == Exposed planform area
TOC = Thickness ratio
TW = Type airfoil section indicator
FMCRO = Zero-lift critical Mach number for complete configuration

and the output is

XACR = Low lift aerodynamic center referenced to leading-edge of exposed root chord
CLAX = Lift-curve slope of exposed planfom
XACS = Aerodynamic center at stall

3. Subroutine ADCL computes the effect of camber on the displacement of the drag polar. For Mach numbers less than 1.0 the lift coefficient for minimum profile drag is computed; for Mach numbers greater than or equal to 1.0, the lift coefficient for minimum drag is computed. This subroutine is called with the statement

CALL ADCL(SPEED, CLOPT)

where SPEED is the Mach number. At subsonic speeds, CLOPT is the C_L for minimum profile drag at supersonic speeds CLOPT is the polar displacement C_L for $C_{D_{min}}$.

4. Subroutine AT62 uses the altitude and equations representing the 1962 U.S. Standard Atmosphere to calculate the unit Reynolds number over Mach number, $\frac{R_N}{\text{Length} \cdot \text{Mach}}$. This subroutine is called with the statement

CALL AT 62 (ALT, RNOMFT, IFT)

where ALT = altitude
RNOMFT = Reynolds number divided by Mach x Length
IFT = Indicator to specify English or metric units.

If IFT = 0 units are in feet
If IFT = 1 units are in meters

5. Subroutine ADJUST adjusts an aerodynamic parameter, y_1, to a new value, y_2, by the equation

$$y_2 = y_1 \cdot V_M + V_A$$

where V_M and V_A are correlation multiplier and adder factors determined from input. These factors are functions of either Mach number or C_L. Subroutine LNTP is also called. This subroutine is called with the statement

CALL ADJUST (ID, ID2, XVAR, YVAR)

where ID = Parameter identification number for Mach number cases
ID2 = Parameter identification number for C_L cases
XVAR = Value of Mach number if ID greater than 0, or value of C_L if ID = 0 and ID2 greater than 0
YVAR = Input value y_1 is changed to output value y_2

6. Subroutine AERA calculates the angle of attack for a given untrimmed C_L condition. For supersonic Mach numbers

$$\alpha = C_L/C_{L_\alpha} + \alpha_{LO}$$

For subsonic conditions the angle of attack is calculated by one of the three different methods depending on whether the wing is a high aspect ratio, low aspect ratio or a cranked planform. For low-aspect-ratio and cranked planform the effect of vortex lift is accounted for in the angle of attack calculations. Other subroutines called are: DLNT and LNTP. This subroutine is called with the statement

CALL AERA (SPEED, CL, ALPHA)

where SPEED = Mach number (I)
CL = Lift coefficient (I)
ALPHA = Angle of attack (0)

7. Program AEROA controls the sequence of calculations required to compute minimum drag for a given Mach-altitude or Mach-Reynolds number condition which are contained in COMMON OUTPUT.

The sequence of the calling of subroutines for aerodynamic calculations is controlled by the parameter JPASS which is defined by program SURVEY. The parameter JPASS is used to prevent calling certain subroutines on repeat passes through AEROA if the values they calculate remains fixed. AEROA calls subroutines AT62, DMIN, FDRG, LNTP and WDRG, and AEROA is called from program SURVEY with the statement

CALL OVERLAY(4 HOVLY, 3,3)

8. Program AEROB controls the sequence of calculations required to compute the trimmed lift, drag, moment, and angle of attack for a given set of conditions. The results of the aerodynamic calculations are contained in COMMON OUTPUT.

The subroutines directly called from program AEROB are ADJUST, AERA, AFTCD, CDDR, CDL1, CDL2, CDRG, CLBRK, CLWBT, CMOW, TDRG, and WBAC. The parameter JPASS is used to prevent calling certain subroutines on repeat passes through AEROB if the value they calculate remains fixed. Program AEROB is called from program SURVEY with the statement

CALL OVERLAY(4 HOVLY, 3,4)

9. Subroutine AER2 computes the lift-curve slope for a wing surface defined by the data in COMMON CALC. The lift-curve slope is computed using a modified Polhamus expression in the subsonic range which is extended to match the two-dimensional linear-theory value at high supersonic Mach numbers. This subroutine is called with the statement

CALL AER2 (SPEED, CLA)

where SPEED = Mach number
CLA = Computed lift-curve slope (ΔC_L per degree).

10. Subroutine AFTCD computes the drag increment due to the fuselage aft-end upsweep as a function of angle of attack. Subroutine DLNT is also called. This subroutine is called with the statement

CALL AFTCD (ALPHA, CDAFT)

where ALPHA = Angle of attack of the wing
CDAFT = Drag increment due to aft-end upsweep

11. Subroutine BDRG computes the base drag of an arbitrary body. An empirical equation is used to determine base drag. This subroutine is called with the statement

CALL BDRG (SPEED, AB, SREF, CBD)

where SPEED = Mach number (I)
AB = Base area (I)
SREF = Reference area (I)
CDB = Base drag coefficient (0)

12. BLOCK DATA is used to enter in tabular form the data from various tables and charts that are needed by other subroutines in the program.

13. Subroutine CDDR calculates the drag rise, the two limit Mach numbers, and the lift-curve slope at those Mach numbers. The limit Mach numbers and associated lift-curves are used later in subroutine CDL1 to define the polar shape in the transonic region. CDDR has two entry points, on the first pass, the constants in the drag rise equation, the two limit Mach numbers, and their associated lift-curves are computed. Only the drag rise is computed in subsequent passes. Other subroutines called are: CLWBT, FDRG, LNTP, and WDRG. This subroutine is called with the statement

CALL CDDR (CL, XMACH, RNOFT, CDR)
CALL CDDR1

where CL = Lift coefficient
XMACH = Mach number
RNOFT = Reynolds number per unit length
CDR = Drag rise

14. Subroutine CDL1 calculates the constants which are used by subroutine CDL2 to determine the drag polar. Other subroutines called are: ADCL, DLNT, KGIN, and LNTP. This subroutine is called with the statement

CALL CDL1 (SPEED, RNOFT, FK, DELCL, PRIMEK, AKD, AKB)

where SPEED = Mach number (I)
RNOFT = Reynolds No. per unit length (I)
FK = Polar shape factor below polar break (0)
DELCL = Polar lift displacement (0)
PRIMEK = Additional drag factor for drag polar above polar break (0)
AKD = Theoretical drag-due-to-lift factor (0)
AKB = Separation drag factor used to calculate drag polar above separation lift coefficient (0)

15. Subroutine CDL2 uses the polar shape factors determined by subroutine CDL1 and the polar break and separation lift coefficients determined by subroutine CLBRK to compute the drag due to lift. This subroutine is called with the statement

CALL CDL2 (SPEED, CL, AEROK, DELCL, PRIMEK, AKD, AKB, CDL)

where SPEED = Mach number (I)
CL = Lift coefficient (I)
AEROK = Polar shape factor below polar break (I)
DELCL = Polar lift displacement (I)
PRIMEK = Additional drag factor for drag polar above polar break (I)
AKD = Theoretical drag-due-to-lift factor (I)
AKB = Separation drag factor (I)
CDL = Drag due to lift (O)

16. Subroutine CDRG calculates the drag increment due to wing camber. Wing camber causes a lift displacement in the drag polar; this displacement lift increment is related to the difference between the minimum profile drag and the minimum drag of the polar. Subroutine CDRG is called with the statement

CALL CDRG (SPEED, AEROK, DELCL, CDC)

where SPEED = Mach number (I)
AEROK = Polar shape factor (I)
DELCL = Polar lift displacement (I)
CDC = Camber drag (O)

17. Subroutine CDWN calculates the nose wave drag of body and nacelle components. Subroutine DLNT is also called. This subroutine is called with the statement

CALL CDWN (AMAX, XLNOS, RIN, BETA, CDW)

where AMAX = Maximum cross-sectional area (I)
XLNOS = Length of nose (I)
RIN = Radius of inlet area (I)
BETA = $\sqrt{M^2 - 1}$
CDW = Wave drag of component based on maximum cross-sectional area (O)

18. Subroutine CDWT calculates the boattail wave drag of body and nacelle components. Subroutine LNTP is also used. Subroutine CDWT is called with the statement

CALL CDWT (AMAX, XLAFT, REX, BETA, CDW)

where AMAX = Maximum cross-sectional area (I)
XLAFT = Length of boattail (I)
REX = Exit or base area (I)
BETA = $\sqrt{M^2 - 1}$
CDW = Wave drag of component based on maximum cross-sectional area (O)

19. Subroutine CDWW calculates the wave drag for component represented as surfaces. It is called with the statement

CALL CDWW (CDOSR)

where CDOSR = wing wave drag based on the configuration reference area

20. Subroutine CFEQ calculates the flat-plate skin friction coefficient using the White-Christoph technique for turbulent flow, the Blasius relation for laminar flow, and a momentum thickness matching technique for partial laminar-turbulent flow. This subroutine is called with the statement

CALL CFEQ (RNOFT, ZMACH, CBAR, XTR, CF)

where RNOFT = Reynolds number per unit length
ZMACH = Mach number
CBAR = Length
XTR = Distance along CBAR where transition occurs
CF = Skin friction coefficient

21. Subroutine CFK3 calculates the equivalent, flat-plate friction drag coefficient for a component with nuclear damage. The surface condition can be described by defining the area and the corresponding roughness height for as many as three different regions on each component. It is called with the statement

CALL CFK3 (LENGTH, RNOFT, XOL1, XOL2, K1, K2, K3, XMACH, CF)

where LENGTH = Component length
RNOFT = Reynolds number per unit length
XOL1 = Fraction of length where regions 1 and 2 merge
XOL2 = Fraction of length where regions 2 and 3 merge
K1 = Roughness height of region 1
K2 = Roughness height of region 2
K3 = Roughness height of region 3
XMACH = Mach number
CF = Equivalent friction coefficient

22. Subroutine CLBRK calculates the lift coefficients for polar break, separation drag onset, and maximum lift. This subroutine also calculates other aerodynamic parameters used in subroutine AERA to compute the angle of attack as a funtion of C_L. This subroutine is called with the statement

CALL CLBRK (SPEED, RE, RNOFT)

where SPEED = Mach number (I)
RE = Reynolds number parameter (I)
RNOFT = Reynolds number/foot (I)

The output, which is entered in COMMON CALC, is defined by

CLPB = Polar break lift coefficient
CLDB = Separation lift coefficient
CLMAX = Maximum lift coefficient
ABRK = Angle of attack for polar break
AMAX = Angle of attack at CLMAX
DAMAX = Increment in angle of attack between AMAX and a linear value of alpha at CLMAX
DEL = Tail lift increment to CLMAX
CLS = Lift coefficient where CL versus alpha becomes nonlinear
ARLO = Aspect ratio limit between low AR and high

23. Subroutine CLWBT controls the sequence of calculations that compute the total wing-body-tail lift curve slope, zero lift angle of attack, and the factors used to compute drag and lift increments due to a horizontal tail deflection. Other subroutines that are called are AALO, ADJUST, AER2, and TAI1. Output is entered into COMMON CALC. This subroutine is called with the statement

CALL CLWBT (SPEED)

where SPEED = Mach number

24. Subroutine CMOW computes the moment at zero-lift for the wing-body configuration. Subroutine TLNT is also used. CMOW is called with the statement

CALL CMOW (SPEED, CMO)

where SPEED = Mach number (I)
CMO = Wing-body C_{m_o}

25. Subroutine CPUOV calculates the pressure coefficient, Cp at x/c for an infinite sheared wing. The Cp is obtained by solving Equations (93) and (90) in the Royal Aero. Soc. TDM-6312. The arrays corresponding to $S^{(1)}(x)$, $S^{(2)}(x)$, $S^{(3)}(X)$, $S^{(4)}(x)$, and $S^{(5)}(x)$ in Equation (93) are obtained from COMMON CALC, and were defined in subroutine CPZT. This subroutine is called by the statement

CALL CPUOV (S, A, SWP, IV, CPI, CP, XM)

where S = Sign indicator (+1 for upper surface Cp, and -1 for lower surface Cp) (I)
A = Angle of attack (I)
SWP = Sweep angle (I)
IV = Control point, $x/c = \frac{1}{2}(1 - \cos(\frac{IV}{32}))$ (I)
CPI = Incompressible Cp (O)
CP = Compressible value of Cp (O)
XM = Mach number (I)

26. Subroutine CPZT computes the critical Mach number using the local Mach number normal to the isobar in the mid-span region of the wing. The mid-chord sweep and the aspect ratio of the wing are used to define an effective isobar sweep. The subroutine obtains the airfoil geometry from subroutine SECT and the pressure distribution around the airfoil from subroutine CPUOV. The subroutine then uses the incompressible pressure at the crest of the airfoil and isentropic flow relationships to calculate the critical Mach number on the wing. The predicted wing critical Mach number is prevented from exceeding the critical Mach number of the fuselage, which is calculated as a function of fuselage fineness ratio. The output, which consists of a series of critical Mach number and lift combinations, are entered into COMMON CALC. Subroutines CPUOV, LNTP, and SECT are also used. CPZT is called with the statement

CALL CPZT (ID, XMACH, TOC, CLD, SWEEP)

where	ID	=	Type identification of the airfoil (I)
	XMACH	=	Mach number for compressible Cp solutions (I) (set equal to 0.6 in Program MCRIT)
	TOC	=	Airfoil section thickness ratio (I)
	CLD	=	Airfoil section camber (I)
	SWEEP	=	Mid-chord sweep at mid-semi-span of the wing (I)

27. Subroutine DINPT is called only if damage calculations are to be made. DINPT reads the damage input and prints out the values as a check that the data are correctly entered in the problem input. This subroutine is called with the statement

CALL OVERLAY (4HOVLY, 1, 2)

28. Function DLNT is a two-dimensional, nth-order Lagrangian interpolation procedure. Function DLNT also uses Subroutine LNTP. It is called with the statement

CALL DLNT (XBAR, YBAR, X, Y, F, NX, NY, NXMAX, LOX, LOY)

where	XBAR	=	The X value at which a value of the function is to be interpolated (I)
	YBAR	=	The Y value at which a value of the function is to be interpolated (I)
	X	=	The array of X values (I)
	Y	=	The array of Y va ues (I)
	F	=	The values of the function f(x,y) (I)
	NX	=	The size of the X array and the F array in the X direction (I)
	NY	=	The size of the Y array and the F array in the Y direction (I)
	NXMAX	=	The dimension of the F array in the X direction in the calling routine (I)
	LOX, LOY	=	Number of points to be used in the X and Y directions, respectively, in the interpolations: 1 for step, 2 for linear, 3 for parabolic, 4 for cubic, etc. (I)

29. Subroutine DMIN controls the sequence of calculations necessary to compute minimum drag. It uses subroutines ADJUST, BDRG, FDRG, and WDRG. It is called with the statement

CALL DMIN (SPEED, RNOFT, CDMIN)

where SPEED = Mach number (I)
RNOFT = Reynolds number per unit length (I)
CDMIN = Minimum drag (O)

30. Subroutine ELEVON computes the lift, moment, and drag effectiveness of a wing trailing-edge flap control surface (elevon). The elevon is used as a trim surface if no horizontal tail or canard is specified in the input geometry. The computed parameters for the elevon are stored in COMMON OUTPUT and latter used in subroutine TDRG for trim calculations. Subroutine ELEVON uses subroutine DLNT and is called with the statement

CALL ELEVON(SPEED), where SPEED is the Mach number

31. Subroutine FDRG calculates friction, form and interference drag for all the components on the airplane. All output is entered into COMMON OUTPUT. Other subroutines called are CFEQ, FFACT, and IFACT. It is called with the statement

CALL FDRG (SPEED, RNOFT)

where SPEED = Mach number
RNOFT = Reynolds number per unit length

32. Subroutine FFACT computes the form factors for each component. It is called with the statement

CALL FFACT (ID, GEOM, TYP, CLD, SPEED, CRITM, FF)

where ID = Identification for bodies, nacelles, or surface components
GEOM = Fineness ratio for bodies and nacelle components; or thickness ratio for surface components
TYP = Airfoil type identification number
CLD = Airfoil camber
SPEED = Mach number
CRITM = Critical Mach number of configuration
FF = Form factor

33. Subroutine EFFQ calculates the ratio of local dynamic pressure to freestream dynamic pressure. It is called with the statement

CALL EFFQ (RNOFT, XI, XMACH, HEIGHT, QEFF)

where RNOFT = Reynolds number per unit length
XI = Distance from nose of component to station at which local dynamic pressure is to be calculated.

XMACH = Mach number
HEIGHT = Distance above surface at which local dynamic pressure is to be calculated
QEFF = Local to freestream dynamic pressure, q/q

34. Program GEOM uses input to compute additional geometric parameters that are required by other subroutines. The values computed for individual bodies and surfaces are entered into COMMON INPUT. Computed parameters which apply to the total airplane are inserted into COMMON CALC. This program is called with the statement

CALL OVERLAY (4HOVLY, 2, 0)

35. Subroutine IFACT computes the interference factors for each component. It also uses subroutine DLNT and is called with the statement

CALL IFACT (ID, PARM, CRITM, SPEED, FI)

where ID = Identification or bodies or surface components (I)
PARM = Fuselage Reynolds number for bodies, or maximum thickness sweep for surface components (I)
CRITM = Critical Mach number of configuration (I)
SPEED = Mach number (I)
FI = Interference factor (O)

36. Program INPT reads the geometry of the undamaged configuration and the aerodynamic conditions at which the problem is to be run. It is called with the statement

CALL OVERLAY (4HOVLY, 1, 1)

37. Subroutine KGIN computes the polar shape factor and polar displacement for a drag polar with drag rise. A least-squares, second-degree curve is fitted to the drag polar with drag rise. This is needed by subroutine CDL1 in order to interpolate the polar shape in the transonic region between M_{L1} and M_{L2}. It uses subroutines CDDR and LSPCF. It is called with the statement

CALL KGIN (CLDB, AKIN, DECLIN, SPEED, AKOUT, DCLOUT)

where
- CLDB — Upper C_L limit for polar calculation (I)
- AKIN — Polar shape factor without drag rise (I)
- DECLIN — Polar displacement C_L without drag rise (I)
- SPEED — Mach number (I)
- AKOUT — Equivalent polar shape factor with drag rise (O)
- DCLOUT — Equivalent polar displacement with drag rise (O)

38. Subroutine LNTP is a one-dimensional, nth-order Lagrangian interpolation procedure. It uses no other subroutines and is called with the statement

CALL LNTP (XBAR, YBAR, X, Y, M, NO)

where
- XBAR — The abscissa value at which an ordinate is to be interpolated (I)
- YBAR — The interpolated ordinate (O)
- X — The array of abscissas (I)
- Y — The array of ordinates (I)
- M — The size of the arrays (I)
- NO — The number of points to be used in the interpolation: NO=1 for step, NO=2 for linear, NO=3 for parabolic, NO=4 for cubic, etc. (I)

39. Subroutine LSPCF is a least squares, polynominal curve fit subroutine. It is called by the statement

CALL LSPCF (X, Y, N, MP, SA, RE, SIGMA, IT)

where
- X — The independent variable array
- Y — The dependent variable array
- N — The number of points
- MP — The degree of polynomial fit
- SA — The array containing the coefficients

IT An error indicator
=0 if the coefficients were found correctly
≠0 if the coefficients were not found correctly

SIGMA An unbiased estimate of the standard deviation
RE The array containing the residuals

40. Program MCRIT constructs a table of critical Mach number versus C_L from either an input table or by using an empirical method. It uses subroutines ADJUST, CPZT, and LNTP. MCRIT is called with the statement

CALL OVERLAY (4HOVLY, 3, 2)

41. Program MRIT (MAIN PROGRAM controls the logic of the calling of the four primary overlays. The lengths of the principal COMMON blocks used in all four overlays are also specified in this program. The primary overlays are called with the following statements.

```
OVERLAY(4HOVLY, 1, 0)     (INPT)
OVERLAY(4HOVLY, 2, 0)     (GEOM)
OVERLAY(4HOVLY, 3, 0)     (SURVEY)
OVERLAY(4HOVLY, 4, 0)
```

42. Subroutine NTRIM computes the lift and drag increments required to trim the damaged aircraft pitching moments. It also computes the rolling moments that result from wing or tail damage and makes a rough estimate of the drag that results from trimming these moments. NTRIM is called from subroutine NUCDAM by the statement

CALL NTRIM(J)

where J = Number of the survey condition at which data are being evaluated

43. Program NUCDAM computes the effects of damage on minimum drag, and equations are included to account for any of the fourteen different modes. Data describing the damage are stored under the variable names DBOD(I,J), DSUR(I,J), and DWING(I). These variables are included in COMMON INPUT. Minimum drag increments

computed by this subroutine are stored in the variable DAMCD (I,J), which is included in COMMON NUCOUT. It calls subroutines WRITE, NUCDAM 2, NTRIM, and WRITE 2. It is called with the statement

CALL OVERLAY (4HOVLY, 4, 0)

44. Subroutine NUCDAM2 computes the effects of damage on the lift-curve slope, polar shape factor, and pitching moment curve. It then computes, for the untrimmed configuration, the lift, drag, and pitching moment. It is called by the statement

CALL NUCDAM2(J)

where J = Number of the survey condition at which data are being evaluated.

45. Subroutine RINPT is used to make minor changes to the basic problem input after the initial problem has been run. This provision allows additional problems to be run with ease if changes are minor. It is called with the statement

CALL OVERLAY (4HOVLY, 1, 3)

46. Subroutine SECT calculates the thickness and camber airfoil ordinates which are used in subroutine CPZT to calculate pressure distributions. SECT can calculate the section data for the standard NACA 6-series and 4-digit airfoils and the boconvex airfoil. The subroutine can also obtain the airfoil ordinates at the control points, x/c, needed for pressure solutions by interpolation on a table of input ordinates. It uses subroutine LNTP and is called by the statement

CALL SECT (ID, TOC, CLD)

where ID = Airfoil section identification number (I)
TOC = Airfoil thickness ratio (I)
CLD = Airfoil camber (I)

47. Subroutine SETUP places initial values of 0.0 in the input and geometry common blocks. This is done to prevent random data from being used in program calculations if input data does not specify each value. It is called with the statement

CALL SETUP

48. Program SURVEY controls the sequence of calculations to produce a lift, moment and drag variation for each high-speed survey condition specified by the input.

For variable sweep configurations the program will first call program VGEOM with the wing leading-edge sweep set at the forward position and then recall program VGEOM with the sweep set at the aft position. This is done in order to setup program VGEOM for geometry calculations at any arbitrary sweep position. SURVEY then enters a DO LOOP where the high sweep survey conditions are set up. SURVEY calls VGEOM and MCRIT to recalculate the geometry and the configuration critical Mach number each time the leading-edge sweep is changed in a survey. SURVEY then enters an inner DO LOOP where a sequence of untrimmed C_L are generated and programs AEROA and AEROB are called to obtain the trimmed lift, moment and drag, SURVEY then prints out the aerodynamic results. It is called with the statement

CALL OVERLAY (4HOVLY, 3, 0)

The following secondary overlays are called from SURVEY:

```
OVERLAY(4HOVLY, 3, 1)      (VGEOM)
OVERLAY(4HOVLY, 3, 2)      (MCRIT)
OVERLAY(4HOVLY, 3, 3)      (AEROA)
OVERLAY(4HOVLY, 3, 4)      (AEROB)
```

49. Subroutine TAIL computes the lift curve slope contribution of the tail along with factors used to compute lift and drag increments due to a horizontal tail deflection. These factors are computed by first solving for the downwash, dynamic-pressure, exposed area lift-curve slope, carry-over lift factors and induced drag for the tail. Subroutines AER2 and LNTP are also used. The output is entered in COMMON CALC. TAIL is called with the statement

CALL TAIL (SPEED)

where SPEED = Mach number (I)
CLAT = Lift-curve slope contribution of the tail (O)
A = Trim drag factor (O)
B = Trim drag factor (O)
AOH = Angle of zero lift of the tail (O)
CLDH = Change in lift due to tail deflection factor (O)
DEDA = Change in downwash per change in angle of attack (O)

50. Subroutine TDRG calculates moment using the wing-body C_{m_o} and aerodynamic center and the tail lift and moment arm. The moment is calculated at zero horizontal tail setting. The tail deflection required to trim and the resulting lift and drag increments are also computed. TDRG is called with the statement

CALL TDRG (DCLT, DCDT)

Where DCLT = Increment in lift due to trim
DCDT Increment in drag due to trim

51. Subroutine TLNT is a triple-linear interpolation procedure. It uses subroutines DLNT and LNTP. It is called with the statement

CALL TLNT (XBAR, YBAR, ZBAR, FBAR, X, Y, Z, F, NX, NY, NZ, NXMAX, NYMAX)

where XBAR = The X value at which a value of the function is to be interpolated (I)
YBAR = The Y value at which a value of the function is to be interpolated (I)
ZBAR = The Z value at which a value of the function is to be interpolated (I)
FBAR = The interpolated value of the function F(X,Y,X) (I)
X = The array of X values (I)
Y = The array of Y values (I)
Z = The array of Z values (I)
F = The three-dimension array F values (I)
NX = The size of the X array and the F array in the X direction (I)
NY = The size of the Y array and the F array in the Y direction (I)
NZ = The size of the Z array and the F array in the Z direction (I)
NXMAX = The dimension of the F array in the X direction in the calling routine (I)
NYMAX = The dimension of the F array in the Y direction in the calling routine (I)

52. Program VGEOM computes the geometry parameters that vary with wing sweep for variable-sweep configurations. The program is first called by program SURVEY at the forward and most aft sweep positions in order to set up VGEOM for any arbitrary sweep calculation. It is called with the statement

CALL OVERLAY (4HOVLY, 3, 1)

53. Subroutine WBAC computes the wing-body aerodynamic center location. It first calculates the aerodynamic center of the wing carry-over lift on the body and the aerodynamic center of the forebody. A composite aerodynamic center is then computed. Subroutines ACCR, DLNT and LNTP are used. It is called with the statement

CALL WBAC (SPEED, XACR)

where SPEED = Mach number (I)
XACR = Aerodynamic center of the wing-body configuration reference to the leading edge of the exposed root chord

54. Subroutine WDRG calculates the wave drag for all the components on the airplane and enters the results into COMMON OUTPUT. The following subroutines are used: CDWH, CDWT, and CDWW. It is called with the statement

CALL WDRG (FMACH)

where FMACH = Mach number

55. Subroutine WRITE prints the minimum drag increments calculated by subroutine NUCDAM. The total drag increment on each component is shown and the total increment due to each mode of damage is also shown. It is called from subroutine NUCDAM with the statement

CALL WRITE (J)

Where J = Number of the survey condition at which data are being evaluated.

56. Subroutine WRITE2 prints a summary of the aerodynamic data that are calculated by the AAT computer code. Lift and drag data are shown for both the undamaged and damaged configurations. It is called from subroutine NUCDAM by the statement

CALL WRITE2 (J)

where J = Number of the survey condition at which data are being evaluated.

57. Subroutine XINPT reads the card in the input deck that directs the program to evaluate a damaged configuration, change parameters in the basic input, or end the problem. XINPT interprets the message on the card and calls the appropriate subroutines required to comply with the directive. It is called with the statement

CALL OVERLAY (4HOVLY, 1, 0)

4.3 COMMON BLOCK DESCRIPTIONS

Common blocks in the AAT procedure have been utilized in a manner that helps the programmer keep track of variables that are used. Most of the data used internally are contained in one of the following common blocks:

1. Common block INPUT contains all input data.

2. Common block CALC contains geometric parameters that are computed by the program.

3. Common block OUTPUT contains the aerodynamic parameters that are computed by the program.

4. Common block NUCOUT contains the aerodynamic data that are computed for the damaged configuration.

Table 4-1 lists the name of each variable in the common block INPUT as it is used in the programming statements. The position of each variable within the common block is indicated, and a brief description of each variable is given. Similar information for the OUTPUT, CALC, and NUCOUT common blocks is given in Tables 4-2 through 4-4.

TABLE 1

Variables in Common Block Input

A( )	VARIABLE NAME	DESCRIPTION
	Input Data Block #1 (General Information)	
(1)	NBODYS	Number of Bodies
(2)	NNAC	Number of Nacelles
(3)	NPNLS	Number of Panels on Main Wing
(4)	NHT	Number of other lifting surfaces
(5)	NVT	Number of Non-Lifting Surfaces
(6)	ISWP	Indicator for Sweep
(7)	IREF	Indicator for Reference Alpha
(8)	IWNG	Indicator for Wing Planform Definition
(9)	NAFO	Number of airfoil ordinates
(10)	METER	Indicator for metric units
(11)	SREF	Reference Area
(12)	CMAC**	Reference Aerodynamic chord
(13)	XMAC**	X-Station of leading edge of CMAC
(14)	ZCG	Z-Station of CG
(15)	TWIST	Wing twist
(16)	ROUGHK	Surface Roughness Height (inches)
(17)	FMISC	Percent friction drag for misc. items
(18)	NPODS	NBODYS + NNAC
(19)	SPLAN	Planform Area
(20)	TAPR	Tip chord/root chord
(21)	SWP	Wing leading-edge sweep
(22)	ITRIM	Trim Indicator
(23)	RLE	Leading-Edge radius
(24)	TOCR	Reference t/c for input airfoil (NAFO > 0)
(25)	CLDR	Reference CLD for input airfoil (NAFO > 0)
(26)	TECH	Technology Factor
(27)	CONCL	Conical Camber
(28)	UPFUS	Aft Fuselage Upsweep Angle
(29)	AOB	Aft Fuselage width-to-height ratio

(Table 1 Continued)

A( )	VARIABLE NAME	DESCRIPTION
	Input Data Block #2 (Body Geometry)	
(29+I)	BNAME (I)	Name of Body I, I=1, 7
(36+I)	BOD (I,1)	Length
(43+I)	(I,2)	Width
(50+I)	(I,3)	Height
(57+I)	(I,4)**	Wetted area, Total for Type I Bodies
(64+I)	(I,5)	Interference Factor
(71+I)	(I,6)	Number of bodies of Type I
(78+I)	(I,7)**	Maximum cross-sectional area
(85+I)	(I,8)	Base streamtube area
(92+I)	(I,9)	Nose length
(99+I)	(I,10)	Boattail length
(106+I)	(I,11)	Base drag area
(113+I)	I,12)	Inlet area
(120+I)	(I,13)	Vacant
(127+I)	(I,14)*	Fineness ratio
(134+I)	(I,15)	Vacant (Reserved for future expansion)
⋮	⋮	↓
(169+I)	(I,20)	
	(Input Data Block #3 (Surface Geometry)	
(176+I)	SNAME(I)	Name of surface I, I=1,7
(183+I)	SUR (I,1)	Airfoil section type
(190+I)	(I,2)	Airfoil 2-D camber
(197+I)	(I,3)	Thickness ratio, t/c
(204+I)	(I,4)	Leading-edge sweep
(211+I)	(I,5)*	x/c for maximum t/c
(218+I)	(I,6)	Vacant
(225+I)	(I,7)**	Wetted Area (Total)
(232+I)	(I,8)	Exposed root chord
(239+I)	(I,9)	Tip chord
(246+I)	(I,10)	Exposed Semi-span
(253+I)	(I,11)	X-station at leading edge of root chord
(260+I)	(I,12)	Y-station at leading edge of root chord
(267+I)	(I,13)	Z-station at leading edge of root chord
(274+I)	(I,14)	Incidence
(281+I)	(I,15)	Vacant
(288+I)	(I,16)	Vacant

(Table 1 Continued)

A( )	VARIABLE NAME	DESCRIPTION
	(Input Data Block #3 (Surface Geometry), Continued)	
(295+I)	SUR (I,17)*	Exposed taper ratio
(302+I)	(I,18)*	Exposed planform area (Total)
(309+I)	(I,19)*	Aspect ratio
(316+I)	(I,20)*	Characteristic length
(323+I)	(I,21)*	Sweep of quarter-chord
(330+I)	(I,22)*	Sweep of mid-chord
(337+I)	(I,23)*	Sweep of trailing edge
(344+I)	(I,24)*	Sweep of maximum thickness
(351+I)	(I,25)	Vacant, Reserved for future expansion
⋮	⋮	↓
(386+I)	(I,30)	

Input Data Block #4 (Arbitrary Airfoil)

A( )	VARIABLE NAME	DESCRIPTION
(394-423)	AFX(I)	x/c for airfoil input, I = 1, 30
(424-453)	AFC(I)	Camber distribution
(455-483)	AFT(I)	Thickness distribution

Input Data Block #5 (Variable Sweep)

A( )	VARIABLE NAME	DESCRIPTION
(484)	XPIVOT	X-station of wing pivot
(485)	YPIVOT	Y-station of wing pivot
(486)		Vacant
(487)	AFTSW	Maximum aft sweep of leading edge
(488)	AFTCB	MAC of movable panel in aft position
(489)	AFTOC	t/c of movable panel in aft position
(490)	AFTAW	A_{wet} of movable panel in aft position

Input Data Block #6 (Survey Conditions)

A( )	VARIABLE NAME	DESCRIPTION	
(491)	NSURV	Number of surveys	
(492)	NCLAS	Number of evenly spaced C_L's	
(492+I)	FMSURV(I)	Mach numbers for surveys	I = 1, 20
(512+I)	ALT(I)	Altitude for surveys	↓
(532+I)	CG(I)	X-station of cg for surveys, fraction of MAC	↓
(552+I)	SWPV(I)	Leading-edge sweep for surveys	↓
(572+I)	CLLO(I)	Low C_L for surveys	↓
(592+I)	CLHI(I)	High C_L for surveys	↓
(612+I)		Vacant	

(Table 1 Continued)

Input Data Block #7 (Adjustment Factors)

A( )	VARIABLE NAME	DESCRIPTION
(632+I)	IVAL(I)	Indicator for parameters to be adjusted, I=1,20
(653)	NXVAR	Number of Mach numbers in table of Mach function adjustment factors
(654)	NADJ	Number of parameters to be adjusted as a function of Mach number
(655)	NXCL	Number of C_L values in the table of lift function adjustment factors
(656)	NADJ2	Indicator for M_{CR}
(656+I)	X(I)	Mach numbers for the table of Mach adjustment factors I=1,15
(672-806)	YM(I,J)	Multiplier factors, I=15, J=9
(807-941)	YA(I,J)	Adder factors, I=15,J=9
(941+I)	XCL(I)	C_L values for the table of lift function adjustment factors, I=1,15

Input Data Block #8

A( )	VARIABLE NAME	DESCRIPTION
(957) - (1641)	Reserved for future expansion	

Input Data Block #9 (Damage Mode Indicators)

A( )	VARIABLE NAME	DESCRIPTION
(1641+I)	IDAM(I)	Indicator to select damage modes, I=1,17

Input Data Block #10 (Body Damage Parameters)

A( )	VARIABLE NAME	DESCRIPTION
(1661+I)	DBOD(I,1)	Roughness factor before damage on Body I, I=1,17
(1668+I)	(I,2)	Roughness factor of d[illegible]ed area on Body I, I=1,17
(1675+I)	(I,3)	x/l where damage starts
(1682+I)	(I,4)	x/l where damage ends
(1689+I)	(I,5)	Fraction of area affect by damage
(1697-1731)	(I,6)-(I,10)	Vacant
(1731+I)	(I,11)	Number of forward-facing steps on Body I, I=1,7
(1738+I)	(I,12)	Width of each step
(1745+I)	(I,13)	Height of each step
(1752+I)	(I,14)	x/l at first step
(1759+I)	(I,15)	x/l at last step
(1767-1801)	(I,16)-(I,20)	Vacant

(Table 1 Continued)

A( )	VARIABLE NAME	DESCRIPTION
	(Input Data Block #10 (Body Damage Parameters), Continued)	
(1801+I)	DBOD(I,21)	Number of aft-facing steps on Body I,I=1,7
(1808+I)	(I,22)	Width of each step
(1815+I)	(I,23)	Height of each step
(1822+I)	(I,24)	x/1 at first step
(1829+I)	(I,25)	x/1 at last step
(1837-1871)	(I,26)-(I,30)	Vacant
(1871+I)	(I,31)	Total number of holes in Body I,I=1,7
(1878+I)	(I,32)	Number of these holes over wing
(1885+I)	(I,33)	Length of each hole
(1892+I)	(I,34)	Width of each hole
(1899+I)	(I,35)	Depth of each hole
(1906+I)	(I,36)	x/1 where first hole starts
(1913+I)	(I,37)	x/1 where last hole starts
(1921 + I)	(I,38)	Type hole
(1928-1941)	(I,39)-(I,40)	Vacant
(1941+I)	(I,41)	Total number of waves in Body I,I=1,7
(1948+I)	(I,42)	Number of these waves over wing
(1955+I)	(I,43)	Length of each wave
(1962+I)	(I,44)	Width of each wave
(1969+I)	(I,45)	Amplitude of each wave
(1976+I)	(I,46)	x/1 where first wave starts
(1983+I)	(I,47)	x/1 where last wave starts
(1990 -2001)	(I,48)-(I,50)	Vacant
(2001+I)	(I,51)	Number of protuberances on BodyI,I=1,7
(2018+I)	(I,52)	Height of each protuberance
(2025+I)	(I,53)	Parasite area of each protuberance, Δf
(2032+I)	(I,54)	x/1 of first protuberance
(2039+I)	(I,55)	x/1 of last protuberance.
(2047-2081)	(I,56) - (I,60)	Vacant

(Table 1 Continued)

A( )	VARIABLE NAME	DESCRIPTION
	(Input Data Block #10 (Body Damage Parameters), Continued)	
(2081+I)	DBOD(I,61)	Frontal area of blunted body at point of damage
(2088+I)	(I,62)	l/d at point of damage
(2096-2151)	(I,63)-(I,70)	Vacant
(2152-2361)	(I,71)-(I,100)	Reserved for additional modes of body damage

Input Data Block #11 (Surface Damage Parameters)

A( )	VARIABLE NAME	DESCRIPTION
(2361+I)	DSUR(I,1)	Roughness factor before damage on surface I,I=1,7
(2368+I)	(I,2)	Roughness factor after damage on surface I,I=1,7
(2375+I)	(I,3)	x/c where damage starts
(2382+I)	(I,4)	x/c where damage ends
(2389+I)	(I,5)	Fraction of area affected by damage
(2396+I)	(I,6)	Lower surface (0) or upper surface (1)
(2404-2431)	(I,7)-(I,10)	Vacant
(2431+I)	(I,11)	Number of forward-facing steps on surface I, lower surface
(2438+I)	(I,12)	Number of forward facing steps on surface I, upper surface
(2445+I)	(I,13)	Width of each step
(2452+I)	(I,14)	Height of each step
(2459+I)	(I,15)	x/c at first step
(2466+I)	(I,16)	x/c at last step
(2474-2571)	(I,17)-(I,20)	Vacant
(2501+I)	(I,21)	No. of aft-facing steps on surface I, lower surface
(2508+I)	(I,22)	No. of aft facing steps on surface I, upper surface
(2515+I)	(I,23)	Width of steps
(2522+I)	(I,24)	Height of steps
(2529+I)	(I,25)	x/c at first step
(2536+I)	(I,26)	x/c at last step
(2544-2571)	(I,27)-(I,30)	Vacant

(Table 1 Continued)

A( )	VARIABLE NAME	DESCRIPTION
		(Input Data Block #11 (Surface Damage Parameters), Continued)
(2571+I)	DSUR(I,31)	No. of holes on surface I.lower surface
(2578+I)	(I,32)	No. of holes on surface I,upper surface
(2585+I)	(I,33)	Length of each hole
(2592+I)	(I,34)	Width of each hole
(2599+I)	(I,35)	Depth of each hole
(2606+I)	(I,36)	x/c where first hole starts
(2613+I)	(I,37)	x/c where last hole starts
(2620+I)	(I,38)	Type hole
(2627±I)	(I,39)	Porosity factor
(2634+I)	(I,40)	Vacant
(2641+I)	(I,41)	No. of waves on surface I, lower surface
(2648+I)	(I,42)	No. of waves on surface I, upper surface
(2655+I)	(I,43)	Length of each wave
(2662+I)	(I,44)	Width of each wave
(2669+I)	(I,45)	Amplitude of each wave
(2676+I)	(I,46)	x/c where first wave starts
(2683+I)	(I,47)	x/c where last wave starts
(2691-2711)	(I,48) - (I,50)	Vacant
(2711+I)	(I,51)	No. of protuberances on surface I,I=1,7
(2718+I)	(I,52)	Height of each protuberance
(2725+I)	(I,53)	Parasite area of each protuberance, Δf
(2732+I)	(I,54)	x/c of first protuberance
(2739+I)	(I,55)	x/c of last protuberance
(2747-2781)	(I,56) - (I,60)	Vacant
(2785-3061)	(I,61) - (I,100)	Reserved for additional modes of surface damage
(3062)	DWING(1)	$\Delta c/c$ ~ chord of missing leading edge
(3063)	(2)	η_1 ~ inboard edge of missing leading edge
(3064)	(3)	$\Delta\eta$ ~ span of missing leading edge
(3065)	(4)	z/t ~ Ratio of leading-edge to maximum thickness
(3066)	(5)	corner sharpness factor for leading edge
(3067)	(6)	left, right, or both sides, leading edge
(3068)	(7)	$\Delta c/c$ ~ chord of missing trailing edge

(Table 1 Continued)

A( )	VARIABLE NAME	DESCRIPTION
	(Input Data Block #11 (Surface Damage Parameters), Continued)	
(3069)	DWING(I,8)	η_1 ~ inboard edge of missing trailing edge
(3070)	(9)	$\Delta\eta$ ~ span of missing trailing edge
(3071)	(10)	left, right, or both sides, trailing edge
(3072)	(11)	$\Delta\eta$ ~ span of missing wing tip, left
(3073)	(12)	$\Delta\eta$ - span of missing wing tip, right
(3074)	(13)	moment arm of roll trimming device
(3075)	(14)	Fraction of area lost from left H.T.
(3076)	(15)	Fraction of area lost from right H.T.
(3077)	(16)	Fraction of area lost from other surfaces
(3078)	(17)	↓
(3079)	(18)	↓
(3080)	(19)	↓
(3081)	(20)	

*Items marked with an asterisk are computed internally. But, since they logically group with the input variables, they are stored as shown in Table I.

**Optional Input. If zero is input for these parameters, the value to be used will be computed internally.

TABLE 2
Variables In Common Block OUTPUT

B( )	VARIABLE	DEFINITION
(1-21)	CLTAB(21)	Table of C_L
(22-42)	TABMCR(21)	Table of Mach Critical
(43)	CL	C_L Value for Survey
(44)	CD	C_D at C_L
(45)	CM	C_M at C_L
(46)	ALPHA	α at C_L
(47)	CDM	$C_{D_{min}}$
(48)	CDL	Drag Due to Lift
(49)	CDR	Drag Rise at Lift
(50)	CDRO	Drag Rise at Zero Lift
(51)	CLT	Trimmed C_L at α Corresponding to C_L
(52)	CDT	Trimmed C_D at α Corresponding to C_L
(53)	DH	Trim Deflection (Horizontal)
(54)	FK	Polar Shape Factor
(55)	DELCL	Polar Displacement
(56)	CMO	Wing-Body Moment at Zero Lift
(57)	DCMCL	(dC_M/dC_L) Wing-Body
(58)	XACWB	A.C. Location-Wing-Body
(59)	CLA	Lift Curve Slope
(60)	ALO	Zero-Lift α
(61)	R	Leading-Edge Suction
(62)	CMOH	C_M at DH=0
(63)	XH	Moment Arm Length From CG to $\bar{c}/4$ Horizontal Tail
(64)	OMEGA	Angle of Line Between CG and HT A.C.
(65)	XCT	Fuselage Station of CG Location
(66)	EO	Span Efficiency of Wing-Body, R=1
(67)	BMCRB	Body Mach Critical
(68)	CLDE	$C_{L_{\delta e}}$
(69)	AKD	Elevon Drag Factor
(70)	XCP	Elevon Center-of-Pressure
(71)	DE	Elevon Deflection
(72-94)		Vacant
(95)	DEDA	$d\epsilon/d\alpha$
(96)	CLAW	Wing C_{L_α}
(97)	CLAB	Body C_{L_α}
(98)	CLAT	Tail C_{L_α}
(99)	AH	Tail Drag Factor

(TABLE 2, Continued)

B( )	VARIABLE	DEFINITION
(100)	BH	Tail Drag Factor
(101)	AOH	Tail Incidence for Zero Lift
(102)	CH	Tail Drag Factor
(103)	ABREAK	Tail Drag Factor
(104)	CLDH	C_{L_δ}
(105)	CLPB	Polar Break Lift Coefficient
(106)	CLDB	Drag Break (Polar)
(107)	CLMAX	$C_{L_{MAX}}$
(108)	ABRK	α at CLPB
(109)	AMAX	α at $C_{L_{MAX}}$
(110)	DAMAX	$\Delta \alpha_{max}$
(111)	DEL	Lift Curve Parameter
(112)	CLS	Lift Curve Parameter
(113)	ARLO	Aspect Ratio Boundary
(114)	FMCRO	Mach Critical at Zero Lift
(115)	FML1	Mach Limit 1
(116)	FML2	Mach Limit 2
(117)	CDC	Camber Drag
(118)	CLAMCR	C_{L_α} at M_{CRO}
(119)	CLAML2	C_{L_α} at M_{L2}
(120)	CDMCR	Wing Drag at M_{CRO}
(121)	CDML2	Wing Drag at M_{L2}
(122)	CDATM	Wing Drag at M
(123-129)		Vacant
(130)	XACS	Aerodynamic Center at Stall
(131)	CDO	C_D at Zero Lift
(132)	A2	Drag Rise Parameter
(133)	A3	Drag Rise Parameter
(134)	PL	Drag Rise Parameter
(135)	CDMISC	Miscellaneous Drag
(136)		Vacant
(137)	CDAFT	Aft End Upsweep Drag
(138-142)		Vacant
(143)	ALT	Survey Condition
(144)	SPEED	Survey Condition
(145)	SWEEP	Survey Condition

(TABLE 2, Continued)

B( )	VARIABLE	DEFINITION
(146)	JPASS	Program Control Parameter
(147)		Vacant
(148)	RNOFT	RN/FT
(149-153)	TOTCD(5)	Drag Table - Total
(154-188)	CDSUR(I,J)	Drag Table - Surfaces
(189-223)	CDBOD(I,J)	Drag Table - Bodies

TABLE 3
Variables in Common Block CALC

C( )	VARIABLE NAME	DESCRIPTION
(1)	BO2	Wing Semi-Span (b/2)
(2)	DOB	Body Dia./Wing Span (d/b)
(3)	CR	Wing Root Chord at Centerline
(4)	SPLAN	Theoretical Wing Planform Area
(5)	XCRTE	X-Sta. of Theor. Centerline Root Chord (CR) Trailing Edge
(6)	SWPQC	Equivalent Wing Quarter-Chord Sweep
(7)	SWPMC	Equivalent Wing Mid-Chord Sweep
(8)	SWPTE	Equivalent Wing Trailing-Edge Sweep
(9)	SWMT	Equivalent Wing Max. Thickness Sweep
(10)	SEXW	Equivalent Wing Exposed Area
(11)	CBAR	Equivalent Wing Exposed MAC
(12)	TOCW	Equivalent Wing Thickness Ratio
(13)	CLD	Equivalent Wing 2-D Design C_L
(14)	ARXR	Equivalent Wing Exposed Aspect Ratio
(15)	CRX	Chord Length at Wing Root (Exposed)
(16)	CTX	Chord Length at Wing Tip
(17)	XRX	X-Station at LE of Wing Root
(18)	XTX	X-Station at LE of Wing Tip
(19)	YRX	Y-Station at Wing Root
(20)	YTX	Y-Station at Wing Tip
(21)	XB	X-Station of LE of Wing MAC
(22)	YB	Y-Station of LE of Wing MAC
(23)	SWET	Outboard Panel A_{wet}
(24)		Vacant
(25)	SWPLE	Equivalent-Wing LE Sweep
(26)	FOC	Max. Camber Ordinate
(27)	SWPR	Variable-Sweep Angle (Outboard Panel)
(28)	KPASS	Program Control Indicators
(29)	AR	b^2/S_{Plan}
(30)	TR	λ (Equivalent Taper Ratio)
(31)	YIX	Span of Inboard Wing
(32)	SIX	Exposed Area of Inboard Wing
(33)	ARI	Aspect Ratio of Inboard Wing
(34)	CBXP	Root Chord of Equivalent Outboard Wing
(35)	SOXP	Exposed Area of Equivalent Outboard Wing
(36)	AROP	Aspect Ratio of Equivalent Outboard Wing
(37)	DXQC	Equivalent Wing Parameter

(TABLE 3, Continued)

C( )	VARIABLE NAME	DESCRIPTION
(38)	TWIST	Wing Twist
(39)	WINC	Winc Incidence
(40)	XHT	X-Station for HT $\frac{c}{4}$
(41)	TOCS	t/c for Swept Panel
(42)	CBAR2	Exposed CBAR for Swept Panel
(43)	CLDS	CLD for Swept Panel
(44)	YOX	Span of Outboard Swept Panel
(45)	SWPMCS	Mid Chord Sweep of Outboard Swept Panel
(46)	DA1	Correlation Factor for Variable Sweep
(47)	DAC1	Correlation Factor for Variable Sweep
(48)	DA2	Correlation Factor for Variable Sweep
(49)	DAC2	Correlation Factor for Variable Sweep
(50)	DTOC	Correlation Factor for Variable Sweep

TABLE 4
Variables in Common Block NUCOUT

D( )	VARIABLE	DESCRIPTION
(1-119)	DAMCD(I,K)	$\Delta C_{D_{min}}$ on Each Component Due to Damage I=Body or Surface No. I=1,7 K=Mode No. K=1,17
(120-238)	SYM(I,K)	Indicates Symmetrical Damage if SYM (I,J)=1.0 I=Body or Surface No. I=1,7 K=Mode No. K=1,17
(239)	DCDMIN	Total $\Delta C_{D_{min}}$ Due to Damage
(239+I)	ALP(I)	Undamaged Aircraft α I=1,21 (applies through (449+I))
(260+I)	CL(I)	Undamaged Aircraft C_L
(281+I)	CD(I)	Undamaged Aircraft C_D
(302+I)	CM(I)	Undamaged Aircraft C_M
(323+I)	CLT(I)	Undamaged Aircraft Trimmed C_L
(344+I)	CDT(I)	Undamaged Aircraft Trimmed C_D
(365+I)	CLP(I)	Damaged Aircraft C_L
(386+I)	CDP(I)	Damaged Aircraft C_D
(407+I)	CMP(I)	Damaged Aircraft C_M
(428+I)	CLTP(I)	Damaged Aircraft Trimmed C_L
(449+I)	CDTP(I)	Damaged Aircraft Trimmed C_D
(470+I)	RCLA(I)	Increment in C_L That Produces Rolling Moment, Caused by: I=1 Missing LE I=2 Missing TE I=3 Missing Left Wing Tip I=4 Missing Right Wing Tip I=5 Missing Left H.T. I=6 Missing Right H.T.
(477)	DXAC	Shift in Wing-Body A.C. (Δ X/C)
(477+I)	CROLL(I)	Rolling Moment Coefficient Caused by Asymmetric Wing or Tail Damage. Corresponds to ALP(I), I=1,21.
(498+I)	CDRT(I)	Drag Due to Trimming Out the Rolling Moment, I=1,21.

