

**NASA TECHNICAL
MEMORANDUM**

NASA TM X-62,326

NASA TM X-62,326

(NASA-TM-X-62326) NORMAL LOADS PROGRAM
FOR AERODYNAMIC LIFTING SURFACE THEORY
(NASA) 94 p HC \$7.75 CSCL 01A

N74-17702

G3/01 Unclass
31337

**NORMAL LOADS PROGRAM FOR AERODYNAMIC LIFTING
SURFACE THEORY**

Richard T. Medan and K. Susan Ray

**Ames Research Center
Moffett Field, Calif. 94035**

and

**Computer Sciences Corporation
Mt. View, Calif. 94043**

February 1974

TABLE OF CONTENTS

1	INTRODUCTION	1
2	PROGRAM DESCRIPTION	2
3	DEFINITION OF COEFFICIENTS	4
4	USER'S INSTRUCTIONS	11
	4.1 INITIAL SETUP FOR AMES' TSS SYSTEM	11
	4.2 CONVERSATIONAL USE ON AMES' TSS SYSTEM	11
	4.3 AMES' TSS BATCH JOBS	14
	4.4 OTHER COMPUTERS	14
5	DESCRIPTION OF COMMANDS	15
	CONTINUE	15
	ECP	15
	ETA	17
	KNEW	18
	NETLOADS	18
	NEW	18
	PRESSURE	19
	PWEIGHTS	19
	SPANLOADS	19
	STOP	19
	TSS	19
	WEIGHTS	19
	XCP	20
6	SAMPLE TERMINAL SESSION	23
7	REFERENCES	28
	FIGURE	29
	APPENDIX I - OUTPUT FROM SAMPLE SESSION	30
	APPENDIX II - COMPUTER PROGRAM LISTING	55

NORMAL LOADS PROGRAM FOR
AERODYNAMIC LIFTING SURFACE THEORY

Richard T. Medan
Ames Research Center
and
K. Susan Ray
Computer Sciences Corporation

ABSTRACT

This document is a description of and users manual for a USA FORTRAN IV computer program which evaluates spanwise and chordwise loading distributions, lift coefficient, pitching moment coefficient, and other stability derivatives for thin wings in linearized, steady, subsonic flow. The program is based on a kernel function method lifting surface theory and is applicable to a large class of planforms including asymmetrical ones and ones with mixed straight and curved edges.

NORMAL LOADS PROGRAM FOR
AERODYNAMIC LIFTING SURFACE THEORY

Richard T. Medan
Ames Research Center
and
K. Susan Ray
Computer Sciences Corporation

1 INTRODUCTION

This document is a description of and users manual for a USA FORTRAN IV computer program which evaluates spanwise and chordwise loading distributions, lift coefficient, pitching moment coefficient, and other stability derivatives for thin wings in linearized, steady, subsonic flow. The program is based on a kernel function method lifting surface theory and is applicable to a large class of planforms including asymmetrical ones and ones with mixed straight and curved edges. This program is used in conjunction with other, separately documented programs which (1) set up the geometry (ref. 1), (2) set up the boundary conditions (ref. 2), (3) determine the aerodynamic influence matrix (ref. 3), and (4) solve a system of linear equations (ref. 4). The aerodynamic theory is described in ref. 5.

Questions concerning either this document or the computer program or the associated computer programs should be directed to

R. T. Medan
Mail Stop 247-1
Ames Research Center
Moffett Field,
Ca. 94035

NORMAL LOADS PROGRAM

2 PROGRAM DESCRIPTION

The normal loads program evaluates spanwise distributions of loading, lift coefficient, pitching moment, and center of pressure; the chordwise lifting pressure distribution; lift, induced drag, rolling moment, root bending moments, and pitching moment coefficients; and vortex drag factor, spanwise centers of pressure, and the lift on each half of the wing.

The program uses what is called command format programming. With this type of program the user himself controls the program flow calculating just what he needs and in the order that he wants to calculate it. For the most part the card data is entered in the order that the user chooses and always in a standard format (8F10.0 or 16I5). This makes the program very easy to use. A description of the available commands is given in section 5.

The program has commands allowing the user to select the spanwise and/or chordwise stations at which he wishes to evaluate any of the various distributions. The user can choose sets of default stations (no action required by the user) or he can enter various types of parameters causing the program to automatically compute sets of stations, or he can enter tables of stations manually.

Although some data is input from cards, the majority of the data that the program requires is stored on two disk or tape files: the geometry file and the solution file.

The geometry file contains all of the geometrical data such as the local chord distribution, the aspect ratio, and the longitudinal reference length. This file is created by the geometry program (ref. 1).

The other file is the solution file, which contains the coefficients in the expression for the lifting pressure coefficient. This file is created by the equation solving program (ref. 4). Several sets of coefficients may be contained on this file. For example the file may contain sets of coefficients corresponding to (1) angles of attack, (2) pitching about some axis (for computing quasi-steady pitching derivatives), (3) basic

NORMAL LOADS PROGRAM

camber and twist distribution, and (4) rolling (for computing quasi-steady rolling derivatives).

Each set of coefficients is termed a case. The program works with linear combinations of cases called combinations (because induced drag, for example, is nonlinear). The program uses a matrix of weights to determine the combinations from the cases. The columns of the matrix correspond to the combinations while the rows correspond to the cases. If the default option is chosen by the user, the weight matrix is computed automatically and is an identity matrix. Otherwise the user enters the weight matrix.

NORMAL LOADS PROGRAM

3 DEFINITION OF COEFFICIENTS

This section presents the definitions and expressions for the various coefficients calculated by the program. Figure 1 shows how the various geometrical quantities referred to below are defined for a yawed wing.

PRESSURE COEFFICIENT

$$\begin{aligned}\Delta C_p &= 2(P_l - P_u) / (\rho U_\infty^2) \\ &= 2b/c \sum_{N=1}^{NMAX} \sum_{K=1}^{KK} BNK(N,K) * HN(N,\theta) * \sin(K*\theta)\end{aligned}$$

where

- b = the effective span (see fig. 1),
- c = the local chord (a function of eta),
- eta = the nondimensional spanwise variable such that $-1 \leq \eta \leq +1$,
- θ = $-\cos^{-1} x$,
- x = the nondimensional local chordwise variable such that $-1 \leq x \leq 1$,
- θ = $+\cos^{-1} \eta$,

$$HN(1,\theta) = 2./PI * \cot(\theta/2.)$$

or

$$HN(N,\theta) = 2./PI * \sin((N-1)*\theta) \text{ for } N > 1,$$

and

$$PI = 3.141592\dots$$

NORMAL LOADS PROGRAM

SECTIONAL LIFT COEFFICIENT

$$\begin{aligned} CL &= .5 * \int_{-1}^{+1} \Delta C_p * dx \\ &= 2b/c \sum_{K=1}^{KK} \text{SIN}(K * \theta) * B(K) \end{aligned}$$

where
 $B(K) = \text{BNK}(1, K) + \text{BNK}(2, K) / 2.$

(B is denoted by STOR1 in the program.)

NONDIMENSIONAL CIRCULATION

$$\text{GAMMA} = CL * c / (2 * b)$$

$$= \sum_{K=1}^{KK} \text{SIN}(K * \theta) * B(K)$$

NORMALIZED LOADING

$$\text{CLCCLC} = CL * c / (\text{CCL} * \text{CAVG})$$

where
 $\text{CCL} = \text{overall lift coefficient}$

and
 $\text{CAVG} = \text{average chord}$

NORMAL LOADS PROGRAM

NONDIMENSIONAL PITCHING MOMENT

$$QMOMNT = 2 * cm(1/4) * c/b$$

$$= .5 * \sum_{K=1}^{KK} \sin(K * \theta) * (-BNK(2, K) + BNK(3, K))$$

where

$$cm(1/4) = -.25 * \int_{-1}^{+1} \Delta Cp * (x + .5) * dx$$

CENTER OF PRESSURE DISTRIBUTION

This center of pressure is normalized by the local chord and is measured from the local 1/4 chord.

$$\begin{aligned} COP &= -cm(1/4) / CL \\ &= -.25 * QMOMNT / GAMMA \end{aligned}$$

OVERALL LIFT COEFFICIENT

$$CCL = 1./S \iint_{WING} \Delta Cp * dS$$

$$\begin{aligned} &= AR / (4. * BRATIO ** 2) \int_{-1}^{+1} \int_{-1}^{+1} c * \Delta Cp * dx * d(\eta) \\ &= PI * AR / (2. * BRATIO ** 2) * B(1) \end{aligned}$$

where

S = wing reference area,

AR = aspect ratio

= 4. * BREF ** 2 / S,

NORMAL LOADS PROGRAM

BREF = lateral (spanwise) reference length
(usually $b/2$),

and
BRATIO = BREF/($b/2$)

INDUCED DRAG

The drag is calculated using a Trefftz plane analysis.

$$CDI = \pi \cdot AR / 4 \cdot \sum_{K=1}^{KK} K \cdot B(K)^2$$

VORTEX DRAG FACTOR

The vortex drag factor is the reciprocal of the ideal Oswald drag efficiency factor.

$$\begin{aligned} VORD &= \pi \cdot AR \cdot CDI / CCL^2 \\ &= 1 + 2 \cdot (B(2)/B(1))^2 \\ &\quad + 3 \cdot (B(3)/B(1))^2 \\ &\quad + 4 \cdot (B(4)/B(1))^2 + \dots \end{aligned}$$

ROLLING MOMENT

$$CMR = 1 / (S \cdot 2 \cdot BREF) \iint_{WING} \Delta C_p \cdot Y \cdot dS$$

where
Y = dimensional spanwise coordinate
= $b \cdot \eta / 2$.

NORMAL LOADS PROGRAM

RIGHT ROOT BENDING MOMENT

CMBP = the nondimensional moment required to hold the portion of the wing on $\eta \geq 0$ in equilibrium.

$$= 1./(\text{S} \cdot \text{BREF}) \iint_{\text{right half}} \Delta C_p \cdot Y \cdot dS$$

$$= \text{AR}/(4 \cdot \text{BRATIO} \cdot \cdot 3) \cdot (+\text{PI} \cdot \text{B}(2)/2 +$$

$$\sum_{\substack{K=1 \\ K \neq 2}}^{KK} \text{B}(K) \cdot (\text{SIN}((K-2) \cdot \text{PI}/2)/(K-2)$$

$$- \text{SIN}((K+2) \cdot \text{PI}/2)/(K+2))$$

LEFT ROOT BENDING MOMENT

CMBM = the nondimensional moment required to hold the portion of the wing on $\eta \leq 0$ in equilibrium.

$$= 1./(\text{S} \cdot \text{BREF}) \iint_{\text{left half}} \Delta C_p \cdot Y \cdot dS$$

$$= \text{AR}/(4 \cdot \text{BRATIO} \cdot \cdot 3) \cdot (-\text{PI} \cdot \text{B}(2)/2 +$$

$$\sum_{\substack{K=1 \\ K \neq 2}}^{KK} \text{B}(K) \cdot (\text{SIN}((K-2) \cdot \text{PI}/2)/(K-2)$$

$$- \text{SIN}((K+2) \cdot \text{PI}/2)/(K+2))$$

NORMAL LOADS PROGRAM

LIFT ON RIGHT SIDE OF WING

$$\begin{aligned} \text{CLP} &= 1./S \iint_{\text{right half}} \Delta C_p * dS \\ &= AR / (2 * \text{BRATIO} ** 2) * (PI * B(1) / 2. + \\ &\quad \sum_{K=2}^{KK} B(K) * (\text{SIN}((K-1) * PI / 2.) / (K-1) - \\ &\quad \quad \quad \text{SIN}((K+1) * PI / 2.) / (K+1))) \end{aligned}$$

LIFT ON LEFT SIDE OF WING

$$\begin{aligned} \text{CLM} &= 1/S \iint_{\text{left half}} \Delta C_p * dS \\ &= AR / (2 * \text{BRATIO} ** 2) * (PI * B(1) / 2. - \\ &\quad \sum_{K=2}^{KK} B(K) * (\text{SIN}((K-1) * PI / 2.) / (K-1) - \\ &\quad \quad \quad \text{SIN}((K+1) * PI / 2.) / (K+1))) \end{aligned}$$

LATERAL CENTER OF PRESSURE OF THE RIGHT HALF

- CPP = The spanwise center of pressure of the right half of the wing nondimensionalized by BREF
- = CMBP/CLP

NORMAL LOADS PROGRAM

LATERAL CENTER OF PRESSURE OF THE LEFT HALF

- CPM = The spanwise center of pressure of the left half of the wing nondimensionalized by BREF
- = -CMBM/CLM

PITCHING MOMENT

$$\text{CMP} = -1./(\text{S} \cdot \text{CBAR}) \iint_{\text{WING}} \Delta C_p \cdot X \cdot dS$$

where

- CBAR = The reference chord,
and
X = the dimensional chordwise coordinate.
 = $x_{si} = b/2$ (see fig. 1)

Although the above integral can be evaluated analytically in the chordwise direction, it can not be analytically evaluated spanwise. The spanwise integration is done using SUBROUTINE INTGRT. To allow the user to assess the convergence this integration is done repeatedly with more and more points. The maximum number of points used is the minimum of JJMAX and JJ. JJMAX is the maximum number of available integration points and is read from the geometry file while JJ is a number entered by the user.

CHORDWISE CENTER OF PRESSURE

- XCP = The chordwise center of pressure nondimensionalized by CBAR
- = -CMP/CCL.

NORMAL LOADS PROGRAM

4 USER'S INSTRUCTIONS

4.1 INITIAL SETUP FOR AMES' TSS SYSTEM

For either batch or conversational processing the following TSS commands must be given. These commands are required once and only once for each user ID. The first three commands create the identification number file named IDFILE. This file contains four zeroes in binary form.

```
SHARE MEDAN,FSARTM,INIDFILE
CDS MEDAN,IDFILE
DELETE MEDAN
SHARE MEDAN,FSARTM,LSPROG.V1
```

4.2 CONVERSATIONAL USE ON AMES' TSS SYSTEM

All integer data should be entered in a 1615 format, all floating point data in 8F10.0 format, and all logical data in 10L1 format.

USER: After logging on enter the following:

```
AMES USYSLIB
JOBLIBS SYSULIB
JBLB MEDAN
```

It is not necessary to issue DDEFs for anything except the input data since the program automatically issues them using the subroutines GEMFIL and BCFIL.

USER: CALL FORCE\$

PROG: ENTER BATCH

USER: Enter carriage return for conversational mode.

PROG: ENTER ODISK (NEG. HALTS)

USER: For terminal output enter carriage return. For output to a disk file enter a positive non-zero number less than 10. Such a disk file will be referred to as the output file. For the AMES' TSS system the output will be found on the file named OUTPUT.FOR.NX where X is the numerical

NORMAL LOADS PROGRAM

value of ODISK. The program issues its own DDEF commands so no control cards are needed. The value entered must be different from previous values for which the corresponding output datasets have not yet been printed. The program uses logical unit 4 for this output. If a negative value is entered, the program will terminate.

PROG: ENTER ID1, ID2, ID3, ID4

USER: Enter identification numbers

ID1 Identification number of the geometry file from which the AIM and BC files have been derived.

ID2 Identification number of AIM file.

ID3 Identification number of the boundary condition file.

ID4 Identification number of the solution file.

PROG: ENTER COMBINATION CODE

USER: Enter combination code (LCOMB) to define a new set of weights as described below. If the output is being placed on an output file (i.e. $1 \leq \text{ODISK} \leq 9$), then the weight matrix will be printed on the output file.

If $\text{LCOMB} > 0$, then LCOMB equals the number of combinations and the weights of each combination must be entered by the user. This data is prompted for and entered in the following fashion:

PROG: COMBINATION 1

PROG: ENTER WEIGHTS OF FIRST NSYM CASES

USER: Enter the set of weights. During execution the actual value of NSYM, which is the number of symmetric cases on the solution file, is inserted in the above message. After the above message is

NORMAL LOADS PROGRAM

given the user should enter the NSYM weights corresponding to the symmetric cases of combination 1.

