AD NUMBER

AD872191

NEW LIMITATION CHANGE

TO

Approved for public release, distribution unlimited

FROM

Distribution authorized to U.S. Gov't. agencies and their contractors; Critical Technology; JAN 1970. Other requests shall be referred to Naval Air Systems Command, ATTN: AIR-530214, Washington, DC 20360.

AUTHORITY

USNASC ltr dtd 26 Oct 1971

MSD-P70-192

AD872191

AD No. ____

DDC FILE COPY

CONTRACT NO. 00019-69-C-0427

COLLOCATION FLUTTER ANALYSIS STUDY II

THIS DOCUMENT IS SUBJECT TO SPECIAL EXPORT CONTROLS AND TRANSMITTAL TO FOREIGN GOVERNMENTS OR FOREIGN NATIONALS MAY BE MADE ONLY WITH PRIOR APPROVAL OF THE NAVAL AIR SYSTEMS COMMAND (AIR-530214).

VOLUME III

STRUCTURAL ANALYSIS PROGRAM FLUENC-100

COMPONENT MODE SYNTHESIS PROGRAM - COMSYN

AND

MODAL FLUTTER ANALYSIS PROGRAM

APRIL 1970

MISSILE SYSTEMS DIVISION

HUGHES

HUGHES AIRCRAFT COMPANY

COFA II

COLLOCATION FLUTTER ANALYSIS STUDY II

VOLUME III

STRUCTURAL ANALYSIS PROGRAM ∿ FLUENC-100C

COMPONENT MODE SYNTHESIS PROGRAM ∿ COMSYN

and

MODAL FLUTTER ANALYSIS PROGRAM

PREPARED BY DYNAMICS & ENVIRONMENTS SECTION PERSONNEL, HUGHES AIRCRAFT COMPANY, MISSILE SYSTEMS DIVISION, CONTRACT NO. 00019-69-C-0427

JANUARY 1970

TABLE OF CONTENTS

1.0 INTRODUCTION

In order to determine the flutter characteristics of an aerodynamic surface, it is necessary to know the unsteady aerodynamic forces, the elastic properties, and the mass distribution of the structure. This volume contains a set of three programs that calculate the mass and stiffness distributions and the flutter speeds. The programs are FLUENC-100C, COMSYN, and MOFA.

FLUENC-100C is a structural analysis program that uses the direct stiffness method to generate stiffness, flexibility and mass matrices, and then to perform vibration analyses. In addition, there is an option to generate special structural parameters for use in the program COMSYN. FLUENC-100C is essentially the same as FLUENC except that the capability has been expanded to analyze lumped parameter systems that contain up to 200 nodes of which up to 100 may be free. The program COMSYN uses the component mode synthesis technique to analyze large structures. The structure is divided into component parts; the component modes and frequencies are obtained from FLUENC-100C and then entered into COMSYN where the analysis for the combined structure is performed. In addition, COMSYN calculates generalized aerodynamic forces and generalized masses for use in MOFA, the Modal Flutter Analysis Program. MOFA accepts data from FLUENC/FLUENC-100C and COMSYN and/or comparable data and performs flutter analyses using the normal mode (modal) method.

Both FLUENC-100C and COMSYN are tailored for use in flutter analyses. As such, only a capability to analyze planar structures is presented. The analysis includes displacements normal to the plane of the structure and approximates the two orthogonal rotations in the plane of the structure.

The FLUENC-100C and COMSYN programs have not been completely checked out for eigenvalue problems requiring more than fifty degrees of freedom. When exceeding fifty degrees of freedom, the user should check results carefully and if errors are suspected, the program should be rewritten in double precision for problems requiring degrees of freedom exceeding fifty. It should also be noted that the triangular plate elements are directional dependent which may cause small differences in deflections in symmetrical vibration modes. This problem can be eliminated by using a more refined and more complicated triangular plate element.

2.0 FLUENC-100C STRUCTURAL ANALYSIS PROGRAM

2.1 Theoretical Derivation

The theoretical derivation of the formulation for the FLUENC-100C Program is identical to that of the FLUENC Program presented in Volume II of Collocation Flutter Analysis Study, Reference 1; therefore, no new presentation of the derivation will be presented here. The additional capabilities added to the FLUENC-100C Program involved only computing changes associated with increasing the size of the program, and collecting certain structural data for use in COMSYN.

2.2 Program Description

The purpose of the computer program FLUENC-100C is twofold; namely, to provide structural influence coefficients and mass matrices for use in the Collocation Flutter Program and to provide stiffness and mass matrices, mode shapes, and frequencies for use in the Component Mode Synthesis Program. When using FLUENC-100C, a decision must be made whether to analyze a structure as one complete unit or to divide the structure into several components. When this is done, the analysis of the structure or the component is handled in essentially the same manner. Only special attention must be given to nodes that are common to two or more components when the structure is divided. Briefly, the program which is written in FORTRAN IV performs a structural analysis by the direct stiffness method. The structure is assumed to be representable by a planar network of beams and triangular plate elements connected at discrete joints. At each joint, if there are no restraints, the program assumes three degrees of freedom; that is, one displacement normal to the plane of the structure and two rotations. The program first synthesizes the stiffness and mass matrices for the entire structure, including all degrees of freedom from the data input for the beam and triangular plate elements and from the restraint information input for the joints. It then reduces the stiffness and mass matrices by solving for the rotational degrees of freedom in terms of the normal displacements by using static deflection relationships. As a final step, the program inverts the reduced stiffness matrix to obtain the influence coefficients. The dynamical matrix is then set up and a vibration analysis is performed.

If the option to generate data for COMSYN is used, a node or set of nodes are designated as common joints. Common joints are those structural points which exist on more than one component. The analysis is performed first with the common joints (junction nodes) restrained which yields the K_{FF} and M_{FF} matrices and an eigenvalue solution. Then new mass and stiffness matrices are generated based on an analysis with all common joints free. It is from these matrices that K_{FJ}, K_{JJ}, M_{FJ} are obtained. It is to be noted that in the second analysis, the rotational degrees of freedom for the junction nodes remain in order to insure slope compatibility, while those associated with the free joints (other than common joints) are reduced out of the system.

The FLUENC-100C Program allows a maximum of 200 nodal points in a structural idealization; of these 200 nodal points up to 100 nodal points may be allowed translational freedom. The remaining nodal points must be constrained in translation. Previously, the program FLUENC allowed a maximum of 50 points all of which could be free in translation; thus, the maximum size of the eigen value problem has increased; also, additional lattitude is allowed in developing structural idealizations. The above change creates the following program restriction: The number of joints minus the number of joints restrained in translation must equal or be less than one hundred (NJTS-NR $\leq$ 100)

Other features of the program include the option to input lumped masses or to compute the consistent mass matrices for the beam and triangular plate elements or both. The triangular plate elements may have either isotropic or orthotropic properties.

