1147

NASA TECHNICAL MEMORANDUM

NASA TM X-2913

NASA TM X-2913

COMPUTER PROGRAM FOR SUPERSONIC KERNEL-FUNCTION FLUTTER ANALYSIS OF THIN LIFTING SURFACES

by Herbert J. Cunningham

Langley Research Center
Hampton, Va. 23665

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • APRIL 1974

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TM X-2913		

4. Title and Subtitle	5. Report Date
COMPUTER PROGRAM FOR SUPERSONIC KERNEL-FUNCTION FLUTTER ANALYSIS OF THIN LIFTING SURFACES	April 1974
	6. Performing Organization Code
7. Author(s)	8. Performing Organization Report No.
Herbert J. Cunningham	L-9126
	10. Work Unit No.
9. Performing Organization Name and Address	501-22-04-01
NASA Langley Research Center Hampton, Va. 23665	11. Contract or Grant No.
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	Technical Memorandum
National Aeronautics and Space Administration Washington, D.C. 20546	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

This report describes a computer program (program D2180) that has been prepared to implement the analysis described in NASA TN D-6012 for calculating the aerodynamic forces on a class of harmonically oscillating planar lifting surfaces in supersonic potential flow. The planforms treated are the delta and modified-delta (arrowhead) planforms with subsonic leading and supersonic trailing edges, and (essentially) pointed tips. The resulting aerodynamic forces are applied in a Galerkin modal flutter analysis. The required input data are the flow and planform parameters including deflection-mode data, modal frequencies, and generalized masses.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Flutter analysis Aeroelasticity Supersonic kernel function Computer program	Unclassified – Unlimited STAR Category 32

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	~~70~~ 72	$3.75

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

CONTENTS

	Page
SUMMARY	1
INTRODUCTION	1
SYMBOLS	1
PROBLEM DESCRIPTION	4
PROGRAM ORGANIZATION	5
Overlay Structure	5
MAIN Labelled COMMON	5
D2180 Subprogram Descriptions, Flow Charts, and Listings	6
OVERLAY (JMF,0,0); MAIN	7
OVERLAY (JMF,1,0)	8
D2181	8
PART1	9
KERNEL	18
MTXMPY	20
SSSSS	24
GAUSS	24
LCOMP	24
PRT2	25
GLSP	28
OVERLAY (JMF,2,0); D2182	31
Description of External File TAPE9	42
USAGE	42
Program Information and Input Deck Arrangement	42
Input Description	44
Listing of Input Data for Sample Case	47
Discussion of Printed Output	48
Discussion of Punched Output	49
OUTPUT FOR SAMPLE CASE	50
APPENDIX – EVALUATION OF THE SUPERSONIC KERNEL FUNCTION	64
REFERENCES	68

PRECEDING PAGE BLANK NOT FILMED

COMPUTER PROGRAM FOR SUPERSONIC KERNEL-FUNCTION FLUTTER ANALYSIS OF THIN LIFTING SURFACES

By Herbert J. Cunningham
Langley Research Center

SUMMARY

This report describes a computer program (program D2180) that has been prepared to implement the analysis described in NASA TN D-6012 for calculating the aerodynamic forces on a class of harmonically oscillating planar lifting surfaces in supersonic potential flow. The planforms treated are the delta and modifed-delta (arrowhead) planforms with subsonic leading and supersonic trailing edges, and (essentially) pointed tips. The resulting aerodynamic forces are applied in a Galerkin modal flutter analysis. The required input data are the flow and planform parameters including deflection-mode data, modal frequencies, and generalized masses.

INTRODUCTION

References 1 and 2 present the analytical background for, and results from, the flutter analysis of thin lifting surfaces based on a supersonic kernel function procedure. The present report describes the current version of the computer program used to obtain such results, and the user should have access to references 1 and 2 for an adequate understanding of the program. An appendix describes the evaluation of the supersonic kernel function that is used.

SYMBOLS

$a_{nm}^{(j)}$	weighting factor for the term (denoted by subscript n,m) in series for Δp_j
b_0	semichord length at root or plane of symmetry
g	modal-independent damping coefficient
g_i	coefficient of structural, solid-friction damping for mode i
h_i	amplitude of natural mode-shape deflection for mode i

III	transfer matrix
I_c	elements of integrating matrices for chordwise integrations
K	kernel function of integral equation (1)
k	reduced frequency with reference length b_0, $k = \frac{\omega b_0}{V}$
l_n^*	distribution function in lifting pressure series, $n = 1, 2, \ldots$
M	Mach number of stream flow
m_{ij}	generalized mass
$\Delta p, \Delta p_j$	lifting pressure, general, and for mode j, respectively
Q_{ij}, Q_{ij}^*	dimensional and nondimensional generalized aerodynamic forces
q_i, q_j	generalized coordinates of motion for modes i and j
Q_n	integrals ($n = 1$ to 5) (see eqs. (A7) to (A12))
R	region of integration on wing (eq. (1))
Re( ),Im( )	real and imaginary parts of ( )
s	value of spanwise coordinate at right-hand wing tip
t	time
V	velocity of undisturbed stream flow
w	instantaneous downwash at wing surface, positive with z axis
x	chordwise coordinate
$\bar{x}$	nondimensional local section chordwise coordinate, referred to local chord
x_c	x coordinate of points at which integrands are evaluated for numerical chordwise integration, where $c = 1, 2, \ldots$

x_{le},x_{te} local values of chordwise coordinate at leading and trailing edges, respectively

$x_0 = x - \xi$

y spanwise coordinate

$y_0 = y - \eta$

y_σ y coordinate of span stations at which chordwise integrations are done for subsequent spanwise integration, where $\sigma = 1, 2, \ldots$

α mass of air contained in volume $4\pi b_0^3$, $\alpha \equiv 4\pi\rho b_0^3$

$\beta = \sqrt{M^2 - 1}$

δ width of subregion III (see fig. 3 of ref. 1)

η dummy variable for coordinate y

η_R,η_L limits of integration (ref. 1)

θ chordwise coordinate (ref. 1)

ξ dummy variable for coordinate x

ξ_{le} local value of ξ on leading edge

ρ air density

τ dummy variable of integration

ω circular frequency of oscillation

ω_B chosen base or reference frequency

ω_i natural frequency of mode i

PROBLEM DESCRIPTION

For oscillating and steady thin lifting surfaces in supersonic flow, the linear integral equation that relates the distributions of downwash and lifting pressure on such surfaces is (see eq. (1) of ref. 1 and eq. (6) of ref. 2)

$$-\frac{w(x,y,t)}{V} = \frac{(2b_0)^2}{4\pi\rho V} \iint_R \Delta p(\xi,\eta,t)\ K(M,k,x_0,y_0)\ d\xi\ d\eta \tag{1}$$

where the region of integration R is the area of the wing surface bounded by the forward-facing Mach cone with its apex at x,y. Generally, the downwash $w(x,y,t)$ is known but the lifting pressure distribution $\Delta p(\xi,\eta,t)$ is not known. A finite series is assumed for Δp and each term in the series includes an unknown weighting factor. (See equations (11) to (17) of ref. 1.) Equation (1) is solved by collocation or by a least-squares solution technique.

The surface integrations and the singularity extractions indicated in equation (1) are performed mainly by numerical quadrature, with some aid from closed-form integration. The surface integrals are evaluated and the simultaneous equations are solved for the weighting factors and hence the pressure distribution under the control of subprogram D2181.

Subprogram D2182 uses the results from subprogram D2181 plus the input modal deflection data to compute the generalized aerodynamic force elements (see eqs. (3), (5), and (11) of ref. 2)

$$\left.\begin{aligned} Q_{ij} &= 2b_0 \int_0^s dy \int_{x_{le}}^{x_{te}} h_i(x,y)\ \frac{\Delta p_j(x,y,t)}{e^{i\omega t}}\ dx = 4\pi\rho V^2 b_0 Q_{ij}^* \\ Q_{ij}^* &= \int_0^s dy \int_{x_{le}}^{x_{te}} h_i \left\lfloor l_n{}^* y^m \right\rfloor \left[III\right] \left\{a_{nm}^{(j)}\right\} dx \end{aligned}\right\} \tag{2}$$

and then, for a sequence of input values of an air density parameter, to solve the flutter stability equations (see eqs. (A9) to (A13) of ref. 2)

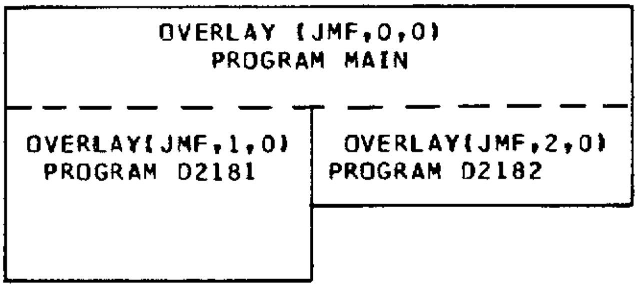
$$q_i\left[-\left(\frac{\omega_B}{\omega}\right)^2\left(1 + ig_i\right)\right] + \sum_j q_j\left[\frac{m_{ij}}{m_{ii}}\left(\frac{\omega_B}{\omega_i}\right)^2 + \frac{4\pi\rho b_0}{m_{ii}}\left(\frac{\omega_B}{\omega_i}\right)^2 \frac{Q_{ij}^*}{k^2}\right] = 0 \qquad (i = 1, 2, \ldots) \tag{3}$$

in searching for the value(s) of air density that correspond to neutrally stable motion(s). The value that corresponds to the lowest dynamic pressure is on a critical boundary, that is, a flutter boundary. Program execution continues for any particular case until a specified number of air density parameters is used. Succeeding cases (other planforms, modes, k-values, M-values, and so forth) are executed until an end of file is reached.

PROGRAM ORGANIZATION

Overlay Structure

This program is organized in overlays to reduce the required field length. The overlay structure is described by the following diagram:



MAIN Labelled COMMON

The following listing contains the FORTRAN variables that appear in labelled COMMON:

COMMON label	FORTRAN variable	Description
/TRA2182/	XM	Mach number of free stream
	XK	reduced frequency k
	TANLE	tangent of leading-edge sweep angle
	TANLTE	tangent of trailing-edge sweep angle
	S	ratio of wing half span to root-chord length
	NSYM	integer quantity: = 0 for spanwise antisymmetry, ≠ 0 for spanwise symmetry of lift distribution
	IDENT	identification information, up to 80 Hollerith characters
/DEFAULT/	HMLTPLY	two-word array, if nonzero, multiplies real and imaginary parts of downwash input, respectively

D2180 Subprogram Descriptions, Flow Charts, and Listings

Program D2180 is overlaid in order to reduce the maximum required field length. The main overlay 0,0 consists of subprogram MAIN. Primary overlay 1,0 is controlled by subprogram D2181, and primary overlay 2,0 is made up of subprogram D2182. There are no secondary overlays.

This section of the report presents a brief description of each subprogram, its function, flow chart, and listing. A list of the subprograms and their descriptions follows:

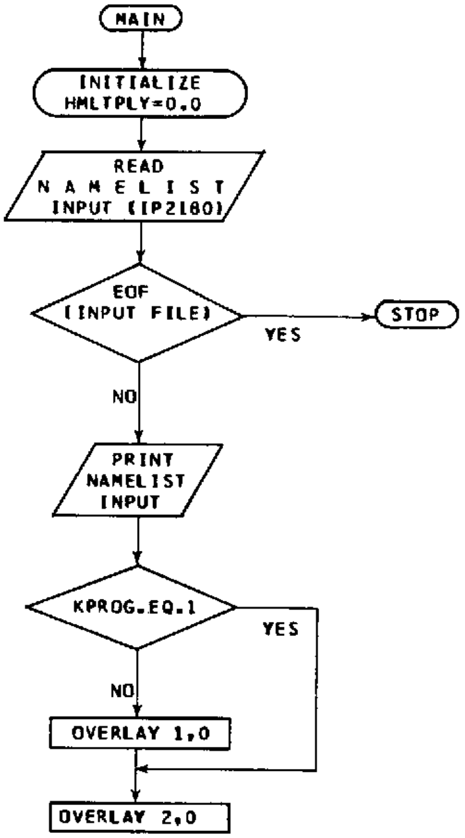
Subprogram	Description
MAIN	Reads a NAMELIST input parameter, writes a heading, and directs the access to overlay levels 1,0 and 2,0.
D2181	Writes headings, reads and writes IDENT input, reads and writes NAMELIST input, and calls PART1 and PRT2.
PART1	Computes elements of II_{nm} matrix (eq. (32) of ref. 1)
KERNEL	Computes $\bar{K}$ (KBAR) (eqs. (3) and (4) of ref. 1)
MTXMPY	Computes matrix III (eqs. (28) to (31) and table I of ref. 1), multiplies as in equation (28) of reference 1 to get II_{nm} row by row, and writes III matrix on TAPE9.
SSSSS	Stores span stations at which chordwise integrations are to be obtained and also stores the integrating factors for the spanwise integrations to obtain elements of II_{nm}^{*}.
GAUSS	For each subregion of surface integration, computes the span stations and spanwise Gaussian quadrature factors.
LCOMP	Computes the $l_n{}^*$ (eq. (13c), ref. 1)
PRT2	Optionally multiplies the real parts $-(\partial h_1/\partial x)$ and the imaginary parts $-2h_1$ of the downwash input array by HMLTPLY(1) and HMLTPLY(2), respectively; multiplies the imaginary part also by k; solves equation (32) of reference 1 for the $a_{nm}^{(j)}$ matrix; writes the $a_{nm}^{(j)}$ matrix and the matrix of residuals from the least-squares solution; optionally punches the $a_{nm}^{(j)}$ matrix elements.
GLSP	Solves $N \times M$ system of equations ($N \leqq M$) by least squares (for $N < M$), or as a linear system (for $N = M$)
SIMEQ	Solves set of linear simultaneous equations

Subprogram	Description
D2182	Prints heading, reads and writes IDENT input, reads and writes NAMELIST input, reads the III and $a_{nm}^{(j)}$ matrices from TAPE9, and computes the Q_{ij}^{*} of equation (11b) of reference 2; optionally writes matrices (2), (3), (4), (5), (6), (7), (8), and (8x) of equations (14), (16), and (18) of reference 2; optionally punches Q_{ij}^{*} matrix and seven associated parameters; writes matrix of h_i and span stations, $a_{nm}^{(j)}$, m_{ij}, ω_{ij}, g_i; writes Q_{ij}^{*}; solves equation (A13) of reference 2 for eigenvalues and eigenvectors; for each α, it writes α, flutter determinant, eigenvalues, eigenvectors, eigenfrequencies, g values, and stiffness parameters.

OVERLAY (JMF,0,0); MAIN

This is the only subprogram in the 0,0 overlay and its function is to read a single-parameter NAMELIST input IP2180, to write that parameter in a sentence, then to call overlays 1,0 and 2,0 in sequence if KPROG < 1, or to call only overlay 2,0 if KPROG = 1. The flow chart for subprogram MAIN follows.

MAJOR STEP FLOW CHART

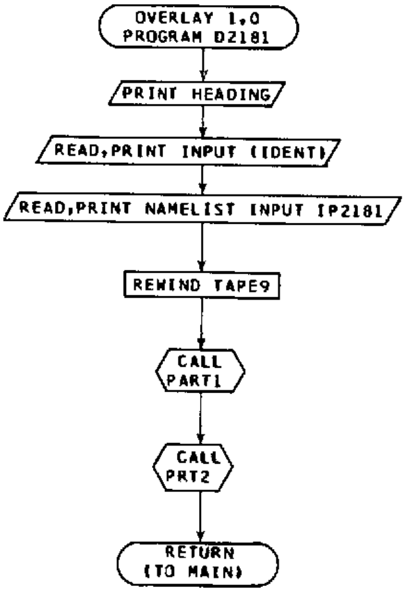


```
                                                                                ************
                                                                                * SEQUENCE *
                                                                                * FOR MODS *
                                                                                ************
      OVERLAY(JMF,0,0)                                                A  1         100000
      PROGRAM MAIN(INPUT=1,OUTPUT=1,PUNCH=2,TAPE9=1,TAPE5=INPUT,      A  1+        200000
     1TAPE6=OUTPUT)                                                   A  1+        300000
C                                                                     A   2        400000
C     PROGRAM CARD SPECIFIES MINIMUM BUFFERS TO REDUCE FIELD LENGTH   A   3        500000
C                                                                     A   4        600000
      DIMENSION IDENT(8), HMLTPLY(2)                                  A   5        700000
      COMMON/TRA2182/XM,XK,TANLE,TANLTE,S,NMODE,NSYM,KPROG            A   6        800000
      NAMELIST /IP2180/KPROG                                          A   7        900000
      COMMON/DEFAULT/HMLTPLY                                          A   8       1000000
      HMLTPLY(1)=0.0$HMLTPLY(2)=0.0                                   A   9       1100000
10    READ (5,IP2180)                                                 A  10       1200000
      IF (ENDFILE 5) 20,30                                            A  11       1300000
20    STOP 71                                                         A  12       1400000
30    PRINT 40, KPROG                                                 A  13       1500000
40    FORMAT (1H1,///51H ***BEGIN PROGRAM D2180, NAMELIST IP2180 IS (KPR  A  14       1600000
     1OG=,I1,1H))                                                     A  15       1700000
      IF (KPROG.EQ.1) GO TO 50                                        A  16       1800000
      CALL OVERLAY (3HJMF,1,0)                                        A  17       1900000
50    CONTINUE                                                        A  18       2000000
      CALL OVERLAY (3HJMF,2,0)                                        A  19       2100000
      GO TO 10                                                        A  20       2200000
      END                                                             A  21-      2300000
```

OVERLAY (JMF,1,0)

D2181.- Subprogram D2181 is the controlling subprogram in overlay 1,0. Its function is to direct the calculation of the weighting factors $a_{nm}^{(j)}$ of the lifting pressure series for each mode j of the downwash as described in reference 1. The subprogram first calls a NAMELIST input, and then calls PART1 and PRT2. Matrices III and a_{nm} are written on TAPE9 for subsequent communication to D2182. The flow chart for subprogram D2181 follows.

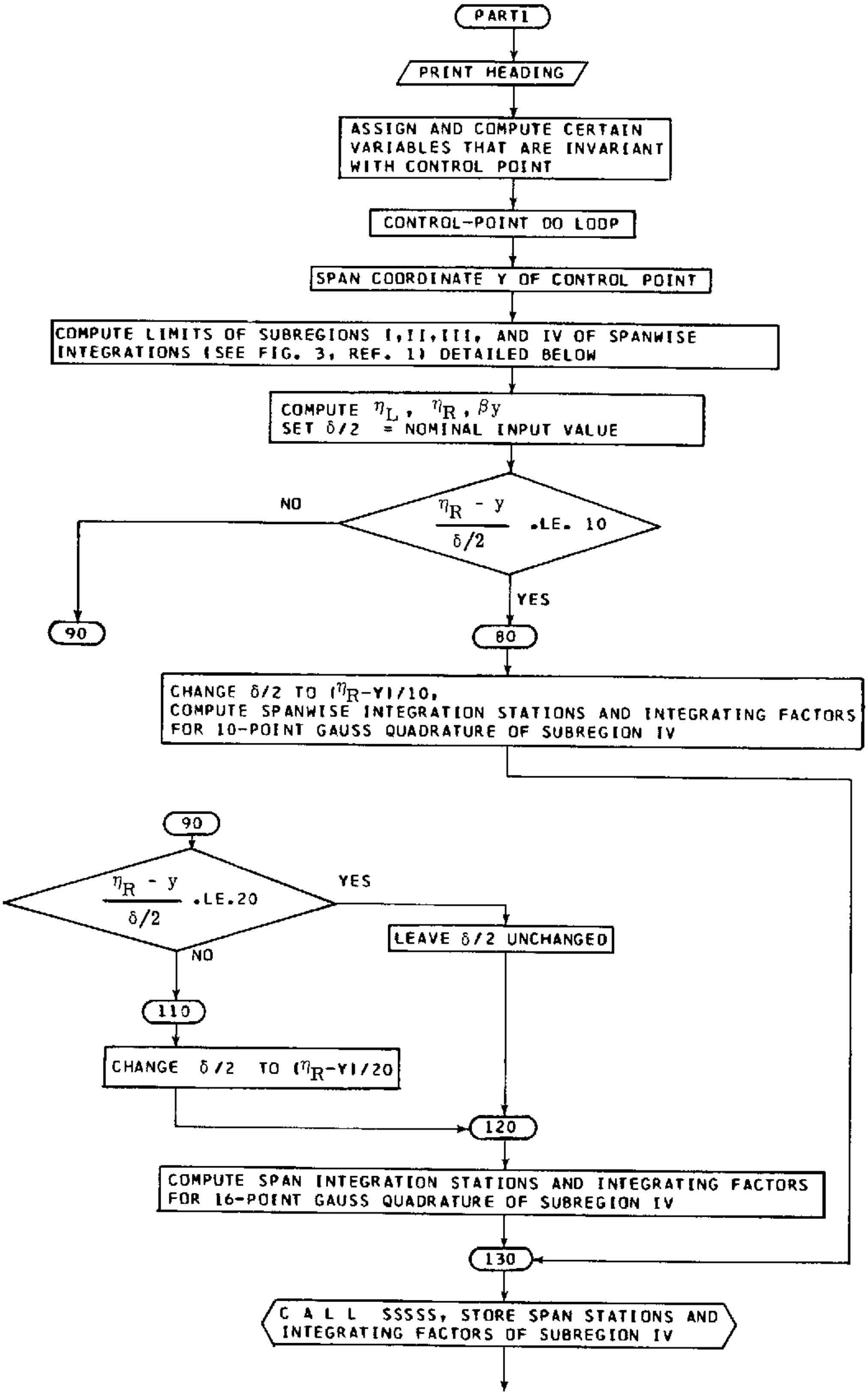
MAJOR STEPS IN PROGRAM D2181

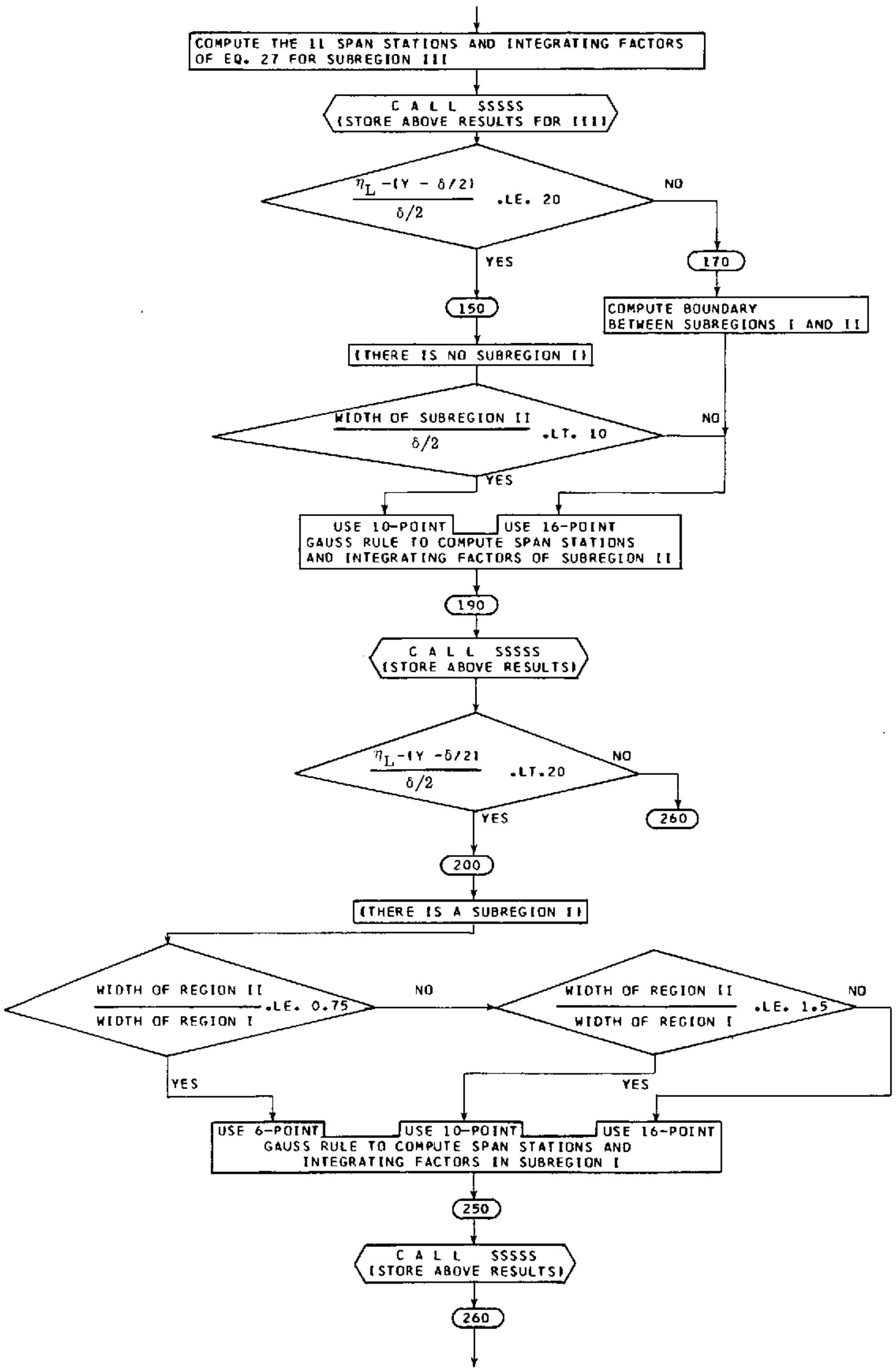


```
      OVERLAY(JMF,1,0)
      PROGRAM D2181
       DIMENSION X(48),Y(48),IDENT(8)
      DIMENSION DHX(48,6), H(48,6), HMLTPLY(2)
      COMPLEX XMAT2(48,16)
      DIMENSION WT(96)
      COMMON/TRA2182/XM,XK,TANLE,TANLTE,S,NMODE,NSYM,KPROG
      COMMON/DEFAULT/HMLTPLY
      COMMON NXY,DEL,X,Y,DHX,H,WT
      COMMON XA,XB,XC,XN4,XN5,XN6,XN7,XN8,XN9,BETASQ,B2Y01,X0,Y0,XK0B2,
     1XKMB2,XKAY02,XOTERM,SKAY0,XKY0SQ,XKREAL,XKIMAG,NKROW,XMAT2,KMPT
      NAMELIST /IP2181/XK,XM,TANLE,TANLTE,S,NXY,
     1NSYM,DEL,X,Y,DHX,H,NMODE,WT,HMLTPLY
      WRITE (6,10)
      WRITE (6,20)
10    FORMAT (1H0,///35X,45HNASA - LANGLEY RESEARCH CENTER - HAMPTON, VA
     1.///)
20    FORMAT (///10X,63HJEAN FOSTER FOR HERB CUNNINGHAM- D2180(INCLUDESD
     12181 AND D2182)///)
      WRITE (6,30)
30    FORMAT (20X,76HUNSTEADY LIFTING-SURFACE THEORY BY THE SUPERSONIC K
     1ERNEL FUNCTION METHOD AND A/,20X,76HGALERKIN MODAL FLUTTER ANALYSI
     2S OF ARROWHEAD PLANFORMS WITH SUBSONIC LEADING/20X,35HEDGES AND SU
     3PERSONIC TRAILING EDGES//,20X,45H***REF. 1 = AIAA JOUR., NOV.1966,
     4 P.1961-1968/,20X,26H***REF. 2 = NASA TN D-6012 /20X,60H***REF. 3
     5= NASA TM X-2913, THE DESCRIPTION OF THIS PROGRAM
C     DEFAULT VALUES OF DEL,NSYM, WT  ***
      DEL=0.04$NSYM=1$WT=-1.0
      READ (5,40) IDENT
40    FORMAT (8A10)
      WRITE (6,50) IDENT
50    FORMAT (/8A10)
      READ (5,IP2181)
      WRITE (6,60)
   60 FORMAT (//1X,32H ECHO OF NAMELIST IP2181 FOLLOWS )
      WRITE (6,IP2181)
      REWIND 9
C     TAPE9 IS A COMMUNICATION CHANNEL TO OVERLAY 2,0
      CALL PART1
      CALL PRT2
      RETURN
      END
```

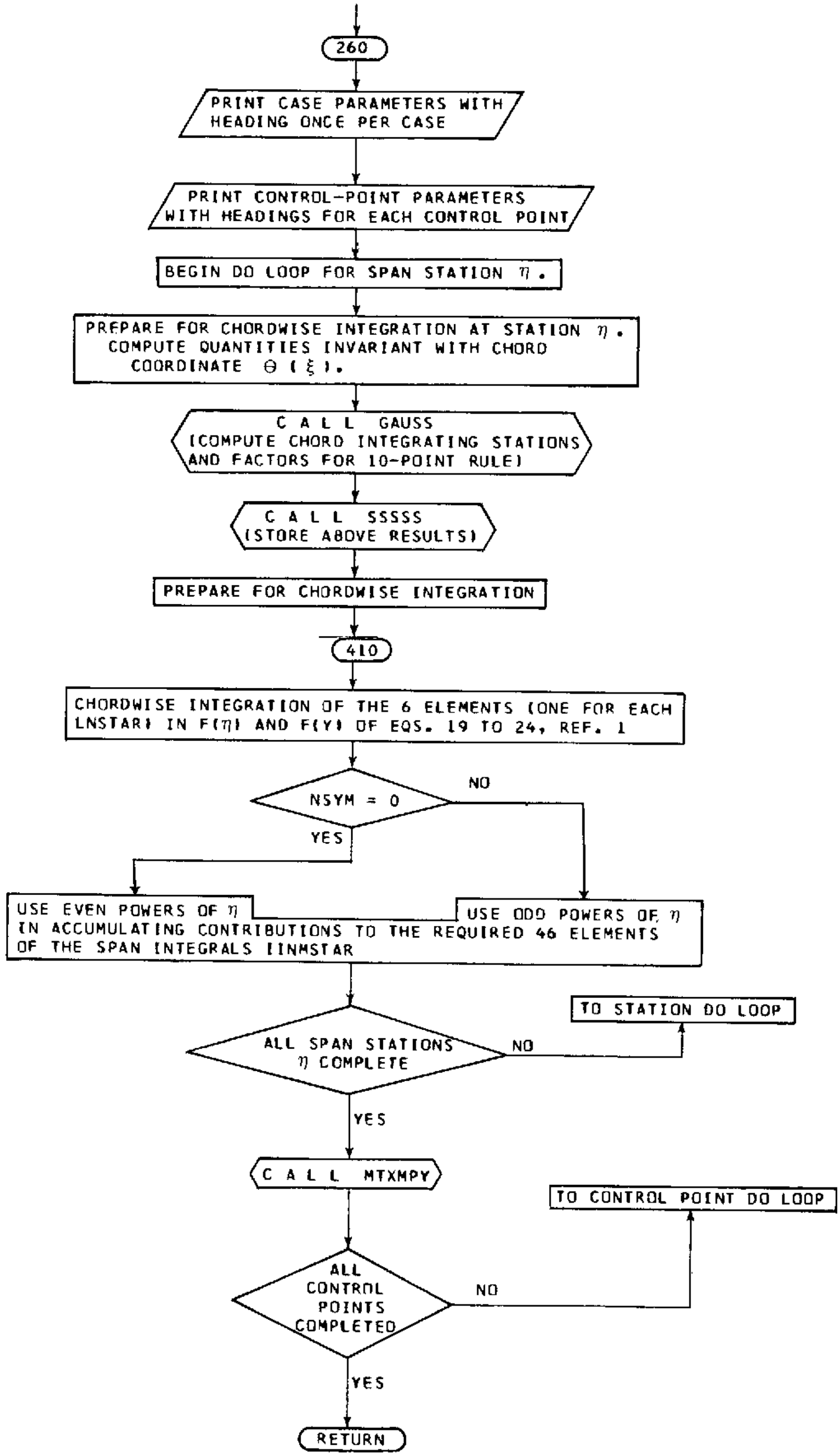
PART1.- The function of PART1 is to calculate the II_{nm} matrix of equation (28) of reference 1. By using a number of called subprograms, it first calculates the subdivision of the region of surface integration as shown in figure 3 of reference 1, determines for subregions I, II, and IV whether 6-, 10-, or 16-point Gaussian quadrature is employed, carries out the chordwise and spanwise quadrature to calculate the II^*_{nm} matrix, and multiplies the latter by the III matrix of equation (28) of reference 1. The flow chart for PART1 follows.