PROG: ENTER WEIGHTS OF LAST NASYM CASES

USER: During execution the actual value of NASYM, which is the number of antisymmetric cases on the solution file, is inserted in the above message. After the above message is given the user should enter the NASYM weights corresponding to the antisymmetric cases of combination 1.

PROG: COMBINATION 2

PROG: ENTER WEIGHTS OF FIRST NSYM CASES

USER: Continue entering weights as above until the weights of all the combinations have been entered.

If LCOMB=0, then the number of combinations equals the number of cases (solutions) and each solution with a factor of 1.0 is treated as a combination, i.e., the weight matrix will be automatically computed as an identity matrix.

If LCOMB=-1, then the number of combinations equals the number of symmetric cases and each symmetric solution with a factor of 1.0 is treated as a combination, i.e., the weight matrix is an identity matrix with the last NASYM diagonal elements set to zero.

If LCOMB=-2, then the number of combinations equals the number of antisymmetric cases and each antisymmetric solution with a factor of 1.0 is treated as a combination, i.e. the weight matrix is an identity matrix with the first NSYM diagonal elements set to zero.

If LCOMB \leq -3, then the current set of weights will be used. Do not enter -3 unless a set of weights has been previously entered.

NORMAL LOADS PROGRAM

PROG: +

USER: At this point the user must begin entering commands. After each command he will be prompted for supplemental input or with another + sign indicating that he should enter another command. A full description of the commands is given in section 5. After a NEW or KNEW command the next input will be ODISK followed by ID1, ID2, ID3, and ID4 followed (for the NEW command) by LCOMB followed by more commands.

4.3 AMES' TSS BATCH JOBS

The batch mode operates the same as the conversational mode with the sole exception that a "T" must be put in column 1 on the first card. This "T" suppresses all subsequent conversational prompts.

4.4 OTHER COMPUTERS

Remove all calls to GEMFIL, BNKFIL, OBEY, and CVRT in the main program and SUBROUTINE SBLOAD and use appropriate tape or disk control cards in their place. These, hopefully, are the only changes that need to be made since considerable effort was made to code the program in standard FORTRAN. Then follow the instructions, where appropriate, in sections 4.2 and 4.3.

NORMAL LOADS PROGRAM

5 DESCRIPTION OF COMMANDS

A description of the commands, which control the flow of the program, is given in this section. In all cases the first three letters of a command are sufficient input. All integer data should be entered in a 1615 format and all floating point data in 8F10.0 format. The input, if any, associated with each command is to be entered on the following line in conversational processing and in batch processing is to be on cards immediately following the command card. Any input required is prompted for in the conversational mode. The same command may occur more than once. This is useful in correcting data entered in error.

The commands which are the most basic and useful are: SPANLOADS, which causes the program to compute and print the span loading information; NETLOADS, which causes the program to compute and print the overall force and moment data; PRESSURE, which causes the program to compute and print pressure coefficient distributions; NEW or KNEW, which cause the program to start a new case; and STOP, which terminates execution. These and all the other commands are fully explained below:

CONTINUE

EFFECT: Causes the program to continue execution in the batch mode even if an invalid command is encountered.

ECP

INPUT: NTYPE plus other input which varies with NTYPE. Valid values for NTYPE are 0, 1, 2, 3, and 5.

EFFECT: Defines spanwise (ETA) stations at which the chordwise pressure distributions will be computed and printed following a PRESSURE command. If this command is not given, the spanwise control points on the geometry file will be used. This command allows a direct comparison to be made with other theories and/or experimental data. The input required

NORMAL LOADS PROGRAM

for the various values of NTYPE and the spanwise stations defined thereby are given below:

INPUT FOR NTYPE=0: NONE

The following type of stations are prepared:

$$\text{ETA}=\text{COS}(I*\text{PI}/(\text{MREF}+1)) \text{ for } I=1,\text{MM}.$$

MREF comes from the solution file. The equation solving program transferred this number from the influence matrix file. The influence matrix program obtains this number either from the geometry file or as user input after the MREF command. If obtained from the geometry file, MREF will be equal to the geometry program variable NN.

INPUT FOR NTYPE=1: NQ

The following type of stations are prepared:

$$\text{ETA}=\text{COS}(I*\text{PI}/(\text{JJMAX}+1)) \text{ for } I=\text{NQ},\text{JJMAX},\text{NQ}$$

In the conversational mode the program prompts the user for the value of NQ. The maximum allowable value for NQ is equal to the value of NDIM3 in SUBROUTINE SBLOAD. Currently NDIM3=47. JJMAX is the number of spanwise stations at which the wing data is given. JJMAX is read from the geometry file.

INPUT FOR NTYPE=2: NSTA

The following type of stations are prepared:

$$\text{ETA}=\text{COS}(I*\text{PI}/(\text{NSTA}+1)) \text{ for } I=1,\text{NSTA}.$$

In the conversational mode the program prompts the user for the value of NSTA. The maximum allowable value for NSTA is

NORMAL LOADS PROGRAM

equal to the value of NDIM3 in SUBROUTINE SBLOAD. Currently NDIM3=47.

INPUT FOR NTYPE=3: a table of spanwise (ETA) stations

In the conversational mode the program prompts the user to enter the table. This table is to be entered one value per line or card. Values entered need not be in any particular order. The last entry must be followed by a line or card containing a number greater than 1.0 in order to signal the end of the table to the program. The allowable number of entries including the value greater than 1.0 is equal to the value of NDIM3 in SUBROUTINE SBLOAD. Currently NDIM3=47.

INPUT FOR NTYPE=5: ETMIN,ETMAX,DETA

The following type of stations are prepared:

ETA=ETMIN,ETMIN+DETA,ETMIN+2*DETA, ... , ETMAX.

ETMAX and DETA have default values. The default for DETA is ETMIN, while the default for ETMAX is 1.0. The user must insure that no more than NDIM3 stations are defined in this way. NDIM3 is a variable in SUBROUTINE SBLOAD and currently is equal to 47.

ETAS

INPUT: Same as for ECP command

EFFECT: Defines spanwise (ETA) stations at which the various spanwise loading distributions will be computed and printed following a SPANLOADS command. If this command is not given, the spanwise control points on the geometry file will be used. This command allows a direct comparison to be made with other theories and/or experimental data. Follow the

NORMAL LOADS PROGRAM

Instructions for the ECP command with the exception that the maximum number of stations is determined by NDIM1 in SUBROUTINE SBLOAD. Currently NDIM1=200.

KNEW

EFFECT: Performs the same function as NEW (see below) with the exception that the current stations for pressures and spanloads are retained.

NETLOADS

INPUT: JJ, NOUT

EFFECT: Computes and prints the following overall results: CCL (lift coefficient), CMP (pitch moment), CMR (roll moment), left and right root bending moments, left and right lifts, left and right centers of pressure, CDI, and the vortex drag factor. JJ is the maximum number of points to use in integrating for the pitching moment. The default for JJ is JJMAX, which comes from the geometry file. If JJ exceeds JJMAX the program will use only up to JJMAX points. NOUT is the output level for subroutine INIGRI (see listing). NOUT=0 is the usual choice.

NEW

EFFECT: Starts a new case. After this command is given the program returns to the point at which ODISK is requested (section 4.2). The value entered for ODISK must be different from previous values for which the corresponding output datasets have not been printed. Subsequently ID1, ID2, ID3, and ID4 are requested. Then, after the user enters the new set of identification numbers, the program reads a new solution file and, if necessary, a new geometry file and resets the spanwise stations at which the spanwise loading distributions are calculated and the spanwise and chordwise stations at which the lifting pressure distribution is calculated. Then the program requests the combination code (LCOMB)

NORMAL LOADS PROGRAM

and, if LCOMB>0, a new set of weights as described in section 4.2. Following this the program is in the command mode again.

PRESSURE

EFFECT: Computes and prints the lifting pressures at the chordwise locations defined by the XCP command and at the spanwise locations defined by the ECP command. If the XCP or ECP commands have not been given, then the chordwise and spanwise stations used will be the spanwise control points.

PWEIGHTS

INPUT: NONE

EFFECT: Prints the current weights matrix on the user's terminal. This command is generally only used in the conversational mode.

SPANLOADS

EFFECT: Prints the loading distributions at the spanwise stations defined by the ETAS command. If the ETAS command has not been given, then the spanwise control points will be used.

STOP

EFFECT: Halts execution.

TSS

INPUT: A TSS command of 80 characters or less.

EFFECT: The command is passed to the AMES' TSS operating system. After the system processes the command, control returns to the program. This command is a special one for the AMES' TSS version of the program.

WEIGHTS

INPUT: LCOMB and, if LCOMB>0, a new set of weights.

NORMAL LOADS PROGRAM

EFFECT: Allows the user to define a new set of weights in the same manner as when starting a new case (see section 4.2).

XCP

INPUT: NTYPE plus other input which varies with NTYPE

EFFECT: NTYPE defines the chordwise (x) stations at which the lifting pressures will be computed following a PRESSURE command. The quantity x is the local chordwise variable such that $-1 \leq x \leq +1$. If this command is not given, the stations will be identical to the spanwise control points on the geometry file (using the correspondence $x = \eta$). This command allows a direct comparison to be made with other theories and/or experimental data. The input required for the various values of NTYPE and the chordwise stations defined thereby are given below:

INPUT FOR NTYPE=0: NONE

The following type of stations are prepared:

$$x = \cos(i \cdot \pi / (MREF + 1)) \text{ for } i = 1, MM.$$

MREF comes from the solution file. The equation solving program transferred this number from the influence matrix file. The influence matrix program obtains this number either from the geometry file or as user input after the MREF command. If obtained from the geometry file, MREF will be equal to the geometry program variable NN.

INPUT FOR NTYPE=1: NQ

The following type of stations are prepared:

$$x = \cos(i \cdot \pi / (JJMAX + 1)) \text{ for } i = NQ, JJMAX, NQ.$$

In the conversational mode the program

NORMAL LOADS PROGRAM

prompts the user for the value of NQ. The maximum allowable value for NQ is equal to the value of NDIM2 in SUBROUTINE SBLOAD. Currently NDIM2=100. JJMAX is the number of spanwise stations at which the wing data is given. JJMAX is read from the geometry file.

INPUT FOR NTYPE=2: NSTA

The following type of stations are prepared:

$$x = \cos(i \cdot \pi / (NSTA + 1)) \text{ for } i = 1, NSTA.$$

In the conversational mode the program prompts the user for the value of NSTA. The maximum allowable value for NSTA is equal to the value of NDIM2 in SUBROUTINE SBLOAD. Currently NDIM2=100.

INPUT FOR NTYPE=3: table of values of x

In the conversational mode the program prompts the user to enter the table. This table is to be entered one value per line or card. Values entered need not be in any particular order. The last entry must be followed by a line or card containing a number greater than 1.0 in order to signal the end of the table to the program. The allowable number of entries including the value greater than 1.0 is equal to the value of NDIM2 in SUBROUTINE SBLOAD. Currently NDIM2=100.

INPUT FOR NTYPE=4: table of values of CHI (CHI=(x+1)/2.)

In the conversational mode the program prompts the user to enter the table. This table is to be entered one value per line or card. Values entered need not be in any particular order. The last entry must be followed by a line or card containing a number greater than 1.0 in order to signal the end of the table to

NORMAL LOADS PROGRAM

the program. The allowable number of entries including the value greater than 1.0 is equal to the value of NDIM2 in SUBROUTINE SBLOAD. Currently NDIM2=100.

INPUT FOR NTYPE=5: XMIN,XMAX,DX

The following type of stations are prepared:

$x = XMIN, XMIN+DX, XMIN+2*DX, \dots, XMAX.$

XMAX and DX have default values. The default for XMAX is 1.0, while the default for DX is XMIN-(-1.). The user must insure that no more than NDIM2 stations are defined in this way. NDIM2 is a variable in SUBROUTINE SBLOAD and currently is equal to 100.

NORMAL LOADS PROGRAM

6 SAMPLE TERMINAL SESSION

A sample conversational terminal session on the Ames' 360/67 TSS computer system is reproduced in this section with additional comments added in parentheses. During this session the 5 sets of solutions obtained in the sample case of the equation solving program (ref. 4) were utilized. These sets of solutions were for an aspect ratio 2, rectangular wing and differed from one another only in the number of control points and/or pressure modes used.

The first set (ID4=15) was first used to illustrate the simplest treatment, i.e., only the NETLOADS and SPANLOADS commands. The default weight matrix and default spanwise stations (the spanwise control points) were used. Next the use of the WEIGHTS command is illustrated. Then the chordwise and spanwise stations for the lifting pressure distribution were set up and the pressure distribution obtained. These stations were chosen to be identical to the stations used in ref. 6 so that a direct comparison among the 4 theories could be made. Then the spanwise stations used in ref. 6 for the spanwise load distributions were set up (NSTA = 15 = the number of spanwise control points used in ref. 6) and the spanwise loading was obtained. This is all that was done with the first solution set.

For the remaining 4 solution sets the various stations existing at the end of the first solution set and the weights matrix were retained (using the KNEW command and LCOMB=-3) in order to facilitate a direct comparison among the sets of results obtained using the author's computer programs and those obtained in ref. 6. For each of these sets net loads, span load distributions, and pressures were obtained.

The output from the following test run was directed to disk files, which were later printed and are given in appendix 1:

```
LOGON userid,password,terminal id
AMES USYSLIB
JOBLIBS SYSULIB
JBLB MEDAN
DDNAME=JBLB0001
CALL FORCE$
```

NORMAL LOADS PROGRAM

```
ENTER BATCH
(carriage return)
ENTER ODISK (NEG. HALTS )
  1
OUTPUT IS ON   OUTPUT.FOR.N1
  CANCELLED: DDNAME FT04F001 UNKNOWN
(Messages such as the above occur because of the
automatic file defining feature of the Ames' version.
They do not indicate any error.)
ENTER ID1, ID2, ID3, ID4
  4      2      9      15
  CANCELLED: DDNAME FT07F001 UNKNOWN
  CANCELLED: DDNAME FT12F001 UNKNOWN
ENTER COMBINATION CODE
  0
+
TSS
CPU TIME?
  3.796 SECONDS
+
NETLOADS
ENTER JJ AND NOUT
  0      0
+
SPANLOADS
+
WEIGHTS
ENTER COMBINATION CODE
  1
COMBINATION  1
ENTER WEIGHTS OF FIRST  2 CASES
1.          1.
ENTER WEIGHTS OF  LAST  1 CASES
1.
+
SPANLOADS
+
NETLOADS
ENTER JJ AND NOUT
  50
+
ECP
ENTER NTYPE
  3
ENTER TABLE OF ETAS--1 PER LINE ENDING WITH VALUE
GREATER THAN 1
```

NORMAL LOADS PROGRAM

0.
0.3827
0.7071
0.9239
1
+
XCP
ENTER NTYPE
4
ENTER TABLE OF CHI VALUES
0.005
0.0125
0.025
0.05
0.1
0.15
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
0.95
1
+
WEIGHTS
ENTER COMBINATION CODE
1
COMBINATION 1

ENTER WEIGHTS OF FIRST 2 CASES
1.
ENTER WEIGHTS OF LAST 1 CASES
0.
+
PRESSURES
+
ETAS
ENTER NTYPE
2
ENTER NSTA
15
+
SPANLOADS
+