A series of problems was run to establish computer computation time. The results of this study are shown in Figure 2.2.1. The results are based upon the analysis of a flat plate using plate elements and the consistent mass matrix option. Five modes were requested for the analysis. As a result, the graph is only indicative of the computing time required, as the actual time not only depends on the number of degrees of freedom but also on the type of structural element, mass matrix, and number of modes requested.

2.2.1 Processing Information

A. Operation -- Standard FORTRAN IV processor system. Operable on the GE635 computer.

B. Core Storage -- The program FLUENC-100C requires a minimum of 54,000 memory units for execution.

C. Tape Units -- Standard input, output, and punch tape units, and 9 scratch tape units.

2.3 Description of Program Input

The following instructions describe the input data, their physical units, and their input FORTRAN format. The input quantities' names, in all capitals, are their FORTRAN names.

1.0 Title Card, format (12A6) two cards always

Column	1--- ---72
Name	Any alphanumeric statement

Column	1--- ---72
Name	Any alphanumeric statement

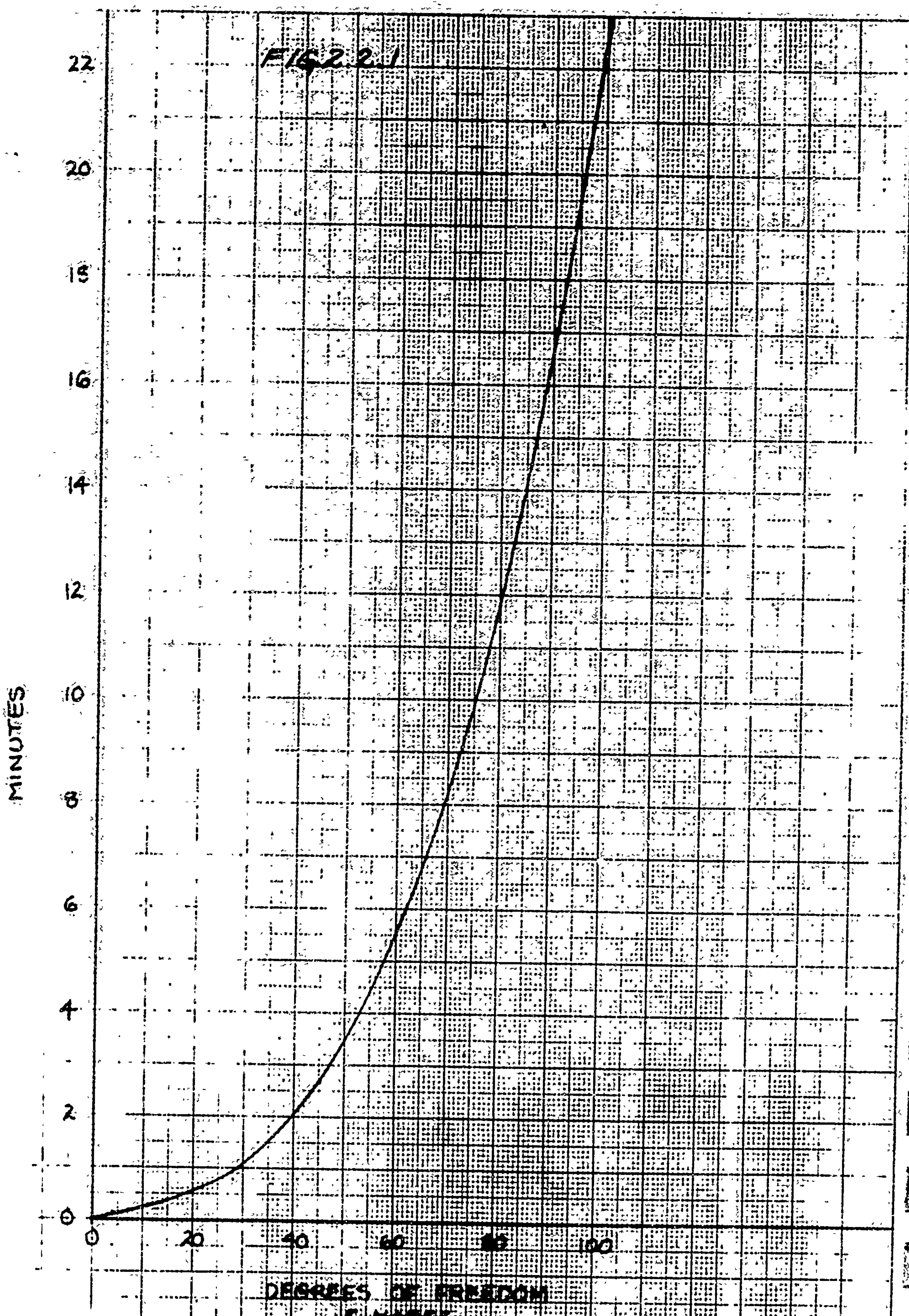


FIG 2.2.1
22
20
16
14
12
10
8
6
4
2
0
MINUTES
0
20
40
60
80
100
DEGREES OF FREEDOM
5 MODES

2.0 Problem Size and Control Information Format (16I5)

Column	1-5	6-10	11-15	16-20	21-25	26-30	31-35	36-40	41-45
Name	NJTS	NR	NBE	NPE	NMØDE	MKEY	NLUMP	NCJT	NPUNJ

NJTS = Number of joints in structure (200 maximum)

NR = Number of joints with one or more restraints

NBE = Number of beam elements in structure

NPE = Number of plate elements in structure

NMØDE = Number of eigenvalues and eigenvectors desired (9 maximum)

MKEY = 1. Do not compute consistent mass terms for beam and/or triangular plate elements

2. Compute consistent mass terms for beam and/or triangular plate elements

NLUMP = Number of lumped masses input. Only lumped masses corresponding to the normal displacement at each joint may be input.

NCJT = Number of common joints on the component (12 maximum)

= 0, if the complete structure is to be analyzed (no common joints involved).

NOTE: NPUNJ = 0 when NCJT>0

NPUNJ = -1 No Punched Output

= 0 Both Mass and Flexibility Matrices Punched Out

= 1 Only Mass Matrix and ILOW, IHIGH Code Punched Out

= 2 Only Flexibility Matrix Punched Out

3.0 Material Properties

(a) Number of Materials, format (I5)

Column	1 - 5
Name	NMAT

NMAT = number of materials for which properties are input (10 max.)

(b) Properties, format (4E10.3)

Input NMAT number of cards, one for each material.