FLOW CHART FOR SUBROUTINE PART1





COMPUTE THE 11 SPAN STATIONS AND INTEGRATING FACTORS OF EQ. 27 FOR SUBREGION III
C A L L SSSSS (STORE ABOVE RESULTS FOR III)
$\frac{\eta_L - (Y - \delta/2)}{\delta/2}$ .LE. 20
NO
YES
170
COMPUTE BOUNDARY BETWEEN SUBREGIONS I AND II
150
(THERE IS NO SUBREGION I)
$\frac{\text{WIDTH OF SUBREGION II}}{\delta/2}$ .LT. 10
NO
YES
USE 10-POINT
USE 16-POINT
GAUSS RULE TO COMPUTE SPAN STATIONS AND INTEGRATING FACTORS OF SUBREGION II
190
C A L L SSSSS (STORE ABOVE RESULTS)
$\frac{\eta_L - (Y - \delta/2)}{\delta/2}$ .LT.20
NO
YES
260
200
(THERE IS A SUBREGION I)
$\frac{\text{WIDTH OF REGION II}}{\text{WIDTH OF REGION I}}$ .LE. 0.75
NO
YES
$\frac{\text{WIDTH OF REGION II}}{\text{WIDTH OF REGION I}}$ .LE. 1.5
NO
YES
USE 6-POINT
USE 10-POINT
USE 16-POINT
GAUSS RULE TO COMPUTE SPAN STATIONS AND INTEGRATING FACTORS IN SUBREGION I
250
C A L L SSSSS (STORE ABOVE RESULTS)
260


260
PRINT CASE PARAMETERS WITH HEADING ONCE PER CASE
PRINT CONTROL-POINT PARAMETERS WITH HEADINGS FOR EACH CONTROL POINT
BEGIN DO LOOP FOR SPAN STATION η.
PREPARE FOR CHORDWISE INTEGRATION AT STATION η. COMPUTE QUANTITIES INVARIANT WITH CHORD COORDINATE Θ (ξ).
CALL GAUSS (COMPUTE CHORD INTEGRATING STATIONS AND FACTORS FOR 10-POINT RULE)
CALL SSSSS (STORE ABOVE RESULTS)
PREPARE FOR CHORDWISE INTEGRATION
410
CHORDWISE INTEGRATION OF THE 6 ELEMENTS (ONE FOR EACH LNSTAR) IN F(η) AND F(Y) OF EQS. 19 TO 24, REF. 1
NSYM = 0
NO
YES
USE EVEN POWERS OF η
USE ODD POWERS OF η
IN ACCUMULATING CONTRIBUTIONS TO THE REQUIRED 46 ELEMENTS OF THE SPAN INTEGRALS (INMSTAR
ALL SPAN STATIONS η COMPLETE
NO
TO STATION DO LOOP
YES
CALL MTXMPY
ALL CONTROL POINTS COMPLETED
NO
TO CONTROL POINT DO LOOP
YES
RETURN

```
      SUBROUTINE PART1
C        COMPUTE IINM MATRIX OF EQS. 18,28, AND 32 AND OUTPUT IT
      COMPLEX XM2STR(48),XMAT2(48,16)
      DIMENSION IDENT(8)
      DIMENSION X(48),Y(48),RNTAB(11),TBL10(10),TBL6(6),
     1TBL16(16),RTBL6(6),RTBL10(10),RTBL16(16),
     2ETRAN(16),RTRAN(16),ETASTN(60),WGHSTN(60),RGSST(16),THESTN(16),
     3AL(6),FNSPR(6,3),FNSPI(6,3),FSI(6),FSR(6),FIPR(6),
     4FIPI(6),AN(10),FFIR(10,6),FFII(10,6),ANSR(48),ANSI(48),
     5ALMC(6),TABRN(16),THEWGH(16)
      DIMENSION DHX(48,6),H(48,6)
      DIMENSION WT(96)
      COMMON/TRA2182/XM,XK,TANLE,TANLTE,S,NMODE,NSYM,KPROG
      COMMON NXY,DEL,X,Y,DHX,H,WT
      COMMON XA,XB,XC,XN4,XN5,XN6,XN7,XN8,XN9,BETASQ,B2Y01,X0,Y0,XKOB2,
     1XKMB2,XKAY02,X0TERM,SKAY0,XKY0SQ,XKREAL,XKIMAG,NRROW,XMAT2,KMFT
      DATA RNTAB/
     1 5.93330950E-02, 4.66287340E-01, 2.57857770E-01, 1.43822960E+00,
     2 7.98243660E+00,-2.24082890E+01, 7.98243660E+00, 1.43822960E+00,
     3 2.57857770E-01, 4.66287340E-01, 5.93330950E-02/
      DATA TBL6/
     1 9.32469514E-01, 6.61209386E-01, 2.38619186E-01,-2.38619186E-01,
     2-6.61209386E-01,-9.32469514E-01/
      DATA TBL10/
     1 9.73906528E-01, 8.65063366E-01, 6.79409568E-01, 4.33395394E-01,
     2 1.48874338E-01,-1.48874338E-01,-4.33395394E-01,-6.79409568E-01,
     3-8.65063366E-01,-9.73906528E-01/
      DATA TBL16/
     1 9.89400934E-01, 9.44575023E-01, 8.65631202E-01, 7.55404408E-01,
     2 6.17876244E-01, 4.58016777E-01, 2.81603550E-01, 9.50125098E-02,
     3-9.50125098E-02,-2.81603550E-01,-4.58016777E-01,-6.17876244E-01,
     4-7.55404408E-01,-8.65631202E-01,-9.44575023E-01,-9.89400934E-01/
      DATA RTBL6/
     1 1.71324492E-01, 3.60761573E-01, 4.67913934E-01, 4.67913934E-01,
     2 3.60761573E-01, 1.71324492E-01/
      DATA RTBL10/
     1 6.66713443E-02, 1.49451349E-01, 2.19086362E-01, 2.69266719E-01,
     2 2.95524224E-01, 2.95524224E-01, 2.69266719E-01, 2.19086362E-01,
     3 1.49451349E-01, 6.66713443E-02/
      DATA RTBL16/
     1 2.71524594E-02, 6.22535239E-02, 9.51585116E-02, 1.24628971E-01,
     2 1.49595988E-01, 1.69156519E-01, 1.82603415E-01, 1.89450610E-01,
     3 1.89450610E-01, 1.82603415E-01, 1.69156519E-01, 1.49595988E-01,
     4 1.24628971E-01, 9.51585116E-02, 6.22535239E-02, 2.71524594E-02/
      WRITE (6,10)
10    FORMAT (1H0,45X,41H PART I OF D2181 *** GENERATE IINM MATRIX)
      PRINT 20
20    FORMAT (1H0/116H0              K                 TANLTE              TANLA
     1              S                MACH              DELNUM             NSYMMETRY
     2)
      PRINT 30, XK,TANLTE,TANLE,S,XM,DEL,NSYM
30    FORMAT (6E17.8,I12)
      PRINT 40
40    FORMAT (1H0,24H      NO. OF C.PTS.(=NXY))
      PRINT 50, NXY
50    FORMAT (I20)
      XA=-.329
      XB=-1.4067
      XC=-2.9
      XN1=-.101
      XN2=-.899
      XN3=-.09480933
      DELIP=DEL
      XM2=XM**2
      BETA=SQRT(XM2-1.0)
      XK2=XK*2.0
      BTANLE=BETA*TANLE
      BETASQ=BETA**2
      XKOB2=XK2/BETASQ
      XKMB2=XM*XKOB2
      DO 60 L=1,NXY
      DO 60 J=1,16
```

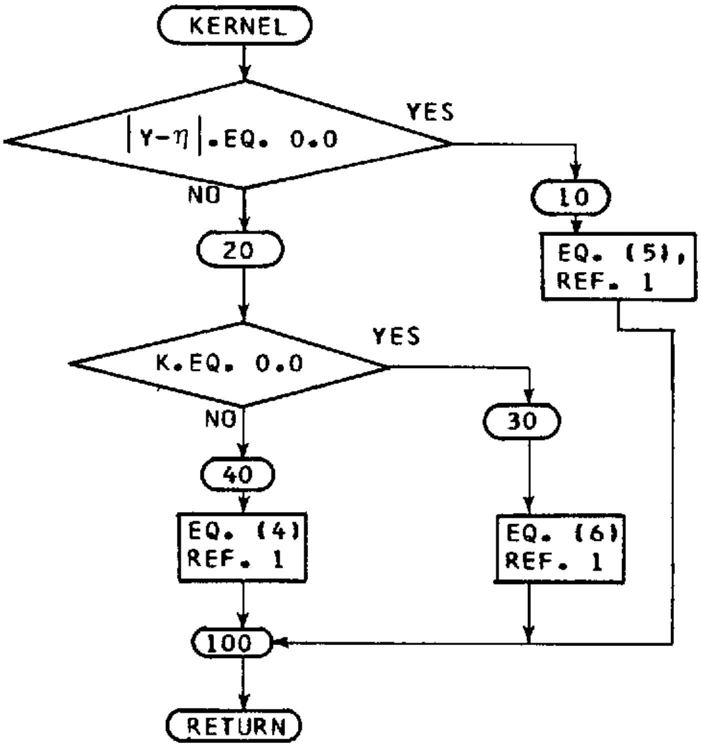
```
60    AMAT2(L,J)=(0.,0.)
      NXYCNT=1
C                          BEGIN CONTROL-POINT DO LOOP *****
C     SUBDIVIDE THE REGION OF INTEGRATION INTO SUBREGIONS I,II,III,IV
C     AS IN FIG. 3 OF REF. 1.  COMPUTE SPAN STATIONS AND SPAN
C     INTEGRATING FACTORS.
C
      DO 780 MQ=1,NXY
      DO 70 JN=1,46
      ANSR(JN)=0.0
70    ANSI(JN)=0.0
      IC2=1
      YTERM=Y(MQ)
      BYMQ=BETA*Y(MQ)
      ETAL=(BYMQ-X(MQ))/BTANLE
      ETAR=(BYMQ+X(MQ))/BTANLE
      DELN2=DEL/2.0
      AE=(ETAR-Y(MQ))/DELN2
      IF (AE-10.0) 80,80,90
80    DEL2=(ETAR-Y(MQ))/10.0
C        PREPARE SUBREGION IV  ***
      ETA5=ETAR
      ETA4=Y(MQ)+DEL2
      ICOUNT=10
      NPTDSK=5
      CALL GAUSS (10,ETA4,ETA5,TBL10(1),RTBL10(1),YTERM,1,ETRAN(1),RTRAN
     1(1))
      GO TO 130
90    IF (AE-20.0) 100,100,110
100   DEL2=DELN2
      ETA4=Y(MQ)+DEL2
      ETA5=ETAR
      GO TO 120
110   DEL2=(ETAR-Y(MQ))/20.0
      ETA4=Y(MQ)+DEL2
      ETA5=ETAR
120   ICOUNT=16
      NPTDSK=8
      CALL GAUSS (16,ETA4,ETA5,TBL16(1),RTBL16(1),YTERM,1,ETRAN(1),RTRAN
     1(1))
130   DSR=ETA5-ETA4
      CALL SSSSS (IC2,ICOUNT,ETASTN(1),WGHSTN(1),ETRAN(1),RTRAN(1))
C        PREPARE SUBREGION III ***
      IC2=ICOUNT+1
      ETA3=Y(MQ)-DEL2
      WIDTH=ETA4-ETA3
      W=WIDTH/10.0
      RG3ST(6)=Y(MQ)
      MK=7
      JK=5
      TABRN(6)=(1.0/DEL2)*RNTAB(6)
      DO 140 LK=1,5
      NK=JK+1
      ML=MK-1
      RG3ST(JK)=RG3ST(NK)+W
      RG3ST(MK)=RG3ST(ML)-W
      TABRN(JK)=(1.0/DEL2)*RNTAB(JK)
      TABRN(MK)=(1.0/DEL2)*RNTAB(MK)
      JK=JK-1
140   MK=MK+1
      ICOUNT=ICOUNT+11
      CALL SSSSS (IC2,ICOUNT,ETASTN(1),WGHSTN(1),RG3ST(1),TABRN(1))
C        PREPARE SUBREGION II  ***
      IC2=ICOUNT+1
      DL=ETA3-ETAL
      IF ((DL/DEL2)-20.0) 150,150,170
C          THERE IS NO SUBREGION I **
150   ETA2=ETAL
      D1L=0
      D2L=ETA3-ETA2
      IF ((DL/DEL2)-10.0) 160,180,180
```

```
160   CALL GAUSS (10,ETA2,ETA3,TBL10(1),RTBL10(1),YTERM,1,ETRAN(1),RTRAN
     1(1))
      ICOUNT=ICOUNT+10
      NPTD2L=5
      NPTD1L=0
      GO TO 190
C        BOUNDARY OF SUBREGION I AND II ***
170   ETA2=Y(MQ)-19.0*DEL2
180   CALL GAUSS (16,ETA2,ETA3,TBL16(1),RTBL16(1),YTERM,1,ETRAN(1),RTRAN
     1(1))
      NPTD2L=8
      NPTD1L=0
      ICOUNT=ICOUNT+16
190   CALL SSSSS (IC2,ICOUNT,ETASTN(1),WGHSTN(1),ETRAN(1),RTRAN(1))
      IC2=ICOUNT+1
      IF ((DL/DEL2)-20.0) 260,260,200
C        PREPARE SUBREGION I   ***
200   ETA1=ETAL
      D1L=ETA3-ETA2
      D2L=ETA2-ETA1
      IF ((D2L/D1L)-.75) 210,210,220
210   CALL GAUSS (6,ETA1,ETA2,TBL6(1),RTBL6(1),YTERM,1,ETRAN(1),RTRAN(1)
     1)
      ICOUNT=ICOUNT+6
      NPTD2L=8
      NPTD1L=3
      GO TO 250
220   IF ((D2L/D1L)-1.5) 230,230,240
230   CALL GAUSS (10,ETA1,ETA2,TBL10(1),RTBL10(1),YTERM,1,ETRAN(1),RTRAN
     1(1))
      ICOUNT=ICOUNT+10
      NPTD2L=8
      NPTD1L=5
      GO TO 250
240   CALL GAUSS (16,ETA1,ETA2,TBL16(1),RTBL16(1),YTERM,1,ETRAN(1),RTRAN
     1(1))
      ICOUNT=ICOUNT+16
      NPTD2L=5
      NPTD1L=8
250   CALL SSSSS (IC2,ICOUNT,ETASTN(1),WGHSTN(1),ETRAN(1),RTRAN(1))
260   DSL=D1L+D2L
      WRITE (6,270)
270   FORMAT (1H0/96H            X                   Y                   DSL
     1          D1L            D2L            DSR)
      WRITE (6,280) X(MQ),Y(MQ),DSL,D1L,D2L,DSR
280   FORMAT (6E17.8)
      WRITE (6,290)
290   FORMAT (47H0      PTSD1L          PTSD2L          PTSDSR )
      WRITE (6,300) NPTD1L,NPTD2L,NPTDSR
300   FORMAT (I9,2I17)
      WRITE (6,310) DEL2
310   FORMAT (41H0  HALF-WIDTH CF CONTRCL POINT STRIP     E17.8)
C
C     COMPUTE QUANTITIES INVARIANT WITH CHORDWISE COORDINATE
C
      THELE=0
      THESTN(1)=THELE
      DO 760 ID=1,ICOUNT
      IDCAN=ID-1
      AETA=ABS(ETASTN(ID))
      YO=Y(MQ)-ETASTN(ID)
      AYO=ABS(YO)
      PSILE=AETA*TANLE
      PSIMC=X(MQ)-BETA*AYO
      PSITE=1.0+AETA*TANLTE
      PPP=PSITE+PSILE
      BBO=PSITE-PSILE
      BBO5=.5*BBO
      DUM1=(PPP-2.0*PSIMC)/BBO
      IF (DUM1) 320,380,330
320   NEG=-1
      GO TO 340
```

```
330   NEG=1
340   ADUM=ABS(DUM1)
      IF (ABS(ADUM-1.0)-.000001) 350,350,380
350   IF (NEG) 360,360,370
360   DUM1=-1.0
      GO TO 380
370   DUM1=1.0
380   SDUM1=ABS(SQRT(1.0-DUM1**2))
      THEMC=ATAN2(SDUM1,DUM1)
      XKAYO=XK*AYO
      XKAYO2=2.0*XKAYO
      XKAYO4=4.0*XKAYO
      XKYOSQ=XKAYO2**2
      XN4=(XN1/(XA**2+XKYOSQ))*XKAYO
      XN5=(XN2/(XB**2+XKYOSQ))*XKAYO
      XN6=3.1415926+XKAYO2
      XN7=3.1415926-XKAYO2
      XC2=XC**2
      XN8=(XN3/(2.0*(XC2+XN6**2)))*XKAYO
      XN9=(XN3/(2.0*(XC2+XN7**2)))*XKAYO
      B2YO1=1.0/(BETASQ*AYO)
      CALL GAUSS (10,THELE,THEMC,TBL10(1),RTBL10(1),0,0,ETRAN(1),RTRAN(1
     1))
      NPSI=5
      JC2=3
      JCOUNT=12
      CALL SSSSS (JC2,JCOUNT,THESTN(1),THEWGH(1),ETRAN(1),RTRAN(1))
      THESTN(2)=THEMC
      IF (YO) 390,400,390
C          FOR EQS. 22,23  REF. 1 ***
390   XS=X(MQ)-PSIMC
      BY=BETASQ*(YO**2)
      THAT=(XK2*XS*XM2)/BETASQ
      COST=COS(THAT)
      SINT=SIN(THAT)
      GO TO 410
400   BY=0
410   DO 530 IE=2,13
C          IN DO LOOP 1ST TIME FOR XSILE , 2ND TIME FOR XSIMC, 3RD TO
C           12TH TIME FOR 10-PT. GAUSS QUADRATURE  ***
      NLOW=IE-1
      SINTH=SIN(THESTN(NLOW))
      PSI=.5*(PPP-(880)*COS(THESTN(NLOW)))
      XO=X(MQ)-PSI
      STHE=THESTN(NLOW)
      IF (NLOW-2) 420,430,420
420   XOTERM=XO/SQRT(XO**2-BY)
      XHERE=XOTERM*COST
      THERE=XOTERM*SINT
C          XSILE   TERMS ARE COMPLETE  ***
      IF (NLOW-1) 430,530,430
430   CALL LCOMP (PSI,TANLE,PSILE,AL(1),STHE)
      IF (NLOW-2) 530,450,440
440   CALL KERNEL
      IF (NLOW-3) 530,470,490
450   DO 460 JI=1,6
460   ALMC(JI)=AL(JI)
C          XSIMC   TERMS ARE COMPLETE  ***
      GO TO 530
470   DO 480 LOW=1,6
      FNSPR(LOW,3)=0
      FNSPI(LOW,3)=0
480   CONTINUE
490   CONTINUE
      DO 520 LOW=1,6
      IF (YO) 500,510,500
C          ACCUMULATE INTEGRAL OF EQ 23, REF. 1 ***
500   FNSPR(LOW,3)=FNSPR(LOW,3)+(THEWGH(NLOW))*(XKREAL*AL(LOW)+ALMC(LOW)
     1*XHERE)*SINTH
      FNSPI(LOW,3)=FNSPI(LOW,3)+(THEWGH(NLOW))*(XKIMAG*AL(LOW)-ALMC(LOW)
     1*THERE)*SINTH
      GO TO 520
```

```
C          ACCUMULATE INTEGRAL OF EQ 24, REF. 1 ***
510     FNSPR(LOW,3)=FNSPR(LOW,3)+(THEWGH(NLOW))*XKREAL*AL(LOW)*SINTH
        FNSPI(LOW,3)=FNSPI(LOW,3)+(THEWGH(NLOW))*XKIMAG*AL(LOW)*SINTH
520     CONTINUE
530     CONTINUE
        IF (Y0) 540,560,540
C           FOR EQ. 22, REF. 1 ***
540     XT=(X(MQ)-PSILE)**2
        TRY=SQRT(XT-BY)
        WHR=TRY*COST
        WHI=TRY*SINT
        DO 550 ICAN=1,6
        FSR(ICAN)=-(ALMC(ICAN))*WHR
550     FSI(ICAN)=(ALMC(ICAN))*WHI
        GO TO 580
560     DO 570 IDAN=1,6
        FSR(IDAN)=0
570     FSI(IDAN)=0
580     DO 590 JC=1,6
        FIPR(JC)=BB05*FNSPR(JC,3)+FSR(JC)
590     FIPI(JC)=BB05*FNSPI(JC,3)+FSI(JC)
C           CHORDWISE INTEGRAL F(ETA) OR F(Y) COMPLETE   ****
C           BEGIN MULTIPLICATION BY POWERS OF ETA AS IN EQ. 18, REF. 1
        IF (NSYM) 620,600,620
600     NIN=1
        DO 610 IN=1,19,2
        AN(NIN)=(ETASTN(ID))**(IN)
610     NIN=NIN+1
        GO TO 670
620     MIN=1
        DO 660 IM=1,19,2
        IF (ETASTN(ID)) 650,630,650
630     IF (IM-1) 650,640,650
640     AN(MIN)=1.0
        GO TO 660
650     AN(MIN)=(ETASTN(ID))**(IM-1)
660     MIN=MIN+1
670     JKN=0
C         COMPUTE 46 TERMS OF IINMSTAR, EQ. 28, REF.1 ****
        DO 750 JCAN=1,6
        IF (JCAN-2) 680,690,700
C           10 POWERS OF ETA FOR N=1 AND 3, 8 POWERS FOR N=2 AND 4,
C            6 FOR N=5, 4 FOR N=6. ***
680     NTM=10
        GO TO 740
690     NTM=8
        GO TO 740
700     IF (JCAN-4) 680,690,710
710     IF (JCAN-6) 720,730,730
720     NTM=6
        GO TO 740
730     NTM=4
740     DO 750 KCAN=1,NTM
        FFIR(KCAN,JCAN)=FIPR(JCAN)*AN(KCAN)
        FFII(KCAN,JCAN)=FIPI(JCAN)*AN(KCAN)
        JKN=JKN+1
        ANSR(JKN)=WGHSTN(ID)*FFIR(KCAN,JCAN)+ANSR(JKN)
        ANSI(JKN)=WGHSTN(ID)*FFII(KCAN,JCAN)+ANSI(JKN)
750     CONTINUE
760     CONTINUE
        DO 770 JJ=1,JKN
770     XM2STR(JJ)=CMPLX(ANSR(JJ),ANSI(JJ))
        KMPT=NXYCNT
C         COMPUTE MATRIX IINM OF EQ. 28, REF. 1 ***
        CALL MTXMPY (XM2STR(1))
780     NXYCNT=NXYCNT+1
C           END OF CONTROL-POINT DO LOOP   ***
        NXY2=NXY*2
        RETURN
        END
```

KERNEL.- The function of subprogram KERNEL is to calculate $\overline{K}$ of the supersonic kernel function of equations (3) to (6) of reference 1. The evaluation of the integral in $\overline{K}$ is based on the approximation of the integrand as in equation (A5) of the appendix. The flow chart for KERNEL follows.

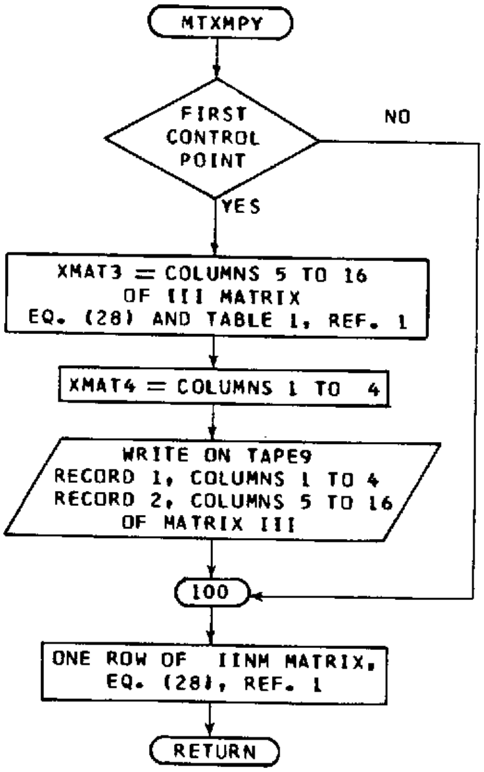



```
      SUBROUTINE KERNEL
C
C        COMPUTE THE KBAR OF EQS. 3 AND 4, REF. 1
C        THE ALGORITHM IS DESIGNED TO MINIMIZE CALCULATIONS OF SINES,
C          COSINES, AND EXPONENTIALS. THE INTEGRAND OF THE TAU INTEGRAL
C          IS APPROXIMATED AS IN EQ. 21 OF NASA TR R-48. CHORDWISE
C          INVARIANT QUANTITIES ARE CALCULATED IN SUBROUTINE PART1 AND
C          PASSED VIA COMMON.
C
      DIMENSION WT(96)
      DIMENSION X(48),Y(48),IDENT(8)
      DIMENSION DHX(48,6),H(48,6)
      COMPLEX XMAT2(48,16)
      COMMON/TRA2182/XM,XK,TANLE,TANLTE,S,NMODE,NSYM,KPROG
      COMMON NXY,DEL,X,Y,DHX,H,WT
      COMMON XA,XB,XC,XN4,XN5,XN6,XN7,XN8,XN9,BETASQ,B2Y01,X0,Y0,XK0B2,
     1XKMB2,XKAY02,XOTERM,SKAY0,XKY0SQ,XKREAL,XKIMAG,NKROW,XMAT2,KMPT
      XKAY04=2.0*XKAY02
      IF (Y0) 20,10,20
10    DUMBA=(2.0*XK)*X0
      XKREAL=-COS(DUMBA)
      XKIMAG=SIN(DUMBA)
      GO TO 100
20    IF (XK) 40,30,40
30    XKTERM=X0**2-BETASQ*(Y0**2)
      XKREAL=-X0/SQRT(XKTERM)
```

```
      XKIMAG=0
      GO TO 100
40    SQRT1=SQRT(X0**2-BETASQ*(Y0**2))
      XOPL=X0+XM*SQRT1
      XOMI=X0-XM*SQRT1
      AXOMI=ABS(XOMI)
      TUP=B2Y01*XOPL
      ABTLO=B2Y01*AXOMI
      XN10=XKOB2*X0
      XN11=XKMB2*SQRT1
      PTUP=(3.1415926)*TUP
      PTLO=(3.1415926)*ABTLO
      ATU=XA*TUP
      ATL=XA*ABTLO
      BTU=XB*TUP
      BTL=XB*ABTLO
      CTU=XC*TUP
      CTL=XC*ABTLO
      EXP1=EXP(ATU)
      EXP2=EXP(ATL)
      EXP3=EXP(BTU)
      EXP4=EXP(BTL)
      EXP5=EXP(CTU)
      EXP6=EXP(CTL)
      CS1=COS(XN10)
      CS2=COS(XN11)
      SN1=SIN(XN10)
      SN2=SIN(XN11)
      XN12=CS1*CS2
      XN13=SN1*SN2
      CS3=XN12-XN13
      CS4=XN12+XN13
      XN14=SN1*CS2
      XN15=SN2*CS1
      SN3=XN14+XN15
      SN4=XN14-XN15
      CS5=COS(PTUP)
      CS6=COS(PTLO)
      SN5=SIN(PTUP)
      SN6=SIN(PTLO)
      IF (XOMI) 50,60,60
50    SN4=-SN4
      SN6=-SN6
C
C     REAL PART
C
60    RQ2=(SN3-SN4)/2.0
      RQ3=((XA*CS3+XKAY02*SN3)*EXP1-(XA*CS4+XKAY02*SN4)*EXP2)*XN4
      RQ4=((XB*CS3+XKAY02*SN3)*EXP3-(XB*CS4+XKAY02*SN4)*EXP4)*XN5
      XN20=CS5*CS3
      XN21=SN5*SN3
      CS7=XN20-XN21
      CS9=XN20+XN21
      XN22=CS6*CS4
      XN23=SN6*SN4
      CS8=XN22-XN23
      CS10=XN22+XN23
      XN24=SN5*CS3
      XN25=CS5*SN3
      SN7=XN24+XN25
      SN9=XN24-XN25
      XN26=SN6*CS4
      XN27=CS6*SN4
      SN8=XN26+XN27
      SN10=XN26-XN27
      RQ5=(((XC*SN7-XN6*CS7)*EXP5+(-XC*SN8+XN6*CS8)*EXP6)*XN8+((XC*SN9-X
     1N7*CS9)*EXP5+(-XC*SN10+XN7*CS10)*EXP6)*XN9)
C
C     IMAGINARY PART
C
      IF (XOMI) 80,70,70
```

```
70     XIQ2=-(-CS3+CS4)/2.0
       XIQ3=((-XA*SN3+XKAY02*CS3)*EXP1+(XA*SN4-XKAY02*CS4)*EXP2)*XN4
       XIQ4=((-XB*SN3+XKAY02*CS3)*EXP3+(XB*SN4-XKAY02*CS4)*EXP4)*XN5
       XIQ5=(((XC*CS7+XN6*SN7)*EXP5-(XC*CS8+XN6*SN8)*EXP6)*XN8+((XC*CS9+X
      1N7*SN9)*EXP5-(XC*CS10+XN7*SN10)*EXP6)*(-XN9))
       GO TO 90
80     XIQ2=-(-CS3-CS4+2.0)/2.0
       XIQ3=((-XA*SN3+XKAY02*CS3)*EXP1-(XA*SN4-XKAY02*CS4)*EXP2-XKAY04)*X
      1N4
       XIQ4=((-XB*SN3+XKAY02*CS3)*EXP3-(XB*SN4-XKAY02*CS4)*EXP4-XKAY04)*X
      1N5
       XIQ5=(((XC*CS7+XN6*SN7)*EXP5+(XC*CS8+XN6*SN8)*EXP6-2.0*XC)*XN8+((X
      1C*CS9+XN7*SN9)*EXP5+(XC*CS10+XN7*SN10)*EXP6-2.0*XC)*(-XN9))
90     RQ1=RQ2+RQ3+RQ4+RQ5
       XIQ1=XIQ2+XIQ3+XIQ4+XIQ5
C
C      INTEGRAL IN K BAR FINISHED
C
       XN31=XOTERM*CS2
       XN32=XN31*CS1-XIQ1
       XN33=-XN31*SN1+RQ1
       GS=2.0*XK*XO
       CS11=COS(GS)
       SN11=SIN(GS)
       XKREAL=-(XN32*CS11+XN33*SN11)
       XKIMAG=XN32*SN11-XN33*CS11
100    RETURN
       END
```