4.4 PROGRAM LISTING

A complete listing of the program is shown in this section. The subprograms and subroutines appear in the order in which they occur within the program.

```
      OVERLAY (OVLY,0,0)
      PROGRAM MAIN(INPUT=514,OUTPUT=514,TAPE5=INPUT,TAPE6=OUTPUT,
     1             TAPE10=514,TAPE21)
C
C     EMPIRICAL AIRCRAFT AERODYNAMIC PREDICTION PROCEDURE
C
      COMMON/INPUT/  A(3081)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/    C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ E(906)
      COMMON /BLKTIL/ TITLE(6)
      COMMON /BLKPRT/ KPRINT(50)
      COMMON /BLKOV1/ IJ
C
   15 IJ      = 0
      WRITE (6,2001)
 2001 FORMAT(5X,15H ENTER INPUT    )
   20 CALL OVERLAY(4HOVLY,1,0)
C
      IF (IJ.EQ.2) GO TO 30
C
      WRITE (6,2002)
 2002 FORMAT(5X,15H ENTER GEOMETRY)
      CALL OVERLAY(4HOVLY,2,0)
C
      WRITE (6,2003)
 2003 FORMAT(5X,15H ENTER SURVEY   )
      CALL OVERLAY(4HOVLY,3,0)
C
      GO TO 20
C
30    WRITE (6,2004)
 2004 FORMAT(5X,15H ENTER NUCDAM   )
      CALL OVERLAY(4HOVLY,4,0)
C
      REWIND 10
      READ (10)
      READ (10) B
      GO TO 20
C
      END
```

```
      SUBROUTINE LNTP(XBAR,YBAR,X,Y,M,NC)
C
C     LAGRANGIAN INTERPOLATION
C           1) INCREASING OR DECREASING X ARRAY
C           2) LINEAR EXTRAPOLATION ONLY
C           3) NC = NUMBER OF POINTS USED IN INTERPOLATION
C           4) M  = TOTAL NUMBER OF POINTS IN X ARRAY
C           5) IF(NC.LE.0.OR.M.LE.0) YBAR = 0.0
C           6) IF(M.EQ.1.AND.NC.GT.0) YBAR=Y(1)
C           7) IF(NC.EQ.1.AND.M.GT.0) YBAR=Y(NEAREST X(I))
C           8) IF(NC.GT.M) INTERPOLATION AS IF NC=M
C
      DIMENSION  X(M) , Y(M)
C
      YBAR=0.0
      N=NC
      IF(N.GT.M) N=M
      IF(M.LE.0.OR.N.LE.0) RETURN
      IF(M.GT.1) GO TO 10
      YBAR=Y(1)
      RETURN
C
   10 MDC=0
      IF(X(M).LT.X(1)) MDC=1
      DO 20 I=2,M
      I1=I-1+MDC
      I2=I-MDC
      IF(XBAR.GE.X(I1).AND.XBAR.LE.X(I2)) GO TO 40
   20 CONTINUE
      I1=1
      IF((MDC.EQ.0.AND.XBAR.GE.X(M)).OR.(MDC.NE.0.AND.XBAR.LE.X(M)))
     1   I1=M-1
      I2=I1+1
      GO TO 45
C
   40 IF(N.GT.2) GO TO 60
   45 IF(N.LT.2) GO TO 55
   50 YBAR=Y(I1)+(XBAR-X(I1))*(Y(I2)-Y(I1))/(X(I2)-X(I1))
      RETURN
C
   55 YBAR=Y(I1)
      IF(ABS(XBAR-X(I2)).LT.ABS(XBAR-X(I1))) YBAR=Y(I2)
      RETURN
C
   60 J=I-N/2
      IF(I.LE.N/2+1) J=1
      IF(I.GT.M-N/2) J=M-N+1
      NT=N+J-1
C
      DO 80 I=J,NT
      ELL=1.0
      DO 70 K=J,NT
      IF(K.NE.I) ELL=ELL*(XBAR-X(K))/(X(I)-X(K))
   70 CONTINUE
   80 YBAR=YBAR+ELL*Y(I)
      RETURN
C
      END
```

```
      FUNCTION DLNT (XBAR,YBAR,X,Y,F,NX,NY,NXMAX,LOX,LOY)
C
C     DOUBLE LAGRANGE INTERPOLATION
C
      DIMENSION X(1),Y(1),F(NXMAX,1),FT(25)
C
      FBAR=0.0
      IF(LOY.GT.25) LOY=25
      LLX=LOX
      IF(LOX.GT.NX) LLX=NX
      IF(NY.GT.1) GO TO 10
      CALL LNTP (XBAR,FBAR,X,F(1,1),NX,LLX)
      GO TO 30
C
   10 LLY=LOY
      IF(LOY.GT.NY) LLY=NY
      DO 20 I=1,NY
      CALL LNTP (XBAR,FT(I),X,F(1,I),NX,LLX)
   20 CONTINUE
      CALL LNTP (YBAR,FBAR,Y,FT,NY,LLY)
C
   30 DLNT=FBAR
C
      RETURN
      END
```

```
      SUBROUTINE TLNT(XBAR,YBAR,ZBAR,FBAR,X,Y,Z,F,NX,NY,NZ,NXMAX,NYMAX)
C
C         TRIPLE LINEAR INTERPOLATOR
C
      DIMENSION X(1),Y(1),Z(1),F(NXMAX,NYMAX,1),FT(25)
C
      DO 10  I=1,NZ
C
   10 FT(I)   = DLNT(XBAR,YBAR,X,Y,F(1,1,I),NX,NY,NXMAX,2,2)
C
      CALL LNTP(ZBAR,FBAR,Z,FT,NZ,2)
C
      RETURN
      END
```

```
      SUBROUTINE LSPCF( X,Y,N,MP,SA,RE,SIGMA,IT)
C
C     LEAST SQUARES POLYNOMIAL CURVE FIT SUBROUTINE
C
      DIMENSION       A( 8,8),ST(8),R(8),T(  8),Y(N),RE(N),SA(8),X(N)
     1          ,XMAX(7),B(8,3)
      EQUIVALENCE (R, ST, B)
C
      II=MP+1
      J=1
      DO 10 I=2,N
      IF (ABS(X(J)).GT.ABS(X(I)))GO TO 10
      J=I
   10 CONTINUE
      DO 6 M1 = 1,II
      DO 7 M2 = 1,II
    7 A(M1,M2)= 0.
    6 CONTINUE
      XMAX(1)=ABS(X(J))
      DO 9 I=2,7
    9 XMAX(I)=XMAX(1)*XMAX(I-1)
      DO 8 I=1,II
    8 R(I) =0.
      T(1)=1.0
      DO 15 L=1,N
      T(2)=X(L) / XMAX(1)
      DO 14 K=2,MP
   14 T(K+1)=2.0*T(2)*T(K)-T(K-1)
      DO 15 I=1,II
      R(I)=Y(L)*T(I)+R(I)
      DO 15 K=1,II
   15 A(K,I)=A(K,I)+T(K)*T(I)
      DO 13 L=2,II
      LM1=L-1
      DO 12 K=1,LM1
   12 A(K,L)=A(L,K)
   13 CONTINUE
      II=0
      GO TO 100
   16 CONTINUE
C
      IF (N .LE. MP) IT = 142
      IF (MP .GT. 7) IT = 198
      IF (XMAX(1).EQ.0.) IT =236
      MR=MP
      IF((MP.LT.1).OR.(MP.GT.7)) CALL EXIT
      GO TO (20,21,22,23,24,25,26),MR
   20 SA(1)=ST(2)/XMAX(1)
      SA(2)=ST(1)
      GO TO 27
   21 SA(1)=2.*ST(3)/XMAX(2)
      SA(2)=ST(2)/XMAX(1)
      SA(3)=ST(1)-ST(3)
      GO TO 27
   22 SA(1)=4.*ST(4)/XMAX(3)
      SA(2)=2.*ST(3)/XMAX(2)
      SA(3)=(ST(2)-3.*ST(4))/XMAX(1)
      SA(4)=ST(1)-ST(3)
      GO TO 27
   23 SA(1)=8.*ST(5)/XMAX(4)
      SA(2)=4.*ST(4)/XMAX(3)
      SA(3)=(2.*ST(3)-8.*ST(5))/XMAX(2)
      SA(4)=(ST(2)-3.*ST(4))/XMAX(1)
      SA(5)=ST(1)-ST(3)+ST(5)
      GO TO 27
   24 SA(1)=16.*ST(6)/XMAX(5)
      SA(2)= 8.*ST(5)/XMAX(4)
      SA(3)=(4.*ST(4)-20.*ST(6))/XMAX(3)
      SA(4)=(2.*ST(3)-8.*ST(5))/XMAX(2)
      SA(5)=(ST(2)-3.*ST(4)+5.*ST(6))/XMAX(1)
      SA(6)=ST(1)-ST(3)+ST(5)
      GO TO 27
   25 SA(1)=32.*ST(7)/XMAX(6)
      SA(2)=16.*ST(6)/XMAX(5)
      SA(3)=(8.*ST(5)-48.*ST(7))/XMAX(4)
      SA(4)=(4.*ST(4)-20.*ST(6))/XMAX(3)
      SA(5)=(2.*ST(3)-8.*ST(5)+18.*ST(7))/XMAX(2)
```

```
      SA(6)=(ST(2)-3.*ST(4)+5.*ST(6))/XMAX(1)
      SA(7)=ST(1)-ST(3)+ST(5)-ST(7)
      GO TO 27
   26 SA(1)=64.*ST(8)/XMAX(7)
      SA(2)=32.*ST(7)/XMAX(6)
      SA(3)=(16.*ST(6)-112.*ST(8))/XMAX(5)
      SA(4)=(8.*ST(5)-48.*ST(7))/XMAX(4)
      SA(5)=(4.*ST(4)-20.*ST(6)+56.*ST(8))/XMAX(3)
      SA(6)=(2.*ST(3)-8.*ST(5)+18.*ST(7))/XMAX(2)
      SA(7)=(ST(2)-3.*ST(4)+ 5.*ST(6)-7.*ST(8))/XMAX(1)
      SA(8)=ST(1)-ST(3)+ST(5)-ST(7)
   27 DO 29 K=1,N
      SIM=SA(II)
      DO 28 I=1,MP
      IIMI=II-I
      SIM=SIM+SA(IIMI)*(X(K))**I
   28 CONTINUE
                    RE(K)=SIM-Y(K)
   29 CONTINUE
      SIG=ABS(RE(1))
      DO 30 I=2,N
   30 SIG=SIG+ABS(RE(I))
      DEAN=FLOAT(N)
      SIGMA=(SIG*1.7724538)/(DEAN*1.414214)
      RETURN
C
 100  CONTINUE
      N      = II
      M      = 1
      DET    = 0.0
C
C     SOLUTION OF MATRIX A*X = B
C       N  NO. OF ROWS AND COLUMNS IN MATRIX A
C       M  NO. OF COLUMNS IN MATRIX B
C     IA  NO. OF ROWS IN ARRAY A
C     IB  NO. OF ROWS IN ARRAY B
C     DET SCALED VALUE OF DETERMINANT
C     IER  =0 FOR NO ERROR, =1 FOR SINGULAR MATRIX
      DO 240 I=1,N
      AMAX = A(I,1)
      DO 210 J=1,N
      IF (ABS(A(I,J)) .GT. ABS(AMAX)) AMAX = A(I,J)
  210 CONTINUE
      DET = DET * AMAX
      IF (AMAX .EQ. .0) GO TO 180
      DO 220 J=1,N
  220 A(I,J) = A(I,J) / AMAX
      DO 230 L=1,M
  230 B(I,L) = B(I,L) / AMAX
  240 CONTINUE
      NT = N-1
      IF (NT)    50,50,60
   50 IER = 0
      GO TO 16
   60 DO 140 I=1,NT
      K6 = I +1
      K = I
      AMAX = A(I,I)
      DO  70 J=K6,N
      IF (ABS(A(J,I)) .LT. ABS(AMAX)) GO TO 70
      AMAX = A(J,I)
      K=J
   70 CONTINUE
      DET = DET * AMAX
      IF (K .EQ. I) GO TO 200
      DET = -DET
      DO  80 J=I,N
      SAV = A(I,J)
      A(I,J) = A(K,J)
      A(K,J) = SAV
   80 CONTINUE
      DO  90 L=1,M
      SAV = B(K,L)
      B(K,L) = B(I,L)
      B(I,L) = SAV
   90 CONTINUE
  200 DO 130 K=K6,N
```

```
      IF (A(I,I) .EQ. 0.) GO TO 180
      IF (A(K,I) .EC. 0.) GO TO 130
      ERAS =-A(K,I) / A(I,I)
      DO 110 J=K6,N
  110 A(K,J) = ERAS * A(I,J) + A(K,J)
      DO 12C L=1,M
  120 B(K,L) = ERAS * B(I,L) + B(K,L)
  130 CONTINUE
  140 CONTINUE
      DET = DET * A(N,N)
      IF (A(N,N) .EC. 0.) GO TO 180
      NX = N
  150 DO 160 L=1,M
  160 B(NX,L) = B(NX,L) / A(NX,NX)
      IF (NX .LE. 1) GO TO 5C
      NT = NX-1
      DO 170 J=1,NT
      DO 170 L=1,M
  170 B(J,L) = B(J,L) - B(NY,L) * A(J,NX)
      NX = NX -1
      GO TO 150
  180 IER =1
      GO TO 16
      END
```

```
      SUBROUTINE TCRG(DCLT,DCDT)
C
C     COMPUTES EFFECT OF TRIM
C
      COMMON /INPUT/ A(2081)
      COMMON /CALC/  C(50)
      COMMON /OUTPUT/ CLTAP(21), TAPMCR(21),
     1                CL, CD, CM, ALPHA, CDM, CDL, CDP, CDPC, CLT, CDT,
     2                DH, FM, DELCL, CMC, DCMCL, XACWB, CLA, ALC,
     3                R,CMCH,XH,OMEGA,XCG,EC,FMCPE,CLDE,AKD,XCP,DE,
     3                B1(24),
     4                CLAW,CLAB,CLAT,AH,BH,HSTAP,CH,APPEAK,CLDH,
     5                CLPB,    CLDB, CLMAX, C2(7), FML1,FML2, CDC,
     6                C3(12), XACS, C4(6), CDAFT, C5(5),
     7                ALT,SPEED,SWEEP,JPASS, P(77)
C
      EQUIVALENCE (XLE1,A(254)), (CMAC,A(12)), (CRX,C(15)),
     1            (ITRIM,A(22)), (NHT,A(4))
C
C
      DCLT = 0.0
      DCDT = 0.0
C
      CGDCR   = (XCG -XLE1)/CRX
      XAC     = XACWB
C
C     A.C. SHIFTS TO POSITION FOR STALLED WING ABOVE CLDB
C
      IF( CL.GT.CLDB ) XAC = XACWB + (XACS - XACWB) * ((CL-CLDB)/
     1                                  (CLMAX-CLDB))**2
      IF( CL.GT.CLMAX ) XAC = XACS
      IF( SPEED.GE.1.0 ) XAC = XACWB
      DCMCL   = (CGDCR - XACWB) * CRX/CMAC
      DH     = 0.0
C
      CLTAIL = CLAT * (ALPHA - HSTAP) + CLDH * DH
      CLWB    = CL - CLTAIL
      XLT     = XH *COS(OMEGA - ALPHA/57.3) /CMAC
C
      CM      = CMO + (CGDCR - XAC) *CRX/CMAC * CLWB - CLTAIL * XLT
      CMOH  =  CM
C
      IF( CLDH.EQ.0.0.OR.XLT.EQ.0.0 ) GO TO 60
      IF (ITRIM.EQ.1) GO TO 60
      DH      = CM/(CLDH * XLT)
      CM      = 0.0
C
   60 CONTINUE
      DE      = 0.0
      IF (NHT.GT.0) GO TO 70
      IF (ITRIM.EQ.1) GO TO 70
      XCLDE = (CGDCR * CRX/CMAC -XCP) *CLDE
      IF (XCLDE.EQ.0.0) GO TO 70
      DE      = CM/XCLDE
      CM      = 0.0
C
 70   CONTINUE
C
      IF( DH.EQ.0.0 ) GO TO 100
      DCLT    = DH*CLDH
C
      DCDT    = AH*DH**2 + BH*DH*(ALPHA -HSTAP)
      IF(ALPHA.GT.APPEAK) DCDT = DCDT  + CH*DH*(ALPHA-APPEAK)
C
 100  IF (DE.EQ.0.0) GO TO 200
C
      DCLT   = DCLT + CLDE * DE
      DCDT   = DCDT + AKD * DE**2
C
 200  RETURN
      END
```

```
      SUBROUTINE AER2(SPEED, CLA)
C
C     LIFT CURVE SLOPE
C
      COMMON /BLKCLA/ SPLAN, TOC, TAPER, ARWS, DMSTR, CLD, DM1, DOB,
     1                EPSL, SWPMC, SREF
      COMMON /BLKPRT/ KPRINT(50)
C
C
C
      COSZ  = COS(SWPMC)
      PI    = 3.14159
      TWOD   = 1.0
      IF( SPEED.LT.1.0 ) TWOD = 1. + EPSL/SQRT(1.-SPEED**2)
C
C
      CSUBO  = (10.0 + 0.91 * ARWS**3)/(10.0 + ARWS**3)
      ZMSTR  = CSUBO + (1.0 -CSUBO) * (1.0 - COSZ)**2
      ZMSTRO  = ZMSTR
      ZMSTR = ZMSTR + DMSTR
      ZM1    = 1.0 -2.0 * TOC * (ARWS**3/(4.0 +ARWS**3))
     1          * COSZ**1.5* (1.0 + 1.5*CLD**1.5)
      ZM10     = ZM1
      IF( ZM10.GT.ZMSTRO ) ZM10= ZMSTRO
      ZM1    = ZM10 + DM1
      IF( ZM1.GT.ZMSTR ) ZMSTR = ZM1
      ZM2    = ZM1 + TOC + DM1*0.5
      ZM30     = 1.0 + TOC
      ZM3      = ZM30
      IF( ZM2.GT.1.0 ) ZM3 = ZM2 + TOC
C
      SIG1=0.0
      SIG2=0.0
      IF( ZM2.NE.ZM1)  SIG1 = 0.5*((SPEED -ZM1)/(ZM2 -ZM1))
      IF( ZM3.NE.ZM2)  SIG2 = 0.5*(1.0 + (SPEED -ZM2)/(ZM3 -ZM2))
C
      TOCL    = 1.0/(4.4*ARWS*COSZ**1.5    )
      DTOCL   =(TOC - TOCL)/COSZ
      IF( DTOCL.LT.0.0) DTOCL = 0.0
      ARDT  = ARWS * DTOCL
      IF( DTOCL.GT.0.07) DTOCL = 0.07
      IF( ARDT.GT.0.1 ) ARDT = 0.1
      GAMMA = 9.0 * (DTOCL/(1.0 + 0.5 * ARDT) )
C
      GAMAO    = GAMMA
      IF( ZM30.NE.ZM10 ) GAMMA = GAMAO *(ZM3-ZM1)/(ZM30-ZM10)
      IF (GAMMA.GT.GAMAO ) GAMMA = GAMAO
C
      XF      = (16.0 +3.0*ARWS**2)/(8.0 +5.0*ARWS**2)
      XKB     =(1.0 +DOB)*(1.0 -DOB)**XF
      XKT    = 1.0
      IF(SPEED.GE.ZM1.AND.SPEED.LE.ZM2) XKT = 1.0 -(4.0*SIG1*(1.0
     1                          - SIG1))**3 * GAMMA
      IF(SPEED.GT.ZM2.AND.SPEED.LE.ZM3) XKT = 1.0 -(4.0*SIG2*(1.0
     1                          - SIG2))**3 * GAMMA
    5 CONTINUE
C
      Y1    = 2.0 + 0.66667 * SQRT(TAPER) - TAPER**2
      Y     = (1.0 + PI *ARWS)/(3.0 + PI * ARWS)   * Y1
      TWOS    = 1.0
      IF( ZMSTR.LT.1.0 ) TWOS = 1. + EPSL/SQRT(1.-ZMSTR**2)
C
      CLAO    = (TWOS * PI * ARWS)/(TWOS + SQRT(TWOS + (1.0 -
     1           COSZ**1.3334) * (ARWS/(2.0*COSZ))**2 ) )
      IF( SPEED.GT.0.0 ) BETAP = (SPEED - ZMSTR) * (1.0 +
     1                 (ZMSTR/SPEED)**Y )**2
      Z1     = PI * ARWS / CLAO
      Z1     = 3.0 * Z1 * ( Z1 - 1.0 ) * COSZ**0.6667
      Z      = ZMSTR * CLAO   + ARWS**2/Z1
C
C
      IF(SPEED.GT.ZMSTR)    GO TO 10
C
      CLAB    = (.0548311*TWOD*ARWS)/(TWOD + SQRT(TWOD + (1.0 -
     1          COSZ**1.3334 *(SPEED/ZMSTR)**2.667)*(ARWS/(2.*COSZ))**2
     2                                                              ))
      GO TO 20
```

```
   10 CLAB   = 1.0/(57.3 *((ZMSTR/SPEED)**2 /CLAO  + BETAP/4.0) )
C
C
C
C
   20 CONTINUE
      CLA      = CLAB * XKT * YKB * SPLAN/SPEF
C
C
C
   25 IF( KPRINT(11).EQ.0 ) GO TO 30
      WRITE(6,1000) SPEED, CLA, CLAB, XKT, XKB, SPLAN, SPEF,
     1               ZMSTR, ZM1, ZM2, ZM3, CCSZ, CSUBO, AEDT,
     2               TCC, TAPER, AEWS, DESTR, CLD, DM1, CDB, TWCD, SWPMC
 1000 FORMAT(10X*AER2 DUMP*, 6F15.5 /(15X,6F15.5) )
C
   30 RETURN
      END
```

```
      BLOCK DATA
C
      COMMON /BLKDAT1/XSWPL(11), YPMIN(11),
     $                AAA(22), BBB(22), CCC(22), DDD(22), XTT(22),
     $                BCT(12),ATSW(6),TPPT(6),FXAC1(216),FXAC2(216),
     $                XBD4(5), YBD4(5), ZBD4(3), FBD4(5,5,3),
     $                XDYO(6), YXMN(6), ZIDO5(6,6),
     $                XBD6(4), YBD6(4), ZBD6(6), FBD61(4,4,6),
     $                FBD62(4,4,6), FBD63(4,4,6),
     $                MAP, TRANS, DY, AMAP(22), BMAP(11),
     $                XIN13(10), YIN13(7), ZCUT13(10,7), ZCUT14(10,7),
     $                NYIN15, XIN15(6), NYIN15, YIN15(5),
     $                FCUT15(6,5), FCUT16(6,5),
     $                X1(6), Y1(6), X9(8), Y9(8), XAP(5), YDCL(5),
     $                XSWP(7), YTP(4), FEP25(7,4), FEP7(7,4),
     $                XCLDP(7), YAKB(7)
C
      COMMON /BLKDAT2/XTP(6), YC1(6), YC2(6), XSWP1(4), YA(4),
     $                YP(4), XDY(8), XM(4), CTAP(8,4), CTAB(8,4),
     $                XXCLM(13), YYDY(6), FCLMX(13,6), XYC2(9),
     $                YYMACH(5), FDCMX(9,5),XDY1(9),  YXMT(4),
     $                 ZC1MAX(9,4), XDY2(6), YFCC(6), ZDC1M(6,6),
     $                Z2DC1M(6,6),
     $                XSP(8), YDYA(6), FDA(8,6), XAR(6), YCD(8),
     $                FKVDFY(6,8), XANG(10), YFTCC(7), FRA(10,7),
     $                XC2(6),YAST(9),FCAM(6,9),XCT(6),YM(5),FCAM2(6,5),
     $                XXXCLM(9), FCLMXY(9,6),
     $                XY(12), XY(7), XF(12,7)
C
      COMMON /BLKDAT3/XMACH3(17),   COMK3(17),
     1                XLDH3(6)   ,  HDX3(5)   ,     CDC3(6,5)   ,
     2                XCC3(6)    ,  CLD3(4)   ,     XM3(6,4)    ,
     3                YMACH3(13),   CPF3(13)  ,
     4                ZMACH3(13),   CPA3(13)  ,
     5                XLDH4(12)  ,  HDX4(7)   ,     CDD4(12,7)  ,
     6                RATTG3(6)  ,  LMACH(10),      DRAG3(6,10),
     7                XLDD(6)    ,  VMACH(6)  ,     DRAG4(6,6)  ,
     8                DETA1(6)   ,    AR1(5)  ,     CLA1(6,5)   ,
     9                DETA2(6)   , SWPCC2(5)  ,     CLA2(6,5)   ,
     1                ETA13(6)   ,    CLA3(6) ,
     2                DETA4(6)   ,    AR4(5)  ,     CLA4(6,5)   ,
     3                DETA5(6)   , SWPCC5(5)  ,     CLA5(6,5)   ,
     4                ETA16(6)   ,    CLA6(6) ,
     5                DEGB(6)    ,    CCDC(6) ,     CLACLA(6,6)
C
      COMMON /BLKDAT4/DETA11(6)  ,    AR11(5) ,     XAC1(6,5)   ,
     1                DETA12(6)  , SWCC12(5)  ,     XAC2(6,5)   ,
     2                ETA113(6)  ,                  XAC3(6)     ,
     3                DETA14(6)  ,    AR14(5) ,     XAC4(6,5)   ,
     4                DETA15(6)  , SWCC15(5)  ,     XAC5(6,5)   ,
     5                ETA116(6)  ,                  XAC6(6)     ,
     6                DCDC7(5)   ,                  XAC7(5)     ,
     7                DCDC8(5)   ,                  XAC8(5)
     8               ,FLD1(6)    ,                  CLAH1(6)    ,
     9                FLD2(6)    ,                  FFH2(6)     ,
     1                ZAR(10)    ,                  YCLA(10)
C
      COMMON /BLKDAT5/ SPAN(6),    YTPP(4),   FKE(6,4),    FKD(6,4),
     1                 SPAN2(11),YTPP2(5),  FKSW(11,5),
     2                 SPAN3(6), YTPP3(4),  FKM(6,4),
     3                 SPAN4(5), YBFI1(5),  FKA(5,5),
     4                 SPAN5(5), YBFI2(3),  FKF(5,3)
C
C
C
      DATA XSWPL / 0.0,.17453,.34906,.5236,.69813,.87266,1.04719,
     1             1.13446,1.22173,1.34390,1.57079 /,
     2     YPMIN / 0.53, 0.53, 0.52, 0.515, 0.505, 0.49, 0.45,
     3             0.39, 0.30, 0.12, 0.10 /
C
      DATA BOT   / 0.0, .4, .8, 1.2, 1.6, 1.8, 2., 2.4, 2.8, 3.2,3.6,4./,
     1     ATSW  / 1.,  2., 3., 4., 5., 6. /,
     2     TPPT  / 0., 0.2, 0.25, 0.33, 0.5, 1.0 /
C
C     FIG. 4.1.4.2-22    DATCOM           ****
      DATA FXAC1/.25,.24,.23,.22,.20,.18,.17,.25,.32,.37,.40,.42,
     1           7*0.335, .39, .44, .48, .495, .5,
```

```
     2        .42, .44, .45, .46, .465,.475,.50, .56, .56, .57,2*.58,
     3  .50, .53, .55, .575,.60, .62, 6*0.67,
     4  .58, .62, .66, .695,.73, .755,.83, .78, .78, .765,.75, .75,
     5  .68, .72, .76, .79, .83, .765,.98, .90, .90, .87, .85, .84,
     6  .29, .27, .25, .25, .23, .215,.20, .28, .23, .39, .44, .46,
     7  .40, .41, .41, .405,.40, .40, .40, .48, .52, .54, .56, .57,
     8  .51, .53, .54, .555,.57, .58, .60, .65, .665,.66, .69, .69,
     9  .64, .66, .68, .70, .73, .76, .795,.80, .81, .815,.805,.80,
     A  .75, .77, .80, .82, .87, .91, .97, .955,.95, .96, .92, .92,
     B  .87, .895,.915,.945,1.0, 1.05,1.15,1.12,1.1, 1.1, 1.05,1.02,
     C  .30, .29, .275,.26, .245,.24, .23, .30, .35, .40, .45, .475,
     D  .41, .42, .425,.425,.42, .415,.41, .50, .54, .57, .595,.60,
     E  .54, .555,.57, .58, .60, .61, .63, .68, .69, .71, .72, .72,
     F  .67, .69, .71, .73, .76, .79, .82, .83, .845,.865,.85, .845,
     G  .80, .82, .84, .86, .90, .95, 1.02,1.01,1.0, 1.01,.98, .97,
     H  .93, .95, .97, .99, 1.05,1.1, 1.25,1.2, 1.16,1.17,1.11,1.1/
      DATA FXAC2 /
     I  .32, .31, .30, .28, .25, .24, .23, .31, .37, .42, .475,.50,
     J  .45, .46, .46, .455,.45, .445,.44, .54, .57, .60, .63, .64,
     K  .595,.60, .61, .62, .64, .65, .665,.72, .74, .76, .78, .775,
     L  .73, .745,.765,.78, .80, .83, .86, .89, .90, .93, .92, .91,
     M  .89, .90, .91, .93, .96, 1.0, 1.07,1.08,1.08,1.1, 1.06,1.05,
     N  1.04,1.05,1.06,1.09,1.12,1.17,1.27,1.26,1.25,1.26,1.21,1.2,
     O  .36, .34, .325,.31, .30, .275,.25, .34, .41, .48, .52, .55,
     P  .52, .52, .52, .52, .51, .505,.50, .60, .63, .67, .71, .72,
     Q  .70, .70, .705,.71, .72, .735,.75, .80, .82, .87, .89, .89,
     R  .88, .885,.89, .90, .915,.94, .98, 1.0, 1.02,1.07,1.05,1.05,
     S  1.04,1.05,1.05,1.06,1.09,1.12,1.19,1.2, 1.22,1.26,1.22,1.21,
     T  1.2, 1.21,1.22,1.24,1.27,1.3, 1.38,1.4, 1.42,1.45,1.4, 1.38,
     U  .50, .48, .46, .44, .40, .38, .34, .42, .50, .60, .70, .76,
     V  .75, .75, .745,.74, .72, .70, .68, .72, .80, .88, .97, 1.0,
     W  5*1.0, .98,.96,1.01,1.08,1.19,1.22,1.22,
     X  5*1.24,1.23,1.2,1.29,1.38,1.48,1.46,1.49,
     Y  5*1.49,1.47,1.44,1.55,1.66,1.77,1.73,1.75,
     Z  3*1.74,1.73,1.71,1.70,1.68,1.81,1.96,2.06,1.98,2.0 /
C
C   ** FIGURE  4.1.4.1-2 , DATCOM   ******
      DATA   XBD4 / 0.0, .34907, .69813, .87267, 1.0472 /,
     1       YBD4 / 0.0, 1.5, 3.5, 6.0, 10.0 /,
     2       ZBD4 / 0.0, 0.5, 1.0 /,
     3       FBD4 / 5*0.0, 0.0, 0.0, 0.0, -.0002, -.0005,
     4              0.0, -.0007, -.002, -.0027, -.0037,
     5              0.0, -.0019, -.0044, -.006, -.008,
     6              0.0, -.008, -.016, -.021, -.0265,
     7              5*0.0, 0.0, 0.0, -.0005, -.0008, -.0011,
     8              0.0, -.001, -.0029, -.0045, -.0068,
     9              0.0, -.002, -.007, -.0093, -.013,
     1              0.0, -.008, -.016, -.021, -.0265,
     2              5*0.0, 0.0, -.0002, -.0006, -.001, -.0012,
     3              0.0, -.0013, -.003, -.004, -.0052,
     4              0.0, -.0031, -.0066, -.0088, -.012,
     5              0.0, -.008, -.016, -.0207, -.026   /
C
      DATA   XCYC / 0.8, 1.1, 1.4, 1.6, 2.0, 2.4 /,
     1       YXMN / 0.2, 0.3, 0.4, 0.5, 0.6, 0.7 /,
     2       Z1BD5/ 3.784,4.395,5.432,6.348,8.545,11.221, 4.029,4.517,
     3              5.249,5.982,7.874,10.254, 4.395,4.639,5.127,5.616,
     4              7.081,8.973, 4.7,4.834,5.127,5.432,6.287,7.691,
     5              5.371,5.213,5.188,5.31,5.799,6.836,
     6              6.226,5.738,5.432,5.335,5.616,6.409 /
C
C
C     FIGURE 4.1.4.2-25     DATCOM     *****
      DATA   XBD6 / 0., 1., 2., 3. /,
     1       YBD6 / -2., -1., 0., 1. /,
     2       ZBD6 / 0., 1., 2., 3., 4., 5. /,
     3      FBD61 / 0.,.125,.16,.17,0.,.115,.17,.2,0.,.15,.195,.195,0.,
     4 .165,.2,.2,.18,.20,.28,.3,.18,.27,.305,.32,.18,.285,.32,.32,.18,
     5 .3,2*.32,.34,.39,.4,.412,.34,.405,.435,.45,.34,.425,2*.445,.34,
     6.445,.45,.45,.5,.505,.515,.53,.5,.525,.54,.55,.5,.545,2*.55,.5,3*
     7.55, .665,.62,.623,.626,.665,.653,.66,.665,8*.665,
     8  .815,.73,.728,.722,.815,.775,.778,.78,.815,.79,.78,.78,.815,
     9   3*.775 /
      DATA FBD62 / 0.,.135,.17,.186,0.,.12,.19,.215,0.,.175,2*.215,0.,.2
     1,2*.235,.2,.28,.298,.314,.2,.3,.35,.365,.2,.32,2*.36,.2,.365,2*.38
     2, .4,.43,.44,.45,.4,.47,.495,.508,.4,.49,2*.51,.4,3*.52,
     3 .6,.575,.578,.58,.6,.61,.63,.65,.6,.64,2*.65,.6,3*.65,
```

```
     4 .8,3*.72,.8,.77,.78,.792,.8,.795,.792,.79,.8,3*.78,
     5 .98,.84,.83,.82,.98,3*.92,.96,.925,2*.915,.98,3*.885 /,
     6 FEP63 / 0.,.15,.18,.2,0.,.135,.23,.28,0.,.2,2*.275,C.,.24,2*.28,
     7 .26,.35,.365,.38,.26,.365,.42,.45,.26,.405,2*.455,.26,3*.475,
     8 .5,.535,.55,.565,.5,.585,.61,.63,.5,.605,2*.63,.5,3*.65,
     9 .75,3*.735,.75,.78,.795,.81,.75,3*.8,.75,3*.82,
     1 .98,.935,.895,.885,.98,3*.95,.95,.95,2*.945,.98,3*.96,
     2 1.17,1.06,1.03,1.,1.17,1.1,1.08,1.065,1.17,1.09,2*1.065,1.17,
     3 3*1.08 /
C
      DATA XIN13/ C.0,1., 2., 3., 4., 5., 6., 7., 8., 20. /,
     1     YIN13/ 0.0, 20.0, 35.0, 50.0, 65.0, 72.5, 90.C /,
     2    ZCUT13/ .00973,.00973,.0093,.009,.0089,.0087,4*0.0086,
     3            .00973,.00973,.0093,.009,.0089,.0087,4*0.0086,
     4            .0097, .0097,.00927,.0090,.0089,.00872,4*.00862,
     5            .0096,.0096,.0092,.009,.0089,.00872, 4*0.00863,
     6            .00943,.00942,.00897,.00882,.00875,.0086,4*.00854,
     7            .0093,.0093,.0088,.0087,.00862,.0085,4*0.00844,
     8            .0093,.0093,.0088,.0087,.00862,.0085,4*0.00844 /
      DATA ZCUT14/1.28,1.28,1.27,1.252,1.244,1.233,1.227,1.217,2*1.21,
     1            1.28,1.28,1.27,1.252,1.244,1.233,1.227,1.217,2*1.21,
     2            1.324,1.324,1.311,1.295,1.281,1.266,1.258,1.24,2*1.23,
     3            1.363,1.363,1.35,1.33,1.313,1.292,1.283,1.261,2*1.25,
     4            1.392,1.392,1.38,1.36,1.339,1.317,1.303,1.279,2*1.27,
     5            1.403,1.403,1.392,1.371,1.349,1.325,1.31,1.285,2*1.28,
     6            1.403,1.403,1.392,1.371,1.349,1.325,1.31,1.285,2*1.28/
C
      DATA AMAP / 22.0, 21.7, 19.2, 18.35, 22.0, 21.2, 19.2, 27.0,11.75,
     1            24.,24.,22.,20.,20.,19.,17.,15.,29.,27.,25., 2*0. /,
     2     BMAP / 4 * 1.75, 3 * 2.0, 4 * 0.0 /
C
      DATA NXIN15/ 6 /,
     1     XIN15/ 0.0, 0.5, 0.6, 0.8, 0.9, 1.0 /,
     2    NYIN15/ 5 /,
     3     YIN15 / 0.0, 5.0, 15.0, 20.0, 23.0 /,
     4    FCUT15/ 0.865, 0.756, 0.71, 0.6, 0.51, 0.45,
     5            0.865, 0.756, 0.71, 0.6, 0.51, 0.45,
     6            0.925, 0.965, 0.958,.905,.845, 0.78,
     7            0.978, 1.02,1.03, 1.02, 0.99, 0.975,
     8            0.99,  1.04,1.05, 1.05, 1.045,1.04 /,
     9    FCUT16/ 6 * 1.05, 1.05, 1.04, 1.02,.94,.88,.84,
     1              1.05,  1.05, 1.04, 1.01, 0.995, 0.985,
     2            6 * 1.05,   6 * 1.05 /
      DATA X1    / 0.7243, 0.8727, 1.0472, 1.2217, 1.3963, 1.5708 /,
     1     Y1    / 0.85, 0.69, 0.575, 0.51, 0.483, 0.48 /,
     2     X9    / 0.3665,0.5236,0.6981,0.8727,1.0472,1.2217,1.3963,1.6/,
     3     Y9    / 0.85,.625,.435,.35,0.375,0.447,0.476, 0.48 /,
     4     XAR   / 1.0, 2.3, 3.0, 4.0, 5.0 /,
     5     YOCL  / -0.16, 0.0, 0.05, 0.085, 0.0855 /
C
      DATA  AAA / 8.48243E-4, 8.03417E-4, 3.31222E-6, -1.72323E-3,
     4            2.02978E-3, 1.36146E-4, 3.79206E-5, 15 * 0.0 /,
     5      BBB / -.0296429, -.0349676, -.0142049, .0310185,
     6            -.07305, -4.05767E-3, -1.03545E-3, 15 * 0.0  /,
     7      CCC / 1.23835, 1.15999, 1.03825, 0.588837, 1.57142, .712684,
     8            .639304, 0.0, 0.0, 5 * 1.1019, 3 * .275475,
     9            3 * 3.3057, 0.0, 0.0 /,
     1      DDD / -2.47002, -1.71448, -1.9292, -.559968, -2.99739,
     2            .0188291, .0607728, 15 * 0.0 /,
     3      XTT / .35, .375, .41, .45, .37, .39, .42, .371, .5, .2,
     4            .3, .4, .5, .6, .3, .4, .5, .3, .4, .5, 0.0, 0.0 /
      DATA  XSWP  / 0.0, 10.0, 20.0, 30.0, 40.0, 50.0, 60.0 /,
     1      YTR   / 0.0, 0.25, 0.5, 1.0 /,
     2      FEP35 / 0.94,0.951,0.96,0.965,0.973,0.974,0.971,
     3              0.992,0.995,0.997,0.999,0.998,0.992,0.984,
     4              1.0,1.0,0.999,0.997,0.986,0.971,0.944,
     5              0.996,0.992,0.986,0.975,0.959,0.93,0.863 /,
     6      FEP7  / 0.85,0.907,0.923,0.936,0.944,0.948,0.942,
     7              0.99,0.994,0.996,0.998,0.991,0.978,0.956,
     8              0.996,0.992,0.987,0.98,0.961,0.924,0.86,
     9              0.986,0.968,0.942,0.91,0.87,0.833,0.79 /
      DATA  XCLDB / 0.0, 0.3, 0.4, 0.5, 0.6, 0.7, 2.0 /,
     1      YAKB  / 0.33,0.352,0.44,0.625,1.0,1.47, 2.0 /
C
C
      DATA XTR    / 0.0, 0.1, 0.2, 0.3, 0.5, 1.0 /,
     1     YC1    / 0.0,0.225,0.47,0.5,0.32, 0.0 /,
```

```
2        YC2     / 0.0,0.22,0.5, 0.91, 1.05, 0.85 /,
3         XSWP1   / 0.0, 30.0, 50.0, 60.0                  /,
4        YA      / 0.9, 1.04, 1.2, 1.3                    /,
5        YB      / 0.0, 0.24, 0.5, 0.71                   /,
6        XCY     / 0.0, 2.0, 2.25, 2.5, 3.0, 4.0, 4.5, 10.0 /,
7        XM      / 0.2, 0.4, 0.6, 1.0                    /
      DATA CTAB  / 8 * 0.0, -.02,-.02,-.26,-.32,-.355,-.41,-.445,-.45,
9                .12,.12, -.36, -.43, -.5, -.64, -.72, -.72,
1                .58,.58, -.15, -.34, -.57, -.76, -.92, -.92 /,
2        DTAB    / 8 * 0.0, -.085,-.085, .02, .055, .045, .02,.02,.02,
3                -.2, -.2, -.01, -.02, -.03, -.07, -.09, -.09,8*0.0/,
4        XXCLM   / 0.0, .2, .4, .6, .8, 1., 1.2, 1.6, 2.0, 2.4, 2.8,
5                3.2, 4.4 /,
6        YYDY    / 0.0, .25, .5, .75, 1.0, 10.0 /
      DATA FCLMX / 0.9, 1.375, 1.57, 1.645, 1.645, 1.575, 1.4, 1.14,
8                1.02, .96, .92, .91, .9,   .75, 1.28, 1.48, 1.55,
9                1.56, 1.5, 1.33, 1.11, 1.0, .94, .9, .88, .875,
1                .68, 1.19, 1.4, 1.47, 1.475, 1.4, 1.27, 1.08,
2                .975, .92, .88, .85, .84,    .6, 1.1, 1.3, 1.375,
3                1.385, 1.33, 1.23, 1.06, .96, .9, .865, .84, .83,
4                .4,1.02, 1.2, 1.29, 1.31, 1.27, 1.19, 1.04, .94,
5                .89, .84, .83, .82,      .4, 1.02, 1.2, 1.29, 1.31,
6                1.27, 1.16, 1.02, .91, .85, .82, .8, .8       /,
7        XXC2    / 0.0, 2.0, 4., 6., 8., 10., 12., 14., 20./,
8        YYMACH  / 0.0, 0.2, 0.4, 0.6, 1.0    /,
9        FDCLMX  / -.11, -.08, 0.0, .115, .23, .335, .36, .22, 0.0,
1                -.11, -.08, 0.0, .115, .23, .335, .36, .22, 0.0,
2                -.11, -.08, 0.0, .1, .2, .3, .32, .18, 0.0,
3                -.11, -.08, 0.0, .08, .15, .21, .205, .08, 10*0.0/
C
C   ** FIGURE  4.1.1.4-5,   DATCOM   *****
C
      DATA  XDY1  / 0.0, 1.1, 2.25, 2.5, 3.0, 3.5, 4., 4.5, 6. /,
1         YXMT   / 0.3, 0.35, 0.4, 0.45 /,
2        ZC1MAX  / 0.8, 0.8, 1.315,1.43,1.58,1.58,1.54,1.47,1.41,
3                0.8, 0.8, 1.315,1.43,1.51,1.52,1.49,1.41,1.36,
4                0.8, 0.8, 1.315,1.39,1.45,1.45,1.43,1.35,1.34,
5                0.8, 0.8, 1.3,  1.33,1.35,1.36,1.33,1.32,1.32/,
6         XDY2   / 0.0,1.0,2.0,2.2,2.4,3.0,5.2,10.0 /,
7         YFCC   / 0.0, 2.0, 4.0, 6.0, 8.0, 100.0 /,
8        ZDC1M   / 10*0.0, 0.1, 0.13, 0.15, 0.145,0.05, 0.0,
9                0.0, 0.11,0.275,0.3,0.3,0.22,0.10,0.0,
1                0.1,0.26,0.44,0.465,0.45,0.30,0.17,0.0,
2                0.2,0.40,0.60,0.620,0.60,0.39,0.25,0.0,
3                0.2,0.40,0.60,0.620,0.60,0.39,0.25,0.0 /
      DATA Z2DC1M/ 9*0.,.15,.25, .28, .27, .18, .07, .07,
1                .1, .28, .49, .47, .41, .26, .12, .12,
2                .2, .41, .61, .60, .55, .35, .17, .17,
3                .33,.52, .68, .65, .60, .39, .20, .20,
4                .33,.52, .68, .65, .60, .39, .20, .20 /
C
      DATA XSP   / 0.0, 10.0, 20.0, 30.0, 40.0, 50.0, 60.0, 90.0/,
1        YDYA  / 0.0, 1.2, 2.0, 3.0, 4.0, 50.0 /,
2        FDA   / 1.8, 2.2, 3.4, 5.0, 7.4, 10.2, 13.4, 13.4,
3                1.8, 2.2, 3.4, 5.0, 7.4, 10.2, 13.4, 13.4,
4                0.1, 1.1, 2.3, 3.9, 5.8, 6.75, 9.9 , 9.9 ,
5                1.3, 1.7, 2.4, 3.2, 4.2, 5.3 , 6.7 , 6.7 ,
6                2.2, 2.0, 2.1, 2.3, 2.5, 2.85, 3.25, 3.25,
7                2.2, 2.0, 2.1, 2.3, 2.5, 2.85, 3.25, 3.25 /
C
      DATA  XAR    / 0.0, 1.0, 2.0, 3.0, 4.0, 6.0 /,
1         YCD    / -1., -.75, -.5, -.25, 0.0, .25, .5, .75 /,
2      FKVDFM    / 1.57, 1.74, 2.02, 2.30, 2.58, 2.75,
3                1.78, 1.95, 2.18, 2.52, 2.85, 2.90,
4                2.03, 2.21, 2.42, 2.71, 3.00, 3.15,
5                2.48, 2.60, 2.73, 2.93, 3.19, 3.33,
6                3.13, 3.15, 3.20, 3.28, 3.45, 3.60,
7                3.79, 3.80, 3.81, 3.82, 3.88, 3.98,
8                6 * 4.45, 6 * 5.0 /
      DATA  XANG   / 0.0,.0698,.1396,.2094,.2792,.3490,.4188,.4886,
1                0.5584, 0.6980 /,
2        YRTRC   / 0.0, 0.03, 0.05, 0.08, 0.12, 0.16, 0.5 /,
3        FRA / 0.45,0.245,0.075, 7*0.0,
4                1.0,.62,.40,.255,.19,.148,.11,.079,.05, 0.0,
5                1.0,.84,.72,.48,.34,.252,.195,.142,.1,  0.0,
6                1.0,.90,.858,.758,.618,.46,.345,.25, .17, 0.0,
7                1.0,.917,.885,.814,.72,.62,.51,.38, .23, 0.0,
```

```
     8              1.0,.925,.895,.835,.76,.665,.565, .46, .35, 0.0,
     9              1.0,.925,.895,.835,.76,.665,.565, .46, .35, 0.0 /
C
C
      DATA  XC2   / 0.0, 1.0, 2.0, 3.0, 4.0, 5.0 /,
     1      YAST  / 0.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 9.0, 30.0 /,
     2      FDAM  / 10.,7.,4.,1.5,0.,.5, 8.6,5.5,2.5,.5,-.3,.5,
     3              7.5,4.,1.,-.6,-1.,5, 5.5,1.5,-1.2,-2.5,-1.,.5,
     4              3.,-1.5,-4.1,-3.,-1.,.5, .2,-4.5,-5.,-3.,-1.,.5,
     5              -2.3,-6.5,-5.,-3.,-1.,.5, -4.5,-7.,-5.,-3.,-1.,.5,
     6              -8.,-7.,-5.,-3.,-1.,0.5  /,
     7      XCT   / 4.0, 6.0, 8.0, 10.0, 12.0, 14.0 /,
     8      YM    / 0.2, 0.4, 0.6, 0.8, 1.0 /,
     9      FDAM2 / 0.,1.5,4.,7.2,11.,15., 0.,.8,3.,6.,9.5,13.5,
     1              0.,.3,1.5,3.2,5.5,8.6, 0.,0.,.5,2.,4.,7.,
     2              0., .8, 3., 6., 9.5, 13.5 /
C
      DATA XXXCLM / 0.0, .2, .4, .6, .8, 1., 1.2, 1.6, 2.0 /,
     1     FCLMXY / 0.7,1.2,1.37,1.43,1.43,1.365,1.25,1.08,.99,
     1              .65,1.12,1.29,1.355,1.365,1.305,1.21,1.07,.98,
     2              .6,1.09,1.23,1.29,1.295,1.24,1.17,1.05,.97,
     3              .55,1.0,1.15,1.23,1.24,1.20,1.15,1.04,.96,
     4              .5,.91,1.09,1.17,1.20,1.175,1.13,1.02,.935,
     5              .5,.91,1.09,1.17,1.20,1.165,1.11,1.0,.91 /
C
      DATA  XX    / 0.0,.4,.8,1.0,1.2,1.4,1.6,2.0,2.4,2.8,3.2,3.27 /,
     1      XY    / 0.,.1309,.1745,.2618,.3490,.4363,1.0 /,
     2      XF    / 8*1.0,.915,.63,.29,0.0,
     3              8*1.0,.915,.63,.29,0.0,
     4              7*1.0,.915,.755,.52,.245,0.0,
     5              1.0,.97,.95,.97,1.0,.95,.87,.69,.552,.38,.185,0.0,
     6              .92,.90,.90,.95,1.0,.95,.845,.63,.48,.31,.150,0.0,
     7              .79,.80,.84,.91,1.0,.945,.83,.585,.425,.27,.128,0.,
     8              .79,.80,.84,.91,1.0,.945,.83,.585,.425,.27,.128,0. /
C
C
C
*****************************************************************
*********                                                  ******
**********  D A T A   F O R   D A M A G E   E Q U A T I O N S   ******
*********                                                  ******
*****************************************************************
C
C
C     COMPRESSIBILITY FACTOR,  FIG 2-15,FZP-1800
C
      DATA XMACH3/0.0, 0.2, 0.4, 0.6, 0.8, 0.9, 0.98, 1.0, 1.05,
     1            1.1, 1.2, 1.4, 1.6, 1.8, 2.0, 2.2,  2.4        /,
     2     CCMK3 /1.00, 1.00, 1.03, 1.05, 1.20, 1.36, 1.95,1.90,1.55,
     3            1.38, 1.22, 1.04, 0.90, 0.76, 0.67, 0.60,0.55  /
C
C     CDC FOR HOLES,FIGURE  2-16  FZP-1800
C
      DATA XLCH3 / 1.0, 2.0, 4.0, 6.0, 7.0, 10.0 /,
     1     HCX3  /0.002, 0.010, 0.020, 0.060, 0.100 /,
     2     CDC3  /0.005, 0.015, 0.048, 0.101, 0.110, 0.110,
     3            0.009, 0.022, 0.080, 0.175, 0.190, 0.190,
     4            0.015, 0.038, 0.126, 0.275, 0.300, 0.300,
     5            0.025, 0.060, 0.215, 0.470, 0.505, 0.505,
     6            0.040, 0.094, 0.330, 0.775, 0.820, 0.820 /
C
C     WING ROUGHNESS MAGNIFICATION FACTOR, FIG 2-14, FZP-1800
C
      DATA XCC3  / 0.0, 0.2, 0.4, 0.6, 0.8, 1.0  /,
     1     CLD3  / 0.0, 0.2, 0.4, 0.6            /,
     2     XM3   / 1.35, 1.55, 1.50, 1.30, 1.07, 0.83,
     3             1.35, 1.55, 1.50, 1.30, 1.07, 0.83,
     4             1.05, 1.75, 2.15, 1.83, 1.30, 0.83,
     5             1.05, 1.75, 2.15, 1.83, 1.30, 0.83 /
C
C     FORWARD-FACING STEP PRESSURE COEFFICIENTS, FIG 2-13, FZP-1800
C
      DATA YMACH3/ 0.0, 0.4, 0.8, 0.9, 0.95, 1.0, 1.05,
     1             1.1, 1.2, 1.6, 2.0, 2.4 , 2.8              /,
     2     CPF3  / 0.40, 0.44, 0.52, 0.56, 0.58, 0.61, 0.60,
     3             0.57, 0.51, 0.38, 0.27, 0.25, 0.25         /
C
```

```
C     AFT-FACING STEP PRESSURE COEFFICIENTS; FIG 2-13, FZP-1800
C
      DATA ZMACH3/ 0.0, 0.4, 0.8, 0.9, 0.95, 1.0, 1.05,
     1           1.1, 1.2, 1.6, 2.0, 2.4 , 2.8                    /,
     2     CPA3  /-0.20, -0.20, -0.20, -0.21, -0.40, -0.53, -0.48,
     3           -0.45, -0.34, -0.19, -0.14, -0.11, -0.10     /
C
C     CAVED-IN PANELS , FIGURE 2-17,  FZP-1800
C
      DATA XICH4 /10.,13.,15.,20.,25.,30.,40.,50.,60.,80.,100.,120./,
     1     HCX4  /.0002,0.0005,0.001,0.003,0.005,0.010,0.015          /,
     2     CDC4  /0.0250, 0.0150, 0.0110, 0.0060, 0.0040, 0.0028,
     3           0.0016, 0.0010, 0.0007, 0.0004, 0.0003, 0.0002,
     4           0.0360, 0.0210, 0.0160, 0.0090, 0.0050, 0.0040,
     5           0.0022, 0.0014, 0.0010, 0.0006, 0.0004, 0.0003,
     6           0.0450, 0.0270, 0.0200, 0.0110, 0.0070, 0.0050,
     7           0.0028, 0.0018, 0.0012, 0.0007, 0.0005, 0.0003,
     8           0.0650, 0.0380, 0.0290, 0.0160, 0.0100, 0.0072,
     9           0.0040, 0.0026, 0.0018, 0.0010, 0.0006, 0.0004,
     $           0.0770, 0.0460, 0.0350, 0.0190, 0.0120, 0.0086,
     1           0.0048, 0.0031, 0.0021, 0.0012, 0.0008, 0.0006,
     2           0.0970, 0.0570, 0.0430, 0.0240, 0.0150, 0.0108,
     3           0.0060, 0.0038, 0.0026, 0.0015, 0.0010, 0.0007,
     4           0.1100, 0.0650, 0.0490, 0.0270, 0.0180, 0.0120,
     5           0.0070, 0.0045, 0.0031, 0.0017, 0.0011, 0.0008 /
C
C     EFFECT OF SEVERING NOSE, SPHERICAL SHAPE, NACA RM L53014A
C
      DATA RATIO3 /0.0, 0.2, 0.4, 0.6, 0.7, 0.8, 0.9, 1.0          /,
     1     UMACH  /0.0,0.8,0.9,0.95,1.0,1.1,1.2,1.6,2.0,3.0       /,
     2     DRAG3  /0.0,0.0,0.0,0.0,0.0,0.0,0.010,0.040,
     3            0.0,0.0,0.0,0.0,0.0,0.0,0.010,0.040,
     4            0.0,0.0,0.0,0.005,0.012,0.030,0.065,0.150,
     5            0.0,0.0,0.0,0.010,0.027,0.036,0.090,0.220,
     6            0.0,0.0,0.0,0.028,0.059,0.090,0.160,0.295,
     7            0.0,-.010,0.0,0.024,0.053,0.120,0.217,0.330,
     8            0.0,-.010,0.005,0.054,0.103,0.183,0.288,0.410,
     9            0.0,0.010,0.050,0.155,0.245,0.358,0.495,0.640,
     $            0.0,0.022,0.080,0.210,0.313,0.425,0.562,0.703,
     1            0.0,0.022,0.080,0.210,0.313,0.425,0.562,0.703 /
C
C     EFFECT OF FLATTING SPHERICAL NOSE, NACA TN 4201 (PP-38)
C
      DATA XLCD   / 0.0, 0.1, 0.2, 0.3, 0.4, 0.5   / ,
     1     VMACH  / 0.0, 0.8, 0.9, 1.0, 1.2, 2.5   / ,
     2     DRAG4  / 0.705,0.535,0.375,0.225,0.098, 0.0,
     3              0.705,0.535,0.375,0.225,0.098, 0.0,
     4              0.695,0.525,0.365,0.215,0.090, 0.0,
     5              0.646,0.490,0.341,0.202,0.083, 0.0,
     6              0.667,0.460,0.313,0.181,0.075, 0.0 ,
     7              0.667,0.460,0.313,0.181,0.075, 0.0 /
C
C     EFFECT OF MISSING L.E. PANELS ON LIFT CURVE
C
      DATA DETA1  / 0.0, 0.2, 0.4, 0.6, 0.8, 1.0        / ,
     1     AR1    / 0.0, 3.0, 5.0, 7.0, 10.0             / ,
     2     CLA1   / 0.0, -0.0021, -.0038, -.0052, -.0064, -.0075,
     3              0.0,-0.0021, -.0038, -.0052, -.0064, -.0075,
     4              0.0, -0.0032, -.0056, -.0080, -.0100, -.0116,
     5              0.0, -0.0039, -.0070, -.0096, -.0120, -.0135,
     6              0.0, -0.0039, -.0070, -.0096, -.0120, -.0135 /
C
C
      DATA DETA2  / 0.0, 0.2, 0.4, 0.6, 0.8, 1.0   /,
     1     SWPOC2 /-9.0, 0.0, 20.0, 40.0, 60.0      /,
     2     CLA2   / 0.0, -.0002, -.0003, -.0004, -.0004, -.0004,
     3              0.0, -.0002, -.0003, -.0004, -.0004, -.0004,
     4              0.0, 0.0   , 0.0   , 0.0   , 0.0   , 0.0   ,
     5              0.0, 0.0005, 0.0009, 0.0012, 0.0015, 0.0018,
     6              0.0, 0.0005, 0.0009, 0.0012, 0.0015, 0.0018 /
C
C
      DATA ETAI3  /  0.0, 0.2, 0.4, 0.6, 0.8, 1.0     /,
     1     CLA3   /  0.0012, 0.0005, 0.0, 0.0008, 0.0024, 0.0042/
C
C     EFFECT OF MISSING T.E. PANELS ON LIFT CURVE
C
```

```
      DATA DETA4  /  0.0, 0.2, 0.4, 0.6, 0.8, 1.0    /,
     1     AR4    /  0.0, 3.0, 5.0, 7.0, 10.0        /,
     2     CLA4   /  0.0, -.0020, -.0036, -.0050, -.0063, -.0073,
     3               0.0, -.0020, -.0036, -.0050, -.0063, -.0073,
     4               0.0, -.0030, -.0055, -.0077, -.0096, -.0110,
     5               0.0, -.0037, -.0065, -.0091, -.0114, -.0130,
     6               0.0, -.0037, -.0065, -.0091, -.0114, -.0130 /
C
C
      DATA DETA5  /  0.0, 0.2, 0.4, 0.6, 0.8, 1.0  .  /,
     1     SWFCC5 /-0.0, 0.0, 20.0, 40.0, 60.0      /,
     2     CLA5   /  0.0, -.0002, -.0003, -.0004, -.0004, -.0005,
     3               0.0, -.0002, -.0003, -.0004, -.0004, -.0005,
     4               0.0, 0.0   , 0.0   , 0.0   , 0.0   , 0.0   ,
     5               0.0, 0.0007, 0.0012, 0.0015, 0.0017, 0.0019,
     6               0.0, 0.0007, 0.0012, 0.0015, 0.0017, 0.0019 /
C
      DATA ETAI6  / 0.0, 0.2, 0.4, 0.6, 0.8, 1.0   /,
     1     CLA6   /-0.0017, -.0010, -.0002, 0.0008, 0.0021, 0.0036/
C
C
C     EFFECT OF CUTOUT CHORD LENGTH ON LIFT CURVE
C       FROM BRITISH DATA SHEET, VOL 2A, WINGS 01.01.04
C
      DATA DROB   / 0.0, 0.2, 0.4, 0.6, 0.8, 1.0    / ,
     1      CCCC  / 0.0, 0.2, 0.3, 0.4, 0.5, 0.6    / ,
     2   CLACLA   / 1.0, 1.045, 1.088, 1.112, 1.155, 1.185,
     3              1.0, 1.0  , 1.0  , 1.0  , 1.0  , 1.0  ,
     4              1.0, 0.975, 0.950, 0.930, 0.910, 0.890,
     5              1.0, 0.965, 0.910, 0.870, 0.830, 0.790,
     6              1.0, 0.920, 0.850, 0.780, 0.720, 0.670,
     7              1.0, 0.880, 0.770, 0.680, 0.600, 0.550 /
C
C     EFFECT OF MISSING L.E. PANELS ON AERODYNAMIC CENTER
C
      DATA DETA11 / 0.0, 0.2, 0.4, 0.6, 0.8, 1.0    / ,
     1     AR11   / 0.0. 3.0, 5.0, 7.0, 10.0        / ,
     2     XAC1   / 0.000, 0.036, 0.074, 0.113, 0.153, 0.184,
     3              0.000, 0.036, 0.074, 0.113, 0.153, 0.184,
     4              0.000, 0.030, 0.060, 0.091, 0.123, 0.155,
     5              0.000, 0.025, 0.051, 0.078, 0.106, 0.135,
     6              0.000, 0.025, 0.051, 0.078, 0.106, 0.135  /
C
C
      DATA DETA12 / 0.0, 0.2, 0.4, 0.6, 0.8, 1.0    / ,
     1     SWCC12 /-9.0, 0.0, 20.0, 40.0, 60.0      /,
     2     XAC2   / 0.000, 0.003, 0.005, 0.008, 0.010, 0.013,
     3              0.000, 0.003, 0.005, 0.008, 0.010, 0.013,
     4              0.0  , 0.0  , 0.0  , 0.0  , 0.0  , 0.0  ,
     5              0.0  , -.003, -.006, -.010, -.014, -.020,
     6              0.0  , -.003, -.006, -.010, -.014, -.020  /
C
C
      DATA ETAI13   /0.0, 0.2, 0.4, 0.6, 0.8, 1.0  / ,
     1     XAC3     /0.055, 0.030, 0.006, -.015, -.034, -.050 /
C
C     EFFECT OF MISSING T.E. PANELS ON AERODYNAMIC CENTER
C
      DATA DETA14 /0.0, 0.2, 0.4, 0.6, 0.8, 1.0     /,
     1     AR14   /0.0, 3.0, 5.0, 7.0, 10.0         /,
     2     XAC4   /0.000, -.011, -.020, -.027, -.033, -.037,
     3             0.000, -.011, -.020, -.027, -.033, -.037,
     4             0.000, -.012, -.022, -.031, -.039, -.045,
     5             0.000, -.014, -.026, -.036, -.045, -.052,
     6             0.000, -.014, -.026, -.036, -.045, -.052   /
C
C
      DATA DETA15 /0.0, 0.2, 0.4, 0.6, 0.8, 1.0     /,
     1     SWCC15 /-9.0, 0.0, 20.0, 40.0, 60.0      /,
     2     XAC5   /0.000, 0.005, 0.009, 0.013, 0.016, 0.019,
     3             0.000, 0.005, 0.009, 0.013, 0.016, 0.019,
     4             0.0  , 0.0  , 0.0  , 0.0  , 0.0  , 0.0  ,
     5             0.0  , -.008, -.015, -.021, -.026, -.030,
     6             0.0  , -.008, -.015, -.021, -.026, -.030   /
C
C
      DATA  ETAI16  / 0.0, 0.2, 0.4, 0.6, 0.8, 1.0        /,
```

```
     1       XAC6    / 0.008, 0.004, 0.001, 0.002, 0.006, 0.012 /
C
C
      DATA DCDC7  / 0.0, 0.2, 0.4, 0.6, 0.8            /,
     1      XAC7  /-0.031, 0.0, 0.025, 0.047, 0.064 /
C
C
      DATA DCDC8  / 0.0, 0.2, 0.4, 0.6, 0.8            /,
     1      XAC8  /0.017, 0.0, -0.014, -0.024, -0.030/
C
C     EFEECT OF HOLES IN WING ON LIFT-CURVE AND POLAR SHAPE
C
      DATA FL01   / 0.0, 0.02, 0.04, 0.06, 0.08, 0.2 /,
     1      CLAH1 / 1.0,0.981,0.952,0.910,0.858, 0.858/
C
C
      DATA FL02   / 0.0, 0.02, 0.04, 0.06, 0.08, 0.20 /,
     1      FFH2  / 1.0,0.904,0.759,0.584,0.404, 0.404 /
C
C     VORTEX LIFT INCREMENT FOR MISSING WING TIP (NUCDAM2)
C
      DATA ZAR    /0.0,0.5,1.0,1.5,2.0,2.5,3.0,3.5,4.0,10.0/,
     1      XCLA  /0.0285, 0.0250, 0.0175, 0.0104, 0.0040,
     2             -0.0020,-0.0065,-0.0100,-0.0110,-0.0110  /
C
C    ***********************************************************
C
C     DATA FOR ELEVON CALCULATIONS   COMMON ELKDAT5