Column	1 - 10	11 - 20	21 - 30	31 - 40
Name	YM (i)	PR (i)	GE (i)	DENS (i)

YM (i) = Young's modulus of elasticity divided by 10^6; psi

PR (i) = Poisson's ratio

GE (i) = modulus of rigidity; psi. If input as 0, it will be computed from the following formula:

$$GE\ (i) = \frac{YM\ (i)}{2\left[1 + PR\ (i)\right]}$$

DENS (i) = material density; lb/in^3. Not required if MKEY = 1

4.0 Joint Coordinate Cards, format (10X, 2E10.3)

Input NJTS number of cards, one for each joint. Also, the structure is assumed to lie in the x-y plane.

Column	1 - 10	11 - 20	21 - 30
Name	m	X(m)	Y(m)

m = joint number (must be input consecutively starting with 1). May be placed anywhere between columns 1 and 10

X(m) = x coordinate of joint m; inches

Y(m) = y coordinate of joint m; inches

NOTE: If NCJT> 0, the common joints must be numbered last.
Example: If NJTS = 10 and NCJT = 3
Then joints 8, 9 and 10 are the common joints.
When reference is made to these joints in the program COMSYN, common joint 1 should be joint 8, common joint 2 -joint 9, common joint 3- joint 10.

5.0 Joint Restraint Information, format (4I5)

Input NR number of cards, one for each joint with one or more restraints.

Column	1 - 5	6 - 10	11 - 15	16 - 20
Name	JT	M1	M2	M3

JT = number of joint having one or more restraints

M1 = 0 free in the z direction
= 1 fixed in the z direction

M2 = 0 free to rotate about the x axis

= 1 fixed about the x axis

M3 = 0 free to rotate about the y axis

= 1 fixed about the y axis

NOTE: If NCJT> 0 then M1=M2=M3=1 for all common joints.

6.0 <u>Lumped Masses</u>, format (I5, 5X, E10.3)

Input NLUMP number of cards, one for each lumped mass.

Column	1 - 5	6 - 10	11 - 20
Name	JMASS	blank	RSMASS

JMASS = number of joint for which lumped mass is input

RMASS = lumped mass, lb.

If more than one lumped mass is input for a particular joint, the program will sum the masses.

7.0 <u>Beam Element Properties</u>, format (3E10.3, 3I5)

Input NBE number of cards, one for each beam element.

Column	1 - 10	11 - 20	21 - 30	31 - 35	36 - 40	41 - 45
Name	AR	XI	YJ	MAT	JTNR	JTFR

AR = area of beam cross section, in^2

XI = moment of inertia of area, in^4

YJ = effective torsional moment of inertia, in^4

MAT = material code corresponding to one of the materials input under paragraph 4.1.3.

JTNR, JTFR = joint numbers at the ends of the beam element

8.0 <u>Triangular Plate Element Properties</u>, format (E10.3, 5I5)

Input NPE number of cards, one for each triangular plate element.

Column	1 - 10	11 - 15	16 - 20	21 - 25	26 - 30	31 - 35
Name	PTH	MAT	JT1	JT2	JT3	NDX

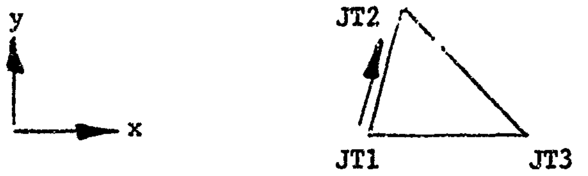
PTH = plate thickness, in.

MAT = material code corresponding to one of the materials input under Item 3,

JT1, JT2, JT3 = joint numbers at the three corners of the triangular plate

Restrictions:

a) The order of the joint numbers must be given in a clockwise manner as follows:



b) The angle formed by the edges of the triangular plate at JT1 must not be 90°.

c) The angle that the directed line defined by JT1 and JT2 makes with the global or system y-axis must be acute ($<90^{\circ}$).

NDX = 0 the plate has isotropic properties and the flexural rigidty terms are computed from

$$DX = DY = \frac{YM(MAT) \times PTH^3}{12\left\{1 - [PR(MAT)]^2\right\}}$$

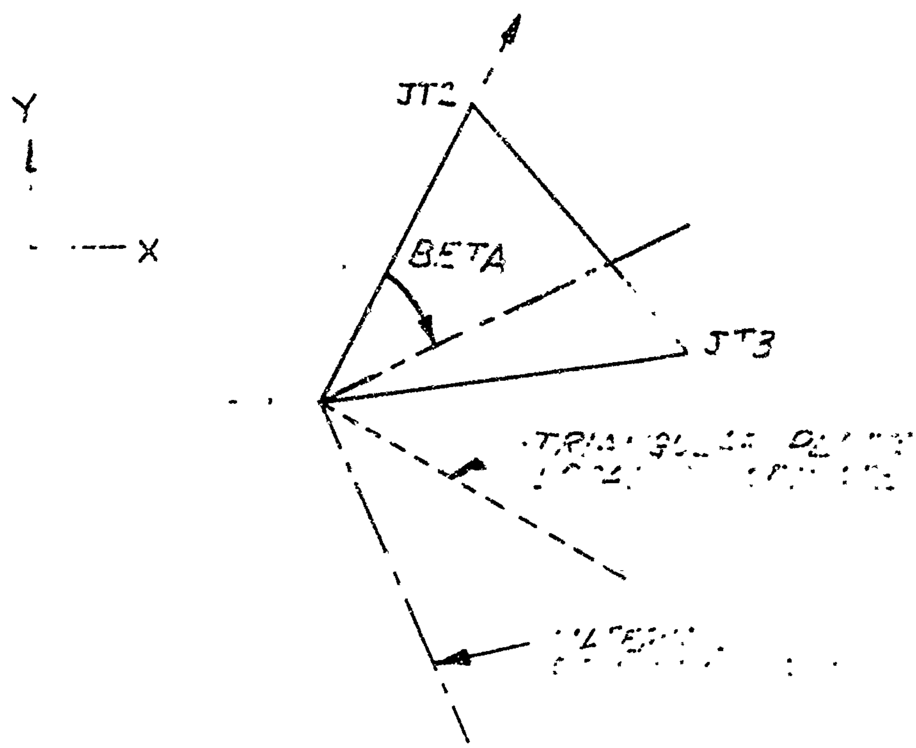
$$D1 = \left[PR(MAT)\right] \times DX$$

NDX = 1 the plate has orthotropic properties and the flexural rigidity terms are input by the next card [format (4E10.3)]

Column	1 - 10	11 - 20	21 - 30	31 - 40	41 - 50
Name	DX	DY	D1	DXY	BETA

DX, DY, D1, DXY = flexural rigidity terms, in.lb.

BETA = angle between material principal axes and the triangular plate local coordinates as shown below



All the components to be considered subsequently in the Component Mode Synthesis Analysis may be analyzed on one computer run of FLUENC-100C. Repeat the input requirements for each additional component.