NORMAL LOADS PROGRAM

KNEW
ENTER ODISK (NEG.HALTS)
2
OUTPUT IS ON OUTPUT.FOR.N2
ENTER ID1, ID2, ID3, ID4
4 2 9 16
ENTER COMBINATION CODE
-3

+
NETLOADS
ENTER JJ AND NOUT
50

+
SPANLOADS

+
PRESSURES

+
KNEW
ENTER ODISK (NEG.HALTS)
3
OUTPUT IS ON OUTPUT.FOR.N3
ENTER ID1, ID2, ID3, ID4
4 2 9 17
ENTER COMBINATION CODE
-3

+
NETLOADS
ENTER JJ AND NOUT
50

+
SPANLOADS

+
PRESSURES

+
KNEW
ENTER ODISK (NEG.HALTS)
4
OUTPUT IS ON OUTPUT.FOR.N4
ENTER ID1, ID2, ID3, ID4
4 2 9 18
ENTER COMBINATION CODE
-3

+
NETLOADS
ENTER JJ AND NOUT
50

+

NORMAL LOADS PROGRAM

```
SPANLOADS
+
PRESSURES
+
KNEW
ENTER ODISK (NEG,HALTS )
  5
OUTPUT IS ON   OUTPUT.FOR.N5
ENTER ID1,ID2,ID3,ID4
  4   2   9   19
ENTER COMBINATION CODE
  -3
+
NETLOADS
ENTER JJ AND NOUT
  50
+
SPANLOADS
+
PRESSURES
+
TSS
CPUTIME?
  17.438 SECONDS
+
STOP
  TERMINATED: STOP
PRINT OUTPUT.FOR.N1,PRTSP=EDIT,STATION=RMT05
PRINT BSN=9132,      200 LINES
PRINT OUTPUT.FOR.N2,PRTSP=EDIT,STATION=RMT05
PRINT BSN=9133,      100 LINES
PRINT OUTPUT.FOR.N3,PRTSP=EDIT,STATION=RMT05
PRINT BSN=9134,      100 LINES
PRINT OUTPUT.FOR.N4,PRTSP=EDIT,STATION=RMT05
PRINT BSN=9135,      100 LINES
PRINT OUTPUT.FOR.N5,PRTSP=EDIT,STATION=RMT05
PRINT BSN=9136,      100 LINES
LOGOFF
```

NORMAL LOADS PROGRAM

7 REFERENCES

1. Medan, R. T.: Geometry Program for Aerodynamic Lifting Surface Theory. NASA Rept. No. TMX-62,309, Sept. 1973.
2. Medan, R. T., and Ray, K. S.: Boundary Condition Program for Aerodynamic Lifting Surface Theory, NASA Rept. No. TMX-62,323, Dec. 1973.
3. Medan, R. T., and Ray, K. S.: Influence Matrix Program for Aerodynamic Lifting Surface Theory, NASA Rept. No. TMX-62,324, Dec. 1973.
4. Medan, R. T., and Lemmer, O. J.: Equation Solving Program for Aerodynamic Lifting Surface Theory, NASA Rept. No. TMX-62,325, Jan. 1974.
5. Medan, R. T.: Improvements to the Kernel Function Method of Steady, Subsonic Lifting Surface Theory. NASA Rept. No. TMX-62,327, Mar. 1974.
6. Garner, H. C.; Hewitt, B. L.; Labrujere, T. E.: Comparison of Three Methods for the Evaluation of Subsonic Lifting Surface Theory. A.R.C. R.&M. 3597, June 1968.

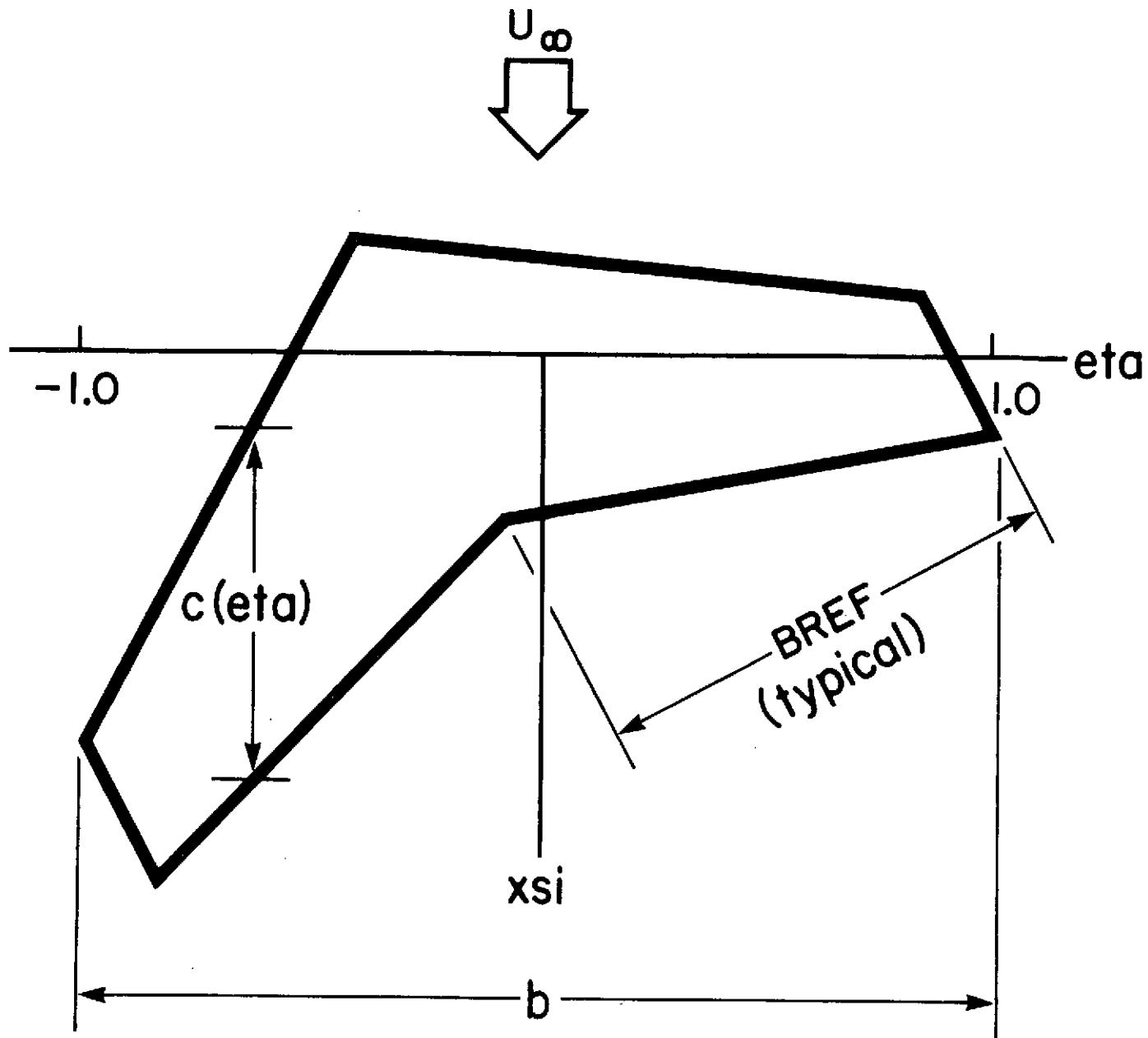


Fig. 1 - Definition of geometrical parameters.

APPENDIX I

OUTPUT FROM SAMPLE SESSION

LOADS ON THIN, LIFTING WING

RECTANGULAR WING AR = 2 11-13-73

ID1 = 4
ID2 = 2
ID3 = 9
ID4 = 15
DELTA0 = 4.0000
EPS = 0.5000
MACH = 0.0000
JJ = 191
JJMAX = 191
NMAX2 = 5
KK2 = 11
PPNEW = 5
MMNEW = 11
NROWSA = 30
CWTYP = 0
SWTYP = 0
BCS = YTTTTTTTT
BCAS = FTTTTTTTT

MACH = 0.0000
(B/2)/BREF = 1.0000
CBAR/BREF = 1.0000
ASPECT RATIO = 2.0000

WEIGHTS

COMBINATION 1
 1.000000 0.000000 0.000000

COMBINATION 2
 0.000000 1.000000 0.000000

COMBINATION 3
 0.000000 0.000000 1.000000

COMBINATION 1

CL = 2.474174
 INDUCED DRAG = 0.974904
 VORTEX DRAG FACTOR = 1.000646
 ROLLING MOMENT $M/(Q*S*2*BREF)$ = 0.000000
 RIGHT ROOT BENDING MOMENT $MBR/(Q*S*BREF)$ = 0.529675
 LEFT ROOT BENDING MOMENT $MBL/(Q*S*BREF)$ = 0.529675
 LIFT ON ETA,GT,0 = 1.237088
 LIFT ON ETA,LT,0 = 1.237088
 CENTER OF PRESSURE OF RIGHT HALF, Y/BREF = 0.428163
 CENTER OF PRESSURE OF LEFT HALF, Y/BRFF = -0.428163
 PITCHING MOMENT ABOUT X = 0 AND CENTER OF PRESSURE

J	CM/(Q*S*CBAR)	X(C,P,)/CBAR
11	=0.518157	0.209426
23	=0.518157	0.209426
47	=0.518157	0.209426
95	=0.518157	0.209426
191	=0.518157	0.209426

COMBINATION 2

CL = 1,956263
 INDUCED DRAG = 0,610638
 VORTEX DRAG FACTOR = 1,002557
 ROLLING MOMENT $M/(Q*S*2*BREF)$ = 0,000000
 RIGHT ROOT BENDING MOMENT $MBR/(Q*S*BREF)$ = 0,422256
 LEFT ROOT BENDING MOMENT $MBL/(Q*S*BREF)$ = 0,422256
 LIFT ON ETA,GT,0 = 0,978132
 LIFT ON ETA,LT,0 = 0,978132
 CENTER OF PRESSURE OF RIGHT HALF, Y/BREF = 0,431697
 CENTER OF PRESSURE OF LEFT HALF, Y/BREF = -0,431697
 PITCHING MOMENT ABOUT X = 0 AND CENTER OF PRESSURE

J	CM/(Q*S*CBAR)	X(C,P.)/CBAR
11	-0,756382	0,386646
23	-0,756382	0,386646
47	-0,756382	0,386646
95	-0,756382	0,386646
191	-0,756382	0,386646

COMBINATION 3

CL = 0,000000
 INDUCED DRAG = 0,183247
 ROLLING MOMENT $M/(Q*S*2*BREF)$ = 0,189680
 RIGHT ROOT BENDING MOMENT $MBR/(Q*S*BREF)$ = 0,189680
 LEFT ROOT BENDING MOMENT $MBL/(Q*S*BREF)$ = -0,189680
 LIFT ON ETA,GT,0 = 0,321319
 LIFT ON ETA,LT,0 = 0,321319
 CENTER OF PRESSURE OF RIGHT HALF, Y/BREF = 0,590318
 CENTER OF PRESSURE OF LEFT HALF, Y/BREF = -0,590318
 PITCHING MOMENT ABOUT X = 0 AND CENTER OF PRESSURE

J	CM/(Q*S*CBAR)	X(C,P.)/CBAR
11	0,000000	
23	0,000000	
47	0,000000	
95	0,000000	
191	0,000000	

ETA	CL*C/2R	CL	CL*C/CCL*CAVG	2CM(1/4)C/B	C.P.(1/4)
COMBINATION 1					
0,965926	0,211735	0,846941	0,342312	0,067269	=0,079426
0,866025	0,405291	1,621162	0,655233	0,105695	=0,065197
0,707107	0,565220	2,260880	0,913791	0,113871	=0,050366
0,500000	0,682165	2,728661	1,102856	0,106477	=0,039022
0,258819	0,752493	3,009970	1,216554	0,097121	=0,032266
0,000000	0,775844	3,103375	1,254306	0,093296	=0,030063

COMBINATION 2					
0,965926	0,175510	0,702039	0,358867	=0,109422	0,155863
0,866025	0,329908	1,319633	0,674568	=0,196098	0,148601
0,707107	0,452074	1,808294	0,924361	=0,255870	0,141498
0,500000	0,538337	2,153348	1,100744	=0,292654	0,135907
0,258819	0,589050	2,356199	1,204437	=0,311984	0,132410
0,000000	0,605716	2,422864	1,238515	=0,317951	0,131230

COMBINATION 3					
0,965926	0,121608	0,486431	=0,000000	0,066301	=0,136301
0,866025	0,210255	0,841018	=0,000000	0,099890	=0,118773
0,707107	0,241649	0,966596	=0,000000	0,096968	=0,100319
0,500000	0,208073	0,832293	=0,000000	0,071447	=0,085844
0,258819	0,119579	0,478316	=0,000000	0,036796	=0,076927
0,000000	0,000000	0,000000	=0,000000	0,000000	=0,077357

WEIGHTS

COMBINATION 1		
1,000000	1,000000	1,000000

ETA	CL*C/2R	CL	CL*C/CCL*CAVG	2CM(1/4)C/B	C,P.(1/4)
COMBINATION 1					
0,965926	0,508853	2,035412	0,459415	0,024149	=0,011864
0,866025	0,945453	3,781812	0,853597	0,009486	=0,002508
0,707107	1,258936	5,035744	1,136623	=0,045031	0,008942
0,500000	1,428568	5,714272	1,289774	=0,114729	0,020078
0,258819	1,461112	5,844448	1,319157	=0,178067	0,030468
0,000000	1,381555	5,526218	1,247328	=0,224656	0,040653
=0,258819	1,221951	4,887802	1,103230	=0,251658	0,051487
=0,500000	1,012422	4,049686	0,914059	=0,257624	0,063616
=0,707107	0,775645	3,102578	0,700287	=0,238967	0,077022
=0,866025	0,524944	2,099776	0,473943	=0,190293	0,090626
=0,965926	0,265637	1,062547	0,239829	=0,108454	0,102069

COMBINATION 1					

CL	=	4,430435			
INDUCED DRAG	=	3,311415			
VORTEX DRAG FACTOR	=	1,059986			
ROLLING MOMENT M/(Q*S*2*BREF)	=		0,189680		
RIGHT ROOT BENDING MOMENT MBR/(Q*S*BREF)	=		1,141611		
LEFT ROOT BENDING MOMENT MBL/(Q*S*BREF)	=		0,762251		
LIFT ON ETA,GT,0	=	2,536537			
LIFT ON ETA,LT,0	=	1,893900			
CENTER OF PRESSURE OF RIGHT HALF, Y/BREF	=		0,450067		
CENTER OF PRESSURE OF LEFT HALF, Y/BREF	=		=0,402477		
PITCHING MOMENT ABOUT X = 0 AND CENTER OF PRESSURE					
J	CM/(Q*S*CBAR)		X(C,P.)/CBAR		
11	=-1,274535		0,287677		
23	=-1,274535		0,287677		
47	=-1,274535		0,287677		

WEIGHTS

COMBINATION 1
1,000000 0,000000 0,000000

COMBINATION 1

ETA = 0,000000

X	CHI	DELTA=CP
=0,990000	0,005000	31,516403
=0,975000	0,012500	19,775360
=0,950000	0,025000	13,798651
=0,900000	0,050000	9,498217
=0,800000	0,100000	6,356733
=0,700000	0,150000	4,904237
=0,600000	0,200000	4,006256
=0,400000	0,300000	2,894391
=0,200000	0,400000	2,198984
=0,000000	0,500000	1,706087
0,200000	0,600000	1,328721
0,400000	0,700000	1,020911
0,600000	0,800000	0,751148
0,800000	0,900000	0,485080
0,900000	0,950000	0,329260

ETA = 0.382700

X	CHI	DELTA-CP
0.990000	0.005000	30.069534
0.975000	0.012500	18.851593
0.950000	0.025000	13.135118
0.900000	0.050000	9.014387
0.800000	0.100000	5.994587
0.700000	0.150000	4.594013
0.600000	0.200000	3.727465
0.400000	0.300000	2.658195
0.200000	0.400000	1.997470
0.000000	0.500000	1.537664
0.200000	0.600000	1.192542
0.400000	0.700000	0.915312
0.600000	0.800000	0.673715
0.800000	0.900000	0.434571
0.900000	0.950000	0.294249

ETA = 0.707100

X	CHI	DELTA-CP
0.990000	0.005000	25.631668
0.975000	0.012500	15.983688
0.950000	0.025000	11.039835
0.900000	0.050000	7.449440
0.800000	0.100000	4.800804
0.700000	0.150000	3.577179
0.600000	0.200000	2.831459
0.400000	0.300000	1.941271
0.200000	0.400000	1.421081
0.000000	0.500000	1.078485
0.200000	0.600000	0.832476
0.400000	0.700000	0.639926
0.600000	0.800000	0.472996
0.800000	0.900000	0.306167
0.900000	0.950000	0.207428