C
C
      DATA    SPAN / 0.0, 0.2, 0.4, 0.6, 0.8, 1.0 /,
     1        YTPR / 0.0, 0.25, 0.5, 1.0 /,
     2        FKB  / 0.0, 0.3, 0.555, 0.77, 0.925, 1.0,
     3               0.0, 0.28,0.53, 0.75, 0.915, 1.0,
     4               0.0, 0.265, 0.51, 0.73, 0.9, 1.0,
     5               0.0, 0.25, 0.48, 0.70, 0.885, 1.0 /,
     7        FKD  / 0.0, 0.38, 0.64, 0.84, 0.965, 1.0,
     8               0.0,0.295, 0.54, 0.74, 0.9,   1.0,
     9               0.0, 0.24, 0.475, 0.68, 0.855, 1.0,
     1               0.0, 0.2, 0.4, 0.6, 0.8, 1.0 /
C
      DATA SPAN2 /0.0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1.0 /,
     1     YTPR2 / 0.0, 0.2, 0.332, 0.5, 1.0 /,
     2     FKSW  / 0.0, 0.0265, 0.0435, 0.0490, 0.0480, 0.0417,
     3         0.0320, 0.0212, 0.0112, 0.0035, 0.0,
     4            0.0, 0.0210, 0.0460, 0.0545, 0.0565, 0.0530,
     5         0.0460, 0.0350, 0.0240, 0.0120, 0.0,
     6            0.0, 0.0290, 0.0460, 0.0560, 0.0590, 0.0570,
     7         0.0510, 0.0410, 0.0290, 0.0150, 0.0,
     8            0.0, 0.0270, 0.0450, 0.0560, 0.0610, 0.0600,
     9         0.0545, 0.0450, 0.0230, 0.0150, 0.0,
     1            0.0, 0.0225, 0.0400, 0.0575, 0.0605, 0.0620,
     2         0.0600, 0.0520, 0.0400, 0.0220, 0.0 /
C
      DATA SPAN3 / 0.0, 0.2, 0.4, 0.6, 0.8, 1.0 /,
     1     YTPR3 / 0.0, 0.25, 0.5, 1.0 /,
     2     FKM   / 0.0, 0.645, 1.045, 1.240, 1.220, 1.330,
     3             0.0, 0.440, 0.760, 0.955, 1.070, 1.120,
     4             0.0, 0.310, 0.575, 0.775, 0.930, 1.030,
     5             0.0, 0.200, 0.400, 0.600, 0.800, 1.000  /
      DATA SPAN4 /  0.0, 0.4, 0.8, 1.6, 2.4 /,
     1     YBFI1 /  0.0, 0.05, 0.1, 0.15, 0.2 /,
     2     FKA   /  0.0, 0.38, 0.68, 1.00, 1.13,
     3              0.0, 0.40, 0.74, 1.12, 1.32,
     4              0.0, 0.49, 0.83, 1.38, 1.75,
     5              0.0, 0.57, 1.05, 1.81, 2.43,
     6              0.0, 0.70, 1.27, 2.34, 3.37 /
      DATA SPAN5 /  0.2, 0.4, 0.6, 0.8, 1.0 / ,
     1     YBFI2 /  0.0, 0.1, 0.2 /,
     2     FKF   /  4.00, 1.60, 0.60, 0.17, 0.0,
     3              3.40, 1.55, 0.65, 0.26, 0.12,
     4              2.96, 1.46, 0.72, 0.40, 0.25 /
C
      END
```

```
      OVERLAY(1,0)
      PROGRAM XINPT
C
      COMMON/INPUT/  A(3081)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/    C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ E(968)
      COMMON /BLKTIL/ TITLE(6)
      COMMON /BLKPRT/ KPRINT(50)
      COMMON /BLKOV1/ IJ
C
      DIMENSION W(4)
      DATA  W  /4HNEW , 4HDAMA, 4HCHAN, 4HEND /
C
      IF( IJ.GE.1 ) GO TO 70
   20 CONTINUE
C
C
      READ(5,1000) (TITLE(I), I = 1,6 )
      IF (EOF(5).NE.0) GO TO 75
      WRITE(6,2000) (TITLE(I), I= 1,6 )
C
      READ(5,1001) (KPRINT(I), I=1,50)
C
      DO 50 I = 1, 50
      IF( KPRINT(I).GT.0 ) WRITE(6,2001) I, KPRINT(I)
   50 CONTINUE
      CALL SETUP
C     PROGRAM INPT CALLED
      CALL OVERLAY(4HOVLY,1,1)
C
      IJ      = 1
      GO TO 80
C
   70 READ(5,1002) WORD
      IF( EOF(5).NE.0 ) CALL EXIT
      IF( WORD.EQ.W(3) ) CALL OVERLAY(4HOVLY,1,3)
      IF( WORD.EQ.W(3) ) IJ = 1
      IF( WORD.EQ.W(2) ) CALL OVERLAY(4HOVLY,1,2)
      IF( WORD.EQ.W(2) ) IJ = 2
      IF (WORD.EQ.W(2).OR.WORD.EQ.W(3) ) GO TO 80
      IF( WORD.EQ.W(4) ) GO TO 20
C
      WRITE(6,2002) WORD
      CALL EXIT
C
   75 IF (IJ.GE.1.AND.KPRINT(2).EQ.1) CALL XCOPY4(2)
      CALL EXIT
C
   80 CONTINUE
C
 1000 FORMAT( 6A10 )
 1001 FORMAT( 50I1 )
 1002 FORMAT( A4 )
 2000 FORMAT(1H1,/////, 27X,*EMPIRICALLY BASED COMPUTER PROGRAM* //
     1                  23X,*TO PREDICT THE AERODYNAMIC CHARACTERISTICS*
     2  //              35X,*OF AIRCRAFT* //
     3                  39X,*INCLUDING*    //
     4                  29X,*ROUGHNESS, PROTUBERANCES, AND* //
     5                  34X,*DAMAGE CALCULATIONS* (8(/)), 27X, 6A10)
 2001 FORMAT(5X,*KPRINT(*,I2,*) = *,I2)
 2002 FORMAT(5X, *UNRECOGNIZED CONTROL WORD *,A4,* READ*)
C
      END
```

```
      SUBROUTINE SETUP
C
C     ZEROES OUT INPUT COMMON BLOCKS
C
      COMMON/INPUT/  A(3081)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/   C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ E(908)
C
      DO 10  I=  1, 3081
      A(I)  = 0.0
   10 CONTINUE
C
      RETURN
      END
```

```
      OVERLAY(1,1)
      PROGRAM INPT
C
C     PROBLEM DATA INPUT
C
      COMMON /INPUT/ NBODYS,NMAC,NPNLS,NHT,NVT,ISWP,IREF,IWNG,NAFO,
     1               METER, SREF, CMAC, XCG, ZCG, TWIST,
     1               ROUGHK,FMISC,NPODS,SPLAN,TAPR,SWP,ITRIM,RLE,
     2               TOCR,CLDP,TECH,CONCL,UPFUS,ACB,
     3               PNAME(7), POD(7,12), CFCC,EI,EO,A(4),
     4               SNAME(7), SUR(7,30),
     4               AFX(30), AFC(30), AFT(30),
     5               XPIVOT, YPIVOT, XAPEX, AFTSW, AFTCR, AFTCC, AFTAW,
     6               NSLPV, NCLAS, FMSURV(20), ALT(20), CG(20),
     7               SWPV(20), CLDP(20), CLMI(20), FHSV(20),
     8               IVAL(20), NXVAR, NADJ, NXCL, NADJ2, X(15),
     9               YM(15,9), YA(15,9), XCL(15),
     1               NFUSPT, IFUSC, NCPCO, NPCOPT, XFUS, YFUS, ZFUS,
     2               FUSY(100), FUSC(100), FUSA(100),
     3               PODX(6), PODY(6), PODZ(6), DELX(6,30), DELA(6,30),
     4               IDAM(20), DEDP(7,100), DSUR(7,100), DWING(20)
C
C
      DIMENSION TTT(20), W(1), ASUR(7)
C
      DATA TTT / 3H62-,3H64-,3H65-,3H66-,3H63A,3H64A,3H65A,3HINP,
     1           3HPIC,3H-62,3H-63,3H-64,3H-65,3H-66,3H-33,3H-34,
     1           3H-35,3H-93,3H-94,3H-95 /
      DATA W     / 4HADJU /
C
C     CONFIGURATION DEFINITION
C
C                    OPTIONS CARDS
C
  100 READ(5,1001) NPODS, NPNLS,NHT, NVT, ISWP,
     1             IREF, IWNG, NAFO, METER, ITRIM
      READ(5,1002) SREF, CMAC, XCG,ZCG,UPFUS, ACB
      READ(5,1002) ROUGHK, FMISC, TWIST, CONCL
C
      WRITE(6,2001)  NPODS, SREF  , ROUGHK
      WRITE(6,2002)  NPNLS, CMAC  , FMISC
      WRITE(6,2003)  NHT  , XCG   , TWIST
      WRITE(6,2004)  NVT  , ZCG   , CONCL
      WRITE(6,2005)  ISWP , UPFUS
      WRITE(6,2006)  IREF , ACB
      WRITE(6,2007)  IWNG , NAFO, METER, ITRIM
      IF (METER.EQ.0) WRITE(6,4002)
      IF (METER.EQ.1) WRITE(6,4003)
      IF (METER.EQ.2)  WRITE(6,4004)
      IF (IWNG.GT.0) READ(5,1002) SPLAN,TAPR,SWP
      IF (IWNG.GT.0) WRITE(6,4001) SPLAN,TAPR,SWP
C
      IF (NHT.EQ.0.AND.ITRIM.EQ.0) READ(5,1002) CFCC, EI, EO
C                  FUSELAGE GEOMETRY
C
      IF (NPODS.EQ.0)  GO TO 104
      IF (NPODS.GT.7) CALL EXIT
      READ(5,1004) (PNAME(I), I = 1,NPODS)
      DO 101 J = 1,12
      READ (5,1002)  (POD(I,J), I=1,NPODS)
  101 CONTINUE
  105 WRITE(6,2009) (PNAME(I),  I = 1,  NPODS)
      WRITE(6,2010) (POD(I,1) , I = 1,  NPODS)
      WRITE(6,2011) (POD(I,2) , I = 1,  NPODS)
      WRITE(6,2012) (POD(I,3) , I = 1,  NPODS)
      WRITE(6,2013) (POD(I,4) , I = 1,  NPODS)
      WRITE(6,2014) (POD(I,5) , I = 1,  NPODS)
      WRITE(6,2015)  (POD(I,6) , I = 1,  NPODS)
      WRITE(6,2016) (POD(I,7) , I = 1,  NPODS)
      WRITE(6,2017) (POD(I,8) , I = 1,  NPODS)
      WRITE(6,2018) (POD(I,9) , I = 1,  NPODS)
      WRITE(6,2019) (POD(I,10), I = 1,  NPODS)
  106 WRITE(6,2020) (POD(I,11), I = 1,  NPODS)
      WRITE(6,2021) (POD(I,12), I = 1,  NPODS)
C
      N       =  0
      DO 110  I = 1, NPODS
```

```
      IF (RCD(I,12).GT.0.0)  N = N + 1
  110 CONTINUE
      NNAC   = N
      NPODYS = NPODS - NNAC
C
C                   SURFACE GEOMETRY
C
  104 NSUR    =   NPNLS + NHT + NVT
      IF (NSUR.GT.7) CALL EXIT
      READ(5,1004) (SNAME(I) , I = 1, NSUR)
      READ(5,1005) (ASUR(I), I=1,NSUR)
      DO 107  J=2,14
      IF (J.EQ.5.OR.J.EQ.6) GO TO 107
      READ (5,1002)  (SUR(I,J),I =1,NSUR)
  107 CONTINUE
      WRITE(6,2023) (SNAME(I), I = 1,NSUR)
      WRITE(6,2024) (ASUR(I),  I =1,NSUR)
      WRITE(6,2025) (SUR(I, 2),   I =   1,NSUR)
      WRITE(6,2026) (SUR(I, 3),   I =   1,NSUR)
      WRITE(6,2027) (SUR(I, 4),   I =   1,NSUR)
      WRITE(6,2030) (SUR(I, 7),   I =   1,NSUR)
      WRITE(6,2031) (SUR(I, 8),   I =   1,NSUR)
      WRITE(6,2032) (SUR(I, 9),   I =   1,NSUR)
      WRITE(6,2033) (SUR(I,10),   I =   1,NSUR)
      WRITE(6,2034) (SUR(I,11),   I =   1,NSUR)
      WRITE(6,2035) (SUR(I,12),   I =   1,NSUR)
      WRITE(6,2036) (SUR(I,13),   I =   1,NSUR)
      WRITE(6,2037) (SUR(I,14),   I =   1,NSUR)
C
      IF (IWNG.EQ.0) SWP = SUR(NPNLS,4)
C
C              AIRFOIL AND SWEEP OPTIONS
C
      DO 109  I = 1,NSUR
  109 IF (ASUR(I).EQ.TTT(8))  GO TO 108
      GO TO 112
C
  108 READ(5,1002) RLE, TOCR, CLDR, TECH
      READ(5,1002)   (AFX(I), I =1,NAFC)
      READ(5,1002)   (AFC(I), I =1,NAFC)
      READ(5,1002)   (AFT(I), I =1,NAFC)
C
      WRITE(6,2040) RLE, TOCR, CLDR, TECH
      DO 111    I = 1, NAFC
      WRITE(6,2041) AFX(I),AFC(I), AFT(I)
  111 CONTINUE
C
C             VARIABLE SWEEP DATA
C
  112 IF (ISWP.NE.1) GO TO 113
      READ(5,1002) XPIVOT, YPIVOT, AFTSW, AFTCB, AFTOC, AFTAW
C
C             SURVEY CONTROL
C
  113 READ (5,1001)  NSURV, NCLAS
      DO 114  I = 1, NSURV
      READ(5,1002) FMSURV(I), ALT(I), CG(I), SWPV(I), CLLO(I),
     1              CLHI(I)
      IF (CLHI(I).EQ.0.0) CLHI(I) = 1.0
      IF (ISWP.EQ.1) GO TO 114
      IF (SWPV(I).EQ.0.0.AND.SWPV(I).NE.SWP) SWPV(I) = SWP
  114 CONTINUE
      WRITE(6,2042) NSURV
      DO 115  I = 1, NSURV
      WRITE(6,2043) FMSURV(I), ALT(I), CG(I), SWPV(I), CLLO(I),
     1              CLHI(I)
  115 CONTINUE
C
C              ADJUSTMENT FACTORS
C
  116 READ(5,1006) WORD, WRD
      IF (WORD.EQ.W(1)) GO TO 117
      GO TO 122
  117 READ(5,1007) (IVAL(I),I=1,20), NXVAR, NADJ, NXCL, NADJ2
      IJ     = 0
      IF (NXVAR.EQ.0) GO TO 119
      READ(5,1002) (X(I), I = 1, NXVAR)
```

```
      DO 118 I = 1, 3
      IF (IVAL(I).EQ.1) IJ = IJ + 1
      IF (IVAL(I).EQ.1) IVAL(I) = IJ
      IF (I.GT.NADJ) GO TO 118
      READ(5,1003) (YM(J,I), YA(J,I),  J = 1,NXVAR)
  118 CONTINUE
  119 IF (NXCL.EQ.0)  GO TO 121
      READ(5,1002) (XCL(I),  I = 1, NXCL)
      DO 120 I = 1, 2
      I1     = I + 3
      IF (IVAL(I1).EQ.1) IJ = IJ + 1
      IF (IVAL(I1).EQ.1) IVAL(I1) = IJ
      IF (I.GT.NADJ2) GO TO 120
      READ(5,1003) (YM(J,NADJ+I), YA(J,NADJ+I), J = 1,NXCL)
  120 CONTINUE
  121 CONTINUE
  122 WRITE(6,2044) WORD, WRD
C
C
C
C             DEFINE AIRFOIL SECTION
C
      DO 125   I = 1, NSUP
      DO 124   J = 1, 20
C
      IF (ASUP(I).EQ.TTT(J)) GO TO 126
      GO TO 124
 126  SUP(I,1) = J
      GO TO 125
  124 CONTINUE
      WRITE(6,2045) J, ASUP(J)
  125 CONTINUE
C
C
 260  CONTINUE
C
C
C     CHANGE DEGREES TO RADIANS
C
      AA      =  1. / 57.29578
C
      DO 280  I = 1, NSUP
      SUR(I,4)  =   SUR(I,4)   * AA
  280 CONTINUE
      SWP     =   SWP * AA
      AFTSW       =  AFTSW      * AA
C
C         CONVERT UNITS TO INCHES ,  IF METER = 2
C
      IF (METER.NE.2) GO TO 350
      DO 300  I  = 1,NBODS
  300 BOD(I,4)   = BOD(I,4) * 144.
      DO 310  I  = 1,NSUP
  310 SUR(I,7)   = SUR(I,7) * 144.
      AFTAW      = AFTAW    * 144.
      SREF       = SREF     * 144.
      SPLAN      = SPLAN    * 144.
      GO TO 400
  350 CONTINUE
C     CONVERT ROUGHNESS HEIGHT FROM INCHES TO FT OR METERS
      AA     = 1./12.
      IF (METER.EQ.1) AA = 0.0254
      ROUGHK   = ROUGHK * AA
  400 CONTINUE
C
C             INPUT FORMAT STATMENT
C
 1001 FORMAT(14I5)
 1002 FORMAT(7F10.0)
 1003 FORMAT(6F10.0)
 1004 FORMAT(7A10)
 1005 FORMAT(A3,7X,A3,7X,A3,7X,A3,7X,A3,7X,A3,7X,A3)
 1006 FORMAT(A4,A10)
 1007 FORMAT(2OI1, 4I5)
 1008 FORMAT(2OI1)
C
C             PRINT FORMAT STATEMENTS
```

```
C
 2001 FORMAT(1H1, 10X, * PROBLEM INPUT PARAMETERS * , //,
     2        5X,*NRODS =*, I8, 20X, *SREF  =*, F10.2,20X, *ROUGHK=*,
     3        F10.5)
 2002 FORMAT(5X,*NPNLS =*, I8,20X,*CMAC  =*,F10.3,20X,*EMISC =*,F10.2)
 2003 FORMAT(5X,*NHT   =*, I8,20X,*XMAC  =*,F10.3,20X,*TWIST =*,F10.3)
 2004 FORMAT(5X,*NVT   =*, I8,20X,*ZCG   =*,F10.3,20X,*CONCL =*,F10.3)
 2005 FORMAT(5X,*ISWP  =*, I8,20X,*UPRLS =*,F10.3)
 2006 FORMAT(5X,*IPCF  =*, I8,20X,*ACR   =*,F10.3)
 2007 FORMAT(5X,*IWNC  =*, I8, /,
     1        5X,*MAFC  =*, I8, /,
     2        5X,*METER =*, I8, /,
     3        5X,*ITRIM =*, I8, /)
 4002 FORMAT(//, 10X, *DIMENSIONS ARE IN FEET *)
 4003 FORMAT(//, 10X, *DIMENSIONS ARE IN METERS*)
 4004 FORMAT(//, 10X, *DIMENSIONS ARE IN INCHES EXCEPT FOR *,/,
     1              10X, *REFERENCE, PLANFORM, AND WETTED AREAS*,/,
     2              10X, *AND ALSO ALTITUDE AND REYNOLDS NUMBER*)
 4001 FORMAT(////,2X,*REFERENCE WING FOR TWO PANEL CASE *,//,
     1        5X,*SPLAN =*,F9.3,5X,*TAPR =*,F6.4,5X,*SWP =*,F8.3)
C
 2009 FORMAT(////,20X,7A10)
 2010 FORMAT(//,2X,*LENGTH               = *,7F10.3)
 2011 FORMAT( /,2X,*WIDTH                = *,7F10.3)
 2012 FORMAT( /,2X,*HEIGTH               = *,7F10.3)
 2013 FORMAT( /,2X,*WETTED AREA          = *,7F10.3)
 2014 FORMAT( /,2X,*INTER.FACTOR         = *,7F10.3)
 2015 FORMAT( /,2X,*NO.OF THIS TYPE      = *,7F10.3)
 2016 FORMAT( /,2X,*MAXIMUM AREA         = *,7F10.3)
 2017 FORMAT( /,2X,*BASE AREA            = *,7F10.3)
 2018 FORMAT( /,2X,*NOSE LENGTH          = *,7F10.3)
 2019 FORMAT( /,2X,*BOATTAIL LENGTH      = *,7F10.3)
 2020 FORMAT( /,2X,*BASE DRAG AREA       = *,7F10.3)
 2021 FORMAT( /,2X,*INLET AREA           = *,7F10.3)
C
 2022 FORMAT(1H1,///,20X,7A10)
 2024 FORMAT(//,2X,*SECTION              = *,7(7X,A3))
 2025 FORMAT( /,2X,*CAMBER (2-D)         = *, 7F10.3)
 2026 FORMAT( /,2X,*THICKNESS RATIO      = *, 7F10.3)
 2027 FORMAT( /,2X,*L.E. SWEEP           = *, 7F10.3)
 2030 FORMAT( /,2X,*WETTED AREA          = *, 7F10.3)
 2031 FORMAT( /,2X,*ROOT CHORD           = *, 7F10.3)
 2032 FORMAT( /,2X,*TIP  CHORD           = *, 7F10.3)
 2033 FORMAT( /,2X,*SPAN OF PANEL        = *, 7F10.3)
 2034 FORMAT( /,2X,*X- LE ROOT CHORD= *, 7F10.3)
 2035 FORMAT( /,2X,*Y- LE ROOT CHORD= *, 7F10.3)
 2036 FORMAT( /,2X,*Z- LE ROOT CHORD= *, 7F10.3)
 2037 FORMAT( /,2X,*INCIDENCE            = *, 7F10.3)
C
 2040 FORMAT(///,10X,* ORDINATES FOR INPUT AIRFOIL*, //,
     1        * RLE =*,F7.4,5X,*TOCF =*,F7.4,5X,*CLDR =*,F7.4,5X,
     2        *TOCH =*,F6.3, //,
     3   2X,  *    XDIST/CHORD          CAMBER      THICKNESS * )
 2041 FORMAT(1X,3F14.4)
 2042 FORMAT(1H1,//,10X,I5,* DRAG POLARS TO BE GENERATED AT THE*,
     1     * FOLLOWING CONDITIONS*, /,T10,             *MACH NO.*,
     1        T20,*ALTITUDE*,T34,*C.G.*,T40,*SWEEP*,T50,*FROM CL*,
     1        T62,*TO CL*,/)
 2043 FORMAT(5X,F10.2,F13.2,F9.2,F7.1,F11.2,F11.2,I12)
 2044 FORMAT(//5X, A4, A10, 10X, 5H*****, /1H1 )
 2045 FORMAT( /,2X,*AIRFOIL SECTION SUR(*,I1,*,1) = *,A3,
     1            * NOT FOUND ON LIST* ///)
C
      END
```

```
      OVERLAY(1,2)
      PROGRAM DINPT
C
C     DAMAGE PARAMETERS ARE READ IN
C
C
C
      COMMON /INPUT/ NBODYS,NNAC,NPNLS,NHT,NVT,ISWP,IREF,ICDW,NAFC,
     1               METER, SREF, CMAC, XCG, ZCG, FIN,
     2               ROUGHK,FMISC,IDAMG, AR, TAPR, SWP, XOCMT, RLE,
     2               TOCR,CLER,IECH, DUM(2),
     3               BNAME(7), BOD(7,20), SNAME(7), SUF(7,20),
     4               AFX(30), AFC(30), AFT(30),
     5               XPIVOT, YPIVOT, XAPEX, AFTSW, AFTCR, AFTOC, AFTAW,
     6               NSURV, NCLAS, FMSURV(20), ALT(20), CG(20),
     7               SWPV(20), CLIC(20), CLPI(20), DHSV(20),
     8               IVAL(20), NXVAR, NADJ, NACL, NADJ2, X(15),
     9               YM(15,9), YA(15,9), XCL(15),
     1               NFUSPT, IFUSC, NOPCD, NFCTPT, XFUS, YFUS, ZFUS,
     2               FUSX(100), FUSR(100), FUSA(100),
     3               PODX(6), PODY(6), PODZ(6), DELX(6,30), DELA(6,30),
     4               IDAM(20), DBOD(7,100), DSUF(7,100), DWING(20)
C
      COMMON /BLKTIL/ TITLE(6)
C
      DIMENSION  WORD1(16), WORD2(16)
      DATA  WORD1 /10HBODY, ROUG,10HBODY,FWD-F,10HBODY,AFT-F,
     1     10HBODY, HOLE,10HBODY, WAVI,10HBODY, PROT,10HBODY, BLUN,
     2     1H ,1H ,1H ,10HSURF, ROUG,10HSURF,FWD-F,10HSURF,AFT-F,
     3     10HSURF, HOLE,10HSURF, WAVI,10HSURF, PROT /
      DATA  WORD2 /10HHNESS       ,10HACING STEP,10HACING STEP,
     1     10HS          ,10HNESS      ,10HUPERANCES ,10HTNESS     ,
     2     1H ,1H ,1H ,10HHNESS      ,10HACING STEP,10HACING STEP,
     3     10HS          ,10HNESS      ,10HUPERANCES  /
C
      NBODS  = NBODYS + NNAC
      NSUR   = NPNLS  + NHT  + NVT
C
C
C     ******** DAMAGE PARAMETERS ********
C
C     READ TITLE FOR DAMAGE CASE
      READ(5,1004)  (TITLE(I) , I = 1,6)
C
      WRITE(6,5010) (TITLE(I) ,  I = 1,6)
C     READ INDICATORS FOR DAMAGE INPUT
C
C
      DO 100   I = 1, 20
      IDAM(I) = 0
  100 CONTINUE
      READ(5,1008) (IDAM(I) ,    I = 1,17)
      WRITE (6,3001)
      DO 210  I = 1, 7
      WRITE (6,3002)  I,IDAM(I), I+10,IDAM(I+10)
  210 CONTINUE
C
C     FUSELAGE  DAMAGE PARAMETERS
C
      NBCASE=IDAM(1)+IDAM(2)+IDAM(3)+IDAM(4)+IDAM(5)+IDAM(6)+IDAM(7)
      IF (NBCASE.EQ.0)    GO TO 235
      WRITE (6,3003)
      READ(5,4002)    DUMMY
      DO 230  K = 1,  7
      IF (IDAM(K).NE.1)   GO TO 230
      IF (K.EQ.1)    NVARI   =  5
      IF (K.EQ.2)    NVARI   =  5
      IF (K.EQ.3)    NVARI   =  5
      IF (K.EQ.4)    NVARI   =  8
      IF (K.EQ.5)    NVARI   =  7
      IF (K.EQ.6)    NVARI  = 5
      IF (K.EQ.7)    NVARI   =  2
      K1   =(K-1)*10 + 1
      K2   =(K-1)*10 + NVARI
      WRITE(6,4003)  K, WORD1(K), WORD2(K)
      DO 220  J =  K1 , K2
      READ (5,1002) (DBOD(I,J) , I = 1, NBODS)
```

```
C
      WRITE(6,3004)    J ,(DPCD(I,J) , I = 1, NPCDS)
  220 CONTINUE
  230 CONTINUE
C
C     SURFACE DAMAGE PARAMETERS
C
      WRITE(6,5010) (TITLE(I) ,  I = 1,6)
  235 WRITE(6,3005)
      NNCASE=IDAM(11)+IDAM(12)+IDAM(13)+IDAM(14)+IDAM(15)+IDAM(16)
      IF (NNCASE.EQ.0)    GO TO 255
      READ(5,4002)    DUMMY
      DO 250  K = 11, 16
      IF (IDAM(K).NE.1)  GO TO 250
      IF (K.EQ.11)    NVARI = 6
      IF (K.EQ.12)    NVARI = 6
      IF (K.EQ.13)    NVARI = 6
      IF (K.EQ.14)    NVARI  = 9
      IF (K.EQ.15)    NVARI  = 7
      IF (K.EQ.16)    NVARI  = 5
      K1  = (K - 11) * 10  + 1
      K2  = (K - 11) * 10  + NVARI
      WRITE(6,4003)  K, WORD1(K), WORD2(K)
      DO 240  J =K1,K2
      READ(5,1002)    (DSUR(I,J), I= 1, NSUR)
C
      WRITE(6,3004)    J ,(DSUR(I,J) , I = 1,   NSUR)
  240 CONTINUE
  250 CONTINUE
C
C     MISSING PARTS OF SURFACES
C
  255 DWING(13) = 0.999999
      IF (IDAM(17).NE.1) GO TO 260
      READ(5,1002)  (DWING(I)   , I = 1, 6 )
      READ(5,1002)  (DWING(I)   , I = 7,10)
      READ(5,1002)  (DWING(I)   , I =11,13)
      READ(5,1002)  (DWING(I)   , I =14,20)
      WRITE(6,3006)     DWING(1), DWING(07),DWING(13)
      WRITE(6,3007)     DWING(2), DWING(08),DWING(14)
      WRITE(6,3008)     DWING(3), DWING(09),DWING(15)
      WRITE(6,3009)     DWING(4), DWING(10),DWING(16)
      WRITE(6,3010)     DWING(5), DWING(11),DWING(17)
      WRITE(6,3011)     DWING(6), DWING(12),DWING(18)
      WRITE(6,3012)   DWING(19)
      WRITE(6,3013)  DWING(20)
      IF (DWING(13).EQ.0.0)   WRITE(6,5020)
      IF (DWING(13).EQ.0.0)   DWING(13) = 0.5
  260 CONTINUE
C     CONVERT ROUGHNESS HEIGHT FROM INCHES TO FT OR METERS
C
      AA     = 1./12.
      IF (METER.EQ.1) AA = 0.0254
      DO 300 I =1,7
      DPCD(I,1) = DPCD(I,1) * AA
      DPCD(I,2) = DPCD(I,2) * AA
      DSUR(I,2) = DSUR(I,2) * AA
      DSUR(I,1) = DSUR(I,1) * AA
  300 CONTINUE
C
C
C              INPUT FORMAT STATMENT
C
 1001 FORMAT(14I5)
 1002 FORMAT(7F10.0)
 1004 FORMAT(7A10)
 1005 FORMAT(A3,7X,A3,7X,A3,7X,A3,7X,A3,7X,A3,7X,A3)
 1006 FORMAT(A4,A10)
 1007 FORMAT(20I1, 4I5)
 1008 FORMAT(20I1)
C
C
C    FORMAT STATEMENTS FOR DAMAGE PARAMETERS
C
 3001 FORMAT(  //, 30X, *DAMAGE PARAMETERS *, / )
 3002 FORMAT (10X,* IDAM(*,I2,*) =*, I2, 10X,
     1               * IDAM(*,I2,*) =*, I2)
```

```
C
 3003 FORMAT(//,25X,* BODY DAMAGE PARAMETERS, DBOD(I,J) *,//,
     1          13X,*I = 1*,5X,*I = 2*,5X,*I = 3*,5X,*I = 4*,5X,
     2              *I = 5*,5X,*I = 6*,5X,*I = 7*,/)
 3004 FORMAT (2X, *J =*,I3,3X, 7F10.6)
 3005 FORMAT(  //, 25X,*SURFACE DAMAGE PARAMETERS, DSUF(I,J)*,//,
     1          13X,*I = 1*,5X,*I = 2*,5X,*I = 3*,5X,*I = 4*,5X,
     2              *I = 5*,5X,*I = 6*,5X,*I = 7*,/)
 3006 FORMAT ( //, 20X, * SURFACE MISSING PART PARAMETERS, DWING(I)*,
     1        //,15X,*DWING(1)  =*,F8.3,5X,*DWING( 7) =*,F8.3,5X,
     2              *DWING(13) =*, F8.3)
 3007 FORMAT(15X,*DWING(2)  =*,F8.3,5X,*DWING( 8) =*,F8.3,5X,
     1              *DWING(14) =*,F8.3)
 3008 FORMAT(15X,*DWING(3)  =*,F8.3,5X,*DWING( 9) =*,F8.3,5X,
     1              *DWING(15) =*,F8.3)
 3009 FORMAT(15X,*DWING(4)  =*,F8.3,5X,*DWING(10) =*,F8.3,5X,
     1              *DWING(16) =*,F8.3)
 3010 FORMAT(15X,*DWING(5)  =*,F8.3,5X,*DWING(11) =*,F8.3,5X,
     1              *DWING(17) =*,F8.3)
 3011 FORMAT(15X,*DWING(6)  =*,F8.3,5X,*DWING(12) =*,F8.3,5X,
     1              *DWING(18) =*,F8.3)
 3012 FORMAT (62X, *DWING(19) =* , F8.3)
 3013 FORMAT (62X, *DWING(20) =* , F8.3)
 4002 FORMAT(A2)
 4003 FORMAT ( /, 10X,*MODE *, I2, 8X, 2A10)
 5010 FORMAT (1H1, //, 10X, 6A10)
 5020 FORMAT ( ////,30X, *I N P U T    E R R O R *, //, 20X,
     1          *MOMENT ARM OF DEVICE THAT TRIMS ROLLING MOMENT, *,
     2          *DWING(13), WAS INPUT AS 0.0 *, /, 20X,
     3          *PROGRAM WILL SET DWING(13) = 0.5  AND CONTINUE PROBLEM*)
C
      END
```

```
      OVERLAY(1,3)
      PROGRAM PINPT
C
C     READ IN INPUT VARIABLES TO CHANGE PREVIOUS PROBLEM
C
      COMMON /INPUT/  A(3081)
      COMMON /BLKTIL/ TITLE(6)
C
      DIMENSION NO(30),AA(30), MO(14),IA(14), I1(10),I5(24)
      EQUIVALENCE (I1(1),A(1)), (I2,A(18)), (I3,A(491)), (I4,A(492)),
     1            (I5(1),A(633)), (I6,A(22))
C
      READ(5,1000)  (TITLE(I), I=1,6)
      READ(5,1001)  M, N
      WRITE(6,2000) (TITLE(I), I=1,6)
C
      IF (M.EQ.0) GO TO 10
      READ(5,1001)  ( (MO(I), IA(I)),   I=1,M)
 10   IF (N.EQ.0) GO TO 20
      READ(5,1002)  ( (NO(I), AA(I)),   I=1,N)
C
 20   IF (N.EQ.0) GO TO 110
      DO 100  I =1,N
      J       = NO(I)
      A(J)    = AA(I)
 100  CONTINUE
C
 110  IF (M.EQ.0) GO TO 210
C
      DO 200  I =1,M
      J       = MO(I)
      IF (J.LE.10) I1(J) = IA(I)
      IF (J.EQ.18) I2    = IA(I)
      IF (J.EQ.22) I6    = IA(I)
      IF (J.EQ.491) I3   = IA(I)
      IF (J.EQ.492) I4   = IA(I)
      IF (J.GE.633) I5(J)= IA(I)
 200  CONTINUE
C
 210  CONTINUE
C
 1000 FORMAT( 6A10 )
 1001 FORMAT(14I5)
 1002 FORMAT( 3(I5,5X,F10.0) )
C
 2000 FORMAT(1H1,///,27X, 6A10)
      END
```

```
      OVERLAY(2,0)
      PROGRAM GEOM
C
C     COMPUTES GEOMETRIC PARAMETERS FOR AERO CALCULATIONS
C
      COMMON /CALC/ BO2, DOP, CR, SPLAN, XCRTE,
     1              SWPCO, SWPMC, SWPTE, SWMT, SEXW, CBAR,
     2              TOCW,CLD,ARXW,CFX,CTX,XFX,XTX,
     3              YFX,YTX,XF,YF,SWET,XYZ, SWPLF,FOC,
     4              SWPF,KPASS,ARPLAN,TF,Y1X,S1X,ARI,CRXP,
     5              SOXF,APOP,OXCC,TWIST,WINC,XHT,TOCS,CPAR2,CLDS,YCX,
     6              SWPMCS, C(5)
C
      COMMON /INPUT/ A(3061)
      COMMON /BLKDAT1/ D(1411)
      DIMENSION   BNAME(7), BOD(7,20), SNAME(7), SUR(7,30)
     1            , W(7),X(7),Y(7),Z(7), AFC(30), XTT(22)
C
      EQUIVALENCE  (A(1),NBODYS),(A(2),NNAC),(A(3),NPNLS),(A(10),METER),
     1             (A(4),NHT)    ,(A(5),NVT ),(A(12),CMAC),
     2             (A(13),XCG), (A(14),ZCG), (A(8),IWNC),
     3              (A(19),WPLAN), (A(20),TAPR), (A(21),SWP),
     4          (A(30),BNAME(1)),(A(37),BOD(1,1)),(D(111),XTT(1)),
     5             (A(424),AFC(1)),(A(25),CLDE),(A(9),NAFC),
     5           (A(177),SNAME(1)),(A(184),SUR(1,1))
C
C     DEFINE SUBSCRIPTS:
C       BODIES              J1=1            TO    J2=NBODYS
C       NACELLES            K1=NBODYS+1     TO    K2=NBODYS + NNAC
C       MAIN WING           L1=1            TO    L2=NPNLS
C       HORIZ. SURFACES     M1=NPNLS+1      TO    M2=NPNLS+NHT
C       VERT.  SURFACES     N1=NPNLS+NHT+1  TO    N2=NPNLS+NHT+NVT
C
      J2  =  NBODYS
      K2  =  NBODYS  + NNAC
      M1  =  NPNLS   + 1
      M2  =  NPNLS   + NHT
      N1  =  NPNLS   + NHT  + 1
      N2  =  NPNLS   + NHT  + NVT
      KPASS  = 0
C
C     **********CALCULATIONS FOR BODIES*********
C
      IF (K2.EQ.0) GO TO 110
C      FINENESS RATIO   =  BOD(I,14)
      DO 100    I = 1, K2
      BDIA2    =  BOD(I,2)  * BOD(I,3)
      IF (BDIA2.EQ.0.0)   BOD(I,14) = 0.0
      IF (BDIA2.GT.0.0)   BOD(I,14) = BOD(I,1) / SQRT(BDIA2)
C
C     MAXIMUM CROSS SECTIONAL AREA = BOD(I,7)
      IF (BOD(I,7).EQ.0.0) BOD(I,7) =0.7854 * BDIA2
C
C     WETTED AREA = BOD(I,4)
      IF (BOD(I,4).EQ.0.0.AND.I.LE.J2) BOD(I,4)= (2.5 *BOD(I,9)
     1        + 2.5*BOD(I,10)*(1.+SQRT(BOD(I,8)/BOD(I,7)))
     2        + (BOD(I,1)- BOD(I,9) - BOD(I,10)) * 4.) *
     3        SQRT(0.7854 * BOD(I,7)) * BOD(I,6)
      IF (BOD(I,4).EQ.0.0.AND.I.GT.J2) BOD(I,4)= (2.5 *BOD(I,9)
     1        *(1.+SQRT(BOD(I,12)/BOD(I,7))) + 2.5 * BOD(I,10)
     2        *(1.+ SQRT(BOD(I,8) / BOD(I,7)))
     3        +(BOD(I,1)- BOD(I,9) -BOD(I,10)) * 4.) *
     4        SQRT(0.7854 * BOD(I,7)) * BOD(I,6)
  100 CONTINUE
C
C     ******CALCULATE GEOMETRY FOR INDIVIDUAL WING PANELS********
C
  110 IF (N2.EQ.0) GO TO 130
C     TAPER RATIO = SUR(I,17)
      DO 200  I = 1 , N2
      SUR(I,17) =   SUR(I,9) / SUR(I,8)
C
C     X/C FOR MAX.T/C = SUR(I,5)
      ID     = SUR(I,1)
      SUR(I,5) = XTT(ID)
C
C     PLANFORM AREA= SUR(I,18)          (EXPOSED)
```

```
      SUR(I,18)   = ((SUR(I,8)+ SUR(I,9)) / 2.) * SUR(I,10)
      IF (I.LE.M2)  SUR(I,18) =  2. * SUR(I,18)
C
C     ASPECT RATIO  = SUR(I,19)
      IF (I.LE.M2) SUR(I,19) = (4.*SUR(I,10)**2) / SUR(I,18)
      IF (I.GE.M1) SUR(I,19) = (4.*SUR(I,10)**2) / (2.*SUR(I,18))
C
C     SYMMETRY CONDITION IF VERTICAL HAS Y DISPLACEMENT
      IF (I.GE.M1.AND.SUR(I,12).GT.0.0) SUR(I,18) = 2.*SUR(I,18)
C     WETTED AREA = SUR(I,7)
      IF (SUR(I,7).EQ.0.0)  SUR(I,7) = SUR(I,18)* (2.
     1     +0.1843 * SUR(I,3) + 1.5268 * SUR(I,3)**2
     2     -0.8295 * SUR(I,3)**3)
C
C     CHARACTERISTIC LENGTH = SUR(I,20)
      SUR(I,20)= 0.6667 * SUR(I,8) * (1. + SUR(I,17)**2
     1              / (1. + SUR(I,17)))
C *** TOCS,CLDS,AND CBAR2 ARE VARIABLE SWEEP PARAMETERS ***
      TOCS   = SUR(NPNLS,3)
      CLDS   = SUR(NPNLS,2)
      CBAR2  = SUR(NPNLS,20)
      SWET   = SUR(NPNLS,7)
C
C
C     SWEEP ANGLES      SUR(I,21) AT C/4
C                       SUR(I,22) AT C/2
C                       SUR(I,23) AT TE
C                       SUR(I,24) AT MAX THICKNESS
      TSWP  =   TAN(SUR(I,4))
      CO    =    4. / SUR(I,19) * (1.-SUR(I,17))/ (1.+ SUR(I,17))
      SUR(I,21)   =   ATAN(TSWP - CO   * 0.25)
      SUR(I,22)   =   ATAN(TSWP - CO   * 0.50)
      SUR(I,23)   =   ATAN(TSWP - CO   * 1.00)
      SUR(I,24)   =   ATAN(TSWP - CO * SUR(I,5))
      SWPMCS = SUR(NPNLS,22)
C
 200  CONTINUE
C
C     ********GEOMETRY FOR SINGLE PANEL WINGS **********
C     ********TOTAL AIRPLANE PARAMETERS           **********
C
 9001 FORMAT (1H1, 8F10.3)
      TWIST = A(15)
      WINC  = A(275)
C
      IF (NPNLS.EQ.2) GO TO 120
      RO2     =   SUR(1,10) + SUR (1,12)
      COF     =   SUR(1,12) / RO2
      CR      =   SUR(1,9)+(SUR(1,8)-SUR(1,9))*RO2 / SUR(1,10)
      SPLAN   =  (CR + SUR(1,9)) * RO2
      ARPLAN  =  4.*RO2**2/SPLAN
      TR      =  SUR(1,9)/CR
      TAPR    = TR
      XCRTE   =   SUR(1,11) - SUR(1,12)   *   TAN(SUR(1,4)) + CR
      SWF     =   SUR(1,4)
      SWPR    = SWP
      SWPLE   = SWP
      SWPCC   =   SUR(1,21)
      SWPMC   =   SUR(1,22)
      SWPTE   =   SUR(1,23)
      SWMT    =   SUR(1,24)
      SEXW    =   SUR(1,18)
C     CBAR    =   EXPOSED WING AERODYNAMIC CHORD
      CBAR    =   SUR(1,20)
      TOCW    =   SUR(1,3)
      CLD   =   SUR(1,2)
      ARXR  =   4. * SUR(1,10)**2 / SUR(1,18)
      CRX   =     SUR(1,8)
      CTX   =     SUR(1,9)
      XRX     = SUR(1,11)
      XTX     =   SUR(1,11) + SUP(1,10) * TAN(SUR(1,4))
      YRX     = SUR(1,12)
      YTX     = RO2
      DXCC    = SUR(1,10) *TAN(SWPCC)
C
C
C
```

```
C      MOMENT GEOMETRY,
C      CB = MAC DEFINED TO CENTERLINE
       CB = 0.6667 * CR * (1.+ TAPR**2 / (1. + TAPR))
       IF (CMAC.EQ.0.0) CMAC = CB
       YB = BO2 /3.  * (1.+ 2. * TAPR) / (1. + TAPR)
       XB       = SUR(1,11) + (YB - SUR(1,12)) * TAN(SWP)
C
       IF (NPNLS.EQ.1) GO TO 125
C
C
C      ********** GEOMETRY FOR MAIN WING WITH TWO PANELS ********
C      **********       TOTAL AIRPLANE PARAMETERS          ********
C
  120  BO2      = SUR(2,12) + SUR(2,10)
       DOB      = SUR(1,12) / BO2
       CR       = SUR(1,9) +(SUR(1,8)-SUR(1,9))*(SUR(1,12)+SUR(1,10))/
      1                                                 SUR(1,10)
       SPLAN = WPLAN
       TR       = TAPR
       IF (IWNG.GT.0) GO TO 122
C
       SPLAN = SUR(2,18) + (CR +SUR(1,9))*(SUR(1,12)+SUR(1,10))
       TR       = BO2 * SUR(2,9)/(SPLAN -BO2*SUR(2,9))
  122  ARPLAN = 4.*BO2**2/SPLAN
       SWPR   = SUR(2,4)
       IF (IWNG.GT.0) SWPR = SWP
C
C      SPLAN IS INPUT WHEN IWNG=1
C      TAPR     IS INPUT FOR TWO-PANEL WING
C      SWP      IS INPUT FOR TWO-PANEL WING
C
C
       SEXW     =   SUR(1,18)  + SUR(2,18)
       SWPLE    = ATAN((TAN(SUR(1,4) )*SUR(1,18)+ TAN(SUR(2,4) )*SUR(2,18))
      1                /SEXW )
       SWPQC    = ACOS((COS(SUR(1,21))*SUR(1,18)+ COS(SUR(2,21))*SUR(2,18))
      1                /SEXW )
       SWPMC    = ACOS((COS(SUR(1,22))*SUR(1,18)+ COS(SUR(2,22))*SUR(2,18))
      1                /SEXW )
       SWPTE    = ATAN((TAN(SUR(1,23))*SUR(1,18)+ TAN(SUR(2,23))*SUR(2,18))
      1                /SEXW )
       SWMT     = ACOS((COS(SUR(1,24))*SUR(1,18)+ COS(SUR(2,24))*SUR(2,18))
      1                /SEXW )
C
C      XCRTE IS DEFINED AS THE X-STA. OF T.E. OF EQUIV. WING ROOT CHORD
       XCRTE    =   SUR(2,11) + SUR(2,10)*TAN(SUR(2,4))
      1             - BO2 * TAN(SWPLE) + CR
C
C      CBAR     =   EXPOSED WING AERODYNAMIC CHORD,  HANDBOOK VII.B.0-7
       CBAR     = (SUR(1,20)*SUR(1,18) + SUR(2,20)*SUR(2,18)) /
      1              (SUR(1,18) + SUR(2,18))
C
C
C      TOCW AND CLD FOR A WING WITH 2 PANELS ARE AREA WEIGHTED
C
       TOCW     = SQRT((SUR(1,18)*SUR(1,3)**2 + SUR(2,18)*SUR(2,3)**2)
      1             /(SUR(1,18) + SUR(2,18)) )
       CLD      = SQRT((SUR(1,18)*SUR(1,2)**2 + SUR(2,18)*SUR(2,2)**2)
      1             /(SUR(1,18) + SUR(2,18)) )
       ARXR     =  4.  *((SUR(1,10)+SUR(2,10))**2) /(SUR(1,18)+SUR(2,18))
C
C      PARAMETERS BELOW ARE FOR EXPOSED WING
       CRX      =  SUR(1,8)
       CBX      =  SUR(1,9)
       CTX      =  SUR(2,9)
       XTX      =  SUR(2,11) + SUR(2,10) * TAN(SUR(2,4))
       XRX      =   XTX  - (BO2 -      SUR(1,12)) * TAN(SWPLE)
       YRX      =  SUR(1,12)
       YTX      =  BO2
       DXCC     = SUR(1,10) *TAN(SUR(1,21)) + SUR(2,10) *TAN(SUR(2,21))
C
C      MOMENT GEOMETRY
C
C      CB       = MAC DEFINED TO CENTERLINE
       CB       = 0.6667 * CR *(1.+ TAPR**2 / (1. + TAPR))
       YB       = BO2 /3. * (1.+2.*TAPR)/(1.+TAPR)
       XB       =  XTX - (BO2 - YB) * TAN(SWPLE)
```

```
C
C
C
C     ****** CALCULATIONS FOR MOMENT PARAMETERS *******
C
 125  SIX    = SUR(1,18)
      YIX    = SUR(1,10)
      ARI    = SUR(1,19)
      IF (NPNLS.EQ.1) GO TO 126
C
      YOX    = SUR(2,10) + 0.5 * YIX
      CRXP   = (YOX * SUR(2,8) - 0.5 * YIX * SUR(2,9))/SUR(2,10)
      SOXP   = (CRXP +SUR(2,9)) *YOX
      AROP   = 4.*YOX**2/SOXP
C
C     HORIZONTAL  TAIL PARAMETERS
C
 126  XH     = 0.0
      OMEGA  = 0.0
      IF (CMAC.EQ.0.0) CMAC = CB
      IF (XCG.EQ.0.0) XCG = XB
      IF (NHT.EQ.0) GO TO 130
C
C     LOCATE  H.T.  MAC/4  (XHT)
      XHT =  SUR(M1,10)/3.   * (1.+ 2.* SUR(M1,17)) / (1.+ SUR(M1,17))
     1           * TAN(SUR(M1,4)) + SUR(M1,20) * 0.25  + SUR(M1,11)
C
C     TAIL LENGTH (XH)
      XH     = SQRT((SUR(M1,13)-ZCG)**2 +(XHT-XCG)**2)
      OMEGA  = ATAN2((SUR(M1,13)-ZCG) , (XHT -XCG))
C
C
C
C
  130 WRITE(6,2002) (PNAME(I) , I = 1,  K2)
      WRITE(6,2003)  (POD(I,14), I = 1,  K2)
      IF (METER.LT.2) WRITE(6,2004)  (BOD(I,4), I = 1, K2)
      IF (METER.LT.2) GO TO 132
      DO 131  I = 1, 7
      W(I)       = BOD(I,4) / 144.
      X(I)       = SUR(I,7) / 144.
      Y(I)       = SUR(I,18)/ 144.
  131 CONTINUE
  132 CONTINUE
      IF (METER.EQ.2) WRITE(6,2004)  (W(I) , I = 1, K2)
      WRITE(6,2005)  (BOD(I,7) , I = 1,  K2)
      WRITE(6,2006)  (SNAME(I) , I = 1,  N2)
      IF (METER.LT.2) WRITE(6,2007)  (SUR(I,7), I =1,N2)
      IF (METER.EQ.2) WRITE(6,2007)  (X(I),     I = 1, N2)
      WRITE(6,2008)  (SUR(I,17), I = 1,  N2)
      IF (METER.LT.2) WRITE(6,2009)  (SUR(I,18),I =1,N2)
      IF (METER.EQ.2) WRITE(6,2009)  (Y(I),     I = 1, N2)
      WRITE(6,2010)  (SUR(I,19), I = 1,  N2)
      WRITE(6,2011)  (SUR(I,20), I = 1,  N2)
C
      DO 140   I = 1, N2
      W(I)    =   SUR(I,21) * 57.2958
      X(I)    =   SUR(I,22) * 57.2958
      Y(I)    =   SUR(I,23) * 57.2958
      Z(I)    =   SUR(I,24) * 57.2958
  140 CONTINUE
C
      WRITE(6,2012)  (W(I)  , I = 1, N2)
      WRITE(6,2013)  (X(I)  , I = 1, N2)
      WRITE(6,2014)  (Y(I)  , I = 1, N2)
      WRITE(6,2015)  (Z(I)  , I = 1, N2)
      WRITE(6,2116)  (SUR(I,5), I = 1, N2)
C
C     MAX. CAMBER ORDINATE (FOC)
      FOC    = 0.06651 * CLD
      IF (SUR(1,1).NE.8) GO TO 170
      FOCM   = AFC(1)
      DO 160 I =1,NAFO
      FOCM   = AMAX1(FOCM,AFC(I))
 160  CONTINUE
      FOC    = FOCM * CLD/CLDR
C
```

```
170   CONTINUE
      WRITE(6,2016)
      WRITE(6,2017)    BO2
      WRITE(6,2018)    DCB
      WRITE(6,2019)    CR
      IF (METER.LT.2) WRITE(6,2020) SPLAN
      S       = SPLAN/144.
      IF (METER.EQ.2) WRITE(6,2020) S
      WRITE(6,2021)    XCPTE
      WRITE(6,2022)    ARPLAN
      WRITE(6,2023)    TR
C
      S1      =   SWPLE   *   57.2958
      S2      =   SWPCC   *   57.2958
      S3      =   SWPMC   *   57.2958
      S4      =   SWPTE   *   57.2958
      S5      =   SWMT    *   57.2958
C
      WRITE(6,2025)    S1
      WRITE(6,2026)    S2
      WRITE(6,2027)    S3
      WRITE(6,4001)  S4
      WRITE(6,2028)  S5
C
      IF (METER.LT.2) WRITE(6,2029) SEXW
      S       = SEXW/144.
      IF (METER.EQ.2) WRITE(6,2029) S
      WRITE(6,2030)    CBAR
      WRITE(6,2031)    TCCW
      WRITE(6,2032)    CLD
      WRITE(6,2033)    ARXR
      WRITE(6,2034)    CRX
      WRITE(6,2036)    CTX
      WRITE(6,2037)    XRX
      WRITE(6,2039)    XTX
      WRITE(6,2040)    YRX
      WRITE(6,2042)    YTX
      WRITE(6,2043)    XB
      WRITE(6,2044)    YB
      WRITE(6,2045)    XH
C
      S1      =   OMEGA   *   57.2958
      WRITE(6,2046)    S1
 2002 FORMAT(1H1,/////,20X,*CALCULATED BODY PARAMETERS*,//,20X,7A10)
 2003 FORMAT (//,2X,*FINENESS RATIO     = * , 7F10.3)
 2004 FORMAT (/ ,2X,*WETTED AREA        = * , 7F10.3)
 2005 FORMAT (/ ,2X,*MAXIMUM AREA       = * , 7F10.3)
 2006 FORMAT (//////,20X,*CALCULATED SURFACE PARAMETERS*,//,20X,7A10)
 2007 FORMAT (/,2X, *WETTED AREA        = *, 7F10.3)
 2008 FORMAT (/,2X, *TAPER RATIO        = *, 7F10.3)
 2009 FORMAT (/,2X, *PLANFORM AREA      = *, 7F10.3)
 2010 FORMAT (/,2X, *ASPECT RATIO       = *, 7F10.3)
 2011 FORMAT (/,2X, *CHARACT. LENGTH    = *, 7F10.3)
 2012 FORMAT (/,2X, *SWP   C/4          = *, 7F10.3)
 2013 FORMAT (/,2X, *SWP   C/2          = *, 7F10.3)
 2014 FORMAT (/,2X, *SWP   TE           = *, 7F10.3)
 2015 FORMAT (/,2X, *SWP   MAX T/C      = *, 7F10.3)
 2116 FORMAT (/,2X, *X/C AT MAX T/C     =*, 7F10.3)
C
 2016 FORMAT (1H1,20X,*TOTAL AIRPLANE PARAMETERS*,//)
 2017 FORMAT (    2X, * WING SEMI-SPAN                        ,BO2   =*,
     1        F10.3)
 2018 FORMAT (    2X, * BODY DIA/ WING SPAN                   ,DCB   =*,
     1        F10.3)
 2019 FORMAT (    2X, * WING ROOT CHORD AT CENTERLINE         ,CR    =*,
     1        F10.3)
 2020 FORMAT (    2X, * THEO. WING PLANFORM AREA              ,SPLAN =*,
     1        F10.3)
 2021 FORMAT (    2X, * X-STA OF ROOT CHORD (CR) T.E.         ,XCPTE =*,
     1        F10.3)
 2022 FORMAT (    2X, * EQUIV. WING ASPECT RATIO              ,AR    =*,
     1        F10.3)
 2023 FORMAT (    2X, * EQUIV. WING TAPER  RATIO              ,TAPR  =*,
     1        F10.3)
C
 2025 FORMAT (    2X, * EQUIV. WING LEADING EDGE SWEEP        ,SWPLE =*,
     1        F10.3)
```

```
2026 FORMAT (   2X, * EQUIV. WING QUARTER-CHORD SWEEP    ,SWPCC =*,
    1          F10.3)
2027 FORMAT (   2X, * EQUIV. WING MID-CHORD      SWEEP    ,SWPMC =*,
    1          F10.3)
4001 FORMAT (   2X, * EQUIV. WING TRAILING-EDGE SWEEP    ,SWPTE =*,
    1          F10.3)
2028 FORMAT (   2X, * EQUIV. WING MAX.THICKNESS SWEEP    ,SWMT  =*,
    1          F10.3)
2029 FORMAT (   2X, * EQUIV. WING EXPOSED AREA            ,SEXW  =*,
    1          F10.3)
2030 FORMAT (   2X, * EQUIV. WING EXPOSED MAC             ,CBAR  =*,
    1          F10.3)
2031 FORMAT (   2X, * EQUIV. WING THICKNESS RATIO         ,TCCW  =*,
    1          F10.3)
2032 FORMAT (   2X, * EQUIV. WING 2-D DESIGN CL           ,CLD   =*,
    1          F10.3)
2033 FORMAT (   2X, * EQUIV. WING EXPOSED ASPECT RATIO    ,ARXR  =*,
    1          F10.3)
2034 FORMAT (   2X, * CHORD LENGTH AT WING ROOT           ,CRX   =*,
    1          F10.3)
2035 FORMAT (   2X, * CHORD LENGTH AT WING BREAK          ,CBX   =*,
    1          F10.3)
2036 FORMAT (   2X, * CHORD LENGTH AT WING TIP            ,CTX   =*,
    1          F10.3)
2037 FORMAT (   2X, * X-STA. AT L.E. OF WING ROOT         ,XRX   =*,
    1          F10.3)
2038 FORMAT (   2X, * X-STA. AT L.E. OF WING BREAK        ,XBX   =*,
    1          F10.3)
2039 FORMAT (   2X, * X-STA. AT L.E. OF WING TIP          ,XTX   =*,
    1          F10.3)
2040 FORMAT (   2X, * Y-STA. AT L.E. OF WING ROOT         ,YRX   =*,
    1          F10.3)
2041 FORMAT (   2X, * Y-STA. AT L.E. OF WING BRAKE        ,YBX   =*,
    1          F10.3)
2042 FORMAT (   2X, * Y-STA. AT L.E. OF WING TIP          ,YTX   =*,
    1          F10.3)
2043 FORMAT (   2X, * X-STA. AT L.E. OF WING MAC          ,XP    =*,
    1          F10.3)
2044 FORMAT (   2X, * Y-STA. AT L.E. OF WING MAC          ,YP    =*,
    1          F10.3)
2045 FORMAT (   2X, * DELTA-X  C/4 WING TO C/4 TAIL       ,XI    =*,
    1          F10.3)
2046 FORMAT (   2X, * INCLINATION LINE BET. WING AND TAIL,OMEGA *),
    1          F10.3)
     END
```

```
      OVERLAY(3,0)
      PROGRAM SURVEY
C
C     AERODYNAMIC SURVEY
C
      COMMON/INPUT/  A(3081)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/    C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ E(905)
      COMMON /BLKTIL/ TITLE(6)
C
      DIMENSION FMSURV(20),ALT(20),SWPV(20),CLLO(20),CLHI(20),
     1          CG(20), BNAME(7), SNAME(7), SUR(7,30)
      DIMENSION TCL(21), TCD(21), TCM(21), TA(21), TCLT(21), TCDT(21)
      DIMENSION TCTCD(5), CDSUR(7,5), CDBOD(7,5), TCTB(5), TCTS(5)
      DIMENSION TERM(5)
C
C
      EQUIVALENCE (A(6),ISWP),(A(7),IPEF),(A(491),NSURV),
     1          (A(492),NCLAS),(A(493),FMSURV(1)),(A(513),ALT(1)),
     2          (A(553),SWPV(1)),(A(573),CLLO(1)),(A(593),CLHI(1)),
     3          (A(533),CG(1)), (A(275),XINC), (A(30),BNAME(1)),
     4          (A(177),SNAME(1)), (A(1),NBODYS), (A(2),NNAC),
     5          (A(3),NPNLS), (A(4),NHT), (A(5),NVT), (A(13),XMAC),
     6          (A(12),CMAC), (A(14),ZCG), (A(164),SUR(1,1)),
     7          (A(22),ITRIM), (A(4),NHT), (A(10),METER)
C
      EQUIVALENCE (B(43),CL), (B(44),CD), (B(45),CM), (B(46),ALPHA),
     1            (B(143),H), (B(144),FM), (B(145),SWP  ),
     2            (B(146),JPASS), (B(60),ALC), (B(71),DE),
     3            (B(137),CDAFT), (B(48),CDL), (B(47),CDM),
     4            (B(49),CDR), (B(55),CLA), (B(106),CLDB),
     5            (B(107),CLMAX), (B(51),CLT), (B(56),CMC),
     6            (B(57),DCYCL),  (B(55),DELCL), (B(53),DH),
     7            (B(54),DK),   (B(52),CDT),   (B(154),CDSUR(1,1)),
     8            (B(159),CDBOD(1,1)), (B(50),CDPC), (B(117),CDC),
     9            (B(135),CDMISC), (B(62),CMCH), (B(63),XH),
     1            (B(64),OMEGA), (C(40),XHT), (B(65),XCG)
C
      DATA TERM /10HFRICTION  ,10HFORM      ,10HINTERF    ,
     1           10HWAVE      ,10HBASE      /
C
      IF (NSURV.LE.0) GO TO 600
      SWP2    = SWPV(1)
      NBODS = NBODYS + NNAC
      NSURS = NPNLS + NHT + NVT
      M1      = NPNLS + 1
C
      REWIND 10
C
      JPASS   = 0
      DO 500 L = 1, NSURV
      FM      = FMSURV(L)
      H       = ALT(L)
      SWEEP   = SWPV(L)
      CLLOW   = CLLO(L)
      XH      = 0.0
      OMEGA = 0.0
      XCG     = XMAC + CMAC *CG(L)
      IF (NHT.EQ.0)  GO TO 200
C
      XHTL    = XHT
      IF (FM.GT.1.0) XHTL = XHT + SUR(M1,20)*0.25
      XH      = SQRT((SUR(M1,13)-ZCG)**2 +(XHTL-XCG)**2)
      OMEGA =ATAN2((SUR(M1,13)-ZCG),      (XHTL-XCG) )
C
  200 CONTINUE
C
      XNCL    = NCLAS - 1
      DCL     = 0.0
      IF( XNCL.GT.0.0 ) DCL = (CLHI(L) - CLLO(L))/XNCL
C
      SWP     = SWEEP/57.2957796
      IF( ISWP.GT.0.AND.SWEEP.NE.SWP2 ) JPASS = 0
      SWP2    = SWEEP
C
```

```
      IF( ISWP.GT.0.AND.JPASS.EQ.0 ) CALL OVERLAY(4HOVLY,3,1)
C             PROGRAM VCEOM CALLED
C
C
      IF( JPASS.EQ.0 ) CALL OVERLAY(4HOVLY,3,2)
C             PROGRAM MCRIT CALLED
C
      CALL OVERLAY(4HOVLY,3,3)
C           PROGRAM AERDA CALLED
C
C
      IF( JPASS.EQ.2 ) JPASS = 1
C
      DO 300 M = 1, NCLAS
      XM1      = M-1
      CL       = CLLOW + DCL * XM1
C
      CALL OVERLAY(4HOVLY,3,4,6HRECALL)
C           PROGRAM AERDB CALLED
C
      IF (M.GT.1) GO TO 220
C
      WRITE (6,1005) (TITLE(K), K=1,6)
      RN      = -H *1.E6
      IF (ITRIM.EQ.1) GO TO 210
      IF (H.GE.0.0.AND.METER.NE.1) WRITE(6,1008) FM, H, SWEEP
      IF (H.GE.0.0.AND.METER.EQ.1) WRITE(6,1012) FM, H, SWEEP
      IF (H.LT.0.0.AND.METER.NE.1) WRITE(6,1009) FM, RN,SWEEP
      IF (H.LT.0.0.AND.METER.EQ.1) WRITE(6,1013) FM, RN,SWEEP
      IF (NPT.EQ.0) WRITE(6,1016)
      GO TO 220
  210 IF (H.GE.0.0.AND.METER.NE.1) WRITE(6,1000) FM, H, SWEEP, DH
      IF (H.GE.0.0.AND.METER.EQ.1) WRITE(6,1014) FM, H, SWEEP, DH
      IF (H.LT.0.0.AND.METER.NE.1) WRITE(6,1006) FM,RN, SWEEP, DH
      IF (H.LT.0.0.AND.METER.EQ.1) WRITE(6,1015) FM,RN, SWEEP, DH
C
  220 ANGLE   = ALPHA
      IF( IREF.EQ.1 ) ANGLE = ALPHA - WINC
C
      CDRA     = CDR + CDAFT
      CDLL    = CDRA + CDI
      DH      = DH + DE
      IF (ITRIM.EQ.0) WRITE(6,1001) CLT, CDT, DH, ANGLE, CDL, CDRA,
     1                              CL, CD
      IF (ITRIM.EQ.1) WRITE(6,1001) CL,  CD,  CM, ANGLE, CDI, CDR,
     1                              CDAFT, CDLL
C
      TCL(M) = CL
      TCD(M) = CD
      TCM(M)   =   CMOH
      TA(M)    = ANGLE
      TCLT(M)= CLT
      TCDT(M)= CDT
C
  300 CONTINUE
C
      ALOR     = ALO
      IF( IREF.EQ.1 ) ALOP   = ALO - WINC