2.4 Description of Program Output

I. Analysis of Complete Structure with No Common Joints Involved (NCJT=0)

A. Printed

1. Input Data

2. Coordinate numbers assigned by the program to the normal displacements at each unrestrained joint.

3. Results of the analysis
   a. Reduced stiffness matrix (lb./in.)
   b. Flexibility matrix (in./lb.)
   c. Reduced weight matrix (lb.)

(NOTE: Since the above matrices are symmetric, only the upper triangle is printed.)

   d. Eigenvalues and eigenvectors (normalized to the largest element) for each mode requested.
   e. Natural frequencies for each mode (CPS)

B. Punched - all matrices are punched in their entirety in Fortran Format (1P6E12.5). Each row starts on a new card. The cards for each matrix are sequenced and identified as follows:

	Matrix	I.D.
1.	Flexibility	FLEX
2.	Weight	WGHT

NOTE: The above punched output is compatible with the input required for COFA, the Collocation Flutter Program

II. Analysis of a Component with Common Joints (NCJT>0)

A. Printed Output

1. Common joints restrained. Same as indicated in I-A.

2. Common joints free.
   a. New list of coordinate numbers with the common joints added.
   b. CKFJ Matrix (Stiffness) and CMFJ Matrix (MASS) - relates common joints to free joints.
   c. Upper triangles of CKJJ Matrix (Stiffness) and CMJJ Matrix (MASS) - relates common joints to common joints.

B. Punched Output - Full Matrices are punched for all items

	Matrix	I.D.	*Theory Reference
1.	Common Joints Restrained		
	Stiffness	CKFF	$\bar{K}_{FF}$
	Flexibility	FLEX	$\bar{K}_{FF}{-1}$
	Weight (lbs.)	WGHT	386 ($\bar{M}_{FF}$)
	Frequencies (Cps)	FREQ	$\omega/2\pi$
	Mode Shapes (Eigenvectors)	MODE	ϕ
2.	Common Joints Free		
	CKFJ	CKFJ	$\bar{K}_{FJ}$
	CKJJ	CKJJ	$\bar{K}_{JJ}$
	CMFJ	CMFJ	$\bar{M}_{FJ}$
	CMJJ	CMJJ	$\bar{M}_{JJ}$

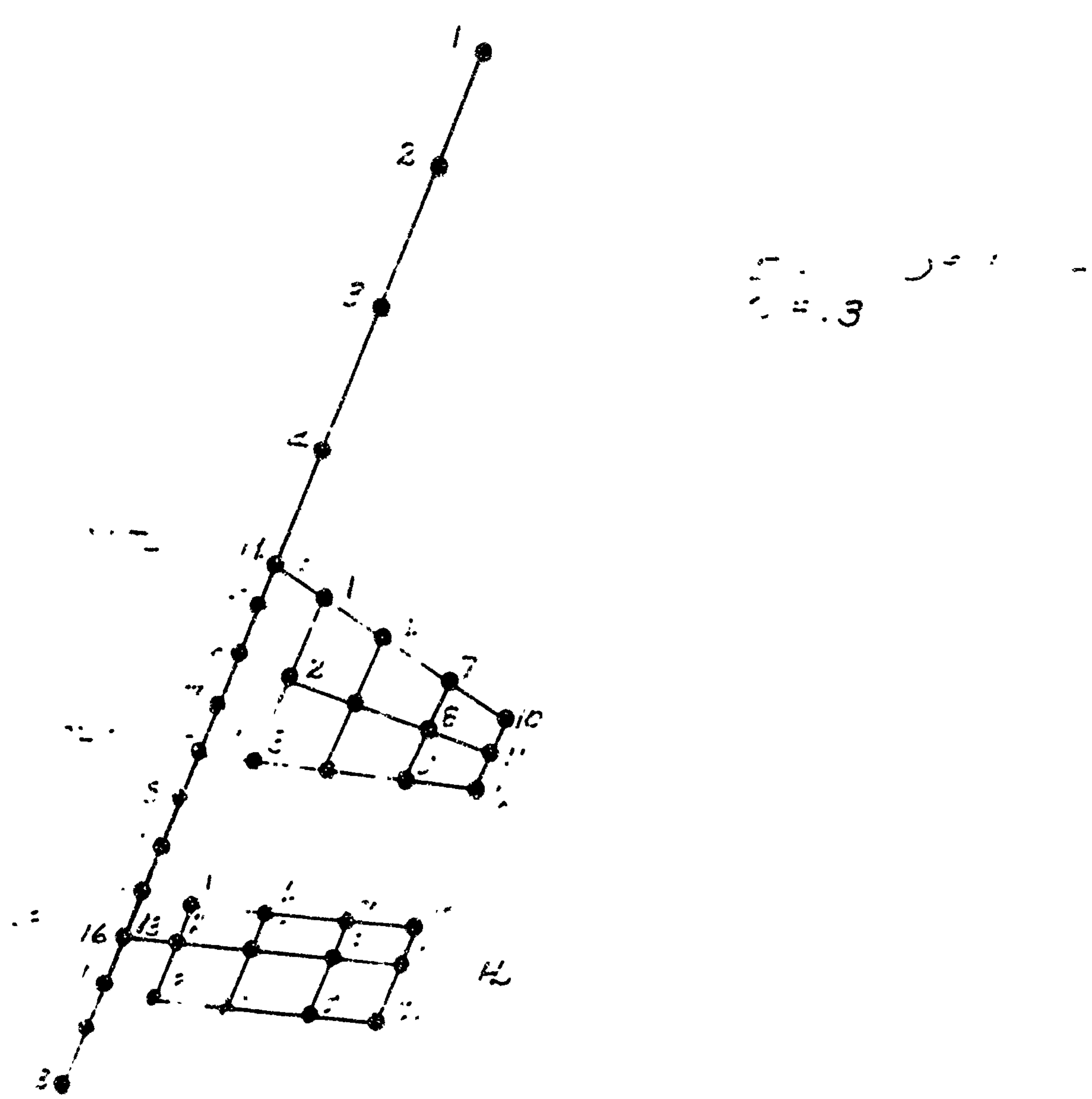
NOTE: The above punched output is compatible with the input required for the Component Mode Synthesis Program.

* Section 3.2, Volume III.

2.5 Sample Problems

The sample problems presented in Volume II of Reference 1 will demonstrate the operation of this program for the case when NCJT=0. To demonstrate the case for NCJT>0 a typical missile is analyzed.

The analysis will be performed for the missile being divided into three components the fuselage, the wing, and the control surface. There are three common joints; two attach the wing to the fuselage (Joints 1 and 2) and one attaches the control surface to the fuselage (Joint 3).



SAMPLE PROBLEM-TYPICAL MISSILE
COMPONENT 1 - FUSELAGE

NJTS = 16   NR = 16   NBE = 15   NPE = 0   NMODE = 7   MKEY = 1   NLUMP = 16   NCUT = 3   NPUNJ = 0

M A T E R I A L   P R O P E R T I E S

NO.	YOUNG*S MODULUS	POISSON RATIO	MODULUS OF RIGIDITY	DENSITY
1	0.10000E 08	0.30000	0.38462E 07	0.