MTXMPY.- The function of subprogram MTXMPY is to compute the elements of the matrix III of table I of reference 1 and carry out the indicated matrix multiplication of equation (28) of reference 1. The flow chart for subprogram MTXMPY follows.



```
      SUBROUTINE MTXMPY (XM2STR)
C        COMPUTE THE III MATRIX  OF EQS. 28 TO 31,AND TABLE 1 OF REF. 1-
C          AND MULTIPLY AS IN EQ. 28 TO GET  IINM MATRIX OF EQS. 18,28,32
      REAL K4B,K4C,K4D,K4
      REAL KE,KF,K2B,K2C,K2D,K2
      DIMENSION WT(96)
      DIMENSION E(9),F(25),KE(9),KF(15)
      DIMENSION X(48),Y(48),IDENT(8)
      DIMENSION DHX(48,6),H(48,6)
      COMPLEX XMAT3(48,12),XMAT2(48,16),XM2STR(48)
      COMPLEX XMAT4(48,4)
      COMMON/TRA2182/XM,XK,TANLE,TANLTE,S,NMODE,NSYM,KPROG
      COMMON NXY,DEL,X,Y,DHX,H,WT
      COMMON XA,XB,XC,XN4,XN5,XN6,XN7,XN8,XN9,BETASQ,B2Y01,X0,Y0,XK0B2,
     1XKMB2,XKAY02,XOTERM,SKAY0,XKY0SQ,XKREAL,XKIMAG,NKROW,XMAT2,KMPT
      K2=2.0*XK
C     JUMP TO 1000 FOR OTHER THAN FIRST CONTROL POINT ******
      IF (KMPT-1) 10,10,100
C       COMPUTE ELEMENTS OF III MATRIX, EQS.29 TO 31 AND TABLE 1, REF.1
10    B11=0.5
      B12=0.21132486
      B22=0.78867514
      B13=0.11270166
      B23=0.5
      B33=0.88729834
      C11=0.75*(1.0/S)*(TANLE+TANLTE)
      C12=C11+(0.43301270/S)*(TANLE-TANLTE)
      C22=2.0*C11-C12
      C13=C11+(0.58094750/S)*(TANLE-TANLTE)
      C23=C11
      C33=2.0*C11-C13
      CNST=(-1.0/(3.0*S**2))
      D11=CNST*C11
      D12=CNST*C12
      D22=CNST*C22
      D13=CNST*C13
      D23=CNST*C23
      D33=CNST*C33
      E(1)=6.0
      E(2)=(-6.0*B22-6.0*B12)
      E(3)=(-6.0*C22-6.0*C12)
      E(4)=(-6.0*D22-6.0*D12)
      E(5)=(-6.0*B12)*(-B22)
      E(6)=-6.0*B12*(-C22)-6.0*C12*(-B22)
      E(7)=-6.0*B12*(-D22)-6.0*C12*(-C22)-6.0*D12*(-B22)
      E(8)=-6.0*C12*(-D22)-6.0*D12*(-C22)
      E(9)=-6.0*D12*(-D22)
C     F(1) TO F(25) HERE ARE G(1) TO G(25) IN REF. 1 ***
      F(1)=-20.0
      F(2)=20.0*B23+20.0*B13
      F(3)=20.0*C23+20.0*C13
      F(4)=20.0*D23+20.0*D13
      F(5)=20.0*B13*(-B23)
      F(6)=20.0*B13*(-C23)+20.0*C13*(-B23)
      F(7)=20.0*B13*(-D23)+20.0*C13*(-C23)+20.0*D13*(-B23)
      F(8)=20.0*C13*(-D23)+20.0*D13*(-C23)
      F(9)=20.0*D13*(-D23)
      F(10)=F(1)
      F(11)=F(2)-B33*F(1)
      F(12)=F(3)-C33*F(1)
      F(13)=F(4)-D33*F(1)
      F(14)=F(5)-B33*F(2)
      F(15)=F(6)-B33*F(3)-C33*F(2)
      F(16)=F(7)-B33*F(4)-C33*F(3)-D33*F(2)
      F(17)=F(8)-C33*F(4)-D33*F(3)
      F(18)=F(9)-D33*F(4)
      F(19)=-B33*F(5)
      F(20)=-B33*F(6)-C33*F(5)
      F(21)=-B33*F(7)-C33*F(6)-D33*F(5)
      F(22)=-B33*F(8)-C33*F(7)-D33*F(6)
      F(23)=-B33*F(9)-C33*F(8)-D33*F(7)
```

```
      F(24)=-C33*F(9)-D33*F(8)
      F(25)=-D33*F(9)
      DO 20 J=1,46
      DO 20 L=1,12
20    XMAT3(J,L)=(0.,0.)
      K2B=2.0*XK*(-B11)
      K2C=2.0*XK*(-C11)
      K2D=2.0*XK*(-D11)
      BSTAR=2.0*B11
      CSTAR=2.0*C11
      DSTAR=2.0*D11
      K4B=-4.0*XK*(-B11)
      K4C=-4.0*XK*(-C11)
      K4D=-4.0*XK*(-D11)
      K4=-4.0*XK
C       COMPUTE COLUMNS 5 TO 8 OF III MATRIX, TABLE 1 ***
      DO 30 J=1,4
      XMAT3(J,J)=CMPLX(BSTAR,0.)
      L=J+1
      XMAT3(L,J)=CMPLX(CSTAR,0.)
      L=L+1
      XMAT3(L,J)=CMPLX(DSTAR,0.)
      L=L+8
      XMAT3(L,J)=CMPLX(-2.0,0.)
      L=L+6
      XMAT3(L,J)=CMPLX(-2.0,K4B)
      L=L+1
      XMAT3(L,J)=CMPLX(0.,K4C)
      L=L+1
      XMAT3(L,J)=CMPLX(0.,K4D)
      L=L+8
30    XMAT3(L,J)=CMPLX(0.,K4)
      DO 40 J=1,9
40    KE(J)=2.0*XK*E(J)
      ETAN=E(1)*TANLE**2+E(6)
C       COMPUTE COLUMNS 9 TO 12 OF III MATRIX  ***
      DO 50 J=5,8
      L=J-4
      XMAT3(L,J)=CMPLX(E(5),0.)
      L=L+1
      XMAT3(L,J)=CMPLX(ETAN,0.)
      L=L+1
      XMAT3(L,J)=CMPLX(E(7),0.)
      L=L+1
      XMAT3(L,J)=CMPLX(E(8),0.)
      L=L+1
      XMAT3(L,J)=CMPLX(E(9),0.)
      L=L+6
      XMAT3(L,J)=CMPLX(E(2),0.)
      L=L+1
      XMAT3(L,J)=CMPLX(E(3),0.)
      L=L+1
      XMAT3(L,J)=CMPLX(E(4),0.)
      L=L+6
      XMAT3(L,J)=CMPLX(E(2),KE(5))
      L=L+1
      XMAT3(L,J)=CMPLX(E(3),KE(6))
      L=L+1
      XMAT3(L,J)=CMPLX(E(4),KE(7))
      L=L+1
      XMAT3(L,J)=CMPLX(0.,KE(8))
      L=L+1
      XMAT3(L,J)=CMPLX(0.,KE(9))
      L=L+6
      XMAT3(L,J)=CMPLX((3.0*E(1)),KE(2))
      L=L+1
      XMAT3(L,J)=CMPLX(0.,KE(3))
      L=L+1
      XMAT3(L,J)=CMPLX(0.,KE(4))
      L=L+6
50    XMAT3(L,J)=CMPLX(0.,KE(1))
      FTAN15=F(15)+F(10)*TANLE**2
```

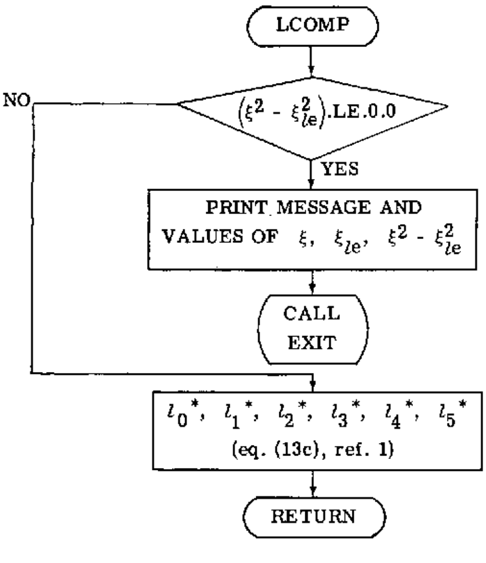
```
      FTAN20=F(20)+F(11)*TANLE**2
      FTAN21=F(21)+F(12)*TANLE**2
      FTAN22=F(22)+F(13)*TANLE**2
      DO 60 J=1,16
      JJ=J+9
60    KF(J)=2.0*XK*F(JJ)
C       COMPUTE COLUMNS 13 TO 16 OF III MATRIX ***
      DO 70 J=9,12
      L=J-8
      XMAT3(L,J)=CMPLX(F(19),0.)
      L=L+1
      XMAT3(L,J)=CMPLX(FTAN20,0.)
      L=L+1
      XMAT3(L,J)=CMPLX(FTAN21,0.)
      L=L+1
      XMAT3(L,J)=CMPLX(FTAN22,0.)
      L=L+1
      XMAT3(L,J)=CMPLX(F(23),0.)
      L=L+1
      XMAT3(L,J)=CMPLX(F(24),0.)
      L=L+1
      XMAT3(L,J)=CMPLX(F(25),0.)
      L=L+4
      XMAT3(L,J)=CMPLX(F(14),0.)
      L=L+1
      XMAT3(L,J)=CMPLX(FTAN15,0.)
      L=L+1
      XMAT3(L,J)=CMPLX(F(16),0.)
      L=L+1
      XMAT3(L,J)=CMPLX(F(17),0.)
      L=L+1
      XMAT3(L,J)=CMPLX(F(18),0.)
      L=L+4
      XMAT3(L,J)=CMPLX(F(14),KF(10))
      L=L+1
      XMAT3(L,J)=CMPLX(F(15),KF(11))
      L=L+1
      XMAT3(L,J)=CMPLX(F(16),KF(12))
      L=L+1
      XMAT3(L,J)=CMPLX(F(17),KF(13))
      L=L+1
      XMAT3(L,J)=CMPLX(F(18),KF(14))
      L=L+1
      XMAT3(L,J)=CMPLX(0.,KF(15))
      L=L+1
      XMAT3(L,J)=CMPLX(0.,KF(16))
      L=L+4
      XMAT3(L,J)=CMPLX((3.0*F(11)),KF(5))
      L=L+1
      XMAT3(L,J)=CMPLX((3.0*F(12)),KF(6))
      L=L+1
      XMAT3(L,J)=CMPLX((3.0*F(13)),KF(7))
      L=L+1
      XMAT3(L,J)=CMPLX(0.,KF(8))
      L=L+1
      XMAT3(L,J)=CMPLX(0.,KF(9))
      L=L+4
      XMAT3(L,J)=CMPLX((4.0*F(10)),KF(2))
      L=L+1
      XMAT3(L,J)=CMPLX(0.,KF(3))
      L=L+1
      XMAT3(L,J)=CMPLX(0.,KF(4))
      L=L+4
70    XMAT3(L,J)=CMPLX(0.,KF(1))
      XK2=2.*XK
      DO 80 KJ=1,4
      DO 80 JK=1,46
80    XMAT4(JK,KJ)=CMPLX(0.,0.)
C       COMPUTE COLUMNS 1 TO 4 OF III MATRIX   ***
      DO 90 KJ=1,4
90    XMAT4(KJ,KJ)=CMPLX(1.,0.)
      XMAT4(19,1)=CMPLX(0.,XK2)
```

```
      XMAT4(20,2)=CMPLX(0.,XK2)                                          E 217      76700000
      XMAT4(21,3)=CMPLX(0.,XK2)                                          E 218      76800000
      XMAT4(22,4)=CMPLX(0.,XK2)                                          E 219      76900000
C     TAPE9 IS A COMMUNICATION CHANNEL TO OVERLAY 2,0                    E 220      77000000
      WRITE (9) ((XMAT4(JK,KJ),JK=1,46),KJ=1,4)                          E 221      77100000
      WRITE (9) ((XMAT3(JK,KJ),JK=1,46),KJ=1,12)                         E 222      77200000
C        MULTIPLY, EQ. 28, REF. 1, RESULT IS THE IINM MATRIX ***          E 223      77300000
100   DO 110 J=1,4                                                       E 224      77400000
      RPT=REAL(XM2STR(J))-K2*AIMAG(XM2STR(J+18))                         E 225      77500000
      XIPT=AIMAG(XM2STR(J))+K2*REAL(XM2STR(J+18))                        E 226      77600000
110   XMAT2(KMPT,J)=CMPLX(RPT,XIPT)                                      E 227      77700000
      DO 130 LL=1,12                                                     E 228      77800000
      L=LL+4                                                             E 229      77900000
      DO 120 J=1,46                                                      E 230      78000000
120   XMAT2(KMPT,L)=XM2STR(J)*XMAT3(J,LL)+XMAT2(KMPT,L)                  E 231      78100000
130   CONTINUE                                                           E 232      78200000
      PRINT 140, KMPT                                                    E 233      78300000
140   FORMAT (1H0,4HROW ,I2,27H OF COEFFICIENT MATRIX IINM)              E 234      78400000
      WRITE (6,150)                                                      E 235      78500000
150   FORMAT (12X,2HRE,17X,4HIMAG,16X,2HRE,17X,4HIMAG,16X,2HRE,17X,4HIMA E 236      78600000
     1G)                                                                 E 237      78700000
      WRITE (6,160) (XMAT2(KMPT,J),J=1,16)                               E 238      78800000
160   FORMAT (6(5X,E15.8))                                               E 239      78900000
      RETURN                                                             E 240      79000000
      END                                                                E 241-     79100000
```

SSSSS.- The function of subprogram SSSSS is to store in sequence the span stations η and the associated spanwise integrating factors for subregions I, II, III, and IV of figure 3 of reference 1. No flow chart is needed.

GAUSS.- The function of subprogram GAUSS is to calculate the span stations η and select spanwise integrating factors for a 6-, 10-, or 16-point Gaussian quadrature in regions I, II, and IV of figure 3 of reference 1. No flow chart is needed.

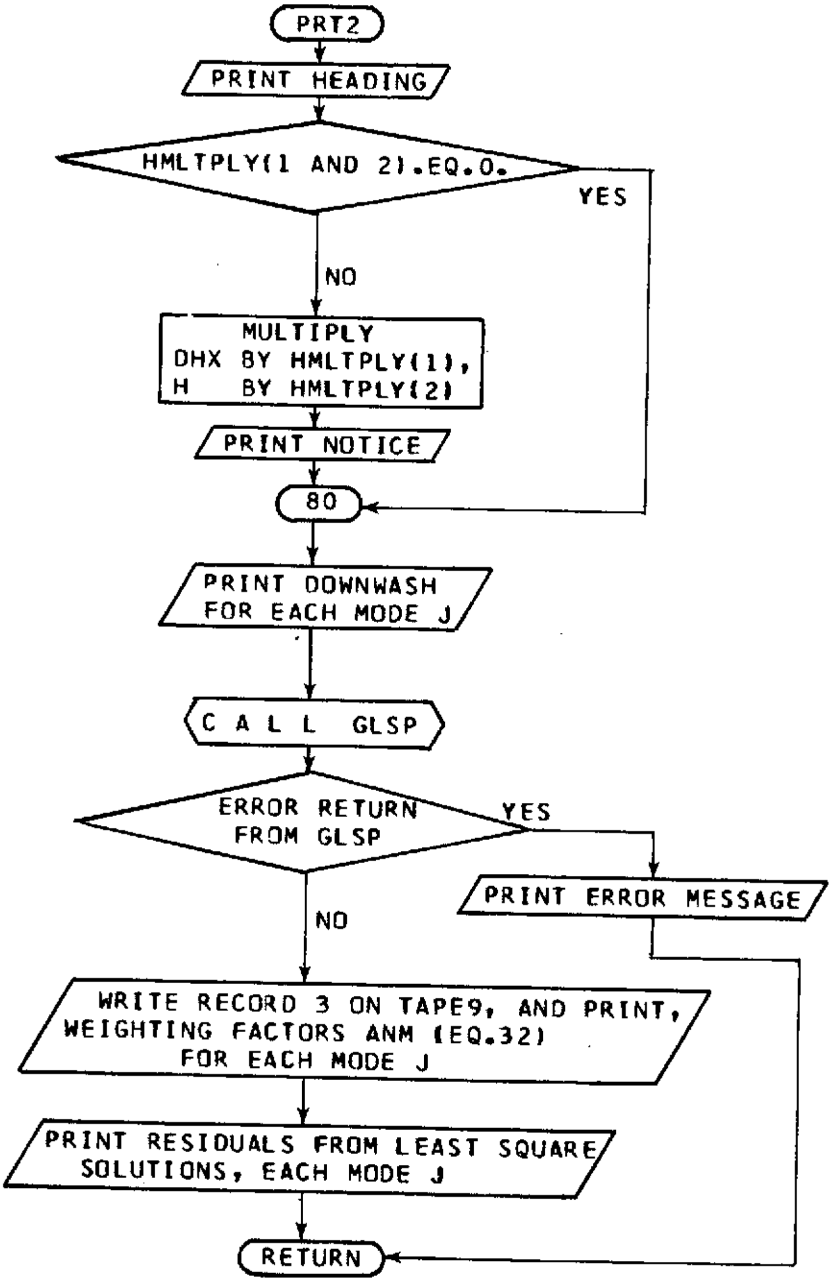
LCOMP.- The function of subprogram LCOMP is to calculate the values of $l_n^*(\xi,\eta)$ (n = 0,1,2,3,4,5) of equation (13c) of reference 1. The flow chart for subprogram LCOMP follows.



```
      SUBROUTINE SSSSS (MC,MCOUNT,STNETA,STNWGH,ETATRA,WGHTRA)
C
C     STORAGE OF SPAN STATIONS ETA AND THEIR ASSOCIATED SPANWISE
C      INTEGRATING FACTORS
C
      DIMENSIONSTNETA(60),STNWGH(60),ETATRA(16),WGHTRA(16)
      NAY=1
      DO 10 IB=MC,MCOUNT
      STNETA(IB)=ETATRA(NAY)
      STNWGH(IB)=WGHTRA(NAY)
   10 NAY=NAY+1
      RETURN
      END

      SUBROUTINE GAUSS (NGAUSS,A,B,U,R,YGAUSS,NQUEST,STATN,WGHFCT)
C
C        FOR NGAUSS=6,10, OR 16 COMPUTE SPAN STATIONS (STATN) AND
C         SPANWISE GAUSSIAN QUADRATURE FACTORS (WGHFCT) FOR EACH
C         SUBREGION, WITH LOWER LIMIT  A  AND UPPER LIMIT  B. ***
C
      DIMENSIONU(16),R(16),STATN(16),WGHFCT(16)
      BMA=(B-A)/2.0
      BPAT=(B+A)/2.0
      DO 30 I=1,NGAUSS
      STATN(I)=BMA*U(I)+BPAT
      IF (NQUEST) 20,10,20
   10 VALUE=1.0
      GO TO 30
   20 VALUE=(YGAUSS-STATN(I))**2
   30 WGHFCT(I)=BMA*R(I)/VALUE
      RETURN
      END

      SUBROUTINE LCOMP (DPSI,TAN,DPSILE,XLNO,DTHE)
C
C        COMPUTE THE LNSTAR (N=0  TO 5) OF EQ. 13C, REF. 1 ****
C
      DIMENSION XLNO(6)
      ANC=DPSI**2-DPSILE**2
      IF (ANC) 10,10,40
   10 WRITE (6,20)
   20 FORMAT (20H1ERROR IN LCOMP SQRT)
      WRITE (6,30) DPSI,DPSILE,ANC
   30 FORMAT (1H0/5HDPSI=E17.8,5X,7HDPSILE=E17.8,5X,4HANC=E17.8)
      CALL EXIT
   40 THIS=SQRT(ANC)
      DO 80 JJ=1,6
      IF (JJ-2) 50,60,70
   50 XLNO(JJ)=DPSI/THIS
      GO TO 80
   60 XLNO(JJ)=DPSI**2/THIS
      GO TO 80
   70 XLNO(JJ)=(DPSI**(JJ-3))*THIS
   80 CONTINUE
      RETURN
      END
```

<u>PRT2</u>.- The function of subprogram PRT2 is to solve equation (32) of reference 1 for the weighting factors $a_{nm}^{(j)}$. A heading is written. If HMLTPLY(1) and HMLTPLY(2) are both nonzero, the real part of the downwash input $\left(-\partial h_i/\partial x\right)$ is multiplied by HMLTPLY(1), and the imaginary part $\left(-2h_i\right)$ by HMLTPLY(2). The imaginary part is also multiplied by k. When there are more than 16 control points, the simultaneous equation set is solved by a least-squares subroutine GLSP. The resulting $a_{nm}^{(j)}$ matrix is printed, written on TAPE9 for communication to D2182, and can be optionally punched. The flow chart for PRT2 follows.

S U B R O U T I N E   PRT2



```
      SUBROUTINE PRT2
C
C        COMPUTE WEIGHTING FACTORS ANM(J) OF EQ 32, REF. 1 AND EQS. 9
C         AND 18, REF. 2 FOR A MINIMUM OF 16 AND A MAXIMUM OF 48
C          CONTROL POINTS, USING A LEAST-SQUARES SOLUTION.
C          WRITE ANM(J) MATRICES AND MATRICES OF LEAST-SQUARES RESIDUALS
C          PUNCH ANM MATRICES IF NPUNCH = 1    ***
C
      COMMON/DEFAULT/HMLTPLY
      COMMON/TRA2182/XM,XK,TANLE,TANLTE,S,NMODE,NSYM,KPROG
      COMMON NXY,DEL,X,Y,DHX,H,WT
      COMMON XA,XB,XC,XN4,XN5,XN6,XN7,XN8,XN9,BETASQ,B2Y01,X0,Y0,XKOB2,
     1XKMB2,XKAYO2,XOTERM,SKAYO,XKYOSQ,XKREAL,XKIMAG,NRROW,XMAT2,KMPT
      DIMENSION DCOEF(96,32),BVECT(96,6),WFACT(32,6),U(96,6),SUM(6),WT(9
     16),TB(96,32),C(32,32)
      DIMENSION DHX(48,6),H(48,6),HR(48,6)
      DIMENSION HMLTPLY(2)
      DIMENSION X(48),Y(48),IDENT(8)
C     ABOVE DIMENSIONS FOR A MAXIMUM OF 48 CONTROL POINTS AND 6 MODES **
      COMPLEX XMAT2(48,16)
      COMPLEX ANM(16,6)
      WRITE (6,10)
10    FORMAT (///37X,58H PART II OF 02181 *** SOLVE SIMULTANEOUS EQUATIO
     1NS FOR ANM///)
      IF (HMLTPLY(1).EQ.0.0.AND.HMLTPLY(2).EQ.0.0) GO TO 40
      DO 20 J=1,NMODE
      DO 20 I=1,NXY
      H(I,J)=HMLTPLY(2)*H(I,J)
      DHX(I,J)=HMLTPLY(1)*DHX(I,J)
20    CONTINUE
      PRINT 30, HMLTPLY(1),HMLTPLY(2)
30    FORMAT (//,5X,75H*****NOTICE- INPUT DOWNWASH QUANTITIES HAVE BEEN
     1MULTIPLIED AS FOLLOWS - BY,F6.3,25H  FOR THE SLOPES DHX, AND/10X,3
     2H BY,F6.3,30H  FOR THE DEFLECTIONS  H   ****,//)
40    NXY2=2*NXY
      DO 50 LXY=1,NXY
      DO 50 J=1,16
      ARG1=REAL(XMAT2(LXY,J))
      ARG2=AIMAG(XMAT2(LXY,J))
      DCOEF(LXY,J)=ARG1
      LN=LXY+NXY
      DCOEF(LN,J)=ARG2
      JN=J+16
      DCOEF(LXY,JN)=-ARG2
50    DCOEF(LN,JN)=ARG1
      DO 80 MODE=1,NMODE
      DO 60 J=1,NXY
      HR(J,MODE)=XK*H(J,MODE)
      BVECT(J,MODE)=DHX(J,MODE)
      L=NXY+J
60    BVECT(L,MODE)=HR(J,MODE)
      WRITE (6,70) MODE
70    FORMAT (//,1X,19HDOWNWASH W/V, MODE(,I2,50H), NXY REAL PARTS FIRST
     1, IN CONTROL POINT SEQUENCE//)
80    WRITE (6,90) (BVECT(J,MODE),J=1,NXY2)
90    FORMAT (6(5X,E15.8))
      CALL GLSP (DCOEF,NXY2,32,BVECT,NMODE,WFACT,U,SUM,WT,ERRET,TB,C,96,
     132)
      IF (ERRET-1) 140,100,120
100   PRINT 110
110   FORMAT (//1X,70H**** ERROR RETURN FROM LEAST SQUARES ROUTINE, FEWE
     1R THAN 16 EQNS. ****//)
      GO TO 190
120   PRINT 130
130   FORMAT (//1X,66H**** ERROR RETURN FROM LEAST SQUARES ROUTINE, DETE
     1RMINANT = 0 ****//)
      GO TO 190
140   DO 160 MODE=1,NMODE
      WRITE (6,150) MODE
150   FORMAT (//,1X,28HWEIGHTING FACTORS ANM, MODE(,I2,22H), 16 REAL PAR
     1TS FIRST//)
      WRITE (6,90) (WFACT(J,MODE),J=1,32)
```

```
      DO 160 JK=1,6
      DO 160 KJ=1,16
      RP=WFACT(KJ,JK)
      KKJ=KJ+16
      XIP=WFACT(KKJ,JK)
160   ANM(KJ,JK)=CMPLX(RP,XIP)
C     TAPE9 IS A COMMUNICATION CHANNEL TO OVERLAY 2,0
      WRITE (9) (ANM(JK,MODE),JK=1,16)
      WRITE (6,170) MODE
170   FORMAT (//,1X,29HLEAST SQUARE RESIDUALS, MODE(,I2,50H), NXY REAL P
     1ARTS FIRST, IN CONTROL POINT SEQUENCE//)
180   WRITE (6,90) (U(J,MODE),J=1,NXY2)
190   RETURN
      END
```

GLSP.- The function of subprogram GLSP is to solve for the weighting factors $a_{nm}^{(j)}$ of equation (32) of reference 1 with the aid of subprogram SIMEQ. A weighted least-square-error solution is made if the number of equations M exceeds the number (N = 16) of unknowns. The matrix of weights is supplied by the user. For WT(1) = -1.0, all weights are made 1.0. After a reduction to a normal form, solution of the simultaneous equations is done by SIMEQ. If the number of equations equals the number of unknowns, SIMEQ is called directly. If the number of equations is less than the number of unknowns, solution is not made. Instead a RETURN is made with IER = 1. The listings for subprograms GLSP and SIMEQ follow.

```
      SUBROUTINE GLSP (A,M,N,B,IP,X,U,SUM,WT,IER,TB,WA,MAXM,MAXN)
C     *** DOCUMENT DATE 08-01-68   SUBROUTINE REVISED  07-09-71 *********
C        THIS ROUTINE IS ADAPTED FOR USE AT LANGLEY FROM SMITHSONIAN
C        ASTROPHYSICAL OBSERVATORY SUBROUTINE GLSP.
C         A(MAXM,MAXN) = GIVEN RECTANGULAR MATRIX
C         M = NO. ROWS OF A
C         N = NO. COLS OF A
C         B(MAXM,IP) = MATRIX  OF RIGHT SIDE
C         IP = NO. COLS OF B
C         X(MAXN,IP) = SOLUTION MATRIX
C         U(MAXM,IP) = RESIDUAL MATRIX
C         SUM(IP) = SUM OF WEIGHTED SQUARES OF RESIDUALS
C         WT(MAXM) = WEIGHTS.  IF WT(1)=-1.0, ALL WEIGHTS ARE SET = 1.0
C         IER = ERROR RETURN.  = 0, NORMAL.  = 1, M LT N.  = 2, DET = 0.
C         TB(MAXN) = TEMP. STORAGE IN SIMEQ
C         WA(MAXN,MAXN) = MATRIX OF NORMAL EQUATIONS
C         MAXM = MAXIMUM ROWS OF A
C         MAXN = MAXIMUM COLS OF A
C         SIMEQ = ROUTINE TO SOLVE NORMAL EQUATIONS
C
      DIMENSION A(MAXM,MAXN), B(MAXM,IP), X(MAXN,IP), U(MAXM,IP),
     1 SUM(IP),WT(MAXM),TB(MAXN    ),WA(MAXN,MAXN)
C
      DET=0.0
      IER=0
C        SET ALL WEIGTS = 1.0 FOR NO WEIGHTING (WT(1) = -1.0)
      IF (WT(1).GE.0.) GO TO 20
      DO 10 I=1,M
10    WT(I)=1.0
20    IF (M-N) 30,40,70
C        IF M LESS THAN N, NO SOLUTION,  IER = 1
30    IER=1
      RETURN
```