      WRITE(6,1002) CLA, ALOP, CLDR, FK, DELCL, CLMAX, CMO, DCMCL
C
      WRITE(6,1003) (FNAME(I), I=1,NBODS)
      DO 310 J =1,5
      TOTB(J) = CDBOD(1,J)+CDBOD(2,J)+CDBOD(3,J)+CDBOD(4,J)+CDBOD(5,J)
     1                      +CDBOD(6,J)+CDBOD(7,J)
  310 WRITE(6,1004) TERM(J), TOTB(J), (CDBOD(I,J), I=1,NBODS)
C
      WRITE(6,1010) (SNAME(I), I=1,NSURS)
      DO 320 J =1,4
      TOTS(J) = CDSUR(1,J)+CDSUR(2,J)+CDSUR(3,J)+CDSUR(4,J)+CDSUR(5,J)
     1                      +CDSUR(6,J)+CDSUR(7,J)
  320 WRITE(6,1004) TERM(J), TOTS(J), (CDSUR(I,J), I=1,NSURS)
C
      WRITE(6,1011)  CDC, CDPD, CDMISC
      WRITE(6,1007) CDM
C
      IF (NPNLS.EQ.2.AND.FM.GT.1.0) WRITE(6,1020) SNAME(2),SNAME(1)
C
```

```
      WRITE (10) TCL, TCD, TCM, TA, TCLT, TCDT
      WRITE (10)  B
C
  500 CONTINUE
C
  600 CONTINUE
C
 1000 FORMAT(1X, *MACH NO. =*F6.3, 5X,*ALTITUDE =*F7.0,* FT.*, 5X,
     1         *SWEEP ANGLE =*F6.2,* DEG.*, 5X,*TAIL DEFL. (DH) =*F6.2,
     2         * DEG.* //T10,*CL*, T22,*TOTAL CD*, T38,*CM*, T53,*ALPHA*,
     3         T70,*CDL *    , T83,*CDR*, T97,*CDAFT*,
     4         T113,*CD(L+R+AFT)* , /    )
 1014 FORMAT(1X,*MACH NO. =*F6.3, 5X,*ALTITUDE = *F7.0,* METERS*, 5X
     1         *SWEEP ANGLE =*F6.2,* DEG.*, 5X,*TAIL DEFL. (DH) =*F6.2,
     2         * DEG.* //T10,*CL*, T22,*TOTAL CD*, T38,*CM*,T53,*ALPHA*,
     3         T70,*CDL *    , T83,*CDR*, T97,*CDAFT*,
     4         T113,*CD(L+R+AFT)* , /  )
 1001 FORMAT(5X,F8.3,2F15.5, F15.2, 4F15.5)
 1002 FORMAT(/10H      CLA =,F8.5,1X,*PER DEG.*,10H      ALO =,F8.5,1X,
     1    *DEG.*,  29X,10H     CLDR =,F8.5 /
     1         10H        K =,F8.5, 9X,              10H   DELCL =,F8.5 ,
     2              34X,10H   CLMAX =,F8.5 /
     3         10H      CMO =,F8.5,1H*,8X,     10H   CM/CL =,F8.5,1H*,
     4         10X, 12H(*TAIL OFF) , //   )
 1003 FORMAT(* DRAG BREAKDOWN ----- *,5X,7(A10,5X))
 1004 FORMAT(5X,A10,F7.5,7F15.5)
 1005 FORMAT(1H1, /1X, 6A10 )
 1006 FORMAT(1X, *MACH NO. =*F6.3, 5X,*RN/FT =*1PE15.4,5X,
     1         *L.E. SWEEP ANGLE =* 0PF6.2,* DEG.*,5X,*TAIL DEFL. (DH) =*
     2         F6.2,* DEG.*
     2                  //T10,*CL*, T22,*TOTAL CD*, T38,*CM*, T53,*ALPHA*,
     3         T70,*CDL *    , T83,*CDR*, T97,*CDAFT*,
     4         T113,*CD(L+R+AFT)* , /    )
 1015 FORMAT(1X, *MACH NO. =*F6.3, 5X,*RN/METER =*1PE15.4,5X,
     1         *L.E. SWEEP ANGLE =* 0PF6.2,* DEG.*,5X,*TAIL DEFL. (DH) =*
     2         F6.2,
     2         * DEG.* //T10,*CL*, T22,*TOTAL CD*, T38,*CM*,T53,*ALPHA*,
     3         T70,*CDL *    , T83,*CDR*, T97,*CDAFT*,
     4         T113,*CD(L+R+AFT)* , /   )
 1007 FORMAT(/5X, 10H   CDMIN =,F7.5 //)
 1008 FORMAT(1X, *MACH NO. =*F6.3, 5X,*ALTITUDE =*F7.0,* FT.*, 5X,
     1        *L.E. SWEEP ANGLE =*F6.2,* DEG.*,5X,*TRIMMED CONDITION*
     2        //T10,*CL*, T22,*TOTAL CD*, T38,*DH*, T53,*ALPHA*,
     3             T68,*CD LIFT*, T82,*CD R+AFT*, T97,*CL AT DH =0*,
     4         T113,*CD AT DH=0*, / )
 1012 FORMAT(1X, *MACH NO. =*F6.3,5X,*ALTITUDE =*F7.0,* METERS*, 5X,
     1        *L.E. SWEEP ANGLE =*F6.2,* DEG.*,5X,*TRIMMED CONDITION*
     2        //T10,*CL*, T22,*TOTAL CD*, T38,*DH*, T53,*ALPHA*,
     3             T68,*CD LIFT*, T82,*CD R+AFT*, T97,*CL AT DH =0*,
     4         T113,*CD AT DH=0*,/ )
 1009 FORMAT(1X, *MACH NO. =*F6.3, 5X,*RN/FT =*1PE15.4,5X,
     1         *L.E. SWEEP ANGLE =* 0PF6.2,* DEG.*,5X,*TRIMMED CONDITION*
     2        //T10,*CL*, T22,*TOTAL CD*, T38,*DH*, T53,*ALPHA*,
     3             T68,*CD LIFT*, T82,*CD R+AFT*, T97,*CL AT DH=0*,
     4        T113,*CD AT DH=0*, / )
 1013 FORMAT(1X, *MACH NO. =*F6.3, 5X,*RN/METER =*1PE15.4,5X,
     1         *L.E. SWEEP ANGLE = * 0PF6.2,* DEG.*,5X,*TRIMMED CONDITION*
     2        //T10,*CL*, T22,*TOTAL CD*, T38,*DH*, T53,*ALPHA*,
     3             T68,*CD LIFT*, T82,*CD R+AFT*, T97,*CL AT DH =0*,
     4         T113,*CD AT DH=0*,/ )
 1010 FORMAT(*                              *,5X,7(A10,5X))
 1011 FORMAT(5X,10HCAMBER    =,F7.5,
     1        /5X,10HDRAG RISE=,F7.5,
     2        /5X,10HMISC      =,F7.5   )
 1016 FORMAT(36X, *ELEVON* )
C
 1020 FORMAT (  ///, 5X, *NOTE.    WAVE DRAG FOR *,A10,
     1      * IS INCLUDED IN WAVE DRAG OF *, A10)
      END
```

```
      SUBROUTINE ADJUST(ID, ID2, XVAR, YVAR)
C
C     ADJUST Y-VARIABLE USING CORRELATION FACTORS
C
      COMMON /INPUT/ A(30F1)
      COMMON /BLKPRT/ KPRINT(50)
C
      DIMENSION IVAL(20),X(15),YM(15,9),YA(15,9),XCL(15)
C
      EQUIVALENCE (A(633),IVAL(1)), (A(657),X(1)), (A(672),YM(1,1)),
     1            (A(807),YA(1,1)), (A(942),XCL(1)),
     2            (A(653), NXVAR), (A(655), NXCL)
C
      YVAR1   = YVAR
      IF( ID.EQ.0 ) GO TO 300
   50 NV      = IVAL(ID)
C
      IF( NV.EQ.0 ) RETURN
  100 CALL LNTP(XVAR, VM, X,YM(1,NV), NXVAR, 2)
  200 CALL LNTP(XVAR, VA, X,YA(1,NV), NXVAR, 2)
      GO TO 500
C
  300 NV      = IVAL(ID2)
      IF( NV.EQ.0 ) RETURN
  400 CALL LNTP(XVAR, VM, XCL,YM(1,NV), NXCL, 2)
      CALL LNTP(XVAR, VA, XCL,YA(1,NV), NXCL, 2)
C
  500 YVAR    = YVAR1 * VM + VA
C
      IF( KPRINT(24).EQ.0 ) RETURN
      WRITE(6,1000) ID, YVAR1, XVAR, VM, VA, YVAR
 1000 FORMAT(5X,*ADJUST DUMP*, 5X, *ID =*,I5/1X,5F15.5 )
      RETURN
      END
```

```
      OVERLAY(3,1)
      PROGRAM VGEOM
C
C     COMPUTES GEOMETRY FOR VARIABLE-SWEEP CONFIGURATIONS
C
      COMMON /CALC/ RO2, DCP, CR, SPLAN, XCRTE,
     1              SWPCC, SWPMC, SWPTE, SWMT, SFXW, CBAR,
     2              TOCW,CLD,AFXR,CFX,CTY,XRY,XTX,
     3              YFX,YTX,XF,YR,SWET,YYZ, SWHLF,FOC,
     4              SWPV,KPASS,AWPLAN,TR,YJX,STX,AFT-CEXP,
     5              SOXP,AROP,CXOC,TWIST,WINC,XHT,TWMS,CBAR2,CLDS,
     6              YGX,SWPMCS,DA1,DAC1,DA2,DAC2,DTOC
      COMMON /BLKPRT/ KPRINT(50)
C
      COMMON /INPUT/ A(3081)
      COMMON /OUTPUT/ B(223)
C
      DIMENSION X(6), Y(6), XT(6), YT(6), D(6), P(6)
      DIMENSION SUR(7,30)
      EQUIVALENCE (A(184),SUR(1,1)), (NPNLS,A(3)), (AFTAW,A(490)),
     1            (AFTCR,A(488)),  (AFTOC,A(486)),  (AFTSW,A(487)),
     2            (XPIVOT,A(484)), (YPIVOT,A(485)), (SWPR,A(21))
C
      EQUIVALENCE (SWP, B(145))
C
C
      T       = ABS(SWP-SWPV)
C
      IF (KPASS.EQ.0.AND.T.LE.0.005)  GO TO 100
C
      IF (KPASS.EQ.1) GO TO 25
C
      DA1    = 0.0
      DAC1   = 0.0
      DA2    = 0.0
      DAC2   = 0.0
      DTOC   = 0.0
      SWPX   = SWP
      SWP    = SWPR
C
C
C
 25   N      = NPNLS
      Y(2)   = SUR(N,12) + SUR(N,10)
      ETA    = SUR(N,12)/Y(2)
      CR     = (SUR(N, 8) -ETA *SUR(N, 9))/(1.-ETA)
      TANSR  = TAN(SUR(N,23))
      DZ     = SUR(N,9) * TAN(A(15)/57.3)
      CPOCR  = 0.5 *( COS(SUR(N,4))/COS(SUR(N,4) -SUR(N,22))
     1               + COS(SUR(N,23))/COS(SUR(N,22)-SUR(N,23)) )
      X(1)   = SUR(N,11) - SUR(N,12) *TAN(SUR(N,4))
      X(4)   = X(1) + CR
      Y(1)    = 0.0
      Y(4)    = 0.0
      X(2)   = X(1) + Y(2) * TAN(SUR(N,4))
      X(3)   = X(2) + SUR(N,9)
      Y(3)    = Y(2)
      DO 30 I =1,4
      D(I)    = SQRT((X(I) - XPIVOT)**2 + (Y(I) - YPIVOT)**2)
      P(I)    = ATAN((X(I) - XPIVOT) / ( Y(I) - YPIVOT) )
      IF((Y(I) - YPIVOT).LE.0.0) P(I) = P(I) + 3.141592
 30   CONTINUE
C
 35   CONTINUE
      DSWP   = SWP - SWPR
      SWPV   = SWP
C
      DO 40 I =1,4
      XT(I)   = YPIVOT +D(I) * SIN(DSWP + P(I))
      YT(I)   = YPIVOT +D(I) * COS(DSWP + P(I))
   40 CONTINUE
C
      DXCC   = 0.75*XT(2) +0.25*X(3) - SUR(1,11) -SUR(1,8)*0.25
      SWPTE   = ATAN((XT(3) - XT(4))/(YT(3) - YT(4)))
      SWPLE = SWP
      CC      = TAN(SWPTE) / TAN(SWPLE )
      SWPCC   = ATAN((1.0 - (1.0 - CO)/4.0)* TAN(SWPLE) )
```

```
      SWPMC   = ATAN((1.0 - (1.0 - CC)/2.0)* TAN(SWPLE) )
      XFWD     = YT(2) - YT(2) * TAN(SWPLE)
      XAFT     = XT(3) - YT(3) * TAN(SWPTE)
      CRCLP   = XAFT - XFWD
      CPCC     = 0.5 * ( COS(SWPLE)/COS(SWPLE - SWPMC) + COS(SWPTE)/
     1                                      COS(SWPMC - SWPTE) )
      CLCS    = SUR(N,2) * CPCC/CPCCR
      TCCS    = SUR(N,3) * CPCC/CPCCR
      BC2     = YT(2)
      DCF     = SUR(1,12)/BC2
C
   45 CONTINUE
      TTIP    = 1.0 + YT(2)* TAN(SWPLE) * (CC - 1.0)/CRCLP
      TWIST = ATAN(DZ/(CRCLP*TTIP)) * 57.296
      WINC   = A(275) *(1.-TAN(DSWP)*TANSR) *COS(DSWP)
C
      C3       = CRCLP * (1.0 -(YT(3)/YT(2))*(1.0 - TTIP) )
      SPLAN = (CRCLP +C3) *YT(3) +C3 *(YT(2)-YT(3))
      ARPLAN = 4.*BC2**2/SPLAN
      YMID    = YT(3) +(YT(2) - YT(3)) /(1.0/SQRT(C3/(TTIP*CRCLP)) + 1.0)
      TR       = 1. - (YMID/YT(2)) * (1.-TTIP)
      YWC     = SUR(NPNLS,12)
C
      XTE2    = XT(4) +YWD * TAN(SWPTE)
      CRX2    = CRCLP * (1.-YWD/YT(2) *(1.-TTIP))
      SX2      = SPLAN - (CRCLP +CRX2) *YWC
      STOTAL = SX2
      TOCW    = TCCS
      CLD      = CLDS
      SWPMCS = SWPMC
      IF( NPNLS.EQ.1 ) GO TO 46
C
      S1         = (XT(4)-XT(3))/(YT(4)-YT(3))
      S2         = (SUR(1,11)+SUR(1,8) -SUR(2,11)-SUR(2,8))/SUR(1,10)
      IF (S1.EQ.S2) GO TO 46
C     SOLVE FOR TE LINE INTERSECTION        YI              **************
C
      YI         = (XT(4)-SUR(1,11)-SUR(1,8) +S1*YT(4) -S2*SUR(1,12))/
     1                                    (S1-S2)
      DA         = (XTE2 -SUR(2,11)-SUR(2,8)) *(SUR(2,12) -YI)
      SX2       = SX2 + DA
      STOTAL  = SUR(1,18) + SX2
      TOCW    = SQRT((SUR(1,3)**2*SUR(1,18) +TCCS**2*SX2)/STOTAL)
      CLD      = SQRT((SUR(1,2)**2*SUR(1,18) +CLDS**2*SX2)/STOTAL)
C
   46 SWET     = SX2 * (2.+.1843*TCCS +1.5268*TCCS**2 -.8395*TCCS**3)
     1            +CFA1 +(CFA2-CFA1) * (SWPLE-SWPR)/(AFTSW-SWPR)
      TP         = TR * CRCLP/CRX2
      CBAR2    = 0.66667 * CRX2 * (1. + (TP**2/(1.+TP)) )
      CBAR2    = CBAR2 + CAC1 + (CAC2 - CAC1) * DSWP/(AFTSW-SWPR)
C
      SWMT    = SUR(N,24) + DSWP
      TOCW     = TOCW + DTOC * DSWP/(AFTSW - SWPR)
C
      IF (NPNLS.EQ.1) GO TO 47
C
      COSMC    = COS(SUR(1,22)) *SUR(1,18) +COS(SWPMC)*SX2
      COSCC    = COS(SUR(1,21)) *SUR(1,18) +COS(SWPCC)*SX2
      TANLE     = TAN(SUR(1, 4)) *SUR(1,18) +COS(SWPLE)*SX2
      TANTE     = TAN(SUR(1,23)) *SUR(1,18) +COS(SWPTE)*SX2
      COSMT    = COS(SUR(1,24)) *SUR(1,18) +COS(SWMT) *SX2
C
      SWPMC    = ACOS(COSMC/STOTAL)
      SWPCC    = ACOS(COSCC/STOTAL)
      SWPLE    = ATAN(TANLE/STOTAL)
      SWPTE    = ATAN(TANTE/STOTAL)
      SWMT     = ACOS(COSMT/STOTAL)
C
   47 CONTINUE
C
      SFXW     = STOTAL
      CRX       = CRX2
      CTX       = CRCLP * TP
      SCX       = SX2
      YOX       = YMID - YWC
      IF (NPNLS.EQ.1) GO TO 48
      YW2       = (SUR(1,12)+SUR(2,12))*0.5
```

```
      CRXP   = CRCLP *(1.-YW2 *(1.-TR)/YMID)
      SCXP   = (CRXP +CTX) *(YMID -YW2)
      ARCP   = 4.*(YMID -YW2)**2/SCXP
 48   ARXP   = 4.*(YMID -SUR(1,12))**2/SEXW
      FCC    = 0.06651 * CLD
C
      IF (KPASS.EQ.1) GO TO 50
C
      IF (KPASS.EQ.2) GO TO 49
      DA1    = SUR(N,7) -SWET
      DAC1   = SUR(N,20) -CBAR2
      SWP    = AFTSW
      KPASS = 2
      GO TO 35
 49   CONTINUE
C
      IF( AFTAW.GT.0.0 ) DA2 = AFTAW - SWET
      IF( AFTCB.GT.0.0 ) DAC2 = AFTCB - CBAR2
      IF( AFTCC.GT.0.0 ) DTCC = AFTCC - TCCS
      KPASS = 1
      SWP    = SWPX
      GO TO 35
   50 CONTINUE
C
      IF (KPRINT(12).EQ.1) WRITE(6,1000) KPASS, SWPV, DA1, DAC1,
     1              CPRCF, DA2, DAC2, DTCC, DA, SWPMCS, CPCC,
     2              TWIST,WINC,SWMT,CBAR2,CLDS,TCCS,SWET,DCB,TCCW,
     3              CLD,SEXW,SWPLE,SWPCC,SWPMC,SWPTE,ARPLAN,TR,SPLAN
 1000 FORMAT(5X, *VGEOM DUMP,  KPASS =*,I3, 10X,*SWPLE =* F12.4 /
     1            (1X, 7F15.5 ) )
C
 100  CONTINUE
      END
```

```
      OVERLAY(3,2)
      PROGRAM MCRIT
C
C     COMPUTES CRITICAL MACH NUMBER
C
      COMMON/INPUT/  A(3081)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/   C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ E(908)
      COMMON /BLKMCR/ CLMCR(10), XMCR(10), NMCR, SA(32), SB(32),
     1                SC(32), SD(32), SE(32), XL(32), ZT(32), ZS(32)
     1               ,  BOC
C
      DIMENSION CLTAB(21), TABMCR(21), BCD(7,20)
      EQUIVALENCE (B(1),CLTAB(1)), (B(22),TABMCR(1)),(A(37),BCD(1,1))
     1            , (A(3),NPNLS), (B(67),FMCRP)
C
      ID     = A(184)
      TOC    = C(12)
      SWEEP  = C(7)
      IF (NPNLS.EQ.1) GO TO 10
C
      ETA    = A(262) /C(1)
      IF (ETA.LE.0.5)  SWEEP = A(332)
   10 CONTINUE
      CLD    = C(13)
      AR     = C(29)
      RAMX   = BCD(1,7)
      BLNS   = BCD(1,9)
      FMCRB  = 0.985
      FRN    = 10.
      IF (RAMX.GT.0.0) FRN = BLNS/SQRT(.7854 *RAMX)
      IF (FRN.LE.6.25) FMCRB = .82 * FRN**.1
C
      IF (ID.LE.23) CALL CPZT(ID,C.6, TOC,CLD,SWEEP,AR,FMCRP)
C
  100 CLTAB(1) = 0.0
      DO 200 I = 1, 11
      IF( I.GT.1 ) CLTAB(I) = CLTAB(I-1) + 0.1
      CALL LNTP(CLTAB(I), TABMCR(I), CLMCR, XMCR, NMCR, 2)
      CALL ADJUST(0,4, CLTAB(I), TABMCR(I))
      IF( TABMCR(I).LT.0.0 ) TABMCR(I) = 0.0
      IF ( TABMCR(1).GT.FMCRP ) FMCRP = TABMCR(1)
C
  200 CONTINUE
C
  300 WRITE(6,1000) (CLTAB(I), TABMCR(I) , I =1,11)
C
 1000 FORMAT(1H1, 5X, * MACH CRITICAL TABLE * /// 5X *  CL   *, 5X
     1               * MACH CRITICAL * //(5X,2F15.4) )
C
      END
```

```
      SUBROUTINE CPZT(ID,XMACH,TOC,CLD,SWP,AR,FMCRP)
C
C     CALCULATES AIRFOIL PRESSURES - BRITISH METHOD
C     AND COMPUTES MACH CRITICAL USING  CREST CRITERIA
C
      COMMON /BLKMCR/ CLMCR(10), XMCR(10), NMCR, SA(32), SB(32),
     1                SC(32), SD(32), SE(32), XU(32), ZT(32), ZS(32)
     1               , ROC
      COMMON /BLKPRT/ KPRINT(50)
C
      DIMENSION COSE(32), SINE(32), YCPI(32), DZDXLT(32)
C
      PI      = 3.1415927
      SWEEP = SWP * 57.29578
      COSPHI = ( COS(SWP) )**(AR/(1.4+AR))
      IF( SWEEP.GT.40.0 ) COSPHI = 0.5 * (COSPHI +.76604**(AR/(1.4+AR)))
C     COSPHI IS THE COSINE OF THE EFFECTIVE ISOBAR SWEEP
      SWP     = ACOS(COSPHI)
C
      NCP     = 31
      NMCR    = 0
C
      DO 100 I= 1,NCP
C
      XI      = I
      XN      = NCP + 1
      NX        = NCP + 1
      COSE(I)= COS(XI*PI/XN)
      SINE(I)= SIN(XI*PI/XN)
      XU(I)   = 0.5*(1.0 + COSE(I))
  100 CONTINUE
C
      CALL SECT(ID, TOC, CLD)
C
C
      ALPHA   = 6.0
  350 A       = ALPHA/57.296
      NMCR    = NMCR + 1
      XM      = XMACH
      IF( XM.GT.0.9 ) XM = 0.9
      IF(XM.LE.0.0 ) XM = 0.01
      XM2       = XM**2
      IF( KPRINT(1).EQ.1 ) WRITE(6,1001) XM, ALPHA, SWEEP
      CLI     = 0.0
      CL      = 0.0
      DCPIM1  = 0.0
      DCPM1   = 0.0
      CMI     = 0.0
      CM      = 0.0
      DO 500 IV = 1, NCP
      SA(IV)      = 0.0
      IF(ID.LT.10.OR.ID.GT.20) SB(IV) = 0.0
      S3NX        = ((-1.0)**IV -1.0)/(XN*(1.0 + COSE(IV)))
      SC(IV)      = S3NX        * SQRT(ROC/2.0)
      SD(IV)      = 0.0
      SE(IV)      = 0.0
      DO 400 IU = 1, NCP
C
C
      IMV     = IU - IV
      S1          = XN/SINE(IU)
      S2          = COSE(IU)/(SINE(IU)**2)
      S3          = XN/SINE(IU)
      S4          = XN/SINE(IU) - 2.0 * ((-1.0)**IU -1.0) /
     1                ( XN * SINE(IU) * (1.0 - COSE(IU)) )
      S5          = - S2
      IF( IU.EQ.IV ) GO TO 150
C
      SIGN        = ((-1.0)**IMV - 1.0)/XN
      DEM         = (COSE(IU) - COSE(IV))
      S1          = SIGN * 2.0 * SINE(IU) /(DEM**2)
      S2          = (-2.0*(-1.0)**IMV*SINE(IU))/(SINE(IV)*DEM)
      S3          = S1         - SIGN * 2.0 * (1.0/(SINE(IU)*DEM))
      S4          = 2.0 * SIGN * (1.0 - COSE(IU)*COSE(IV) ) /
     1               (SINE(IV) * DEM**2 )
     2             - 2.0 * ((-1.0)**IU -1.0) /
     3               (XN * SINE(IV) * (1.0 - COSE(IU)) )
```

```
      S5          = -2.0 * (-1.0)**IPV /DEM
C
 150  CONTINUE
C
      SA(IV)      = SA(IV) + S1         *ZT(IL)
      IF(ID.LT.10.OR.ID.GT.20) SB(IV) = SB(IV) +S2        *ZT(IL)
      SC(IV)      = SC(IV) + S3         *ZT(IU)
      SD(IV)      = SD(IV) + S4         *ZS(IU)
      SE(IV)      = SE(IV) + S5         *ZS(IU)
  400 CONTINUE
C
      CALL CPUCV(1.0, A, SWP, IV, CPIL, CPL, XM)
C
      CALL CPUCV(-1., A, SWP, IV, CPIL, CPL, XM)
C
      DCPI    = CPIL - CPIU
      DCP     = CPL  - CPU
C
      DZDXU   = SB(IV) + SE(IV)
      DZDXL   =-SB(IV) + SE(IV)
C
      YCPI(IV) = CPIU
      DZDXUT(IV) = DZDXL
C
      IF( KPRINT(1).EQ.1 ) WRITE(6,1000) XU(IV),CPIL,CPIL,DCPI,DZDXU,
     1                         DZDXL, CPU, CPL, DCP, XU(IV)
C
      DX      = XU(NCP)
      XB      = 0.5 * (1. + XU(1))
      IF( IV.NE.1 )  DX = XU(IV-1) - XU(IV)
      IF( IV.NE.1 ) XB = 0.5 * (XU(IV-1) + XU(IV))
      CLI     = CLI + DX * (DCPI + DCPIM1) * 0.5
      CL      = CL  + DX * (DCP  + DCPM1 ) * 0.5
      CMI     = CMI + DX * (DCPI + DCPIM1)*0.5 * (XB - 0.25)
      CM      = CM  + DX * (DCP + DCPM1).*0.5 * (XB - 0.25)
C
      DCPIM1 = DCPI
      DCPM1  = DCP
C
  500 CONTINUE
C
      CLI     = CLI + XU(NCP) * DCPI* 0.5
      CL      = CL  + XU(NCP) * DCP * 0.5
      CMI     = CMI + XU(NCP) * DCPI * 0.5 * (0.5*XU(NCP) -0.25)
      CM      = CM  + XU(NCP) * DCP * 0.5 * (0.5*XU(NCP) -0.25)
      CPCRIT = (1.42857/XM2) * (0.52828 *(1. + 0.2 *XM2)**3.5 -1.)
C
      IF( KPRINT(1).GT.0) WRITE(6,1002) CLI,CMI,CPCRIT,CL,CM
  600 CONTINUE
C
      IF( A.LT.DZDXUT(NCP) ) GO TO 610
      NMCP    = NMCP - 1
      ALPHA   = ALPHA -1.
      GO TO 350
C
  610 CALL LNTP(A, XCREST, DZDXUT, XU, NCP, 4)
C
  620 CALL LNTP(XCREST, CPCR, XU, YCPI, NCP, 4)
C
      CPCRST = CPCR/COSPHI**2
  630 CPI2    = CPCRST * CPCRST
      XMDDN   = 1./(1.0323 -.8369*CPCRST -.361*CPI2 -.1336*CPCRST**3
     1                -.0173*CPCRST**4)
      IF( ID.EQ.8 )  XMDDN = 1./(1.012 -.8551*CPCRST
     1             -.4493*CPI2 -.2219*CPCRST**3 -.0447*CPCRST**4)
C
      XMDD    = XMDDN/COSPHI
      IF( XMDD.LE.0.82.OR.ID.EQ.8 ) GO TO 640
      IF( XMDD.LE.1.06 ) XMDD = .2666 -.5729*XMDD +2.7123*XMDD**2
     1                                       -1.4882*XMDD**3
      IF( XMDD.GT.1.06 ) XMDD = 0.8545 +.1132*XMDD
C
 640  CONTINUE
      IF( XMDD.GT.FMCRB ) XMDD = FMCRB
C
      BMDD2   = 1. - (XMDD * COS(SWP))**2
      IF( BMDD2.GT.0.0 ) CLMDD = CLI/SQRT(BMDD2)
```

```
C
      IF( KPRINT(1).GT.0 ) WRITE(6,1003) XMDD,CLMDD,XCREST,CPCR,COSPHI
      CLMCR(NMCR) = CLMDD
      XMCR(NMCR)  = XMDD
C
  700 IF( CLI.LE.0.0 ) RETURN
      IF( NMCR.GE.10 ) RETURN
      ALPHA  = ALPHA - 1.
      GO TO 350
C
 1000 FORMAT(5X,10F12.4)
C
 1001 FORMAT(1H1, ////10X, 57HPRESSURE DISTRIBUTION AND SURFACE SLOPES F
     1OR 2-D AIRFOIL  , 6X,*MACH =*F6.4, 5X,*ALPHA =*F6.3 ///
     1                  10X,*SWEEP ANGLE =*, F7.3 //
     2        12X,*X/C*,7X,*CPI(UPPER)*,2X,*CPI(LOWER)*,4X,*DCPI*,
     3        5X,*DZ/DX(UPPER)*,2X,*DZ/DX(LOWER)*,1X,*CP(UPPER)*,
     4        2X,*CP(LOWER)*,4X,*DCP*, 9X,*X/C* // )
C
 1002 FORMAT(//10X,*CLI =*F7.4, 10X,*CMI =*F7.4, 20X,*CP(M=1) = *,
     1        F8.4,      /11X, *CL =*F7.4, 6X, *CM(C/4) =*
     2        F7.4 / )
C
 1003 FORMAT( /10X,*MACH CRIT =*F7.4, 10X, *CL AT MDD =*F7.4, 10X,
     1              *X-CREST =*F7.4, 10X,*CP-CREST =*F8.4 /
     2              50X, *COSINE OF EFFECTIVE ISOBAR =*F7.4 //)
C
      END
```

```
      SUBROUTINE SECT(ID, TCC, CLD)
C
C     CALCULATES THICKNESS AND CAMBER
C
      COMMON /BLKMCR/ CLMCR(10), XMCR(10), NMCR, SA(32), SB(32),
     1               SC(32), SD(32), SE(32), XU(32), ZT(32), ZS(32)
     1              , FCC
      COMMON /INPUT/ A(2081)
      COMMON /BLKPRT/ KPRINT(50)
      COMMON /BLKDAT1/ D(1411)
C
      DIMENSION XSECT(26),Y164A(26),CAMB(26)
      DIMENSION Y163A(26),Y165A(26),Y163(26),Y164(26),Y165(26),
     1          Y166(26),CAMB1(26)
      DIMENSION A0(11), A1(11), A2(11), A3(11), D1(11), D2(11), D3(11)
      DIMENSION AA(22), BB(22), CC(22), DD(22), XTT(22)
      DIMENSION XT(30), YC(30), YT(30)
      EQUIVALENCE (D(23),AA(1)),(D(45),BB(1)),(D(67),CC(1)),
     1            (D(89),DD(1)), (D(111),XTT(1)),
     2            (A(23),RLE), (A(9),NXSECT),
     3            (A(394),YT(1)), (A(424),YC(1)), (A(454),YT(1)),
     4            (A(24),TCCR), (A(25),CLDR), (A(26),TECH)
C
      DATA XSECT/ 0.0,0.005,0.0075,0.0125,0.025,0.05,0.075,0.1,
     1            0.15,0.2,0.25,0.3,0.35,0.4,0.45,0.5,0.55,0.6,
     2            0.65,0.7,0.75,0.8,0.85,0.9,0.95,1.0 /
      DATA Y163 / 0.0,.00829,.01004,.01275,.01756,.0244,.0295,.03362,
     1            .03994,.04445,.04753,.04936,.05,.04938,.04766,.04496,
     2            .0414,.03715,.03234,.02712,.02166,.01618,.01088,
     3            .00604,.00214,0.0 /
      DATA Y164 / 0.0,.0082,.00985,.0125,.01701,.02343,.02826,.03221,
     1            .03842,.04302,.04639,.04854,.0498,.04988,.04843,
     2            .04626,.04738,.0282,.03345,.02827,.02281,.01722,
     3            .01176,.00671,.00308,0.0 /
      DATA Y165 / 0.0,.00772,.00932,.01165,.01574,.02177,.02647,.0304,
     1            .03666,.04143,.04502,.0476,.04924,.04996,.04963,
     2            .04812,.0452,.04146,.03682,.03156,.02594,.01987,
     3            .01384,.0081,.00306,0.0 /
      DATA Y166 / 0.0,.00755,.00913,.01141,.01516,.02087,.02536,.02917,
     1            .0352,.04001,.04363,.04636,.04832,.04952,.05,.04971,
     2            .04865,.04665,.04302,.03787,.03176,.02494,.01773,
     3            .01054,.00408,0.0 /
      DATA Y163A/ 0.0,.00816,.00983,.0125,.01737,.02412,.02917 .03324,
     1            .0398,.044,.04714,.04913,.04995,.04966,.04837,.04613,
     2            .04311,.03942,.03517,.03044,.02545,.0204,.01535,.0103.
     3            .00525,.00021 /
      DATA Y164A/ 0.0,0.00804,0.00969,0.01225,0.01688,0.02327,
     1            0.02915,0.03199,0.03813,0.04272,0.04606,0.04837,
     2            0.04966,0.04995,0.04954,0.04684,0.04358,0.04021,
     3            0.03597,0.03127,0.02623,0.02103,0.01582,0.01062,
     4            0.00541,0.00021 /
      DATA Y165A/ 0.0,.00765,.00928,.01183,.01623,.02182,.0265,.0304,
     1            .03658,.04127,.04483,.04742,.04912,.04995,.04983,
     2            .04865,.04637,.04204,.03899,.03422,.02912,.02352,
     3            .01771,.0118,.00604,.00021 /
      DATA CAMB / 0.0,0.00281,0.00396,0.00603,0.01055,0.01803,
     1            0.02432,0.02981,0.03953,0.04611,0.05257,0.05742,
     2            0.06120,0.06394,0.06571,0.06651,0.06631,0.06509,
     3            0.06274,0.05913,0.05401,0.04673,0.03607,0.02452,
     4            0.01226,0.0 /
      DATA CAMB1/ 0.0,.0025,.0035,.00535,.0093,.0158,.0212,.02585,
     1            .03365,.0395,.04475,.0486,.0515,.05315,.05475,.05515,
     2            .05475,.05355,.0515,.0486,.04475,.0398,.03365,.02585,
     3            .0158,0.0 /
      DATA A0   / 5 * 0.2969, 3 * 0.14845, 3 *0.514246 /,
     1     A1   / 0.213337, -.096082, -.246867, -.310275, -.27118,
     2            0.412103, 0.192233, 0.083362, -.840115, 2*0.0   /,
     3     A2   / -2.931954, -.54331, 0.175324, 0.3417, 0.1402,
     4            -1.67261, -.556166, -.18315,  1.1101,  2*0.0    /,
     5     A3   / 5.22917, 0.559395, -.266917, -.32182, -.082137,
     6                 1.68869, .268208, -.00691, -1.09401, 2*0.0  /,
     7     D1   / 0.2,  0.234, 0.315, 0.465, 0.7, 0.234, 0.315,
     8            0.465, 0.234, 0.315, 0.465                      /,
     9     D2   / -.140625, -.068571, -.233333, -.684, -1.6625,
     1            -.068571, -.233333, -.684. -.068571, -.233333, -.684/,
     2     D3   / -.070312, -.093878, -.032407, 0.292, 1.3125,
     3            -.093878, -.032407, 0.292, -.093878, -.032407, 0.292/
```

```
C
      NCP      = 31
      ROC      = AA(ID) + BB(ID)*TOC +CC(ID)*TOC*TOC + DD(ID)*TOC**3
C
C     ID   = 1 TO 4, 6  SERIES AIRFOIL             IJ = 1
C     ID   = 5 TO 7, 6A SERIES AIRFOIL             IJ = 1
C     ID   = 8, INPUT AIRFOIL          IJ = 5
C     ID   = 9, BICONVEX                           IJ = 3
C     ID   = 10 TO 20, 4 DIGIT AIRFOIL             IJ = 4
C
      IF( ID.LE.7 ) IJ = 1
      IF (ID.EQ.8) IJ = 5
      IF( ID.EQ.9 ) IJ = 3
      IF( ID.GE.10.AND.ID.LE.20 ) IJ = 4
      IF( ID.GE.21 ) IJ = 5
C
      GO TO (200,300,100,900,700) , IJ
C
  100 WRITE(6,2000)
      DO 110 I = 1,NCP
      ZT(I)      = 2.0*TOC*XU(I)*(1.0 - XU(I))
      SB(I)    = 2.*TOC * (1.-XU(I))
      ZS(I)      = 0.0
  110 CONTINUE
      ROC        = 0.0
      GO TO 960
C
  200 IF( ID.LE.4 ) WRITE(6,2003)
      IF( ID.GT.4 ) WRITE(6,2001)
      DO 220 I = 1,NCP
      X          = XU(I)
      IF( ID.LE.4 ) CALL LNTP(X, ZS1, XSECT, CAMB1, 26, 4)
      IF( ID.GT.4 ) CALL LNTP(X, ZS1, XSECT, CAMB,  26, 4)
      ZS(I)        = ZS1 * CLD
  210 IF( ID.EQ.1 ) CALL LNTP(X, ZT(I), XSECT, Y163,  26, 4)
      IF( ID.EQ.2 ) CALL LNTP(X, ZT(I), XSECT, Y164,  26, 4)
      IF( ID.EQ.3 ) CALL LNTP(X, ZT(I), XSECT, Y165,  26, 4)
      IF( ID.EQ.4 ) CALL LNTP(X, ZT(I), XSECT, Y166,  26, 4)
      IF( ID.EQ.5 ) CALL LNTP(X, ZT(I), XSECT, Y163A, 26, 4)
      IF( ID.EQ.6 ) CALL LNTP(X, ZT(I), XSECT, Y164A, 26, 4)
      IF( ID.EQ.7 ) CALL LNTP(X, ZT(I), XSECT, Y165A, 26, 4)
C
  220 ZT(I)        = ZT(I) * TOC/0.1
      GO TO 960
C
  300   CONTINUE
C
      GO TO 960
C
C
  700 ROC      = RLE
C
      DO 800 I = 1, NCP
      X        = XU(I)
      CALL LNTP(X, ZT1, XT, YT, NXSECT, 4)
      CALL LNTP(X, ZS1, XT, YC, NXSECT, 4)
      ZT(I) = ZT1 * TOC/TOCR
      ZS(I) = ZS1 * CLD/CLDR
  800 CONTINUE
      GO TO 960
C
  900 WRITE(6,2004)
      I4D      = ID - 9
      ROC      = 0.5 * (AO(I4D)*TOC/.2)**2
C
      DO 950 I =1, NCP
      X        = XU(I)
      IF( X.GT.XTT(ID) ) GO TO 910
C
      ZT(I)    = AO(I4D) * SQRT(X) + A1(I4D) * X   + A2(I4D) * X*X
     1                   + A3(I4D) * X**3
      SB(I)    = 0.5 * AO(I4D)/SQRT(X) +A1(I4D) +2.*A2(I4D)*X
     1                 + 3.* A3(I4D) * X**2
      GO TO 940
C
  910 X        = (1. - X)
      ZT(I)    = 0.002    + D1(I4D) * X + D2(I4D) * X*X  + D3(I4D) * X**3
```

```
      SR(I)   = -D1(I4D) -2.* D2(I4D)*X -3.* D3(I4D)*X*X
C
  940 ZT(I)   = TOC/.2 * ZT(I)
      SR(I)   = TOC/.2 * SR(I)
      CALL LNTP(X, ZS1, XSECT, CAMB1, 26, 4)
      ZS(I)   = ZS1 * CLD
C
  950 CONTINUE
C
  960 IF( KPRINT(1).EQ.0 ) RETURN
      WRITE(6,1004) TOC, CLD, ROC
      DO 970 I = 1, NCP
      INV     = NCP + 1 - I
      ZUP     = ZT(INV) + ZS(INV)
      ZLO     =-ZT(INV) + ZS(INV)
      WRITE(6,1005) XU(INV), ZT(INV), ZS(INV), ZUP, ZLO
  970 CONTINUE
C
 1000 FORMAT(/5X,*XMU =*,F10.6,3X,*ZMU =*,F10.6,3X,*XML =*,F10.6,
     1                 3X,*ZML =*F10.6, /
     1                3X,*ZPTE =*F10.6,3X,*ZTHIK =*,F10.6 // )
 1001 FORMAT(5X, 7F10.6 / )
 1002 FORMAT( I5,5X,5F10.0 )
 1003 FORMAT( 10F6.0 )
 1004 FORMAT(10X, *T/C =*F7.4, 10X,*CAMBER =*F7.4, 10X,*L.E.RADIUS =*,
     1             F8.5 //T17,*X/C*,T26,*THICKNESS*,T44,*CAMBER*,T60,
     2             *UPPER*,T75,*LOWER*  )
 1005 FORMAT(5X, 5F15.5 )
 2000 FORMAT(/ 10X, *BICONVEX AIRFOIL SECTION*/ )
 2001 FORMAT(/10X, *6A SERIES AIRFOIL SECTION * /)
 2002 FORMAT(/ 10X, *SUPERCRITICAL AIRFOIL SECTION*  /)
 2003 FORMAT( /10X, *6 SERIES AIRFOIL SECTION* /)
 2004 FORMAT( /10X, *4-DIGIT AIRFOIL SECTION* /)
      RETURN
      END
```

```
      SUBROUTINE CPUDV(S, A, SWP, IV, CPI, CP, XM)
C
C     COMPUTES CP FOR AN INFINITELY SHEARED WING
C
      COMMON /BLKMCR/ CLMCR(10), XMCR(10), NMCR, SA(32), SB(32),
     1                SC(32), SD(32), SE(32), XU(32), ZT(32), ZS(32)
     1              , RCC
C
   10 COSA    = COS(A)
      SINA    = SIN(A)
      COSL    = COS(SWP)
      SINL    = SIN(SWP)
C
   20 F1      = SA(IV)
      F2      = SB(IV)
      F3      = SC(IV)
      F4      = SD(IV)
      F5      = SE(IV)
      FX      = SQRT((1. - XU(IV))/XU(IV))
   30 DUL     = 1.0/(1. +((F2 +S* F5)/COSL)**2 )
C
   40 UTOV2   = DUL * ( COSA * (1. + F1 * COSL +S* F4 * COSL)
     1            +S* SINA * COSL * (1. + F3/COSL) * FX )**2
     2          + DUL * ( COSA * SINL * (F1 +S* F4)
     3            +S* SINA * SINL * (1. + F3/COSL) * FX )**2
     4          + (SINL * COSA)**2 * (1. - DUL)
C
      CPI     = 1.0 - UTOV2
C
      CP      = CPI
C
   50 CPIO    = (1. - SINL * SINL + F1 * F1 + 2.0 * F1 * COSL)
     1            /(1. + (F2/COSL)**2 )    - SINL * SINL
      CPIO    = 1.0  - CPIO
      IF( CPIO.GT.0.0 ) CPIO = 0.0
C
      XMN     = XM * COSL
      IF( XM.LE.0.01.OR.XMN.GE.1.0 ) RETURN
C
      XM2     = XM * XM
      BETA  . = SQRT(1. - XMN * XMN)
C
      SINA    = SINA/BETA
   60 TEST    = (COSL * COSL - CPIO * XMN) * XM2
      IF( TEST.GE.1.0 ) WRITE(6,1000)
      IF( TEST.GE.1.0 ) GO TO 200
C
   70 B       = SQRT(1.0 - TEST)
C
      F1      = F1/B
      F4      = F4/BETA
      F3      = F3/B
   80 DUL     = 1.0/(1. + ((F2 +S* F5)/(B * COSL))**2 )
C
   90 UOV2    = DUL * ( COSA * (1. + F1 * COSL +S* F4 * COSL)
     1            +S* SINA * COSL * (1. + F3/COSL) * FX )**2
     2          + DUL * ( COSA * SINL * (F1 +S* F4)
     3            +S* SINA * SINL * (1. + F3/COSL) * FX )**2
     4          + (SINL * COSA)**2 * (1. - DUL)
C
  100 CP      = -1.42857/XM2
C
      TEST    = 1.0 + 0.2 * XM2 * (1. - UOV2)
      IF( TEST.LE.0.0 ) GO TO 200
      CP      = -CP * ( TEST**3.5 - 1.0 )
C
  200 CONTINUE
      RETURN
 1000 FORMAT(10X,*KUCHEMAN-WEBER CORRECTION FACTOR REACHES LIMIT* )
      END
```

```
      OVERLAY(3,3)
      PROGRAM AERO
C
C     CONTROLS CALLS TO AERO ROUTINES FOR MIN DRAG CALCULATIONS
C
C
C
      COMMON /INPUT/ A(3081)
      COMMON /OUTPUT/ CLTAB(21), TABMCR(21),
     1                CL, CD, CM, ALPHA, CDM, CDL, CDR, CDFO, CLT, CDT,
     2                DH, FK, DELCL, CMG, DCMCL, XACWB, CLA, ALC,
     3                R, CMOH, XH, OMEGA, YCG, EC, FMCRR,
     3                C1(38),CLDE,CLMAX,ABRK,AMAX,DAMAX,DEL,CLS,
     4                ARLO,FMCRO,FML1,FML2,CDC,CLAMCR,CLAML2,CDMCR,
     5                CDML2,CDATM,C2(.7),XACS,CFD,A2,A3,PL,CDMISC,C3(7),
     6                ALT,SPEED,SWEEP,JPASS,ITRIM,RNOFT,
     7                TCD(5),CDSUR(7,5),CDBCD(7,5)
C
      COMMON /CALC/  C(50)
      COMMON /BLKPRT/ KPRINT(50)
C
C
      EQUIVALENCE (NPNLS,A(3)), (SWPMC,C(7)), (TOC,C(12)), (FOC,C(26)),
     1            (IFT,A(10))
C
      RNOFT = ABS(ALT) * 10.0**6
      IF (IFT.EQ.2) RNOFT = RNOFT / 12.
      IF( ALT.LT.0.0) GO TO 80
      CALL ATM2(ALT,RNOFT,IFT)
      RNOFT  = RNOFT * SPEED
   80 CONTINUE
C
C
      IF (JPASS.GE.1) GO TO 300
C
C
C
      RNOFT1 = RNOFT/SPEED
      CALL LNTP(0.0, FMCRO, CLTAB, TABMCR, 11, 2)
C
      CALL FDRG(1.0, RNOFT1)
      CDFF    = TCD(2) + TCD(3)
C
      CALL WDRG(1.00001)
      CDW1    = TCD(4)
C
      CALL WDRG(1.01)
      CDW2    = TCD(4)
C
      PL      = 5.4 *(TOC +2.*FOC)**0.3333 *COS(SWPMC)
C
      XMO    = 1.-FMCRO
      CDRL1 = PL * XMO**2
      IF (XMO.GT.0.12) CDRL1 = 0.0144*PL +0.24*PL*(XMO-.12)
C
      CDRLP = 2.*PL *XMO
      IF (XMO.GT.0.12) CDRLP = 0.24 *PL
C
      CDWP   = (CDW2 -CDW1)*100. -CDRLP
      CDW1   = CDW1 -CDFF -CDRL1
C
      XM      = 1.-FMCRO
      A3       = (XM * CDWP -2.* CDW1)/XM**3
      A2       = (3.* CDW1 - XM * CDWP)/XM**2
C
C
      FML1     = 0.95
      IF( FMCRO.GT.0.90 ) FML1 = FMCRO + .05
      IF( FML1.GT.1.0 ) FML1 = 1.0
      FML2     = FML1 + 0.15
C
      RNOFT2 = RNOFT *FMCRO/SPEED
      CALL FDRG(FMCRO,RNOFT2)
      CALL WDRG(FMCRO)
      CDMCR    = 0.0
      DO 100 I =1,NPNLS
      CDMCR    = CDMCR + CDSUR(I,1)+CDSUR(I,2)+CDSUR(I,3)+CDSUR(I,4)
```

```
 100  CONTINUE
C
C
      RNOFT2 = RNOFT *FML2/SPEED
      CALL FDRG(FML2,RNOFT2)
      CALL WDRG(FML2)
      CDML2  = 0.0
      DO 200 I =1,NPNLS
      CDML2  = CDML2 + CDSUR(I,1)+CDSUR(I,2)+CDSUR(I,3)+CDSUR(I,4)
 200  CONTINUE
C
 300  CONTINUE
C
      CALL DMIN(SPEED, RNOFT, CDO)
C
C
      END
```

```
      SUBROUTINE AT62(HGT,RNOFT,IFT)
C
C     COMPUTES RENOLDS NO. AS FUNCTION OF ALTITUDE
C
C
C     RNOFT EQUALS RN/FT OR METER AT M=1
C
C     CONVERTS HGT TO FEET IF IFT = 1
      ALT    = HGT
      IF (IFT.EQ.1) ALT = ALT * 3.28084
      X      = 1.-6.8753E-6 * ALT
C
      RNOFT = (X +0.38312)/1.38312 *7100574. * X**3.2559
C
      IF (ALT.LE.36089.) GO TO 10
C
      X      = (36089.- ALT) *4.806118E-5
C
      RNOFT = 2.302229E6 * EXP(X)
C
C     CONVERT RNOFT TO RN/METER IF IFT=1
   10 IF (IFT.EQ.1) RNOFT = RNOFT*3.28084
C
C         CONVERT RNOFT TO RN/INCH,  IF IFT  = 2
C
      IF (IFT.EQ.2)  RNOFT = RNOFT/12.
C
      RETURN
      END
```

```
      SUBROUTINE DMIN(SPEED, RNOFT, CDMIN)
C
C     MINIMUM DRAG
C
      COMMON/INPUT/  A(2081)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/   C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ E(906)
C
      DIMENSION BOD(7,20),TOTCD(5),CDBCD(7,5)
C
      EQUIVALENCE (A(11),SREF), (A(1),NBODYS), (A(2),NNAC),
     1            (A(37),BOD(1,1)), (B(149),TOTCD(1)),
     2            (B(185),CDBOD(1,1)), (A(17),FMISC), (B(135),CDMISC)
C
      NBOD   = NBODYS + NNAC
      DO 10 I =1,NBOD
   10 CDBOD(I,5) = 0.0
C
C
      CALL FDRG(SPEED, RNOFT)
C
      CDMISC = (TOTCD(1)+TOTCD(2)+TOTCD(3)) * FMISC * 0.01
      CALL ADJUST(2,0, SPEED, CDMISC)
C
      DO 20 I=1,7
      CDB    = 0.0
      BNO    = 1.
      IF (I.GT.NBOD) GO TO 15
      BASE   = BOD(I,11)
      CALL BDRG(SPEED,BASE,SREF,CDB)
      BNO    = BOD(I,6)
      IF (BNO.EQ.0.0) BNO = 1.
   15 CDBOD(I,5) = CDB * BNO
C                         BASE DRAG
   20 CONTINUE
C
      TOTCD(5) = 0.0
      DO 25 I =1,NBOD
      TOTCD(5) = CDBOD(I,5) + TOTCD(5)
   25 CONTINUE
C
      CALL WDRG(SPEED)
C
      CDMIN  = TOTCD(1)+TOTCD(2)+TOTCD(3)+TOTCD(4)+TOTCD(5)
      CDMIN  = CDMIN + CDMISC
C
C
   30 RETURN
      END
```

```
      SUBROUTINE FORG(SPEED, RNOFT)
C
C     THIS SUBROUTINE CALCULATES FRICTION, FORM, AND INTERFERENCE
C
      COMMON /INPUT/ A(3081)
      COMMON /OUTPUT/ B1(148), TOTCD(5),
     1                CDSUR(7,5), CDBOD(7,5)
      COMMON /CALC/ C(50)
      COMMON /BLKPRT/KPRINT(50)
C
      DIMENSION BOD(7,20), SUR(7,30)
      EQUIVALENCE (A(37),BOD(1,1)), (A(184),SUR(1,1)),(A(16),ROUGHK),
     1            (A(1),NBODYS),(A(2),NNAC),(A(3),NPNLS),(A(4),NHT),
     2            (A(5),NVT),(A(11),SREF),(B1(114),CRITM)
      EQUIVALENCE (C(23),SWET), (C(41),TOCS), (C(42),CBAR2),
     1            (C(43),CLDS), (C(45),SWPMCS), (ISWP,A(6))
C
      DO 10 I=1,3
      TOTCD(I) = 0.0
      DO  5 J=1,7
      CDBOD(J,I) = 0.0
    5 CDSUR(J,I) = 0.0
   10 CONTINUE
C
      J2    = NBODYS
      K2    = NBODYS + NNAC
      L2    = NPNLS
      N2    = NPNLS +NHT +NVT
C
C     BODY CONTRIBUTION TO FRICTION, FORM, AND INTERFERENCE DRAG
C
      CALL IFACT(4,REFUS,CRITM,SPEED,FI)
      IF (K2.LE.0) GO TO 30
      DO 20 I =1,K2
C
      XL    = BOD(I,1)
      CALL CFEC(RNOFT,SPEED,XL,0.,ROUGHK, CDF)
      FR    = BOD(I,14)
      K     = 1
      IF (I.GT.J2) K = 2
      CALL FFACT(K, FR, 0, 0, SPEED, 0.0, FF)
      AWET  = BOD(I,4)
      DRAG  = CDF * AWET/SREF
      FI    = 1.
      IF ( BOD(I,5).NE.0.0 ) FI = BOD(I,5)
      IF (SPEED.GT.1.0) FI =1.
      IF (I.GT.1) GO TO 15
      REFUS = RNOFT * XL
      CALL IFACT(1, REFUS, CRITM, SPEED, FI)
C
   15 CDBOD(I,1) = DRAG
      CDBOD(I,2) = DRAG *(FF -1.)
      CDBOD(I,3) = DRAG *FF *(FI -1.)
   20 CONTINUE
C
C     SURFACE CONTRIBUTION
C
   30 IF (N2.EQ.0) GO TO 50
C               CALCULATE DRAG FOR SURFACES
C
      DO 40 I =1,N2
      XL     = SUR(I,20)
      IF (ISWP.EQ.1.AND.I.EQ.L2) XL = CBAR2
      CALL CFEC(RNOFT,SPEED,XL,0.,ROUGHK, CDF)
C
      TOC    = SUR(I,3)
      IF (ISWP.EQ.1.AND.I.EQ.L2) TOC = TOCS
      TYP    = SUR(I,1)
      CAM    = SUR(I,2)
      IF (ISWP.EQ.1.AND.I.EQ.L2) CAM = CLDS
      CALL FFACT(3, TOC, TYP, CAM, SPEED, CRITM, FF)
C
      MTSWP = SUR(I,24)
      IF (ISWP.EQ.1.AND.I.EQ.L2) MTSWP = SWPMCS
      ID     = 3
      IF (I.LE.L2) ID = 2
      CALL IFACT(ID, MTSWP, CRITM, SPEED, FI)
```

```
C
      AWET  = SUR(I,7)
      IF (ISWF.EQ.1.AND.I.EQ.L2) AWET = SWET
      DRAG  = CDF * AWET/SREF
      CDSUR(I,1) = DRAG
      CDSUR(I,2) = DRAG *(FF -1.)
      CDSUR(I,3) = DRAG *FF *(FI -1.)
C
   40 CONTINUE
C
   50 DO 60 I =1,3
      DO 60 J =1,K2
C
   60 TOTCD(I) = CDPOD(J,I) + TOTCD(I)
C
      DO 70 J =1,N2
   70 TOTCD(I) = CDSUR(J,I) + TOTCD(I)
   80 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CFEC(RNCFT,ZMACH,CBAR,XTR,ROUGHK, CF)
C
C     THIS SUBROUTINE CALCULATES THE SKIN FRICTION COEFFICIENT
C     USING THE WHITE-CHRISTOPH TECHNIQUE.
C
      COMMON /BLKPRT/ KPRINT(50)
C
      FTURB(X) = T*F**2*0.430/(ALOG10(RNL*X*T**1.67*F))**2.56
C
      FLAM(Y)  = 1.328*CFCFIL/SQRT(Y*RNCFT)
C
      FPRIM(X) = 0.43*T*F*F*((ALOG10(RNL   *X*T**1.67*F))**(-2.56)
     1           -1.11178*(ALOG10(RNL   *X*T**1.67*F))**(-3.56))
C
      IF( KPRINT(19).GT.0 ) WRITE(6,1001) RNCFT,ZMACH,CBAR,XTR,ROUGHK
      IF( CBAR.LE.0.0.OR.CBAR.GT.10000.) GO TO 500
C
      ZMACH2 = ZMACH*ZMACH
      T   =1.0/(1.0+ 0.178*ZMACH2)
      F   =1.0 + 0.03916*ZMACH2*T
      DXNP1   = 0.0
      RNCC    = 0.0
      RNL     = RNCFT
      RN1     = RNL
      Z       = RNL *CBAR *T**1.67 *F
      IF (Z.LE.1.0) GO TO 500
      K       = 0
      IF( ROUGHK.LE.0.0 ) GO TO 5
C
      CFR     = T/(1.89 + 1.62*ALOG10(CBAR/ROUGHK))**2.5
    2 CFS     = FTURB(CBAR)
      CFP     = FPRIM(CBAR) - CFS
C
      D       = CFR - CFS
      IF (K.EQ.0.AND.D.LE.0.0) GO TO 5
      IF (ABS(D).LE.0.00001) GO TO 5
      RNL     = RNL *(1.+ D/CFP)
      Z       = RNL *CBAR *T**1.67 *F
      IF (Z.LE.1.0) RNL = RN1 *0.1
      RN1     = RNL
      K       = K + 1
      IF (K.GT.10) GO TO 4
      RNCC    = RNL * CBAR
      GO TO 2
    4 WRITE(6,1000) RNCFT, RNL, CFR
C
    5 CONTINUE
C
      CFCFIL    = (1.0+ 0.1256*ZMACH2)**(-.12)
      IF (XTR.LE.0.0) GO TO 100
      DXN   = 0.1 * XTR
C
   10 DP         = FPRIM(DXN)
      IF (DP.EQ.0.0) GO TO 200
      DXNP1      = DXN - (DXN * FTURB(DXN) - XTR * FLAM(XTR))/DP
      IF ( DXNP1.LE.0.0) DXNP1 = 0.5 * DXN
      DX         = ABS(DXNP1 - DXN)
      DY         = 0.0001*CBAR
      IF (DX.LE.DY) GO TO 20
      DXN        = DXNP1
      GO TO 10
C
   20 XP         = DXNP1 + CBAR - XTR
      CF         = (XP/CBAR)*FTURB(XP)
      GO TO 300
C
  100 CF         = FTURB(CBAR)
      GO TO 300
C
  200 WRITE (6,1000) DXN, RNL
      CF         = 0.0
C
  300 RNL     = RNCFT
      CFBAR = FTURB(CBAR)
      IF ( KPRINT(19).EQ.0) GO TO 400
      WRITE(6,1003) CF, CFBAR, DXNP1, CFCFIL, RNL, RNCC
C
```

```
 400 RETURN
 500 CF       = 0.0
     WRITE(6,1002) CBAR
     RETURN
C
 1000 FORMAT (10X*SUBROUTINE CFEQ WILL NOT CONVERGE*/ 1X,3F15.7 )
 1001 FORMAT(10X,*CFEQ INPUT*, 1PE15.4,0P4F15.7  )
 1002 FORMAT(10X,*CBAR =*,1PE14.5,2X,*OUT OF RANGE IN CFEQ, CF SET EQUAL
    1 TO ZERO* // )
 1003 FORMAT(10X,*CFEQ OUTPUT*, 4F15.7, 1P2E15.4 )
     END
```

```
      SUBROUTINE FFACT(ID,GEOM, TYP,CLD,SPEED,CRITM,FF)
C
C     THIS SUBROUTINE CALCULATES FORM FACTORS FOR EACH COMPONENT.
C
      ITYP    = TYP
      FF = 1.0
      IF (SPEED.GT.1.0) GO TO 40
C
      GO TO (10,20,30) ID
C
C     FUSELAGE OR BODIES (ID=1)
   10 FR = GEOM
      FF = 1.0 + 60./FR**3.0 + 0.0025*FR
      GO TO 40
C
C     NACELLE (ID=2)
   20 FR = GEOM
      FF = 1.0 + 0.35/FR
      GO TO 40
C
C     WING AND SURFACES (ID=3)
   30 TOC = GEOM
      TOC2= TOC*TOC
      IF (ITYP.LE. 7) FF = 1.0 + 1.44*TOC + 2.0*TOC2
      IF (ITYP.EQ. 9) FF = 1.0 + 1.2 *TOC + 100.*TOC2*TOC2
      IF (ITYP.GE. 10.AND. ITYP.LE.20) FF = 1.0 + 1.68*TOC +3.*TOC2
      IF (ITYP.NE. 8) GO TO 40
      DELTAM = SPEED - CRITM
      IF (DELTAM.GE.0.0) ZK = 0.202
      IF (DELTAM.LE.-.2) ZK = 0.12
      IF (DELTAM.GT.-.2.AND.DELTAM.LT.0.0) ZK = 0.202 + .8972*DELTAM
     1      +2.1944*DELTAM**2 -1.2240*DELTAM**3
      FF =ZK*(CLD/.4+1.44*TOC + 2.0*TOC2 +1.
C
   40 RETURN
      END
```

```
      SUBROUTINE IFACT(ID,PARAM,CRITM,SPEED,FI)
C
C     THIS SUBROUTINE CALCULATES INTERFERENCE FACTORS FOR THE
C     FUSELAGE AND LIFTING SURFACES.
C
      DIMENSION XMACH(7),REFUS(9), WBR(63)
C
      DATA     XMACH /     0.25,0.4,0.6,0.7,0.8,0.85,0.9 /,
     1         REFUS / 3.0,10.,20.,30.,40.,60.,80.,200.,100000./,
     1         WBR   / 1.063,1.02,.979,.955,.926,.902,.867,
     1                 1.076,1.036,.996,.977,.947,.92,.884,
     1                 1.066,1.059,1.02,.996,.966,.943,.91,
     1                 1.044,1.049,1.037,1.013,.988,.968,.939,
     1                 0.993,1.018,1.032,1.015,1.003,.993,.973,
     1                 0.940,0.992,1.018,1.014,1.014,1.01,1.007,
     1                 0.932,0.984,5*1.015,
     1                 0.924,0.976,5*1.015,0.924,0.976,5*1.015 /
C
      FI = 1.0
      IF (SPEED.GT.1.0) GO TO 30
C
      GO TO (10,20,20,40), ID
C
C     FUSELAGE CORRELATION FACTOR (FWB)
   10 PAR10 = PARAM/10.0**6
      XM1    = CRITM - .1
      XM     = SPEED
      IF( XM.GT.XM1 ) XM = XM1
C
      FI     = DINT( XM, PAR10, XMACH, REFUS, WBR, 7,9,7,2,2)
      RWB = FI
      GO TO 30
C
C     LIFTING SURFACES
   20 A = COS(PARAM)
      IF (A.GE.0.75) RLS = -.9946 + 5.3578*A -4.5389*A*A +1.2487*A**3
      IF (A.GT.0.65.AND.A.LT.0.75) RLS=.9966 -.752*(.75-A)
      IF (A.LE.0.65) RLS = 0.9214
      B = SPEED - 0.25
      IF (B.LT.0.0) B=0.0
      RLS = RLS - .0015 + .1818*B - .2756*B*B + 1.0677*B**3
      FI = RLS*RWB
      IF (ID.EQ.3) FI = RLS
C
   30 RETURN
C
   40 RWB    = 1.0
      RETURN
      END
```

```
      SUBROUTINE WDRG(FMACH)
C
C     COMPUTES TOTAL AIRPLANE WAVE DRAG
C
      COMMON/INPUT/  A(3081)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/   C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ E(908)
      COMMON /BLKWPD/ WPD(9)
C
      DIMENSION CDSUR(7,5), CDBOD(7,5), TOTCD(5), SUR(7,20),BOD(7,20)
      EQUIVALENCE (B(149),TOTCD(1)),  (B(154),CDSUR(1,1)),
     1            (B(189),CDBOD(1,1)), (A(37),BOD(1,1)),
     2            (A(184),SUR(1,1))
C
      EQUIVALENCE (AR,C(29)), (CLD,C(13)), (DOP,C(2)), (SWPLE,C(25)),
     1            (SWPTE,C(8)), (TS,A(164)), (NBODYS,A(1)),
     2            (NHT, A(4)),(NNAC,A(2)), (NPNLS,A(3)),
     3            (NVT, A(5)), (SEXW,C(10)), (SPLAN,C(4)),
     4            (SREF,A(11)), (TOCW,C(12)),(TAPR,C(30))
C
C
      TOTCD(4) = 0.0
      DO 10 I = 1,7
      CDBOD(I,4) = 0.0
   10 CDSUR(I,4) = 0.0
C
C
      J2    = NBODYS
      K2    = NBODYS + NNAC
      L2    = NPNLS
      M1    = NPNLS +1
      M2    = NPNLS +NHT +NVT
C
C
      IF( FMACH.LE.1.0 ) RETURN
      BETA    = SQRT( FMACH**2 -1.)
C
C     BODY CONTRIBUTIONS
C
      IF( K2.EQ.0 ) GO TO 50
C
      DO 20 I =1,K2
      RAMX  = BOD(I,7)
      BLNS  = BOD(I,9)
      RIN   = 0.0
      REX   = SQRT(BOD(I,8)/3.14159)
      BLFT  = BOD(I,10)
      BNO   = BOD(I,6)
      IF (BNO.EQ.0.0) BNO = 1.0
      IF (I.GT.J2) RIN = SQRT(BOD(I,12)/3.14159)
C
      CALL CDWN(RAMX,BLNS,RIN,BETA,CDW1)
      CALL CDWT(RAMX,BLFT,REX,BETA,CDW2)
C
      CDBOD(I,4) = (CDW1 +CDW2) * RAMX/SREF * BNO
      TOTCD(4) = TOTCD(4) + CDBOD(I,4)
   20 CONTINUE
C
   50 CONTINUE
C