J O I N T   C O O R D I N A T E S

JOINT NO.	X COORD.	Y COORD.
1	0.	0.
2	15.00000	0.
3	25.00000	0.
4	35.00000	0.
5	50.00000	0.
6	55.00000	0.
7	60.00000	0.
8	70.00000	0.
9	75.00000	0.
10	80.00000	0.
11	90.00000	0.
12	95.00000	0.
13	100.00000	0.
14	45.00000	0.
15	65.00000	0.
16	85.00000	0.

J O I N T   R E S T R A I N T   C O D E

JOINT NO.	Z DISPLACEMENT	ROTATION ABOUT X	ROTATION ABOUT Y
1	0	1	0
2	0	1	0
3	0	1	0
4	0	1	0
5	0	1	0
6	0	1	0
7	0	1	0
8	0	1	0
9	0	1	0
10	0	1	0
11	0	1	0
12	0	1	0
13	0	1	0
14	1	1	1
15	1	1	1
16	1	1	1

5	16.6000
6	16.7000
7	16.6000
8	25.0000
9	25.0000
10	25.0000
11	5.0000
12	5.0000
13	15.0000
14	49.5000
15	74.5000
16	74.5000

B E A M   E L E M E N T   P R O P E R T I E S

ELEMENT NO.	A	I	J	MAT	JOINT 1	JOINT 2
1	1.5000	35.0000	70.0000	1	1	2
2	1.5000	35.0000	70.0000	1	2	3
3	1.5000	35.0000	70.0000	1	3	4
4	3.0000	70.0000	140.0000	1	4	14
5	3.0000	70.0000	140.0000	1	14	5
6	3.0000	70.0000	140.0000	1	5	6
7	3.0000	70.0000	140.0000	1	6	7
8	3.0000	70.0000	140.0000	1	7	15
9	3.0000	70.0000	140.0000	1	15	8
10	3.0000	70.0000	140.0000	1	8	9
11	3.0000	70.0000	140.0000	1	9	10
12	3.0000	70.0000	140.0000	1	10	16
13	1.5000	35.0000	70.0000	1	16	11
14	1.5000	35.0000	70.0000	1	11	12
15	1.5000	35.0000	70.0000	1	12	13

COMMON JOINTS CONSTRAINED

COORDINATE NUMBERS FOR EACH Z DISPLACEMENT AT EACH UNRESTRAINED JOINT

JOINT NO.	COORD. NO.
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13

THE TIME ELAPSED FOR MATRIX INVERSION = 0.9844E-01 SECONDS

R E D U C E D   U P P E R   T R I A N G U L A R   S T I F F N E S S   M A T R I X

ROW 1

```
     0.19753E 06  -0.57531E 06   0.48689E 06  -0.15556E 06   0.          0.          0.          0.          0.
     0.            0.             0.             0.
 ROW  2
     0.22531E 07  -0.27849E 07   0.15556E 07   0.           0.          0.          0.          0.          0.
     0.            0.             0.
 ROW  3
     0.52000E 07  -0.49000E 07   0.            0.           0.          0.          0.          0.          0.
     0.            0.
 ROW  4
     0.91000E 07   0.             0.            0.           0.          0.          0.          0.          0.
     0.
 ROW  5
     0.10560E 09  -0.67200E 08   0.28800E 08   0.           0.          0.          0.          0.          0.
 ROW  6
     0.84000E 08  -0.67200E 08   0.            0.           0.          0.          0.          0.
 ROW  7
     0.10560E 09   0.             0.            0.           0.          0.          0.
 ROW  6
     0.10560E 09  -0.67200E 08   0.28800E 08   0.           0.          0.
 ROW  9
     0.84000E 08  -0.67200E 08   0.            0.           0.
 ROW 10
     0.10560E 09   0.             0.            0.
 ROW 11
     0.51692E 08  -0.29723E 08   0.77538E 07
 ROW 12
     0.28431E 08  -0.10338E 08
 ROW 13
     0.45231E 07
```

THE TIME ELAPSED FOR MATRIX INVERSION = 0.1018E 00 SECONDS

R E D U C E D   U P P E R   T R I A N G U L A R   F L E X I B I L I T Y   M A T R I X

```
 ROW  1
     0.63809E-04   0.30595E-04   0.13214E-04   0.29762E-05   0.          0.          0.          0.          0.
     0.            0.             0.             0.
 ROW  2
     0.16667E-04   0.78571E-05   0.19048E-05   0.           0.          0.          0.          0.          0.
     0.            0.             0.
 ROW  3
     0.42857E-05   0.11905E-05   0.            0.           0.          0.          0.          0.          0.
     0.            0.
 ROW  4
     0.47[illegible]E-06   0.   0.   0.   0.   0.   0.   0.   0.
```

```
   0.

ROW   5
   0.25112E-07   0.29742E-07   0.12991E-07   0.          0.          0.          0.          0.          0.
ROW   6
   0.59524E-07   0.29742E-07   0.            0.          0.          0.          0.          0.
ROW   7
   0.25112E-07   0.            0.            0.          0.          0.          0.
ROW   8
   0.25112E-07   0.29742E-07   0.12991E-07   0.          0.          0.
ROW   9
   0.59524E-07   0.29742E-07   0.            0.          0.
ROW  10
   0.25112E-07   0.            0.            0.
ROW  11
   0.11905E-05   0.29742E-06   0.47619E-06
ROW  12
   0.95238E-06   0.16667E-05
ROW  13
   0.32143E-05
-15-
R E D U C E D   U P P E R   T R I A N G U L A R   W E I G H T   M A T R I X
ROW   1
   0.25000E 02   0.            0.            0.          0.          0.          0.          0.          0.
   0.            0.            0.            0.
ROW   2
   0.25000E 02   0.            0.            0.          0.          0.          0.          0.          0.
   0.            0.            0.
ROW   3
   0.50000E 02   0.            0.            0.          0.          0.          0.          0.          0.
   0.            0.
ROW   4
   0.50000E 02   0.            0.            0.          0.          0.          0.          0.          0.
   0.
ROW   5
   0.16600E 02   0.            0.            0.          0.          0.          0.          0.          0.
ROW   6
   0.16700E 02   0.            0.            0.          0.          0.          0.          0.
ROW   7
   0.16600E 02   0.            0.            0.          0.          0.          0.
ROW   8
   0.25[illegible] 02   0.            0.            0.          0.          0.
ROW   9
```

```
  0.25000E 02   0.         0.         0.         0.
ROW  10
  0.25000E 02   0.         0.         0.
ROW  11
  0.50000E 01   0.         0.
ROW  12
  0.50000E 01   0.
ROW  13
  0.15000E 02
```