ETA = 0.923900

X	CHI	DELTA-CP
0.990000	0.005000	17.776794
0.975000	0.012500	10.747533
0.950000	0.025000	7.059169
0.900000	0.050000	4.333611
0.800000	0.100000	2.389219
0.700000	0.150000	1.614014
0.600000	0.200000	1.234967
0.400000	0.300000	0.912861
0.200000	0.400000	0.759190
0.000000	0.500000	0.611839
0.200000	0.600000	0.445402
0.400000	0.700000	0.286298
0.600000	0.800000	0.174114
0.800000	0.900000	0.127857
0.900000	0.950000	0.113100

38

ETA	CL+C/2B	CL	CL+C/(CL+C*AVG)	2CM(1/4)C/B	C.P.(1/4)
COMBINATION	1				
0.980785	0.159810	0.639242	0.258365	0.052435	-0.082027
0.923880	0.311866	1.247465	0.504194	0.090829	-0.072811
0.831470	0.448907	1.795628	0.725747	0.110096	-0.061313
0.707107	0.565220	2.260880	0.913791	0.113871	-0.050366
0.555570	0.657235	2.628938	1.062550	0.108945	-0.041441
0.382683	0.723222	2.892888	1.169231	0.101427	-0.035061
0.195090	0.762717	3.050869	1.233084	0.095487	-0.031298
0.000000	0.775844	3.103375	1.254306	0.093296	-0.030063

LOADS ON THIN, LIFTING WING

RECTANGULAR WING AR = 2 11-13-73

ID1 = 4
ID2 = 2
ID3 = 9
ID4 = 16
DELTA0 = 4.0000
EPS = 0.5000
MACH = 0.0000
JJ = 191
JJMAX = 191
NMAX2 = 5
KK2 = 5
PPNEW = 5
MMNEW = 11
NROWSA = 30
CWTYPE = 0
SWTYPE = 0
BCS = TTTTTTTTTT
BCAS = FTTTTTTTTT

MACH = 0.0000
(B/2)/BREF = 1.0000
CBAR/BREF = 1.0000
ASPECT RATIO = 2.0000

WFIGHTS

COMBINATION 1
 1.000000 0.000000 0.000000

COMBINATION 1

 CL = 2.471549
 INDUCED DRAG = 0.972838
 VORTEX DRAG FACTOR = 1.000648
 ROLLING MOMENT $M/(Q*S*\rho*BREF)$ = 0.000000
 RIGHT ROOT BENDING MOMENT $MBR/(Q*S*BREF)$ = 0.529120
 LEFT ROOT BENDING MOMENT $ML/(Q*S*BREF)$ = 0.529120
 LIFT ON ETA,GT,0 = 1.235776
 LIFT ON ETA,LT,0 = 1.235776
 CENTER OF PRESSURE OF RIGHT HALF, Y/BREF = 0.428168
 CENTER OF PRESSURE OF LEFT HALF, Y/BREF = -0.428168
 PITCHING MOMENT ABOUT X = 0 AND CENTER OF PRESSURE

J	CM/(Q*S*CBAR)	X(C.P.)/CHAR
11	=0.518407	0.209750
23	=0.518407	0.209750
47	=0.518407	0.209750

40

ETA CL*C/2R CL CL*C/CCL*CAVG 2CM(1/4)C/R C.P.(1/4)

COMBINATION 1

0.980785	0.159771	0.639085	0.258576	0.052376	=0.081954
0.923880	0.311998	1.246393	0.504296	0.091260	=0.073219
0.831470	0.448387	1.793549	0.725677	0.110284	=0.061489
0.707107	0.564585	2.258339	0.913733	0.112395	=0.049769
0.555570	0.656547	2.626188	1.062566	0.106105	=0.040403
0.382683	0.722470	2.889881	1.169257	0.099297	=0.034360
0.195090	0.761899	3.047595	1.233068	0.095447	=0.031319
0.000000	0.774995	3.099979	1.254264	0.094374	=0.030443

COMBINATION 1

ETA = 0.000000

X	CHI	DELTA=CP
0.990000	0.005000	31.872513
0.975000	0.012500	19.950165
0.950000	0.025000	13.868890
0.900000	0.050000	9.487086
0.800000	0.100000	6.297503
0.700000	0.150000	4.842047
0.600000	0.200000	3.956280
0.400000	0.300000	2.877397
0.200000	0.400000	2.204339
0.000000	0.500000	1.717242
0.200000	0.600000	1.333607
0.400000	0.700000	1.016149
0.600000	0.800000	0.742330
0.800000	0.900000	0.482747
0.900000	0.950000	0.332425

ETA = 0.382700

X	CHI	DELTA=CP
0.990000	0.005000	29.532990
0.975000	0.012500	18.578003
0.950000	0.025000	13.011273
0.900000	0.050000	9.006115
0.800000	0.100000	6.055693
0.700000	0.150000	4.661861
0.600000	0.200000	3.781092
0.400000	0.300000	2.670871
0.200000	0.400000	1.982543
0.000000	0.500000	1.516855
0.200000	0.600000	1.182024
0.400000	0.700000	0.919981
0.600000	0.800000	0.686526
0.800000	0.900000	0.441180
0.900000	0.950000	0.293837

ETA = 0.707100

X	CHI	DELTA=CP
=0.990000	0.005000	25.732437
=0.975000	0.012500	16.017380
=0.950000	0.025000	11.032462
=0.900000	0.050000	7.410202
=0.800000	0.100000	4.748520
=0.700000	0.150000	3.533155
=0.600000	0.200000	2.802224
=0.400000	0.300000	1.941195
=0.200000	0.400000	1.437692
=0.000000	0.500000	1.097311
0.200000	0.600000	0.843628
0.400000	0.700000	0.640770
0.600000	0.800000	0.468129
0.800000	0.900000	0.304315
0.900000	0.950000	0.209267

ETA = 0.923900

X	CHI	DELTA=CP
=0.990000	0.005000	17.889465
=0.975000	0.012500	10.809661
=0.950000	0.025000	7.092859
=0.900000	0.050000	4.344326
=0.800000	0.100000	2.381964
=0.700000	0.150000	1.599733
=0.600000	0.200000	1.218279
=0.400000	0.300000	0.898071
=0.200000	0.400000	0.750157
=0.000000	0.500000	0.609106
0.200000	0.600000	0.447497
0.400000	0.700000	0.290539
0.600000	0.800000	0.177410
0.800000	0.900000	0.127898
0.900000	0.950000	0.111598

LOADS ON THIN, LIFTING WING

=====

RECTANGULAR WING AR = 2 11-13-73

ID1 = 4
ID2 = 2
ID3 = 9
ID4 = 17
DELTA0 = 4.0000
EPS = 0.5000
MACH = 0.0000
JJ = 191
JJMAX = 191
NMAX2 = 5
KK2 = 5
PPNEW = 5
MMNEW = 5
NRWSA = 15
CNTYPE = 0
SWTYPE = 0
BCS = YTTTTTTTTT
BCAS = FTTTTTTTTT

MACH = 0.0000
(B/2)/BREF = 1.0000
CBAR/BREF = 1.0000
ASPECT RATIO = 2.0000

WEIGHTS

COMBINATION 1
 1.000000 0.000000 0.000000

COMBINATION 1

```

-----
CL = 2.474199
INDUCED DRAG = 0.974926
VORTEX DRAG FACTOR = 1.000648
ROLLING MOMENT M/(Q*S*2*BREF) = 0.000000
RIGHT ROOT BENDING MOMENT MBR/(Q*S*BREF) = 0.529688
LEFT ROOT BENDING MOMENT ML/(Q*S*BREF) = 0.529688
LIFT ON ETA,GT,0 = 1.237101
LIFT ON ETA,LT,0 = 1.237101
CENTER OF PRESSURE OF RIGHT HALF, Y/BREF = 0.428168
CENTER OF PRESSURE OF LEFT HALF, Y/BREF = -0.428168
PITCHING MOMENT ABOUT X = 0 AND CENTER OF PRESSURE
      J          CM/(Q*S*CBAR)          X(C.P.)/CBAR
11          =0.518180          0.209434
23          =0.518180          0.209434
47          =0.518180          0.209434
  
```

ETA	CL+C/2R	CL	CL+C/CCL*CAVG	2CM(1/4)C/B	C,P.(1/4)
COMBINATION 1					

0.980785	0.159972	0.639886	0.258623	0.050629	=0.079122
0.923880	0.311967	1.247869	0.504352	0.089493	=0.071716
0.831470	0.448881	1.795525	0.725698	0.110412	=0.061493
0.707107	0.565174	2.260697	0.913707	0.114814	=0.050787
0.555570	0.657223	2.628892	1.062521	0.109328	=0.041587
0.382683	0.723231	2.892922	1.169234	0.101255	=0.035001
0.195090	0.762726	3.050904	1.233086	0.095298	=0.031236
0.000000	0.775848	3.103393	1.254300	0.093217	=0.030037

COMBINATION 1

ETA = 0.000000

X	CHI	DELTA=CP
=0.990000	0.005000	31.509399
=0.975000	0.012500	19.771072
=0.950000	0.025000	13.795799
=0.900000	0.050000	9.496536
=0.800000	0.100000	6.356160
=0.700000	0.150000	4.904328
=0.600000	0.200000	4.006797
=0.400000	0.300000	2.895379
=0.200000	0.400000	2.199949
=0.000000	0.500000	1.706737
0.200000	0.600000	1.328949
0.400000	0.700000	1.020793
0.600000	0.800000	0.750904
0.800000	0.900000	0.484978
0.900000	0.950000	0.329284

ETA = 0.382700

X	CHI	DELTA=CP
=0.990000	0.005000	30.023071
=0.975000	0.012500	18.827209
=0.950000	0.025000	13.123264
=0.900000	0.050000	9.012363
=0.800000	0.100000	5.999108
=0.700000	0.150000	4.599978
=0.600000	0.200000	3.733028
=0.400000	0.300000	2.661237
=0.200000	0.400000	1.998137
=0.000000	0.500000	1.537046
0.200000	0.600000	1.191726
0.400000	0.700000	0.914943
0.600000	0.800000	0.673855
0.800000	0.900000	0.434792
0.900000	0.950000	0.294315

ETA = 0.707100

X	CHI	DELTA=CP
0.990000	0.005000	25.943954
0.975000	0.012500	16.140030
0.950000	0.025000	11.106818
0.900000	0.050000	7.447138
0.800000	0.100000	4.757178
0.700000	0.150000	3.530241
0.600000	0.200000	2.793994
0.400000	0.300000	1.930276
0.200000	0.400000	1.428168
0.000000	0.500000	1.090131
0.200000	0.600000	0.838607
0.400000	0.700000	0.637441
0.600000	0.800000	0.466127
0.800000	0.900000	0.303472
0.900000	0.950000	0.208941

ETA = 0.923900

X	CHI	DELTA=CP
0.990000	0.005000	16.622696
0.975000	0.012500	10.200644
0.950000	0.025000	6.864388
0.900000	0.050000	4.409828
0.800000	0.100000	2.611207
0.700000	0.150000	1.825129
0.600000	0.200000	1.385116
0.400000	0.300000	0.924651
0.200000	0.400000	0.689179
0.000000	0.500000	0.533364
0.200000	0.600000	0.408714
0.400000	0.700000	0.302620
0.600000	0.800000	0.215505
0.800000	0.900000	0.144678
0.900000	0.950000	0.105740

LOADS ON THIN, LIFTING WING

=====

RECTANGULAR WING AR = 2 11-13-73

ID1 = 4
ID2 = 2
ID3 = 9
ID4 = 18
DELTA0 = 4.0000
EPS = 0.5000
MACH = 0.0000
JJ = 191
JJMAX = 191
NMAX2 = 3
KK2 = 5
PPNEW = 3
MMNEW = 5
NROWSA = 9
CWTYPE = 0
SWTYPE = 0
BCS = TTTTTTTTTT
BCAS = FTTTTTTTTT

MACH = 0.0000
(B/2)/BREF = 1.0000
CBAR/BREF = 1.0000
ASPECT RATIO = 2.0000

OUTPUT.FOR.M4

WEIGHTS

COMBINATION 1
 1.000000 0.000000 0.000000

COMBINATION 1

CL = 2.469037
 INDUCED DRAG = 0.970694
 VORTEX DRAG FACTOR = 1.000476
 ROLLING MOMENT $M/(Q*S^2*BREF)$ = 0.000000
 RIGHT ROOT BENDING MOMENT $MBR/(Q*S*BREF)$ = 0.527946
 LEFT ROOT BENDING MOMENT $MBL/(Q*S*BREF)$ = 0.527946
 LIFT ON ETA,GT,0 = 1.234519
 LIFT ON ETA,LT,0 = 1.234519
 CENTER OF PRESSURE OF RIGHT HALF, Y/BREF = 0.427653
 CENTER OF PRESSURE OF LEFT HALF, Y/BREF = -0.427653
 PITCHING MOMENT ABOUT X = 0 AND CENTER OF PRESSURE

J	CM/(Q*S*CBAR)	X(C,P.)/CBAR
11	=0.512238	0.207465
23	=0.512238	0.207465
47	=0.512238	0.207465

ETA	CL*C/2R	CL	CL*C/CCL*CAVG	2CM(1/4)C/B	C,P.(1/4)
-----	---------	----	---------------	-------------	-----------

COMBINATION 1

0.980785	0.158409	0.633634	0.256632	0.060014	=0.094714
0.923880	0.309436	1.237745	0.501306	0.103865	=0.083914
0.831470	0.446231	1.784924	0.722922	0.123848	=0.069386
0.707107	0.563064	2.252254	0.912198	0.123398	=0.054789
0.555570	0.655877	2.623506	1.062561	0.112760	=0.042981
0.382683	0.722500	2.890001	1.170495	0.101712	=0.035194
0.195090	0.762329	3.049315	1.235021	0.094988	=0.031150
0.000000	0.775547	3.102187	1.256434	0.092922	=0.029954

COMBINATION 1

ETA = 0.000000

X	CHI	DELTA=CP
0.990000	0.005000	31.478989
0.975000	0.012500	19.750305
0.950000	0.025000	13.779659
0.900000	0.050000	9.483920
0.800000	0.100000	6.347512
0.700000	0.150000	4.899052
0.600000	0.200000	4.004580
0.400000	0.300000	2.897932
0.200000	0.400000	2.205022
0.000000	0.500000	1.712043
0.200000	0.600000	1.332597
0.400000	0.700000	1.021610
0.600000	0.800000	0.748692
0.800000	0.900000	0.480876
0.900000	0.950000	0.325402

ETA = 0.382700

X	CHI	DELTA=CP
0.990000	0.005000	29.991379
0.975000	0.012500	18.800644
0.950000	0.025000	13.098129
0.900000	0.050000	8.988827
0.800000	0.100000	5.981541
0.700000	0.150000	4.590199
0.600000	0.200000	3.730894
0.400000	0.300000	2.670218
0.200000	0.400000	2.010678
0.000000	0.500000	1.546372
0.200000	0.600000	1.193763
0.400000	0.700000	0.909153
0.600000	0.800000	0.663239
0.800000	0.900000	0.425072
0.900000	0.950000	0.287623

FTA = 0.707100

X	CHI	DELTA=CP
=0.990000	0.005000	26.233292
=0.975000	0.012500	16.347061
=0.950000	0.025000	11.276095
=0.900000	0.050000	7.586061
=0.800000	0.100000	4.852101
=0.700000	0.150000	3.581182
=0.600000	0.200000	2.802904
=0.400000	0.300000	1.872190
=0.200000	0.400000	1.334875
=0.000000	0.500000	0.994303
0.200000	0.600000	0.766515
0.400000	0.700000	0.604484
0.600000	0.800000	0.474023
0.800000	0.900000	0.337608
0.900000	0.950000	0.243238