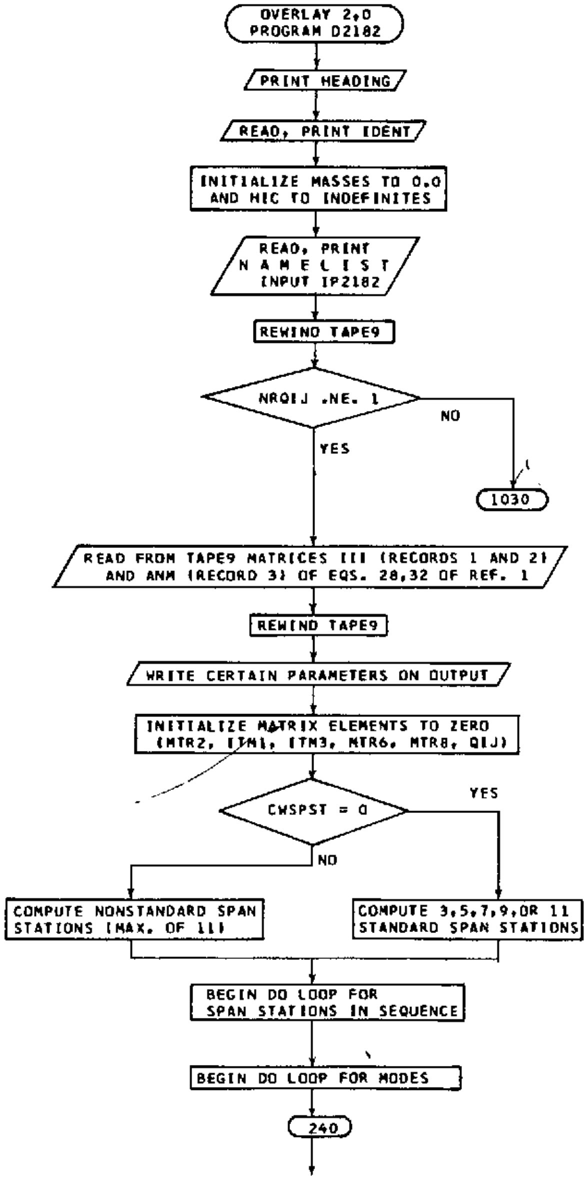
```
C        IF M = N, SOLUTION WITH NORMAL EQUATIONS, AX = B                  J  35     97000000
40    DO 60 I=1,M                                                          J  36     97100000
      DO 50 J=1,N                                                          J  37     97200000
50    WA(I,J)=A(I,J)                                                       J  38     97300000
      DO 60 K=1,IP                                                         J  39     97400000
60    X(I,K)=B(I,K)                                                        J  40     97500000
      GO TO 100                                                            J  41     97600000
C        IF M GT N, WA = A(T) * A * WT,  X = A(T) * B * WT                 J  42     97700000
C                                                                          J  43     97800000
70    DO 80 K=1,N                                                          J  44     97900000
      DO 80 J=1,N                                                          J  45     98000000
      WA(J,K)=0.0                                                          J  46     98100000
      DO 80 I=1,M                                                          J  47     98200000
80    WA(J,K)=WA(J,K)+A(I,K)*A(I,J)*WT(I)                                  J  48     98300000
C        WEIGHT B                                                          J  49     98400000
      DO 90 K=1,IP                                                         J  50     98500000
      DO 90 J=1,N                                                          J  51     98600000
      X(J,K)=0.0                                                           J  52     98700000
      DO 90 I=1,M                                                          J  53     98800000
90    X(J,K)=X(J,K)+A(I,J)*B(I,K)*WT(I)                                    J  54     98900000
C        SOLVE  EQUATIONS BY SIMEQ                                         J  55     99000000
100   CALL SIMEQ (WA,N,X,IP,DET,TB,MAXN,IS)                                J  56     99100000
      IF (DET.NE.0.) GO TO 110                                             J  57     99200000
      IER=2                                                                J  58     99300000
      GO TO 140                                                            J  59     99400000
C        SOLVE RESIDUALS AND SUM OF SQUARES OF RESIDUES                    J  60     99500000
110   DO 130 K=1,IP                                                        J  61     99600000
      SUM(K)=0.0                                                           J  62     99700000
      DO 130 I=1,M                                                         J  63     99800000
      U(I,K)=-B(I,K)                                                       J  64     99900000
      DO 120 J=1,N                                                         J  65    100000000
120   U(I,K)=U(I,K)+(A(I,J)*X(J,K))                                        J  66    100100000
130   SUM(K)=SUM(K)+(U(I,K))**2*WT(I)                                      J  67    100200000
140   RETURN                                                               J  68    100300000
      END                                                                  J  69-   100400000


      SUBROUTINE SIMEQ (A,N,B,M,DETERM,IPIVOT,NMAX,ISCALE)                 K   2    100600000
C     *** DOCUMENT DATE 08-01-68   SUBROUTINE REVISED 08-01-68 *********   K   3    100700000
C     SOLUTION OF SIMULTANEOUS LINEAR EQUATIONS                            K   4    100800000
C                                                                          K   5    100900000
      DIMENSION IPIVOT(N),A(NMAX,N),B(NMAX,M)                              K   6    101000000
      EQUIVALENCE (IROW,JROW),(ICOLUM,JCOLUM),(AMAX,T,SWAP)                K   7    101100000
C                                                                          K   8    101200000
C     INITIALIZATION                                                       K   9    101300000
C                                                                          K  10    101400000
10    ISCALE=0                                                             K  11    101500000
      R1=10.0**100                                                         K  12    101600000
      R2=1.0/R1                                                            K  13    101700000
      DETERM=1.0                                                           K  14    101800000
      DO 20 J=1,N                                                          K  15    101900000
20    IPIVOT(J)=0                                                          K  16    102000000
      DO 380 I=1,N                                                         K  17    102100000
C                                                                          K  18    102200000
C     SEARCH FOR PIVOT ELEMENT                                             K  19    102300000
C                                                                          K  20    102400000
      AMAX=0.0                                                             K  21    102500000
      DO 70 J=1,N                                                          K  22    102600000
      IF (IPIVOT(J)-1) 30,70,30                                            K  23    102700000
30    DO 60 K=1,N                                                          K  24    102800000
      IF (IPIVOT(K)-1) 40,60,390                                           K  25    102900000
40    IF (ABS(AMAX)-ABS(A(J,K))) 50,60,60                                  K  26    103000000
50    IROW=J                                                               K  27    103100000
      ICOLUM=K                                                             K  28    103200000
      AMAX=A(J,K)                                                          K  29    103300000
60    CONTINUE                                                             K  30    103400000
70    CONTINUE                                                             K  31    103500000
      IF (AMAX) 90,80,90                                                   K  32    103600000
80    DETERM=0.0                                                           K  33    103700000
      ISCALE=0                                                             K  34    103800000
      GO TO 390                                                            K  35    103900000
90    IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1                                      K  36    104000000
```

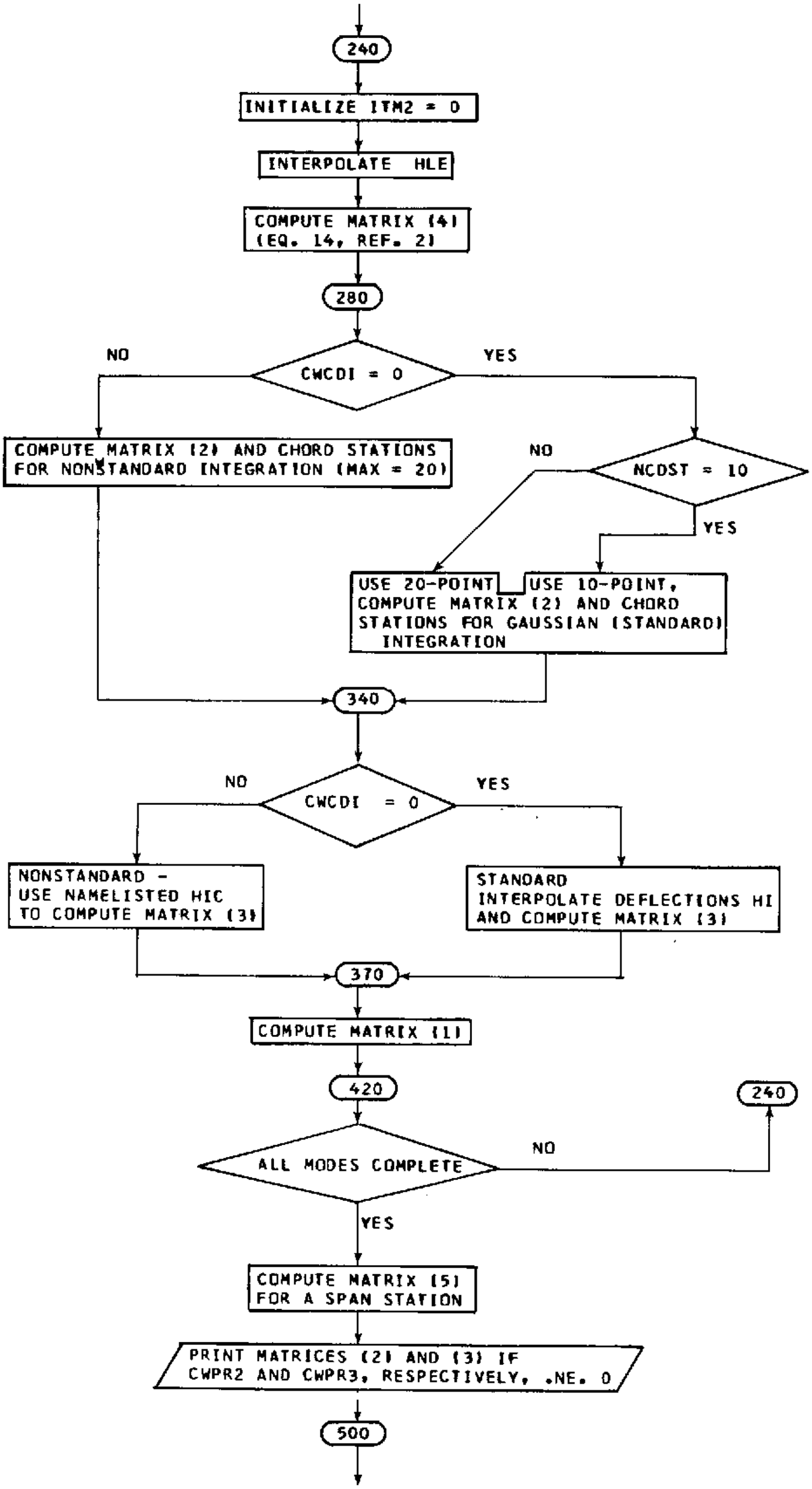
```
C
C     INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL
C
      IF (IROW-ICOLUM) 100,140,100
100   DETERM=-DETERM
      DO 110 L=1,N
      SWAP=A(IROW,L)
      A(IROW,L)=A(ICOLUM,L)
110   A(ICOLUM,L)=SWAP
      IF (M) 140,140,120
120   DO 130 L=1,M
      SWAP=B(IROW,L)
      B(IROW,L)=B(ICOLUM,L)
130   B(ICOLUM,L)=SWAP
140   PIVOT=A(ICOLUM,ICOLUM)
      IF (PIVOT) 150,80,150
C
C     SCALE THE DETERMINANT
C
150   PIVOTI=PIVOT
      IF (ABS(DETERM)-R1) 180,160,160
160   DETERM=DETERM/R1
      ISCALE=ISCALE+1
      IF (ABS(DETERM)-R1) 210,170,170
170   DETERM=DETERM/R1
      ISCALE=ISCALE+1
      GO TO 210
180   IF (ABS(DETERM)-R2) 190,190,210
190   DETERM=DETERM*R1
      ISCALE=ISCALE-1
      IF (ABS(DETERM)-R2) 200,200,210
200   DETERM=DETERM*R1
      ISCALE=ISCALE-1
210   IF (ABS(PIVOTI)-R1) 240,220,220
220   PIVOTI=PIVOTI/R1
      ISCALE=ISCALE+1
      IF (ABS(PIVOTI)-R1) 270,230,230
230   PIVOTI=PIVOTI/R1
      ISCALE=ISCALE+1
      GO TO 270
240   IF (ABS(PIVOTI)-R2) 250,250,270
250   PIVOTI=PIVOTI*R1
      ISCALE=ISCALE-1
      IF (ABS(PIVOTI)-R2) 260,260,270
260   PIVOTI=PIVOTI*R1
      ISCALE=ISCALE-1
270   DETERM=DETERM*PIVOTI
C
C     DIVIDE PIVOT ROW BY PIVOT ELEMENT
C
      DO 290 L=1,N
      IF (IPIVOT(L)-1) 280,290,390
280   A(ICOLUM,L)=A(ICOLUM,L)/PIVOT
290   CONTINUE
      IF (M) 320,320,300
300   DO 310 L=1,M
310   B(ICOLUM,L)=B(ICOLUM,L)/PIVOT
C
C     REDUCE NON-PIVOT ROWS
C
320   DO 380 L1=1,N
      IF (L1-ICOLUM) 330,380,330
330   T=A(L1,ICOLUM)
      DO 350 L=1,N
      IF (IPIVOT(L)-1) 340,350,390
340   A(L1,L)=A(L1,L)-A(ICOLUM,L)*T
350   CONTINUE
      IF (M) 380,380,360
360   DO 370 L=1,M
370   B(L1,L)=B(L1,L)-B(ICOLUM,L)*T
380   CONTINUE
390   RETURN
      END
```

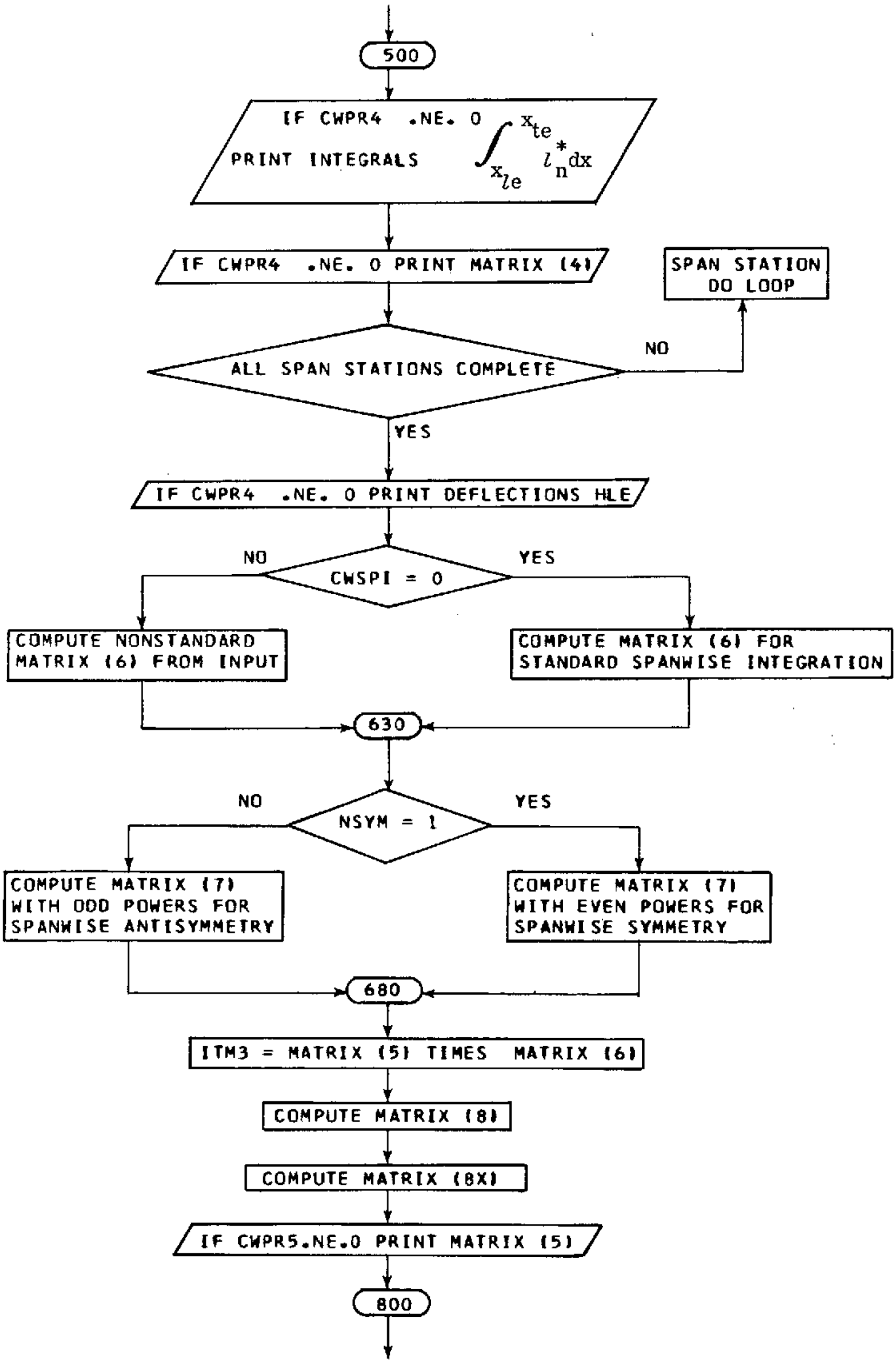
OVERLAY (JMF,2,0); D2182

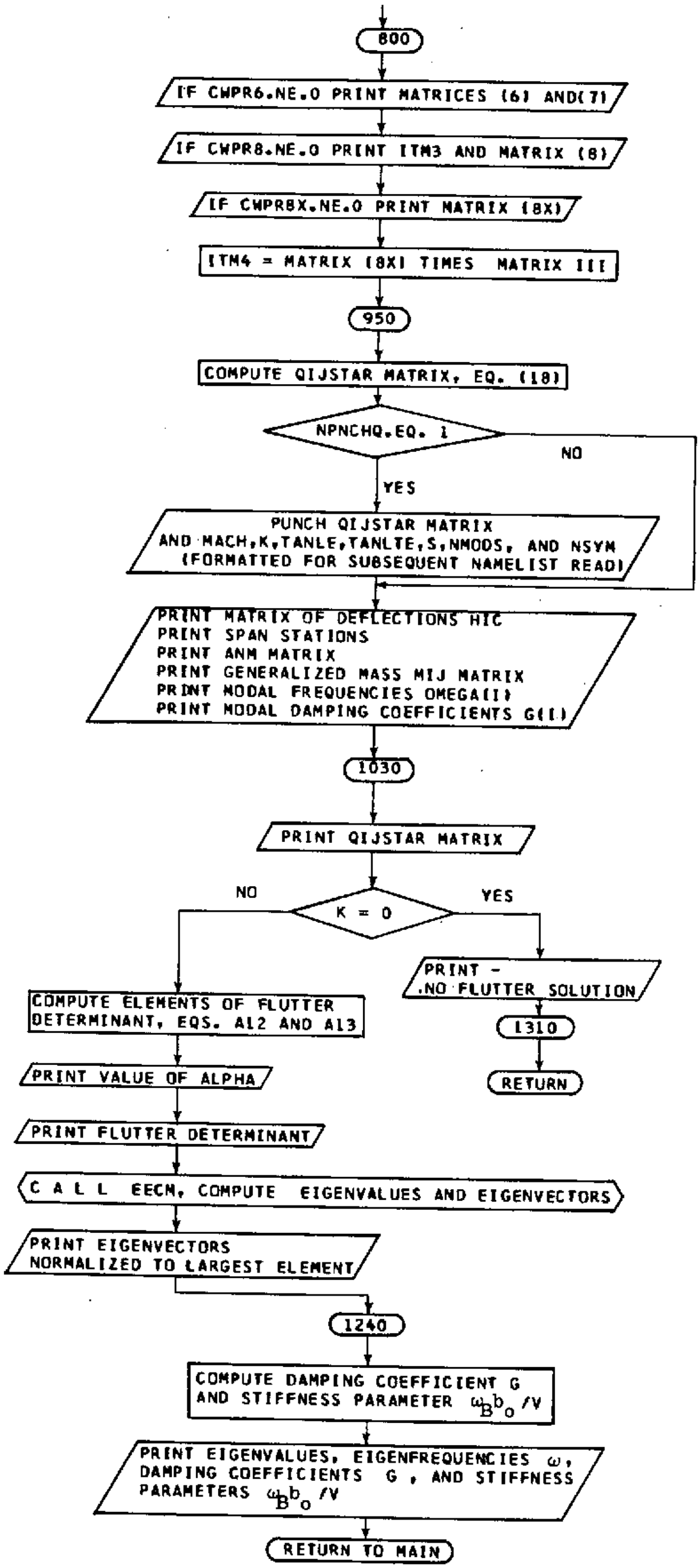
Subprogram D2182 is the controlling subprogram in overlay 2,0. Its function is to direct the calculation of the generalized aerodynamic forces Q^*_{ij} and, optionally, of the flutter solutions as described in reference 2. A NAMELIST input is read. If it includes previously calculated Q^*_{ij} values, their calculation is bypassed. Otherwise, the III and $a^{(j)}_{nm}$ matrices are obtained from TAPE9. During the calculation of Q^*_{ij}, most of the intermediate matrices can be optionally printed. Finally, flutter solutions are made for up to 25 values of air density, and the output is printed. The flow chart for D2182 follows.

FLOW CHART FOR PROGRAM D2182




240
INITIALIZE ITM2 = 0
INTERPOLATE HLE
COMPUTE MATRIX (4) (EQ. 14, REF. 2)
280
CWCDI = 0
NO
YES
COMPUTE MATRIX (2) AND CHORD STATIONS FOR NONSTANDARD INTEGRATION (MAX = 20)
NCDST = 10
NO
YES
USE 20-POINT
USE 10-POINT, COMPUTE MATRIX (2) AND CHORD STATIONS FOR GAUSSIAN (STANDARD) INTEGRATION
340
CWCDI = 0
NO
YES
NONSTANDARD - USE NAMELISTED HIC TO COMPUTE MATRIX (3)
STANDARD INTERPOLATE DEFLECTIONS HI AND COMPUTE MATRIX (3)
370
COMPUTE MATRIX (1)
420
ALL MODES COMPLETE
NO
240
YES
COMPUTE MATRIX (5) FOR A SPAN STATION
PRINT MATRICES (2) AND (3) IF CWPR2 AND CWPR3, RESPECTIVELY, .NE. 0
500


500
IF CWPR4 .NE. 0
PRINT INTEGRALS $\int_{x_{le}}^{x_{te}} l_n^* dx$
IF CWPR4 .NE. 0 PRINT MATRIX (4)
SPAN STATION DO LOOP
NO
ALL SPAN STATIONS COMPLETE
YES
IF CWPR4 .NE. 0 PRINT DEFLECTIONS HLE
NO
CWSPI = 0
YES
COMPUTE NONSTANDARD MATRIX (6) FROM INPUT
COMPUTE MATRIX (6) FOR STANDARD SPANWISE INTEGRATION
630
NO
NSYM = 1
YES
COMPUTE MATRIX (7) WITH ODD POWERS FOR SPANWISE ANTISYMMETRY
COMPUTE MATRIX (7) WITH EVEN POWERS FOR SPANWISE SYMMETRY
680
ITM3 = MATRIX (5) TIMES MATRIX (6)
COMPUTE MATRIX (8)
COMPUTE MATRIX (8X)
IF CWPR5.NE.0 PRINT MATRIX (5)
800