HERE ARE THE EIGENVALUES AND EIGENVECTORS.

```
EIGENVECTOR NUMBER     1
          CORRESPONDING TO  1.8092636E 05
  1.0000000E 00  4.9977673E-01  2.2487001E-01  5.2882105E-02  1.4042848E-21  2.6392101E-21
  1.5257985E-21  2.5619965E-21  4.7516382E-21  2.8063698E-21  9.7991295E-19  3.3882388E-18
  6.4729337E-18

EIGENVECTOR NUMBER     2
          CORRESPONDING TO  3.5120826E 06
 -9.1114038E-01  8.1743019E-01  1.0000000E 00  4.0531432E-01  3.2731321E-20  6.1486956E-20
  2.5493798E-20  6.1124787E-20  1.1477558E-19  6.6241056E-20  9.2478333E-17  3.1996522E-16
  6.1112638E-16

EIGENVECTOR NUMBER     3
          CORRESPONDING TO  7.2932065E 06
 -7.9416677E-15  2.9671028E-15  3.9747945E-15  1.6969911E-15  4.9113985E-19  9.2221834E-19
  5.3159581E-19  9.3145163E-19  1.7478525E-18  1.0065902E-18  1.5134908E-01  5.2360443E-01
  1.0000000E 00

EIGENVECTOR NUMBER     4
          CORRESPONDING TO  3.4296412E 07
 -3.0366978E-01  1.0000000E 00 -2.8470807E-01 -6.4352918E-01 -1.0661500E-18 -1.9951662E-18
 -1.1378397E-18 -2.3272531E-18 -4.3450350E-18 -2.4634330E-16 -3.7315757E-17 -1.3161968E-16
 -2.5258010E-16

EIGENVECTOR NUMBER     5
          CORRESPONDING TO  1.0997790E 08
 -1.4426186E-01  7.9128975E-01 -7.9372644E-01  1.0000000E 00 -4.5499143E-18 -8.4400637E-18
 -4.6924243E-18 -2.2887571E-17 -4.2227768E-17 -2.3177809E-17 -3.8069533E-18 -1.5004371E-17
 -3.1393151E-17

EIGENVECTOR NUMBER     6
          CORRESPONDING TO  1.6803353E 08
  1.8285851E-15 -1.0627352E-15 -3.4603975E-17  3.9675308E-15  2.3316629E-15  4.3022805E-15
  2.3591758E-15  5.4404375E-01  1.0000000E 00  5.4404374E-01  1.4249374E-16  8.9737328E-16
  2.3090458E-15

EIGENVECTOR NUMBER     7
          CORRESPONDING TO  2.5210840E 08
  7.5146910E-16 -8.7624328E-16  3.9354686E-16  9.0185434E-16  5.4386315E-01  1.0000000E 00
  5.4386310E-01  1.9544599E-15  3.6418880E-15  2.0280415E-15 -1.2788794E-16  1.0938936E-16
  1.0066665E-15
```

HERE ARE THE NATURAL FREQUENCIES

```
THE NATURAL FREQUENCY NUMBER   1  IS     67.697  CPS
THE NATURAL FREQUENCY NUMBER   2  IS    310.743  CPS
THE NATURAL FREQUENCY NUMBER   3  IS    429.813  CPS
THE NATURAL FREQUENCY NUMBER   4  IS    932.061  CPS
THE NATURAL FREQUENCY NUMBER   5  IS   1669.063  CPS
THE NATURAL FREQUENCY NUMBER   6  IS   2063.090  CPS
THE NATURAL FREQUENCY NUMBER   7  IS   2527.050  CPS
```

COMMON JOINTS FREE

COORDINATE NUMBERS FOR EACH Z DISPLACEMENT AT EACH UNRESTRAINED JOINT

JOINT NO.	COORD. NO.
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	15
16	16

THE TIME ELAPSED FOR MATRIX INVERSION = 0.1014E 00 SECONDS

CKFJ MATRIX (STIFFNESS) - RELATES COMMON JOINTS TO FREE JOINTS - FOR COMPONENT MODE SYNTHESIS PROGRAM

ROW 1	0.44444E 05	0.	0.	0.	0.	0.	0.14815E 06	0.	0.
ROW 2	-0.44444E 06	0.	0.	0.	0.	0.	-0.14815E 07	0.	0.
ROW 3	0.20000E 07	0.	0.	0.	0.	0.	0.66667E 07	0.	0.
ROW 4	-0.56000E 07	0.	0.	0.	0.	0.	-0.32667E 08	0.	0.
ROW 5	-0.50000E 08	-0.72000E 07	0.	0.	0.	0.	0.15600E 09	-0.12000E 08	0.
ROW 6	0.25200E 08	0.25200E 08	0.	0.	0.	0.	-0.42000E 08	0.42000E 08	0.
ROW 7	-0.72000E 07	-0.60000E 08	0.	0.	0.	0.	0.12000E 08	-0.15600E 09	0.
ROW 8	0.	-0.60000E 08	-0.72000E 07	0.	0.	0.	0.	0.15600E 09	-0.12000E 08
ROW 9	0.	0.25200E 08	0.25200E 08	0.	0.	0.	0.	-0.42000E 08	0.42000E 08
ROW 10	0.	-0.72000E 07	-0.60000E 08	0.	0.	0.	0.	0.12000E 08	-0.15600E 09
ROW 11	0.	0.	-0.29723E 08	0.	0.	0.	0.	0.	0.77538E 08

ROW 12

NOTICE TO USERS

Portions of this document have been judged by the Clearinghouse to be of poor reproduction quality and not fully legible. However, in an effort to make as much information as possible available to the public, the Clearinghouse sells this document with the understanding that if the user is not satisfied, the document may be returned for refund.

If you return this document, please include this notice together with the IBM order card (label) to:

Clearinghouse
Attn: 152.12
Springfield, Va. 22151

```
 0.          0.          0.11631E 08  0.          0.          -0.         0.          0.          -0.19385E 08
ROW  13
 0.          0.          -0.19385E 07  0.          0.          0.          0.          0.          0.32308E 07

UPPER TRIANGLE OF CKJJ MATRIX (STIFFNESS) - COMMON JOINTS - FOR COMPONENT MODE SYNTHESIS PROGRAM

ROW   1
 0.44200E 08  0.15000E 07  0.          0.          0.          0.          -0.95667E 08  0.30000E 07  0.
ROW   2
 0.30400E 08  0.18000E 07  0.          0.          0.          -0.30000E 07  -0.46770E-01  0.30000E 07
ROW   3
 0.60231E 08  0.          0.          0.          0.          -0.30000E 07  0.61615E 08
ROW   4
 0.16154E 09  0.          0.          0.          0.          0.
ROW   5
 0.21538E 09  0.          0.          0.          0.
ROW   6
 0.16154E 09  0.          0.          0.
ROW   7
 0.71611E 09  -0.50000E 07  0.
ROW   8
 0.97000E 09  -0.50000E 07
ROW   9
 0.72731E 09

CMFJ MATRIX (MASS) - RELATES COMMON JOINTS TO FREE JOINTS - FOR COMPONENT MODE SYNTHESIS PROGRAM

ROW   1
 0.          0.          0.          0.          0.          0.          0.          0.          0.
ROW   2
 0.          0.          0.          0.          0.          0.          0.          0.          0.
ROW   3
 0.          0.          0.          0.          0.          0.          0.          0.          0.
ROW   4
 0.          0.          0.          0.          0.          0.          0.          0.          0.
ROW   5
 0.          0.          0.          0.          0.          0.          0.          0.          0.
ROW   6
 0.          0.          0.          0.          0.          0.          0.          0.          0.
ROW   7
 0.          0.          0.          0.          0.          0.          0.          0.          0.
ROW   8
```

```
 0.          0.          0.          0.          0.          0.          0.          0.          0.
ROW   9
 0.          0.          0.          0.          0.          0.          0.          0.          0.
ROW  10
 0.          0.          0.          0.          0.          0.          0.          0.          0.
ROW  11
 0.          0.          0.          0.          0.          0.          0.          0.          0.
ROW  12
 0.          0.          0.          0.          0.          0.          0.          0.          0.
ROW  13
 0.          0.          0.          0.          0.          0.          0.          0.          0.

UPPER TRIANGLE OF CM.IJ MATRIX (MASS) - COMMON JOINTS - FOR COMPONENT MODE SYNTHESIS PROGRAM

ROW   1
 0.12821E 00 0.          0.          0.          0.          0.          0.          0.          0.
ROW   2
 0.19296E 00 0.          0.          0.          0.          0.          0.          0.
ROW   3
 0.19296E 00 0.          0.          0.          0.          0.          0.
ROW   4
 0.          0.          0.          0.          0.          0.
ROW   5
 0.          0.          0.          0.          0.
ROW   6
 0.          0.          0.          0.
ROW   7
 0.          0.          0.
ROW   8
 0.          0.
ROW   9
 0.
```

SAMPLE PROBLEM-TYPICAL MISSILE
COMPONENT 2 - WING

NJTS = 14   NR = 2   NBE = 19   NPE = 0   NNODE = 7   MKEY = 1   NLUMP = 14   NC-T = 2   NFUNJ = 0

M A T E R I A L   P R O P E R T I E S

NO.	YOUNG'S MODULUS	POISSON RATIO	MODULUS OF RIGIDITY	DENSITY
1	0.40000E 08	0.30000	0.15385E 08	0.

J O I N T   C O O R D I N A T E S

JOINT NO.	X COORD.	Y COORD.
1	46.05000	5.00000
2	55.62500	5.00000
3	65.20000	5.00000
4	48.13000	15.00000
5	56.87500	15.00000
6	65.62500	15.00000
7	50.20500	25.00000
8	58.12500	25.00000
9	66.04000	25.00000
10	52.29000	35.00000
11	59.37500	35.00000
12	66.45500	35.00000
13	45.00000	0.
14	65.00000	0.

-21-

J O I N T   R E S T R A I N T   C O D E

JOINT NO.	Z DISPLACEMENT	ROTATION ABOUT X	ROTATION ABOUT Y
13	1	1	1
14	1	1	1

L U M P E D   W E I G H T S

JOINT NO.	WEIGHT
1	0.3330
2	0.3330
3	0.3330
4	0.2350
5	0.2350
6	0.2350
7	0.2550
8	0.2550
9	0.2550
10	0.2200
11	0.2200
12	0.2200
13	0.5000
14	0.5000

B E A M   E L E M E N T   P R O P E R T I E S

ELEMENT NO.	A	I	J	MAT	JOINT 1	JOINT 2
1	[illegible]	0.2666	0.0266	1	13	1

2	0.5000	0.2660	0.0266	1	1	4
3	0.5000	0.2660	0.0266	1	4	7
4	0.5000	0.2660	0.0266	1	7	10
5	0.5000	0.2660	0.0266	1	2	5
6	0.5000	0.2660	0.0266	1	5	8
7	0.5000	0.2660	0.0266	1	8	11
8	0.5000	0.2660	0.0266	1	14	3
9	0.5000	0.2660	0.0266	1	3	6
10	0.5000	0.2660	0.0266	1	6	9
11	0.5000	0.2660	0.0266	1	9	12
12	0.5000	0.2660	0.0266	1	1	2
13	0.5000	0.2660	0.0266	1	2	3
14	0.5000	0.2660	0.0266	1	4	5
15	0.5000	0.2660	0.0266	1	5	6
16	0.5000	0.2660	0.0266	1	7	8
17	0.5000	0.2660	0.0266	1	8	9
18	0.5000	0.2660	0.0266	1	10	11
19	0.5000	0.2660	0.0266	1	11	12

COMMON JOINTS CONSTRAINED

COORDINATE NUMBERS FOR EACH Z DISPLACEMENT AT EACH UNRESTRAINED JOINT

JOINT NO.	COORD. NO.
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12

THE TIME ELAPSED FOR MATRIX INVERSION = 0.5024E 00 SECONDS

R E D U C E D   U P P E R   T R I A N G U L A R   S T I F F N E S S   M A T R I X

ROW 1
```
 0.71929E 06  -0.40075E 05   0.19132E 05  -0.20199E 06   0.83991E 03  -0.40317E 03   0.64964E 05   0.17700E 03   0.87777E 02
-0.10379E 05   0.49325E 02  -0.31280E 02
```

ROW 2
```
 0.96612E 05  -0.38832E 05   0.13353E 04  -0.41191E 05   0.16378E 04   0.17127E 03   0.25128E 05  -0.75251E 02  -0.22504E 02
-0.42037E 04   0.23843E 02
```

ROW 3
```
 0.76142E 06  -0.53977E 02  -0.40413E 03  -0.21403E 06  -0.53476E 02   0.69954E 03   0.68913E 05   0.18637E 02  -0.12320E 03
-0.11488E 05
```

ROW 4
```
 0.16437E 06  -0.51049E 05   0.24244E 05  -0.10107E 06   0.22992E 04  -0.34296E 03   0.26796E 05  -0.43715E 03   0.74566E 02
```