ETA = 0.923900

X	CHI	DELTA=CP
=0.990000	0.005000	17.194107
=0.975000	0.012500	10.620198
=0.950000	0.025000	7.216769
=0.900000	0.050000	4.706879
=0.800000	0.100000	2.817730
=0.700000	0.150000	1.935975
=0.600000	0.200000	1.403832
=0.400000	0.300000	0.798974
=0.200000	0.400000	0.493142
=0.000000	0.500000	0.340709
0.200000	0.600000	0.275350
0.400000	0.700000	0.256767
0.600000	0.800000	0.252835
0.800000	0.900000	0.225860
0.900000	0.950000	0.180197

LOADS ON THIN, LIFTING WING

=====

RECTANGULAR WING AR = 2 11-13-73

ID1 = 4
ID2 = 2
ID3 = 9
ID4 = 19
DELTA0 = 4.0000
EPS = 0.5000
MACH = 0.0000
JJ = 191
JJMAX = 191
NMAX2 = -3
KK2 = 11
PPNEW = 3
MMNEW = 11
NROWBA = 18
CWTYPE = 0
SWTYPE = 0
BCS = TTTTTTTTTT
BCAS = FTTTTTTTTT

MACH = 0.0000
(B/2)/BREF = 1.0000
CBAR/BREF = 1.0000
ASPECT RATIO = 2.0000

OUTPUT FOR .NS

WEIGHTS

COMBINATION 1
 1.000000 0.000000 0.000000

COMBINATION 1

CL = 2.469045
 INDUCED DRAG = 0.970690
 VORTEX DRAG FACTOR = 1.000464
 ROLLING MOMENT $M/(Q*S^2*BREF)$ = 0.000000
 RIGHT ROOT BENDING MOMENT $MBR/(Q*S*BREF)$ = 0.527907
 LEFT ROOT BENDING MOMENT $MBL/(Q*S*BREF)$ = 0.527907
 LIFT ON ETA,GT,0 = 1.234524
 LIFT ON ETA,LT,0 = 1.234524
 CENTER OF PRESSURE OF RIGHT HALF, Y/BREF = 0.427620
 CENTER OF PRESSURE OF LEFT HALF, Y/BREF = -0.427620
 PITCHING MOMENT ABOUT X = 0 AND CENTER OF PRESSURE

J	CM/(Q*S*CBAR)	X(C,P.)/CHAR
11	=0.510568	0.206788
23	=0.510568	0.206788
47	=0.510568	0.206788

52

ETA	CL+C/2B	CL	CL+C/CCL*CAVG	2CM(1/4)C/B	C,P.(1/4)
COMBINATION 1					

0.980785	0.158293	0.633171	0.256443	0.081057	=0.128018
0.923880	0.309171	1.236685	0.500875	0.117553	=0.095055
0.831470	0.446048	1.784190	0.722623	0.123410	=0.069168
0.707107	0.563105	2.252419	0.912262	0.121034	=0.053735
0.555570	0.655981	2.623924	1.062727	0.112398	=0.042836
0.382683	0.722562	2.890248	1.170591	0.102298	=0.035394
0.195090	0.762372	3.049488	1.235085	0.095699	=0.031382
0.000000	0.775587	3.102347	1.256495	0.093224	=0.030049

COMBINATION 1

ETA = 0,000000

X	CHI	DELTA=CP
=0,990000	0,005000	31,495758
=0,975000	0,012500	19,760391
=0,950000	0,025000	13,786187
=0,900000	0,050000	9,487723
=0,800000	0,100000	6,349156
=0,700000	0,150000	4,899653
=0,600000	0,200000	4,004548
=0,400000	0,300000	2,897223
=0,200000	0,400000	2,204043
=0,000000	0,500000	1,711021
0,200000	0,600000	1,331676
0,400000	0,700000	1,020876
0,600000	0,800000	0,748191
0,800000	0,900000	0,480619
0,900000	0,950000	0,325263

ETA = 0,382700

X	CHI	DELTA=CP
=0,990000	0,005000	30,062637
=0,975000	0,012500	18,842285
=0,950000	0,025000	13,123680
=0,900000	0,050000	9,001715
=0,800000	0,100000	5,984263
=0,700000	0,150000	4,588173
=0,600000	0,200000	3,726260
=0,400000	0,300000	2,663537
=0,200000	0,400000	2,004232
=0,000000	0,500000	1,541389
0,200000	0,600000	1,190875
0,400000	0,700000	0,908549
0,600000	0,800000	0,664668
0,800000	0,900000	0,427675
0,900000	0,950000	0,290082

ETA = 0,707100

X	CHI	DELTA-CP
=0,990000	0,005000	25,829910
=0,975000	0,012500	16,113449
=0,950000	0,025000	11,135412
=0,900000	0,050000	7,519027
=0,800000	0,100000	4,844490
=0,700000	0,150000	3,601092
=0,600000	0,200000	2,837557
=0,400000	0,300000	1,917151
=0,200000	0,400000	1,376247
=0,000000	0,500000	1,024673
0,200000	0,600000	0,782193
0,400000	0,700000	0,604663
0,600000	0,800000	0,460818
0,800000	0,900000	0,317323
0,900000	0,950000	0,224658

ETA = 0,923900

X	CHI	DELTA-CP
=0,990000	0,005000	18,919891
=0,975000	0,012500	11,624463
=0,950000	0,025000	7,827420
=0,900000	0,050000	5,006718
=0,800000	0,100000	2,867131
=0,700000	0,150000	1,868619
=0,600000	0,200000	1,272831
=0,400000	0,300000	0,619891
=0,200000	0,400000	0,323218
=0,000000	0,500000	0,210693
0,200000	0,600000	0,201001
0,400000	0,700000	0,242450
0,600000	0,800000	0,291610
0,800000	0,900000	0,294927
0,900000	0,950000	0,245205

PROGRAM HALTED DUE TO STOP COMMAND

APPENDIX II

COMPUTER PROGRAM LISTING

INTEGER UCI,UCO,U6,U7,U12,CWTYPE,SWTYPE,UNSYM,TJJMAX,
PPNEW,PP,ODISK

REAL MACH
DOUBLE PRECISION DUM1, DUM2, ETA, STHETA
LOGICAL BATCH, CONV, BCS, BCAS, SYM, READ7, OPEN6, CHECK

DIMENSION TITLE(26),BNK(630),BNK1(750),CHICP(20),
BCS(10),BCAS(10),NINDEX(47),STOR1(383),STOR2(383),
STHETA(384),XSILIP(383),CORDIP(383),ETAD(50),WEIGHT(50),
ETA(383)

.....NBK1 AND NWEI ARE THE DIMENSIONS OF (BNK1) AND (WEIGHT),
DATA UCI,UCO,U7,U12,NBNK1,NWEI,COM/5,6,7,12,750,50,34 /

ESTABLISH WHETHER THIS IS A BATCH OR CONVERSATIONAL JOB
WRITE(UCO,802)
READ(UCI,803) BATCH
CONV = .NOT.BATCH
IDIR = .999

OPEN6 INDICATES WHETHER OR NOT UNIT U6 IS OPEN.
THIS VARIABLE IS NOT NEEDED IN STANDARD FORTRAN.
OPEN6 = .FALSE.

10 CONTINUE
CALL OBEY(16,16HRELEASE FT04F001)
IF(CONV) WRITE(UCO,805)

ODISK DETERMINES WHETHER OR NOT TO PUT OUTPUT ON DISK FILE (UNIT 4)
IF ODISK=0, OUTPUT WILL BE ON UNIT 6
IF ODISK .NE. 0 OUTPUT WILL BE ON UNIT 4 ON A FILE WHOSE NAME
IS GENERATED BY THIS PROGRAM.

READ(UCI,810) ODISK
ODISK=MOD(ODISK,10)

ENTER ODISK.LT.0 TO TERMINATE EXECUTION.
IF(ODISK.LT.0) STOP
U6 = UCO

```
IF(ODISK,NE,0)U6=4
OPEN6 = UCO,NE,U6
IF(,NOT,OPEN6) GO TO 20
```

```
C
C.....THE CODE BELOW IS FOR THE AMES' TSS VERSION. OBEY GIVES COMMANDS
C.....TO THE TSS OPERATING SYSTEM. THE VARIABLE ODISK IS NOT
C.....NECESSARY ON A STANDARD VERSION.
```

```
C
WRITE(UCO,812) ODISK
CALL CVRT(ODISK,1,
1 44H('DDEF FT04F001',,OUTPUT,FOR,N1,I1,6X)
1STOR1,8,8H(8A4) )
CALL OBEY (32,STOR1)
```

```
C
C
20 IF(CONV) WRITE(UCO,816)
READ(UCI,810) IDD1,IDD2,IDD3,IDD4
CHECK = IDD1,GT,0
READ7 = IDIR,NE,IDD1
ID1 = IDD1
ID2 = IDD2
ID3 = IDD3
ID4 = IDD4
IF(,NOT,READ7) GO TO 40
```

```
C
C OPEN AND READ FIRST RECORD OF THE GEOMETRY FILE.
```

```
C.....FOR AMES' TSS VERSION ONLY. GEMFIL ISSUES DDEF COMMANDS TO THE
C.....TSS OPERATING SYSTEM.
```

```
C
CALL GEMFIL(ID1)
READ(U7) ID,PP,MM,CNTYPE,SWTYPE,UNSYM,NDL,NDT,MREF,JJMAX,
INFLAPS,TITLE,NTITL
```

```
C NDL = NUMBER OF DISCONTINUITIES ON LEADING EDGE
C (MINIMUM VALUE OF 1)
```

```
C NDT = NUMBER OF DISCONTINUITIES ON TRAILING EDGE
C (MINIMUM VALUE OF 1)
```

```
C PP = ORIGINAL NUMBER OF CHORDWISE CONTROL POINTS
```

```
C MM = ORIGINAL NUMBER OF SPANWISE CONTROL POINTS ON
```

```

C           ENTIRE WING
C
C CHECK ID NUMBER
C IF(,NOT,CHECK) GO TO 30
C IF(ID1,EQ,ID) GO TO 30
C IF(BATCH) CALL STOP2(UCO,
1' IDD1 DOES NOT MATCH GEOM FILE VALUE ',
2FLOAT(ID))
C PAUSE 'IDD1 DOES NOT MATCH GEOM FILE VALUE'
30 CONTINUE
C IDIR = ID1
C
C OPEN AND READ FIRST RECORD OF SOLUTION FILE
40 CONTINUE
C
C.....FOR AMES; TSS VERSION ONLY, BNKFIL GIVES DDEF COMMANDS TO THE TSS
C.....OPERATING SYSTEM,
C
58 CALL BNKFIL(ID1, ID2, ID3, ID4)
C READ(U12) ID1, ID2, ID3, ID4, TITLE, SYM, NSYM, NASYM,
C 1(BCS(I), I=1, 10), (BCAS(I), I=1, 10), NMAX2, KK2, JJ,
C 2PPNEW, MMNEW, NROWSA, DELTA, EPS, MACH
C
C NMAX2 = NEW NUMBER OF CHORDWISE MODES
C KK2 = NEW NUMBER OF SPANWISE MODES
C PPNEW = NEW NUMBER OF CHORDWISE CONTROL POINTS
C MMNEW = NEW NUMBER OF SPANWISE CONTROL POINTS
C
C IF(,NOT,CHECK) GO TO 60
C
C IDENTIFICATION CHECKING
C IF(IDD1,EQ,ID1) GO TO 50
C IF(BATCH) CALL STOP2(UCO,
1' IDD1 DOES NOT MATCH SOLUTION FILE VALUE',
2FLOAT(ID1))
C PAUSE 'IDD1 DOES NOT MATCH SOLUTION FILE VALUE'
50 IF(IDD2,EQ,ID2) GO TO 52
C IF(BATCH) CALL STOP2(UCO,
1' IDD2 DOES NOT MATCH SOLUTION FILE VALUE',

```

```

2FLOAT(ID2))
  PAUSE 'IDD2 DOES NOT MATCH SOLUTION FILE VALUE'
52 IF(IDD3.EQ.ID3) GO TO 54
  IF(BATCH) CALL STOP2(UCO,
  1) 'IDD3 DOES NOT MATCH SOLUTION FILE VALUE',
  2FLOAT(ID3))
  PAUSE 'IDD3 DOES NOT MATCH SOLUTION FILE VALUE'
54 IF(IDD4.EQ.ID4) GO TO 60
  IF(BATCH) CALL STOP2(UCO,
  1) 'IDD4 DOES NOT MATCH SOLUTION FILE VALUE',
  2FLOAT(ID4))
  PAUSE 'IDD4 DOES NOT MATCH SOLUTION FILE VALUE'
60 CONTINUE

```

```

C
C PRINT HEADING
C WRITE(U6,819)
C WRITE(U6,820)(TITLE(I),I=1,NTITL),ID1,ID2,ID3,ID4,
59 C IDELTA,EPS,MACH,JJ,JJMAX,NMAX2,KK2,PPNEW,MMNEW,NROWSA,CWTYPE,
C 2SWTYPE,(BCS(I),I=1,10),(BCAS(I),I=1,10)

```

```

C NCDIM = NSYM + NASYM
C NDLPT = NDL + NDT

```

```

C
C.....LLMAX1 AND LLMAX2 ARE TWO LIMITS FOR THE NUMBER OF
C.....COMBINATIONS IN SUBROUTINE SBLOAD.
  LLMAX2 = NBNK1 / (NMAX2 * KK2)
  LLMAX1 = NWE1 / NCDIM
  TJJMAX = 2 * JJMAX
  CALL SBLOAD(UCI,U6,UCO,U7,U12,SYM,NSYM,NASYM,NCDIM,
  1NMAX2,KK2,JJMAX,PP,MM,MREF,NDLPT,TJJMAX,READ7,
  2BNK,BNK1,CHIC,NINDEX,STOR1,STOR2,ETA,STHETA,
  3XSILIP,CORDIP,ETAD,CONV,WEIGHT,LLMAX1,LLMAX2,COM,MACH)
  GO TO 10

```

```

C
802 FORMAT(1H , 'ENTER BATCH' )
803 FORMAT(L1)
805 FORMAT(1H , 'ENTER ODISK (NEG.HALTS )' )
810 FORMAT(16I5)
812 FORMAT(29H OUTPUT IS ON OUTPUT,FOR,N ,11)

```

```

816 FORMAT(1H , 'ENTER ID1, ID2, ID3, ID4' )
819  FORMAT('1LOADS ON THIN, LIFTING WING' /
1' #####)
820  FORMAT( 1X, 20A4//, I ID1   = 1, I3//, I ID2   = 1,
1I3//, I ID3   = 1, I3//, I ID4   = 1, I3//
2I DELTA0 = 1, F8, 4//, I EPS     = 1, F8, 4//, I MACH   = 1,
3F8, 4//, I JJ     = 1, I3//, I JJMAX = 1, I3//, I NMAX2 = 1, I3//,
4I KK2     = 1, I3//, I PPNEW  = 1, I3//, I MMNEW  = 1,
5I3//, I NROWSA = 1, I3//, I CWTYPE = 1, I3//
6I SWTYPE = 1, I3//, I BCS     = 1, 10L1//, I BCAS = 1, 10L1)

```

```

END
SUBROUTINE SBLOAD(UCI, U6, UCO, U7, U12, SYM, NSYM, NASYM, NCDIM,
1NMAX, KK, JJMAX, PP, MM, MREF, NDLPT, TJJMAX, READ7, BNK,
2BNK1, CHICP, NINDEX, STOR1, STOR2, ETA, STHETA, XSILIP, CORDIP,
3ETAD, CONV, WEIGHT, LLMAX1, LLMAX2, COM, MACH)

```

```

DOUBLE PRECISION ETA, STHETA, PID, THETSW, DTHETA, THETA,
1THETCW, THETSC, DELTHE
INTEGER U6, UCO, U7, U12, PP, TJJMAX, UCI
LOGICAL SYM, CONV, PRESUR, PRINT, SLOAD, ASLOAD, TEST, TESTN1,
1TESTN2, READ7, CONTIN, HALF, TEST2, TEST3
REAL MACH
REAL NETS, NEWS, KNE$