C     WING AND SURFACE CONTRIBUTION
C
  200 WPD(1) = AR * (1.-DOP)**2 * SPLAN/SEXW
      WPD(2) = TAPR/(1.-DOP*(1.-TAPR))
      WPD(3) = SWPLE
      WPD(4) = SWPTE
      WPD(5) = FMACH
      WPD(6) = SEXW/SREF
      WPD(7) = TS
      WPD(8) = CLD
      WPD(9) = TOCW
C
      CALL CDWW(CDW1)
      CDSUR(1,4) = CDW1
      TOTCD(4) = TOTCD(4) + CDW1
```

```
C
 250  IF (N2.EQ.L2) GO TO 300
      DO 290 J = M1, N2
C
      WPD(1) = SUR(J,19)
      WPD(2) = SUR(J,17)
      WPD(3) = SUR(J,4)
      WPD(4) = SUR(J,23)
      WPD(6) = SUR(J,18)/SREF
      WPD(7) = SUR(J,1)
      WPD(8) = SUR(J,2)
      WPD(9) = SUR(J,3)
C
      CALL CDWW(CDW1)
      CDSUR(J,4) = CDW1
      TOTCD(4) = TOTCD(4) + CDW1
  290 CONTINUE
C
 300  CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE CDWN(AMAX, XLNOS, RIN, BETA, CDW)
C
C     NOSE WAVE DRAG (BODY COMPONENT)
C
C
      DIMENSION XLD(7), YBETA(4), FCDW(7,4)
C
      DATA  XLD / 1., 2., 3., 4., 6., 8., 12. /,
     1      YBETA/ C.0, C.32, 0.458, 0.6633 /,
     2      FCDW / .1656,.0617,.0373,.0267,.0196,.0149,.0063,
     3             .2376, .1254, .086, .061, .0306, .0177, .0082,
     4             .2952, .1667, .1067, .066, .0311, .0175,.0081,
     5             .3816, .2010, .1031, .0624,.0297, .0172,.0079 /
C
      CDW   = C.0
      IF( AMAX.LE.0.0) GO TO 50
      DMAX=SQRT(AMAX/C.785398 )
      ELOD=XLNOS/DMAX
      IF(RIN.GT.0.0) GO TO 40
      IF( ELOD.LE.0.0 ) GO TO 50
C
      IF( BETA.GE.C.6633 ) GO TO 20
C
   10 CDW     = DLNT(ELOD,BETA, XLD, YBETA, FCDW, 7,4,7, 2,2)
      GO TO 50
C
   20 X       = BETA/SQRT(1.+ ELOD**2)
      CDW     = (1.2 + 1.15 * X)/(1. + 1.9 * X) /(1. + ELOD**2)
      GO TO 50
C
C
C     FOR A BODY WITH AN OPEN NOSE, SUCH AS A NACELLE, THE CONTRIBUTION
C     OF THE NOSE TO THE TOTAL BODY WAVE DRAG COEFFICIENT IS COMPUTED AS
C     INDICATED BELOW. (THE EXPRESSIONS USED FOR OPEN AND CLOSED BODY
C     BOATTAIL CONTRIBUTIONS DO NOT REQUIRE THIS DISTINCTION--SEE CDWT.)
C
   40 SQRTB   = SQRT(BETA)
      IF( SQRTB.LT.0.8144 ) SQRTB = 0.8144
      CDW     = ((1.-2.*RIN/DMAX)/ELOD)**1.5/SQRTB
C
   50 RETURN
      END
```

```
      SUBROUTINE CDWT(AMAX, XLAFT, REX, BETA, CDW)
C
C     BOATTAIL WAVE DRAG (BODY COMPONENT)
C
C
      DIMENSION DBOD(5), CDWBT(5)
C
      CDW    = 0.0
      IF( AMAX.LE.0.0) RETURN
      IF (XLAFT .LE. 0.0001) RETURN
      DMAX=SQRT(AMAX/0.785398 )
      ELOD=XLAFT/DMAX
      X=BETA/ELOD
      Y=X*X
      Z=ELOD*ELOD
C
C
C     BOATTAIL WAVE DRAG IS A FUNCTION OF BOATTAIL FINENESS RATIO, BASE
C     DIAMETER/MAXIMUM DIAMETER, AND MACH NUMBER.  COMPUTE THE BOATTAIL
C     WAVE DRAG COEFFICIENT AT FOUR  DBOD(I)  VALUES AND INTERPOLATE TO
C
      DBOD(1)=0.0
      CDWBT(1) = (1.165 -.5112*X -.5372*Y +.3964*X*Y)/Z
      IF( X.GT.1.0 ) CDWBT(1) = 0.513/X/Z
C
      DBOD(2)=0.4
      CDWBT(2) = (1.067 -1.709*X +1.6632*Y -.666*X*Y)/Z
      IF( X.GT.1.0 ) CDWBT(2) = 0.3352/X/Z
C
      DBOD(3)=0.6
      CDWBT(3) = (0.7346 -1.4616*X +1.5795*Y -.6542*X*Y)/Z
      IF( X.GT.1.0 ) CDWBT(3) = 0.1980/X/Z
C
      DBOD(4)=0.8
      CDWBT(4) = (0.2555 -.5008*X +.5024*Y -.2077*X*Y)/Z
      IF( X.GT.1.0 ) CDWBT(4) = 0.0494/X/Z
C
      CDWBT(5)=0.0
      DBOD(5)=1.0
      BDOD=2.0*REX/DMAX
      CALL LNTP(BDOD, CDW, DBOD, CDWBT, 5, 2)
C
      RETURN
      END
```

```
      SUBROUTINE CDWW(CDOSF)
C     PROCEDURE FOR WING PRESSURE DRAG
C
      COMMON /BLKWPD/ AR, ZLAM, ZLE, ZTE, ZM, SOSF, TYPE, CAM, TOC
      COMMON /BLKPRT/ KPRINT(50)
      COMMON /BLKDAT1/    (411)
C
      DIMENSION AA(2          2), CC(22), DD(22), XT(22)
C
      EQUIVALENCE (AA(1),D(23)), (BB(1),D(45)), (CC(1),D(67)),
     1            (DD(1),D(89)), (XT(1),D(111))
C
   10 N       = TYPE
C
      HOT     = CAM * 0.055 /TOC
      XTOC=XT(N)
      TOC2= TOC**2
      ROOT=(AA(N)+BB(N)*TOC+CC(N)*TOC2+DD(N)*TOC2*TOC)/TOC
      IF (N.EQ.8) ROOT= 0.88216*(TOC)**1.606/TOC
C
C
C     CALCULATION OF QUANTITIES THAT ARE FUNCTIONS OF GEOMETRY.
C
   17 ZKW  =1.0
      TOC2   = TOC**2
      TOC13  = TOC**0.33333333
      TOC53  = TOC**1.6666667
      ARW    = AR*TOC13
      ARW3   = TOC*AR**3
      DEN1   = 1.0/ARW+ARW3
      DEN2   = 2.0/ARW3+1.0
      ZKT     = 1.+4.*(.5-XTOC*(1.+.5*SQRT(ROOT)))**2
     1          -0.25 * SQRT(ROOT) * (1.-XTOC)**2
      ZKC  = 1.0 +          2.5* HOT **2
      ZKB     = 1.069
      IF( ZKW.EQ.1.2 ) ZKB = 1.0
      TR1     = 0.5/(1.+ZLAM)**2
      TR2     = 1./(1.+ZLAM**2)**2
      ZLAM1   = 1. + TR2 * (TAN(ZLE) + TAN(ZTE))**2
      Z1      = TAN(ZLE)**2
      Z2      = TAN(ZTE)**2
      IF (Z1.GE.Z2) ZKP = (COS(ZLE) + TR1*(Z1-Z2))/ZLAM1
      IF (Z2.GT.Z1) ZKP = (COS(ZTE) + TR1*(Z2-Z1))/ZLAM1
C
      ZK     = (2.0/DEN1 + 3.33/DEN2)*ZKP
      X      = ARW*ZKP
      IF(ARW.GE.1.0) X=SQRT(ZKP/(1.7321*(1.0-ZKP)+1.0/(ZKP*ARW**2)))
C
C     THIS DO LOOP SOLVES FOR *ARE* BY NEWTON*S METHOD.
C
      DO 100 I=1,10
      X3     = X**3
      X4     = X*X3
      F      = (2.0*X)/(X4+1.0) + (3.33*X3)/(X3+2.0) - ZK
      FP     = (2.0-6.0*X4)/((X4+1.0)**2) + (20.0*X**2)/((X3+2.0)**2)
      FFP    = F/FP
      IF(KPRINT(20).EQ.2 .AND. I.EQ.1) WRITE(6,90)
C
      IF(KPRINT(20).EQ.2) WRITE(6,95) I, X, F, FP, FFP
C
      X      = X-FFP
      IF (ABS(FFP).LE.0.0001) GO TO 110
  100 CONTINUE
  110 IF(KPRINT(20).EQ.2) WRITE(6,36)
C
      IF(KPRINT(20).EQ.2) WRITE(6,45) ZLE, ZTE, TOC2, TOC13, TOC53,
     1                ARW, ARW3, DEN1, DEN2, ZKT, ZKC, ZLAM1, ZKP
C
C
      BETA2  = ZM**2-1.0
      BETA   = SQRT(BETA2)
      DIV    = BETA/TOC2
      BETAW  = BETA/TOC13
      BETAW2= BETAW**2
      CD1    = 0.0
      CD2    = 0.0
      CDA    = TOC2*(2.0+3.333/(ZKW**2.8))
```

```
      ZKLIN = CDA/TCC2
      IF (ZM .GT. 1.0)  CD1 = CDA/BETA
      TEST1 = 1.0/COS(ZLE) + 0.01
      IF (ZM .GT. TEST1)CD2 = CDA/SQRT(BETA2-TAN(ZLE)**2)
      ARE=X
      TEST    = ABS(TAN(ZLE))
      UEXP    = (1.+2.* ZLAM**0.33333)
      FB  = 0.3 + 0.7*ZKP**UEXP
      XM      = 0.5* (1.+ZLAM**2 *(2.-ZLAM)**3 )
      Z       = COS(ZLE) + COS(ZTE)
      IF( BETA.GT.TEST ) XM = (TEST/BETA)**Z
  130 ZML   = SQRT(TEST**2+1.0)
      ARE3 = ARE**3
      ZKW1    = ZKW**3.8
      T1      = 1./ARE /(1.+(1.+ZLAM) * FB * BETAW**(1.+ZKW) )
      T2      = ARE3 /(1.+0.33333 *ARE3 *BETAW**4)
      T3      = 2./ARE3 /(1.+(.66667+ZLAM) *BETAW**(1.+ZKW1) *FB)
      T4      = 1./(1.+3.* ARE * BETAW**4)
      ZFEXP   = FB**XM
      FBXM=FB**XM
      D1      = ZKB * ZKW * BETAW * FBXM +(T1+T2)
      D2      = ZKB * ZKW1 * BETAW * FBXM +(T3+T4)
      CDW     = ZKT * ZKW * ZKC * ZKP * TCC53 * (2./D1 +3.33/D2)
      CDWA    = CDW/TCC53
      CDDSR=CDW*SDSR
      IF (ZM.GT.1.0) CDD= CDW*DIV
      AREW = ARE
      ARE  = AREW/TCC13
C
      IF(KPRINT(20).EQ.2) WRITE(6,47)
C
      IF (KPRINT(20).EQ.2)WRITE(6,48)DIV,BETA2,BETAW,BETAW2,CDA,CD1,CD2,
     1         TEST,XM,FB,ZML,T1,T2,T3,T4,D1,D2,CDW,CDD
C
      VLE     = ZLE * 57.2956
      VTE     = ZTE * 57.2956
      IF (KPRINT(20).GT.0) WRITE(6,25) AR,ZKF,ZLAM,ZKT,VLE,ZKC,VTE,
     1                  ZKLIN,TCC,ARW,TYPE,CAM,TCC,TCC13,XTCC,TCC53,RCDT,
     2              FB, HOT, ZML, ZKW, ARE, SDSR, AREW
C
  550 IF (KPRINT(20).GT.0) WRITE(6,600)
C
      IF (KPRINT(20).EQ.1) WRITE(6,610) ZM,BETAW,XM,ZFEXP,CDWA,CDW,
     1                                  CD1,CD2,CDDSR
C
C
   90 FORMAT(4H0  I,5X,1HX,14X,1HF,14X,2HFP,13X,3HFPP/)
   95 FORMAT(1H ,I3,3X,1P4E15.7)
   36 FORMAT(1H0,2X,3HZLE,12X,3HZTE,12X,4HTCC2,11X,5HTCC13,10X,5HTCC53/
     1         1H ,2X,3HARW,12X,4HARW2,11X,4HDEN1,11X,4HDEN2,11X,3HZKT/
     1              3X,3HZKC,12X,5HZLAM1,10X,3HZKF )
   45 FORMAT(1H0,1P5E15.7/(1X,1P5E15.7))
   47 FORMAT(1H0,2X,3HDIV,12X,5HBETA2,10X,5HBETAW,10X,6HBETAW2,9X,3HCDA/
     1         1H ,2X,3HCD1,12X,3HCD2,12X,4HTEST,11X,2HXM,13X,2HFB/
     1         1H ,2X,3HZML,12X,2HT1,13X,2HT2,13X,2HT3,13X,2HT4/
     1         1H ,2X,2HD1,13X,2HD2,13X,3HCDW,12X,6HKLINFC)
   48 FORMAT(1H0,1P5E15.7/(1X,1P5E15.7))
   25 FORMAT(///,16X,36H W I N G   P R E S S U R E   D R A G//
     111X,17H INPUT PARAMETERS,16X,18H OUTPUT PARAMETERS//
     17X,19H ASPECT RATIO    =    ,F7.4,10X,12H KFLARE   =   ,F7.4/
     17X,19H TAPER RATIO     =    ,F7.4,10X,12H KTHICK   =   ,F7.4/
     17X,19H LE SWEEP DEG    =    ,F7.4,10X,12H KCAMB    =   ,F7.4/
     17X,19H TE SWEEP DEG    =    ,F7.4,10X,12H KLINTH   =   ,F7.4/
     17X,19H THICK RATIO     =    ,F7.4,10X,12H ARWIG    =   ,F7.4/
     17X,19H AIRFOIL TYPE,   =    ,F3.0,F4.3,F5.3,5X,12H T/C 1/3 =   ,F7.4/
     17X,19H XT/C            =    ,F7.4,10X,12H T/C 5/3 =   ,F7.4/
     17X,19H RO/T            =    ,F7.4,10X,12H F        =   ,F7.4/
     17X,19H H/T             =    ,F7.4,10X,12H M*       =   ,F7.4/
     17X,19H KW              =    ,F7.4,10X,12H ARE      =   ,F7.4/
     17X,19H SW/SREF         =    ,F7.4,10X,12H AREW     =   ,F7.4/)
  600 FORMAT (//31X,8H RESULTS //,72H MACH    BWIG     EXP     FEXP      CDWIG
     1       CD        CDTH2D    CDTH3D     CD/SREF )
  610 FORMAT(1H , 4(F5.2,2X),5(F7.5,2X))
      RETURN
      END
```

```
      SUBROUTINE BDRG(SPEED, AB, SREF, CDB)
C
C     BASE DRAG
C
      COMMON/INPUT/  A(3081)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/    C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ E(908)
      COMMON /BLKPRT/ KPRINT(50)
C
      CDB    = 0.0
      IF( AB.LE.0.0 ) RETURN
      IF(SPEED.GE.1.0) GO TO 10
      CDB    = (0.1 + 0.1222 * SPEED**8) * AB/SREF
      GO TO 20
C
   10 CONTINUE
      CDB    = (1.42 /(3.15 + SPEED*SPEED) ) * AB /SREF
      IF( SPEED.LE.1.8 )  CDB = 0.2222 * AB/SREF
C
   20 CONTINUE
      IF(KPRINT(15).EQ.0) GO TO 30
      WRITE(6,1000) SPEED, AB, SREF, CDB
 1000 FORMAT(10X*MACH =*F10.7,5X*BASE AREA =*F10.7,5X*SREF =*
     1                    F12.5,5X*BASE DRAG =*F10.7 / )
C
   30 RETURN
      END
```

```
      OVERLAY(3,4)
      PROGRAM AEROR
C
C     CONTROLS CALLS TO AERO ROUTINES FOR VARYING CL CONDITIONS
C
C
C
      COMMON /INPUT/ A(3081)
      COMMON /OUTPUT/ CLTAB(21), TABMCR(21),
     1                CL, CD, CM, ALPHA, CDM, CDL, CDP, CDRO, CLT, CDT,
     2                DF, FK, DELCL, CMO, DCMCL, XACWB, CLA, ALC,
     3                CI(45),CLDF,CLMAX,ABBK,AMAX,CAMAX,CFL,CLS,
     4                AFLD,FMCRD,FML1,FML2,CDC,CLAMCR,CLAML2,CDMCR,
     5                CDML2,CDATM,C2(7),XACS,CDC,PO,A3,PL,CDMISC,C3,
     6                CDAFT,C4(5),ALT,SPEED,SWEEP,JPASS,ITRIM,RNOFT,
     7                TCD(5),CDSUB(7,5),CDRCD(7,5)
C
      COMMON /CALC/  C(50)
      COMMON /BLKPRT/ KPRINT(50)
C
      EQUIVALENCE (NPNLS,A(3)), (CB,C(11)), (BFUS,A(28))
C
      FMACH = SPEED
C
      IF (JPASS.GT.0) GO TO 20
      CDATM = CDMCR
      CALL CLWBT(FMCRD)
      CLAMCR = CLA
C
      CDATM = CDML2
      CALL CLWBT(FML2)
      CLAML2 = CLA
C
C     CLAMCR AND CLAML2 ARE CLA VALUES AT KEY MACH NOS. USED IN CDL
C                                                         CALCULATIONS
 20   IF (JPASS.GT.1) GO TO 30
      CALL CDRO(0.0, FMACH, CDRO)
      CDATM = 0.0
      DO 25 I =1,NPNLS
      CDATM = CDATM + CDSUB(I,1)+CDSUB(I,2)+CDSUB(I,3)+CDSUB(I,4)
 25   CONTINUE
      CALL CLWBT(FMACH)
      RE       = RNOFT * CB
      IF( RE.GT.4.0E6 ) RE = 4.0E6
      RE       = RE/1.0E6
C
      CALL CLBRK(FMACH, RE, RNOFT)
      CALL CDL1(FMACH, RNOFT, FK, DELCL, PRIMEK, AKD, AKB)
      CALL CDFG(FMACH, FK, DELCL, CDC)
      CALL CMCW(FMACH, CMO)
      CALL ADJUST(1,0, FMACH, CMO)
C
      CALL WBAC(FMACH, XACWB)
C
   30 CONTINUE
      CALL CDCP(CL,FMACH,CDRCL)
      CDP    = CDRCL - CDRO
C
      CALL CDL2(FMACH, CL, FK, DELCL, PRIMEK, AKD, AKB, CDL)
      CALL ADJUST(0,5, CL, CDL)
C
      IF( JPASS.LE.1 ) CALL AERA(FMACH, CL, ALPHA)
      CALL AERA1(FMACH, CL, ALPHA)
      IF( JPASS.GT.0 ) GO TO 50
      IF( BFUS.EQ.0.0 ) GO TO 50
C
      DO 40 I = 1, 21
      ANG     = I - 1
      CALL AFTCD(ANG, DCD)
      IF ( I.EQ.1 ) WRITE(6,1000)
   40 WRITE(6,1010)  ANG, DCD
   50 CALL AFTCD(ALPHA, CDAFT)
      CALL TDFG(DCLT,DCDT)
C
      CDM     = CDF + CDC + CDRO
      CD      = CDM + CDL + CDP + CDAFT
C
```

```
      CDT = CD + DCDT
      CLT = CL + DCLT
C
C
      JPASS  = 2
C
 1000 FORMAT( /// 10X, *FUSLAGE AFT-END UPSWEEP DRAG * //
     1                10X, *WING ANGLE*, 10X, *DELCD*  / )
 1001 FORMAT( 1X, 2F15.5 )
      END
```

```
      SUBROUTINE CDDR(CL, XMACH, CDR)
C
C     COMPUTES DRAG RISE
C
      COMMON /INPUT/ A(3081)
      COMMON /CALC/  C(50)
      COMMON /OUTPUT/ CLTAB(21), TABMCR(21), B(71),FMCRC,FML1,FML2,
     1                CDC,CLAMCR,CLAML2,EX(12),A2,A3,PL, FXY(89)
      COMMON /BLKPRT/ KPRINT(50)
C
C
      FMCRR = B(25)
C
      CDR     = 0.0
      CDRD    = 0.0
      CALL LNTP(CL, XMCR, CLTAB, TABMCR, 11, 2)
      IF( XMACH.GT.1.0  ) RETURN
      XM      = XMACH - XMCR
      XMC     = XMACH -FMCRR
      IF (XMC.GT.0.0) CDRD = A2 *XMC**2  +A3 *XMC**3
      IF( XM.GT.0.0 )  CDR  = PL * XM**2
      IF( XM.GT.0.12 ) CDR = 0.0144*PL + 0.24*PL *(XM -.12)
      CDR     = CDR + CDRD
C
      RETURN
 1000 FORMAT(5X, *CDDR DUMP* /(1X, 7F15.5))
      END
```

```
      SUBROUTINE CLWBT(SPEED)
C
C     COMPUTES WING-BODY-TAIL LIFT CONTRIBUTION
C
      COMMON /BLKCLA/ CLAI(11)
      COMMON /INPUT/ A(2061)
      COMMON /OUTPUT/ D(58),CLA,ALO,PX(35),CLAW,CLAB,CLAT,AA,BB,HSTAR,
     1              CC,ASPEAK,CLDH, BYX(119)
      COMMON /CALC/ BD2, DOB, CP, SPLAN, XCPTE,
     1              SWPCC, SWPMC, SWPTE, SWMT, SEXW, CBAR,
     2              TOC ,CLD,ARXF,CPX,CTX,XPX,XTX,
     3              YPX,YTX,XP,YP,XH,OMEGA,SWPLE,FCC,
     4              SWPR,KPASS,AR,     TR,YIX,SIX,ARI,CPXF,
     5              SGXF,AROP,FXCC,TWIST,WINC,XHT,TOCS,CBAR2,CLDS,
     6              C(7)
C
      DIMENSION BOD(7,20)
      EQUIVALENCE (A(37),BOD(1,1)), (NHT,A(4)), (NBODYS,A(1)),
     1            (TECH,A(26)), (SREF,A(11)), (ITRIM,A(22))
C
C
C
      CLAW   = 0.0
C
      CLAI(1) = SPLAN
      CLAI(2) = TOC
      CLAI(3) = TR
      CLAI(4) = AR
      CLAI(5) = 0.0
      IF (TECH.GT.0.) CLAI(5) = 0.0334
      CLAI(6) = CLD
      CLAI(7) = 0.0
      IF (TECH.GT.0.) CLAI(7) = 0.09
      CLAI(8) = DOB
      CLAI(9) = 0.0
      IF (TECH.GT.0.) CLAI(9) = 1.173763 * TOC
      CLAI(10) = SWPMC
      CLAI(11)= SREF
C
      CALL AER2(SPEED,,CLA)
      CLAW   = CLA
C
      CLAT   = 0.0
      AA  = 0.0
      BB  = 0.0
      CC  = 0.0
      ASPEAK = 90.0
      HSTAR = 0.0
      CLDH  = 0.0
C
      IF( NHT.EQ.0 ) GO TO 30
C
   20 CALL TAIL(SPEED)
   30 IF (NHT.EQ.0.AND.ITRIM.EQ.0) CALL ELEVON(SPEED)
C
      CLAB= 0.0
      IF (NBODYS.EQ.0) GO TO 50
C
      WB  = BOD(1,2)
      HB  = BOD(1,3)
      AB  = BOD(1,7)
C     AB  = MAX.CROSS-SECTIONAL AREA OF FUSELAGE
      BL    = A(254)
      BLN = BOD(1,9)
      PERIM = 3.14159*SQRT((WB**2 + HB**2)/2.)
      BHPF = WB/HB*PERIM/SQRT(AB)
      AK1    = 0.009942*BHPF -.000379*BHPF**2 +.00001*BHPF**3
      CLAB   = AK1 * (BL/BLN)**0.3333 * AB/SREF
C
   50 CLA = CLAW + CLAT + CLAB
C
      CALL AALO(SPEED)
      CALL ADJUST(3,0, SPEED, ALO)
C
      RETURN
      END
```

```
      SUBROUTINE TAIL(SPEED)
C
C     COMPUTES TAIL FLOW FIELD AND LIFT
C
      COMMON/INPUT/  A(3001)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/    C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ E(908)
      COMMON /BLKCLA/ CLAI(11)
      COMMON /BLKPRT/ KPRINT(50)
C
      DIMENSION SUR(7,30), CDSUR(7,5), YSWPL(11),YBMIN(11)
      DIMENSION XZCC(7),YSC(4),ZEL(7,4), XM(11),YFM(11)
C
      EQUIVALENCE (SUR(1,1),A(184)), (CDSUR(1,1),B(154)), (AR,C(29)),
     1            (CLAW,B(96)),      (NPNLS,A(3)),       (WINC,C(39)),
     2            (TR, C(30)),       (SWPOC,C(6)),       (BC2, C(1)),
     3            (XB,  C(21)),      (AH,B(99)),         (BH,B(100)),
     4            (ADH,B(101)),      (CB,A(12)),         (SREF,A(11)),
     5            (XHT,C(40)),       (XCBTE,C(5)),       (YSWPL(1),D(1)),
     6            (YBMIN(1),D(12)),(CLAT,B(98)),       (CLCH,B(104)),
     7            (DEDA,B(95)), (CDATM,B(127)),  (SPLAN,C(4))
C
      DATA  XZCC/ 0.0,0.1,0.2,0.3,0.4,0.6,0.8 /,
     1       YSC / 0.0, 0.2, 0.6, 1.0 /,
     2       ZEL / 1.2,.935,.720,.570,.465,.314,.200,
     3             1.1,.875,.684,.541,.440,.300,.195,
     4             .95,.762,.605,.488,.400,.281,.190,
     5             .81,.646,.522,.436,.365,.270,.187 /,
     6       XM  / 0.0,0.8,1.0,1.1,1.2,1.3,1.4,1.6,1.8,2.0,2.2 /,
     7      YFM  / 1.0,1.0,1.0,.99,.94,.77,.66,.53,.46,.415,.4 /
C
C
C  ** DOWNWASH IS CALCULATED ***
C
      K       = NPNLS + 1
C
      ADHT   = SUR(K,19)
      SEXHT  = SUR(K,18)
      SWLHT  = SUR(K,4)
      HTLAM  = SUR(K,17)
      CAMHT  = SUR(K,2)
      HTCC   = SUR(K,3)
      HTLE   = SUR(K,11)
      HTY    = SUR(K,12)
      HTZ    = SUR(K,13)
      SWHCHT= SUR(K,22)
      TINC   = SUR(K,14)
C
C
   10 ZKA     = 1./AR -1./(1. + AR**1.7)
      IF( AR.LT.2.3 ) ZKA = 0.37 - 0.0567 * AR
      ZKTR    = (10. - 3.0 * TR)/7.0
C
      DEDA    = 0.0
      XMAC   = XB + 0.25 * CB
      XLHT   = XHT - XMAC
      IF( XLHT.LE.0.0 ) GO TO 30
C
      ZKH    = (1.-0.5*HTZ/BC2)/(XLHT/BC2)**0.33333
C
      COSQ    = SQRT(COS(SWPOC))
   20 DEDAO   = 4.44 * (ZKA * ZKTR * ZKH * COSQ )**1.19
C
      IF( KPRINT(21).EQ.0 ) GO TO 21
      WRITE(6,1000) CLAW, DEDAO, CDATM, HTZ, XHT, XB,
     1                 HTLE,CB,ZKA,ZKTR,ZKH,COSQ
   21 CONTINUE
      CALL AER2(0.1, CLAMO)
      DEDA    = DEDAO *CLAW/CLAMO
C
C  ** DYNAMIC PRESSURE AT THE TAIL  ***
C
 30   CDG    = 0.0
      DQCQ   = 0.0
      IF (NPNLS.EQ.0) GO TO 50
```

```
C
      CDO    = CDATM
C
      XOC     = (XHT - XCRTE)/CB
      IF( XOC.LE.0.0 ) GO TO 50
C
      ZWCC    = 0.68 * SQRT(CDO * (XOC + 0.15))
      DCDCO   = 2.42 * SQRT(CDO)/(XOC + 0.3)
C
      GAMMA   = ATAN(HTZ/(XOC*CB))
      AW      = 2.0
      CL      = CLAW* AW
      EW      = 0.51566 * CL/AR
      ZOC     = XOC * TAN(GAMMA + EW - AW/57.3)
      ZOZW    = ZOC/ZWCC
C
      IF( ZOZW.GE.1.0 ) GO TO 50
C
      DOCO    = DCDCO * (COS(1.570796* ZOZW))**2
C
   50 QOQ     = 1.0 - DOCO
C
C  ** CARRY-OVER FACTORS ZKWB AND ZKBW ARE COMPUTED ***
C
   60 DOBHT   = 2.0 * HTY/(2.0 * HTY + SQRT(ARHT * SEXHT) )
C
      ZKWB    = 1.0028 +.7116*DOBHT +.42*DOBHT**2 -.1366*DOBHT**3
      ZKBW    = .0004 +1.2662*DOBHT +.6013*DOBHT**2 +.1263*DOBHT**3
C
C  ** LIFT-CURVE-SLOPE OF THE EXPOSED SURFACE IS COMPUTED ***
C
      CLAI(1)  = SEXHT
      CLAI(2)  = HTOC
      CLAI(3)  = HTLAM
      CLAI(4)  = ARHT
      CLAI(5)  = 0.0
      CLAI(6)  = CAMHT
C
      CLAI(7)  = 0.0
      CLAI(8)  = 0.0
      CLAI(9)  = 0.0
      CLAI(10) = SWMCHT
C
   70 CALL AER2(SPEED, CLA)
      CLAT    = CLA * (ZKWB + ZKBW) * (1. - DEDA) * QOQ
      CLDH    = CLA * ZKWB * QOQ
C
C  ** INDUCED DRAG FACTOR FOR TAIL IS COMPUTED ***
C
      CALL LNTP(SWLHT, RTSUB, XSWPL, YRMIN, 11, 2)
      PTAIL   = RTSUB
      IF( SPEED.LE.0.9 ) GO TO 80
      ZNDM    = 12.0 * (COS(SWLHT)**1.6) * (SPEED - 0.9)
      FNDM    = 1.0/(1. + ZNDM + ZNDM*ZNDM)
      RTAIL   = RTSUB * FNDM
C
   80 HTK1    = (1.-RTAIL) * 0.01745 /(CLA * ZKWB)
      HTK2    = RTAIL * 0.31831/ARHT * SREF/SEXHT
      HTK     = HTK1 + HTK2
C
      AH     = HTK * CLDH * CLDH
      BH     = 2. * HTK * CLAT * CLDH
      AOH   = (1.-DEDA) * (WINC - TINC)
C
      IF (XLHT.GE.0.0) GO TO 90
C
C     CANARD INTERFERENCE FACTOR ON WING (EL) IS COMPUTED
C
      ZOC   = ABS(HTZ/CB)
      SCOSW = SEXHT/SPLAN
C
      ELS    = PLNT(ZOC,SCOSW, XZCC,YSC,ZEL, 7,4,7, 4,2)
      CALL LNTP(SPEED, FM, XM,YFM, 11, 4)
C
      EL     = ELS * FM
      IF (EL.GT.1.0) EL = 1.0
```

```
C
      CLAW  = CLAW - EL * CLAT
C
   90 CONTINUE
C
      IF( KPRINT(21).EQ.1 ) WRITE(6,1000) CLAT, SPEED, CLA, ZKWB, COBHT,
     1              ZKBW, DECA, QOQ, CLDH, AH, BH, ACH, CECAD,
     2                   RTSUB,RTAIL,HTK1,HTK2,HTK
C
      RETURN
 1000 FORMAT(10X,*TAIL LIFT DUMP* /(1X, 7F15.5) )
      END
```

```
      SUBROUTINE ELEVON(SPEED)
C
C     COMPUTES AERODYNAMIC EFFECT OF ELEVONS FOR TAILLESS AIRCRAFT
C
      COMMON /INPUT/   A(3081)
      COMMON /CALC/    C(50)
      COMMON /OUTPUT/  B(223)
      COMMON /ELKPNT/  KPPINT(50)
      COMMON /ELKDATE/ SPAN(6),    YTPR(4),   FKB(6,4),   FKD(6,4),
     1                 SPAN2(11),YTPR2(5),  FKSW(11,5),
     2                 SPAN3(6),  YTPR3(4),  FKM(6,4),
     3                 SPAN4(5),  YBFI1(5),  FKA(5,5),
     4                 SPAN5(5),  YBFI2(3),  FKF(5,3)
C
      DIMENSION XCF(5), YTOC(7), ZDCML(5,7)
C
      EQUIVALENCE (CF, A(170)), (BFI,A(171)), (BFO,A(172)),
     1            (AR,C(29)),   (SWPMC,C(7)),   (SWPLE,C(25)),
     2            (TPR,C(30)),  (SWPCC,C(6)),   (FMCRO,B(114)),
     3            (TOC,C(12))
      EQUIVALENCE (CLDE,B(68)), (AKD,B(69)), (XCP,B(70))
C
      DATA  XCF / 0.05, 0.15, 0.25, 0.35, 0.4 /,
     1     YTOC / 0.03, 0.06, 0.09, 0.12, 0.15, 0.18, 0.21 /,
     2    ZDCML / -.51,-.39,-.298,-.241,-.220, -.470,-.368,-.288,
     3            -.238,-.22, -.43,-.347,-.279,-.236,-.22,
     4            -.39,-.326,-.271,-.233,-.22, -.353,-.305,-.262,
     5            -.231,-.22, -.32,-.284,-.253,-.229,-.22,
     6            -.301,-.272,-.247,-.227,-.22 /
C
      XMC2  = SPEED * COS(SWPMC)
      IF (SPEED.GT.FMCRO) XMC2 = FMCRO * COS(SWPMC)
      BETA  = SQRT(1.-XMC2**2)
C
      DF    = ACOS(1.-2.*CF)
      CID   = 2.*(DF +SIN(DF)) /57.3
C
      CLOC  = AR/(2.+ SQRT(4.+(1.-XMC2**2)*(AR/COS(SWPMC))**2 ))
C
      ZKBI   = DLNT(BFI, TPR, SPAN, YTPR, FKB,  6,4, 6,2,2)
      ZKBO   = DLNT(BFO, TPR, SPAN, YTPR, FKB,  6,4, 6,2,2)
      ZKB    = ZKBO - ZKBI
C
      ASC    = SQRT(1. - (1.-CF)**1.61 )
      ZKC    = 1. + 1.375 * (1.-ASC)**2 /AR
C
C     CLDE IS THE FLAP LIFT EFFECTIVENESS PER DEG. ELEVON
      CLDE   = 0.77 * CID * CLOC * ZKC * ZKB
      SWPHL  = ATAN(TAN(SWPLE) - 4./AR * (1.-TPR)/(1.+TPR) * (1.-CF) )
C
      ZKDI   = DLNT(BFI, TPR, SPAN, YTPR, FKD,  6, 4,  6, 2,2)
      ZKDO   = DLNT(BFO, TPR, SPAN, YTPR, FKD,  6, 4,  6, 2,2)
      ZKD    = ZKDO - ZKDI
C     PROFILE DRAG TERM AKDP
C
      AKDP  = 1./BETA *0.00168 * CF**2.322 *COS(SWPHL) *ZKD
C
      E      = AR/6.283
      D      = BFO
C
      ZKA = DLNT(E, BFI,SPAN4, YBFI1,FKA, 5, 5, 5, 4, 2 )
      ZKF = DLNT(D, BFI,SPAN5, YBFI2,FKF, 5, 3, 5, 2, 2 )
C
C     INDUCED DRAG TERM AKDI
      AKDI  = ZKA *ZKF *0.31831 *CLDE**2/AR
C
      AKD   = AKDP + AKDI
C
      DCLP  = 0.7208/(CLOC * ZKB)
C
      ZKSWI= DLNT(BFI, TPR, SPAN2, YTPR2, FKSW, 11, 5, 11, 2, 2)
      ZKSWO= DLNT(BFO, TPR, SPAN2, YTPR2, FKSW, 11, 5, 11, 2, 2)
      ZKSW = ZKSWO - ZKSWI
C
      ZKMO = DLNT (BFO, TPR, SPAN3, YTPR3, FKM, 6, 4, 6, 2, 2)
      ZKMI = DLNT (BFI, TPR, SPAN3, YTPR3, FKM, 6, 4, 6, 2, 2)
      ZKM = ZKMO - ZKMI
```

```
C
      DCML   = DLNT(CF,TOC, XCF,YTOC,ZDCML, 5,7,5, 2,2)
C
      DCMCLP = DCML * ZKM  *ZKSW *AR/1.5 *TAN(SWPCC)
C
      XCP    = 0.25 - DCMCLP *DCLP
C
      IF (KPRINT(21).GT.0) WRITE(6,1000) CF,SWPLE,SWPCC,SWPMC,AR,
     1             ZKP,ZKC,CLDE,ZKE,AKDP,ZKA,ZKF,AKDI,
     2             ZKSW,ZKM,DCML,XCP
 1000 FORMAT(10X,*FLEVON DUMP*/(1X,7F15.5))
C
      RETURN
      END
```

```
      SUBROUTINE AALO(SPEED)
C
C     CALCULATES ZERO LIFT ANGLE OF ATTACK
C
      COMMON/INPUT/  A(3081)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/    C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ F(908)
      COMMON /BLKPRT/ KPRINT(50)
C
      EQUIVALENCE (TWIST,C(28)), (WINC,C(39)), (BINC,A(275)),
     1            (DCB,C(2)),    (TOC,C(12)),  (SWPCC,C(6)),
     2            (CLD,C(13)),   (TAP,C(30)),  (CLA,B(59)),
     3            (ALO,F(60)),   (C1,B(96)),   (C2,B(97)),
     4            (C3,B(98)),    (NPNLS,A(3)), (NHT,A(4))
C
C
C
      ALOC    = 0.0
      ALOT    = 0.0
      ALOI    = 0.0
      CLAB   = 0.0
      SWPCCB = 90.0
      TAU     = 0.0
      ZKBW    = 0.0
      ZKWB    = 0.0
      TINC    = A(275 +NPNLS )
C
      XMCR     = 0.75
      XMNO    = SPEED * COS(SWPOC)
      TOCC    = TOC/COS(SWPOC)
C
      IF( TOCC.LT.0.1 ) XMCR = 0.75 + 1.25 * (0.1 - TOCC)
      DALODC = 5.6
      IF(  XMNO.GT.XMCR ) DALODC = 5.6 - 249.0 * (XMNO - XMCR)**2
      IF( DALODC.LT.0.0 ) DALODC = 0.0
C
      ALOC     = - DALODC * CLD
C
C
  200 IF( TWIST.EQ.0.0 ) GO TO 300
      IF( SPEED.GT.1.0 ) GO TO 300
      TAU = TWIST
      BETA = SQRT(1.0 - SPEED**2)
      SWPCCB = 90.0
      IF (BETA.GT.0.0) SWPCCB = ATAN(TAN(SWPCC)/BETA) * 57.29578
      DALC    = 0.093 - 0.000571*SWPCCB + 0.5761*TAP -0.2645*TAP**2
      ALOT    = - DALC * TAU
C
  300 IF( WINC.EQ.0.0.AND.TINC.EQ.0.0 ) GO TO 400
      ZKBW    = .0004 +1.2662*DCB +.6018*DCB**2 +.1263*DCB**3
      ZKWB    = 1.0026 +.7116*DCB +.42 *DCB**2 -.1366*DCB**3
      CLAB    = C3 + ZKBW/(ZKBW + ZKWB) * C1
      ALOI    = (CLAB * WINC +C2 *(WINC -TINC))/CLA
     1                +(BINC -WINC)
C
  400 ALO      =          + ALOC + ALOT + ALOI
      IF( KPRINT(13).GT.0 ) WRITE(6,1000) ALO, ALOC, ALOT,
     1            ALOI, SPEED,          TOC, CLD, TAU, SWPCCB, DCB,
     2            TAP, C1, C2, C3, CLAB, ZKBW, ZKWB, WINC,
     3            TINC, BINC
C
 1000 FORMAT(5X,*AALO DUMP*,5X,*ALO =*,F6.3,5X,*ALOC =*,F6.3,
     1        5X,*ALOT =*,F6.3,5X,*ALOI =*,F6.3 /(5X,7F15.5) )
      RETURN
      END
```

```
      SUBROUTINE CLBPK(SPEED, RE, RNDFT)
C
C     CALCULATES LIFT BREAK CL AND CLMAX
C
      COMMON /INPUT/ A(2081)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/  C(50)
      COMMON /BLKPRT/ KPPINT(50)
      COMMON /BLKDAT1/XSWPL(11), YFMIN(11),
     $                AAA(22), BBB(22), CCC(22), DDD(22), XTT(22),
     $                BCT(12),ATSW(6),TPRT(6),FXAC1(216),FXAC2(216),
     $                XPD4(5), YPD4(5), ZPD4(3), FPD4(5,5,3),
     $                XDYO(6), YXMN(6), Z1PD5(6,6),
     $                XPD6(4), YPD6(4), ZPD6(6), FPD61(4,4,6),
     $                FPD62(4,4,6), FPD63(4,4,6),
     $                YZ(3),AMAP(22), BMAF(11),
     $                XIN13(10), YIN13(7), ZOUT13(10,7), ZOUT14(10,7),
     $                NXIN15, XIN15(6),  NYIN15, YIN15(5),
     $                FOUT15(6,5), FOUT16(6,5),
     $                Y1(6), Y1(6), X9(8), Y9(8), XAP(5), YDCL(5),
     $                XSWP(7), YTR(4), FEP35(7,4), FEP7(7,4),
     $                XCLDP(7), YAMP(7)
C
      COMMON /BLKDAT2/XTR(6), YC1(6), YC2(6), XSWP1(4), YA(4),
     $                YP(4), XDY(6), XM(4), CTAP(6,4), DTAP(6,4),
     $                XYCLM(13), YYDY(6), FCLMY(13,6), XXC2(9),
     $                YYMACH(5), FDCLMX(9,5),XDY1(9),  YXMT(4),
     $                 ZC1MAX(9,4), XDY2(8), YFDD(6), ZDC1M(8,6),
     $                ZZDC1M(8,6),
     $                XSP(8), YDYA(6), FDA(8,6), YAR(6), YCD(8),
     $                FKVDEM(6,8), XANG(10), YETCC(7), FEA(10,7),
     $                XC2(6),YAST(9),FDAM(6,9),XCT(6),YM(5),FDAM2(6,5),
     $                XXYCLM(9), FCLMXX(9,6),
     $                XX(12), XY(7), XF(12,7)
C
      DIMENSION XF1(7), YF1(7), YF2(7)
C
      EQUIVALENCE (ALC,B(60)),  (CLA,B(59)),  (TCC,C(12)),  (CLD,C(13)),
     1          (CLAT,B(98)),  (CLAW,B(96)),  (CLS,B(112)),  (CP,C(11)),
     2          (SWPCC,C(6)),  (CLFP,B(105)), (CLDP,B(106)), (DEL,B(111)),
     3          (CLMAX,B(107)),(ABRK,B(108)), (AMAX,B(109)), (TS,A(184)),
     4          (DAMAX,B(110)),(ABLD,B(113)), (SWPLE,C(25)), (CONCL,A(27))
     5          (AR,C(29)),     (TR,C(30)),    (FCC,C(26))
      DATA   XF1 / 5.0, 6.0, 6.47712, 6.77815, 6.95424, 7.39794, 8. /,
     1       YF1 / -.125, -.11, -.07, -.01, 0., 0.02, 0.025 /,
     2       YF2 / -.1063, -.0713, -.055, -.0175, 0., 0.0375, 0.0463/
C
      NT     = TS
      CALL LNTP(TR, C1, XTR, YC1, 6,4)
      C1COS  = ((1+1.) * COS(SWPLE)
      ARUP   = 4./C1COS
      ARLO   = 3./C1COS
C
      IF (SPEED.GE.1.0) GO TO 20
      DYO    = AMAP(NT) * TCC
      IF( DYO.LT.0.8 ) DYO = 0.8
      IF( DYO.GT.2.4 ) DYO = 2.4
C
      XMN     = SPEED * COS(SWPCC)
      IF( XMN.LT.0.2 ) XMN = 0.2
      XMNO    = XMN
      CLIN    = CLD/COS(SWPCC)**2
      IF( XMN.GT.0.7 ) XMNO = 0.7
C
      ABRKO  = DLNT(DYO, XMNO,XDYO, YXMN, Z1PD5, 6,6, 6, 4,2)
      ABRK   = ( ABRKO + (12.05-4.1*XMN)*CLIN ) * COS(SWPCC)
      ABRKLC = 2./COS(SWPLE)
      IF( ABRK.LT.ABRKLC ) ABRK = ABRKLC
      CLPBO  = CLA * ABRK
C
      DCLPB  = 0.0
      FPB     = 1.54
      IF( XMN.GT.0.5 ) FPB = 1.54 - 2.9 * (XMN - 0.5)
      IF( CONCL.GT.0.0 ) DCLPB = FPB * ((CONCL + .0643)*COS(SWPCC))**2
C
      CLPB   = CLPBO + DCLPB
C
```

```
      DCLDRE = 0.1226 -.00714*SPEED -.12857*SPEED**2
C
      IF( NI.LE.9 ) DY = AMAP(NI)*TOC +BMAP(NI)*CLD
      IF( NI.GT.9 ) DY = AMAP(NI)*TOC +1.75*CLD
C
      TRANS  = 0.0
      IF( DY.GT.1.65 ) TRANS = (DY - 1.65)/.4
      IF( DY.GE.2.05 ) TRANS = 1.0
      IF( SWPLE.GE.0.67 ) TRANS = 0.0
      CLDB   = CLPB + TRANS *(-.0376 -.24414*SPEED -.0685*SPEED**2
     1                +.4149*SPEED**3 + RE * DCLDRE )
C
      GO TO 30
C
 20   CLDB   = 1.0
      BETAC = 10.0
      IF (SWPLE.NE.0.0) BETAC = SQRT(SPEED**2 -1.)/ABS(TAN(SWPLE))
      IF (BETAC.GT.1.0) GO TO 25
      CLSB1  = 0.85
      CLSB9  = 0.85
      IF( SWPLE.GT.0.7243 ) CALL LNTP(SWPLE, CLSB1, X1, Y1, 6, 4)
      IF( SWPLE.GT.0.3665 ) CALL LNTP(SWPLE, CLSB9, X9, Y9, 8, 4)
      CALL LNTP(AR, DCLAR, XAR, YDCL, 5, 4)
      CLSB9  = CLSB9 + DCLAR
      CLSB   = CLSB1 + (CLSB9 - CLSB1) *(BETAC - .1) * 1.25
      CLSB   = CLSB + 0.5 * CLD
      CLFB   = CLSB
      CLDB   = CLSB
 25   CONTINUE
C
C     SUPERSONIC MAXIMUM LIFT COEFFICIENT      ****
      CNA04B = CLAW * 14.325 * SQRT(SPEED*SPEED -1.)
C
      CM     = 1./SPEED
      CLMAX  = 0.7722 + 0.3384 * CM :1.1646 * CM*CM -0.8215 * CM**3
C
      CM1    = 1. - CNA04B
      IF(CM1.LT.0.0 ) CM1 = 0.0
C
      CLMAX  = CLMAX - 0.048 * CM1
C
      AMAX   = 68.5429 -177.2327 * CM +461.9204 *CM*CM
     1              -629.4522 * CM**3 +321.4001 * CM**4
C
      AMAX   = AMAX +15.8074 -3.0001*CNA04B -12.8073*CNA04B**2
C
      IF( CLMAX.GT.1.25 ) CLMAX = 1.25
      IF( AMAX.GT.54.5  ) AMAX  = 54.5
C
      GO TO 300
   30 CONTINUE
C
      CLMAX  = 0.0
      DY     = AMAP(NI) * TOC
      CALL LNTP(TR, C2, XTR, YC2, 6, 4)
C
C
C
      XMT    = XTT(NI)
      IF( XMT.LT.0.3 ) XMT = 0.3
      IF( XMT.GT.0.45) XMT = 0.45
      IF( AR.LE.ARLC ) GO TO 200
C
      C1MAXB = DLNT(DY, XMT, XDY1, YXMT, ZC1MAX, 9, 4, 9, 2,2)
C
      FOC    = 100.*FOC
      DC1MAX = DLNT(DY, FOC, XDY2, YFOC, ZDC1M, 8, 6, 8, 2,2)
      IF( XMT.GE.0.35 ) DC1MAX = DLNT(DY,FOC,XDY2,YFOC,Z2DC1M,8,6,8,2,2)
C
      RNCB   = ALOG10(RNDFT * CB)
      CALL LNTP(RNCB, F1, XF1, YF1, 7, 2)
      CALL LNTP(RNCB, F2, XF1, YF2, 7, 2)
      DC1RN  = F1 + F2 * DY
C
      C1MAX  = C1MAXB + DC1MAX + DC1RN
C
      SWEEP = SWPLE * 57.2956
```

```
 100  CALL LNTP(SWEEP, AY, XSWP1,YA, 4,4)
      CALL LNTP(SWEEP, BY, XSWP1,YB, 4,4)
C
      DYM14  = DY - 1.4
      IF( DYM14.LT.0.0 ) DYM14 = 0.0
      IF( DY.GT.2.5    ) DYM14 = 1.1
C
      CLMOC1 = AY -BY * DYM14
      CLMAX  = CLMOC1 * C1MAX
C
      C3     = DLNT(DY, SPEED, XDY, XM, CTAB, 8,4,8,2,2)
      D      = DLNT(DY, SPEED, XDY, XM, DTAB, 8, 4, 8, 2,2)
C
      DCLMAX = C3 + (D-C3)* SWEEP/60.
      CLMAX  = CLMAX + DCLMAX
C
 120  DAMAX  = DLNT(SWEEP, DY, XSP, YDYA, FDA, 6,6,6,2,2)
      AMAX   = CLMAX/CLAW  +ALO  +DAMAX
      CLS    = CLA * (AMAX - 2.*DAMAX - ALO)
      GO TO 300
C
C     LOW ASPECT RATIO METHOD
C
  200 CONTINUE
C
      ABETA  = AR/SQRT(1. - SPEED * SPEED)
      XCLM   = ABETA * C1COS
      IF( XMT.LE.0.35.OR.XCLM.GE.2.0 ) CLMXB =  DLNT(XCLM, DY, XXCLM,
     1                     YYDY, FCLMX, 13, 6, 13, 2,2)
      IF( XMT.GT.0.35.AND.XCLM.LT.2.0 ) CLMXB =  DLNT(XCLM, DY, XXXCLM,
     1                     YYDY, FCLMXX, 9, 6, 9, 2,2)
C
      C2TAN  = (C2 + 1.) * AR * TAN(SWPLE)
      DCLMX  = DLNT(C2TAN, SPEED, XXC2, YYMACH, FDCLMX, 9, 5, 9, 2,2)
C
      CLMAX  = CLMXB + DCLMX
      AST    = AR * COS(SWPLE) * (1.+ 4.0 * TR*TR)
      AMAXB  = 35.0
      IF( XCLM.GT.0.9 ) AMAXB = 49.8473 -20.6922*XCLM +5.0674*XCLM**2
     1                          -0.4279*XCLM**3
      IF( XCLM.GT.3.6 ) AMAXB = 21.0
      IF( C2TAN.LE.4.5 ) DAM = DLNT(C2TAN,AST, XC2, YAST,FDAM,6,9,6,4,2)
      IF( C2TAN.GT.4.5 ) DAM = DLNT(C2TAN,SPEED,XCT,YM,FDAM2,6,5,6,4,2)
C
      AMAX   = AMAXB + DAM
C
  300 CONTINUE
C     TAIL CONTRIBUTION TO CLMAX       ******
C
      DEL    = CLAT * 57.3 * SIN(AMAX/57.3) * COS(AMAX/57.3)**2
      CLMAX  = CLMAX + DEL
C
      IF( KPRINT(16).EQ.1 ) WRITE(6,1000) CLFB, CLOF, ABFKC, ABRK,
     1              XMN, DYC, DCLPB, TRANS, BETAC, DY, C1, C2, AMAX, ABUP,
     2                 ARLO,XMT,C1MAXB,DC1MAX,CLMOC1,DCLMAX,DEL,
     3              XCLM, CLMXB, C2TAN, DCLMX, CLMAX, DC1RN, RNCB
 1000 FORMAT(10X,*CLBRK DUMP* /(1X, 7F15.5) )
C
      RETURN
      END
```

```
      SUBROUTINE CDL1(SPEED, RNOFT, FK, DELCL, PRIMEK, AKD, AKB)
C
C     COMPUTES DRAG DUE TO LIFT CONSTANTS
C
      COMMON/INPUT/  A(3081)
      COMMON /CLTPLT/ B(223)
      COMMON /CALC/    C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ E(908)
      COMMON /BLKPRT/ KPRINT(50)
C
      DIMENSION CCLNV(13), RCURV(13), XRN(9), YDRT1(9), YDRT2(9)
      DIMENSION XSWPL(11), YKMIN(11), AA(22), BB(22), CC(22), DD(22),
     1          XSWP(7), YTR(4), FEP35(7,4), FEP7(7,4), XCLDB(7),YAKB(7)
     2       ,SUR(7,20)
C
      EQUIVALENCE (D(1),XSWPL(1)),(D(12),YKMIN(1)),(D(23),AA(1)),
     1            (D(45),BB(1)),(D(67),CC(1)),(D(89),DD(1)),
     2            (D(1331),XSWP(1)),(D(1338),YTR(1)),
     3            (D(1370),FEP7(1,1)),(D(1398),XCLDB(1)),(D(1405),
     4            YAKB(1)),(D(1342),FEP35(1,1)),
     5            (A(194),SUR(1,1)), (RLER,A(23)), (TOCR,A(24)),
     6            (TOCS,C(41)),      (CBARS,C(42)),  (CDMCL,A(27)),
     7            (DCB,C(2)),        (ISWP,A(6)),     (CLD,C(13)),
     8            (SWPQC,C(6)),      (SWPLE,C(25)),   (NPNLS,A(3)),
     9            (TP,C(29)),        (TR,C(30)),      (R,B(61))
     1            ,(EO , B(66))
C
C
      DATA DCURV / 2., 4., 6., 10., 20., 30., 40., 50., 60.,
     1             80., 100., 130., 200./,
     2     RCURV / 0.17,0.37,0.48,0.6,0.73,0.782,0.81,0.83,0.84,
     3             0.855,0.865,0.874,0.874 /
      DATA  XRN   / 0., 5., 7., 10., 20., 40., 100., 300., 600. /,
     1     YDRT1 / .07,.07,.05,.025, 5*0.0 /,
     2     YDRT2 / .175,.175,.165,.12,.05,.03,.015,.005,0.0 /
C
C
      SREF = A(11)
      SPLAN = C(4)
      CB   = C(11)
      CLM   = B(59) *57.29578 * SREF/SPLAN
      RD2X   = C(1) - C(1)*C(2)
      FKL1    = 0.0
      DCL1    = 0.0
      FKL2    = 0.0
      DCL2    = 0.0
      CLMO    = CLM
      CLM2 = B(119) *57.29578 * SREF/SPLAN
      RBAR    = 0.0
C
      CLDB = B(106)
      CALL LNTP(CLDB, AKB, XCLDB, YAKB, 7, 2)
      FAKB     = 1.24 -.04 *.RNOFT * CB * 1.0E-6
      IF( FAKB.LT.1.0 ) FAKB = 1.0
      AKB      = FAKB * AKB * SREF/SPLAN
C
      BX       = -1.41 +1.442*TR -1.26*TR**2 +.528*TR**3
      CX       = 0.7125 -1.497*TR +1.476*TR**2 -.6909*TR**3
      EO       = 1.0 + BX* DCB + CX* DCB**2
C
      FMACH   = SPEED
      FMCRD= B(114)
      FML1 = B(115)
      FML2 = B(116)
      IF (FMACH.GT.FMCRD) FMACH = FMCRD
      IF(SPEED.GE.FMCRD.AND.SPEED.LT.FML2) CLMC = B(118) *
     1                 57.29578 * SREF/SPLAN
      IF( SPEED.GT.1.0 ) CALL ADCL(SPEED, CLOPT)
      IF( SPEED.LT.FML2 ) CALL ADCL(FMACH, CLOPT)
C
      DO 100 I = 1, NPNLS
C
      NI    = SUR(I,1)
      TC    = SUR(I,3)
      RLEDC= AA(NI) +BB(NI)*TC +CC(NI)*TC*TC + DD(NI)*TC**3
      IF (NI.EQ.8) RLEDC = RLER *(TC/TOCR)**1.5
```

```
      RLE   = FLEOC * SUF(I,20)
      IF( ISWP.EC.O.OR.I.NE.NPNLS ) GO TO 90
      RLEOC   = AA(NI) +BP(NI)*TOCS + CC(NI)*TOCS**2 + DD(NI)*TOCS**3
      TC      = TOCS
      IF (NI.EQ.8) RLEOC = RLEF *(TC/TOCR)**1.5
      RLE     = RLEOC * CRAR2
C
   90 CONTINUE
C
      RNLER   = RLE * RNOFT/10.0**3
      COTANS = 5. -6.511 * SUR(I,4)
      IF (SUR(I,4).GT.0.35) COTANS = 1./TAN(SUR(I,4))
      OMEGA   = RNLER * COTANS * SQRT(1.-(FMACH* COS(SUR(I,4)))**2 )
C
      CALL LNTP(OMEGA, RT, OCURV, RCURV, 13, 2)
      CALL LNTP(SUR(I,4),RMIN,XSWPL,YRMIN, 11,2)
      IF(RT.LT.RMIN )  RT = RMIN
      DRT     = 0.0
C
      CALL LNTP(RNLER, DRT1, XRN, YDRT1,5,2)
      CALL LNTP(RNLER, DRT2, XRN, YDRT2,5,2)
      IF( TC.GT.0.03 ) DRT = DRT1 * (TC - 0.03)/0.03
      IF( TC.GT.0.06 ) DRT = DRT1 + DRT2 * (TC - 0.06)/0.06
      RT      = RT + DRT
C
      RI      = RT +(0.824 -RT) *(CLD + CONCL)/0.6
C
      IF(RI.GT.0.874 ) RI = 0.874
C
      YW1     = SUR(I,12)
      YW2    = C(1)
      IF(NPNLS.EQ.2.AND.I.EQ.1) YW2 = SUR(2,12)
      RBAR    = RBAR + RI *(YW2 -YW1)/BO2X
  100 CONTINUE
C
      AT      = AR * TR/COS(SWPLF)
      DELR    = 0.0482*AT -0.01102*AT**2 +0.001197*AT**3
     1          -0.0000483З*AT**4
C
      R       = RBAR + DELR
C
      FK      = ((1.-R)/CLMG +R/(3.14159*AR*EC)) * SREF/SPLAN
C
      EF      = 1./(3.14159 *AR *FK) * SREF/SPLAN
      DELCL   = (CLOPT *(1.-EF)
      PRIMEK = 0.518/SQRT(AR)
C
      SWPC4   = SWPOC * 57.29578
      IF( SWPC4.GT.60.0 ) SWPC4 = 60.0
      EP35    = DLNI(SWPC4, TR, XSWP, YTR, EEP35, 7, 4, 7, 2,2)
      EP7     = DLNI(SWPC4, TR, XSWP, YTR, EEP7,  7, 4, 7, 2,2)
      IF( AR.LE.3.5 ) EP = 1. - (1.-EP35) * AR/3.5
      IF( AR.GT.3.5 ) EP = EP35 + (EP7 - EP35)*(AR-3.5)/3.5
C
      EPP     = EP * (1.-DDR**2)
      AKD     = 1./(3.14159 * AR * EPP) * SREF/SPLAN
C
C
      RO      = R
      H       = 1.1
      ARTANS = AR * TAN(SWPLF)
      IF( ARTANS.GT.3.5 ) H = 1.1 + 0.1*(ARTANS - 3.5)
      IF( SPEED.GE.1.0 ) PRIMEK = H/CLM * SREF/SPLAN
C
      IF( SPEED.LE.FML2 )  GO TO 200
      ZNOM    = 12. * (COS(SWPLF)**1.6) * (SPEED-FMCRD)
      FNOM    = 1./(1. + ZNOM + ZNOM**2)
      R       = RO * FNOM
      FK      = ((1.-R)/CLM  +R/(3.14159*AR*EC)) * SREF/SPLAN
      DELCL   = CLOPT
C
      GO TO 500
C
  200 CALL KGIN(CLDB, FK, DELCL, SPEED, FKL1, DCL1)
      IF( SPEED.LE.FML1 ) GO TO 500
C     DRAG POLAR IS CALCULATED BY LINEAR INTERPOLATION BETWEEN THE
C     LIMITS FML1 AND FML2.
```

```
C
      ZNDM    = 12. * (COS(SWPLE)**1.6) * (FML2 - FMCRC)
      FNDM    = 1./(1. + ZNDM + ZNDM**2)
      RL2     = RC * FNDM
      FKL2    = ((1.-RL2)/CLM2 + RL2/(3.14159*AR*EC)) * SREF/SPLAN
      CALL ADCL(FML2, DCL2)
C
      CALL KCIN(0.50, FK, DELCL, FML1, FKL1, DCL1)
C
      FK      = FKL1 + (FKL2-FKL1)*(SPEED-FML1)/(FML2-FML1)
      DELCL   = DCL1 + (DCL2-DCL1)*(SPEED-FML1)/(FML2-FML1)
C
  500 IF( KPRINT(22).EQ.0 ) RETURN
      WRITE(6,1000) SPEED, FMACH, FK, DELCL, RETNEK, AKD, AKR,
     1              CLDR, CLM, SREF, SPLAN, FMCRC, FML1, FML2,
     2              RLE,RNLER,OMEGA,R,RBAR,RT,CLD,CDNCL,EF,AR,SWPC4,
     3              FP, DOB, SWPLE, FKL1, DCL1, FKL2, DCL2,RMIN
C
 1000 FORMAT(5X, *COL1 DUMP* /(1X, 7F15.5) )
      RETURN
      END
```

```
      SUBROUTINE ADCL(SPEED, CLOPT)
C
C     POLAR AXIS DISPLACEMENT
C
      COMMON/INPUT/  A(3081)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/   C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ E(908)
C
      CONCL   = A(27)
      SPLAN   = C(4)
      SREF    = A(11)
      FCC     = C(26)
      CLD     = C(13)
      SWPLE   = C(25)
      TECH    = A(26)
C
      CLOPT   = 0.0
      IF (SPEED .GE. 1.0) GO TO 10
C
      IF( CLD.GT.0.0 ) CLOPT = -.001 +16.934*FCC -216.2697*FCC**2
     1                          +1751.3562*FCC**3
      IF( CONCL.GT.0.0)CLOPT = CLOPT -.0017 +1.1324*CONCL
     1                          -1.6498*CONCL**2 +1.0605*CONCL**3
      IF (TECH.GT.0.) CLOPT = 0.51951 * CLD**0.75
      GO TO 20
C
   10 BETAT   = 10.
      IF (SWPLE.GT.0.) BETAT = SQRT(SPEED**2 -1.)/TAN(SWPLE)
      DELCL   = CLD * (0.25 - 0.225 * BETAT )
      IF( BETAT.GE.1.11 ) DELCL = 0.0
      CLOPT   = DELCL
C
   20 CLOPT   = CLOPT * SPLAN/SREF
      RETURN
      END
```

```
      SUBROUTINE KGIN(CLDB, AKIN,DECLIN,SPEED, AKOUT,DCLOUT)
C
C     COMPUTES POLAR USING LEAST-SQUARES CURVE FIT
C
      COMMON /BLKPRT/ KPRINT(50)
      DIMENSION CL(11), CCL(11), SA(12), RE(11)
C
      DCL    = CLDB/10.0
      CL(1) = 0.0
C
      DO 20 I = 1, 11
      IF( I.NE.1 ) CL(I) = CL(I-1) + DCL
      CLX    = CL(I)
C
      CALL CDDF(CLX, SPEED, CDR)
C
      IF( I.EQ.1 ) CDRO = CDR
      DCDR   = CDR - CDRO
C
   40 CDL(I) = AKIN * (CL(I) - DECLIN)**2 + DCDR
   20 CONTINUE
C
      CALL LSPCF(CL, CDL, 11, 2, SA, RE, SIGMA, IT)
C
      AKOUT    = SA(1)
      DCLOUT   = -SA(2)/(2.0 * AKOUT )
      ERR      = SA(3) - AKOUT * DCLOUT**2
C
      IF( ABS(ERR).LT.0.001.OR.KPRINT(14).EQ.0 ) GO TO 30
C
      WRITE (6,1000) SPEED,ERR,AKOUT,DCLOUT,
     1        AKIN, DECLIN, ( CL(I), CDL(I), I = 1,7 )
 1000 FORMAT(10X,*KGIN* 5X,*MACH =*F8.5,2X*ERR =*F8.5,2X,4F8.5,/
     1        1X,14F8.5      )
C
   30 RETURN
      END
```

```
      SUBROUTINE CDL2(SPEED, CL, AEROK, DELCL, PRIMEK, AKD, AKP, CDL)
C
C     COMPUTES DRAG DUE TO LIFT
C
      COMMON/INPUT/  A(3081)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/   C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ E(908)
      COMMON /BLKPRT/ KPRINT(50)
C
      CLPB   = B(105)
      CLDB   = B(106)
C
C
   20 CONTINUE
      CDL     = AEROK * (CL - DELCL)**2
      IF( CL.LE.CLPB ) GO TO 500
C
C     DRAG DUE TO LIFT ABOVE POLAR BREAK
      CDL     = CDL + PRIMEK * (CL - CLPB)**2
      IF( CL.LE.CLDB ) GO TO 500
C
C     DRAG DUE TO LIFT ABOVE DRAG BREAK (CLDB)
C
      IF( SPEED.GE.1.0 ) GO TO 410
      CDPDB   = AEROK * (CLDB - DELCL)**2  - AKD * CLDB**2
     1              + PRIMEK * (CLDB - CLPB)**2
C
      DCDB    = AKP * (CL - CLDB)**2
      DCDB    = DCDB + 0.08 * SQRT(DCDB)
C
      CDL     = CDPDB + DCDB + AKD * CL**2
C
  400 IF( SPEED.LT.1.0 ) GO TO 500
  410 CONTINUE
      CDL     = (AEROK - PRIMEK) * (CLPB - DELCL)**2
     1           + PRIMEK * (CL - DELCL)**2
C
C
C
  500 CONTINUE
      IF( KPRINT(14).LE.0 ) GO TO 50
      WRITE(6,1000) SPEED, CL, CDL, AEROK, DELCL, PRIMEK, CLPB,
     1              CLDB, AKD, AKP
      KPRINT(14) = KPRINT(14) - 1
   50 CONTINUE
      RETURN
 1000 FORMAT (10X, *CDL2 DUMP*/(1X, 7F15.5 )  )
      END
```

```
      SUBROUTINE AERO(SPEED, CL, ALPHA)
C
C     CALCULATES ANGLE OF ATTACK
C
      COMMON/INPUT/  A(3081)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/    C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ E(908)
      COMMON /BLKPRT/ KPRINT(50)
C
      DIMENSION CLTAB(13), ATAB(13), SLP(7,30)
      DIMENSION XAB(6),YCD(8),EKVCFM(6,8), XANG(10),YRTDC(7),FRA(10,7),
     1          XX(12),YY(7),XF(12,7), XIN13(10),YIN13(7),ZOUT13(10,7),
     2          ZOUT14(10,7)
      EQUIVALENCE (CLA,B(89)), (ALO,B(90)), (CLAW,B(96)),