800
IF CWPR6.NE.0 PRINT MATRICES (6) AND(7)
IF CWPR8.NE.0 PRINT ITM3 AND MATRIX (8)
IF CWPR8X.NE.0 PRINT MATRIX (8X)
ITM4 = MATRIX (8X) TIMES MATRIX (1)
950
COMPUTE QIJSTAR MATRIX, EQ. (18)
NPNCHQ.EQ. 1
NO
YES
PUNCH QIJSTAR MATRIX
AND MACH,K,TANLE,TANLTE,S,NMODS, AND NSYM
(FORMATTED FOR SUBSEQUENT NAMELIST READ)
PRINT MATRIX OF DEFLECTIONS HIC
PRINT SPAN STATIONS
PRINT ANM MATRIX
PRINT GENERALIZED MASS MIJ MATRIX
PRINT MODAL FREQUENCIES OMEGA(I)
PRINT MODAL DAMPING COEFFICIENTS G(I)
1030
PRINT QIJSTAR MATRIX
NO
K = 0
YES
PRINT -
NO FLUTTER SOLUTION
1310
RETURN
COMPUTE ELEMENTS OF FLUTTER
DETERMINANT, EQS. A12 AND A13
PRINT VALUE OF ALPHA
PRINT FLUTTER DETERMINANT
C A L L EECN, COMPUTE EIGENVALUES AND EIGENVECTORS
PRINT EIGENVECTORS
NORMALIZED TO LARGEST ELEMENT
1240
COMPUTE DAMPING COEFFICIENT G
AND STIFFNESS PARAMETER $\omega_B b_o/V$
PRINT EIGENVALUES, EIGENFREQUENCIES ω,
DAMPING COEFFICIENTS G, AND STIFFNESS
PARAMETERS $\omega_B b_o/V$
RETURN TO MAIN

```
      OVERLAY(JMF,2,0)
      PROGRAM D2182
      DIMENSION XBTB10(10),XBTB20(20),TBIC10(10),TBIC20(20),YSP(11),
     1XC(20),XLE(11),XTE(11),MTR1(6,20),MTR2(20),MTR3(20,6),MTR4(6,6),
     2MTR4A(6,11),HLE(11,6),MTR5(6,6,11),MTR6(11),MTR7(11,10),     MTR8(6
     3,6,10),MTR8X(6,40),ITM1(6,20),ITM2(6,6),ITM3(6,6,11),IDENT(8),
     4XCBAR(20),KB(6),YSPS(11),HIC(11,20,6),HTAB(20),NDX(6),CFE(6),
     5TEMP(6),G(6),OMEGA(6),MASS(6,6),ALPHA(25),XIC(20),XTAB(20),SMG(6)
     6,EOMEG(6),ISPS(11),XTB(20),INTHACT(6,2)
      COMPLEX TKEY(46,16),ANM(16,6),GAMMA(6),CAPG(6,6),QIJ(6,6),EIG(6)
     1,ITM4(6,46),CFE,AD(6),STOR(6,6),HACT(6,6),VEC(6,6),VECSAV
      REAL MTR1,MTR2,MTR3,MTR4,MTR4A,MTR5,MTR6,MTR7,MTR8,MTR8X,ITM1,ITM2
     1,ITM3,K,MASS,KB,MACH,ISPS
      COMMON/TRA2182/XM,XK,TANLE,TANLTE,S,NMODE,NSYM
      INTEGER CWSPST,CWSPI,CWCDI,CWPR2,CWPR3,CWPR4,CWPR5,
     1CWPR6,CWPR8,CWPR8X
C         DATA TABLES FOR INTEGRATION STATIONS AND FACTORS FOR 10 AND
C           20-POINT GAUSSIAN QUADRATURE  FOR QIJ,S  ***
      DATA (XBTB10(J),J=1,10)/.13046736E-1,.67468317E-1,.16029522,
     1.28330230,.42556283,.57443717,.71669770,.83970478,.93253168,
     2.98695326/
      DATA (XBTB20(J),J=1,20)/.34357004E-2,.18014036E-1,.43882786E-1,
     1.80441514E-1,.12683405,.18197316,.24456650,.31314696,.38610707,
     2.46173674,.53826326,.61389293,.68685304,.75543350,.81802684,
     3.87316595,.91955849,.95611721,.98198596,.99656430/
      DATA (TBIC10(J),J=1,10)/.33335672E-1,.74725675E-1,.10954318,
     1.13463336,.14776211,.14776211,.13463336,.10954318,.74725675E-1,
     2.33335672E-1/
      DATA (TBIC20(J),J=1,20)/.88070035E-2,.20300715E-1,.31336024E-1,
     1.41638371E-1,.50965060E-1,.59097266E-1,.65844319E-1,.71048055E-1,
     2.74586493E-1,.76376694E-1,.76376694E-1,.74586493E-1,.71048055E-1,
     3.65844319E-1,.59097266E-1,.50965060E-1,.41638371E-1,.31336024E-1,
     4.20300715E-1,.88070035E-2/
      NAMELIST /IP2182/NSPST,NCOST,MASS,OMEGA,OMEGB,G,NALPHA,ALPHA,YSPS,
     1 ISPS,HIC,CWSPST,CWSPI,CWCDI,XTAB,CWPR2,CWPR3,
     2 CWPR4,CWPR5,CWPR6,CWPR8,CWPR8X,XCBAR,XIC,NHP
     3,INTOR,NPNCHG,NRQIJ,QIJ,MACH,K,TANLE,TANLTE,S,NMODS,NSYM
      NMODS=NMODE$MACH=XM$K=XK
      PRINT 10
10    FORMAT (1H1,///22X,87HPROGRAM D2182 ... COMPUTE GENERALIZED AERODY
     1NAMIC FORCES AND SOLVE FLUTTER DETERMINANTS)
C     DEFAULT VALUES OF CONTROL PARAMETERS FOR QUADRATURE, PRINTING,
C      PUNCHING, AND INPUT  ***
      CWCDI=CWPR2=CWPR3=CWPR4=CWPR5=CWPR6=CWPR8=CWPR8X =0
      CWSPI=CWSPST=NPNCHG=NRQIJ=0
C         BEGIN INPUT
      READ (5,20) IDENT
20    FORMAT (8A10)
      PRINT 30, IDENT
30    FORMAT (///10X,8A10//)
      DO 40 II=1,36
      QIJ(II)=0.0
40    MASS(II)=0.0
      DO 50 II=1,1320
50    HIC(II)=17770000000000777777B
      READ (5,IP2182)
   60 FORMAT (//1X,32H ECHO OF NAMELIST IP2182 FOLLOWS )
      WRITE (6,60)
      WRITE (6,IP2182)
      IF (NRQIJ.EQ.1) GO TO 1030
      REWIND 9
      READ (9) ((TKEY(JK,KJ),JK=1,46),KJ=1,4)
      READ (9) ((TKEY(JK,KJ),JK=1,46),KJ=5,16)
      DO 70 KJ=1,NMODS
70    READ (9) (ANM(JK,KJ),JK=1,16)
      REWIND 9
      WRITE (6,80) MACH,K,TANLE,TANLTE,S,NMODS,NSYM
80    FORMAT (//6H MACH=E15.8,5X,2HK=E15.8,5X,6HTANLE=E15.8,5X,7HTANLTE=
     1E15.8,5X,2HS=E15.8//7H NMODS=I2,5X,5HNSYM=I2///)
C
C***   INITIALIZE MATRIX ELEMENTS TO ZERO
      DO 90 J=1,NCOST
```

```
   90 MTR2(J)=0.
      DO 110 NSJ=1,6
      DO 110 NJJ=1,NMODS
      DO 100 JS=1,NSPST
100   ITM3(NSJ,NJJ,JS)=0.0
110   CONTINUE
      DO 120 J=1,6
      DO 120 KJ=1,NCOST
120   ITM1(J,KJ)=0.0
C
      DO 130 JP=1,NSPST
  130 MTR6(JP)=0.0
140   CONTINUE
      DO 170 MJ=1,6
      DO 160 MK=1,6
      DO 150 ML=1,10
150   MTR8(MJ,MK,ML)=0.0
160   CONTINUE
170   CONTINUE
      DO 190 MJ=1,NMODS
      DO 180 ML=1,NMODS
180   QIJ(MJ,ML)=0.0
190   CONTINUE
C     SPANWISE STATIONS ESTABLISHED
C
      IF (CWSPST.EQ.0) GO TO 210
C
C     NON-STANDARD SPAN STATIONS
C
      DO 200 J=1,NSPST
200   YSP(J)=YSPS(J)*S
      GO TO 230
C
C     STANDARD SPAN STATIONS (3,5,7,9,OR11 EQUALLY SPACED)
C
210   DY=S/FLOAT(NSPST-1)
      YSP(1)=0
      DO 220 J=2,NSPST
220   YSP(J)=YSP(J-1)+DY
230   CONTINUE
      DO 570 ISPAN=1,NSPST
      DO 420 MODE=1,NMODS
      DO 240 J=1,6
      DO 240 KJ=1,NMODS
240   ITM2(J,KJ)=0.0
C
C     COMPUTE AND STORE CHORD INTEGRALS OF LNSTAR - MTR4A
C
      XLE(ISPAN)=ABS(YSP(ISPAN))*TANLE
      XTE(ISPAN)=1.+ABS(YSP(ISPAN))*TANLTE
      XVAL=XLE(ISPAN)
      DO 250 J=1,NHP
      XTB(J)=XLE(ISPAN)+XTAB(J)*(XTE(ISPAN)-XLE(ISPAN))
250   HTAB(J)=HIC(ISPAN,J,MODE)
C     INTERPOLATE LE DEFLECTION
      CALL FTLUP (XVAL,HVAL,INTOR,NHP,XTB(1),HTAB(1))
      HLE(ISPAN,MODE)=HVAL
      XLTSQ=XTE(ISPAN)**2-XLE(ISPAN)**2
      XLTSR=SQRT(XLTSQ)
      XLTS3=XLTSR**3
      XLTS5=XLTSR**5
      XLE2=XLE(ISPAN)**2
      IF (XLE(ISPAN).EQ.0.) GO TO 260
      XTOXL=ABS(XTE(ISPAN)/XLE(ISPAN))
      XCOSH=ALOG(XTOXL+SQRT(XTOXL**2-1.))
      GO TO 270
260   XCOSH=0.0
270   CONTINUE
      MTR4A(1,ISPAN)=XLTSR
      MTR4A(2,ISPAN)=.5*XTE(ISPAN)*XLTSR+.5*XLE2*XCOSH
      MTR4A(3,ISPAN)=.5*XTE(ISPAN)*XLTSR-.5*XLE2*XCOSH
      MTR4A(4,ISPAN)=.33333333*XLTS3
```

```
      MTR4A(5,ISPAN)=.25*XTE(ISPAN)*XLTS3+.125*XLE2*XTE(ISPAN)*XLTSR-.12
     15*XLE2**2*XCUSH
      MTR4A(6,ISPAN)=.2*XLTS5+.33333333*XLE2*XLTS3
      DO 280 NN=1,6
280   MTR4(NN,MODE)=MTR4A(NN,ISPAN)*HLE(ISPAN,MODE)
C
C     COMPUTE AND STORE CHORD INTEGRATING MATRIX - MTR2
C
      IF (CWCOI.EQ.0) GO TO 300
C
C     NON-STANDARD CHORDWISE INTEGRATION
C
      DO 290 J=1,NCDST
      MTR2(J)=XIC(J)*(XTE(ISPAN)-XLE(ISPAN))
290   XC(J)=XLE(ISPAN)+XCBAR(J)*(XTE(ISPAN)-XLE(ISPAN))
      GO TO 340
C
C     STANDARD CHORDWISE INTEGRATION (10 OR 20-POINT GAUSS)
C
300   IF (NCDST.EQ.10) GO TO 320
      DO 310 J=1,NCDST
      MTR2(J)=TBIC20(J)*(XTE(ISPAN)-XLE(ISPAN))
310   XC(J)=XLE(ISPAN)+XBTB20(J)*(XTE(ISPAN)-XLE(ISPAN))
      GO TO 340
320   DO 330 J=1,NCDST
      MTR2(J)=TBIC10(J)*(XTE(ISPAN)-XLE(ISPAN))
330   XC(J)=XLE(ISPAN)+XBTB10(J)*(XTE(ISPAN)-XLE(ISPAN))
340   CONTINUE
      DO 410 KCHD=1,NCDST
C
C     COMPUTE AND STORE DEFLECTIONS H-HLE(MODE I) - MTR3
C
      IF (CWCDI.NE.0) GO TO 360
C
C     STANDARD INTERPOLATION OF DEFLECTIONS
C
      DO 350 J=1,NHP
      XTB(J)=XLE(ISPAN)+XTAB(J)*(XTE(ISPAN)-XLE(ISPAN))
350   HTAB(J)=HIC(ISPAN,J,MODE)
      XVAL=XC(KCHD)
C     INTERPOLATE DEFLECTIONS
      CALL FTLUP (XVAL,HVAL,INTCR,NHP,XTB(1),HTAB(1))
      MTR3(KCHD,MODE)=HVAL-HLE(ISPAN,MODE)
      GO TO 370
C
C     NON-STANDARD, USE DEFLECTIONS HIC W/O INTERPOLATION
C
  360 MTR3(KCHD,MODE)=HIC(ISPAN,KCHD,MODE)-HLE(ISPAN,MODE)
370   CONTINUE
      SQRTRAD=SQRT(XC(KCHD)**2-XLE(ISPAN)**2)
      XCN=1.0
      DO 400 N=1,6
      IF (N.EQ.1) GO TO 380
      IF (N.EQ.2) GO TO 390
      MTR1(N,KCHD)=XCN*SQRTRAD
      XCN=XCN*XC(KCHD)
      GO TO 400
  380 IF(SQRTRAD.NE. 0.0) GO TO 382
      MTR1(N,KCHD)=0.0
      GO TO 400
  382 MTR1(N,KCHD)=XC(KCHD)/SQRTRAD
      GO TO 400
390   MTR1(N,KCHD)=XC(KCHD)*MTR1(N-1,KCHD)
400   CONTINUE
410   CONTINUE
420   CONTINUE
C          END OF MODAL DO LOOP ***
C
C     MATRIX OPERATION FOR CHORDWISE INTEGRATION
C
C
```

```
C     COMPUTE AND STORE CHORD INTEGRALS - MTR5
C
      DO 430 J=1,6
      DO 430 JJ=1,NCDST
  430 ITM1(J,JJ)=MTR1(J,JJ)*MTR2(JJ)
      DO 440 J=1,6
      DO 440 JJ=1,NMODS
      DO 440 JJJ=1,NCDST
440   ITM2(J,JJ)=ITM1(J,JJJ)*MTR3(JJJ,JJ)+ITM2(J,JJ)
      DO 450 J=1,6
      DO 450 JJ=1,NMODS
450   MTR5(JJ,J,ISPAN)=ITM2(J,JJ)+MTR4(J,JJ)
      IF (CWPR2.EQ.0) GO TO 480
      WRITE (6,460)
  460 FORMAT (//10X,21HMATRIX MTR1 - BY ROWS )
      WRITE (6,1000) ((MTR1(JK,KJ),KJ=1,NCDST),JK=1,6)
      WRITE (6,470)
  470 FORMAT (//10X, 30HMATRIX MTR2, DIAGONAL ELEMENTS )
      WRITE (6,1000) (MTR2(JK), JK=1,NCDST)
480   IF (CWPR3.EQ.0) GO TO 500
      WRITE (6,490)
  490 FORMAT (//10X,34HDEFLECTIONS H-HLE , MTR3 , BY ROWS )
      WRITE (6,1000) ((MTR3(JK,KJ),JK=1,NCDST),KJ=1,NMODS)
  500 IF (CWPR4.EQ.0) GO TO 520
      WRITE (6,510)
510   FORMAT (//10X,33HCHORD INTEGRALS OF LNSTAR - MTR4A/)
      WRITE (6,1000) (MTR4A(JK,ISPAN),JK=1,6)
  520 IF (CWPR4.EQ.0) GO TO 560
      WRITE (6,530)
  530 FORMAT (//10X,21HMATRIX MTR4 - BY ROWS )
      WRITE (6,1000) ((MTR4(JK,KJ),KJ=1,NMODS),JK=1,6)
      WRITE (6,540)
  540 FORMAT (//10X,33HMATRIX ITM1 = MTR1*MTR2 , BY ROWS )
      WRITE (6,1000) ((ITM1(JK,KJ),KJ=1,NCDST),JK=1,6)
      WRITE (6,550)
  550 FORMAT (//10X,33HMATRIX ITM2 = ITM1*MTR3 , BY ROWS )
      WRITE (6,1000) ((ITM2(JK,KJ),KJ=1,NMODS),JK=1,6)
560   CONTINUE
570   CONTINUE
C          END OF SPAN-STATION DO LOOP ***
      IF (CWPR4.EQ.0) GO TO 590
      WRITE (6,580)
  580 FORMAT(//10X,51HDEFLECTIONS HLE OF MTR4, BY SPAN STA., MODE 1 FIRS
     1T )
      WRITE (6,1000) ((HLE(JK,KJ),JK=1,NSPST),KJ=1,NMODS)
590   CONTINUE
C
C     COMPUTE AND STORE SPAN INTEGRATING MATRIX - MTR6
C
      IF (CWSPI.EQ.0) GO TO 610
C
C     NON-STANDARD SPANWISE INTEGRATION
C
      DO 600 JS=1,NSPST
  600 MTR6(JS)=ISPS(JS)*S
      GO TO 630
C
C     STANDARD SPANWISE INTEGRATION
C
610   QUANT=S/FLOAT(3*(NSPST-1))
      MTR6(1)=QUANT
      NSM=NSPST-1
      DO 620 JS=2,NSM,2
      MTR6(JS)=4.0*QUANT
  620 MTR6(JS+1)=2.0*QUANT
      MTR6(NSPST)=QUANT
C
630   CONTINUE
      DO 640 JS=1,NSPST
640   MTR7(JS,1)=1.0
      DO 680 JS=1,NSPST
      IF (NSYM.EQ.1) GO TO 660
```

```
      DO 650 JP=1,10
      JSP=2*(JP-1)+1
650   MTR7(JS,JP)=YSP(JS)**JSP
      GO TO 680
660   DO 670 JP=2,10
      JSP=2*(JP-2)+2
670   MTR7(JS,JP)=YSP(JS)**JSP
680   CONTINUE
      DO 700 NSJ=1,6
      DO 700 NJJ=1,NMODS
      DO 690 JS=1,NSPST
  690 ITM3(NSJ,NJJ,JS)=MTR5(NJJ,NSJ,JS)*MTR6(JS)
700   CONTINUE
C
C     MATRIX OPERATION FOR SPANWISE INTEGRATION
C
C
C     COMPUTE AND STORE QIJ-STAR MATRIX - MTR8
C
      DO 740 MJ=1,6
      DO 730 MK=1,NMODS
      DO 720 ML=1,10
      DO 710 MM=1,NSPST
710   MTR8(MJ,MK,ML)=ITM3(MJ,MK,MM)*MTR7(MM,ML)+MTR8(MJ,MK,ML)
720   CONTINUE
730   CONTINUE
740   CONTINUE
C
C     COMPUTE AND STORE QIJ-STAR REARRANGED - MTR8X
C
      DO 770 MJ=1,NMODS
      NN=0
      DO 760 MK=1,6
      IF (MK.EQ.1.OR.MK.EQ.3) NTM=10
      IF (MK.EQ.2.OR.MK.EQ.4) NTM=8
      IF (MK.EQ.5) NTM=6
      IF (MK.EQ.6) NTM=4
      DO 750 ML=1,NTM
      NN=NN+1
750   MTR8X(MJ,NN)=MTR8(MK,MJ,ML)
760   CONTINUE
770   CONTINUE
      IF (CWPR5.EQ.0) GO TO 800
      WRITE (6,780)
  780 FORMAT (//10X,22HCHORD INTEGRALS - MTR5/ 5X, 15HN=0 (IN LNSTAR),
     111X, 3HN=1, 17X,3HN=2, 17X,3HN=3, 17X,3HN=4, 17X,3HN=5 )
      DO 790 MJ=1,NMODS
      PRINT 782, MJ
  782 FORMAT (5X, 5HMODE ,I2)
790   WRITE (6,1000) ((MTR5(MJ,JK,KJ),JK=1,6),KJ=1,NSPST)
800   IF (CWPR6.EQ.0) GO TO 820
      WRITE (6,810)
810   FORMAT (//10X,47HSPAN INTEGRATING MATRIX - MTR6 (DIAGONAL TERMS)/)
      WRITE (6,1000) (MTR6(JK), JK=1,NSPST)
      WRITE (6,820)
  820 FORMAT(//10X, 23HMATRIX - MTR7 - BY ROWS /)
      WRITE (6,1000) ((MTR7(JK,KJ),KJ=1,10),JK=1,NSPST)
830   IF (CWPR8.EQ.0) GO TO 880
      WRITE (6,840)
  840 FORMAT (//10X,36HMATRIX ITM3 = MTR5*MTR6 , BY COLUMNS )
      DO 850 JK=1,NMODS
850   WRITE (6,1000) ((ITM3(JJ,JK,JM),JJ=1,6),JM=1,NSPST)
      WRITE (6,860)
860   FORMAT (//10X,50HSPAN INTEGRALS,MTR8,BY ROWS,60 ELEMENTS EACH MODE
     1 /)
      DO 870 MJ=1,NMODS
870   WRITE (6,1000) ((MTR8(JK,MJ,KJ),KJ=1,10),JK=1,6)
880   IF (CWPR8X.EQ.0) GO TO 900
      WRITE (6,890)
890   FORMAT (//10X,62HSPAN INTEGRALS REARRANGED,MTR8X, ROWS OF 46 ELEME
     1NTS EACH MODE/)
      WRITE (6,1000) ((MTR8X(JK,KJ),KJ=1,46),JK=1,NMODS)
```

```
900   CONTINUE
      DO 920 MJ=1,NMODS
      DO 910 ML=1,46
910   ITM4(MJ,ML)=0.0
920   CONTINUE
      DO 940 MJ=1,NMODS
      DO 940 ML=1,16
      DO 930 MM=1,46
930   ITM4(MJ,ML)=MTRBX(MJ,MM)*TREY(MM,ML)+ITM4(MJ,ML)
940   CONTINUE
C
C     MATRIX OPERATION FOR GENERALIZED AERODYNAMIC FORCES
      DO 960 MJ=1,NMODS
      DO 960 MI=1,NMODS
      DO 950 ML=1,16
950   QIJ(MJ,MI)=ITM4(MJ,ML)*ANM(ML,MI)+QIJ(MJ,MI)
960   CONTINUE
C        QIJ-STAR MATRIX IS COMPLETE
      WRITE (6,970)
970   FORMAT (//10X,77HINPUT DEFLECTIONS H SUB I, IN ORDER L.E. TO T.E.,
     1 INBOARD FIRST, MODE 1 FIRST/)
      DO 990 MJ=1,NMODS
      DO 980 MI=1,NSPST
980   WRITE (6,1000) (HIC(MI,J,MJ),J=1,NHP)
990   CONTINUE
1000  FORMAT (6(5X,E15.8))
      WRITE (6,1010)
1010  FORMAT (//10X,13HSPAN STATIONS/)
      WRITE (6,1000) (YSP(J),J=1,NSPST)
      WRITE (6,1020)
1020  FORMAT (//10X,46HANM MATRIX, BY MODAL COLUMNS OF COMPLEX ELEMENTS/
     15X,4HREAL,16X,4HIMAG,16X,4HREAL,16X,4HIMAG,16X,4HREAL,16X,4HIMAG)
      WRITE (6,1000) ((ANM(JK,KJ),JK=1,16),KJ=1,NMODS)
1030  CONTINUE
      WRITE (6,1040)
1040  FORMAT (//10X,10HMIJ MATRIX/)
      WRITE (6,1000) ((MASS(JK,KJ),KJ=1,NMODS),JK=1,NMODS)
      WRITE (6,1050)
1050  FORMAT (//10X,17HMODAL FREQUENCIES/)
      WRITE (6,1000) (OMEGA(J),J=1,NMODS)
      WRITE (6,1060)
1060  FORMAT (//10X,31HMODAL DAMPING COEFFICIENTS G(I)/)
      WRITE (6,1000) (G(J),J=1,NMODS)
      WRITE (6,1070)
1070  FORMAT (//10X,26H QIJ MATRIX, BY COMPLEX ROWS/)
      WRITE (6,1000) ((QIJ(JK,KJ),KJ=1,NMODS),JK=1,NMODS)
      IF (NPNCHQ.NE.1) GO TO 1102
      PUNCH 1080, MACH,K,TANLE,TANLTE,S,NMODS,NSYM
1080  FORMAT (1X,5HMACH=E15.8,3H,K=E15.8,7H,TANLE=E15.8,1H,/1X,7HTANLTE=
     1E15.8,3H,S=E15.8,7H,NMODS=I2,6H,NSYM=I2,1H,)
      DO 1100 MI=1,NMODS
      DO 1100 MJ=1,NMODS,2
      RQIJ=REAL(QIJ(MJ,MI))  $  RQIJP1=REAL(QIJ(MJ+1,MI))
      AIQIJ=AIMAG(QIJ(MJ,MI))  $  AIQIJP1=AIMAG(QIJ(MJ+1,MI))
      PUNCH 1090, MJ,MI,RQIJ,AIQIJ,RQIJP1,AIQIJP1
 1090 FORMAT (5H QIJ(,I1,1H,I1,2H)=,2(1H(,E15.8,1H,,E15.8,2H),))
1100  CONTINUE
C
C         BEGIN FLUTTER SOLUTION  ***
C
 1102 IF (K.NE.0.) GO TO 1120
      PRINT 1110
1110  FORMAT (//33H *** K=^, NO FLUTTER SOLUTION ***//)
      GO TO 1310
1120  DO 1300 NAF=1,NALPHA
      DO 1140 MI=1,NMODS
```

```
      KPT=((OMEGB/OMEGA(MI))**2/(MASS(MI,MI)*(1.+G(MI)**2)))
      XIPT=-G(MI)*KPT
      GAMMA(MI)=CMPLX(KPT,XIPT)
      DO 1130 MJ=1,NMODS
 1130 CAPG(MI,MJ)=GAMMA(MI)*(MASS(MI,MJ)+ALPHA(NAF)*(Q1J(MI,MJ)/K**2))
 1140 CONTINUE
      WRITE (6,1150) ALPHA(NAF)
 1150 FORMAT (////10X,6HALPHA=E15.8)
      WRITE (6,1160)
 1160 FORMAT (//10X,36HFLUTTER DETERMINANT, BY COMPLEX ROWS/)
      WRITE (6,1000) ((CAPG(JK,KJ),KJ=1,NMODS),JK=1,NMODS)
      NMAX=6
      INTHACT(1,1)=INTHACT(2,1)=NMODS
      CALL EECM (CAPG,EIG,VEC,STOR,HACT,CFE,AD,INTHACT,NMAX)
      IF (INTHACT(1,1).GE.NMODS) GO TO 1180
      WRITE (6,1170)
 1170 FORMAT (/60H ********* NON-CONVERGENCE RETURN FROM SUBROUTINE ****
     1******/)
      GO TO 1300
 1180 PRINT 1190
 1190 FORMAT (/6X,98HEIGENVECTORS IN SAME ORDER AS THE EIGENVALUES...EAC
     1H EIGENVECTOR NORMALIZED TO ITS LARGEST ELEMENT)
      DO 1230 LVEC=1,NMODS
C     RENORMALIZE EIGENVECTOR, MAKE LARGEST ELEMENT 1.0 + 0.0I
      JSAVE=1
      DO 1200 J=2,NMODS
      IF ((VEC(JSAVE,LVEC)*CONJG(VEC(JSAVE,LVEC))).LT.(VEC(J,LVEC)*CONJG
     1(VEC(J,LVEC)))) JSAVE=J
 1200 CONTINUE
      VECSAV=VEC(JSAVE,LVEC)
      DO 1210 J=1,NMODS
      VEC(J,LVEC)=VEC(J,LVEC)/VECSAV
 1210 CONTINUE
      PRINT 1220, LVEC,(VEC(I,LVEC),I=1,NMODS)
 1220 FORMAT (14H  EIGENVECTOR(,I2,1H)/(6E21.8)/)
 1230 CONTINUE
      DO 1250 JITE=1,NMODS
      EIGENR=REAL(EIG(JITE))
      EIGENI=AIMAG(EIG(JITE))
      IF (EIGENR.GT.0) GO TO 1240
      SMG(JITE)=EOMEG(JITE)=KB(JITE)=17770000000000777777B
      GO TO 1250
 1240 SMG(JITE)=EIGENI/EIGENR
      EOMEG(JITE)=OMEGB/SQRT(EIGENR)
      KB(JITE)=K*OMEGB/EOMEG(JITE)
 1250 CONTINUE
      WRITE (6,1260)
 1260 FORMAT (//10X,41HEIGENVALUES. RE(1),IM(1),RE(2),IM(2),ETC./)
      WRITE (6,1000) (EIG(J),J=1,NMODS)
      WRITE (6,1270)
 1270 FORMAT (//10X,16HEIGENFREQUENCIES/)
      WRITE (6,1000) (EOMEG(J),J=1,NMODS)
      WRITE (6,1280)
 1280 FORMAT (//10X,22HDAMPING COEFFICIENTS G/)
      WRITE (6,1000) (SMG(J),J=1,NMODS)
      WRITE (6,1290)
 1290 FORMAT (//10X,20HSTIFFNESS PARAMETERS/)
      WRITE (6,1000) (KB(J),J=1,NMODS)
 1300 CONTINUE
 1310 RETURN
      END
```

Description of External File TAPE9

The file TAPE9 is written in Overlay 1,0 and read in Overlay 2,0. As shown on the flow charts, it is handled as follows, for KPROG = 0:

(1) Rewound in subprogram D2181.

(2) In subprogram MTXMPY the first four columns of the matrix III are written as record no. 1, and columns 5 to 16 are written as record no. 2.

(3) In subprogram PRT2 the matrix of $a_{nm}^{(j)}$ is written as record no. 3.

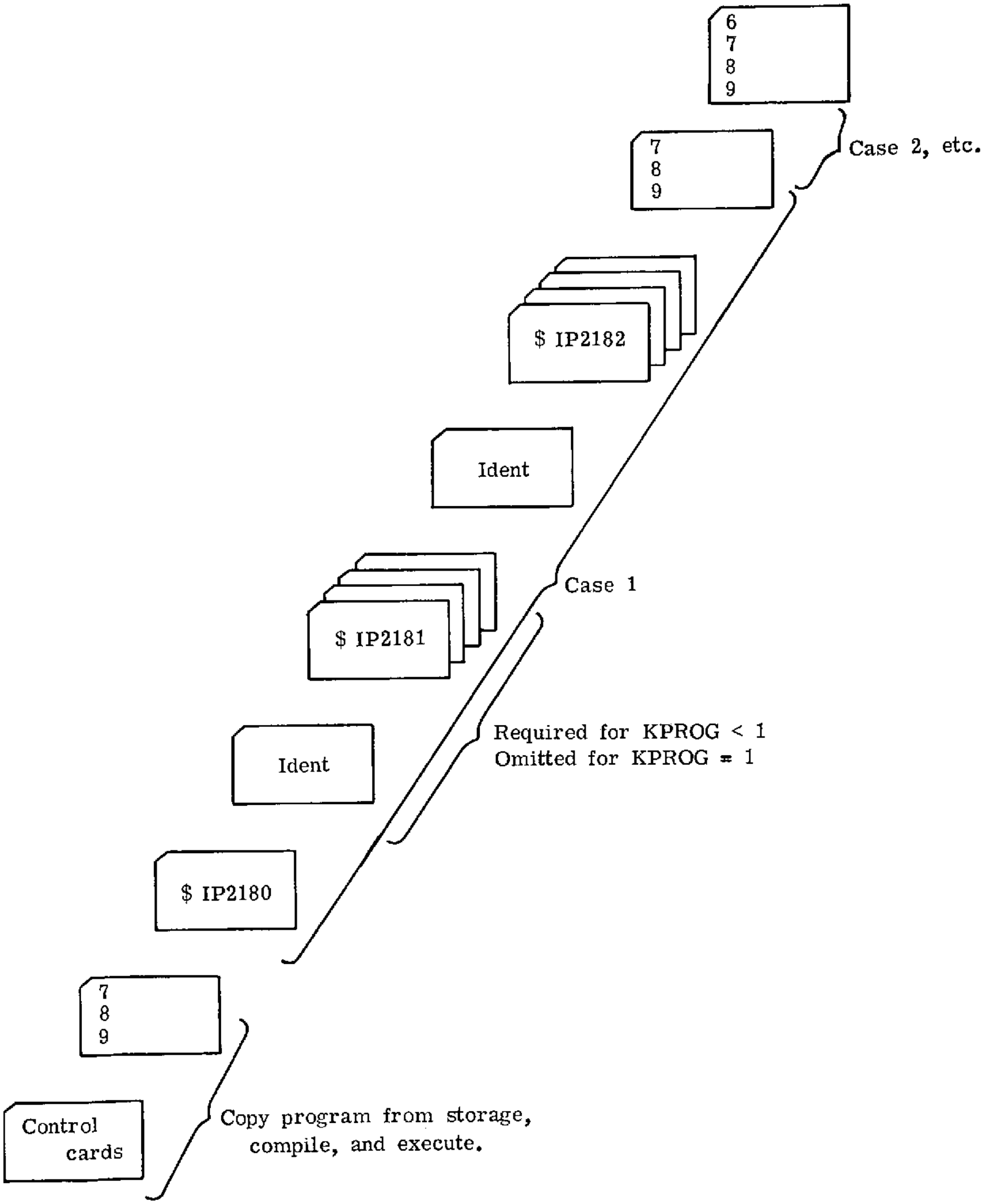
(4) In subprogram D2182, TAPE9 is rewound, the matrix III is read as the first two records, and the matrix $a_{nm}^{(j)}$ is read as record no. 3. TAPE9 is then rewound.

For KPROG ≠ 0 TAPE9 is not used.

USAGE

Program Information and Input Deck Arrangement

The program D2180 is written in the FORTRAN IV language for the Control Data 6000 series digital computer under the SCOPE 3.0 operating system. To compile, load, and run requires a field length of 63000 (octal) locations with variables dimensioned as follows: 48 control points, 6 vibration modes, 25 air density values, and 220 points on the half-span wing for modal deflections. An example of a central processor time to compile and run is 166 seconds on a 6400 for a case with 48 control points, 3 modes, 21 density values, and 110 deflection points. The following sketch shows the input deck arrangement


6
7
8
9
7
8
9
Case 2, etc.
$ IP2182
Ident
Case 1
$ IP2181
Required for KPROG < 1
Omitted for KPROG = 1
Ident
$ IP2180
7
8
9
Control cards
Copy program from storage, compile, and execute.

Input Description

The input quantities are loaded by using the FORTRAN IV NAMELIST, with the exception of the IDENT cards that are in 8A10 format. The input symbols are as follows: (Recall that all NAMELIST input begins in columns 2 to 80. Also, see description below of "standard integration" for Q_{ij}^* values).

$IP2180

KPROG	control word: if less than 1, execute both programs D2181 and D2182; if equal to 1, execute only D2182.

$END

IDENT	identification information, one card in 8A10 format
$IP2181	(equation numbers below refer to the number given in ref. 1).
DEL	nominal value of width δ of subregion III. Default value = 0.04.
DHX	real part of downwash ratio on left-hand side of equation (18), $-\partial h_j/\partial x$, one value per control point
H	imaginary part of downwash ratio on left-hand side of equation (18) (ref. 1) not including k, -2h, one value per control point
HMLTPLY	optional two-word array that if nonzero multiplies: HMLTPLY(1) times DHX, HMLTPLY(2) times H. Default values are 0.0.
NMODE	number of modes of downwash (maximum of 6)
NSYM	control word for spanwise symmetry in lift series A_n^* of equation (13b) of reference 1 and A_n of equation (14) of reference 1: 0 for spanwise antisymmetry (odd powers on η^m), not 0 for spanwise symmetry (even powers on η^m). Default value = 1.
NXY	number of downwash control or collocation points, maximum of 48
S	ratio of wing semispan to root semichord
TANLE	tangent of leading-edge sweep angle, positive for sweepback
TANLTE	tangent of trailing-edge sweep angle, positive for sweepback
WT	array of weights used in the least-square-error solution of the simultaneous equations (32). WT(1) = -1.0 causes all equal weights. Default value = -1.0.
XK	reduced frequency k
XM	Mach number of undisturbed flow
X	array of x-coordinates of control points
Y	array of y-coordinates of control points
$END	
IDENT	identification information, one card in 8A10 format, may differ from preceding IDENT

$IP2182 (equation numbers below refer to the numbers given in ref. 2)

ALPHA array of values of α in flutter equation (A12), maximum of 25. ALPHA and MASS must be in consistent units.

CWCDI control word for chordwise integration in equation (11b) of reference 2: 0 for standard, ≠0 for nonstandard, integration. Default value = 0.

CWPR2,CWPR3,CWPR5, CWPR6, CWPR8,CWPR8X } control words for printing matrices (2), (3), (5), (6), (8), and (8x), respectively of equations (14), (16), and (18): =0, do not print; ≠0, print. Default value = 0.

CWPR4 control word for printing matrix (4), its integrals of $l_n^*(x,y_\sigma)$, and $h_i(x_{le},y_\sigma)$: =0, do not print; ≠0, print. Default value = 0.