```

```

DIMENSION BNK(NMAX, KK, NCDIM), BNK1(NMAX, KK, LLMAX2),
1WEIGHT(NCDIM, LLMAX1), CHICP(PP), NINDEX(MM), STOR1(JJMAX),
2STOR2(JJMAX), ETA(JJMAX), STHETA(JJMAX), XSILIP(JJMAX),
3ETAD(NDLPT), THETSW(200), THETCW(100), THETSC(47), CARDIM(4),
4SLOAD(10), ASLOAD(10), CORDIP(JJMAX)

```

```

NDIM1, NDIM2 AND NDIM3 ARE THE DIMENSIONS OF THETSW,
THETCW, AND THETSC

```

```

DIMENSION TSSCOM(20)
DATA NDIM1, NDIM2, NDIM3, 200, 100, 47 /
DATA PID, PI2, FOPI, FOPI, PI8, PI4/3, 141592653589793 D0,
11, 570796, 1, 273240, 2, 546479, 0, 3926991, 0, 7853982 /
DATA ETAS / 3HETA /

```

```
DATA XCPS / 3HXCP /
DATA SPAS / 3HSPA /
DATA ECPS / 3HECP /
DATA NETS / 3HNET /
DATA PRES / 3HPRE /
DATA FLAS / 3HFLA /
DATA CONS / 3HCON /
DATA STOS / 3HSTO /
DATA NEWS / 3HNEW /
DATA WEIS / 3HWEI /
DATA KNE$ / 3HKNE /
DATA PWES / 3HPWE /
DATA TSS$ / 3HTSS /
```

C
C

```
READ SECOND RECORD OF GEOMETRY FILE
IF(READ7)READ(U7) CHICP,NINDEX,STOR1,STOR2,STOR1,STOR2,
1ETA,STHETA,
1XSILIP,CORDIP,BRATIO,CBARBR,AR,TR,ZMACH,ETAD
```

61
C

```
WRITE BALANCE OF HEADING
WRITE(U6,8000) MACH,BRATIO,CBARBR,AR
```

C
C

```
CONTIN = CONV
2 CONTINUE
IF(SYM) GO TO 3
```

C
C

```
READING SOLUTIONS FOR UNSYMMETRIC WING
DO 2001 I=1,NCDIM
2001 READ(U12) ((BNK(N,K,I),N=1,NMAX),K=1,KK)
GO TO 11
3 IF(NSYM,EQ,0) GO TO 6
```

C
C

```
READING SYMMETRIC SOLUTIONS FOR SYMMETRIC WING
DO 2002 I=1,NSYM
2002 READ (U12) ((BNK(N,K,I),N=1,NMAX),K=1,KK,2)
DO 4 I = 1,NSYM
DO 4 K = 2,KK,2
DO 4 N = 1,NMAX
```



```
4 BNK(N,K,I) = 0.
6 IF(NASYM, EQ, 0) GO TO 11
  I1 = NSYM + 1
```

```
C
C   READING ANTISYMMETRIC SOLUTIONS FOR SYMMETRIC WING
```

```
DO 2003 I=I1, NCDIM
2003 READ(U12) ((BNK(N,K,I), N=1, NMAX), K=2, KK, 2)
DO 7 I = I1, NCDIM
DO 7 K = 1, KK, 2
DO 7 N = 1, NMAX
7 BNK(N,K,I) = 0.
11 CONTINUE
```

```
C
C   TESTN1 = NMAX, GT, 1
C   TESTN2 = NMAX, GT, 2
C   JUPPR = (JJMAX+1) / 2
C   DELTHE = PID / DFLOAT(JJMAX+1)
C   AREFF = AR / BRATIO**2
C   CBARB2 = CBARBR * BRATIO
C   CON1 = PI2 * AREFF
C   CON2 = CON1 / 2.
C   CON3 = CON2 / 2.
C   JJMAX1 = JJMAX + 1
C   ISUB = JJMAX1 + JJMAX1
```

62

```
C
C   THE DIMENSION OF STHETA IN THE CALLING PROGRAM MUST
C   BE ONE LARGER THAN JJMAX
C   STHETA(JJMAX1) = 0.00
```

```
C
C   ENTER COMBINATION CODE
C   DEFINES A NEW SET OF WEIGHTS
15 IF(CONV) WRITE(UCO, 8, 15)
```

```
C
C.....IF LCOMB, GT, 0, THEN LCOMB=THE NUMBER OF COMBINATIONS AND
C.....THE WEIGHTS OF EACH COMBINATION WILL BE ENTERED BY THE
C.....USER.
C.....IF LCCOM, EQ, 0, THEN THE NUMBER OF COMBINATIONS = THE NUMBER OF
C.....CASES (SOLUTIONS) AND EACH SOLUTION WITH A FACTOR OF 1.0
C.....IS TREATED AS A COMBINATION.
```

C.....IF LCOMB, EQ, -1, THEN THE NUMBER OF COMBINATIONS = THE NUMBER
 C.....OF SYMMETRIC CASES AND EACH SYMMETRIC SOLUTION WITH A
 C.....FACTOR OF 1.0 IS TREATED AS A COMBINATION.
 C.....IF LCOMB, EQ, -2, THEN THE ABOVE HOLDS, EXCEPT THAT THE
 C.....ANTI-SYMMETRIC CASES ARE INVOLVED.
 C.....IF LCOMB, LE, -3, THEN THE CURRENT SET OF WEIGHTS WILL BE
 C.....USED.

C

```

READ(UCI,8020) LCOMB
IF(IABS(LCOMB+1),LT,2) GO TO 32
IF(LCOMB,GT,0) LL=LCOMB
16 IF(LL,LE,MINO(LLMAX1,LLMAX2)) GO TO 17
IF(,NOT,CONV) CALL STOP2(UCO,
1' TOO MANY COMBINATIONS ',FLOAT(LL))
PAUSE ' TOO MANY COMBINATIONS '

```

GO TO 15

17 IF(LCOMB,LE,-3) GO TO 39

I1 = NSYM+1

DO 30 L = 1,LL

IF(CONV) WRITE(UCO,8025) L

IF(NSYM,FG,0) GO TO 20

IF(CONV) WRITE(UCO,8030) NSYM

READ(UCI,8035)(WEIGHT(I,L),I=1,NSYM)

20 CONTINUE

IF(NASYM,EQ,0) GO TO 30

IF(CONV) WRITE(UCO,8040) NASYM

READ(UCI,8035)(WEIGHT(I,L),I=I1,NCDIM)

30 CONTINUE

GO TO 39

C

32 IF(LCOMB,EQ,0) LL = NCDIM

IF(LCOMB,EQ,-1) LL = NSYM

IF(LCOMB,EQ,-2) LL = NASYM

IF(CONV) GO TO 33

IF(LL,GT,MINO(LLMAX1,LLMAX2)) CALL STOP2(UCO,

1' TOO MANY CASES',FLOAT(LL))

IF(LL,LE,0) CALL STOP2(UCO,

1' NUMBER OF CASES IS NOT POSITIVE ',FLOAT(LL))

GO TO 34

```

33 CONTINUE
   IF(LL.GT,MINO(LLMAX1,LLMAX2))
   1 PAUSE 'TOO MANY CASES'
   IF(LL.LE,0) PAUSE 'NUMBER OF CASES IS NOT POSITIVE'
34 CONTINUE
   DO 35 L = 1,LL
   DO 35 I = 1,NCDIM
35 WEIGHT(I,L) = 0.
   I = 0
   IF(LL.EQ,=2) I = NSYM
   DO 36 L=1,LL
   I = I + 1
36 WEIGHT(I,L) = 1.
39 CONTINUE
   IF(CONV,AND,UCO,EQ,U6) GO TO 49

```

```

C
C.....THE SET OF WEIGHTS IS TO BE PRINTED EXCEPT IN
C.....THE CONVERSATIONAL MODE AND WHEN ALL
C.....OUTPUT IS ON THE TERMINAL

```

49

```

C
   WRITE(U6,8045)
   DO 45 L=1,LL
45 WRITE(U6,9051)L,(WEIGHT(I,L),I=1,NCDIM)
49 CONTINUE

```

```

C
C.....NOW THAT THE WEIGHTS HAVE BEEN DETERMINED, THE
C.....COEFFICIENTS OF EACH COMBINATION WILL BE CALCULATED.

```

```

   DO 150 L = 1,LL
   DO 140 K = 1,KK
   DO 140 N = 1,NMAX
140 BNK1(N,K,L) = 0.
   DO 150 I = 1,NCDIM
   DUM1 = WEIGHT(I,L)
   DO 150 K = 1,KK
   DO 150 N = 1,NMAX
150 BNK1(N,K,L) = BNK1(N,K,L) + DUM1 * BNK(N,K,I)
   TEST = KK,LT,2
   DO 190 L = 1,LL
   SLOAD(L) = TFST

```



```

IF(MM,GT,NDIM3) CALL STOP2(UCO,
1'NOT ROOM FOR DEFAULT SPANWISE PRESSURE STATIONS,
2INC, NDIM3 TO 1,FLOAT(MM))
JRATIO = JJMAX1 / (MREF+1)
DO 195 M = 1,MM
INDEX = NINDEX(M) * JRATIO
THETA = DFLOAT(INDEX)*DELTHE
THETSW(M) = THETA
THETCW(M) = THETA
THETSC(M) = THETA
195 CONTINUE
NTABS = MM
NTABC = MM
NTABSC = MM
50 CONTINUE

```

C-----

```

99 C 200 CONTINUE
IF(CONV)WRITE(UCO,8010)
ENTER COMMANDS
READ(UCI,8000)(CARDIM(I),I=1,4)
COM = CARDIM(1)
IF (COM .EQ. ETAS) GO TO 300
IF (COM .EQ. XCP$) GO TO 322
IF (COM .EQ. SPAS) GO TO 342
IF (COM .EQ. ECP$) GO TO 360
IF (COM .EQ. NET$) GO TO 380
IF (COM .EQ. PRES) GO TO 400
IF (COM .EQ. FLAS) GO TO 440
IF (COM .EQ. CON$) GO TO 460
IF (COM .EQ. STOS) GO TO 480
IF (COM .EQ. NEWS) GO TO 485
IF (COM .EQ. WEI$) GO TO 15
IF (COM .EQ. KNE$) GO TO 490
IF (COM .EQ. PWES) GO TO 500
IF (COM .EQ. TSS$) GO TO 520
WRITE(UCO,8005) (CARDIM(I),I=1,4)
IF(CONTIN) GO TO 200
STOP

```

C-----

```
C
C   ETAS----GIVES ETAS FOR PRINTING THE LOADS.  IF NOT GIVEN
C   THE RESULTS WILL BE THE SAME AS IF GIVEN WITH
C   NTYPE = 0
C
```

```
ENTER VALUE FOR NTYPE
```

```
300 IF(CONV) WRITE(UCO,8070)
    READ(UCI,8020) NTYPE
    IF(NTYPE,EQ,0) GO TO 301
    IF(NTYPE,EQ,1) GO TO 304
    IF(NTYPE,EQ,2) GO TO 308
    IF(NTYPE,EQ,3) GO TO 312
    IF(NTYPE,EQ,5) GO TO 317
    IF(.NOT.CONV) CALL STOP2(UCO,
1) NTYPE INVALID 1,FLOAT(NTYPE))
    WRITE(UCO,8075)
    GO TO 300
```

```
NTYPE = 0
```

```
301 NQ = (JJMAX+1)/(MREF+1)
    GO TO 305
```

```
NTYPE = 1
```

```
304 IF(CONV) WRITE(UCO,8080)
    READ(UCI,8020) NQ
```

```
305 I1 = 0
```

```
I2 = (JJMAX + 1) / NQ
```

```
NTABS = I2 = 1
```

```
IF(HALF) NTABS = I2/2
```

```
DO 307 J = NQ,JUPPR,NQ
```

```
I1 = I1 + 1
```

```
THETSW(I1) = DFLOAT(J)*DELTHE
```

```
IF(HALF) GO TO 307
```

```
I2 = I2 - 1
```

```
THETSW(I2) = PID-THETSW(I1)
```

```
307 CONTINUE
```

```
GO TO 200
```

```
NTYPE = 2
```

```

308 IF(CONV) WRITE(UCO,8085)
    READ(UCI,8020) NSTA
    I2 = NSTA + 1
    NUPPR = I2/2
    NTABS = NSTA
C   IF(HALF) NTABS = NUPPR
    DTHETA = PID / DFLOAT(NSTA+1)
    DO 310 I1 = 1, NUPPR
    THETSW(I1) = DFLOAT(I1) * DTHETA
C   IF(HALF) GO TO 310
    I2 = I2 - 1
    THETSW(I2) = PID - THETSW(I1)
310 CONTINUE
    GO TO 200

C
C   NTYPE = 3
312 NTABS = 0
C   ENTER TABLE OF VALUES FOR ETA
    IF(CONV) WRITE(UCO,8090)
89 313 NTABS = NTABS + 1
    READ(UCI,8035) THETSW(NTABS)
    IF(THETSW(NTABS).LT.1,DO) GO TO 313
    IF(NTABS.LE.NDIM1) GO TO 314
    CALL STOP2(UCO,
    1' TOO MANY ENTRIES IN SPANLOAD ETA TABLE 1,
    2'SNGL(THETSW(NTABS)))
314 NTABS = NTABS - 1
    DO 316 N = 1, NTABS
316 THETSW(N) = DARCOS(THETSW(N))
    GO TO 200

C
C   NTYPE = 4
317 IF(CONV) WRITE(UCO,8087)
    READ(UCI,8035) ETMIN,ETMAX,DETA
    IF(DETA,EQ,0) DETA = ETMIN
    IF(ETMAX,EQ,0) ETMAX = .9999
    THETSW(1) = ETMIN
    NTABS = 1
318 IF(THETSW(NTABS).GT,ETMAX) GO TO 319

```

```

        NLAST = NTABS
        NTABS = NTABS + 1
        THETSW(NLAST) = THETSW(NLAST) + DELTA
        GO TO 318
319  IF(NTABS.GT.NDIM1)CALL STOP2(UCO,
        1'SPANWISE LOAD TABLE SIZE EXCEEDED ',FLOAT(NTABS))
        NTABS = NTABS - 1
        DO 321 I = 1,NTABS
321  THETSW(I) = DARCOS(THETSW(I))
        GO TO 200

```

C-----

```

C
C      XCP--DEFINES XIS FOR PRINTING THE CHORDWISE PRESSURE
C      DISTRIBUTIONS

```

```

69 322 IF(CONV) WRITE(UCO,8070)
        READ(UCI,8020) NTYPE
        IF(NTYPE.EQ.0) GO TO 323
        IF(NTYPE.EQ.1) GO TO 324
        IF(NTYPE.EQ.2) GO TO 328
        IF(NTYPE.EQ.3) GO TO 332
        IF(NTYPE.EQ.4) GO TO 332
        IF(NTYPE.EQ.5) GO TO 338
        IF(,NOT,CONV) CALL STOP2(UCO,
        1' NTYPE INVALID ',FLOAT(NTYPE))
        WRITE(UCO,8075)
        GO TO 322

```

```

C
C      NTYPE = 0
323  NQ = (JJMAX+1)/(MREF + 1)
        GO TO 325

```

```

C
C      NTYPE = 1
324 IF(CONV) WRITE(UCO,8080)
        READ(UCI,8020) NQ
325  I1 = 0
        I2 = (JJMAX + 1) / NQ
        NTABC = I2 - 1
        DO 327 J = NQ,JUPPR,NQ
        I1 = I1 + 1