     1            (CLCP,B(106)),(CLMAX,B(107)),(AMAX,B(109)),
     2            (DAMAX,B(110)),(DCL,B(111)), (CLS,B(112)),
     4            (ISWP,A(6)),
     5            (NPNLS,A(3)), (TOC,C(12)), (TR,C(30)),
     6            (SUR(1,1),A(184)),(YIX,C(31)), (YCX,C(44)),
     7            (SWPLE,C(25)), (SWPMC,C(7)), (SWPTE,C(8)),
     8            (ARI,C(33)), (AR,C(29)),(ARLO,B(113)), (SREF,A(11)),
     9            (SEXW,C(10)), (SWPR,C(27)), (CDI,B(48))
C
      EQUIVALENCE (XAB(1),E(484)),(YCD(1),E(490)),(EKVCFM(1,1),E(498)),
     1       (XANG(1),E(546)),(YRTDC(1),E(556)),(FRA(1,1),E(563)),
     2       (XX(1),E(806)),(XY(1),E(818)),(XF(1,1),E(825)),
     3       (XIN13(1),D(1063)),(YIN13(1),D(1073)),(ZOUT13(1,1),
     4 D(1080)),  (ZOUT14(1,1),D(1150))
C
      ID     = SLP(1,1)
      SWEEP = SWPR * 57.29578
      KPRT   = KPRINT(17)
      CLWM1  = 0.0
C
   10 ALPHA = CL/CLA + ALO
      IF( SPEED.GT.1.0 ) RETURN
      IF (NPNLS.EQ.1) GO TO 15
      IF (SUR(1,4).LE.0.7) GO TO 15
      SWO   = SUR(2,4) +    0.087
      IF (ISWP.EQ.1) SWO = SWPR + 0.087
      IF (SUR(1,4).LE.SWO) GO TO 15
      GO TO 250
 15   CONTINUE
C
      IF( AR.LE.ARLO ) GO TO 100
C
   20 CONTINUE
C
      IF( KPRT.GT.0 ) WRITE(6,1000) CL, ALPHA, AR, ARLO, CLMAX, CLS,
     1                    DAMAX, CLA, ALO
C
      IF( CL.LE.CLS ) GO TO 200
C
C     HIGH ASPECT RATIO LIFT METHOD      ****
C
      DA     = 0.0
      DCL     = CLMAX - CLS
      DA      = ((CL - CLS)/DCL)**2 * DAMAX
   30 ALPHA = ALPHA + DA
      IF( CL.LE.CLMAX ) GO TO 200
      ALPHA = ALPHA + 5.0
      IF( ALPHA.GT.90.0 ) ALPHA = 90.0
C
      GO TO 200
  100 CONTINUE
C     LOW ASPECT  RATIO LIFT METHOD     *****
C
      AS       = 0.0
      Z        = 2. * COS(SWPMC) / AR
      HCP      = (AMAX - AS)/114.6 /AR    * 1.5 * (TR +TR**2)/(1.+TR+TR**2)
      XEP      = 1.0014 -1.069*HCP +3.0021*HCP**2 -2.0072*HCP**3
      DCLEP    = (Z + SQRT(1.+ Z**2))/(XEP*Z + SQRT(1. +(XEP*Z)**2))
      DCLPA    = (DCLEP -1.) * CLAW * 57.3
      CLPA     = CLA * 57.3
      DCLO     = - CLA * ALO
```

```
      CLVM1  = CLMAX - CLPA * SIN(AMAX/57.3) * COS(AMAX/57.3)**2 -DCLO
      CLVM2  = CLVM1 - DCLPA * SIN(AMAX/57.3) * COS(AMAX/57.3)**2
      IF( CLVM1.LE.0.0 ) DAMAX = AMAX - CLMAX/CLA - ALO
      IF( CLVM1.LE.0.0 ) CLS = CLA * (AMAX - 2.*DAMAX - ALO)
      IF( CLVM1.LE.0.0 ) GO TO 20
C
      CLVA   = 0.0
      TOCR   = 0.0
      ABETA  = 0.0
      IF (SWPLE.EQ.0.0) GO TO 112
C
      CO      = TAN(SWPTE)/TAN(SWPLE)
      BETA    = SQRT(1. - SPEED**2)
      IF (CC.NE.1.0) ABETA = 4./TAN(SWPLE) /(1.-CC) *BETA
      TOCR    = TOC
      IF ( ID.EQ.9 ) TOCR = 0.0
C
  110 CLVOFM = DLNT(ABETA, CC, XAB, YCD, FKVCFM, 6,8,6,2,2)
      TS2     = TAN(SWPLE)**2
      FM      = SQRT((1. + TS2)/(BETA**2 + TS2))
      CLVA    = CLVCFM * FM * SEXW/SREF
C 112 CONTINUE
C
      ATAB(1)= 0.0
      DO 120 I = 1, 13
      IF( I.GT.1 ) ATAB(I) = ATAB(I-1) + 3.0
      ANG     = ATAB(I)* 0.01745
      DCLP    = 0.0
      IF( ATAB(I).LE.AS ) GO TO 115
C
C     TIP VORTEX EFFECT
C
      HOB     = (ATAB(I) -AS)/114.6 /AR    *1.5 *(TR+TR*TR)/(1.+TR+TR*TR)
      XEP     = 1.0014 -1.969*HOB +5.0021*HOB**2 -2.0072*HOB**3
      IF( HOB.LE.0.0 ) XEP = 1.0
      DCLEP   = (Z +SQRT(1.+Z*Z))/(XEP*Z + SQRT(1.+ (XEP*Z)**2))
      DCLPA   = (DCLEP -1.) * CLAW * 57.3
      DCLP    = DCLPA * SIN(ANG)
  115 CONTINUE
C     LEADING EDGE EFFECT (RA)
      RA      = DLNT(ANG, TOCR, XANG, YRTCC, FRA, 10, 7, 10, 2,2)
C
C     VORTEX BREAKDOWN EFFECT (FVL)
      X       = AR
      FVL     = 0.0
      IF( X.LT.3.37 ) FVL = DLNT(X,ANG, XX, XY, XF, 12, 7, 12, 4,2)
C
      CLV     = (1.-RA) * FVL * CLVA * SIN(ANG)**2 * COS(ANG)
      IF( CLV.GT.CLVM2 ) CLV = CLVM2
      CLP     = CLPA * SIN(ANG) * COS(ANG)**2
      CLTAB(I) = DCLO + CLV + CLP + DCLP
      IF( KPRINT(17).EQ.2.AND.I.EQ.1 ) WRITE(6,1002)
      IF( KPRINT(17).EQ.2 ) WRITE(6,1001) CLTAB(I), ATAB(I), RA,
     1              CLP, CLV, DCLP
C
  120 CONTINUE
C
      IF( KPRT.GT.0 ) WRITE(6,1003) CL, ALPHA, AR, ARLO, Z, HOB, DCLPA,
     1                 CLPA, DCLO, CLVM1, CLVM2, TOCR, CLVA, RA, FVL,
     2                 ANG
C
  125 IF( CLVM1.LE.0.0 ) GO TO 20
  130 CALL LNTP(CL, ALPHA, CLTAB, ATAB, 13, 4)
C
      GO TO 200
C
C     CALCULATES ANGLE OF ATTACK AT HIGH LIFT BY MODIFIED
C     WINSTAN METHOD.
C
C
C
  290 CONTINUE
  300 BETA   = SQRT(1. - SPEED**2)
      BETAN  = BETA *TAN(SLR(1,4))
      AA     = DLNT(BETAN,SWEEP, XIN13,YIN13,ZOUT13, 10,7,10,2,2)
      EN     = DLNT(BETAN,SWEEP, XIN13,YIN13,ZOUT14, 10,7,10, 2,2)
C
C
```

```
      ACLARU = CLDB/CLA + ALO
      EFRK   = YIX/(YIX + YOX)
      CLB    = AA * ACLARU**FN * FRRK/ARI * CLAW * 57.29578
  310 IF( CL.LE.CLDB ) GO TO 200
      CLT    = CLB + CL - CLDB
      ALPHA = (CLT *API/(AA *CLAW *57.29578 *FRRK))**(1./FN)
C
      IF( KPRT.GT.0 ) WRITE(6,1004) CL, ALPHA, AR, A, EN, ACLARU,
     1                    FRRK, CLB, CLDB, CLAW
C
  200 CONTINUE
      IF( ALPHA.LT.6.0 ) GO TO 205
      CDLA   = CL * TAN(ALPHA/57.296)
      IF( CDLA.LT.CDL ) CDL = CDLA
  205 RETURN
C
 1000 FORMAT(5X, *AERA DUMP   HIGH ASPECT RATIO* /(1X,7F15.5) )
 1003 FORMAT(5X, *AERA DUMP   LOW ASPECT RATIO* /(1X,7F15.5) )
 1004 FORMAT(5X, *AERA DUMP   CRANKED WING* /(1X,7F15.5) )
 1001 FORMAT(5X,*VORTEX LIFT*, 6F15.5)
 1002 FORMAT(///5X,*VORTEX LIFT*, 8X,*CL*,12X,*ALPHA*,13X,*R*, 13X,
     1               *CLP*, 11X,*CLV*, 11X, *DCLF*, / )
      ENTRY AERA1
  400 KPRT   = KPRINT(17) - 1
      ALPHA  = CL/CLA + ALO
      IF( SPEED.GT.1.0 ) RETURN
      IF( AR.LE.ARLO.AND.CLVM1.GT.0.0 ) GO TO 125
      IF (NPNLS.EQ.1) GO TO 20
      IF (SUR(1,4).LE.0.7) GO TO 20
      IF (SUR(1,4).LE.SWO) GO TO 20
      GO TO 310
C
C
      END
```

```
      SUBROUTINE AFTCD(ALPHA, CDAFT)
C
C     COMPUTE DRAG INCREMENT DUE TO FUSELAGE LPSWEEP
C
      COMMON/INPUT/  A(3061)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/   C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ F(908)
C
      DIMENSION X(6), Y(4), Z(6,4)
C
      DATA  X  / -5., 0., 5., 10., 15., 20. /,
     1      Y  / 0., 5., 10., 15. /,
     2      Z  / 2*0.0, .01, .045, .125, .26, 0.,.0005,.013,.054,.142,
     3           .33, 0.,.004,.0175,.065,.172,.41, 0.,.014,.023,
     4           .065, .215, .53 /
C
      WINC   = C(39)
      BFUS   = A(28)
      AB     = A(29)
      IF( AB.EQ.0.0 ) AB = 1.0
      PAMX   = A(79)
      SREF   = A(11)
C
      CDAFT  = 0.0
      IF( BFUS.EQ.0.0 ) RETURN
      AFUS    = ALPHA - WINC
      BMA     = BFUS - AFUS
C
      DCD     = DLNT(BMA, AFUS, X,Y,Z, 6,4,6, 4,2)
      CDAFT   = DCD * PAMX/SREF * AB/1.5
C
      RETURN
      END
```

```
      SUBROUTINE CDRG(SPEED, AEROK, DELCL,CDC)
C
C     CAMBER DRAG
C
      COMMON /INPUT/ A(3081)
      COMMON /CALC/  C(50)
C
      EQUIVALENCE (AR,C(29)), (SEXW,C(10)), (SPLAN,C(4)), (SREF,A(11))
      CDC   = 0.0
      ARSREF = AR * SPLAN/SREF
      E      = 1.0/(3.14159*ARSREF*AEROK)
C
C
      COEF1  = 1.0/(3.14159*ARSREF*(1.0-E))
      COEF2  = 0.7 * SEXW/SREF
      IF( E.GE.1.0 ) GO TO 15
      IF( SPEED.GE.1.0 ) GO TO 5
      IF( COEF1.GT.COEF2 ) GO TO 15
    5 CDC     = COEF1 * DELCL**2
      GO TO 10
   15 CONTINUE
      CDC     = COEF2 * DELCL**2
C
   10 RETURN
      END
```

```
      SUBROUTINE CMOW(SPEED, CMO)
C
C     COMPUTES ZERO LIFT PITCHING MOMENT OF WING
C
      COMMON/INPUT/  A(3081)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/    C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ E(908)
      COMMON /BLKPRT/ KPRINT(50)
C
      DIMENSION CMC4(20), CAM(2), SEX(2), XBD4(5),YBD4(5),ZBD4(3),
     1          FBD4(5,5,3)
      EQUIVALENCE (FMCR,B(114)), (TS,A(184)), (ARXR,C(14)),
     1            (TOC,C(12)),(TWIST, C(38)), (SEXW,C(10)),
     2            (CAM(1),A(191)), (SEX(1),A(203)),
     3            (NPNLS,A(3)), (SWPCC,C(6)), (CRX,C(15)),(CTX,C(16)),
     4            (XBD4(1),D(589)),(YBD4(1),D(594)),(ZBD4(1),D(599)),
     5            (FBD4(1,1,1), D(602))
      DATA CMC4 / 4*-0.25, 3*-0.219, -0.3, 0.0, 11*-0.2066 /
C
C
      ID     = TS
      CMOS   = 0.0
C
      DO 100 I = 1, NPNLS
      CMOS   = CMOS + CAM(I)* SEX(I) * CMC4(ID)
  100 CONTINUE
C
      CMO     = 0.0
      IF( SPEED.GE.1. ) RETURN
C
      CMOS     = CMOS/SEXW
C
      CMOB   = ARXR * COS(SWPCC)**2/(ARXR +2.*COS(SWPCC)) * CMOS
C
      TRX    = CTX/CRX
      ESWCC = SWPCC
      CALL TLNT(ESWCC,ARXR,TRX, CMDCT, XBD4,YBD4,ZBD4,FBD4, 5,5,3, 5,5)
      CMDT    = CMDCT * TWIST
C
      FMACH  = SPEED
      IF( FMACH.GT.FMCR ) FMACH = FMCR
      CMACH   = (1. + 5.9*TOC*FMACH**5)/SQRT(1.-(FMACH*COS(ESWCC))**2)
C
      CMO     = (CMOB + CMDT) * CMACH
C
      IF( KPRINT(25).GT.0 ) WRITE(6,1000) CMO, CMOB, CMDT, CMACH,
     1                       CMOS, ARXR, TOC, TWIST, ESWCC, TRX, CMDCT, FMCR,
     2               CTX,  CRX, SWPCC, SEXW, CMC4(ID),
     2             TS, ( CAM(I), SEX(I), I =1,NPNLS)
C
 1000 FORMAT(5X, *CMOW DUMP*/(1X,7F15.5) )
C
      RETURN
      END
```

```
      SUBROUTINE WBAC(SPEED, XACR)
C
C     COMPUTES PITCHING MOMENT SLOPE OF WING
C
C
      COMMON /INPUT/  A(3081)
      COMMON /OUTPUT/ B(223)
      COMMON /BLKPRT/ KPRINT(50)
      COMMON /CALC/ BD2, DOB, CR, SPLAN, XCRTE,
     1              SWPCC, SWPMC, SWPTE, SWMT, SEXW, CPAR,
     2              TOC, CLD,ARXR,CRX,CTX,XLF,XTX,
     3              YRX,YTX,XB,YB,XH,OMEGA,SWPLE,FOC,
     4              SWPR,KPASS,AR,     TR,YIX,SIY,ARI,CBXP,
     5              SCXP,ARCP,DXCC,TWIST,WINC,XHT,TCCS,CPAR2,CLDS,
     6              YCX,SWPMCS,DA1,DAC1,DA2,CAC2,CTOC
C
      DIMENSION XDB(8), YDOB(8), XBDOL(5), YAON(6), FCP(5,6)
      DIMENSION SUR(7,30)
      EQUIVALENCE (SUR(1,1),A(184)), (NPNLS,A(3)), (XACS,B(130))
     1           ,(ISWP,A(6)), (NBODYS,A(1))
C
      DATA XDB  / 0., .05, .1, .15, .2, .3, .4, .5 /,
     1     YDOB / 0., .1, .154, .19, .219, .266, .3, .33 /
C
      DATA XBDOL / 0.4, .7, 1.0, 1.25, 1.67 /,
     1     YAON  / 0., .4, .8, 1.2, 1.6, 2.0 /,
     2     FCP   / .54,.535,.525,.516,.5, .42,.435,.45,.46,.46,
     3             .35,.377,.4,.414,.425, .295,.33,.355,.375,.394,
     4             .246,.285,.32,.345,.365, .21,.25,.288,.315,.34 /
C
       AR1     = ARXR
       TR1     = CTX/CRX
       TW     = A(184)
       FMCRO  = B(114)
       ELN    = A(93)
       CLAB   = B(97)
       CLAW   = B(96)
       SWPLEI = SUR(1,4)
       SWPLEO = SUR(2,4)
       SWPMCI = SUR(1,22)
       SWPMCO = SUR(2,22)
C
       IF(NPNLS.GT.1) AR1 = ARI
       IF (NPNLS.GT.1) TR1 = SUR(1,9)/CRX
       IF (ISWP.EQ.0) GO TO 20
       SWPLEO = SWPR
       SWPMCO = SWPMCS
C
   20  CALL ACCR(SPEED,AR1,SWPLEI,SWPMCI,TR1,SIY,TCC,TW,FMCRO,XACR,CLAI,
     1                    XACSW)
       XACRX   = XACR
       IF( NPNLS.EQ.1 ) GO TO 100
C
C      CRANKED WING PLANFORMS
C.
       TR2     = CTX/CBXP
C
       CALL ACCR(SPEED,AROP,SWPLEO,SWPMCO,TR2,SCXP,TCC,TW,FMCRO,
     1                    XACRP, CLACP, XACSOP)
C
       XACRO   = XACRP * CBXP/CRX - YIX*0.5*TAN(SWPLEO)/CRX
     1                 + YIX * TAN(SWPLEI)/CRX
       XACSO   = XACSOP * CBXP/CRX +XACRO - XACRP*CBXP/CRX
C
       XACR    = (CLAI * SIX * XACRX + CLACP * SCXP * XACRO )/
     1                (CLAI * SIX + CLACP * SCXP)
C
       XACSW   = (CLAI * SIX * XACRX + CLACP * SCXP * XACSO )/
     1                (CLAI * SIY + CLACP * SCXP)
  100  XACWB   = XACR
C
C
C      WING-BODY COMBINATION
C
C      XLE IS THE X-STA. OF L.E. OF INBOARD EXPOSED WING ROOT
C
       DIA     = DOB * BD2 * 2.
```

```
      IF( SPEED.GE.1.2 ) GO TO 200
C     SUBSONIC CALCULATION OF XACN (NOSE) AND XACBW (WING CARRY-OVER)
      XLEQ    = BLN + 1.6 *(XLE - BLN)
      XACN    = -0.54 * XLEQ/CRX
C
      BARE    = ARXR * SQRT(1.-FMCRC**2)
      IF( SPEED.LT.FMCRC ) BARE = ARXR * SQRT(1.-SPEED**2)
      CALL LNTP(DCP, FDCB, XCP,YDCB, 8,4)
      XACBW   = 0.25 + DXOC * FDOB/CRX
C
      IF( BARE.GE.4. ) GO TO 190
      XACBWO = 0.125 * ARXR * TAN(SWPLEI) * (1.+CTX/CRX)
C
      XACBW   = (XACBWO - XACBW) * (BARE -4.)**2/16. + XACBW
  190 XACBW1  = XACBW
      XACN1   = XACN
C
      IF( SPEED.LE.FMCRC ) GO TO 290
C
C     SUPERSONIC CALCULATION OF XACN + XACBW
  200 BETA    = 0.663325
      XACN    = 0.
      XACBW   = 0.
      IF (NBODYS.EQ.0) GO TO 290
      IF( SPEED.GT.1.2 ) BETA = SQRT(SPEED**2 -1.)
C
      AON     = (XLE - BLN)/BLN
      IF( AON.LT.0. ) AON = 0.
      BDOL    = BETA *DIA /BLN
C
C     FIGURE 4.2.2.1-23A DATCOM                           *******
      XCPOL   = DLNT(BDOL, AON, XBDOL,YAON,FCP, 5,6,5, 2,2)
      XACN    = XLF/CRX * (XCPOL -1.)
C
      BDOC    = BETA * DIA/CRX
      BCOT    = BETA * TAN(SWPLEI)
      A1      = 0.5845
      IF( BCOT.GE.1. ) A1 = 0.5985 + 0.00214*ALOG(BCOT)
      IF( BCOT.LT.1.3.AND.BCOT.GT.0.1) A1 = 0.5985 + 0.00607*ALOG(BCOT)
C
C     FIGURE 4.3.2.1-37A DATCOM                           *******
      XACBW   = 0.5 + A1*BDOC -.1057*BDOC**2 +.0172*BDOC**3
      IF( SPEED.GE.1.2 ) GO TO 290
C
      XACN    = XACN1 + (XACN -XACN1) *(SPEED-FMCRC)/(1.2-FMCRC)
      XACBW   = XACBW1+ (XACBW-XACBW1)*(SPEED-FMCRC)/(1.2-FMCRC)
C
  290 CONTINUE
      FKBW    = 0.0004 +1.2662*DOB +.6018*DOB**2 +.1263*DOB**3
      FKWB    = 1.0025 +.7116*DOB +.42*DOB**2 -.1266*DOB**3
      CLABW   = FKBW/(FKBW + FKWB) * CLAW
      CLAWB   = FKWB/(FKBW + FKWB) * CLAW
C
  300 XACB    = (XACN * CLAB + XACWB * CLAWB + XACBW * CLABW)/
     1                 (CLAB + CLAW)
C
      XACS    = (XACN * CLAB + XACSW * CLAWB + XACBW * CLABW)/
     1                 (CLAB + CLAW)
C
      IF( KPRINT(IP).EQ.0 ) GO TO 400
      WRITE(6,1000) XACB, XACN,CLAB, XACWB,CLAWB, XACBW,CLABW,
     1                 XACS, SPEED, XLEQ, FDOB
C
  400 RETURN
 1000 FORMAT(5X,*WBAC DUMP* /(5X,6F15.5) )
      END
```

```
      SUBROUTINE ACCR( SPEED,AR,SWPLE,SWPMC,TR,SPLAN,TCC,TW,FMCRD,
     1                 XACR, CLAX, XACS )
C
C     COMPUTES AERODYNAMIC CENTER OF SINGLE PANEL WINGS
C
C
C     XACR IS THE A.C. REFERENCED TO L.E. OF EXP. ROOT CHORD
C     CLAX IS THE LIFT-CURVE SLOPE OF EXP. PLANFORM
C     XACS IS THE A.C. AT STALL
C
      COMMON/INPUT/  A(3061)
      COMMON /OUTPUT/ B(223)
      COMMON /CALC/   C(50)
      COMMON /BLKDAT1/ D(1411)
      COMMON /BLKDAT2/ E(908)
      COMMON /BLKCLA/ X(11)
      COMMON /BLKPRT/ KPRINT(50)
C
      DIMENSION XDY(6), YAC(6), XVAL(3), VAL(3)
      DIMENSION RDT(12),ATSW(6),TPRT(6),FYAC(12,6,6),
     1          XBD6(4),YBD6(4),ZBD6(6),FBD61(4,4,6),FBD62(4,4,6),
     2          FBD63(4,4,6), AMAP(22)
C
      EQUIVALENCE (RDT(1),D(133)),(ATSW(1),D(145)),(TPRT(1),D(151)),
     1          (FXAC(1,1,1),D(157)), (XBD6(1),D(725)),
     2          (YBD6(1),D(729)),(ZBD6(1),D(733)),(FBD61(1,1,1),D(739)
     3          ),(FBD62(1,1,1),D(835)),(FBD63(1,1,1),D(931)),
     4          (AMAP(1),D(1030))
C
      DATA  XVAL / 0.0,  0.2,  0.5 /
      DATA  XDY / .2, .4, .6, .8, 1.2, 1.6 /,
     1      YAC / .67,.585,.55,.555,.59,.61 /
C
      NI     = TW
      DY     = AMAP(NI) * TCC
      ARLO   = B(113)
C
      SB     = 2.
      IF( SPEED.EQ.1.0 ) GO TO 20
      IF( SPEED.GT.1.0 ) GO TO 10
C
      TANOB  = TAN(SWPLE)/SQRT(1.-SPEED**2)
      IF( TANOB.LE.1.0 ) SB = TANOB
      IF( TANOB.GT.1.0 ) SB = 2. - 1./TANOB
      GO TO 20
C
   10 TANOB  = TAN(SWPLE)/SQRT(SPEED**2 -1.)
      IF( TANOB.LE.1. ) SB = 4.-TANOB
      IF( TANOB.GT.1. ) SB = 2.+1./TANOB
C
   20 ARTSW  = AR * TAN(SWPLE)
C
      CALL TLNT(SB,ARTSW,TR, XAC1, RDT,ATSW,TPRT,FYAC, 12,6,6, 12,6)
C
      KPR11       = KPRINT(11)
      KPRINT(11) = 0
      X(1)    = SPLAN
      X(2)    = 0.0
      X(3)    = TR
      X(4)    = AR
      X(5)    = 0.0
      IF( TW.EQ.6. ) X(5) = 0.0334
      X(6)    = 0.0
      X(7)    = 0.0
      IF( TW.EQ.8. ) X(7) = 0.09
      X(8)    = 0.0
      X(9)    = 0.0
      IF( TW.EQ.8. ) X(9) = 1.173763 * TCC
      X(10)   = SWPMC
   30 X(11)   = SPLAN
C
      CALL AER7(SPEED, CLAX)
C
      XAC2    = XAC1
      VBAR    = (SPEED**2 -1.)/TCC**0.6667
      ABAR    = AR * TCC**0.3333
C
```

```
      IF( VBAR.GE.1.0.OR.VBAR.LE.-2. ) GO TO 40
C
C     TRANSONIC AERODYNAMIC CENTER
C
      CALL TLNT(ABAR,VBAR,ARTSW,VAL(1),XBD6,YBD6,ZBD6,FBD61,4,4,6,4,4)
      CALL TLNT(ABAR,VBAR,ARTSW,VAL(2),XBD6,YBD6,ZBD6,FBD62,4,4,6,4,4)
      CALL TLNT(ABAR,VBAR,ARTSW,VAL(3),XBD6,YBD6,ZBD6,FBD63,4,4,6,4,4)
C
      CALL LNTP(TR, XAC2, XVAL, VAL, 3, 2)
C
   40 XACR    = XAC1
      IF( SPEED.LE.FMCRD.OR.SPEED.GE.1.2 ) GO TO 50
      FM1     = FMCRD + 0.05
      FM2     = SQRT(1.+ TCC**.66667)
      XACR    = XAC2
C
      IF( SPEED.GT.FM1.AND.SPEED.LT.FM2 ) GO TO 50
      IF( SPEED.LE.FM1 ) XACR = XAC1 + (XAC2-XAC1)*(SPEED-FMCRD)/.05
      IF( SPEED.GE.FM2 ) XACR = XAC2 + (XAC1-XAC2)*(SPEED-FM2)/
     1                          (1.2 - FM2)
C
   50 ARHI    = 2.* ARLO
      XACS    = (1.+2.*TR)/12.* ARTSW  + (1. + TR**2/(1.+TR))/3.
      IF( AR.GT.ARHI ) GO TO 60
      CALL LNTP(DY, XACS1, XDY, YAC, 6, 2)
      XACS    = XACS1 + (1.+2.2*TR)/17.544 * ARTSW -0.2
C
   60 IF( KPRINT(18).EQ.2 ) WRITE(6,1000) SPEED, AR, SWPLE, SWPMC,
     1                         TR, SPLAN, TCC, TW, FMCRD, XACR, CLAX,
     2                         SB, ARTSW, XAC1, ABAR,  VBAR, VAL, XAC2,
     3                         FM1, FM2, ARHI, XACS
C
 1000 FORMAT(5X, *ACCR DUMP* /(5X,6F15.5) )
C
      KPRINT(11) = KPR11
      RETURN
      END
```

```
      OVERLAY (4,0)
      PROGRAM NUCDAM
C
C     CALCULATES NUCLEAR DAMAGE AERODYNAMIC EFFECTS
C
      COMMON /INPUT/   A(3081)
      COMMON/OUTPUT/ B(223)
      COMMON /CALC/ C(50)
      COMMON /BLKPRT/  KPRINT(50)
      COMMON /BLKTIL/  TITLE(6)
      COMMON /NUCOUT/  DAMCD(7,17), SYM(7,17), DCDMIN,
     1          ALP(21), CL(21),    CD(21),    CM(21),    CLT(21),   CDT(21),
     2                   CLP(21),   CDP(21),   CMP(21),   CLTP(21),  CDTP(21),
     3                   BCLA(6), DXAC,
     4                   CROLL(21), CDRT(21), X3(2), X4(2)
      DIMENSION  BNAME(7), BOD(7,20), SNAME(7), SUR(7,30), FMSURV(20),
     1           ALT(20), DBOD(7,100), DSUR(7,100), IDAM(20)
     2          ,DWING(20), CDSUR(7,5), DRGSUR(7), CDBOD(7,5)
      EQUIVALENCE  (NBODYS,A(1)), (NNAC,A(2)), (NPNLS,A(3)),
     1             (NHT,   A(4)), (NVT , A(5)), (SREF ,A(11)),
     2             (METRF,A(10)), (ROUGHK,A(16)), (BNAME(1),A(30)),
     3             (BOD(1,1),A(37)), (SNAME(1),A(177)),
     4             (SUR(1,1),A(184)), (NSURV,A(491)),
     5             (FMSURV(1),A(492)), (ALT(1),A(513)),
     6             (DBOD(1,1),A(1662)), (DSUR(1,1),A(2362))
     7            ,(IDAM(1)  ,A(1642)),  (DWING(1), A(3062))
     8            ,(CDSUR(1,1), B(154)), (BD2, C(1))
     9            ,(CR    ,C(3) ), (CBAR  ,C(11)), (TOCW    ,C(12)),
     1             (CTX, C(16)), (CDBOD(1,1),B(189))
C
      COMMON /BLKDAT3/XMACH3(17),  CDMK3(17),
     1                XLOH3(6)  ,  HOX3(5)   ,     CDD3(6,5)  ,
     2                XCC3(6)   ,  CLD3(4)   ,     XM3(6,4)   ;
     3                YMACH3(13),  CPF3(13) ,
     4                ZMACH3(13),  CPA3(13) ,
     5                XLOH4(12) ,  HOX4(7)   ,     CDD4(12,7) ,
     6                RATIO3(8) ,  UMACH(10),      DRAG3(8,10),
     7                XLOD(6)   ,  VMACH(6) ,      DRAG4(6,6)
      NBODS    =   NBODYS   + NNAC
      NSUR     = NPNLS + NHT + NVT
C
C     ZERO OUT THE DAMCD(I,K) ARRAY
      DO 6   I= 1,7
      DO 2   K = 1, 17
      DAMCD(I,K) =  C.0
      SYM(I,K)   =   0.C
    2 CONTINUE
    6 CONTINUE
C
C             DETERMINE IF DAMAGE TO AIRCRAFT IS SYMMETRICAL
C
      DO   8    I = 1, 7
      DO   7    J = 1, 61, 10
      K      =  1   + (J-1)/10
      IF (DBOD(I,J).LT.0.0)     SYM(I,K)      = 1.C
      IF (DSUR(I,J).LT.C.0)     SYM(I,K+10) = 1.C
      IF (DSUR(I,J+1).LT.0.0) SYM(I,K+10) = 1.C
      DBOD(I,J)     =   ABS( DBOD(I,J))
      DSUR(I,J)     =   ABS( DSUR(I,J))
      DSUR(I,J+1) =   ABS( DSUR(I,J+1))
    7 CONTINUE
    8 CONTINUE
C
      REWIND 10
C
      DO 500 J =1, NSURV
      XMACH = FMSURV(J)
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5000)   XMACH
C
      READ (10) CL, CD, CM, ALP, CLT, CDT
      READ (10) B
C
      AAA    = ABS(B(144)-XMACH)
      IF (AAA.GT.C.CC1) WRITE(6,1001)
 1001 FORMAT(//,20X,* INCORRECT MACH NUMBER WAS READ FROM *,
     1                      *TAPE UNIT NUMBER 10   * )
```

```
C
      RNOFT = B(146)
C
C
C*************             MODE 1                  **********************
C************* DRAG DUE TO ROUGNESS ON BODIES **********************
C
   10 IF (IDAM(1).NE.1)  GO TO 20
C
C
   11 DO 12   I =  1,  NBODS
      BBB     =   DBOD(I,1) + DBOD(I,2)
      IF(BBB.EQ.0.0)  GO TO 12
C     CALCULATE DRAG OF UNDAMAGED BODIES
C     CALL CFK3 (LENGTH, RN/FT, XCL)1, XCL)2, K1, K2, K3, MACH, CF)
      CALL CFK3 (BOD(I,1), RNOFT, 0.0, 0.0, DBOD(I,1),0,0,XMACH, CF)
      CDI     =   CF * BOD(I,4) / SREF
C     CALCULATE DRAG OF DAMAGED BODIES
      CALL CFK3 (BOD(I,1), RNOFT, DBOD(I,3), DBOD(I,4), DBOD(I,1),
     1            DBOD(I,2), DBOD(I,1), XMACH, CF)
      CDF     =   CF * BOD(I,4) / SREF
C
C     CALCULATE FORM FACTOR (FF) FROM SURVEY DRAG RESULTS
      FF      = (CDBOD(I,1) +CDBOD(I,2))/CDBOD(I,1)
C
      DAMCD(I,1)     = (CDF-CDI) *DBOD(I,5) *FF
      IF (DBOD(I,5).GT.1.0) DAMCD(I,1) = (CDF-CDI) *DBOD(I,5)
     1                                   *FF /BOD(I,4)
      IF (KPRINT(28).EQ.1)
     1WRITE (6,5011) J,I, CDI, CDF, CF, DAMCD(I,1)
   12 CONTINUE
C
C*************             MODE 2                         ***************
C************* DRAG DUE TO FWD-FACING STEPS ON BODIES ***************
C
   20 IF (IDAM(2).NE.1)  GO TO 30
      CALL LNTP (XMACH, CPF, YMACH3, CPF3, 13, 4)
      DO 23    I = 1, NBODS
      SUMQ     = 0.0
      IF (DBOD(I,11).EQ.0.0)  GO TO 22
      IF (DBOD(I,11).EQ.1.0)  DXCL =0.0
      IF (DBOD(I,11).GT.1.0)  DXCL =(DBOD(I,15)-DBOD(I,14))
     1                              /(DBOD(I,11)- 1.0)
C     DO 21  SUMS  Q)EFF / Q)FREESTREAM  FOR ALL STEPS
      N       = DBOD(I,11)
      DO 21   K = 1, N
C
      XOL     = DBOD(I,14)   +  DXCL * (K-1)
      XI      = XOL   *   BOD(I,1)
      HEIGHT =   DBOD(I,13)
      CALL EFFQ(RNOFT, XI, XMACH, HEIGHT, QEFF)
      SUMQ    = SUMQ + QEFF
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5021)  J,I,K, XI, QEFF, CPF, SUMQ
   21 CONTINUE
C
C
   22 DAMCD(I,2) = CPF * SUMQ * DBOD(I,12) * DBOD(I,13)/SREF
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5022)  DAMCD(I,2)
   23 CONTINUE
C
C*************             MODE 3                         ***************
C************* DRAG DUE TO AFT-FACING STEPS ON BODIES ***************
C
   30 IF (IDAM(3).NE.1)  GO TO 40
C
      CALL LNTP (XMACH, CPA, ZMACH3, CPA3, 13, 4)
      DO 33    I = 1, NBODS
      SUMQ     = 0.0
      IF (DBOD(I,21).EQ.0.0)  GO TO 32
      IF (DBOD(I,21).EQ.1.0)  DXCL  = 0.0
      IF (DBOD(I,21).GT.1.0)  DXCL  =(DBOD(I,25) - DBOD(I,24))
     1                               /(DBOD(I,21) - 1.0)
C     DO 31  SUMS  Q)EFF / Q)FREESTREAM FOR ALL STEPS
      N       = DBOD(I,21)
      DO 31   K = 1, N
```

```
      XOL     = DBOD(I,24) + DXOL * (K-1)
      XI      = XOL + POD(J,1)
      HEIGHT  = DBOD(I,23)
      CALL EFFQ(ENOFT, XI, XMACH, HEIGHT, QEFF)
      SUMQ    = SUMQ + QEFF
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5031) J, I,K, XI, QEFF, CPA, SUMC
   31 CONTINUE
   32 DAMCD(I,3) = -CPA * SUMQ * DBOD(I,22) * DBOD(I,23)/SREF
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5032) DAMCD(I,3)
   33 CONTINUE
C
C*************              MODE  4
C************* DRAG DUE TO HOLES IN FUSELAGE ************************
C                                           ************************
   40 IF (IDAM(4).NE.1)  GO TO 50
C
      DO 44      I = 1,  NPODS
C     DXOL    =  DELTA-X FOR EACH HOLE
      IF (DBOD(I,31).EQ.0.0) GO TO 44
      IF (DBOD(I,31).EQ.1.0) DXOL = 0.0
      IF (DBOD(I,31).GT.1.0) DXOL = (DBOD(I,37)-DBOD(I,36))
     1                              / (DBOD(I,31)-1.0)
      CDO     =  0.0
C
C     DO 42   SUMS CDO FOR ALL HOLES, BASED ON CORRECT X-STA., BODY(I)
      N       = DBOD(I,31)
      DO 42   K = 1, N
C
      XOL     = DBOD(I,36) +(K-1) * DXOL
      X       =  XOL *  POD(I,1)
      HOX     = DBOD(I,35) / X
      XLOH    = DBOD(I,33) /DBOD(I,25)
C
      IF (DBOD(I,38).EQ.2) GO TO 41
C
C     CALCULATIONS FOR MISSING PANELS
      IF (HOX.GT.0.1)  HOX = 0.1
      IF (XLOH.GT.10.0)  XLOH = 10.0
      DCDO    =  DLNT(XLOH, HOX, XLOH3, HOX3, CDO3, 6, 5, 6, 2, 2)
      GO TO 43
C
C     CALCULATIONS FOR CAVED-IN PANELS
   41 IF (HOX.GT.0.015) HOX = 0.015
      IF (XLOH.GT.120.0) XLOH = 120.0
      DCDO =   DLNT(XLOH, HOX, XLOH4, HOX4, CDO4, 12, 7, 12, 2, 2)
C
C
   43 CDO     =   CDO + DCDO
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5041) J, I, K, X, DCDO, CDO
   42 CONTINUE
      CALL LNTP (XMACH, CONK, XMACH3, CONK3, 17, 4 )
      DRAG    =(CDO * CONK *DBOD(I,34)*DBOD(I,35)) / SREF
C     MAGNIFY DRAG OF HOLES OVER WING BY 2.31
      DRAG    = DRAG*( 1.+  1.31 * DBOD(I,32) / DBOD(I,31))
C
C     DAMCD(I,K) = DAMCD(BODY NO, DAMAGE MODE)
      DAMCD(I,4) = DRAG
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5042) CONK ,DAMCD(I,4)
   44 CONTINUE
C
C*************              MODE 5
C************* DRAG DUE TO WAVES ON BODIES ************************
C                                          ************************
   50 IF (IDAM(5).NE.1)  GO TO 60
C
      DO 58  I=  1, NPODS
      IF (DBOD(I,41).EQ.0.0)  GO TO 58
      IF (XMACH.GE.1.0)  GO TO 53
C
C                S U B S O N I C
C
      SUMCD   = 0.0
      IF (DBOD(I,41).EQ.0.0)  GO TO   511
```

```
      IF (DBOD(I,41).EQ.1.0)  DXOL  = 0.0
      IF (DBOD(I,41).GT.1.0)  DXOL  = (DBOD(I,47)-DBOD(I,46))
     1                               / (DBOD(I,41)- 1.0)
      N       = DBOD(I,41)
      DO 51   K = 1, N
      XOL     =  DBOD(I,46) + DXOL * (K-1)
      XI      =  XOL  + BOD(I,1)
      HEIGHT = DBOD(I,45)
      CALL EFFO(RNOFT, XI, XMACH, HEIGHT, OEFF)
      DRAG    = 15.0 * (DBOD(I,45)/DBOD(I,43))**2 * OEFF
     1              *  DBOD(I,45)*DBOD(I,44)/SREF
      SUMCD   = SUMCD + DRAG
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5051) J, I, K, XI, OEFF, DRAG, SUMCD
   51 CONTINUE
C     AMPLIFY DRAG OF WAVES OVER WING BY 2.31
      SUMCD   = SUMCD   + SUMCD + (DBOD(I,42)/DBOD(I,41)) *1.31
  511 DAMCD(I,5) = SUMCD
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5052) J, I, DAMCD(I,5)
      GO TO 58
C
C              S U P E R S O N I C
C
   53 CONTINUE
      DRAG    = (9.87/SQRT(XMACH**2 -1.0)) * (DBOD(I,45)/DBOD(I,43))
     1        * DBOD(I,41) *(DBOD(I,45)*DBOD(I,44)) / SREF
C     AMPLIFY DRAG OF WAVES OVER WING BY 2.31
      DRAG    =  DRAG   + DRAG *(DBOD(I,42)/DBOD(I,41)) * 1.31
      DAMCD(I,5) = DRAG
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5053) J, I, DAMCD(I,K)
   58 CONTINUE
C
C*************             MODE 6                 ***************
C************* DRAG DUE TO PROTUBERANCES ON BODIES ****************
C
   60 IF (IDAM(6).NE.1)  GO TO 70
C
      DO 63      I = 1, NPODS
      SUMCD  .= 0.0
      IF (DBOD(I,51).EQ.0.0)   GO TO 62
      IF (DBOD(I,51).EQ.1.0)   DXOL = 0.0
      IF (DBOD(I,51).GT.1.0)   DXOL = (DBOD(I,55)-DBOD(I,54))
     1                               / (DBOD(I,51) - 1.0)
      DOQ     = DBOD(I,53)
      HEIGHT  = DBOD(I,52)
      N       = DBOD(I,51)
      DO 61   K = 1, N
      XOL     = DBOD(I,54) + DXOL * (K-1)
      XI      =  XOL + BOD(I,1)
C
      CALL EFFO(RNOFT, XI, XMACH, HEIGHT, OEFF)
      DRAG    =   OEFF * DOQ / SREF
      SUMCD   = SUMCD + DRAG
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5061) J, I, K, XI, OEFF, DRAG, SUMCD
   61 CONTINUE
   62 DAMCD(I,6) = SUMCD
   63 CONTINUE
C
C*************             MODE  7                ************************
C************* DRAG DUE TO BLUNTED BODIES ************************
C
   70 IF (IDAM(7).NE.1) GO TO 110
C
      DO 71       I = 1, NPODS
      IF (DBOD(I,61).EQ.0.0)   GO TO 71
      CD1     =   0.0
      CD2     =   0.0
C     CALCULATE DRAG FOR SPHERICAL, SEVERED NOSE
      RMAX    = SQRT(BOD(I,7)    /3.14159)
      RNOSE   =    SQRT(DBOD(I,61) / 3.14159)
      RATIO   =  RNOSE / RMAX
      DRAG1   =  DLNT(RATIO, XMACH, RATIO3, UMACH, DRAG3,8,10,8,2,2)
      CD1     =  DRAG1 *    BOD(I,7) / SREF
C
```

```
C       CALCULATE DRAG FOR FLATTING NOSE OF SPHERE
        DRAG2   = DLNT(DPOD(I,62),XMACH,XLCD,VMACH,DRAG4,6,6,6,2,2)
        CD2     =  DRAG2  *  DPOD(I,61) / SREF
C
        DAMCD(I,7) = CD1 + CD2
        IF (KPRINT(28).EQ.1)
       1WRITE(6,5071) J,I,RMAX, RNOSE, DRAG1, DRAG2,
       1                  CD1, CD2, DAMCD(I,7)
   71 CONTINUE
C**************              MODE 11                    ********************
C************** DRAG DUE TO ROUGHNESS ON SURFACES ********************
C
  110 IF (IDAM(11).NE.1)  GO TO 120
  111 DO 113   I = 1, NSUR
C       CALCULATE DRAG OF UNDAMAGED WING
C       CALL CFK3 (LENGTH, RN/FT, XOL)1, XOL)2, K1, K2, K3, MACH, CF)
        CALL CFK3 (SUR(I,20),RNOFT,0,0, DSUR(I,1),0,0, XMACH, CF)
        CDI     =  CF * SUR(I,7) / SREF
C       CALCULATE DRAG OF DAMAGED SURFACE
        CALL CFK3 (SUR(I,20), RNOFT, DSUR(I,3), DSUR(I,4),
       1           DSUR(I,1),  DSUR(I,2), DSUR(I,1), XMACH, CF)
        CDF     =  CF * SUF(I,7) / SREF
C
        DCD      = (CDF- CDI) * DSUR(I,5) / SUR(I,7)
        IF (DSUR(I,5).LE.1.) DCD = (CDF-CDI) * DSUR(I,5)
C
C       MAGNIFY DRAG IF DAMAGE IS ON UPPER SURFACE
        IF (DSUR(I,6).EQ.0.0)  GO TO 112
        TOC     = SUR(I,3)
        XM      = 1.+3.*TOC
        DCD      = DCD * XM
  112 DAMCD(I,11) = DCD
        IF (KPRINT(28).EQ.1)
       1WRITE(6,5111) J, I, CF, CD1, CDF, DCD, XM, DAMCD(I,11)
  113 CONTINUE
C
C**************              MODE 12                   *************
C************** DRAG DUE TO FWD-FACING STEPS ON SURFACES *************
C
  120 IF (IDAM(12).NE.1) GO TO 130
        CALL LNTP (XMACH, CPF, YMACH3, CPF3, 13, 4)
        DO 125  I = 1, NSUR
        SUMCDL = 0.0
        SUMCDH = 0.0
        CO      =   4.0/ SUR(I,19)* (1. -SUR(I,17))/ (1. + SUR(I,17))
C           CALCULATE DRAG FOR STEPS ON LOW-VELOCITY SURFACES
        IF (DSUR(I,11).EQ.0.0)   GO TO 122
        IF (DSUR(I,11).EQ.1.0)   DXOC = (DSUR(I,16) -DSUR(I,15)) /2.
        IF (DSUR(I,11).GT.1.0)   DXOC = (DSUR(I,16)-DSUR(I,15))
       1                                / (DSUR(I,11)- 1.0)
        N       = DSUR(I,11)
        DO 121  K = 1, N
        XOC     =  DSUR(I,15)  + DXOC * (K-1)
        IF (N.EQ.1)  XOC = DSUR(I,15) + DXOC
        XI      =  XOC  *  SUR(I,20)
        HEIGHT = DSUR(I,14)
        CALL EFFO(RNOFT, XI, XMACH, HEIGHT, CEFF)
        TSWP    =   TAN(SUR(I,4))
        SWPXOC =   ATAN(TSWP -CO * XOC)
        DRAG    = CPF*(COS(SWPXOC))**2 * CEFF *DSUP(I,13)*DSUP(I,14)/SREF
        SUMCDL = SUMCDL + DRAG
        IF (KPRINT(28).EQ.1)
       1WRITE(6,5121) J,I,K,XI,CEFF,SWPXOC,DRAG,SUMCDL
  121 CONTINUE
C
C           CALCULATE DRAG FOR STEPS ON HIGH-VELOCITY SURFACES
  122 IF (DSUR(I,12).EQ.0.0)   GO TO 124
        IF (DSUR(I,12).EQ.1.0)   DXOC = (DSUR(I,16) -DSUR(I,15)) /2.
        IF (DSUR(I,12).GT.1.0)   DXOC =(DSUR(I,16)-DSUR(I,15))
       1                               /(DSUR(I,12)- 1.0)
        N       = DSUR(I,12)
        DO 123  K = 1, N
        XOC     =  DSUR(I,15) + DXOC * (K-1)
        IF (N.EQ.1)  XOC = DSUR(I,15) + DXOC
        XI      =  XOC  * SUR(I,20)
        HEIGHT = DSUR(I,14)
        CALL EFFO(RNOFT, XI, XMACH, HEIGHT, CEFF)
```

```
      TSWP    =  TAN(SUR(I,4))
      SWPXOC  =  ATAN(TSWP -CC * XOC)
      CLD     =  SUR(I,2)
      XM      =  DLNT(XOC, CLD, XOC3, CLD3, XM3, 6, 4, 6, 2, 2)
      DRAG = CPF*(COS(SWPXOC))**3 * QEFF * XM *
     1                   DSUR(I,13)* DSUR(I,14) / SREF
      SUMCDH =   SUMCDH  + DRAG
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5122) J,I,K,XI,CEFF,SWPXOC,XM,DRAG,SUMCDH
  123 CONTINUE
  124 DAMCD(I,12) = SUMCDL + SUMCDH
  125 CONTINUE
C
C*************               MODE 13                        *************
C************* DRAG DUE TO AFT-FACING STEPS ON SURFACES *************
C
  130 IF (IDAM(13).NE.1) GO TO 140
      CALL LNTP (XMACH, CPA, ZMACH3, CPA3, 13, 4)
      DO 135 I = 1, NSUR
      SUMCDL = 0.0
      SUMCDH = 0.0
      CO      = 4.0/SUR(I,19) * (1. -SUR(I,17))/ (1. + SUR(I,17))
C
C          CALCULATE DRAG DUE TO STEPS ON LOW-VELOCITY SURFACES
      IF (DSUR(I,21).EQ.0.0)  GO TO 132
      IF (DSUR(I,21).EQ.1.0)  DXOC = (DSUR(I,26) -DSUR(I,25)) /2.
      IF (DSUR(I,21).GT.1.0)  DXOC = (DSUR(I,26)-DSUR(I,25))
     1                             / (DSUR(I,21)- 1.0)
      N       = DSUR(I,21)
      DO 131  K = 1, N
      XOC     = DSUR(I,25)  + DXOC * (K-1)
      IF (N.EQ.1)  XOC = DSUR(I,25) + DXOC
      XI      =  XOC  * SUR(I,20)
      HEIGHT = DSUR(I,24)
      CALL EFFO(RNCFT, XI, XMACH, HEIGHT, CEFF)
      TSWP    = TAN(SUR(I,4))
      SWPXOC = ATAN(TSWP - CC * XOC)
      DRAG    = -CPA*(COS(SWPXOC))**2 *QEFF*DSUR(I,23)*DSUR(I,24)/SREF
      SUMCDL = SUMCDL + DRAG
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5131) J,I,K,XI,CEFF,SWPXOC,DRAG,SUMCDL
  131 CONTINUE
C
C          CALCULATE DRAG FOR STEPS ON HIGH-VELOCITY SURFACES
  132 IF (DSUR(I,22).EQ.0.0)  GO TO 134
      IF (DSUR(I,22).EQ.1.0)  DXOC = (DSUR(I,26) -DSUR(I,25)) /2.
      IF (DSUR(I,22).GT.1.0)  DXOC  =(DSUR(I,26)-DSUR(I,25))
     1                             /(DSUR(I,22)- 1.0)
      N       = DSUR(I,22)
      DO 133   K = 1, N
      XOC     = DSUR(I,25) + DXOC * (K-1)
      IF (N.EQ.1)  XOC = DSUR(I,25) + DXOC
      XI      = XOC   * SUR(I,20)
      HEIGHT = DSUR(I,24)
      CALL EFFO(RNCFT, XI, XMACH, HEIGHT, CEFF)
      TSWP    = TAN(SUR(I,4))
      SWPXOC  = ATAN(TSWP - CC * XOC)
      CLD     = SUR(I,2)
      XM      = DLNT(XOC, CLD, XOC3, CLD3, XM3, 6, 4, 6, 2, 2)
      DRAG    = -CPA * (COS(SWPXOC))**2 * CEFF * XM *
     1                   DSUR(I,23) * DSUR(I,14) / SREF
      SUMCDH = SUMCDH + DRAG
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5132) J,I,K,XI,QEFF,SWPXOC,XM,DRAG,SUMCDH
  133 CONTINUE
  134 DAMCD(I,13) = SUMCDL + SUMCDH
  135 CONTINUE
C
C*************               MODE 14                 ***********************
C************* DRAG DUE TO HOLES IN SURFACES ***********************
C
  140 IF (IDAM(14).NE.1)  GO TO 150
      DO 149   I = 1, NSUR
C
C              LOW VELOCITY SURFACES
      CDOL    = 0.0
      IF (DSUR(I,31).EQ.0.0) GO TO 142
```

```
      IF (DSUR(I,31).EQ.1.0)  DXOC = (DSUR(I,37) -DSUR(I,36)) /2.
      IF (DSUR(I,31).GT.1.0)  DXOC =(DSUR(I,37)-DSUR(I,36))
     1                            /(DSUR(I,31)-1.0)
C     DO 141 SUMS CDO FOR ALL HOLES ON LOW VELOCITY SURFACE
      N       = DSUR(I,31)
      DO 141    K = 1, N
C
      XOC     =  DSUR(I,36) + (K-1) * DXOC
      X       = XOC   * SUR(I,20)
      HOX     = DSUR(I,35) / X
      XLOH    = DSUR(I,33) / DSUR(I,35)
C
      IF (DSUR(I,38).EQ.2.0)  GO TO 145
C
C     CALCULATE DCDO FOR MISSING PANELS
      IF (HOX.GT.0.1)  HOX = 0.1
      IF (XLOH.GT.10.0)  XLOH = 10.0
      DCDO    = DLNT(XLOH, HOX, XLOH3, HOX3, CDO3, 6, 5, 6, 2, 2)
      GO TO 146
C
C     CALCULATE DCDO FOR CAVED-IN PANELS
  145 IF (HOX.GT.0.015) HOX = 0.015
      IF (XLOH.GT.120.0) XLOH = 120.0
      DCDO    = DLNT (XLOH, HOX, XLOH4, HOX4, CDO4, 12,7,12,2,2)
C
  146 CDOL    = CDOL + DCDO
C
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5141) J, I, K, X, HOX, XLOH, DCDO, CDOL
  141 CONTINUE
C             HIGH VELOCITY SURFACES
  142 CDOH    = 0.0
      IF (DSUR(I,32).EQ.0.0)  GO TO 144
      IF (DSUR(I,32).EQ.1.0)  DXOC = (DSUR(I,37) -DSUR(I,36)) /2.
      IF (DSUR(I,32).GT.1.) DXOC=(DSUR(I,37)-DSUR(I,36))
     1                        / (DSUR(I,32) - 1.0)
C     DO 143 SUMS CDO FOR ALL HOLES ON HIGH VELOCITY SURFACE
      N       = DSUR(I,32)
      DO 143    K = 1, N
      XOC     =  DSUR(I,36) + (K-1) * DXOC
      X       = XOC   * SUR(I,20)
      HOX     = DSUR(I,35)  / X
      XLOH    = DSUR(I,33)  / DSUR(I,35)
C
      IF (DSUR(I,38).EQ.2.0)  GO TO 147
C
C     CALCULATE DCDO FOR MISSING PANELS
      IF (HOX.GT.0.1)  HOX = 0.1
      IF (XLOH.GT.10.0)  XLOH = 10.0
      DCDO    = DLNT(XLOH, HOX, XLOH3, HOX3, CDO3, 6, 5, 6, 2, 2)
      GO TO 148
C
C     CALCULATE DCDO FOR CAVED-IN PANELS
  147 IF (HOX.GT.0.015)     HOX  = 0.015
      IF (XLOH.GT.120.0)    XLOH  = 120.0
      DCDO    = DLNT(XLOH, HOX, XLOH4, HOX4, CDO4, 12, 7, 12,2,2)
C
  148 CLD     = SUR(I,2)
      XM      = DLNT(XOC , CLD, XOC3 , CLD3, XM3 , 6, 4, 6, 2, 2)
      CDOH    = CDOH + DCDO * XM
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5142) J, I, K, X, HOX, XLOH, DCDO, CDOH
  143 CONTINUE
C
C     CALCULATE DRAG FROM CDO FOR LOW AND HIGH VELOCITY SURFACES
C
 144  CONTINUE
      CALL LNTP (XMACH, COMK, XMACH3, COMK3, 17, 4)
C
      SUMCDO  = CDOL  + CDOH
      DRAG    = SUMCDO * COMK * DSUR(I,34)*DSUR(I,35) / SREF
      ABC     =  DSUR(I,31)    +  DSUR(I,32)
      IF (ABC.GT.0.1.AND.KPRINT(28).EQ.1) WRITE(6,5143) COMK,SUMCDO,DRAG
      CAMCD(I,14) = DRAG
  149 CONTINUE
C
C***** *******          MODE 15            ************************
```

```
C************* DRAG DUE TO WAVES ON SURFACES *************************
C
  150 IF (IDAM(15).NE.1)   GO TO 160
      DO 159  I = 1,NSUR
C
      IF (XMACH.GE.1.0)  GO TO 1551
C
C                S U B S O N I C
C
C     CALCULATE DRAG FOR LOW-VELOCITY SURFACES
      SUMCDL =   0.0
      IF (DSUR(I,41).EQ.0.0)    GO  TO 152
      IF (DSUR(I,41).EQ.1.0) DXOCL= (DSUR(I,47)-DSUR(I,46))/2.
      IF (DSUR(I,41).GT.1.0) DXOCL= (DSUR(I,47)-DSUR(I,46))
     1                              / (DSUR(I,41)-1.0)
      N        = DSUR(I,41)
      DO 151     K = 1, N
      XOCL     =  DSUR(I,46)   +  DXOCL* (K-1)
      XI       =  XOCL  * SUR(I,20)
      HEIGHT = DSUR(I,45)
      CALL EFFO(RNOFT, XI, XMACH, HEIGHT, OEFF)
      DRAGL   = 15.0 *(DSUR(I,45) / DSUR(I,43))**2 * OEFF
     1              *   DSUR(I,45) * DSUR(I,44) / SREF
      SUMCDL = SUMCDL+ DRAGL
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5151)  J, I, K, OEFF, DRAGL, SUMCDL
  151 CONTINUE
C
C     CALCULATE DRAG FOR HIGH-VELOCITY SURFACES
  152 SUMCDH =    0.0
      IF (DSUR(I,42).EQ.0.0)  GO TO 154
      IF (DSUR(I,42).EQ.1.0) DXOCH= (DSUR(I,47)-DSUR(I,46))/2.
      IF (DSUR(I,42).GT.1.0) DXOCH=(DSUR(I,47)-DSUR(I,46))
     1                             / (DSUR(I,42) - 1.0)
      N        =  DSUR(I,42)
      DO 153     K = 1, N
      XOCH     -  DSUR(I,46) +  DXOCH *  (K-1)
      XI       =  XOCH * SUR(I,20)
      HEIGHT = DSUR(I,45)
      CALL EFFO(RNOFT, XI, XMACH, HEIGHT, OEFF)
      XM   =   DLNT(XOCH, SUR(I,2), XOC3, CLD3, XM3, 6,4,6,2,2)
      DRAGH    = 15.0 * (DSUR(I,45) / DSUR(I,43))**2 * OEFF
     1           * XM * DSUR(I,45) * DSUR(I,44) / SREF
      SUMCDH = SUMCDH + DRAGH
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5152)  J, I, K, OEFF, XM, DRAGH, SUMCDH
  153 CONTINUE
  154 DAMCD(I,15) = SUMCDL + SUMCDH
      GO TO 159
C
C                S U P E R S O N I C
C
C     CALCULATE DRAG FOR LOW-VELOCITY SURFACES
 1551 SUMCDL = 0.0
      IF (DSUR(I,41).EQ.0.0) GO TO 1598
      SUMCDL = 9.87/ SQRT(XMACH**2 -1.0) *(DSUR(I,45) /DSUR(I,43))
     1           * DSUR(I,41) * DSUR(I,45) * DSUR(I,44) /SREF
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5153)  J, I, SUMCDL
C
C     CALCULATE DRAG FOR HIGH-VELOCITY SURFACES
 1598 SUMCDH  = 0.0
      IF (DSUR(I,42).EQ.0.0)  GO TO 157
      IF (DSUR(I,42).EQ.1.0) DXOCH= (DSUR(I,47)-DSUR(I,46))/2.
      IF (DSUR(I,42).GT.1.0)  DXOCH  = (DSUR(I,47)- DSUR(I,46))
     1                                 / (DSUR(I,42) -1.0)
      N        = DSUR(I,41)
      DO 156     K = 1, N
C
      XOC      =  DSUR(I,46)   + DXOCH *(K -1)
      CLD      =  SUR(I,2)
      XM       =  DLNT(XOC, CLD, XOC3, CLD3, XM3, 6,4,6,2,2)
      DRAG     = 9.87 /SQRT(XMACH**2-1.0) *(DSUR(I,45) /DSUR(I,43))
     1          *DSUR(I,45) * DSUR(I,44) /SREF    * XM
      SUMCDH = SUMCDH + DRAG
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5154) J, I, K, XM, DRAG, SUMCDH
  156 CONTINUE
```

```
  157 DAMCD(I,15) = SUMCDL + SUMCDH
  159 CONTINUE
C
C*************               MODE 16                    ***************
C************* DRAG DUE TO PROTUBERANCES ON SURFACES ***************
C
  160 IF (IDAM(16).NE.1)  GO TO 170
C
C
      DO 163  I = 1, NSUR
      SUMCD  = 0.0
      IF (DSUR(I,51).EQ.0.0)   GO TO 162
      IF (DSUR(I,51).EQ.1.0)  DXCC = 0.0
      IF (DSUR(I,51).GT.1.0)  DXCC = (DSUR(I,55)-DSUR(I,54))
     1                             / (DSUR(I,51) - 1.0)
      DOC     = DSUR(I,53)
      N       =  DSUR(I,51)
      DO 161    K = 1, N
      XOC     = DSUR(I,54) + DXCC * (K-1)
      HEIGHT = DSUR(I,52)
      XI      =  XOC * SUR(I,20)
      HEIGHT = 3.0 / 12.0
      CALL  EFFC(RNCFT, XI, XMACH, HEIGHT, CEFF)
      DRAG    =   CEFF * DOC / SREF
      SUMCD   =   SUMCD + DRAG
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5161)  J, I, K, XI, CEFF, DRAG, SUMCD
  161 CONTINUE
  162 DAMCD(I,16) = SUMCD
  163 CONTINUE
C
C*************            MODE 17               *********************
C************* DRAG DUE TO MISSING WINGS PARTS *********************
C
C     CALCULATE MINIMUM DRAG DUE TO MISSING LEADING EDGE
C
  170 IF (IDAM(17).NE.1) GO TO 210
      DRAGLE = 0.0
      DRAGTE = 0.0
      DRGTIP = 0.0
C
      IF (DWING(5).LE.1.0)   SHARPF = 0.010
      IF (DWING(5).GT.1.0)   SHARPF = 0.015
      DELB   = DWING(3) *   RO2
      DRAGLE = SHARPF * DWING(4) * DELB * CBAR   / SREF
      IF (DWING(6).EQ.3.0)  DRAGLE = 2. * DRAGLE
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5171) XMACH,SHARPF,DELB,CBAR,DRAGLE
C
C
C     CALCULATE MINIMUM DRAG DUE TO MISSING TRAILING EDGE
C
 171  CONTINUE
C
C     CHARL = LENGTH OF CHORD WITH MISSING T.E. PANEL
 172  CHARL = (1.-DWING(7)) * CBAR
      IF (IDAM(11).EQ.1)    RCHK = DSUR(1,2)
      IF (DSUR(1,2).EQ.0.0)  RCHK = RCUGHK
      IF (IDAM(11).NE.1)    RCHK  = RCUGHK
C     CALL CFK3 (LENGTH, RN/FT, XCL)1, XCL)2, K1, K2, K3, MACH, CF)
      CALL CFK3 (CHARL,RNCFT,0.,0.,RCHK,0.,0.,XMACH, CF)
      CALL CFK3 (CBAR, RNCFT,0.,0.,RCHK,0.,0.,XMACH,CF1)
      DELB    = DWING(8) * RO2
C
      DRAGTE = (0.2/CF**0.33)*DWING(7)**1.33 + TOCW**1.33
     1         *DELB *CBAR/SREF  +(CF1*CHARL -CF *CBAR)*DELB/SREF
      IF (DWING(10).EQ.3.0) DRAGTE = 2. * DRAGTE
C
C     DRAGTE = BASE DRAG + CHANGE IN FRICTION DRAG
C
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5172) RNCFT, CHARL, RCHK, DELB, CF, CF1, DRAGTE
C
C     CALCULATE MINIMUM DRAG DUE TO MISSING TIP