CWSPI control word for spanwise integration of equations (11b) and (16): =0 for standard integration, ≠0 for nonstandard. Default value = 0.

CWSPST control word for span stations at which chordwise integrations are done: =0 for standard, ≠0 for nonstandard. Default value = 0.

G modal damping coefficients of structural, solid-friction damping, g_i. (See eq. (A7).) (Damping force proportional to amplitude, but in phase with velocity).

HIC three-dimensional array of input values of modal deflections from which the values of $h_i(x,y)$ of equation (14) are interpolated or used. In HIC (JSPAN, KCHORD, IMODE) subscripts JSPAN, KCHORD, and IMODE indicate the index of the span station, the chord station, and the deflection mode, respectively, of the value of HIC. JSPAN increases inboard to outboard, KCHORD increases from leading edge toward trailing edge.

ISPS input array of spanwise-integration factors, which the program multiplies by s to produce the I_σ of matrix (6) of equation (16) for nonstandard spanwise quadrature. Omit if CWSPI = 0.

INTOR order of chordwise interpolation (1 if linear, 2 if parabolic) used in obtaining the deflections h_i of equation (14) from the input data HIC.

MASS a two-dimensional array of generalized mass elements m_{ij} of equations (A3) and (A13).

NALPHA number of α values in input (see eq. (A12)), maximum of 25.

NCDST number of chord stations used in every chordwise quadrature, =10 or 20 for standard quadrature (with CWCDI = 0); a maximum of 20 for nonstandard.

NHP number of h (deflection) points input in HIC for each chord, maximum of 20

NPNCHQ	control word for punching matrix of Q_{ij}^* (QIJ) and seven other parameters for subsequent use in NAMELIST input: =1, punch; ≠1, do not punch. Default value = 0.
NRQIJ	control word: =1 when QIJ is included with IP2182 and its calculation is not needed; ≠1 when QIJ is to be calculated. Default value = 0.
NSPST	number of spanwise quadrature stations: 3, 5, 7, 9, or 11 for standard quadrature; a maximum of 11 for nonstandard
OMEGA	array of input modal frequencies ω_i
OMEGB	base or reference value ω_B (see eq. (3))
QIJ	nondimensional generalized-force-matrix elements Q_{ij}^* of equations (11), (18), and (A12)
XCBAR	array of nondimensional local chordwise coordinates $\bar{x}_c$ at which $h_i(x_c,y_\sigma)$ of equation (14) of reference 2 are input and used in nonstandard chordwise quadrature. (See eq. (B2).) Same for all chords. Omit for standard quadrature.
XIC	array of integrating factors for nonstandard chordwise quadrature, which the program multiplies by the local chord length to produce the $I_c(x_c)$ of matrix (2) (eq. (14)). Omit for standard integration.
XTAB	chordwise array of $\bar{x}$ (see eq. (B2)) at which the deflections HIC are input; the same for all span stations
YSPS	array of nondimensional span stations y/s for nonstandard spanwise quadrature. (See ISPS and NSPANST.)
$END	

The "standard integration" (for obtaining the generalized aerodynamic forces) referred to with certain of the input variables is that described in reference 2 and is summarized as follows: The chordwise integration (see eq. (13) of ref. 2) is done by Gaussian quadrature, the values of modal deflection h_i being interpolated (linearly or parabolically according to INTOR) at each span station from the deflection data HIC supplied at local chord locations XTAB. The span stations (at which the chordwise integrations are done) are equally spaced for spanwise integration by Simpson's rule for second-degree parabolas. That is, the wing semispan (from $y = 0$ to $y = s$) is divided into an even number of equal-sized parts, and this division results in an odd number of equally spaced span stations.

For nonstandard spanwise integration the user assigns CWSPI ≠ 0 and inputs the arrays ISPS of spanwise integrating factors and YSPS of span stations. The values of ISPS must be those for a unit semispan; the program multiplies them by s to obtain matrix 6. The values of YSPS are y/s. The array of input deflections HIC must, of course, be at the user-selected span stations YSPS.

For nonstandard chordwise integration, the user assigns CWCDI ≠ 0 and inputs the arrays XCBAR and XIC. Furthermore, the input deflections HIC are not interpolated, except for the leading-edge values, for use in the chordwise integration and must therefore be supplied for the chordwise locations of the user-specified quadrature scheme. The values of XIC must be those for a unit chord length; the program multiplies them by the local chord length $x_{te} - x_{le}$.

Listing of Input Data for Sample Case

```
$IP2180
KPROG=0,
$END
     HT-7 FLUTTER MODEL, 48 CPTS, 3 MODES,  8/22/73
$IP2181
XM=1.2, XK=.4, NMODE=3, TANLE=1.216, TANLTE=.35255, S=.81, NXY=48,
X(1)=.03,.2,.4,.6,.8,1.,  X(7)=.1915,.5635,.9355,
X(10)=.369,.541,.7131,.8851,1.0571,  X(15)=.3745,.5325,.8486,1.0066,
X(19)=.538,.6821,.8261,.9702,1.1142, X(24)=.5575,.6876,.8176,.9477,1.0778,
X(29)=.707,.8231,.9392,1.0553,1.1713, X(34)=.7405,.8426,.9447,1.0468,1.1488,
X(39)=.8761,.9642,1.0523,1.1404,1.2284, X(44)=.9255,.9976,1.0717,1.1458,1.22,
Y(1)=6*0.0,3*.081,5*.162,4*.243,5*.324,5*.405,5*.486,5*.567,5*.648,5*.729,
HMLTPLY(1)=1.0,HMLTPLY(2)=2.0,
DHX(  1,1)= -.185, -.381, -.503, -.267, -.450, -.678, -.334, -.418,
 -.627, -.476, -.473, -.455, -.553, -.619, -.664, -.424,
 -.605, -.601, -.569, -.518, -.587, -.636, -.629, -.654,
 -.573, -.588, -.657, -.648, -.657, -.604, -.687, -.699,
 -.663, -.734, -.696, -.705, -.739, -.733, -.772, -.812,
 -.785, -.808, -.841, -.881, -.873, -.816, -.887,-1.049,
DHX( 49,1)= 1.780, 1.830, 1.620, 1.170, 1.170, 1.310, 1.400, 1.370,
 1.070,  .827,  .782, 1.910, 1.440,  .698, 1.130,  .223,
 2.290,  .650,  .290,  .552, 2.600, 1.530,  .104,  .790,
 -.130, 1.740, 2.610,  .370,  .220,  .370, 1.940, 1.730,
  .990,  .530,  .290,  .240,  .620,  .960,  .420,  .220,
  .071,  .120,  .170,  .420,  .350,  .270,  .200,  .150,
DHX( 97,1)=  .330,  .440,  .440,  .140,  .058,  .130,  .310,  .230,
  .150,  .330,  .260,  .080,  .058,  .100,  .300,  .140,
 -.018,  .003, -.120, -.190, -.200, -.170, -.160, -.530,
 -.500, -.320, -.410, -.770, -.910, -.610, -.680,-1.650,
-2.170,-1.610, -.680,-1.680,-2.760,-1.940,-1.810,-2.350,
-2.990,-2.180,-1.530,-1.790,-1.670,-2.040,-2.170,-1.720,
H(  1,1)=  .2150,  .1690,  .0720, -.0045, -.0670, -.1890,  .1630, -.0090,
 -.1690,  .0785, -.0040, -.0830, -.1680, -.2710,  .0555, -.0310,
 -.1755, -.2730, -.0400, -.1170, -.1955, -.2850, -.3760, -.0790,
 -.1500, -.2330, -.3145, -.4000, -.2180, -.2900, -.3645, -.4466,
 -.5250, -.2900, -.3630, -.4340, -.5080, -.5800, -.4500, -.5200,
 -.5900, -.6600, -.7350, -.5550, -.6200, -.6850, -.7450, -.8200,
H( 49,1)= -.9500, -.6400, -.2875, -.0115,  .2130,  .4660, -.5850, -.1310,
  .3870, -.3380, -.2280,  .0080,  .3250,  .4870, -.3550, -.2610,
  .1920,  .4300, -.2650, -.2450, -.0055,  .3370,  .4160, -.2880,
 -.2610, -.1620,  .1450,  .3425, -.2575, -.2460, -.1090,  .0770,
  .2710, -.3000, -.2570, -.2340, -.1920, -.1080, -.3450, -.3160,
 -.3050, -.2975, -.2640, -.4880, -.4590, -.4350, -.4165, -.4055,
H( 97,1)= -.1130, -.0490,  .0460,  .1055,  .1160,  .1400, -.0075,  .1320,
  .1645,  .1075,  .1610,  .1890,  .1980,  .2130,  .1910,  .2275,
  .2325,  .2320,  .3205,  .2980,  .2690,  .2425,  .2190,  .4200,
```

```
 .3760,  .2960,  .2560,  .1775,  .3965,  .3060,  .2400,  .1100,
-.1240,  .4095,  .2960,  .1950, -.0485, -.2950,  .2110,  .0340,
-.2090, -.4465, -.6000, -.0660, -.3120, -.4740, -.5950, -.7250,
$END
    HT-7 FLUTTER MODEL, 48 CPTS, 3 MODES,  8/22/73
$IP2182
INTOR=2, NPNCHW=1, NSPST=11, NCOST=10,
HIC(1,1,1)=-.218,-.192,-.158,-.110,-.047,.036,.137,.252,.385,.525,.675,
HIC(1,2,1)=-.199,-.163,-.119,-.0557,-.0026,.079,.179,.29,.417,.558,.706,
HIC(1,3,1)=-.169,-.127,-.079,-.0073,.04,.12,.218,.327,.451,.591,.737,
HIC(1,4,1)=-.124,-.083,-.037,.031,.079,.159,.255,.363,.486,.623,.767,
HIC(1,5,1)=-.072,-.034,.0041,.062,.117,.196,.29,.398,.521,.655,.796,
HIC(1,6,1)=-.027,.009,.044,.093,.155,.233,.326,.434,.557,.686,.825,
HIC(1,7,1)=.0046,.044,.0829,.131,.196,.273,.365,.471,.592,.716,.856,
HIC(1,8,1)=.031,.077,.123,.176,.239,.315,.405,.508,.627,.748,.889,
HIC(1,9,1)=.067,.116,.168,.225,.285,.358,.446,.546,.662,.783,.925,
HIC(1,10,1)=.122,.169,.218,.273,.331,.400,.486,.584,.698,.821,.962,
HIC(1,11,1)=.188,.230,.271,.320,.376,.442,.525,.621,.735,.860,1.00,
HIC(1,1,2)=1.0,.721,.545,.455,.388,.35,.328,.328,.385,.504,.639,
HIC(1,2,2)=.622,.587,.429,.355,.309,.288,.284,.30,.364,.488,.621,
HIC(1,3,2)=.64,.464,.338,.288,.265,.257,.258,.275,.345,.473,.605,
HIC(1,4,2)=.459,.355,.282,.261,.259,.261,.254,.257,.328,.459,.591,
HIC(1,5,2)=.288,.249,.228,.233,.245,.254,.246,.245,.316,.447,.576,
HIC(1,6,2)=.138,.131,.136,.149,.165,.182,.202,.234,.309,.436,.558,
HIC(1,7,2)=.012,-.0046,-.008,-.006,.006,.034,.109,.218,.305,.427,.538,
HIC(1,8,2)=-.101,-.147,-.178,-.193,-.186,-.145,-.013,.192,.302,.419,.519,
HIC(1,9,2)=-.213,-.278,-.325,-.347,-.337,-.283,-.127,.155,.298,.412,.506,
HIC(1,10,2)=-.336,-.387,-.423,-.431,-.404,-.343,-.21,.108,.291,.406,.5,
HIC(1,11,2)=-.466,-.482,-.487,-.468,-.418,-.351,-.271,.058,.284,.4,.497,
HIC(1,1,3)=.122,.035,-.053,-.166,-.335,-.454,-.505,-.498,-.375,-.075,.534,
HIC(1,2,3)=.088,.0077,-.08,-.191,-.328,-.42,-.451,-.41,-.292,.066,.587,
HIC(1,3,3)=.049,-.025,-.108,-.213,-.321,-.385,-.397,-.341,-.211,.198,.636,
HIC(1,4,3)=.0016,-.064,-.136,-.227,-.311,-.351,-.347,-.298,-.129,.312,.68,
HIC(1,5,3)=-.046,-.103,-.161,-.235,-.298,-.321,-.306,-.26,-.034,.403,.719,
HIC(1,6,3)=-.084,-.132,-.179,-.236,-.284,-.298,-.274,-.195,.08,.474,.757,
HIC(1,7,3)=-.105,-.146,-.189,-.234,-.269,-.278,-.24,-.088,.209,.535,.798,
HIC(1,8,3)=-.114,-.151,-.194,-.233,-.256,-.256,-.19,.048,.337,.595,.843,
HIC(1,9,3)=-.118,-.154,-.198,-.232,-.243,-.223,-.11,.183,.446,.659,.894,
HIC(1,10,3)=-.127,-.165,-.205,-.232,-.231,-.178,0.0,.295,.53,.727,.946,
HIC(1,11,3)=-.14,-.18,-.213,-.232,-.219,-.126,.124,.388,.60,.798,1.0,
NHP=11,XTAB(1)=0.0,.1,.2,.3,.4,.5,.6,.7,.8,.9,1.0,
MASS(1,1)=.00035704, MASS(2,2)=.0005178, MASS(3,3)=.00026352,
OMEGA(1)=162.5, 391., 725., OMEGB=725., G(1)=3*0.0,
ALPHA(1)=.0004,.0006,.0007,.0008,.0009,.0010,.0011,.0012,.0013,.0014,.0015,
.0016,.0017,.0018,.0019,.0020,.0022,.0024,.0026,.0028,.0030,NALPHA=21,
$END
```

Discussion of Printed Output

The only printing from D2180 is one line that states the value of KPROG. Then follows from D2181 a heading, the contents of the IDENT card, and an echo of the NAMELIST IP2181. Then comes a heading identifying PART I, and the case data K, TANLTE, TANLA, S, MACH, DELNOM, NSYMMETRY, and number of control points NXY. Then follows, for each control point in sequence,

X	x
Y	y
DSL	D1L plus D2L (fig. 3, ref. 1)
D1L	width of subregion I, if any
D2L	width of subregion II
DSR	width of subregion IV
PTSD1L,PTSD2L, PTSDSR	one half the number of Gaussian quadrature stations in subregions I, II, and IV, respectively

the half-width $\delta/2$ of subregion III (the control-point strip), and a row of the II_{nm} matrix (16 complex elements).

After the II_{nm} matrix for the last control point is a heading that identifies the beginning of PRT2. Then, if the input includes a multiplier for the downwash input, a notice is printed regarding that multiplication. The complex downwash elements follow for each downwash mode. Then for each mode comes the 16-element complex $a_{nm}^{(j)}$ matrix and the matrix of residuals from the least-squares solution. (The sequence of subscripts nm in $a_{nm}^{(j)}$ is 00, 02, 04, 06, 10, 12, 14, 16, 20, 22, 24, 26, 30, 32, 34, 36 for spanwise symmetry. For spanwise antisymmetry, m is one unit higher: 01, 03, . . ., etc., . . . 37).

The output from D2182 begins with a heading, followed by the contents of the second IDENT card input, and by an echo of the NAMELIST IP2182 input. Then the input values follow for the case data MACH, K, TANLE, TANLTE, S, NMODS, and NSYM, the matrix of deflections $h_i(x,y)$, the matrix of generalized masses m_{ij}, the modal frequencies ω_i, the modal structural damping coefficients g_i, and the matrix of generalized aerodynamic force elements Q_{ij}^*. Finally, for each value of the air mass parameter α, there is listed α, the flutter determinant, the eigenvector columns $\{q_i\}$, the eigenvalues Ω, the eigenfrequencies ω, the structural damping coefficients g, and the stiffness parameters $b_0\omega_B/V = k\mathrm{Re}(\Omega)$. Where the eigenvalue Ω has a negative real part, ω, g, and $b_0\omega_B/V$ are printed as indefinite quantities.

Discussion of Punched Output

The punched output is obtained when NPNCHQ = 1 and consists of the QIJ matrix and the quantities MACH, K, TANLE, TANLTE, S, NMODS, and NSYM in the format for subsequent NAMELIST/IP2182/ input when flutter solutions are desired for additional air-density parameters ALPHA. For the latter usage, KPROG = 1 and NRQIJ = 1.

OUTPUT FOR SAMPLE CASE

```
                              NASA - LANGLEY RESEARCH CENTER - HAMPTON, VA.

     JEAN FOSTER FOR HERB CUNNINGHAM- D21801(INCLUDESD2181 AND D2182)

              UNSTEADY LIFTING-SURFACE THEORY BY THE SUPERSONIC KERNEL FUNCTION METHOD AND A
              GALERKIN MODAL FLUTTER ANALYSIS OF ARROWHEAD PLANFORMS WITH SUBSONIC LEADING
              EDGES AND SUPERSONIC TRAILING EDGES

              ***REF. 1 = AIAA JOUR., NOV.1966, P.1961-1968
              ***REF. 2 = NASA TN D-6012
              ***REF. 3 = NASA TM X-2913, THE DESCRIPTION OF THIS PROGRAM

   HT-7 FLUTTER MODEL, 48 CPTS, 3 MODES,  8/22/73

 ECHO OF NAMELIST IP2181 FOLLOWS

$IP2181

XK       =  0.4E+00,

XM       =  0.12E+01,

TANLE    =  0.1216E+01,

TANLTE   =  0.35255E+00,

S        =  0.81E+00,

NXY      =  48,

NSYM     =  1,

DEL      =  0.4E-01,

X        =  0.3E-01,  0.2E+00,  0.4E+00,  0.6E+00,  0.8E+00,  0.1E+01,
            0.1915E+00,  0.5635E+00,  0.9355E+00,  0.369E+00,  0.541E+00,
            0.7131E+00,  0.8851E+00,  0.10571E+01,  0.3745E+00,  0.5325E+00,
            0.8486E+00,  0.10066E+01,  0.538E+00,  0.6821E+00,  0.8261E+00,
            0.9702E+00,  0.11142E+01,  0.5575E+00,  0.6676E+00,  0.8176E+00,
            0.9477E+00,  0.10778E+01,  0.707E+00,  0.8231E+00,  0.9392E+00,
            0.10553E+01,  0.11713E+01,  0.7405E+00,  0.8426E+00,  0.9447E+00,
            0.10468E+01,  0.11488E+01,  0.8761E+00,  0.9642E+00,  0.10523E+01,
            0.11404E+01,  0.12284E+01,  0.9235E+00,  0.9976E+00,  0.10717E+01,
            0.11458E+01,  0.122E+01,

Y        =  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.81E-01,  0.81E-01,  0.81E-01,
            0.162E+00,  0.162E+00,  0.162E+00,  0.162E+00,  0.162E+00,
            0.243E+00,  0.243E+00,  0.243E+00,  0.243E+00,  0.324E+00,
            0.324E+00,  0.324E+00,  0.324E+00,  0.324E+00,  0.405E+00,
            0.405E+00,  0.405E+00,  0.405E+00,  0.405E+00,  0.486E+00,
            0.486E+00,  0.486E+00,  0.486E+00,  0.486E+00,  0.567E+00,
            0.567E+00,  0.567E+00,  0.567E+00,  0.567E+00,  0.648E+00,
            0.648E+00,  0.648E+00,  0.648E+00,  0.648E+00,  0.729E+00,
            0.729E+00,  0.729E+00,  0.729E+00,  0.729E+00,

DHX      = -0.185E+00, -0.381E+00, -0.503E+00, -0.267E+00, -0.45E+00,
           -0.678E+00, -0.334E+00, -0.418E+00, -0.627E+00, -0.476E+00,
           -0.473E+00, -0.455E+00, -0.553E+00, -0.619E+00, -0.664E+00,
           -0.424E+00, -0.605E+00, -0.601E+00, -0.569E+00, -0.518E+00,
           -0.587E+00, -0.636E+00, -0.629E+00, -0.654E+00, -0.573E+00,
           -0.588E+00, -0.657E+00, -0.648E+00, -0.657E+00, -0.604E+00,
           -0.687E+00, -0.699E+00, -0.663E+00, -0.734E+00, -0.696E+00,
           -0.705E+00, -0.739E+00, -0.733E+00, -0.772E+00, -0.812E+00,
           -0.785E+00, -0.808E+00, -0.841E+00, -0.881E+00, -0.873E+00,
           -0.816E+00, -0.887E+00, -0.1049E+01,  0.178E+01,  0.183E+01,
            0.162E+01,  0.117E+01,  0.117E+01,  0.131E+01,  0.14E+01,
```

```
            0.137E+01,  0.107E+01,  0.827E+00,  0.782E+00,  0.191E+01,
            0.144E+01,  0.698E+00,  0.113E+01,  0.223E+00,  0.229E+01,
            0.65E+00,  0.29E+00,  0.552E+00,  0.26E+01,  0.153E+01,  0.104E+00,
            0.79E+00, -0.13E+00,  0.174E+01,  0.261E+01,  0.37E+00,  0.22E+00,
            0.37E+00,  0.194E+01,  0.173E+01,  0.99E+00,  0.53E+00,  0.29E+00,
            0.24E+00,  0.62E+00,  0.96E+00,  0.42E+00,  0.22E+00,  0.71E-01,
            0.12E+00,  0.17E+00,  0.42E+00,  0.35E+00,  0.27E+00,  0.2E+00,
            0.15E+00,  0.33E+00,  0.44E+00,  0.44E+00,  0.14E+00,  0.58E-01,
            0.13E+00,  0.31E+00,  0.23E+00,  0.15E+00,  0.33E+00,  0.26E+00,
            0.8E-01,  0.58E-01,  0.1E+00,  0.3E+00,  0.14E+00, -0.18E-01,
            0.3E-02, -0.12E+00, -0.19E+00, -0.2E+00, -0.17E+00, -0.16E+00,
           -0.53E+00, -0.5E+00, -0.32E+00, -0.41E+00, -0.77E+00, -0.91E+00,
           -0.61E+00, -0.68E+00, -0.165E+01, -0.217E+01, -0.161E+01, -0.68E+00,
           -0.168E+01, -0.276E+01, -0.194E+01, -0.181E+01, -0.235E+01,
           -0.299E+01, -0.218E+01, -0.153E+01, -0.179E+01, -0.167E+01,
           -0.204E+01, -0.217E+01, -0.172E+01, I, I, I, I, I, I, I, I, I, I, I, I,
           I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
           I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
           I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
           I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
           I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
           I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,

H        =  0.215E+00,  0.169E+00,  0.72E-01, -0.45E-02, -0.67E-01, -0.189E+00,
            0.163E+00, -0.9E-02, -0.169E+00,  0.785E-01, -0.4E-02, -0.83E-01,
           -0.168E+00, -0.271E+00,  0.555E-01, -0.31E-01, -0.1755E+00,
           -0.273E+00, -0.4E-01, -0.117E+00, -0.1955E+00, -0.285E+00,
           -0.376E+00, -0.79E-01, -0.15E+00, -0.233E+00, -0.3145E+00, -0.4E+00,
           -0.218E+00, -0.29E+00, -0.3645E+00, -0.4466E+00, -0.525E+00,
           -0.29E+00, -0.363E+00, -0.434E+00, -0.508E+00, -0.58E+00, -0.45E+00,
           -0.52E+00, -0.59E+00, -0.66E+00, -0.735E+00, -0.555E+00, -0.62E+00,
           -0.685E+00, -0.745E+00, -0.82E+00, -0.95E+00, -0.64E+00, -0.2875E+00,
           -0.115E-01,  0.213E+00,  0.466E+00, -0.585E+00, -0.131E+00,
            0.387E+00, -0.338E+00, -0.228E+00,  0.8E-02,  0.325E+00,  0.487E+00,
           -0.355E+00, -0.261E+00,  0.192E+00,  0.45E+00, -0.265E+00,
           -0.245E+00, -0.55E-02,  0.337E+00,  0.416E+00, -0.288E+00,
           -0.261E+00, -0.182E+00,  0.145E+00,  0.3425E+00, -0.2575E+00,
           -0.246E+00, -0.109E+00,  0.77E-01,  0.271E+00, -0.3E+00, -0.257E+00,
           -0.234E+00, -0.192E+00, -0.108E+00, -0.345E+00, -0.316E+00,
           -0.305E+00, -0.2975E+00, -0.284E+00, -0.468E+00, -0.459E+00,
           -0.435E+00, -0.4185E+00, -0.4055E+00, -0.113E+00, -0.49E-01,
            0.46E-01,  0.1955E+00,  0.118E+00,  0.14E+00, -0.75E-02,  0.132E+00,
            0.1645E+00,  0.1075E+00,  0.161E+00,  0.189E+00,  0.198E+00,
            0.213E+00,  0.191E+00,  0.2275E+00,  0.2325E+00,  0.232E+00,
            0.3205E+00,  0.298E+00,  0.269E+00,  0.2425E+00,  0.219E+00,
            0.42E+00,  0.376E+00,  0.298E+00,  0.256E+00,  0.1775E+00,
            0.3965E+00,  0.306E+00,  0.24E+00,  0.11E+00, -0.124E+00,
            0.4095E+00,  0.298E+00,  0.195E+00, -0.465E-01, -0.295E+00,
            0.211E+00,  0.34E-01, -0.209E+00, -0.4465E+00, -0.6E+00, -0.66E-01,
           -0.312E+00, -0.474E+00, -0.595E+00, -0.725E+00, I, I, I, I, I, I, I,
           I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
           I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
           I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
           I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
           I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
           I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,

NMODE    =  3,

WT       = -0.1E+01, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
           I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
           I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
           I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
           I, I, I, I, I, I,

HMLTPLY  =  0.1E+01,  0.2E+01,

$END
```

PART I OF 02181 *** GENERATE IINM MATRIX

K	TANLTE	TANLA	S	MACH	DELNOM	NSYMMETRY
4.00000000E-01	3.52550000E-01	1.21600000E+00	8.10000000E-01	1.20000000E+00	4.00000000E-02	1

NO. OF C.PTS.(=NXY)
48

X	Y	DSL	D1L	D2L	DSR
3.00000000E-02	0.	1.43668608E-02	0.	1.43668608E-02	1.43668608E-02

PTSD1L	PTSD2L	PTSDSR
0	5	5

HALF-WIDTH OF CONTROL POINT STRIP 1.59631786E-03

ROW 1 OF COEFFICIENT MATRIX IINM

RE	IMAG	RE	IMAG	RE	IMAG
4.74004853E+00	1.03609188E-01	-8.76460106E-04	7.29465038E-06	-6.90127313E-08	1.07124020E-09
-1.02001030E-11	1.91804522E-13	4.37199271E+00	9.62853525E-02	-8.29653527E-04	6.90942694E-06
-6.57201494E-06	1.02030759E-09	-9.73990773E-12	1.83163797E-13	3.67640126E+00	8.23672533E-02
-7.39906848E-04	6.17071446E-06	-5.93872373E-08	9.22360544E-10	-8.65359850E-12	1.66524951E-13
2.73219527E+00	6.32135190E-02	-6.14380432E-04	5.13573241E-06	-5.04648391E-08	7.84296966E-10
-7.60078415E-12	1.43001278E-13				

X	Y	DSL	D1L	D2L	DSR
2.00000000E-01	0.	9.57790717E-02	0.	9.57790717E-02	9.57790717E-02

PTSD1L	PTSD2L	PTSDSR
0	5	5

HALF-WIDTH OF CONTROL POINT STRIP 1.06421191E-02

ROW 2 OF COEFFICIENT MATRIX IINM

RE	IMAG	RE	IMAG	RE	IMAG
4.76746440E+00	6.86743974E-01	-3.87969774E-02	2.14711711E-03	-1.35192277E-04	1.40166885E-05
-8.86148811E-07	1.11560034E-07	2.20694221E+00	3.67561178E-01	-2.53123758E-02	1.42477625E-03
-9.43559432E-05	9.84505588E-06	-6.36799452E-07	8.04343119E-08	-1.00520425E+00	-6.05331113E-02
-5.64251767E-03	3.64017230E-04	-3.27397959E-05	3.54256769E-06	-2.55113560E-07	3.27819066E-08
-2.48155710E+00	-3.31300079E-01	1.04093482E-02	-5.32759607E-04	2.30910575E-05	-2.23929356E-06
1.06425113E-07	-1.26297046E-08				

X	Y	DSL	D1L	D2L	DSR
4.00000000E-01	0.	1.92842382E-01	0.	1.92842382E-01	1.92842382E-01

PTSD1L	PTSD2L	PTSDSR
0	5	8

HALF-WIDTH OF CONTROL POINT STRIP 2.00000000E-02

ROW 3 OF COEFFICIENT MATRIX IINM

RE	IMAG	RE	IMAG	RE	IMAG
4.84447078E+00	1.34989417E+00	-1.53326171E-01	1.68377327E-02	-2.10897208E-03	4.39873234E-04
-5.49176682E-05	1.40100733E-05	-5.59641912E-01	1.09921166E-01	-4.96076225E-02	6.08943467E-03
-9.10714940E-04	1.97248072E-04	-2.64197258E-05	6.87044663E-06	-3.19926747E+00	-7.46701082E-01
4.75762074E-02	-4.38747998E-03	3.603649296E-04	-6.33655145E-05	5.39130707E-06	-1.14210452E-06
4.06310662E-01	-2.93899561E-01	5.43360544E-02	-6.20152507E-03	7.78678564E-04	-1.60235851E-04
1.94886565E-05	-4.87864609E-06				

X	Y	DSL	D1L	D2L	DSR
6.00000000E-01	0.	2.99263572E-01	0.	2.99263572E-01	2.99263572E-01

PTSD1L	PTSD2L	PTSDSR
0	8	8

HALF-WIDTH OF CONTROL POINT STRIP 2.00000000E-02

ROW 4 OF COEFFICIENT MATRIX IINM

RE	IMAG	RE	IMAG	RE	IMAG
4.95201113E+00	1.97315278E+00	-3.36448091E-01	5.50398283E-02	-1.02419065E-02	3.23555583E-03
-5.92830471E-04	2.31969971E-04	-3.60805086E+00	-7.25121470E-01	-3.51690474E-03	4.92928451E-03
-2.01838955E-03	7.55374178E-04	-1.67374118E-04	7.03671205E-05	-1.12352373E+00	-8.39407132E-01
1.40584350E-01	-2.07047544E-02	3.31815905E-03	-9.39446653E-04	1.51661883E-04	-5.36833647E-05
4.59089168E+00	9.69440398E-01	-6.37356790E-03	-3.73870823E-03	1.79795043E-03	-6.49420111E-04
1.41057456E-04	-5.69578480E-05				