```



```
THETCW(I1) = DFLOAT(J)*DELTHE
I2 = I2 + 1
THETCW(I2) = PID = THETCW(I1)
327 CONTINUE
GO TO 200
```

C
C

```
NTYPE = 2
328 IF(CONV) WRITE(UCO,8085)
READ(UCI,8020) NSTA
I2 = NSTA + 1
NUPPR = I2/2
NTABC = NSTA
DTHETA = PID/DFLOAT(NSTA + 1)
DO 330 I1 = 1,NUPPR
THETCW(I1) = DFLOAT(I1) * DTHETA
I2 = I2 + 1
THETCW(I2) = PID = THETCW(I1)
330 CONTINUE
GO TO 200
```

70

C
C

```
NTYPE = 3 OR 4
332 NTABC = 0
IF(CONV,AND,NTYPE,EQ,3) WRITE(UCO,8091)
IF(CONV,AND,NTYPE,EQ,4) WRITE(UCO,8092)
334 NTABC=NTABC+1
READ(UCI,8035) THETCW(NTABC)
IF(THETCW(NTABC),LT,1,) GO TO 334
IF(NTABC.GT,NDIM2) CALL STOP2(UCO,
1' TOO MANY ENTRIES IN X OR CHI TABLE',
2SNGL(THETCW(NTABC)))
NTABC = NTABC - 1
IF(NTYPE,EQ,4) GO TO 336
DO 335 N = 1,NTABC
335 THETCW(N) = DARCOS(-THETCW(N))
GO TO 200
336 DO 337 N = 1,NTABC
337 THETCW(N) = DARCOS(1,DO=2,DO + THETCW(N))
GO TO 200
```

C

```

C      NTYPE = 5
338 IF (CONV) WRITE(UCO,8095)
      READ(UCI,8035) XMIN,XMAX,DX
      IF(XMAX.EQ.0.) XMAX = .9999
      IF(DX.EQ.0.) DX=1.+XMIN
      NTABC = 1
      THETCW(NTABC) = XMIN
339 IF(THETCW(NTABC).GT.XMAX) GO TO 340
      NLAST = NTABC
      NTABC = NTABC + 1
      THETCW(NTABC) = THETCW(NLAST) + DX
      GO TO 339
340 IF(NTABC.GT.NDIM2) CALL STOP2(UCO,
1 1CHORDWISE PRESSURE TABLE SIZE EXCEEDED 1,FLOAT(NTABC))
      NTABC = NTABC - 1
      DO 341 I = 1,NTABC
341 THETCW(I) = DARCOS(-THETCW(I))
      GO TO 200

```

71

```

C -----
C
C      SPANLOADS---PRINTS RESULTS AT SPANWISE STATIONS
C      DEFINED BY ETAS COMMAND. IF THE ETAS COMMAND HAS
C      NOT BEEN GIVEN THE EFFECT IS THE SAME AS IF IT WERE
C      GIVEN WITH NTYPE = 0
342 WRITE(U6,9000)
      DO 356 L = 1,LL
      WRITE(U6,9014) L
      CLTRM=BNK1(1,1,L)
      IF(TESTN1) CLTRM=CLTRM+.5*BNK1(2,1,L)
      TEST3=CLTRM,NE,0,
      CLCCLC=-1,E=10
      TEST = SLOAD(L).OR.ASLOAD(L)
      KL = 1
      KJUMP = 1
      IF(ASLOAD(L)) KL = 2
      IF(TEST) KJUMP = 2
      DO 355 N = 1,NTABS
      THETA = THETSW(N)
      IF(TEST.AND.THETA.GT.(PI2+1,D=4)) GO TO 355

```

```

C   QMOMNT = QUARTER CHORD MOMENT
    QMOMNT = 0.
    GAMMA = 0.
    DO 345 K = KL, KK, KJUMP
    SKTHET = DSIN(DFLOAT(K)*THETA)
    GAMMA = GAMMA + SKTHET*BNK1(1,K,L)
    IF(TESTN1) GAMMA = GAMMA + SKTHET*.5*BNK1(2,K,L)
    IF(TESTN1) QMOMNT = QMOMNT + .5*SKTHET*BNK1(2,K,L)
    IF(TESTN2) QMOMNT = QMOMNT + .5*SKTHET*BNK1(3,K,L)
345 CONTINUE

```

```

C   GAMMA = NONDIMENSIONAL CIRCULATION = (CL+C)/(2R)

```

```

C   QMOMNT = NONDIMENSIONAL PITCHING MOMENT LOAD
          = (2*CM(1/4)*C)/B

```

```

C   ETAS = DCOS(THETA)

```

```

C   A LINEAR INTERPOLATION WILL BE USED TO OBTAIN THE
    CHORD/(B/2) AT THE CURRENT ETAS STATION

```

```

C   (ETA) IS IN DESCENDING ORDER

```

```

C   IF(ETAS.LT.ETA(2)-1.D-6) GO TO 349

```

```

    J1 = 2
    GO TO 352

```

```

349 CONTINUE

```

```

    DO 350 J = 3, JJMAX

```

```

    J1 = J
    IF(ETAS.GT.ETA(J)) GO TO 352

```

```

350 CONTINUE

```

```

352 CONTINUE

```

```

    CORDB2 = CORDIP(J1) + (CORDIP(J1)-CORDIP(J1-1))/
    1 (ETA(J1)-ETA(J1-1))*(ETAS-ETA(J1))

```

```

C   SECTIONAL LIFT COEFFICIENT

```

```

    CL = 4.*GAMMA/CORB2

```

```

C   OVERALL LIFT COEFFICIENT

```

```

C ASPECT RATIO FROM GEOM FILE IS 4*(BREF**2)/AREA.
C CCL = (PI/2)*(B**2/AREA)*(BNK1(1,1,L)+.5*BNK1(2,1,L)).
C BRATIO FROM GEOM FILE IS BREF/(B/2).
C THUS
C (CL*C)/(CCL*CAVG)=GAMMA*4/PI*BRATIO/(BNK1(1,1)+.5*BNK1(2,1))
C
C FOPI = 4./PI
C IF(TEST3) CLCCLC=GAMMA*FOPI*BRATIO/CLTRM
C
C CENTER OF PRESSURE RELATIVE TO THE LOCAL QUARTER CHORD
C TEST2 = ABS(GAMMA).GT.1.E=5*ABS(QMOMNT)
C IF(TEST2) COP = .25* QMOMNT /GAMMA
C IF(TEST2) WRITE(U6,9005) ETAS,GAMMA,CL,CLCCLC,QMOMNT,
C 1COP
C IF(,NOT,TEST2)WRITE(U6,9005) ETAS,GAMMA,CL,CLCCLC,
C 1QMOMNT
C 355 CONTINUE
C 356 CONTINUE
C GO TO 200

```

73

```

C ECP---DEFINE ETAS FOR PRINTING THE CHORDWISE PRESSURE
C 360 IF(CONV) WRITE(UCD,8070)
C READ(UCI,8020) NTYPE
C IF(NTYPE,EQ,0) GO TO 361
C IF(NTYPE,EQ,1) GO TO 364
C IF(NTYPE,EQ,2) GO TO 368
C IF(NTYPE,EQ,3) GO TO 372
C IF(NTYPE,EQ,5) GO TO 376
C IF(CONV) WRITE(UCD,8075)
C IF(CONV) GO TO 360
C CALL STOP2(UCD,' NTYPE INVALID',FLOAT(NTYPE))
C
C NTYPE = 0
C 361 NQ = (JJMAX+1) / (MREF+1)
C GO TO 365
C
C NTYPE = 1
C 364 IF(CONV) WRITE(UCD,8080)

```

```

      READ(UCI,8020) NQ
365  I1 = 0
      I2 = (JJMAX+1)/NQ
      NTABSC = I2 - 1
C     IF(HALF)NTABSC = I2/2
      DO 367 J = NQ,JUPPR,NQ
      I1 = I1 + 1
      THETSC(I1) = DFLOAT(J)*DELTHE
C     IF (HALF) GO TO 367
      I2 = I2 - 1
      THETSC(I2) = PID-THETSC(I1)
367  CONTINUE
      GO TO 200

C
C
368  NTYPE = 2
      IF(CONV)WRITE(UCO,8085)
      READ(UCI,8020)NSTA
      I2 = NSTA + 1
      NUPPR = I2/2
      NTABSC = NSTA
C     IF(HALF) NTABSC = NUPPR
      DTHETA = PID / DFLOAT(NSTA+1)
      DO 370 I1 = 1,NUPPR
      THETSC(I1) = DFLOAT(I1) * DTHETA
C     IF(HALF) GO TO 370
      I2 = I2 - 1
      THETSC(I2) = PID - THETSC(I1)
370  CONTINUE
      GO TO 200

C
C
372  NTYPE = 3
      NTABSC = 0
C     ENTER TABLE OF VALUES FOR ETA
      IF(CONV) WRITE(UCO,8090)
373  NTABSC = NTABSC + 1
      READ(UCI,8035) THETSC(NTABSC)
      IF(THETSC(NTABSC).LT.1.) GO TO 373
      IF(NTABSC.LE.NDIM3) GO TO 374
      IF(.NOT.CONV) CALL STOP2(UCO,

```

```
1' TOO MANY ENTRIES IN PRESSURE ETA TABLE',
2SNGL(THETSC(NTABSC)))
PAUSE 'TOO MANY ENTRIES IN PRESSURE ETA TABLE'
374 NTABSC = NTABSC - 1
DO 375 N = 1,NTABSC
375 THETSC(N) = DARCOS(THETSC(N))
GO TO 200
```

C
C

```
NTYPE = 5
376 IF(CONV) WRITE(UCO,8087)
READ(UCI,8035) ETMIN,ETMAX,DETA
IF(DETA,EQ,0.) DETA = ETMIN
IF(ETMAX,EQ,0.) ETMAX = .9999
THETSC(1) = ETMIN
NTABSC = 1
```

```
377 IF(THETSC(NTABSC),GT,ETMAX) GO TO 378
NLAST = NTABSC
NTABSC = NTABSC + 1
THETSC(NTABSC) = THETSC(NLAST) + DETA
GO TO 377
```

75

```
378 IF(NTABSC,GT,NDIM3) CALL STOP2(UCO,
1'SPANWISE PRESSURE TABLE SIZE EXCEEDED ',FLOAT(NTABSC))
NTABSC = NTABSC - 1
DO 379 I = 1,NTABSC
379 THETSC(I) = DARCOS(THETSC(I))
GO TO 200
```

C

C

```
NETLOADS---COMPUTES AND PRINTS OVERALL RESULTS, CL,CMP
(PITCH MOMENT), CMR(ROLL MOMENT), LEFT AND
RIGHT ROOT BENDING MOMENTS, LEFT AND
RIGHT LIFTS, LEFT AND RIGHT CENTERS OF PRESSURE,
CDI, VORTEX DRAG FACTOR (CDI/CL**2/(PI*AR))
```

C

```
380 IF(CONV) WRITE(UCO,9012)
READ(UCI,8020) JJ,NOUT
DO 399 L = 1,LL
WRITE(U6 ,9014) L
```

C

C

```
THE ASPECT RATIO IN ALL THE FORMULAS WORKED OUT
```

```

C HAS BEEN THE EFFECTIVE ASPECT RATIO, B**2/AREA,
C BUT AR FROM GEOM FILE IS 4*BREF**2/AREA AND BRATIO =
C 2 * BREF/B, SO THE EFFECTIVE ASPECT RATIO IS
C AREFF = AR/BRATIO**2
C ALSO NOTE THAT CBARBR = LONGITUDINAL REFERENCE LENGTH /
C BREF SO CBARB2 = CBARBR * BRATIO =
C LONGITUDINAL REFERENCE LENGTH / (B/2)

```

```

C
C STOR1 = SPANLOAD COEFFICIENTS(B1,B2,...)
C CL*C/(2B) = B1 * SIN(THETA) + B2 * SIN(2*THETA) + ....

```

```

C DO 381 K = 1, KK
C STOR1(K) = HNK1(1,K,L)
381 IF( TESTN1 ) STOR1(K) = STOR1(K) + .5 * HNK1(2,K,L)

```

```

C
C LIFT COEFFICIENT
C CCL = CON1 * STOR1(1)
C WRITE(U6,9016) CCL

```

```

C
C INDUCED DRAG

```

```

C
C SUM = 0.
C K2 = KK + 1
C DO 382 K = 1, KK
C K2 = K2 - 1
382 SUM = SUM + FLOAT(K2)*STOR1(K2)**2
C CD1 = CON2 * SUM
C WRITE(U6,9018) CD1
C IF( STOR1(1).LT.1.E-2*SUM ) GO TO 383

```

```

C
C VORTEX DRAG FACTOR
C VORD = SUM/STOR1(1)**2
C WRITE(U6,9020) VORD

```

```

383 CONTINUE

```

```

C
C ROLLING MOMENT
C CMR = CON3*STOR1(2) / BRATIO
C WRITE(U6,9021) CMR

```

u
u
76
u
u


```

C
C
C
SEMI-SPAN CENTERS OF PRESSURE
TEST3=CLP,NE,0.
IF(TEST3)CPP=CMBP/CLP
IF(TEST3)WRITE(U6,9032) CPP
TEST3=CLM,NE,0.
IF(TEST3)CPM=CMBM/CLM
IF(TEST3)WRITE(U6,9034)CPM

```

```

C
C
C
PITCHING MOMENT

```

```

C
C
C
PITCHING MOMENT CAN NOT BE INTEGRATED ANALYTICALLY FOR
A GENERAL PLANFORM. SUBROUTINE INTGRT WILL BE USED
FOR THE INTEGRATION.

```

```

C
DUM = AREFF/CBAREB2
DO 397 J = 1, JJMAX
SUM = 0.
SUM1 = 0.
DO 396 K = 1, KK
INDEX = J+K
INDEX = MOD(INDEX-1, ISUB) + 1
IF(INDEX.GT. JJMAX1) GO TO 394
SKTHET = STHETA(INDEX)
GO TO 395
394 INDEX = INDEX - JJMAX1
SKTHET = -STHETA(INDEX)
395 CONTINUE

```

```

C
C
C
SKTHET = SIN(K*THETA) WHERE
THETA = ARCOS(ETA(J))

```

```

C
SUM = SUM - SKTHET * STOR1(K)
DUM1 = BNK1(1,K,L)*SKTHET
IF(TESTN1) DUM1 = DUM1 + BNK1(2,K,L)*SKTHET
IF(TESTN2) DUM1 = DUM1 - .5*BNK1(3,K,L)*SKTHET
SUM1 = SUM1 + DUM1
396 CONTINUE

```

```

C
C SUM = -SUMMATION (K=1, KK) SIN(K*THETA)*
C          (B1K + .5 * B2K) * -CL*C/2B
C SUM1 = SUMMATION (K=1, KK) SIN(K*THETA)*
C          (B1K + B2K - .5*B3K) * -2.*CM(LE)*C/B
C STOR2(J) = DUM * (XSILIP(J)*SUM+.25*CURDIP(J)*SUM1)
397 CONTINUE

```

```

C
C THE INTEGRATION FOR PITCHING MOMENT WILL BE DONE WITH
C INCREASING NUMBERS OF INTEGRATION POINTS. IT WILL
C START WITH MREF POINTS AND INCREASE UP TO JJMAX
C BUT NOT EXCEED JJ.
C THE DEFAULT FOR JJ IS JJMAX
C NOUT IS THE OUTPUT LEVEL FOR SUBROUTINE INTGRT,
C NOUT = 0 IS USUAL
C WRITE(U6,9036)
C IF(JJ.EQ.0) JJ = JJMAX
C J = MREF
79 TEST3=CCL.NE.0.
C DO 398 N = 1,10
C CALL INTGRT(J, JJMAX, JJMAX, ETA, STHETA, STOR2, NDLNDT, ETAD,
C 1STOR1, U6, NOUT, NLEPT, CMP)
C IF(TEST3) XCP=CMP/CCL
C IF(TEST3) WRITE(U6,9038) J, CMP, XCP
C IF(.NOT.TEST3) WRITE(U6,9038) J, CMP
C J = J + J + 1
C IF(J.GT. JJMAX) GO TO 399
C IF(J.GT. JJ) GO TO 399
398 CONTINUE
399 CONTINUE
GO TO 200