C
C     CHARACTERISTIC LENGTH OF MISSING WING TIP
C     CBRK    = AVERAGE CHORD AT POINT WHERE BREAK OCCURS
```

```
      DELETA = 0.5 * (DWING(11)+DWING(12))
      CBRK   = CTX +(CR-CTX) * DELETA
      CHORD  = 0.6667 * (CBRK + CTX -CBRK * CTX /(CBRK+CTX))
      AWET   = (CBRK + CTX) *(DWING(11)+DWING(12)) * PC2
C
      CALL CFK3 (CHORD,FMCFT,0,0,RGHK,C,C,XM/CH,CF)
      DRGTIP = - CF * AWET / SREF
C
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5173) CBRK, CHORD, AWET, CF, DRGTIP
C
      DAMCD(1,17) = DRAGLE + DRAGTE + DRGTIP
      IF (KPRINT(28).EQ.1)
     1WRITE(6,5174) DRAGLE, DRAGTE, DRGTIP, DAMCD(1,17)
C
C     MINIMUM DRAG DUE TO MISSING SURFACE PARTS
C
C
C     SUM COMPONENTS OF MINIMUM DRAG FOR EACH SURFACE
      DO 175  I= 1, NSUR
      DRGSUR(I)    = 0.0
      DO 174  N = 1, 4
      DRGSUR(I)    = DRGSUR(I) + CDSUR(I,N)
  174 CONTINUE
  175 CONTINUE
C
C    CALCULATE DRAG CHANGE DUE TO MISSING SURFACE PARTS
C
      IF (NHT.EQ.0)   IHT =  1
      IF (NHT.GT.0)   IHT =  2
      M              =  NPNLS + IHT
      IF (NHT.GT.0) DAMCD(2,17)= -DRGSUR(2)*(DWING(14)+DWING(15))/2.
      DO 176    I  =  M, NSUR
      DAMCD(I,17) = -DRGSUR(I) * DWING(16+I-M)
  176 CONTINUE
  180 IF (KPRINT(28).EQ.1)
     1WRITE(6,5175) (I,DRGSUR(I),  DAMCD(I,17),     I = 1,NSUR)
C
  210 CONTINUE
C
C            INCREASE DRAG IF DAMAGE IS SYMMETRICAL
C
      DO 230    K =   1, 7
      DO 220    I =   1, 7
      IF (SYM(I,K).EQ.1.0) DAMCD(I,K)= BCD(I,6) * DAMCD(I,K)
      IF (SYM(I,K+10).EQ.1.0) DAMCD(I,K+10) =  2.0   * DAMCD(I,K+10)
  220 CONTINUE
  230 CONTINUE
C
      CALL WRITE(J)
      CALL NUCDAM2(J)
      CALL NTRIM(J)
      CALL WRITE2(J)
C
 500  CONTINUE
C
C
C     FORMAT STATEMENTS FOR CHECKOUT
C
 5000 FORMAT (1H1, /, 10X, *DUMP FROM SUBROUTINE NUCDAM,     MACH =*,
     1             F10.3, // )
 5011 FORMAT(10X,*MODE 1, J, I, CDI, CDF, CF(DAMAGED), DAMCD(I,J,1)=*
     1             ,2I3, 4F10.5)
 5021 FORMAT (5X,*MODE 2, J, I, K, XI, QEFF, CPF, SUMQ = *,
     1             3I3, 4F10.3)
 5022 FORMAT (20X, * DAMCD(I,J,2) = * , F10.5)
 5031 FORMAT (10X,*MODE 3, J, I, K, XI, QEFF, CPA, SUMQ = *,
     1             3I3, 4F10.3)
 5032 FORMAT(20X, * DAMCD(I,J,3) = *,F10.5)
 5051 FORMAT (10X,*MODE 5, PROTRUDED, SUPERSONIC, J, I, K, YI, QEFF,*,
     1            * DRAG, SUMCD =*, 3I3, 2F8.3, 2F10.5)
 5053 FORMAT(10X, *MODE 5, PROTRUDED, SUPERSONIC, J, I, *,
     1            *DAMCD(I,J,K) = *,  2I3, F10.5)
 5052 FORMAT(20X, * J, I, DAMCD(I,J,5) = *, 2I3, F10.5)
 5054 FORMAT(10X,*MODE 5, INDENTED, J, I, K, Y, CDD, FRAC, *,
     1           * SUMCD, CDMK =*,3I3,F8.2,F8.4,2F11.6,F8.3)
 5055 FORMAT(20X, * J, I, DAMCD(I,J,5) = *, 2I3, F10.5)
```

```
5041 FORMAT(10X, *MODE 4, J, I, K, X, DCDD, CDD = *,
    1                  3I3, 3F10.3)
5042 FORMAT(20X, * CDMK, DAMCD(I,J,4) = *,  F10.3,F10.5)
5061 FORMAT(5X,*MODE 6, J, I, K, YI, CEFF, DRAG, SUMCD = *,
    1                 3I3, 2F10.3, 2F10.5)
5071 FORMAT (10X,*MODE 7, J, I, RMAX, RNOSE, DRAG1, DRAG2, *,
    1                  *CD1, CD2, DAMCD(I,J,7)=*, 2I3, 4F8.3, 3F8.5)
5111 FORMAT (10X, *MODE 11, J, I, CF(DAMAGED), CD1, CDF, DCD, *,
    1                  *XM, DAMCD(I,J,11)=*, 2I3,4F8.5, F6.3, F9.5)
5121 FORMAT(4X, * MODE 12, LOW VELOCITY, J, I, K, XI, CEFF,*,
    1           *  SWPXDC,        DRAG, SUMCDL=*,3I3,3F7.3,7X,2F8.5)
5122 FORMAT(4X, * MODE 12, HIGH VEL.    , J, I, K, XI, CEFF,*,
    1           *  SWPXDC, XM, DRAG, SUMCDH=*,3I3,4F7.3, 2F8.5)
5131 FORMAT(4X, * MODE 13, LOW VELOCITY, J, I, K, XI, CEFF,*,
    1           *  SWPXDC,        DRAG, SUMCDL=*,3I3,3F7.3,7X,2F8.5)
5132 FORMAT(4X, * MODE 13, HICH VEL.    , J, I, K, XI, CEFF,*,
    1           *  SWPXDC, XM, DRAG, SUMCDH=*,3I3,4F7.3, 2F8.5)
5141 FORMAT(10X,*MODE 14, LOW  VEL., J, I, K, X, HDX, YLDH, DCDD,*,
    1              * CDDL =*, 3I3, 5F7.3)
5142 FORMAT(10X,*MODE 14, HIGH VEL., J, I, K, X, HDX, XLDH, DCDD,*,
    1              * CDDH =*, 3I3, 5F7.3)
5143 FORMAT(20X, *CDMK, SUMCDD, DRAG = * ,  F10.2, F10.3, F10.5)
5151 FORMAT(4X, *MODE 15, PROTRUDED, LOW  VEL., J, I, K, CEFF, *,
    1      *        DRAGL, SUMCDL =*, 3I3, F8.5, 8X, 2F8.5)
5152 FORMAT(4X, *MODE 15, PROTRUDED, HIGH VEL., J, I, K, CEFF, *,
    1             *XM, DRAGH, SUMCDH =*, 3I3, 4F9.5)
5153 FORMAT(4X, *MODE 15, PROTRUDED, LOW  VEL., SUPERSONIC, *,
    1             *J, I, SUMCDL =*,   2I3, F10.5)
5154 FORMAT(4X, *MODE 15, PROTRUDED, HIGH VEL., SUPERSONIC, *,
    1             *J, I, K, XM, DRAG, SUMCDH=*, 3I3, 3F10.5)
5155 FORMAT(4X, *MODE 15, INDENTED , LOW  VEL., J, I, K, XL,  *,
    1             *CDDL, DRAG, SUMCDL =*, 3I3, 2F8.3, 2F8.5)
5156 FORMAT(4X, *MODE 15, INDENTED , HIGH VEL., J, I, K, XH, *,
    1             *CDDH, DRAG, SUMCDH = *, 3I3, 2F10.4,2F14.8)
5161 FORMAT(10X,*MODE 16, J, I, K, XI, CEFF, DRAG, SUMCD  = *,
    1                  3I3,2F9.3, 2F10.5)
5171 FORMAT(///,10X,*MODE 17,   XMACH = *,F10.5,
    1           /,10X,*SHARPE, DELP, CBAR, DRAGLE =*, 4F10.5)
5172 FORMAT( / ,10X, *DNCET, CHARL, RGHK, DELP, CF, CF1,*,
    1           *DRAGTE = *,F11.0, 6F10.5)
5173 FORMAT ( /  , 10X, *CPFK, CHORD, ANET, CF, DRGTIP = *,5F10.4)
5174 FORMAT ( /  , 10X, *DRAGLE, DRAGTE, DRGTIP, DAMCD(1,J,17)=*,
    1              4F11.5)
5175 FORMAT( /  , 10X,*DRAG DUE TO MISSING SURFACE PARTS*,
    1           /,14X,*I      TOTAL DRAG OF EACH SURFACE    DAMCD(I,J,17)*
    2         ,/, ( 13X, I2, 12X, F10.5, 12X, F9.5))
     END
```

```
      SUBROUTINE WRITE(J)
C
C*************              ALL MODES                            ***************
C*************    W R I T E    S T A T E M E N T S               ***************
C
      COMMON /INPUT/  A(3081)
      COMMON /BLKTTL/  TITLE(6)
      COMMON /NUCOUT/  DAMCD(7,17), SYM(7,17), CDOMIN,
     1         ALP(21), CL(21),     CD(21),    CM(21),    CLT(21),   CDT(21),
     2                  CLP(21),    CDP(21),   CMP(21),   CLTP(21),  CDTP(21),
     3                  RCLA(6), DXAC,
     4                  CPOLL(21), CDRT(21), X3(2), X4(2)
      DIMENSION CDBOD(10), CDSUR(10), CDMODE(20),
     1          FMSURV(20), ALT(20), CLLC(20), CLHI(20), IDAM(20),
     2          BNAME(7), SNAME(7)
      EQUIVALENCE     (A(1), NBODYS),(A(2),NNAC),(A(3),NPNLS),
     1                (A(4), NHT)   ,(A(5),NVT)  ,(A(491),NSURV),
     2                (A(492),NCLAS),(A(493),FMSURV(1)),
     3                (A(513),ALT(1)),(A(573),CLLC(1)),
     4                (A(593),CLHI(1)), (A(1642), IDAM(1)),
     5                (A(30),BNAME(1)), (A(177),SNAME(1))
C
C
      NBODS   =   NBODYS + NNAC
      NSUR    =   NPNLS  + NHT   + NVT
      JJJ     =  IDAM(1) + IDAM(2) + IDAM(3) + IDAM(4) + IDAM(5) +
     1            IDAM(6) + IDAM(7) + IDAM(8) + IDAM(9) + IDAM(10)
      WRITE(6,1000)  ( TITLE(I) , I = 1,6)
      WRITE(6,1001) FMSURV(J)
      IF (JJJ.EQ.0)   GO TO 40
      WRITE(6,1002)   (BNAME(I) ,    I = 1,NBODS)
      IF  (IDAM( 1).EQ.1)  WRITE(6,1003)   (DAMCD(I,1)     , I=1,NBODS)
      IF  (IDAM( 2).EQ.1)  WRITE(6,1004)   (DAMCD(I,2)     , I=1,NBODS)
      IF  (IDAM( 3).EQ.1)  WRITE(6,1005)   (DAMCD(I,3)     , I=1,NBODS)
      IF  (IDAM( 4).EQ.1)  WRITE(6,1006)   (DAMCD(I,4)     , I=1,NBODS)
      IF  (IDAM( 5).EQ.1)  WRITE(6,1007)   (DAMCD(I,5)     , I=1,NBODS)
      IF  (IDAM( 6).EQ.1)  WRITE(6,1008)   (DAMCD(I,6)     , I=1,NBODS)
      IF  (IDAM( 7).EQ.1)  WRITE(6,1009)   (DAMCD(I,7)     , I=1,NBODS)
C
C     SUM DRAG ON EACH BODY
      DO 10    I = 1, 10
      COBOD(I)    = 0.0
   10 CONTINUE
C
      DO 30     I = 1, NBODS
      SUM      = 0.0
      DO 20     K = 1, 10
      SUM       = SUM + DAMCD(I,K)
   20 CONTINUE
      CDBOD(I)  = SUM
   30 CONTINUE
      WRITE(6,1010)   (CDBOD(I) ,     I = 1,NBODS)
   40 JJJ      =IDAM(11)+IDAM(12)+IDAM(13)+IDAM(14)+IDAM(15)+
     1           IDAM(16)+IDAM(17)+IDAM(18)+IDAM(19)+IDAM(20)
      IF (JJJ.EQ.0)   GO TO 80
      WRITE(6,1011)   (SNAME(I) ,     I = 1,NSUR  )
      IF  (IDAM(11).EQ.1) WRITE(6,1012) (DAMCD(I,11)    , I=1,NSUR)
      IF  (IDAM(12).EQ.1) WRITE(6,1013) (DAMCD(I,12)    , I=1,NSUR)
      IF  (IDAM(13).EQ.1) WRITE(6,1014) (DAMCD(I,13)    , I=1,NSUR)
      IF  (IDAM(14).EQ.1) WRITE(6,1015) (DAMCD(I,14)    , I=1,NSUR)
      IF  (IDAM(15).EQ.1) WRITE(6,1016) (DAMCD(I,15)    , I=1,NSUR)
      IF  (IDAM(16).EQ.1) WRITE(6,1017) (DAMCD(I,16)    , I=1,NSUR)
      IF  (IDAM(17).EQ.1.AND.NPNLS.EQ.1) WRITE(6,1018)
     1           (DAMCD(I,17), I=1,NSUR)
      IF  (IDAM(17).EQ.1.AND.NPNLS.EQ.2)  WRITE(6,2001) DAMCD(1,17),
     1           (DAMCD(I,17), I=3,NSUR)
C     SUM DRAG ON EACH SURFACE
      DO 50  I = 1, 10
      CDSUR(I) = 0.0
   50 CONTINUE
C
      DO 70    I = 1, NSUR
      SUM      = 0.0
      DO 60    K =11,17
      SUM     = SUM + DAMCD(I,K)
   60 CONTINUE
      CDSUR(I) = SUM
```

```
   70 CONTINUE
      WRITE(6,1019)  (CDSUR(I) ,    I = 1,NSUR)
C
C     SUM DRAG ACCORDING TO MODE OF DAMAGE
C
C        B O D I E S
   80 DO 100  K = 1,7
      SUM     = 0.0
      DO  90  I = 1, NBODS
      SUM     = SUM + DAMCD(I,K)
   90 CONTINUE
      CDMODE(K) = SUM
  100 CONTINUE
C
C        S U R F A C E S
      DO 120  K = 11, 17
      SUM     = 0.0
      DO 110  I = 1 , NSUR
      SUM     = SUM + DAMCD(I,K)
  110 CONTINUE
      CDMODE(K) = SUM
  120 CONTINUE
C
      WRITE(6,1020)
      CD1     = CDMODE(1)  + CDMODE(11)
      CD2     = CDMODE(2)  + CDMODE(12)
      CD3     = CDMODE(3)  + CDMODE(13)
      CD4     = CDMODE(4)  + CDMODE(14)
      CD5     = CDMODE(5)  + CDMODE(15)
      CD6     = CDMODE(6)  + CDMODE(16)
      CD7     = CDMODE(7)
      CD8     = CDMODE(17)
      WRITE(6,1021) CDMODE(1) , CDMODE(11), CD1
      WRITE(6,1022) CDMODE(2) , CDMODE(12), CD2
      WRITE(6,1023) CDMODE(3) , CDMODE(13), CD3
      WRITE(6,1024) CDMODE(4) , CDMODE(14), CD4
      WRITE(6,1025) CDMODE(5) , CDMODE(15), CD5
      WRITE(6,1026) CDMODE(6) , CDMODE(16), CD6
      WRITE(6,1027) CDMODE(7) ,             CD7
      WRITE(6,1028)             CDMODE(17), CD8
C
C     SUM TOTAL DAMAGE DRAG ON BODIES AND WINGS
C
      AAA     = 0.0
      BBB     = 0.0
      DO 130  I = 1,7
      AAA     = AAA + CDMODE(I)
      BBB     = BBB + CDMODE(I+10)
  130 CONTINUE
      CCC     = AAA + BBB
      WRITE(6,1029) AAA, BBB, CCC
      CDOMIN  = CCC
 1000 FORMAT(1H1,//, 21X, 6A10)
 1001 FORMAT(  //,21X,*SUMMARY OF DAMAGE EFFECTS ON MINIMUM DRAG*,
     1          //,32X, *M A C H   = *, F5.2, ///)
 1002 FORMAT (/,   5X, *BODY DAMAGE MODE         *, 4X, 7A10)
 1003 FORMAT (/,   5X, *ROUGHNESS                *, 2X, 7F10.5)
 1004 FORMAT (     5X, *FWD-FACING STEPS         *, 2X, 7F10.5)
 1005 FORMAT (     5X, *AFT-FACING STEPS         *, 2X, 7F10.5)
 1006 FORMAT (     5X, *HOLES                    *, 2X, 7F10.5)
 1007 FORMAT (     5X, *WAVINESS                 *, 2X, 7F10.5)
 1008 FORMAT (     5X, *PROTUBERANCES            *, 2X, 7F10.5)
 1009 FORMAT (     5X, *BLUNTNESS                *, 2X, 7F10.5)
 1010 FORMAT (/,   5X, *TOT.DRAG,EA.BODY =       *, 2X, 7F10.5)
 1011 FORMAT (///,5X, *SURFACE DAMAGE MODE       *, 4X, 7A10)
 1012 FORMAT (/,   5X, *ROUGHNESS                *, 2X, 7F10.5)
 1013 FORMAT (     5X, *FWD-FACING STEPS         *, 2X, 7F10.5)
 1014 FORMAT (     5X, *AFT-FACING STEPS         *, 2X, 7F10.5)
 1015 FORMAT (     5X, *HOLES                    *, 2X, 7F10.5)
 1016 FORMAT (     5X, *WAVINESS                 *, 2X, 7F10.5)
 1017 FORMAT (     5X, *PROTUBERANCES            *, 2X, 7F10.5)
 1018 FORMAT (     5X, *MISSING PARTS            *, 2X, 7F10.5)
 2001 FORMAT (     5X, *MISSING PARTS            *, 2X,F10.5,*----------*,
     1                5F10.5)
 1019 FORMAT (/,   5X, *TOT.DRAG,EA.SURFACE =*, 2X, 7F10.5)
 1020 FORMAT (//////,17X,*NET DRAG FROM EACH MODE OF DAMAGE*,
     1               /,29X,*BODIES    SURFACES      BODIES + SURFACES*)
```

```
1021 FORMAT (     5X, *ROUGHNESS            *, 2F10.5,F16.5)
1022 FORMAT (     5X, *FWD-FACING STEPS     *, 2F10.5,F16.5)
1023 FORMAT (     5X, *AFT-FACING STEPS     *, 2F10.5,F16.5)
1024 FORMAT (     5X, *HOLES                *, 2F10.5,F16.5)
1025 FORMAT (     5X, *WAVINESS             *, 2F10.5,F16.5)
1026 FORMAT (     5X, *PROTUBERANCES        *, 2F10.5,F16.5)
1027 FORMAT (     5X, *BLUNTNESS*,12X,F10.5,6X,*N/A  *,F15.5)
1028 FORMAT (     5X, *MISSING PARTS        *,6X,*N/A *,F10.5,F16.5)
1029 FORMAT (/,   5X, *TOTAL DAMAGE DRAG    *, 2F10.5,F16.5)
     RETURN
     END
```

```
      SUBROUTINE CFK3(XL,RNOFT,XOL1,XOL2,RK1,RK2,R3K,ZMACH, CFR)
C
C     CALCULATES SKIN FRICTION COEFF. FOR A PLATE WITH UP TO
C     THREE ZONES OF ROUGHNESS
C
      CF(X,Y) = AMAX1(T*F**2*0.430/(ALOG10(RNL*X*T**1.67*F))**2.56,
     1          T/(1.89 +1.62*ALOG10(X/Y))**2.5   )
C
      SCF(X)  = T*F**2*0.430/(ALOG10(RNL*X*T**1.67*F))**2.56
C
      SCFP(X) = T*F**2*0.430/(ALOG10(RNL*X*T**1.67*F))**2.56
     1  -T*F*F*0.47807/(ALOG10(RNL*X*T**1.67*F))**3.56
C
      RCFP(X,Y) = T/(1.89 +1.62*ALOG10(X/Y))**2.5
     1            -1.7589 *T/(1.89 +1.62*ALOG10(X/Y))**3.5
C
      T      = 1./(1.+0.178*ZMACH**2)
      F      = 1.+0.03916*ZMACH**2*T
C
      RNL    = RNOFT
      RK3    = R3K
C
      CFR    = SCF(XL)
      IF(RK1.GT.0.0) CFR = CF(XL,RK1)
      IF (XOL1.EQ.0.0.AND.XOL2.EQ.0.0) RETURN
C
      IF (XOL1.GT.0.0) GO TO 100
C     INCREASED ROUGHNESS STARTS AT LEADING EDGE
      X1     = XL * XOL2
      CFR    = SCF(X1)
      IF (RK2.GT.0.0) CFR = CF(X1,RK2)
      IF (XOL2.GE.1.0) RETURN
C
      J      = 2
      CFY1   = CFR
      DX     = X1
      RK     = RK3
      GO TO 10
C
  100 CONTINUE
C
      X1     = XL * XOL1
      RKA    = RK1
      RK     = RK2
      J      = 1
C
    5 CFX1   = SCF(X1)
      IF(RKA.GT.0.0) CFX1 = CF(X1,RKA)
      DX     = X1
 
   10 CFS    = SCF(DX)
      CFX    = CFS
      IF(RK.GT.0.0) CFX = CF(DX,RK)
      DPS    = SCFP(DX)
      IF (CFX.NE.CFS) DPS = RCFP(DX,RK)
      IF (DPS.EQ.0.0) GO TO 200
C
      DX1    = DX -(DX *CFX - X1 *CFX1)/DPS
      IF (DX1.LE.0.0) DX1 = 0.5 *DX
      E1     = ABS(DX1 -DX)
      E2     = 0.0001 *XL
      IF (E1.LE.E2) GO TO 20
      DX     = DX1
      GO TO 10
C
   20 IF( J.EQ.2 ) GO TO 30
      X1     = DX +(XOL2 -XOL1)*XL
      IF (XOL2.GE.1.0) GO TO 25
      RKA    = RK2
      RK     = RK3
      J      = 2
      GO TO 5
C
C     INCREASED ROUGHNESS ENDS AT TRAILING EDGE
C
   25 XP     = X1
      RK3    = RK2
```

```
      GO TO 35
C
C
 30   XP     = DX1 + XL *(1.-XPL2)
 35   CFXP   = SCF(XP)
      IF (RK3.GT.0.0) CFXP = CF(XP,RK3)
      CFR    = (XP/XL) * CFXP
      GO TO 300
C
 200  WRITE (6,1000) DX,CFY,RK
      CFR    = 0.0
 300  CFXL   = SCF(XL)
C
 1000 FORMAT(10X*SUBROUTINE CFK3 WILL NOT CONVERGE*/1X,3F15.7 )
C
      RETURN
      END
```

```
      SUBROUTINE EFFQ(RNOFT, XI, XMACH, HEIGHT, QEFF)
C
C     CALCULATE  EFFECTIVE DYNAMIC PRESSURE RATIO
C
      QEFF  = 1.0
      IF (XI.LE.0.0)  RETURN
      BLT     = 0.37 *XI  /  (RNOFT * XI)**0.2
C
   10 IF (HEIGHT.LE.BLT)   QEFF= .7778 * (HEIGHT /BLT)**0.286
      IF (HEIGHT.GT.BLT)   QEFF= 1.0 - 0.222 *(BLT./HEIGHT)
C
      RETURN
      END
```

```
      SUBROUTINE NUCDAM2(J)
C**********************************************************************
C***********                                                ***********
C***********   EFFECT OF MISSING PARTS ON LIFT CURVE,       ***********
C***********     POLAR SHAPE, AND MOMENTS                   ***********
C**********************************************************************
      COMMON /INPUT/   A(3081)
      COMMON /OUTPUT/  B(222)
      COMMON /CALC/  C(50)
      COMMON /BLKPRT/  KPRINT(50)
      COMMON /BLKCLA/  SPLANF, TOCW,TRP, ARP, DMSTR, CLD, DM1,
     1                 DCPP, EPSL, SWPMC, SPEFP
      COMMON /NUCOUT/  DAMCD(7,17), SYM(7,17), DCDMIN,
     1           ALP(21), CL(21),    CD(21),    CM(21),    CLT(21),   CDT(21),
     2                 CLP(21),  CDP(21),  CMP(21),  CLTP(21),  CDTP(21),
     3                 RCLA(6), DXAC,
     4                 CROLL(21), CLRT(21), X3(2), X4(2)
C
      DIMENSION  DWING(20), FMSURV(20), IDAM(20)
     1         ,DCLA2(2), DXAC2(2), DK2(2), DCLAT(2)
     2          ,SUR(7,20), DSUR(7,100)
C
      EQUIVALENCE  (A(3062),DWING(1)) ,  ( B(54)   ,   XK  ) ,
     1             (B(61)   , R       ) ,  ( B(59)   ,  CLA  ) ,
     2             (C(6)    ,SWPOC    ) ,  ( C(29)   ,  AR   ),
     3             (A(491)  , NSURV   ) ,  (A(492)   , NCLAS) ,
     4             (C(4)    , SPLAN   ) ,  (C(1)     , BO2   ) ,
     5             (C(2)    , DOP     ) ,  (C(3)     , CR    ) ,
     6             (B(56)   , CMO     ) ,  (B(60)    , ALC   ) ,
     7             (B(96)   , CLAW    ) ,  (C(16)    , CTX   ) ,
     8             (C(25)   , SWPLE   ) ,  (C(15)    , CRX  ) ,
     9             (A(493)  ,FMSURV(1)),   (A(12)    , CMAC  )
     1           ,(A(1642), IDAM(1)) ,    (A(11)     , SREF  ),
     2             (C(23)   , XH      ) ,. (B(66)    , EF    ),
     3             (B(98)   , CLAT    ) ,  (A(3)     , NPNLS),
     4             (A(184)  , SUR(1,1)),   (A(2362) ,DSUR(1,1))
     5           ,(B(101)  , HSTAR)
C
      COMMON /BLKDAT3/XMACH3(17),   CDMK3(17),
     1                XLDH3(6)   ,  HDX3(5)    ,    CDD3(6,5)   ,
     2                XDD3(6)    ,  CLD3(4)    ,    YM3(6,4)    ,
     3                YMACH3(13),   CPF3(13)  ,
     4                ZMACH3(13),   CPA3(13)  ,
     5                XLDH4(12)  ,  HDY4(7)    ,    CDD4(12,7) ,
     6                RATIO3(8)  ,  UMACH(10),      DRAG3(8,10),
     7                XLOD(6)    ,  VMACH(6)   ,    DRAG4(6,6)  ,
     8                DETA1(6)   ,     AR1(5)  ,    CLA1(6,5)   ,
     9                DETA2(6)   , SWPOC2(5)   ,    CLA2(6,5)   ,
     1                ETA13(6)   ,    CLA3(6)  ,
     2                DETA4(6)   ,     AR4(5)  ,    CLA4(6,5)   ,
     3                DETA5(6)   , SWPOC5(5)   ,    CLA5(6,5)   ,
     4                ETA16(6)   ,    CLA6(6)  ,
     5                DECP(6)    ,    DCCC(6)  ,   CLACLA(6,6)
C
      COMMON /BLKDAT4/DETA11(6) ,     AR11(5) ,   XAC1(6,5)     ,
     1                DETA12(6) ,  SWOC12(5)  ,   XAC2(6,5)     ,
     2                ETA113(6) ,                 XAC3(6)       ,
     3                DETA14(6) ,     AR14(5) ,   XAC4(6,5)     ,
     4                DETA15(6) ,  SWOC15(5)  ,   XAC5(6,5)     ,
     5                ETA116(6) ,                 XAC6(6)       ,
     6                DCDC7(5)  ,                 XAC7(5)       ,
     7                DCDC8(5)  ,                 XAC8(5)
     8               ,FLD1(6)   ,                 CLAH1(6)      ,
     9                FLD2(6)   ,                 FFH2(6)       ,
     1                ZAR(10)   ,                 XCLA(10)
C
      SWPOC   =     SWPOC * 57.2958
      DO 101  I =1,6
      RCLA(I)   =  0.0
  101 CONTINUE
      DCLA      =   0.0
      DK        =  0.0
      DXAC      =  0.0
      DCLAT(1)   = 0.0
      DCLAT(2)   = 0.0
      DCLA1   = 0.0
      DCCLA2  = 0.0
```

```
      DCLA3   = 0.0
      DCLA4   = 0.0
      DCLA5   =  0.0
      DCLA6   =  0.0
      DXAC1   = 0.0
      DDXAC2  = 0.0
      DXAC3   = 0.0
      DXAC4   = 0.0
      DXAC5   = 0.0
      DXAC6   = 0.0
      DXAC7   = 0.0
      DXAC8   = 0.0
      X3(1)   = 0.0
      X3(2)   = 0.0
      X4(1)   = 0.0
      X4(2)   = 0.0
      DCLA2(1) = 0.0
      DCLA2(2) = 0.0
      DXAC2(1) =0.0
      DXAC2(2) =0.0
      DK1      = 0.0
      DK2(1)   = 0.0
      DK2(2)   = 0.0
C
      IF (KPRINT(29).EQ.1)
     1WRITE(6,1090) FMSURV(J),   ( B(I),  I=1,223)
C
      XMACH   = FMSURV(J)
      AAA     = ABS( B(144) - XMACH)
      IF (AAA.GT.0.001)  WRITE(6,1001)
 1001 FORMAT (//,20X,* INCORRECT MACH NUMBER WAS READ FROM *,
     1                  *TAPE UNIT NUMBER 10   * )
      IF (IDAM(17).EQ.0)  GO TO 146
C
C**************** DEFINE INCREMENT IN LIFT-CURVE SLOPE   ****************
C**************** FOR DAMAGED L.E. AND T.E. ,   DCLA1    ****************
C
      XXX     =  0.0
      YYY     =  0.0
      IF (DWING(3).EQ.0.0)  GO TO 110
      DCLA1 = DLNT(DWING(3), AR    , DETA1,    AR1 , CLA1, 6,5,6,2,2)
      DDCLA2= DLNT(DWING(3), SWPCC, DETA2, SWPCC2, CLA2, 6,5,6,2,2)
      CALL LNTP(DWING(2), DCLA3, ETA13, CLA3, 6, 4)
      XXX     = DCLA1 + DDCLA2 + DCLA3
      IF (DWING(6).LT.3.) XXX  = 0.5 * XXX
  110 IF (DWING(9).EQ.0.0)  GO TO 120
      DCLA4 = DLNT(DWING(9), AR    , DETA4,    AR4 , CLA4, 6,5,6,2,2)
      DCLA5 = DLNT(DWING(9) ,SWPCC, DETA5, SWPCC5, CLA5, 6,5,6,2,2)
      CALL LNTP(DWING(8), DCLA6, ETA16, CLA6, 6, 4)
      YYY     =  DCLA4 + DCLA5 + DCLA6
      IF (DWING(10).LT.3.) YYY = 0.5 * YYY
C
C     DCLA    =  DELTA-CLA-TOTAL
  120 DCLAX   =  XXX  +  YYY
C
C     CLARLE IS   CL-ALPHA RATIO TO ACCOUNT FOR CHORD OF CUTOUT
      CLARLE = DLNT(DWING(2), DWING(1), DRC2, DCC, CLACLA,6,5,6,2,2)
      CLARTE  = DLNT(DWING(9), DWING(7), DRCR, DCCC, CLACLA,6,5,6,2,2)
      IF (DWING(6).LT.3.)  CLARLE = CLARLE +0.5 * (1.0-CLARLE)
      IF (DWING(10).LT.3.) CLARTE = CLARTE +0.5 * (1.0-CLARTE)
C
C     CLAP    =  CLA-PRIME,    PRIME DENOTES DAMAGED CONFIGUATION
      CLAP   = (CLA + DCLAX) * CLARLE * CLARTE
      IF (KPRINT(29).EQ.1)
     1WRITE(6,1010) FMSURV(J), DCLA1, DDCLA2, DCLA3, DCLA4, DCLA5,
     1                   DCLA6, CLARLE, CLARTE, CLAP
      DCLA1   =  CLAP - CLA
C     DEFINE LIFT CURVE INCREMENT REQUIRED FOR ROLL TRIM
C
      RCLA(1)   = (CLA+XXX) * CLARLE  - CLA
      RCLA(2)   = (CLA+YYY) * CLARTE  - CLA
      IF (KPRINT(29).EQ.1)
     1WRITE(6,1020) DCLA1
C
C**************** DEFINE AERODYNAMIC CENTER SHIFT        ****************
C**************** FOR DAMAGED L.E. AND T.E. ,   DXAC1    ****************
C
```

```
      XXX     =  0.0
      YYY     =  0.0
      IF (DWING(3).EQ.0.0) GO TO 130
      DXAC1   = DLNT(DWING(3), AR  ,DETA11, AR11 , XAC1, 6,5,6,2,2)
      DDXAC2  = DLNT(DWING(3),SWPCC,DETA12,SWCC12, XAC2, 6,5,6,2,2)
      CALL LNTP(DWING(2), DXAC3, ETAI13, XAC3,6, 4)
      CALL LNTP(DWING(1), DXAC7, DCCC7,XAC7, 5, 4)
      XXX     =   DXAC1 + DDXAC2 + DXAC3 + DXAC7
      IF (DWING(6).LT.3.)    XXX  = 0.5 * XXX
      IF (KPRINT(29).EQ.1)
     1WRITE(6,1030) DXAC1, DDXAC2, DXAC3, DXAC7, XXX
  130 IF (DWING(9).EQ.0.0) GO TO 140
      DXAC4   = DLNT(DWING(9), AR  ,DETA14, AR14 , XAC4, 6,5,6,2,2)
      DXAC5   = DLNT(DWING(9),SWPCC,DETA15,SWCC15, XAC5, 6,5,6,2,2)
      CALL LNTP(DWING(8), DXAC6, ETAI16, XAC6, 6, 4)
      CALL LNTP(DWING(7), DXAC8, DCCC8, XAC8, 5, 4)
      YYY     =   DXAC4 + DXAC5 + DXAC6 + DXAC8
      IF (DWING(10).LT.3.)  YYY  = 0.5 * YYY
      IF (KPRINT(29).EQ.1)
     1WRITE(6,1035)  DXAC4, DXAC5, DXAC6, DXAC8, YYY
  140 DXAC1   = XXX + YYY
C
      IF (KPRINT(29).EQ.1)
     1WRITE(6,1036)  DXAC1
C
C***************** DEFINE INCREMENT IN SPAN EFFICIENCY   ***************
C***************** FOR DAMAGED L.E. AND T.E  ,  DE1      ***************
C
C           THIS METHOD NOT COMPLETE, MEANWHILE SET
      DE1     = 0.0
      IF (DWING(1).GT.0.0.OR.DWING(7).GT.0.0)  DE1 = -0.05
C
C***************** DEFINE INCREMENT IN POLAR SHAPE FACTOR ***************
C***************** FOR DAMAGED L.E. AND T.E  ,  DK1      ***************
C
      XXX     =  DWING(3)
      IF (DWING(6).LT.3.)  XXX  = 0.5 * XXX
      RP      = R*(1.0 -XXX) + 0.3*R*XXX
      EFP     =  EF + DE1
      XKP     =(1.0-RP)/(CLAP*57.29578*SREF/SPLAN)+ RP/(3.14159*AR*EFP)
      XKP     = XKP * (SREF/SPLAN)
      DK1     =  XKP - XK
      IF (KPRINT(29).EQ.1)
     1WRITE(6,1040) R, RP, EF, EFP, XK, XKP, DK1
C
C***************** DEFINE INCREMENT IN LIFT-CURVE SLOPE  ***************
C***************** FOR MISSING WING TIPS,        DCLA2   ***************
C
      DO 141 I=1,2
      DCLA2(I)   = 0.0
      DXAC2(I)   = 0.0
      DK2(I)     = 0.0
      DCLAT(I)   = 0.0
  141 CONTINUE
      IF (DWING(11).EQ.0.0.AND.DWING(12).EQ.0.0) GO TO 145
C
C     LET I=1  INDICATE LEFT WING
C     LET I=2  INDICATE RIGHT WING
      DO 143  I = 1, 2
      DR      =   BO2 * 2.0 * DWING(10+I)
C
C         DEFINE VARIABLES REQUIRED TO CALL SUBROUTINE AER2
      TOCW    =  C(12)
      DMSTR   =  0.0
      CLD     =  C(13)
      DM1     =  0.0
      EPSL    =  0.0
      SWPMC   =  C(7)
      SREF    =  A(11)
      BW      =  2.* BO2
      BWP     =  BW - DR
      CTXP    =  CR - (BWP/BW) * (CR-CTX)
      SPLAN1  = (CR+CTX) * BW / 2.0
      SPLAN2  = (CR+CTXP)* BWP/ 2.0
      SPLANP  =  SPLAN  + ( SPLAN2 - SPLAN1)
      ARP     =  BWP**2 /  SPLANP
      TRP     =  CTXP / CR
```

```
      D       =  DOB   + BW
      DCBP    =  D     / BWP
      SREFP   =  SREF
C
C            CALC. INCREMENT IN POTENTIAL LIFT-CURVE SLOPE
C
      CLAWP   = 0.0
      IF (DCBP.LT.1.0)  CALL APP2(XMACH,CLAWP)
      POTENT  =  CLAWP - CLAW
C
C            CALC. INCREMENT IN VORTEX    LIFT-CURVE SLOPE
C
      SONICM =    1.0 / (SIN(90.0/57.295 - SWPLE))
      VORK   =    1.0 - (XMACH-1.0) / (SONICM-1.0)
      IF (XMACH.LE.1.0)         VORK = 1.0
      IF (XMACH.GT.SONICM)      VORK = 0.0
      CALL LNTP(AR  , VOR1 ,  ZAP , XCLA, 10, 4)
      CALL LNTP(ARP , VOR2 ,  ZAP , XCLA, 10, 4)
      VORTEX  = (VOR2 - VOR1) * VORK
C
C            CALC. TOTAL DELTA CLA ON ONE WING
C
      DCLA2(I) =   0.5 *(POTENT + VORTEX)
      IF (KPRINT(29).EQ.1.AND.I.EQ.1)
     1WRITE(6,1045)  I
      IF (KPRINT(29).EQ.1.AND.I.EQ.2)
     1WRITE(6,1046)  I
      IF (KPRINT(29).EQ.1)
     1WRITE(6,1050) BWP,CTXP,SPLANP,ARP,TRP,CLAWP,CLAW,
     2          SONICM, VORK, POTENT, VORTEX, DCLA2(I)
C
C     DEFINE LIFT CURVE INCREMENT REQUIRED FOR ROLL TRIM
      RCLA(2+I) = DCLA2(I)
C
C**************** DEFINE AERODYNAMIC CENTER SHIFT       ****************
C**************** FOR MISSING WING TIPS,  DXAC2         **************
C
C     CALCULATIONS BASED ON EXPOSED WING
      CBAR1  = 0.6667*(( CRX+CTX ) - (CRX * CTX ) / (CRX + CTX ))
      CBAR2  = 0.6667*(( CRX+CTXP) - (CRX * CTXP) / (CRX + CTXP))
      TR     =   CTX / CRX
      TRP  =   CTXP / CRX
      ETA1    =   .3333*((1. +2. * TR ) / (1. + TR ))
      ETA2    =   .3333*((1. +2. * TRP) / (1. + TRP))
      BEXP    = BW  -  D
      BPEXP   =  BWP  - D
      Y1      = ETA1  * BEXP  / 2.0
      Y2      = ETA2  * BPEXP / 2.0
      XLE1    =  Y1 * TAN(SWPLE)
      XLE2    =  Y2 * TAN(SWPLE)
      IF (XMACH.LT.1.0)  DX = (XLE2+.25*CBAR2)-(XLE1+.25*CBAR1)
      IF (XMACH.GE.1.0)  DX = (XLE2+.50*CBAR2)-(XLE1+.50*CBAR1)
      DXAC2(I) = DX / CMAC
      DXAC2(I) = 0.5* DXAC2(I)
      IF (I.EQ.1)  X3(1) = Y1 +  DOB * BO2
      IF (I.EQ.1)  X3(2) = Y2 +  DOB * BO2
      IF (I.EQ.2)  X4(1) = Y1 +  DOB * BO2
      IF (I;EQ.2)  X4(2)  = Y2 +  DOB * BO2
C
      IF (KPRINT(29).EQ.1)
     1WRITE(6,1061) CBAR1,CBAR2,ETA1,ETA2,XLE1,XLE2,DXAC2(I)
C
C**************** DEFINE INCREMENT IN SPAN EFFICIENCY   ****************
C**************** FOR MISSING WING TIPS,        DE2       ***************
C
C     FOR NOW, ASSUME
      DE2     = 0.0
C
C**************** DEFINE INCREMENT IN POLAR SHAPE FACTOR ***************
C**************** FOR MISSING WING TIPS,       DK2      ***************
C
      EFP     = EF + DE2
      DK2(I)  =(1.0/(2.0*3.14159))*((CR+CTXP)/(BWP*EFP)-(CR+CTX)/(BW*EF))
      DK2(I)  = DK2(I) * (SREF/SPLAN)
      DK2(I)  = 0.5 * DK2(I)
      IF (KPRINT(29).EQ.1)
     1WRITE(6,1064) EFP, DK2(I)
```

```
  143 CONTINUE
C
C**************** DETERMINE EFFECTS OF MISSING HORIZ. TAIL ***********
C
  145 DO 144    I = 1,2
      DCLAT(I) = CLAT * (1.0- DWING(13+I)) - CLAT
      DCLAT(I) =  0.5 * DCLAT(I)
      RCLA(4+I)=  DCLAT(I)
  144 CONTINUE
C
C************* DEFINE EFFECT OF HOLES IN WING ON LIFT-CURVE ***********
C*************            SLOPE AND POLAR-SHAPE FACTOR           ***********
C
  146 DCLAHO  = 0.0
      DKHOL    = 0.0
      IF (IDAM(14).NE.1)   GO TO 210
C
C     HOLE AREA = NO. OF HOLES * LENGTH   * WIDTH
      AREAUP = DSUR(1,32) * DSUR(1,33) * DSUR(1,34)
      AREALO = DSUR(1,31) * DSUR(1,33) * DSUR(1,34)
C     FLOW AREA = HOLE AREA * POROSITY FACTOR
      FLOW1  = AREAUP      * DSUR(1,39)
      IF (AREALO.LT.AREAUP) FLOW1   = AREALO * DSUR(1,39)
C
      FLOW2  = 0.0
      IF (NPNLS.EQ.1)  GO TO  200
      AREAUP = DSUR(2,32) * DSUR(2,33) * DSUR(2,34)
      AREALO = DSUR(2,32) * DSUR(2,33) * DSUR(2,34)
      FLOW2  = AREAUP      * DSUR(2,39)
      IF (AREALO.LT.AREAUP) FLOW2 =  AREALO * DSUR(2,39)
C
  200 IF (SYM(1,14).EQ.1.0)  FLOW1  =  2. * FLOW1
      IF (SYM(2,14).EQ.1.0)  FLOW2  =  2. * FLOW2
      FLOW    = FLOW1 + FLOW2
      FLOWR = FLOW / (SPLAN/2.)
C
      CALL LNTP(FLOWR , CLAFAC ,  FLO1  , CLAH1 , 5, 4)
      CALL LNTP(FLOWR , DEFAC  ,  FLO2  , EFF2  , 5, 4)
      DCLAHO =   CLA*CLAFAC -  CLA
      DKHOL  =   1./(3.14*EF*DEFAC*AR)  -  1./(3.14*EF*AR)
      IF (KPRINT(29).EQ.1)
     1WRITE(6,1110) FLOW1, FLOW2, FLOWR, CLAFAC, DEFAC, DCLAHO, DKHOL
  210 CONTINUE
C
C***************       SUM EFFECTS ON UNTRIMMED CL, CD, AND CM   ********
C***************               CAUSED BY DAMAGE                  ********
C
      DCLA     = DCLA1+DCLA2(1)+DCLA2(2)+DCLAT(1)+DCLAT(2)+DCLAHO
      DXAC     = DXAC1 + DXAC2(1)   + DXAC2(2)
      DK       = DK1 + DK2(1) + DK2(2) + DKHOL
      IF (KPRINT(29).EQ.1)
     1WRITE(6,1070) DCLA, DXAC, DK, (RCLA(I), I = 1,6)
C
C         CALCULATE UNTRIMMED LIFT, DRAG, AND MOMENT
C               FOR THE DAMAGED AIRPLANE
C
      CLAP    =   CLA + DCLA
      IF (KPRINT(29).EQ.1)
     1WRITE(6,1095)
      DO 150    I = 1, NCLAS
C
      XCL     =   CLA * (ALP(I) - ALO)
      XCLP    =   CLAP* (ALP(I) - ALO)
      CLP(I)  =   CL(I) +  (XCLP - XCL)
C
C
      XCLT    = CLA *(ALP(I)-ALO) + (DCLAT(1)+DCLAT(2)) * (ALP(I)-HSTAB)
      XCD     =   XK *      XCL**2
      XCDT    =   XK *      XCLT**2
      CDTAIL  =   XCD - XCDT
      CALL LNTP(CLP(I), CDX, CL, CD, NCLAS, 4)
      CDP(I) = CDX + CDOMIN + DK*CLP(I)**2 + CDTAIL
C
C
      CLWNGP = (CLAW+DCLA1+DCLA2(1)+DCLA2(2)+ DCLAHO) * (ALP(I)-ALO)
      CMWING  =  - CLWNGP * DXAC
      CMTAIL = -(DCLAT(1)+DCLAT(2))* (ALP(I)-HSTAB) * XH/CMAC
```

```
      CALL LNTP(CLP(I), CMX, CL, CM, NCLAS, 4)
      CMP(I) =  CMX +  CMWING +  CMTAIL
      IF (KPRINT(29).EQ.1)
     1WRITE(6,1100)(ALP(I), CL(I), CLP(I), CD(I), CDTAIL, CDX,
     2              CDP(I),CM(I), CMWING, CMTAIL, CMX, CMP(I))
  150 CONTINUE
C
      SWPCC  =   SWPCC / 57.2958
C
 1010 FORMAT( ////, 10X, *DUMP FROM SUBROUTINE MUCDAM2,  MACH = *,
     1            F6.3,//, 10X, * DCLA1, DCLA2, DCLA3, DCLA4, DCLA5, *,
     2            *DCLA6  =* ,  6F10.4, /, 10X,
     3            *CLARLE, CLARTE, CLAP  = *, 3F10.4)
 1020 FORMAT(        10X,  *DCLA1  = * , F10.4)
 1030 FORMAT (   / ,10X, *DXAC1, DXAC2, DXAC3, DXAC7, XXX =*,5F10.4)
 1035 FORMAT(        10X, *DXAC4, DYAC5, DXAC6, DXAC8, YYY =*,5F10.4)
 1036 FORMAT (       34X, *DXAC1 = *, F10.4)
 1040 FORMAT(   /, 10X,   *R, PF, EF, EFP, XK, XKP, DK1 = *, 7F8.4)
 1045 FORMAT(   //,10X,*DATA FOR LEFT WING, I = *, I2)
 1046 FORMAT(   //,10X,*DATA FOR RIGHT WING, I = *, I2)
 1050 FORMAT(    10X,*EWP, CTXP, SPLANP, ARP, TRP, CLAWP, CLAW, *,
     1            *         = *, 3F9.2, 4F9.4,/,10X,
     2            *SONICM, VORK, POTENT, VORTEX, DCLA2(I)            =*,
     3            5F10.5)
 1061 FORMAT (     10X, *CBAR1, CBAR2, ETA1, ETA2, XLE1, XLE2, *,
     1            *DYAC2(I) = *, 7F8.3)
 1064 FORMAT (     10X, *EPP, DK2(I)   = *, 25X, 2F10.4)
 1070 FORMAT ( /,10X, *TOTAL INCREMENTS FOR THE CONFIGURATION *,
     1           /,10X, *DCLA, DYAC, DK  = *, 3F10.4,/,10X,
     2         *RCLA(1), RCLA(2), RCLA(3), RCLA(4), RCLA(5), RCLA(6) =*,
     3         6F10.5)
 1090 FORMAT(1H1, ///, 10X, *DATA IN COMMON BLOCK OUTPUT, (B MATRIX)*,
     1         //, 10X, *MACH = *, F10.3,//, (10X, 10F10.5))
 1095 FORMAT(1H1,//,3X,
     1   *  ALP(I)        CL(I)       CLP(I)      CD(I)     CDTAIL  *,
     2   *   CDX          CDP(I)      CM(I)      CMWING    *,
     3   *   CMTAIL       CMX         CMP(I) *,//)
 1100 FORMAT( 2X,12F10.5)
 1110 FORMAT( /,10X,*FACTORS FOR HOLES IN WING, FLOW1, FLOW2, FLOWP = *,
     1      3F10.4,/, 28X,*CLAFAC, DEFAC, DCLAHC, DKHCL = *, 4F10.4)
      RETURN
      END
```

```
      SUBROUTINE NTRIM(J)
C**********************************************************************
C***************** EFFECT OF MISSING WING PARTS ON TRIM DRAG ************
C**********************************************************************
C
      COMMON /INPUT/  A(3081)
      COMMON /OUTPUT/  B(223)
      COMMON /CALC/  C(50)
      COMMON /BLKPRT/  KPRINT(50)
      COMMON /NUCOUT/  CANCR(7,17), SYM(7,17), DCCMIN,
     1        ALP(21), CL(21),    CD(21),    CM(21),    CLT(21),    CDT(21),
     2                 CLP(21), CDP(21),  CMP(21),  CLTP(21),  CDTP(21),
     3                 RCLA(6), DXAC,
     4                 CROLL(21), CDRT(21), X3(2), X4(2)
      DIMENSION DWING(20), SUR(7,30)
      EQUIVALENCE  ( B(43)   ,    CL1) ,  ( B(44)   ,    CD1) ,
     1             ( A(4)    ,   NHT ) ,  ( B(46)   , ALPHA) ,
     2             ( A(3)    , NPNLS) ,   ( A(492)  , NCLAS) ,
     3             ( C(1)    ,    BO2) ,  ( C(2)    , DOF   ) ,
     4             ( B(60)   ,    ALO) ,  (A(12)    , CMAC ) ,
     5             ( A(3062), DWING(1)),  (A(154),  SUR(1,1))
     6            ,( B(98)   , CLAT  ) ,  ( B(104)  , CLDH )
     7            ,( B(58)   ,XACWB)   ,  ( B(130)  ,  XACS  )
     8            ,( B(144)  , XMACH)  ,  ( A(7)    , IREF   ),
     9             ( A(275)  , WINC  ) ,  ( B(56)   , CLAW   )
     1            ,(B(101)   , HSTAB)
C
C     CALCULATE TRIMMED LIFT AND DRAG
C     FOR THE DAMAGED AIRCRAFT
C
      IF (KPRINT(30).EQ.1)
     1WRITE(6,1010) XMACH     , CLAT, CLDH, XACWB, XACS
      IF (NHT.EQ.0)  GO TO 80
      CLAT    = CLAT * (1.0 - (DWING(14)+DWING(15)) /2.0 )
      CLDH    = CLDH * (1.0 - (DWING(14)+DWING(15)) /2.0 )
   80 CONTINUE
      XACWB   =  XACWB   + DXAC
      XACS    =  XACS    + DXAC
      IF (KPRINT(30).EQ.1)
     1WRITE(6,1020)  CLAT, CLDH , XACWB, XACS
C
      X1      = (DWING(2) +0.4 * DWING(3)) * BO2
      X2      = (DWING(8) + 0.4 * DWING(9)) * BO2
      X5      = SUR(NPNLS+1,12)+ 0.333 * SUR(NPNLS+1,10)
      X6      = X5
C
      IF (KPRINT(30).EQ.1)
     1WRITE(6,1025) (RCLA(I), I=1,6), X1,X2,X3(1),X3(2),X4(1),
     2               X4(2),X5,X6
      DO 100  I  = 1, NCLAS
C
      ALPHA   =  ALP(I)
      IF (IREF.EQ.1)  ALPHA = ALPHA + WINC
      CL1     =  CLP(I)
      CD1     =  CDP(I)
      CALL TDRG(DCLT,DCDT)
      IF (KPRINT(30).EQ.1)
     1WRITE(6,1030) ALPHA, CL1, CD1
C
C
C             CALCULATE DRAG THAT RESULTS FROM TRIMMING OUT
C             ROLLING MOMENTS FROM ASYMMETRIC DAMAGE ON
C                 WING L.E. ,  T.E. ,    OR  TIP
C
C              ROLLING MOMENT DUE TO DAMAGED L.E.
      RCL1    =  RCLA(1) * (ALP(I)-ALO)
      CROLL1 =  RCL1 * (X1 / (2. *BO2))
      IF (DWING(6).EQ.2.0) CROLL1 =  -CROLL1
      IF (DWING(6).EQ.3.0) CROLL1 =    0.0
C
C
C              ROLLING MOMENT DUE TO DAMAGED T.E.
      RCL2    =  RCLA(2) * (ALP(I)-ALO)
      CROLL2 = RCL2 * (X2 / (2. * BO2))
      IF (DWING(10).EQ.2.0) CROLL2 =  -CROLL2
      IF (DWING(10).EQ.3.0) CROLL2 =    0.0
C
```

```
C
C          ROLLING MOMENT DUE TO LEFT WING TIP MISSING
      RCL3    =    RCLA(3) * (ALP(I)-ALC)
      CLWING  =  (CLAW/2.)* (ALP(I)-ALC)
      CRUND   =    CLWING  *  X3(1) /           (2. *BC2)
      CRD     =  (CLWING+RCL3) *  X3(2) / (2. *BC2)
      CROLL3  =  CRD - CRUND
C
C
C          ROLLING MOMENTS DUE TO RIGHT TIP MISSING
      RCL4    =    RCLA(4) * (ALP(I)-ALC)
      CLWING  =  (CLAW/2.)* (ALP(I)-ALC)
      CRUND   = - CLWING  *  X4(1) /           (2. *BC2)
      CRC     = -(CLWING+RCL4) *  X4(2) / (2. *BC2)
      CROLL4  =  CRC - CRUND
C
C
C          ROLLING MOMENTS DUE TO LEFT HORIZ.TAIL AREA MISSING
      RCL5    =    RCLA(5) * (ALP(I)-HSTAR)
      CROLL5  =    RCL5  * (X5 / (2. *BO2))
C
C
C          ROLLING MOMENTS DUE TO RIGHT HORIZ.TAIL AREA MISSING
      RCL6    =    RCLA(6) * (ALP(I)-HSTAR)
      CROLL6  =  -RCL6  * (X6 / (2. *BO2))
C
C     THE TOTAL ROLLING MOMENT IS DEFINED AS:
      CROLL(I) = CROLL1+CROLL2+CROLL3+CROLL4+CROLL5+CROLL6
      IF (KPRINT(20).EQ.1)
     1WRITE(6,1040)  RCL1  , RCL2  , RCL3  , RCL4  , RCL5  , RCL6  ,
     2               CROLL1, CROLL2, CROLL3, CROLL4, CROLL5, CROLL6,
     3               CROLL(I)
      XBAR    =  DWING(13) * BO2
C
C
C     DELTA-CL REQUIRED ON TRIMMING DEVICE IS
      CLRT    =  ABS((CROLL(I) * 2.0 * BO2) /   XBAR )
C
C     RESULTING DRAG
      CDRT(I)   = CLRT * TAN(ALP(I) / 57.2957)
C
C
C          CALCULATE TOTAL LIFT AND DRAG FOR THE DAMAGED AIRCRAFT
C             INCLUDING PITCH AND ROLL TRIM
C
      CLTP(I)   =   CLP(I) + DCLT
      CDTP(I)   =   CDP(I) + DCDT + CDRT(I)
C
      IF (KPRINT(30).EQ.1)
     1WRITE(6,1050) CLP(I), DCLT, CLTP(I),
     3              CDP(I),DCDT, CDRT(I), CDTP(I)
  100 CONTINUE
C
 1010 FORMAT(1H1,//,10X,*DUMP FROM SUBROUTINE LTRIM, MACH=*,F5.3,//,31X,
     1       *CLAT        CLDH        XACWB       XACS *, /, 10X,
     2       *FOR UNDAMAGED A/C*, 4F10.3)
 1020 FORMAT ( /  ,10X,*FOR   DAMAGED A/C*, 4F10.3)
 1025 FORMAT(/ ,5X,
     1       *RCLA(1), RCLA(2), RCLA(3), RCLA(4), RCLA(5), RCLA(6) =*,
     2       6F8.3,/, 5X,
     3       *X1, X2, X3(1), X3(2), X4(1), X4(2), X5, X6          =*,
     4       8F8.3,/ )
 1030 FORMAT( / ,22X, *ALPHA,CL1, CD1 =*,F10.3,2F10.5)
 1040 FORMAT(  , 5X,
     1       *RCL1   , RCL2   , RCL3   , RCL4   , RCL5   , RCL6    =*
     2       6F8.3,/, 5X,
     3       *CROLL1 , CROLL2 , CROLL3 , CROLL4 , CROLL5 , CROLL6  =*
     4       6F8.3,/, 10X, *TOTAL ROLLING MOMENT, CROLL(I) =*,F10.3)
 1050 FORMAT(    10X,*CLP(I), DCLT, CLTP(I),                  = *, 3F10.4,/,
     2       10X,*CDP(I), DCDT, CDRT(I),  CDTP(I)  = *, 4F10.5)
      RETURN
      END
```

```
      SUBROUTINE WRITE2(J)
C
C              P R I N T S   F I N A L   R E S U L T S
C
      COMMON /INPUT/  A(3081)
      COMMON /BLKTIL/  TITLE(6)
      COMMON /BLKPRT/  KPRINT(50)
      COMMON /ALCOLT/  CAMCF(7,17), SYM(7,17), DCDMIN,
     1       ALP(21), CL(21),    CD(21),    CM(21),    CLT(21),    CDT(21),
     2                CLP(21),   CDP(21),   CMP(21),   CLTP(21),   CDTP(21),
     3                RCLA(6), DXAC,
     4                CROLL(21), CDRT(21), X3(2), Y4(2)
      DIMENSION FMSURV(20), EALPT(21), EALPP(21), EALPTP(21),
     1          ECMP(21), ECDT(21), ECDP(21), ECDTP(21), ECROLL(21)
C
      EQUIVALENCE  (A(493) , FMSURV(1) ), (A(491) , NSURV ),
     1             (A(492) , NCLAS) ,  (A(22) , ITRIM)
C
      IF (KPRINT(31).EQ.1)
     1WRITE(6,3000) (ALP(I), CL(I), CLP(I), CLT(I), CLTP(I),
     2                       CM(I), CMP(I),
     3                       CD(I), CDP(I), CDT(I), CDTP(I),
     4                       I = 1, NCLAS)
C
      IF (KPRINT(31).EQ.1)
     1WRITE(6,3010)
C
      XMACH  =   FMSURV(J)
      WRITE(6,2000) (TITLE(I),   I = 1, 6)
      WRITE(6,2010)  XMACH
      WRITE(6,2020)
      WRITE(6,2030)
C
C     INTERPOLATE ON DATA FOR VALUES OF DATA AT EVEN CL
C
      DO 100    I = 1, NCLAS
C
      CALL LNTP( CL(I), EALPT(I)    , CLT  , ALP ,  NCLAS, 4 )
      CALL LNTP( CL(I), EALPP(I)    , CLP  , ALP ,  NCLAS, 4 )
      CALL LNTP( CL(I), EALPTP(I)   , CLTP , ALP ,  NCLAS, 4 )
      CALL LNTP( CL(I), ECMP(I)     , CLP  , CMP ,  NCLAS, 4 )
      CALL LNTP( CL(I), ECDT(I)     , CLT  , CDT ,  NCLAS, 4 )
      CALL LNTP( CL(I), ECDP(I)     , CLP  , CDP ,  NCLAS, 4 )
      CALL LNTP( CL(I), ECDTP(I)    , CLTP , CDTP,  NCLAS, 4 )
      CALL LNTP( CL(I), ECROLL(I)   , CLP  ,CROLL,  NCLAS, 4 )
  100 CONTINUE
      DO 110    I  =  1, NCLAS
      DALP      =   EALPP(I)   -   ALP(I)
      DCM     = ECMP(I)   - CM(I)
      DCDL    =   ECDP(I) - CD(I) - DCDMIN
      DCDTOT   = DCDMIN   + DCDL
      WRITE(6,2040) CL(I), ALP(I),CM(I), CD(I),EALPP(I), ECMP(I),
     1         ECDP(I), DALP, DCM, ECROLL(I),DCDMIN    , DCDL, DCDTOT
  110 CONTINUE
      IF (ITRIM.EQ.1) GO TO 130
      WRITE(6,2050)
      WRITE(6,2020)
      WRITE(6,2060)
      DO 120  I   =   1, NCLAS
      DALPT   =   EALPTP(I) - EALPT(I)
      DCDL    = ECDP(I) - CD(I) - DCDMIN
      DCDTOT    = ECDTP(I)   -   ECDT(I)
      CDTRIM    = DCDTOT   - DCDMIN     - DCDL - CDRT(I)
      WRITE(6,2070)  CL(I), EALPT(I), ECDT(I), EALPTP(I), ECDTP(I),
     1         DALPT, CDTRIM, CDRT(I), DCDMIN    , DCDL, DCDTOT
  120 CONTINUE
  130 CONTINUE
 2000 FORMAT (1H1,     40X, 6A10)
 2010 FORMAT (/   ,51X, *MACH = *, F5.3, //, 33X,
     1        *U N T R I M M E D   A E R O D Y N A M I C     D A T A*)
 2020 FORMAT (14X, *X*,7X, *UNDAMAGED*, 10X, *X*, 9X, *DAMAGED*,10X,
     1             *X*,11X, *INCREMENTS DUE TO NUCLEAR DAMAGE*,11X,
     2             *X*, /, 14X, *X  ---------------------  X  *,
     3             *-------------------- X ---------------------*,
     4     *---------------------------- X *, /,14X, *X*, 26X, *X*,
     5             26X, *X DELTA      DELTA      DELTA      DELTA      DELTA*,
     6             *     DELTA    X *)
```

```
2030 FORMAT (    9X, *CL     X  ALPHA     CM        CD       X  ALPHA     CM*,
    1        *           CD      X ALPHA        CM            ROLL        CD MIN*,
    2        *      CDL       CDTOTAL    X*, /, 14X, *X*, 26X,
    3        *X*,26X, *Y*, 54X, *Y*)
2040 FORMAT (   6X,F6.3, *    X *, F6.2, F8.4, F9.5, *    Y *, F6.2,
    1          F8.4, F9.5, *    X *, F5.2,2F9.4, F10.5,2F9.5,*   X *)
2050 FORMAT ( //,35X,*T R I M M E D      A E R O D Y N A M I C   *,
    1                  *     D A T A *)
2060 FORMAT (    9X, *CL     X  ALPHA                    CD       Y  ALPHA         *,
    1        *           CD      X ALPHA     CD TRIM   CD ROLL    CD MIN        CDL*,
    2        *      CD TOTAL X *, /, 14X, *Y*,26X, *X*, 26X, *Y*,54X,*X*)
2070 FORMAT (   6X,F6.3, *    X *, F6.2, 9X, F8.5, *    X *,F6.2,9X,F8.5,
    1      *    X *, F5.2, F10.5, 4F9.5, *    X * )
3000 FORMAT(1H1,//,15X, *CALCULATED DATA AT EVEN VALUES OF ALPHA*,//,
    1       5X,*   ALPHA(I)      CL(I)          CLP(I)        CLT(I)      CLTP(I) *,
    2          *       CM(I)       CMP(I)         CD(I)         CDP(I)      CDT(I)    *,
    3          *    CDTP(I) *, //,    (5X,11F10.5))
3010 FORMAT (/////,
    1     15X, *ALP                 =      ANGLE OF ATTACK                  *, //,
    2     15X, *CL   , CD   , CM    =      UNDAMAGED,      UNTRIMMED *, //,
    3     15X, *CLP  , CDP  , CMP   =        DAMAGED,      UNTRIMMED *, //,
    4      15X,*CLT  , CDT          =      UNDAMAGED,        TRIMMED *, //,
    5     15X, *CLTP, CDTP          =        DAMAGED,        TRIMMED *   )
      RETURN
      END
```