X	Y	DSL	D1L	D2L	DSR
8.00000000E-01	0.	4.04400525E-01	0.	4.04400525E-01	4.04400525E-01

PTSD1L	PTSD2L	PTSDSR
0	8	8

HALF-WIDTH OF CONTROL POINT STRIP 2.12842382E-02

ROW 5 OF COEFFICIENT MATRIX IINM

RE	IMAG	RE	IMAG	RE	IMAG
5.06894626E+00	2.55167493E+00	-5.86907826E-01	1.25079596E-01	-3.05792844E-02	1.30571803E-02
-3.08885664E-03	1.66421589E-03	-6.95580079E+00	-2.07654832E+00	1.70743303E-01	-1.98450203E-02

4.06436978E-04	6.59964653E-04	-3.53137244E-04	2.53319436E-04	5.99001315E+00	7.35065805E-01
1.20525016E-01	-3.01982141E-02	8.61544063E-03	-3.43469930E-03	8.02524663E-04	-3.93955619E-04
-1.24186023E+00	1.04570932E+00	-1.53175126E-01	2.23833921E-02	-1.56370745E-03	3.12195010E-05
1.77194015E-04	-1.36466489E-04				

X	Y	DSL	D1L	D2L	DSR
1.00000000E+00	0.	5.05500656E-01	0.	5.05500656E-01	5.05500656E-01

PTSD1L	PTSD2L	PTSDSR
0	8	8

HALF-WIDTH OF CONTROL POINT STRIP 2.66052977E-02

ROW 6 OF COEFFICIENT MATRIX IINM

RE	IMAG	RE	IMAG	RE	IMAG
5.16786783E+00	3.09618088E+00	-8.90775565E-01	2.32365561E-01	-6.95388044E-02	3.77732259E-02
-1.06878385E-02	7.51519221E-03	-1.05856866E+01	-3.89439143E+00	5.20834293E-01	-9.06709427E-02
1.44747555E-02	-4.14158903E-03	3.60676713E-04	1.99284872E-04	1.88813716E+01	4.92665916E+00
-2.64629933E-01	2.30348076E-02	6.08011745E-03	-4.42160746E-03	1.72663320E-03	-1.16598689E-03
-3.08003618E+01	-5.31125536E+00	1.26781383E-01	8.03020836E-03	-5.63951917E-03	2.62042586E-03
-4.33140850E-04	1.31554291E-04				

Output for downwash points 7 to 47 not shown

X	Y	DSL	D1L	D2L	DSR
1.22000000E+00	7.29000000E-01	1.10311439E+00	3.19457665E-01	7.83656703E-01	1.59728842E-01

PTSD1L	PTSD2L	PTSDSR
8	8	5

HALF-WIDTH OF CONTROL POINT STRIP 1.77476492E-02

ROW 48 OF COEFFICIENT MATRIX IINM

RE	IMAG	RE	IMAG	RE	IMAG
5.54418511E+00	3.62648213E+00	4.91739907E+00	1.33422667E+00	2.44403309E+00	7.19471056E-01
8.07875314E-01	4.33957400E-01	-5.54970799E-01	-2.96973966E+00	-2.29791531E+00	-8.29958850E-01
-1.71440132E+00	-3.53837521E-01	-1.04869191E+00	-1.69023457E-01	-8.38903682E-01	1.84892788E+00
6.18596692E-01	4.63570668E-01	6.49162815E-01	1.72089808E-01	4.66007649E-01	7.24498117E-02
2.30720796E-01	-4.30066946E-01	-1.35948536E-01	-1.72058246E-01	-1.78162916E-01	-6.03413229E-02
-1.36770096E-01	-2.41181865E-02				

PART II OF D2161 *** SOLVE SIMULTANEOUS EQUATIONS FOR ANN

*****NOTICE- INPUT DOWNWASH QUANTITIES HAVE BEEN MULTIPLIED AS FOLLOWS - BY 1.000 FOR THE SLOPES DHX, AND
BY 2.000 FOR THE DEFLECTIONS H ****

DOWNWASH W/V, MODEL 1?, NXY REAL PARTS FIRST, IN CONTROL POINT SEQUENCE

-1.85000000E-01	-3.81000000E-01	-5.03000000E-01	-2.67000000E-01	-4.50000000E-01	-6.78000000E-01
-3.34000000E-01	-4.18000000E-01	-6.27000000E-01	-4.76000000E-01	-4.73000000E-01	-4.55000000E-01
-5.53000000E-01	-6.19000000E-01	-6.64000000E-01	-4.24000000E-01	-6.05000000E-01	-6.01000000E-01
-5.69000000E-01	-5.18000000E-01	-5.87000000E-01	-6.36000000E-01	-6.29000000E-01	-6.54000000E-01
-5.73000000E-01	-5.88000000E-01	-6.57000000E-01	-6.48000000E-01	-6.57000000E-01	-6.04000000E-01
-6.87000000E-01	-6.99000000E-01	-6.63000000E-01	-7.34000000E-01	-6.96000000E-01	-7.05000000E-01
-7.39000000E-01	-7.33000000E-01	-7.72000000E-01	-8.12000000E-01	-7.85000000E-01	-8.08000000E-01
-8.41000000E-01	-8.81000000E-01	-8.73000000E-01	-8.16000000E-01	-8.87000000E-01	-1.04900000E+00
1.72000000E-01	1.35200000E-01	5.76000000E-02	-3.60000000E-03	-5.36000000E-02	-1.51200000E-01
1.30400000E-01	-7.20000000E-03	-1.35200000E-01	6.28000000E-02	-3.20000000E-03	-6.64000000E-02
-1.34400000E-01	-2.16800000E-01	4.44000000E-02	-2.48000000E-02	-1.40400000E-01	-2.18400000E-01
-3.20000000E-02	-9.36000000E-02	-1.56400000E-01	-2.28000000E-01	-3.00600000E-01	-6.32000000E-02
-1.20000000E-01	-1.86400000E-01	-2.51600000E-01	-3.20000000E-01	-1.74400000E-01	-2.32000000E-01
-2.91600000E-01	-3.57280000E-01	-4.20000000E-01	-2.32000000E-01	-2.90400000E-01	-3.47200000E-01
-4.06400000E-01	-4.64000000E-01	-3.60000000E-01	-4.16000000E-01	-4.72000000E-01	-5.28000000E-01
-5.88000000E-01	-4.44000000E-01	-4.96000000E-01	-5.48000000E-01	-5.96000000E-01	-6.56000000E-01

DOWNWASH W/V, MODE(2), NXY REAL PARTS FIRST, IN CONTROL POINT SEQUENCE

1.78000000E+00	1.83000000E+00	1.62000000E+00	1.17000000E+00	1.17000000E+00	1.31000000E+00
1.40000000E+00	1.37000000E+00	1.07000000E+00	8.27000000E-01	7.82000000E-01	1.91000000E+00
1.44000000E+00	6.98000000E-01	1.13000000E+00	2.23000000E-01	2.29000000E+00	6.50000000E-01
2.90000000E-01	5.52000000E-01	2.60000000E+00	1.53000000E+00	1.04000000E-01	7.90000000E-01
-1.30000000E-01	1.74000000E+00	2.61000000E+00	3.70000000E-01	2.20000000E-01	3.70000000E-01
1.94000000E+00	1.73000000E+00	9.90000000E-01	5.30000000E-01	2.90000000E-01	2.40000000E-01
6.20000000E-01	9.60000000E-01	4.20000000E-01	2.20000000E-01	7.10000000E-02	1.20000000E-01
1.70000000E-01	4.20000000E-01	3.50000000E-01	2.70000000E-01	2.00000000E-01	1.50000000E-01
-7.60000000E-01	-5.12000000E-01	-2.30000000E-01	-9.20000000E-03	1.70400000E-01	3.72800000E-01
-4.68000000E-01	-1.04800000E-01	3.09600000E-01	-2.70400000E-01	-1.82400000E-01	6.40000000E-03
2.60000000E-01	3.89600000E-01	-2.64000000E-01	-2.06800000E-01	1.53600000E-01	3.44000000E-01
-2.12000000E-01	-1.96000000E-01	-4.40000000E-03	2.69600000E-01	3.34400000E-01	-2.30400000E-01
-2.08800000E-01	-1.45600000E-01	1.16000000E-01	2.74000000E-01	-2.36000000E-01	-1.96800000E-01
-8.72000000E-02	6.16000000E-02	2.16800000E-01	-2.40000000E-01	-2.05600000E-01	-1.87200000E-01
-1.53600000E-01	-8.64000000E-02	-2.76000000E-01	-2.52800000E-01	-2.44000000E-01	-2.38000000E-01
-2.27200000E-01	-3.90400000E-01	-3.67200000E-01	-3.48000000E-01	-3.34800000E-01	-3.24400000E-01

DOWNWASH W/V, MODE(3), NXY REAL PARTS FIRST, IN CONTROL POINT SEQUENCE

3.30000000E-01	4.40000000E-01	4.40000000E-01	1.40000000E-01	5.80000000E-02	1.30000000E-01
3.10000000E-01	2.30000000E-01	1.50000000E-01	3.30000000E-01	2.60000000E-01	8.00000000E-02
5.80000000E-02	1.00000000E-01	3.00000000E-01	1.40000000E-01	-1.60000000E-02	3.00000000E-03
-1.20000000E-01	-1.90000000E-01	-2.00000000E-01	-1.70000000E-01	-1.60000000E-01	-5.30000000E-01
-5.00000000E-01	-3.20000000E-01	-4.10000000E-01	-7.70000000E-01	-9.10000000E-01	-6.10000000E-01
-6.80000000E-01	-1.65000000E+00	-2.17000000E+00	-1.61000000E+00	-6.80000000E-01	-1.68000000E+00
-2.76000000E+00	-1.94000000E+00	-1.81000000E+00	-2.35000000E+00	-2.99000000E+00	-2.18000000E+00
-1.53000000E+00	-1.79000000E+00	-1.67000000E+00	-2.04000000E+00	-2.17000000E+00	-1.72000000E+00
-9.04000000E-02	-3.92000000E-02	3.68000000E-02	6.44000000E-02	9.44000000E-02	1.12000000E-01
-6.00000000E-03	1.05600000E-01	1.31600000E-01	8.60000000E-02	1.28800000E-01	1.51200000E-01
1.58400000E-01	1.70400000E-01	1.52800000E-01	1.82000000E-01	1.86000000E-01	1.85600000E-01
2.56400000E-01	2.38400000E-01	2.15200000E-01	1.94000000E-01	1.75200000E-01	3.36000000E-01
3.00800000E-01	2.38400000E-01	2.04800000E-01	1.42000000E-01	3.17200000E-01	2.44800000E-01
1.92000000E-01	8.80000000E-02	-9.92000000E-02	3.27600000E-01	2.38400000E-01	1.56000000E-01
-3.88000000E-02	-2.76000000E-01	1.68800000E-01	2.72000000E-02	-1.67200000E-01	-3.57200000E-01
-4.80000000E-01	-5.28000000E-02	-2.49600000E-01	-3.79200000E-01	-4.76000000E-01	-5.80000000E-01

WEIGHTING FACTORS ANM, MODE(1), 16 REAL PARTS FIRST

-7.60066633E-02	-2.91381947E-01	8.61254616E-01	-9.80036329E-01	1.76614927E-02	-3.84011078E-01
1.19476531E+00	-7.10933995E-01	1.36479880E-02	-4.57041111E-01	1.42233745E+00	-1.49394605E+00
1.34739163E-02	-2.37714097E-01	6.50328748E-01	-4.96627474E-01	3.38066663E-02	-4.78844680E-02
-9.99369462E-02	1.10799670E-01	7.14076992E-03	4.91078733E-02	-2.84021909E-03	-1.28622790E-01
-1.32089031E-03	-3.92313979E-03	4.55482322E-02	6.18161900E-02	-9.03971905E-04	4.27151488E-04
-3.93292997E-02	5.39720891E-02				

LEAST SQUARE RESIDUALS, MODE(1), NXY REAL PARTS FIRST, IN CONTROL POINT SEQUENCE

-1.40368197E-02	5.14616231E-03	6.67687605E-02	-1.53614478E-01	1.90963686E-02	3.98122887E-03
-5.54251023E-02	-1.36290675E-02	7.46636149E-02	2.58029329E-02	2.81163384E-02	-1.11233311E-02
2.23592300E-02	-7.60066575E-02	1.36256012E-01	-4.63261613E-02	4.72127854E-02	-1.68964731E-02
2.33997588E-02	-6.33306516E-03	6.91582489E-03	7.85050900E-04	3.22263846E-02	-3.47412007E-02
-1.66150267E-02	-1.20498169E-02	5.07382470E-03	-9.48553550E-03	-3.43580068E-02	-4.38116075E-02
1.93656313E-02	1.93670617E-03	-1.64246241E-03	-2.57230413E-02	-1.09621469E-02	2.38818190E-03
1.21659485E-02	-1.56697204E-02	3.21601424E-02	7.16822647E-02	1.78352774E-02	-3.69795739E-03
-1.68676468E-02	-6.18429785E-03	1.93297660E-02	-5.27460862E-02	-3.67925728E-02	4.20281262E-02
7.93526353E-03	-6.22542081E-03	-7.18327726E-03	-7.78327791E-03	-1.26357967E-02	-7.03206410E-03
2.11295984E-03	2.79330821E-03	7.68096568E-03	-4.22358801E-03	-2.04165419E-03	6.58585663E-03
1.26639926E-02	-2.30384025E-03	5.10579075E-03	1.05771184E-02	7.04521101E-03	7.98236805E-03
-1.87920746E-03	-1.04731788E-03	-4.34967372E-04	3.09172418E-03	-6.39621310E-03	-5.54350711E-03
-6.62925109E-03	-3.92114983E-03	-3.09463881E-03	-2.51790380E-03	2.06190816E-03	9.35712113E-05
-3.39972060E-03	-7.34514273E-04	2.52350095E-04	-1.87566899E-03	2.51223977E-03	7.00903305E-04
3.15686000E-04	-1.28167609E-03	8.03130016E-04	2.40299415E-03	3.66313035E-03	3.71182105E-03
1.91237765E-02	1.37095461E-03	-2.00697232E-03	-3.29009913E-04	-3.02201741E-03	-1.66841635E-03

WEIGHTING FACTORS ANM, MODE(2), 16 REAL PARTS FIRST

2.51707632E-01	-6.58766466E-02	-1.73364356E+00	2.51002594E+00	1.13301666E-01	1.52159513E-01
-2.51746657E+00	2.40699301E+00	4.72066745E-02	2.84120094E+00	-9.30114202E+00	6.70869273E+00
-1.19401565E-02	2.51891131E+00	-5.63147821E+00	2.74302984E+00	-1.14419510E-01	7.91441028E-02
-2.05068191E-01	2.20425241E-02	-5.01296610E-02	2.30164671E-01	-2.53037340E-01	4.34366000E-01
-9.19805632E-03	1.19103841E-01	-6.65519917E-01	7.77326568E-01	4.84439411E-03	-3.22674902E-02
-6.26792666E-01	7.57484039E-01				

LEAST SQUARE RESIDUALS, MODE(2), NXY REAL PARTS FIRST, IN CONTROL POINT SEQUENCE

6.33799429E-02	-2.98536555E-01	-1.94528549E-01	1.16074986E-01	-2.05923652E-01	1.41736206E-02
1.38734860E-01	-4.02469196E-02	3.61536587E-03	1.35904135E-01	3.74127613E-01	-3.40546227E-01
6.01440154E-02	-7.01825572E-02	-2.99106927E-01	4.77618468E-01	-2.04534667E-01	5.27254696E-01
4.22943962E-02	3.94978103E-01	-5.55759151E-01	5.95191354E-01	-5.92450692E-01	-1.90193392E-01
4.59237344E-01	-4.69215114E-01	-4.63388224E-01	1.06003308E+00	3.74022205E-02	7.08144541E-02
-6.82753232E-01	-2.87076108E-02	-4.42294479E-01	8.24465922E-02	-1.20968839E-01	1.55739920E-01
2.42065557E-01	2.96608098E-02	-1.53068456E-01	-6.75187859E-02	1.24035626E-01	1.87857856E-01
1.40152825E-01	1.70454800E-03	1.07454933E-01	3.41937767E-02	-8.94513130E-02	-1.71956851E-01
-8.50766321E-02	4.53138240E-02	1.12730970E-02	-3.65271565E-02	-1.02177879E-02	4.49928203E-03
-6.19700206E-03	1.08005265E-02	1.01646909E-02	-1.67314537E-02	1.60786675E-02	1.23026494E-02
-2.18315413E-02	-2.01014374E-02	6.42294651E-03	-2.29341076E-02	5.61652066E-03	2.13479873E-02
-3.62670521E-02	7.06348682E-03	1.61466236E-02	-2.42871026E-03	1.19020040E-02	2.49363646E-02
-2.29626517E-02	1.56425751E-02	-3.70626323E-02	4.82664024E-03	-8.69647798E-03	-8.07346850E-04
-9.60313066E-03	-1.35382398E-03	-3.40193525E-02	7.38301581E-03	-9.13265560E-03	1.98964190E-04
2.28011324E-02	3.22890939E-02	-6.29593951E-03	3.56604214E-03	7.60968695E-03	8.49210762E-03
5.99740458E-03	2.31815899E-04	6.46460028E-03	3.86104437E-03	-5.78891345E-03	-2.34895018E-02

WEIGHTING FACTORS ANM, MODE(3), 16 REAL PARTS FIRST

5.52630590E-02	-1.53062741E-01	-2.50266116E+00	3.78740023E+00	3.75081691E-02	-5.68663904E-02
4.56921246E-01	-1.19391256E+00	-1.60245603E-02	4.20232532E-01	-5.18663634E+00	8.13061014E+00
-1.47411662E-02	1.93971364E-01	-1.06140513E-01	-5.06987179E+00	-2.43199945E-04	2.41287364E-01
-2.29390556E-01	-2.63266509E-01	-2.81757078E-02	7.26936562E-05	2.60471049E-01	1.03055026E-01
5.87859177E-03	1.53737130E-02	8.95242954E-02	-4.14519773E-01	1.20601116E-03	1.91249909E-03
2.46981052E-03	5.86136587E-01				

LEAST SQUARE RESIDUALS, MODE(3), NXY REAL PARTS FIRST, IN CONTROL POINT SEQUENCE

-8.67875318E-03	-2.08182970E-02	-7.76513776E-02	3.80751301E-02	-9.06083411E-02	-1.58952280E-01
8.32877632E-02	-1.20678251E-02	-1.57428215E-01	-4.15165518E-02	-4.50320906E-02	5.07973503E-02
6.62269199E-02	2.51684766E-01	-1.43276567E-01	5.81568431E-03	1.31765688E-01	1.82658499E-01
9.57276075E-02	1.76277243E-01	1.35676567E-01	9.45973100E-03	-1.99505704E-01	1.19107960E-01
2.12037871E-01	-6.87827617E-02	-2.06077554E-01	-2.12315529E-01	1.38759857E-01	-2.20854845E-01
-4.70027976E-01	1.29376196E-01	2.43617420E-01	-2.95255357E-03	-6.64635231E-01	-1.34336148E-02
6.67266338E-01	-2.15993445E-01	-1.55225233E-02	3.40266010E-01	4.74161502E-01	-3.73290653E-01
-2.42760712E-02	1.64943035E-03	7.73336475E-02	-1.19445648E-01	-2.64187855E-01	1.78009211E-01
-2.33089947E-03	1.48333915E-02	1.35311269E-02	1.04818850E-04	-5.39261632E-04	2.21456780E-03
-7.44643122E-03	-1.13812699E-02	-1.22516228E-02	4.39744255E-03	-1.92268589E-03	-7.45544516E-03
-5.11649251E-03	-3.51034792E-04	1.18952919E-03	-4.31643512E-03	1.01162209E-02	1.11714225E-02
-1.24166313E-02	4.22952793E-03	1.69042465E-02	1.35948155E-02	1.05831684E-02	-1.22944156E-02
-8.74330436E-03	1.86967563E-02	-7.50417435E-03	-2.32031607E-02	1.37337040E-02	2.04608347E-02
-1.80804527E-02	-4.68018543E-02	5.43122777E-03	2.04223211E-02	-2.72020951E-03	-3.17303018E-02
6.42346189E-03	1.54578800E-02	-2.73077766E-02	-2.14971466E-02	2.75996399E-02	4.00380165E-02
2.34891057E-03	-8.78343193E-04	1.75417709E-02	1.22642137E-02	-2.00251735E-02	-1.79669131E-02

PROGRAM D2182 ... COMPUTE GENERALIZED AERODYNAMIC FORCES AND SOLVE FLUTTER DETERMINANTS

HT-7 FLUTTER MODEL, 48 CPTS, 3 MODES, 8/22/73

ECHO OF NAMELIST IP2182 FOLLOWS

```
$IP2182

NSPST   =  11,

NCDST   =  10,

MASS    =  0.35704E-03,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.5176E-03,  0.0,
           0.0,  0.0,  0.0,  0.0,  0.0,  0.26352E-03,  0.0,  0.0,  0.0,  0.0,
           0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,
           0.0,  0.0,  0.0,  0.0,  0.0,  0.0,

OMEGA   =  0.1625E+03,  0.391E+03,  0.725E+03, I, I, I,

OMEGB   =  0.725E+03,

G       =  0.0,  0.0,  0.0, I, I, I,

NALPHA  =  21,

ALPHA   =  0.4E-03,  0.6E-03,  0.7E-03,  0.8E-03,  0.9E-03,  0.1E-02,
           0.11E-02,  0.12E-02,  0.13E-02,  0.14E-02,  0.15E-02,  0.16E-02,
           0.17E-02,  0.18E-02,  0.19E-02,  0.2E-02,  0.22E-02,  0.24E-02,
           0.26E-02,  0.28E-02,  0.3E-02, I, I, I, I,

YSPS    = I, I, I, I, I, I, I, I, I, I, I,

ISPS    = I, I, I, I, I, I, I, I, I, I, I,

HIC     = -0.218E+00, -0.192E+00, -0.158E+00, -0.11E+00, -0.47E-01,  0.36E-01,
            0.137E+00,  0.252E+00,  0.385E+00,  0.525E+00,  0.675E+00,
          -0.199E+00, -0.163E+00, -0.119E+00, -0.557E-01, -0.26E-02,  0.79E-01,
            0.179E+00,  0.29E+00,  0.417E+00,  0.556E+00,  0.706E+00,
          -0.169E+00, -0.127E+00, -0.79E-01, -0.73E-02,  0.4E-01,  0.12E+00,
            0.218E+00,  0.327E+00,  0.451E+00,  0.591E+00,  0.737E+00,
          -0.124E+00, -0.83E-01, -0.37E-01,  0.31E-01,  0.79E-01,  0.159E+00,
            0.255E+00,  0.363E+00,  0.486E+00,  0.623E+00,  0.767E+00,
          -0.72E-01, -0.34E-01,  0.41E-02,  0.62E-01,  0.117E+00,  0.196E+00,
            0.29E+00,  0.398E+00,  0.521E+00,  0.655E+00,  0.796E+00, -0.27E-01,
            0.9E-02,  0.44E-01,  0.93E-01,  0.155E+00,  0.233E+00,  0.326E+00,
            0.434E+00,  0.557E+00,  0.686E+00,  0.825E+00,  0.46E-02,  0.44E-01,
            0.829E-01,  0.131E+00,  0.196E+00,  0.275E+00,  0.365E+00,
            0.471E+00,  0.592E+00,  0.716E+00,  0.856E+00,  0.31E-01,  0.77E-01,
            0.123E+00,  0.176E+00,  0.239E+00,  0.315E+00,  0.405E+00,
            0.508E+00,  0.627E+00,  0.748E+00,  0.889E+00,  0.67E-01,
            0.116E+00,  0.168E+00,  0.225E+00,  0.285E+00,  0.356E+00,
            0.446E+00,  0.546E+00,  0.662E+00,  0.785E+00,  0.925E+00,
            0.122E+00,  0.169E+00,  0.218E+00,  0.273E+00,  0.331E+00,  0.4E+00,
            0.486E+00,  0.584E+00,  0.698E+00,  0.821E+00,  0.962E+00,
            0.188E+00,  0.23E+00,  0.271E+00,  0.32E+00,  0.376E+00,  0.442E+00,
            0.525E+00,  0.621E+00,  0.735E+00,  0.86E+00,  0.1E+01, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I,  0.1E+01,  0.721E+00,  0.545E+00,  0.455E+00,  0.388E+00,
            0.35E+00,  0.328E+00,  0.328E+00,  0.365E+00,  0.504E+00,
            0.639E+00,  0.822E+00,  0.567E+00,  0.429E+00,  0.355E+00,
            0.309E+00,  0.288E+00,  0.284E+00,  0.3E+00,  0.364E+00,  0.486E+00,
            0.621E+00,  0.64E+00,  0.464E+00,  0.336E+00,  0.288E+00,
            0.265E+00,  0.257E+00,  0.258E+00,  0.275E+00,  0.345E+00,
            0.475E+00,  0.605E+00,  0.459E+00,  0.355E+00,  0.282E+00,
            0.261E+00,  0.259E+00,  0.261E+00,  0.254E+00,  0.257E+00,
            0.326E+00,  0.459E+00,  0.591E+00,  0.288E+00,  0.249E+00,
            0.228E+00,  0.273E+00,  0.245E+00,  0.254E+00,  0.246E+00,
            0.245E+00,  0.316E+00,  0.447E+00,  0.576E+00,  0.138E+00,
            0.131E+00,  0.136E+00,  0.149E+00,  0.165E+00,  0.182E+00,
            0.202E+00,  0.234E+00,  0.309E+00,  0.456E+00,  0.558E+00,
```

```
 0.12E-01, -0.46E-02, -0.8E-02, -0.6E-02,  0.6E-02,  0.34E-01,
 0.109E+00,  0.218E+00,  0.305E+00,  0.427E+00,  0.538E+00,
-0.101E+00, -0.147E+00, -0.178E+00, -0.193E+00, -0.166E+00,
-0.145E+00, -0.13E-01,  0.192E+00,  0.302E+00,  0.419E+00,
 0.519E+00, -0.213E+00, -0.278E+00, -0.325E+00, -0.347E+00,
-0.337E+00, -0.283E+00, -0.127E+00,  0.155E+00,  0.298E+00,
 0.412E+00,  0.506E+00, -0.336E+00, -0.387E+00, -0.423E+00,
-0.431E+00, -0.434E+00, -0.343E+00, -0.21E+00,  0.108E+00,
 0.291E+00,  0.406E+00,  0.5E+00, -0.466E+00, -0.482E+00, -0.487E+00,
-0.466E+00, -0.418E+00, -0.351E+00, -0.271E+00,  0.56E-01,
 0.264E+00,  0.4E+00,  0.497E+00, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,  0.122E+00,
 0.35E-01, -0.53E-01, -0.166E+00, -0.335E+00, -0.454E+00, -0.505E+00,
-0.498E+00, -0.275E+00, -0.75E-01,  0.534E+00,  0.86E-01,  0.77E-02,
-0.8E-02, -0.191E+00, -0.328E+00, -0.42E+00, -0.451E+00, -0.41E+00,
-0.292E+00,  0.66E-01,  0.567E+00,  0.49E-01, -0.25E-01, -0.108E+00,
-0.213E+00, -0.321E+00, -0.385E+00, -0.397E+00, -0.341E+00,
-0.211E+00,  0.198E+00,  0.636E+00,  0.16E-02, -0.64E-01, -0.136E+00,
-0.227E+00, -0.311E+00, -0.351E+00, -0.347E+00, -0.298E+00,
-0.129E+00,  0.312E+00,  0.68E+00, -0.46E-01, -0.103E+00, -0.161E+00,
-0.235E+00, -0.298E+00, -0.321E+00, -0.306E+00, -0.26E+00, -0.34E-01,
 0.403E+00,  0.719E+00, -0.84E-01, -0.132E+00, -0.179E+00,
-0.236E+00, -0.284E+00, -0.298E+00, -0.274E+00, -0.195E+00,  0.8E-01,
 0.474E+00,  0.757E+00, -0.105E+00, -0.146E+00, -0.189E+00,
-0.234E+00, -0.269E+00, -0.278E+00, -0.24E+00, -0.88E-01,  0.209E+00,
 0.535E+00,  0.798E+00, -0.114E+00, -0.151E+00, -0.194E+00,
-0.233E+00, -0.256E+00, -0.256E+00, -0.19E+00,  0.48E-01,  0.337E+00,
 0.595E+00,  0.843E+00, -0.118E+00, -0.154E+00, -0.198E+00,
-0.232E+00, -0.243E+00, -0.223E+00, -0.11E+00,  0.163E+00,
 0.446E+00,  0.659E+00,  0.894E+00, -0.127E+00, -0.165E+00,
-0.205E+00, -0.232E+00, -0.231E+00, -0.176E+00,  0.0,  0.295E+00,
 0.55E+00,  0.727E+00,  0.946E+00, -0.14E+00, -0.16E+00, -0.213E+00,
-0.232E+00, -0.219E+00, -0.126E+00,  0.124E+00,  0.388E+00,  0.6E+00,
 0.798E+00,  0.1E+01, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
```

```
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I,
CWSPST  = 0,
CWSPI   = 0,
CWCDI   = 0,
XTAB    = 0.0, 0.1E+00, 0.2E+00, 0.3E+00, 0.4E+00, 0.5E+00, 0.6E+00,
          0.7E+00, 0.8E+00, 0.9E+00, 0.1E+01, I, I, I, I, I, I, I, I, I,
CWPR2   = 0,
CWPR3   = 0,
CWPR4   = 0,
CWPR5   = 0,
CWPR6   = 0,
CWPR8   = 0,
CWPR8X  = 0,
XCBAR   = I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
XIC     = I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
NHP     = 11,
INTOR   = 2,
NPNCHQ  = 1,
NRQIJ   = 0,
QIJ     = ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0),
          ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0),
          ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0),
          ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0),
          ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0),
          ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0),
          ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0), ( 0.0, 0.0),
          ( 0.0, 0.0),
MACH    = 0.12E+01,
K       = 0.4E+00,
TANLE   = 0.1216E+01,
TANLTE  = 0.35255E+00,
S       = 0.81E+00,
NMODS   = 3,
NSYM    = 1,
$END
```