```

```

C-----
C
C PRESSURES---COMPUTES AND PRINTS THE LIFTING PRESSURES AT
C THE CHORDWISE LOCATIONS DEFINED BY THE XCP
C COMMAND AND THE SPANWISE LOCATIONS DEFINED
C BY THE ECP COMMAND.
C
400 CONTINUE
DO 428 L=1, LL

```

```

WRITE(U6,9014) L
DO 425 NS = 1,NTABSC
ETAS = DCOS(THETSC(NS))
IF( SLOAD(L),AND,ETAS,LT,-1.E-4) GO TO 425
IF(ASLOAD(L),AND,ETAS,LT, 1.E-4) GO TO 425
WRITE(U6,9042) ETAS

```

C
C
C
C
C
C

```

A LINEAR INTERPOLATION WILL BE USED TO OBTAIN THE
CHORD/(B/2) AT THE CURRENT ETAS STATION

```

```

(ETA) IS IN DESCENDING ORDER

```

```

IF(ETAS,LT,ETA(2)=1.D=6) GO TO 401
J1 = 2
GO TO 403
08 401 CONTINUE
DO 402 J = 3,JJMAX
J1 = J
IF(ETAS,GT,ETA(J)) GO TO 403
402 CONTINUE
403 CONTINUE
CORDB2 = CORDIP(J1) + (CORDIP(J1)-CORDIP(J1-1))/
1 (ETA(J1)-ETA(J1-1))*(ETA=ETA(J1))
DO 411 N = 1,NMAX
411 STOR1(N) = 0.
DO 412 K = 1,KK
DTHETA = DSIN(DFLOAT(K)*THETSC(NS))
DO 412 N = 1,NMAX
412 STOR1(N) = STOR1(N) + BNK1(N,K,L) * DTHETA
DO 424 NC = 1,NTABC
THETA = THETCW(NC)
DELCP = STOR1(1) /TAN(.5*THETA)
IF(NMAX,LT,2) GO TO 417
DO 415 N = 2,NMAX
DELCP = DELCP + STOR1(N) * DSIN(DFLOAT(N-1)*THETA)
415 CONTINUE
417 DELCP = EOP1 * DELCP/CORDB2
X = -DCOS(THETA)
CHI = .5 * (1. + X)

```

```
WRITE(U6,9044) X,CHI,DELCP
424 CONTINUE
425 CONTINUE
428 CONTINUE
GO TO 200
```

```
C
C
C FLAP
440 CONTINUE
WRITE(U6,9046)
GO TO 200
```

```
C
C
C CONTINUE---CAUSES PROGRAM TO CONTINUE EXECUTION IN BATCH
C MODE EVEN IF AN INVALID COMMAND IS ENCOUNTERED.
460 CONTIN = .TRUE.
GO TO 200
```

```
18 C
C
C STOP---HALTS EXECUTION
480 WRITE(U6,9050)
STOP
```

```
C
C
C NEW---STARTS A NEW CASE, READS NEW SOLUTION FILE AND
C (MAYBE) A NEW GEOMETRY FILE
485 RETURN
```

```
C
C
C KNEW---STARTS A NEW CASE, READS A NEW SOLUTION FILE
C AND (MAYBE) A NEW GEOMETRY FILE, RETAINS CURRENT
C STATIONS FOR PRESSURE AND SPANLOADS
490 RETURN
```

```
C
500 CONTINUE
```

```
C
C.....PWEIGHTS COMMAND PRINTS THE CURRENT WEIGHTS
WRITE(UC0,8045)
DO 505 L=1,LL
```

```
505 WRITE(UCD,9051) L, (WEIGHT(I,L),I=1,NCDIM)
GO TO 200
```

```
C-----
```

```
C
520 CONTINUE
```

```
C
C.....TSS COMMAND ALLOWS USER TO GIVE TSS COMMANDS TO OPERATING SYSTEM.
C.....TSSCOM IS AN ARRAY IN WHICH TO STORE THE COMMAND AND IS NOT NEEDED
C.....OTHERWISE.
```

```
C
READ (UCI,9052) TSSCOM
CALL OBEY(80,TSSCOM)
GO TO 200
```

```
C-----
```

```
C
C
C
C
```

82

```
8000 FORMAT(/,1 MACH = 1,F8.4/,1 (B/2)/BREF = 1,
1F8.4/,1 CBAR/BREF = 1,F8.4/,
2' ASPECT RATIO = 1,F8.4/)
8010 FORMAT(1 + 1 )
8015 FORMAT(' ENTER COMBINATION CODE: )
8020 FORMAT(16I5)
8025 FORMAT(' COMBINATION 1,13/)
8030 FORMAT(' ENTER WEIGHTS OF FIRST 1,13,1 CASES: )
8035 FORMAT(BF10.0)
8040 FORMAT(' ENTER WEIGHTS OF LAST 1,13,1 CASES: )
8045 FORMAT(1H1/' WEIGHTS!/)
8060 FORMAT(4A3)
8065 FORMAT(' INVALID COMMAND 1,4A3/)
8070 FORMAT(' ENTER NTYPE: )
8075 FORMAT(' NTYPE INVALID!/)
8080 FORMAT(' ENTER NQ: )
8085 FORMAT(' ENTER NSTA: )
8087 FORMAT(' ENTER ETMIN,ETMAX,DETA: )
8090 FORMAT(' ENTER TABLE OF ETAS--1 PER LINE 1,
1'ENDING WITH VALUE GREATER THAN 1' )
8091 FORMAT(' ENTER TABLE OF X VALUES: )
```



```
DIMENSION ETA(JJMAX), STHETA(JJMAX), F(JJR), X(ND), Y(ND)
DIMENSION ET(4), FUN(4)
EQUIVALENCE (YTEMP,YDUM)
```

```
C
DATA PI /3.141592653589793      DO/
DATA EPS /1.E-5/
```

```
C
C.....JJ, JJR, AND JJMAX MUST LINE UP PROPERLY. IF THEY DO NOT THE
C.....PROGRAM WILL WRITE A MESSAGE TO THAT EFFECT AND HALT EXECUTION.
```

```
C
IF(MOD(JJMAX+1, JJR+1) .NE. 0) GO TO 910
IF(MOD(JJR +1, JJ +1) .NE. 0) GO TO 910
JJ1=JJ+1
JR1 =(JJR+1)/JJ1
JR2 =(JJMAX+1)/JJ1
DELTA = PI/DFLOAT(JJ1)
SUM=0
OUT= OUTR
```

```
85 C
C.....THE 50 LOOP DOES THE UNCORRECTED INTEGRATION
```

```
C
DO 50 J=1, JJ
J1=J+JR1
J2=J+JR2
SUM=SUM+STHETA(J2)*F(J1)
50 CONTINUE
VALU = DELTA*SUM
```

```
C
C.....IF THERE DISCONTINUITIES, TRANSFER TO 200.
```

```
C
IF(ND.GT.0) GO TO 200
60 NLEFT=0
IF(OUT.LT.1) RETURN
WRITE (W,1)VALU , JJ
IF(OUT .LT. 2) RETURN
IF(ND .EQ. 0) GO TO 70
WRITE(W,2)
WRITE(W,3) X
70 IF(OUT.LT.3) RETURN
```



```
WRITE(W,4)
DO 75 J=1,JJ
J1=J+JR1
J2=J+JR2
75 WRITE (W,5) ETA(J2), F(J1)
RETURN
```

```
C-----
200 NLEFT = ND
```

```
C
C.....PRIOR TO CORRECTING FOR DISCONTINUITIES IN SLOPE AND VALUE THE
C.....EXTRANEIOUS POINTS ARE ELIMINATED FROM CONSIDERATION. A TRANSFER
C.....BACK TO 60 WILL OCCUR IF THERE ARE NO REAL DISCONTINUITIES.
```

```
C
C
C.....ARRANGING X IN Y IN DESCENDING ORDER
```

```
C
DO 210 N=1,ND
210 Y(N)=X(N)
DO 220 N=1,ND
DO 220 J=N,ND
IF(Y(N),GE,Y(J))GO TO 220
YTEMP=Y(J)
Y(J)=Y(N)
Y(N)=YTEMP
220 CONTINUE
IF(NLEFT,EQ,1)GO TO 300
LMIN=1
LMAX=ND-1
```

```
C
C.....REPEATED VALUES ARE ELIMINATED NEXT
```

```
C
230 DO 240 L=LMIN,LMAX
IF(ABS(Y(L)-Y(L+1)),LT,EPS)GO TO 250
240 CONTINUE
GO TO 260
250 CALL CRUNCH (Y,L,LMAX)
IF(L,GT,LMAX) GO TO 260
LMIN=L
GO TO 230
```

```

260 NLEFT = LMAX+1
300 CONTINUE
C
C.....TOO SMALL Y-VALUES ARE ELIMINATED NEXT
C
      YDUM = EPS-1,
      DO 310 L=1,NLEFT
      IF( Y(L).LT.YDUM) GO TO 320
310 CONTINUE
      GO TO 330
320 NLEFT=L-1
      IF(NLEFT.EQ.0) GO TO 60
330 IF(Y(NLEFT).GT.=YDUM) GO TO 60
400 IF(NLEFT, EQ, 1) GO TO 500
      LMAX=NLEFT-1
      YDUM=YDUM
C
C.....TOO LARGE Y-VALUES ARE ELIMINATED NEXT
87 C
410 IF(Y(1).LE.YDUM) GO TO 440
      CALL CRUNCH(Y,1,LMAX)
      GO TO 410
440 NLEFT = LMAX+1
500 LMIN=1
C
C.....Y-VALUES NEARLY COINCIDENT WITH INTEGRATION STATIONS ARE ELIMINATED NEXT
C
      DO 530 J=JR2,JJMAX,JR2
      ES = ETA(J)
      DO 510 L=LMIN, NLEFT
      IF(ABS(Y(L)-ES).LT. EPS) GO TO 520
510 CONTINUE
      GO TO 530
520 CALL CRUNCH (Y,L, NLEFT)
      IF(NLEFT.EQ.0)GO TO 60
      IF(L.GT.NLEFT)GO TO 540
      LMIN = L
530 CONTINUE
C

```

C.....AT THIS POINT ALL THE EXTRANEIOUS DISCONTINUITIES HAVE BEEN
C.....ELIMINATED AND THERE ARE STILL SOME REMAINING, THE REMAINING DIS-
C.....CONTINUITIES ARE STORED IN (Y).

C
540 KOUNT = 0
CONT = .FALSE.
VALU1=VALU
IF(JJ.LY. 4) GO TO 800

C
C.....AT THIS POINT THE PROGRAM WILL MAKE THE NECESSARY CORRECTIONS.
C.....IT WILL NOT BE ABLE TO DO THIS RIGHT UNDER CERTAIN CONDITIONS,
C.....IF THERE ARE DISCONTINUITIES IN THE FIRST OR LAST TWO INTERVALS,
C.....OR MORE THAN ONE DISCONTINUITY IN AN INTERVAL, OR DISCONTINUITIES
C.....IN ADJACENT INTERVALS, THE CORRECTIONS WILL NOT BE DONE RIGHT.

C
ET(1)=ETA(JR2)
ET(2)=ETA(2*JR2)
ET(3)=ETA(3*JR2)
CORR=0
CONT1=.FALSE.
JMAX=JJ-3
LMIN=1
DO 700 J=1,JMAX
INDEX=(J+3)*JR2
ET(4)=ETA(INDEX)
DO 600 L= LMIN,NLEFT
IF(Y(L) .GT. ET(2)) GO TO 600
IF(Y(L) .GT. ET(3)) GO TO 610
CONT1=.FALSE.
GO TO 690
600 CONTINUE
GO TO 790
610 INDEX=0
JP3=J+3
DO 620 K=J,JP3
K1=K+JR1
K2=K+JR2
INDEX=INDEX+1
620 FUN(INDEX)=F(K1)*STHETA(K2)

```

TD = DARCOS(DBLE(Y(L)))
D2 = TD/DELTA=DFLOAT(J+1)
D3 = 1. - D2
CORR = CORR + D2**2*(FUN(2)=FUN(1)) + D3**2*(FUN(3)=FUN(4))
1 + (2.*D2 - 1.)*(FUN(2)=FUN(3))
CONT = CONT,OR,CONT1
CONT1 = ,TRUE,
KOUNT=KOUNT+1
IF(KOUNT,EQ,NLEFT)GO TO 790

```

```

690 CONTINUE
ET(1)=ET(2)
ET(2)=ET(3)
ET(3)=ET(4)
LMIN=L

```

```

700 CONTINUE

```

```

790 VALU=VALU +CORR*DELTA/2,

```

```

C-----

```

```

C

```

```

68 C.....THIS IS THE END OF THE COMPUTATION. THE REMAINDER OF THE PROGRAM
C.....IS OUTPUT,

```

```

C

```

```

800 IF (OUT,LT,0) RETURN
IF(KOUNT,NE,NLEFT)GO TO 810
IF(CONT) GO TO 810
IF(JJ,LT,4) GO TO 810
IF (OUT,GT,0) GO TO 820
RETURN

```

```

810 OUT =MAX0(OUT,2)

```

```

820 WRITE(W,6) VALU1,VALU,JJ
IF (OUT,LT,2) RETURN
IF (KOUNT,EQ,NLEFT) GO TO 830
KOUNT = NLEFT = KOUNT
WRITE(W,7) KOUNT

```

```

830 IF (,NOT,CONT) GO TO 840
WRITE(W,8)

```

```

840 IF(JJ,GT,3)GO TO 850
WRITE(W,9)

```

```

850 WRITE(W,10)
WRITE(W,3) X

```

```

WRITE(W,11)
WRITE(W,3) (Y(N),N=1,NLEFT)
GO TO 70
910 WRITE (W, 12) JJ, JJR, JJMAX
STOP
1 FORMAT(25H0VALUE OF THE INTEGRAL IS ,F15.7 , 10X,
140H THE NUMBER OF INTEGRATION POINTS USED = ,I5)
2 FORMAT(54H0NONE OF THE X-VALUES WERE CONSIDERED DISCONTINUITIES /-
117H THE X VECTOR IS )
3 FORMAT ( 6F20.6 )
4 FORMAT (10H0 ETA , 10X, 1HF)
5 FORMAT ( F12.6, F13.6)
6 FORMAT (
141H0THE UNCORRECTED VALUE OF THE INTEGRAL = ,F15.7/
241H THE CORRECTED VALUE OF THE INTEGRAL = ,F15.7/
341H THE NUMBER OF INTEGRATION POINTS WAS = ,I7 )
7 FORMAT(25H0---WARNING---THERE WERE , IS,
152H DISCONTINUITIES WHICH COULD NOT BE CORRECTED FOR,
8 FORMAT ( 80H0---WARNING---THERE WERE DISCONTINUITIES IN CONTIGUOUS-
1 INTEGRATION INTERVALS )
9 FORMAT (100H0---WARNING---NO DISCONTINUITIES ARE CORRECTED FOR WHE-
IN THERE ARE LESS THAN 4 INTEGRATION POINTS )
10 FORMAT (26H0ORIGINAL DISCONTINUITIES )
11 FORMAT (25H0REDUCED DISCONTINUITIES )
12 FORMAT(53H0JJ, JJR, AND JJMAX ARE NOT PROPERLY RELATED, /-
19H0 JJ = , I7/
19H JJR = , I7/
19H JJMAX = , I7/
134H0EXECUTION TERMINATED IN INTGRT )
END
SUBROUTINE CRUNCH (Y, L, LMAX)
DIMENSION Y(1)
DO 10 K = L, LMAX
10 Y(K) = Y(K+1)
LMAX = LMAX - 1
RETURN
END
SUBROUTINE STOP2(N,MESSAGE,VAL)
DIMENSION MESSAGE(20)

```

```
WRITE(N,1) MESSAGE,VAL  
STOP 13  
1 FORMAT(///1 ***** 1,20A4,1 *****1/1 VAL=1,1PE15,7)  
END
```

16