DISTRIBUTION LIST

DEPARTMENT OF DEFENSE

Assistant to the Secretary of Defense
Atomic Energy
ATTN: Executive Assistant

Defense Documentation Center
12 cy ATTN: DD

Defense Nuclear Agency
ATTN: DDST
ATTN: SPAS
ATTN: STSP
4 cy ATTN: TITL

Field Command
Defense Nuclear Agency
ATTN: FCPR

Livermore Division, Field Command, DNA
Lawrence Livermore Laboratory
ATTN: FCPRL

Commandant
NATO School (SHAPE)
ATTN: U.S. Documents Officer

Under Secretary of Defense for Rsch. & Engrg.
ATTN: Strategic & Space Systems (OS)

DEPARTMENT OF THE ARMY

Harry Diamond Laboratories
ATTN: DELHD-N-NP
ATTN: DELHD-N-P, J. Gwaltney

U.S. Army Ballistic Research Labs.
ATTN: DRXBR-BLE, W. Taylor
ATTN: DRDAR-BLV, J. Meszaros

U.S. Army Materiel Dev. & Readiness Cmd.
ATTN: DRCDE-D, L. Flynn

U.S. Army Nuclear & Chemical Agency
ATTN: Library

DEPARTMENT OF THE NAVY

Naval Material Command
ATTN: MAT 08T-22

Naval Research Laboratory
ATTN: Code 2627

Naval Surface Weapons Center
ATTN: K. Caudle

Naval Weapons Evaluation Facility
2 cy ATTN: L. Oliver

Office of Naval Research
ATTN: Code 464

Strategic Systems Project Office
ATTN: NSP-272

DEPARTMENT OF THE AIR FORCE

Aeronautical Systems Division, AFSC
ATTN: ENFT, R. Bachman
4 cy ATTN: ENFTV, D. Ward
10 cy ATTN: ENFTA, R. Sarver

Air Force Materials Laboratory
ATTN: MBE, G. Schmitt

Air Force Weapons Laboratory
ATTN: SUL
ATTN: DYV, G. Campbell
4 cy ATTN: DYV, A. Sharp

Foreign Technology Division, AFSC
ATTN: PDBF, Mr. Spring

Strategic Air Command/XPFS
ATTN: XPFS, B. Stephan

DEPARTMENT OF DEFENSE CONTRACTORS

Aerospace Corporation
ATTN: W. Barry

Avco Research & Systems Group
ATTN: J. Patrick
ATTN: P. Grady

Boeing Company
ATTN: S. Strack
ATTN: R. Dyrdahl
ATTN: E. York

Boeing Wichita Company
ATTN: R. Syring

Calspan Corporation
ATTN: M. Dunn

Effects Technology, Inc.
ATTN: R. Parisse
ATTN: E. Bick
ATTN: R. Wengler

General Dynamics Corporation
Pomona Division
ATTN: R. Schemensky

General Dynamics Corporation
Forth Worth Division
ATTN: G. Howell

General Electric Co.-TEMPO
Center for Advanced Studies
ATTN: DASIAC

General Research Corp.
Santa Barbara Division
ATTN: T. Stathacopoulos

Kaman AviDyne
Division of Kaman Sciences Corp.
ATTN: E. Criscione
ATTN: N. Hobbs
ATTN: R. Ruetenik

DEPARTMENT OF DEFENSE CONTRACTORS (Continued)

Kaman Sciences Corporation
ATTN: D. Sachs

Los Alamos Technical Associates
ATTN: P. Hughes

McDonnell Douglas Corporation
ATTN: J. McGrew

Prototype Development Associates, Inc.
ATTN: C. Thacker
ATTN: J. McDonald

R&D Associates
ATTN: C. MacDonald
ATTN: F. Field
ATTN: J. Carpenter

DEPARTMENT OF DEFENSE CONTRACTORS (Continued)

Rockwell International Corp.
ATTN: R. Sparling

Sandia Laboratories
ATTN: Doc. Con. for A. Lieber

Science Applications, Inc
ATTN: D. Hove

SRI International
ATTN: G. Abrahamson