MACH= 1.20000000E+00   K= 4.00000000E-01   TANLE= 1.21600000E+00   TANLTE= 3.52550000E-01   S= 8.10000000E-01

NMODS= 3   NSYM= 1

INPUT DEFLECTIONS H SUB I, IN ORDER L.E. TO T.E., INBOARD FIRST, MODE 1 FIRST

-2.18000000E-01	-1.99000000E-01	-1.69000000E-01	-1.24000000E-01	-7.20000000E-02	-2.70000000E-02
4.60000000E-03	3.10000000E-02	6.70000000E-02	1.22000000E-01	1.88000000E-01	
-1.92000000E-01	-1.63000000E-01	-1.27000000E-01	-8.30000000E-02	-3.40000000E-02	9.00000000E-03
4.40000000E-02	7.70000000E-02	1.16000000E-01	1.69000000E-01	2.30000000E-01	
-1.58000000E-01	-1.19000000E-01	-7.90000000E-02	-3.70000000E-02	4.10000000E-03	4.40000000E-02
8.29000000E-02	1.23000000E-01	1.66000000E-01	2.18000000E-01	2.71000000E-01	
-1.10000000E-01	-5.57000000E-02	-7.30000000E-03	3.10000000E-02	6.20000000E-02	9.30000000E-02
1.31000000E-01	1.76000000E-01	2.25000000E-01	2.73000000E-01	3.20000000E-01	
-4.70000000E-02	-2.60000000E-03	4.00000000E-02	7.90000000E-02	1.17000000E-01	1.55000000E-01
1.96000000E-01	2.39000000E-01	2.85000000E-01	3.31000000E-01	3.76000000E-01	
3.60000000E-02	7.90000000E-02	1.20000000E-01	1.59000000E-01	1.96000000E-01	2.33000000E-01
2.73000000E-01	3.15000000E-01	3.58000000E-01	4.00000000E-01	4.42000000E-01	
1.37000000E-01	1.79000000E-01	2.18000000E-01	2.55000000E-01	2.90000000E-01	3.26000000E-01
3.65000000E-01	4.05000000E-01	4.46000000E-01	4.86000000E-01	5.25000000E-01	
2.52000000E-01	2.90000000E-01	3.27000000E-01	3.63000000E-01	3.98000000E-01	4.34000000E-01
4.71000000E-01	5.08000000E-01	5.46000000E-01	5.84000000E-01	6.21000000E-01	
3.85000000E-01	4.17000000E-01	4.51000000E-01	4.86000000E-01	5.21000000E-01	5.57000000E-01
5.92000000E-01	6.27000000E-01	6.62000000E-01	6.98000000E-01	7.35000000E-01	
5.25000000E-01	5.58000000E-01	5.91000000E-01	6.23000000E-01	6.55000000E-01	6.86000000E-01
7.16000000E-01	7.48000000E-01	7.83000000E-01	8.21000000E-01	8.60000000E-01	
6.75000000E-01	7.06000000E-01	7.37000000E-01	7.67000000E-01	7.96000000E-01	8.25000000E-01
8.54000000E-01	8.89000000E-01	9.25000000E-01	9.62000000E-01	1.00000000E+00	
1.00000000E+00	8.22000000E-01	6.40000000E-01	4.59000000E-01	2.88000000E-01	1.38000000E-01
1.20000000E-02	-1.01000000E-01	-2.13000000E-01	-3.36000000E-01	-4.60000000E-01	
7.21000000E-01	5.87000000E-01	4.64000000E-01	3.55000000E-01	2.49000000E-01	1,31000000E-01
-4.60000000E-03	-1.47000000E-01	-2.78000000E-01	-3.87000000E-01	-4.62000000E-01	
5.45000000E-01	4.29000000E-01	3.38000000E-01	2.82000000E-01	2.26000000E-01	1.36000000E-01
-8.00000000E-03	-1.78000000E-01	-3.25000000E-01	-4.23000000E-01	-4.87000000E-01	
4.55000000E-01	3.55000000E-01	2.68000000E-01	2.61000000E-01	2.33000000E-01	1.49000000E-01
-6.00000000E-03	-1.93000000E-01	-3.47000000E-01	-4.31000000E-01	-4.68000000E-01	
3.88000000E-01	3.09000000E-01	2.65000000E-01	2.59000000E-01	2.45000000E-01	1.65000000E-01
6.00000000E-03	-1.86000000E-01	-3.37000000E-01	-4.04000000E-01	-4.18000000E-01	
3.50000000E-01	2.88000000E-01	2.57000000E-01	2.61000000E-01	2.54000000E-01	1.82000000E-01
3.40000000E-02	-1.45000000E-01	-2.63000000E-01	-3.43000000E-01	-3.51000000E-01	
3.28000000E-01	2.84000000E-01	2.56000000E-01	2.54000000E-01	2.46000000E-01	2.02000000E-01
1.09000000E-01	-1.30000000E-02	-1.27000000E-01	-2.10000000E-01	-2.71000000E-01	
3.28000000E-01	3.00000000E-01	2.75000000E-01	2.57000000E-01	2.45000000E-01	2.34000000E-01
2.18000000E-01	1.92000000E-01	1.55000000E-01	1.08000000E-01	5.80000000E-02	
3.85000000E-01	3.64000000E-01	3.45000000E-01	3.28000000E-01	3.16000000E-01	3.09000000E-01
3.05000000E-01	3.02000000E-01	2.96000000E-01	2.91000000E-01	2.84000000E-01	
5.04000000E-01	4.88000000E-01	4.73000000E-01	4.59000000E-01	4.47000000E-01	4.36000000E-01
4.27000000E-01	4.19000000E-01	4.12000000E-01	4.06000000E-01	4.00000000E-01	
6.39000000E-01	6.21000000E-01	6.05000000E-01	5.91000000E-01	5.76000000E-01	5.58000000E-01
5.38000000E-01	5.19000000E-01	5.06000000E-01	5.00000000E-01	4.97000000E-01	
1.22000000E-01	8.80000000E-02	4.90000000E-02	1.60000000E-03	-4.60000000E-02	-8.40000000E-02
-1.05000000E-01	-1.14000000E-01	-1.18000000E-01	-1.27000000E-01	-1.40000000E-01	
3.50000000E-02	7.70000000E-03	-2.50000000E-02	-6.40000000E-02	-1.03000000E-01	-1.32000000E-01
-1.46000000E-01	-1.51000000E-01	-1.54000000E-01	-1.65000000E-01	-1.80000000E-01	
-5.30000000E-02	-8.00000000E-02	-1.08000000E-01	-1.36000000E-01	-1.61000000E-01	-1.79000000E-01
-1.89000000E-01	-1.94000000E-01	-1.98000000E-01	-2.05000000E-01	-2.13000000E-01	
-1.66000000E-01	-1.91000000E-01	-2.13000000E-01	-2.27000000E-01	-2.35000000E-01	-2.36000000E-01
-2.34000000E-01	-2.33000000E-01	-2.32000000E-01	-2.32000000E-01	-2.32000000E-01	
-3.35000000E-01	-3.28000000E-01	-3.21000000E-01	-3.11000000E-01	-2.98000000E-01	-2.84000000E-01
-2.69000000E-01	-2.56000000E-01	-2.43000000E-01	-2.31000000E-01	-2.19000000E-01	
-4.54000000E-01	-4.20000000E-01	-3.85000000E-01	-3.51000000E-01	-3.21000000E-01	-2.96000000E-01
-2.78000000E-01	-2.56000000E-01	-2.23000000E-01	-1.78000000E-01	-1.26000000E-01	
-5.05000000E-01	-4.51000000E-01	-3.97000000E-01	-3.47000000E-01	-3.06000000E-01	-2.74000000E-01
-2.40000000E-01	-1.90000000E-01	-1.10000000E-01	0.	1.24000000E-01	
-4.98000000E-01	-4.19000000E-01	-3.41000000E-01	-2.98000000E-01	-2.60000000E-01	-1.95000000E-01
-8.80000000E-02	4.80000000E-02	1.83000000E-01	2.95000000E-01	3.68000000E-01	
-3.75000000E-01	-2.92000000E-01	-2.11000000E-01	-1.29000000E-01	-3.40000000E-02	8.00000000E-02
2.09000000E-01	3.37000000E-01	4.40000000E-01	5.30000000E-01	6.00000000E-01	
-7.50000000E-02	6.60000000E-02	1.96000000E-01	3.12000000E-01	4.03000000E-01	4.74000000E-01
5.35000000E-01	5.95000000E-01	6.59000000E-01	7.27000000E-01	7.98000000E-01	
5.34000000E-01	5.87000000E-01	6.36000000E-01	6.80000000E-01	7.19000000E-01	7.57000000E-01
7.98000000E-01	8.43000000E-01	8.94000000E-01	9.46000000E-01	1.00000000E+00	

SPAN STATIONS

0.	8.10000000E-02	1.62000000E-01	2.43000000E-01	3.24000000E-01	4.05000000E-01
4.86000000E-01	5.67000000E-01	6.48000000E-01	7.29000000E-01	8.10000000E-01	

ANM MATRIX, BY MODAL COLUMNS OF COMPLEX ELEMENTS

REAL	IMAG	REAL	IMAG	REAL	IMAG
-7.60066633E-02	3.38066663E-02	-2.91361947E-01	-4.78644660E-02	8.81254618E-01	-9.99389482E-02
-9.80036329E-01	1.10799670E-01	1.76614927E-02	7.14078992E-03	-3.84011078E-01	4.91078733E-02
1.19476531E+00	-2.84021909E-03	-7.10933995E-01	-1.26622790E-01	1.36479886E-02	-1.32089001E-03
-4.57041111E-01	-3.92313979E-03	1.42233745E+00	4.55482322E-02	-1.49394605E+00	6.18161900E-02
1.34739180E-02	-9.03971905E-04	-2.37714097E-01	4.27151488E-04	6.50328748E-01	-3.93292997E-02
-4.96827474E-01	5.39720891E-02	2.51707632E-01	-1.14419510E-01	-6.56766466E-02	7.91441028E-02
-1.73364356E+00	-2.05058191E-01	2.51002594E+00	2.20425241E-02	1.13301686E-01	-5.61296618E-02
1.52159513E-01	2.30184671E-01	-2.51748657E+00	-2.53037340E-01	2.40699301E+00	4.34366000E-01
4.72066745E-02	-9.19805632E-03	2.84120094E+00	1.19103841E-01	-9.00114202E+00	-6.65319917E-01
6.70869275E+00	7.77326568E-01	-1.19401565E-02	4.84439411E-03	2.51891131E+00	-3.22874902E-02
-5.63147621E+00	-6.26792866E-01	2.74302964E+00	7.57484039E-01	5.52630590E-02	-2.43199945E-04
-1.53062741E-01	2.41267364E-01	-2.50266118E+00	-2.29390556E-01	3.78740023E+00	-2.63266509E-01
3.75061691E-02	-2.81757078E-02	-5.66663904E-02	7.26930562E-05	4.50921246E-01	2.89471049E-01
-1.19391256E+00	1.03055076E-01	-1.80245603E-02	5.87659177E-03	4.20232532E-01	1.53737130E-02
-5.18663634E+00	8.95242954E-02	8.13061014E+00	-4.14519773E-01	-1.47411662E-02	1.20601116E-03
1.93971364E-01	1.91249909E-03	-1.06140513E-01	2.40981052E-03	-5.06967179E+00	5.86136567E-01

MIJ MATRIX

3.57040000E-04	0.	0.	0.	5.17800000E-04	0.
0.	0.	2.65520000E-04			

MODAL FREQUENCIES

1.62500000E+02	3.91000000E+02	7.25000000E+02

MODAL DAMPING COEFFICIENTS G(I)

0.	0.	0.

WIJ MATRIX, BY COMPLEX ROWS

-2.92966970E-02	-8.25876865E-03	3.99667003E-02	-1.10165588E-02	-2.69961655E-02	-1.96619433E-03
-1.80472195E-02	1.62523872E-03	3.66207026E-02	-2.46528617E-02	-3.61717331E-03	-3.90196428E-03
6.06116883E-03	-1.20921728E-03	-1.30576177E-02	-1.03557966E-03	6.45197516E-03	-1.26046534E-02

ALPHA= 4.00000000E-04

FLUTTER DETERMINANT, BY COMPLEX ROWS

1.58220274E+01	-1.15198586E+00	5.57352163E+00	-1.53573828E+00	-3.76293463E+00	-2.76830639E-01
-2.99578590E-01	2.69784774E-02	4.04602461E+00	-4.12550258E-01	-6.00440225E-02	-6.47718015E-02
5.75019811E-02	-1.14717790E-02	-1.23878811E-01	-9.82448831E-03	1.08018343E+00	-1.21477055E-01

EIGENVECTORS IN SAME ORDER AS THE EIGENVALUES...EACH EIGENVECTOR NORMALIZED TO ITS LARGEST ELEMENT

EIGENVECTOR(1)

1.00000000E+00	0.	-2.56573256E-02	7.56934359E-04	4.19650134E-03	-4.92218400E-04

EIGENVECTOR(2)

-5.00845262E-01	9.93651128E-02	1.00000000E+00	0.	-4.81480535E-02	-4.58958959E-03

EIGENVECTOR(3)

2.34776516E-01	2.93443124E-02	4.17669477E-02	2.67807249E-02	1.00000000E+00	0.

EIGENVALUES. RE(1),IM(1),RE(2),IM(2),ETC.

1.56631461E+01	-1.10646652E+00	4.19996013E+00	-4.52435748E-01	1.06910924E+00	-1.26210907E-01

EIGENFREQUENCIES

1.63186627E+02	3.53933194E+02	6.94707975E+02

DAMPING COEFFICIENTS G

-7.06413966E-02	-1.07825598E-01	-1.15684524E-01

STIFFNESS PARAMETERS

1.58306771E+00	8.19363669E-01	4.17441568E-01

ALPHA= 6.00000000E-04

FLUTTER DETERMINANT, BY COMPLEX ROWS

1.37803784E+01	-1.72662878E+00	8.36028274E+00	-2.30360742E+00	-5.64440194E+00	-4.15245958E-01
-4.49367885E-01	4.04677160E-02	4.54997111E+00	-6.18825367E-01	-9.00660337E-02	-9.71577022E-02
8.62529717E-02	-1.72076685E-02	-1.85616216E-01	-1.47367325E-02	1.12027515E+00	-1.82215583E-01

EIGENVECTORS IN SAME ORDER AS THE EIGENVALUES...EACH EIGENVECTOR NORMALIZED TO ITS LARGEST ELEMENT

EIGENVECTOR(1)

1.00000000E+00	0.	-5.00808825E-02	-1.17183808E-03	7.68256572E-03	-4.02015384E-04

EIGENVECTOR(2)

1.00000000E+00	0.	-9.90097266E-01	-1.36976412E-01	7.06066075E-02	1.64087404E-02

EIGENVECTOR(3)

3.80340505E-01	6.34762253E-02	7.62127555E-02	4.44858039E-02	1.00000000E+00	0.

EIGENVALUES. RE(1),IM(1),RE(2),IM(2),ETC.

1.33143310E+01	-1.62206352E+00	4.79562690E+00	-7.12931495E-01	1.14066678E+00	-1.92674743E-01

EIGENFREQUENCIES

1.98691062E+02	3.31966557E+02	6.78826221E+02

DAMPING COEFFICIENTS G

-1.21828391E-01	-1.48662836E-01	-1.68914135E-01

STIFFNESS PARAMETERS

1.45955232E+00	8.75956793E-01	4.27208011E-01

Results for most alpha values are not shown

```
ALPHA= 9.0000000E-04

FLUTTER DETERMINANT, BY COMPLEX ROWS

 1.0717704E+01   -2.58954318E+00    1.25404241E+01   -3.45541113E+00   -6.46660291E+00   -6.22808937E-01
-6.74051627E-01   6.07015740E-02    4.00569085E+00   -9.26238081E-01   -1.35099051E-01   -1.45736553E-01
 1.29379456E-01  -2.58115028E-02   -2.78727324E-01   -2.21259067E-02    1.18341272E+00   -2.73323374E-01
  EIGENVECTORS IN SAME ORDER AS THE EIGENVALUES...EACH EIGENVECTOR NORMALIZED TO ITS LARGEST ELEMENT
EIGENVECTOR( 1)
  1.00000000E+00   0.               -1.40270650E-01   -6.66514885E-02    2.34594109E-02    7.07330668E-03
EIGENVECTOR( 2)
  1.00000000E+00   0.               -2.94752519E-01    6.31611614E-02    3.69475422E-02   -5.85556957E-03
EIGENVECTOR( 3)
  6.31594581E-01   1.56219954E-01    1.44426006E-01    8.53310572E-02    1.00000000E+00    0.

EIGENVALUES. RE(1),IM(1),RE(2),IM(2),ETC.

 8.48434425E+00  -2.99348796E+00    6.99157029E+00   -5.01674287E-01    1.22789395E+00   -2.96142385E-01

EIGENFREQUENCIES

 2.48902275E+02   2.74181545E+02    6.54273856E+02

DAMPING COEFFICIENTS G

-3.52824506E-01  -7.17786641E-02   -2.41179122E-01

STIFFNESS PARAMETERS

 1.16511591E+00   1.05769336E+00    4.43241506E-01


ALPHA= 1.00000000E-03

FLUTTER DETERMINANT, BY COMPLEX ROWS

 9.69708043E+00  -2.87771464E+00    1.39336046E+01   -3.65934570E+00   -9.40733657E+00   -6.92076597E-01
-7.48946474E-01   6.74461934E-02    4.95786410E+00   -1.03137565E+00   -1.50110056E-01   -1.61929504E-01
 1.43754953E-01  -2.86794475E-02   -3.09697020E-01   -2.45612208E-02    1.23045658E+00   -3.03692638E-01
  EIGENVECTORS IN SAME ORDER AS THE EIGENVALUES...EACH EIGENVECTOR NORMALIZED TO ITS LARGEST ELEMENT
EIGENVECTOR( 1)
  1.00000000E+00   0.               -1.14719834E-01   -1.35543599E-01    1.63071173E-02    1.48889205E-02
EIGENVECTOR( 2)
  1.00000000E+00   0.               -2.07655821E-01    1.81463194E-01    3.27905182E-02   -1.82852619E-02
EIGENVECTOR( 3)
  7.23047277E-01   2.75768809E-01    1.70988099E-01    1.04556444E-01    1.00000000E+00    0.

EIGENVALUES. RE(1),IM(1),RE(2),IM(2),ETC.

 7.41626101E+00  -4.47863857E+00    7.17921649E+00    5.97342589E-01    1.25990360E+00   -3.31486947E-01

EIGENFREQUENCIES

 2.66222596E+02   2.70582323E+02    6.45906044E+02

DAMPING COEFFICIENTS G

-6.03892762E-01   8.32044030E-02   -2.63105009E-01

STIFFNESS PARAMETERS

 1.08931399E+00   1.07176255E+00    4.48981710E-01
```

Note change of sign.

```
ALPHA= 3.00000000E-02

FLUTTER DETERMINANT, BY COMPLEX ROWS

 -1.07194096E+01     -8.63314392E+00      4.18014137E+01     -1.15160371E+01     -2.62220097E+01     -2.07622979E+00
 -2.24065942E+00      2.02338580E-01      7.59752905E+00     -3.09412694E+00     -4.50350108E-01     -4.85788911E-01
  4.31264059E-01     -8.60383426E-02     -9.29091079E-01     -7.36036623E-02      1.60137574E+00     -9.11077914E-01

    EIGENVECTORS IN SAME ORDER AS THE EIGENVALUES...EACH EIGENVECTOR NORMALIZED TO ITS LARGEST ELEMENT
EIGENVECTOR( 1)
   1.00000000E+00       0.                   9.41646284E-02      -7.66525330E-02     -1.16045695E-02      1.67625504E-02
EIGENVECTOR( 2)
   1.00000000E+00       0.                   2.63796367E-01       2.48078726E-01     -3.41657222E-02     -1.06012119E-01
EIGENVECTOR( 3)
   1.00000000E+00       0.                   3.91647385E-01       9.63148040E-02      1.38496012E-01     -2.95940422E-01

     EIGENVALUES. RE(1),IM(1),RE(2),IM(2),ETC.

 -7.29962210E+00     -1.34273430E+01      3.90923487E+00      1.76126911E+00      2.26966242E+00     -9.72274908E-01

     EIGENFREQUENCIES

         11111       3.66684139E+02      4.81233066E+02

     DAMPING COEFFICIENTS G

         11111       4.59549622E-01     -4.28374869E-01

     STIFFNESS PARAMETERS

         11111       7.90871404E-01      6.02616606E-01
```

```
12/18/73  LRC ICOPS SCHEDULR 6400Z-131K 10/08/73
13.13.07.GT03643.
13.13.07.                                 LRC COMPUTER COMPLEX
13.13.07.JOB,1,0100,064000,2000.                 A3541
13.13.07.                                    8648 R202
13.13.07.USER.CUNNINGHAM, HERBERT
13.13.07.
13.13.07.LINECNT(10000)
13.13.11.FETCH(A3541,SPPZ12,SOURCE)
13.13.17.TIME          BG ATTACH
13.13.33.TIME          ED ATTACH
13.13.33.END     FETCH
13.13.33.RUN(S,,,SCFILE)
13.15.11.SETINDF.
13.15.13.LGO.
13.30.25.STOP 71
13.30.25.RFL(10000)
13.30.26.MEMORY  010000 CM
13.30.26.RFL        0000433 O/S CALLS
13.30.26.RFL CPU   156.644663 SEC.
13.30.26.REWIND(INPUT)
13.30.27.COPYSBF.
13.30.29.REWIND(INPUT)
13.30.30.COPYSBF.
13.30.32.REWIND(OUTPUT)
13.30.33.SPPRINT(OUTPUT,3)
13.30.34.      0000461 O/S CALLS
13.30.34.CPU    156.759572 SEC.
13.30.34.PPU     74.137600 SEC.
13.30.35.THIS JOB HAS---PUNCH---OUTPUT.
13.30.35.COST OF THIS JOB WAS $    9
13.30.35.KWH.     2.16      KILOWORD HOURS
14.14.42.GT03643.    4662 LINES PRINTED. LP24
```

Langley Research Center,

National Aeronautics and Space Administration,

Hampton, Va., December 28, 1973.

APPENDIX

EVALUATION OF THE SUPERSONIC KERNEL FUNCTION

Based on equation (16c) of reference 3, the kernel function can be expressed in a separated form

$$K(M,k,x_0,y_0) = \frac{2\overline{K}(M,k,x_0,y_0)}{(2b)^2 y_0{}^2 V} \tag{A1}$$

where $x_0 = x - \xi$, $y_0 = y - \eta$, and the reduced kernel $\overline{K}$ is a dimensionless part of K.

For the steady case,

$$\overline{K}(M,0,x_0,y_0) = \frac{-x_0}{\sqrt{x_0{}^2 - \beta^2 y_0{}^2}} \tag{A2}$$

For $y_0 = 0$ (that is, $\eta = y$)

$$\overline{K}(M,k,x_0,0) = -e^{-i2kx_0} \tag{A3}$$

The general expression for $\eta \neq y$ is

$$\overline{K} = -e^{-i2kx_0}\left\{\frac{x_0 e^{-i2kx_0/\beta^2}}{\sqrt{x_0{}^2 - \beta^2 y_0{}^2}} \cos\left[\frac{2kM}{\beta^2}\left(x_0{}^2 - \beta^2 y_0{}^2\right)^{1/2}\right] + ik\left|y_0\right| \int_{\tau_l}^{\tau_u} \frac{\tau}{\sqrt{1 + \tau^2}} e^{-i2k\left|y_0\right|\tau} d\tau\right\} \tag{A4}$$

where the upper limit $\tau_u = \tau_1 + \tau_2$ and lower limit $\tau_l = \tau_1 - \tau_2$, and

$$\left.\begin{aligned} \tau_1 &= \frac{x_0}{\beta^2\left|y_0\right|} \\ \tau_2 &= \frac{M\sqrt{x_0{}^2 - \beta^2 y_0{}^2}}{\beta^2\left|y_0\right|} \end{aligned}\right\} \tag{A5}$$

In the integrand in equation (A4) the quantity $\tau/\sqrt{1 + \tau^2}$ is closely approximated for positive values of τ as in reference 4

$$\frac{\tau}{\sqrt{1 + \tau^2}} \approx 1 + N_1 e^{a\tau} + N_2 e^{b\tau} + N_3 e^{c\tau} \sin \pi\tau \tag{A6}$$

where $N_1 = -0.101$; $N_2 = -0.899$; $N_3 = -0.0932307$; $a = -0.329$; $b = -1.4067$; $c = -2.90$; and only N_3 varies slightly from the corresponding constant in reference 4. With this approximation substituted in equation (A4), the integral can be evaluated in closed form.

The upper limit τ_u is always positive. The lower limit is sometimes positive and sometimes negative. Since the approximation of equation (A6) holds only for positive τ, negative τ_l is accounted for in the following discussion.

Let the integral be defined

$$Q_1 \equiv \int_{\tau_l}^{\tau_u} \frac{\tau}{\sqrt{1 + \tau^2}} e^{-i2k\left|y_0\right|\tau} \, d\tau \tag{A7}$$

where Q_1 is complex, $Q_1 = \mathrm{Re}(Q_1) + i\mathrm{Im}(Q_1)$. Since there are four terms on the right-hand side of equation (A6),

$$Q_1 \approx Q_2 + Q_3 + Q_4 + Q_5 \tag{A8}$$

and each Q_n is complex as follows:

$$\begin{aligned} Q_2 &\equiv \int_{\tau_l}^{\tau_u} \left(\cos 2k\left|y_0\right|\tau - i \sin 2k\left|y_0\right|\tau\right) d\tau \\ &= \frac{1}{2k\left|y_0\right|}\left[\sin 2k\left|y_0\right|\tau_u - \sin 2k\left|y_0\right|\left|\tau_l\right| \right. \\ &\quad \left. + i\left(\cos 2k\left|y_0\right|\tau_u \mp \cos\left|y_0\right|\left|\tau_l\right| - \begin{Bmatrix} 0 \\ 2 \end{Bmatrix}\right)\right] \end{aligned} \tag{A9}$$

where upper of the two signs $\mp$ and the upper of the two quantities $\begin{Bmatrix} 0 \\ 2 \end{Bmatrix}$ are used for positive τ_l, and the lower of the two are used for negative τ_l.

$$
\begin{aligned}
Q_3 &\equiv N_1 \int_{\tau_l}^{\tau_u} e^{a\tau}\left(\cos 2k|y_0|\tau - i \sin 2k|y_0|\tau\right) d\tau \\
&= N_4\left\{\left[e^{a\tau_u}\left(a \cos 2k|y_0|\tau_u + 2k|y_0|\sin 2k|y_0|\tau_u\right)\right.\right. \\
&\quad \left. - e^{a|\tau_l|}\left(a \cos 2k|y_0||\tau_l| + 2k|y_0|\sin 2k|y_0||\tau_l|\right)\right] \\
&\quad + i\left[e^{a\tau_u}\left(-a \sin 2k|y_0|\tau_u + 2k|y_0|\cos 2k|y_0|\tau_u\right)\right. \\
&\quad \left.\left. \pm e^{a|\tau_l|}\left(a \sin 2k|y_0||\tau_l| - 2k|y_0|\cos 2k|y_0||\tau_l|\right) - \begin{Bmatrix} 0 \\ 4k|y_0| \end{Bmatrix}\right]\right\}
\end{aligned}
\tag{A10}
$$

where

$$
N_4 \equiv \frac{N_1}{a^2 + \left(2k|y_0|\right)^2}
$$

$$
Q_4 \equiv N_2 \int_{\tau_l}^{\tau_u} e^{b\tau}\left(\cos 2k|y_0|\tau - i \sin 2k|y_0|\tau\right) d\tau \tag{A11}
$$

is obtained from the right-hand side of equation (A10) by replacing N_1 by N_2 and a by b.

$$
\begin{aligned}
Q_5 &\equiv N_3 \int_{\tau_l}^{\tau_u} e^{c\tau} \sin \pi\tau\left(\cos 2k|y_0|\tau - i \sin 2k|y_0|\tau\right) d\tau \\
&= N_8\left\{e^{c\tau_u}\left[c \sin N_6\tau_u - N_6 \cos N_6\tau_u\right] - e^{c|\tau_l|}\left[c \sin N_6|\tau_l| - N_6 \cos N_6|\tau_l|\right]\right\} \\
&\quad + N_9\left\{e^{c\tau_u}\left[c \sin N_7\tau_u - N_7 \cos N_7\tau_u\right] - e^{c|\tau_l|}\left[c \sin N_7|\tau_l| - N_7 \cos N_7|\tau_l|\right]\right\} \\
&\quad + i\left(N_8\left\{e^{c\tau_u}\left[c \cos N_6\tau_u + N_6 \sin N_6\tau_u\right] \mp e^{c|\tau_l|}\left[c \cos N_6|\tau_l| + N_6 \sin N_6|\tau_l|\right] - \begin{Bmatrix} 0 \\ 2c \end{Bmatrix}\right.\right. \\
&\quad \left.\left. - N_9\left\{e^{c\tau_u}\left[c \cos N_7\tau_u + N_7 \sin N_7\tau_u\right] \mp e^{c|\tau_l|}\left[c \cos N_7|\tau_l| + N_7 \sin N_7|\tau_l|\right]\right\} - \begin{Bmatrix} 0 \\ 2c \end{Bmatrix}\right\}\right)
\end{aligned}
\tag{A12}
$$

where

$$N_6 = \pi + 2k|y_0| \qquad N_7 = \pi - 2k|y_0|$$

$$N_8 = \frac{N_3}{2\left(c^2 + N_6^{\,2}\right)} \qquad N_9 = \frac{N_3}{2\left(c^2 + N_7^{\,2}\right)}$$

and where the upper or lower of two signs and two quantities are used as described with equation (A9).

In the subprogram KERNEL, the number of calculations of sines and cosines has been minimized to only four of each with simple arguments, and the sines and cosines of the sum and difference arguments in Q_5 are obtained as their "multiply-adds."

REFERENCES

1. Cunningham, Herbert J.: Improved Numerical Procedure for Harmonically Deforming Lifting Surfaces From the Supersonic Kernel Function Method. AIAA J., vol. 4, no. 11, Nov. 1966, pp. 1961-1968.

2. Cunningham, Herbert J.: Application of a Supersonic Kernel-Function Procedure to Flutter Analysis of Thin Lifting Surfaces. NASA TN D-6012, 1970.

3. Watkins, Charles E.; and Berman, Julian H.: On the Kernel Function of the Integral Equation Relating Lift and Downwash Distributions of Oscillating Wings in Supersonic Flow. NACA Rep. 1257, 1956. (Supersedes NACA TN 3438.)

4. Watkins, Charles E.; Woolston, Donald S.; and Cunningham, Herbert J.: A Systematic Kernel Function Procedure for Determining Aerodynamic Forces on Oscillating or Steady Finite Wings at Subsonic Speeds. NASA TR R-48, 1959.

NASA-Langley, 1974 L-9126