ROW 5

```
   0.20294E 06  -0.30666E 05   0.16150E 04  -0.91493E 05   0.20116E 04   0.25457E 02   0.25271E 05  -0.19754E 03

ROW   6
   0.19377E 06  -0.77737E 02   0.15971E 04  -0.10705E 06  -0.65444E 02   0.10472E 02   0.28272E 05

ROW   7
   0.13639E 06  -0.69129E 05   0.32645E 05  -0.39569E 05   0.34129E 04  -0.46724E 03

ROW   8
   0.23796E 06  -0.58975E 05   0.27417E 04  -0.43167E 05   0.29665E 04

ROW   9
   0.14224E 06  -0.10793E 03   0.25542E 04  -0.41321E 05

ROW  10
   0.54224E 05  -0.93278E 05   0.45294E 05

ROW  11
   0.20261E 06  -0.92802E 05

ROW  12
   0.64573E 05

 THE TIME ELAPSED FOR MATRIX INVERSION =  0.7583E-01 SECONDS


R E D U C E D   U P P E R   T R I A N G U L A R   F L E X I B I L I T Y   M A T R I X

ROW   1
   0.38826E-05   0.21241E-05   0.26275E-06   0.14362E-04   0.82639E-05   0.21415E-05   0.23472E-04   0.14374E-04  -0.52743E-05
   0.31827E-04   0.20443E-04   0.90737E-05

ROW   2
   0.13693E-04   0.19535E-05   0.78914E-05   0.94071E-05   0.72820E-05   0.10932E-04   0.10152E-04   0.10190E-04   0.13050E-04
   0.12609E-04   0.12276E-04

ROW   3
   0.36765E-05   0.21700E-05   0.79273E-05   0.13698E-04   0.53572E-05   0.13939E-04   0.22526E-04   0.92545E-05   0.19985E-04
   0.30722E-04

ROW   4
   0.92629E-04   0.55947E-04   0.17267E-04   0.17358E-03   0.10740E-03   0.41799E-04   0.24730E-03   0.15994E-03   0.71765E-04

ROW   5
   0.52720E-04   0.53777E-04   0.11329E-03   0.11291E-03   0.10932E-03   0.16686E-03   0.16596E-03   0.16364E-03

ROW   6
   0.86679E-04   0.41962E-04   0.10416E-03   0.16701E-03   0.72517E-04   0.15581E-03   0.23921E-03

ROW   7
   0.37969E-03   0.23759E-03   0.97457E-04   0.57656E-03   0.37121E-03   0.16459E-03

ROW   8
   0.23880E-03   0.23093E-03   0.37344E-03   0.36891E-03   0.36335E-03

ROW   9
   0.36670E-03   0.16574E-03   0.36309E-03   0.56148E-03

ROW  10
   0.95541E-03   0.61244E-03   0.27338E-03
```

```
ROW  11
   0.60934E-03   0.59981E-03

ROW  12
   0.93068E-03

REDUCED   UPPER   TRIANGULAR   WEIGHT   MATRIX

ROW   1
   0.33300E 00  0.        0.        0.        0.        0.        0.        0.        0.
   0.           0.        0.

ROW   2
   0.33300E 00  0.        0.        0.        0.        0.        0.        0.        0.
   0.           0.

ROW   3
   0.33300E 00  0.        0.        0.        0.        0.        0.        0.        0.
   0.

ROW   4
   0.28800E 00  0.        0.        0.        0.        0.        0.        0.        0.

ROW   5
   0.28800E 00  0.        0.        0.        0.        0.        0.        0.

ROW   6
   0.28800E 00  0.        0.        0.        0.        0.        0.

ROW   7
   0.25500E 00  0.        0.        0.        0.        0.

ROW   8
   0.25500E 00  0.        0.        0.        0.

ROW   9
   0.25500E 00  0.        0.        0.

ROW  10
   0.22000E 00  0.        0.

ROW  11
   0.22000E 00  0.

ROW  12
   0.22000E 00
```

```
HERE ARE THE EIGENVALUES AND EIGENVECTORS.

EIGENVECTOR NUMBER     1
              CORRESPONDING TO  6.2632084E 05
 3.6002336E-02  2.4659671E-02  3.4540911E-02  2.7338783E-01  2.8259907E-01  2.6247168E-01
 6.1943952E-01  6.0996623E-01  5.9617104E-01  1.0000000E 00  9.8182712E-01  9.6469735E-01

EIGENVECTOR NUMBER     2
              CORRESPONDING TO  1.6994201E 06
-7.7886679E-02 -7.3243090E-04  3.7986951E-02 -2.7866540E-01  7.8618528E-04  2.8181170E-01
-6.2072194E-01  5.4340216E-03  6.3263302E-01 -9.7104231E-01  1.3988816E-02  1.0000000E 00

EIGENVECTOR NUMBER     3
              CORRESPONDING TO  2.6766513E 07
 2.4163401E-01  5.2264473E-01  1.9927914E-01  9.9941319E-01  1.0000000E 00  8.1741462E-01
 6.2920426E-01  5.6710900E-01  5.1752926E-01 -8.5026886E-01 -7.6900623E-01 -6.8041854E-01

EIGENVECTOR NUMBER     4
              CORRESPONDING TO  3.5934503E 07
-1.9679316E-01  3.0623353E-02  2.2448287E-01 -8.8301271E-01  6.6308093E-02  1.0000000E 00
-5.7593627E-01  4.2627872E-02  5.5235967E-01  7.6300822E-01 -4.9530195E-02 -6.6664830E-01

EIGENVECTOR NUMBER     5
              CORRESPONDING TO  9.5298457E 07
 6.3768595E-02  1.0000000E 00  5.4463697E-02 -6.7735487E-02  7.4186013E-02 -7.4168263E-02
-2.1020006E-01 -2.3332149E-01 -1.9486479E-01  1.5229288E-01  1.4124142E-01  1.4325635E-01

EIGENVECTOR NUMBER     6
              CORRESPONDING TO  2.4897992E 08
 4.3259953E-01 -7.7543808E-02  1.3257104E-01  1.0000000E 00 -8.5281674E-01  5.1380778E-01
-2.7532252E-01 -9.3619429E-01  3.8302056E-01  2.7441027E-01  8.9590699E-02 -1.4611385E-02

EIGENVECTOR NUMBER     7
              CORRESPONDING TO  2.5916587E 08
-3.1968037E-01  7.6847805E-02  4.7207740E-01 -4.1742046E-01 -4.7010275E-01  7.5017075E-01
 1.0000000E 00 -2.9510236E-01 -6.0307387E-01 -3.4333023E-01 -3.6937620E-02  3.4933783E-01

HERE ARE THE NATURAL FREQUENCIES

THE NATURAL FREQUENCY NUMBER   1   IS     126.157   CPS
THE NATURAL FREQUENCY NUMBER   2   IS     207.478   CPS
THE NATURAL FREQUENCY NUMBER   3   IS     823.417   CPS
THE NATURAL FREQUENCY NUMBER   4   IS     954.061   CPS
THE NATURAL FREQUENCY NUMBER   5   IS    1553.665   CPS
THE NATURAL FREQUENCY NUMBER   6   IS    2511.321   CPS
THE NATURAL FREQUENCY NUMBER   7   IS    2562.275   CPS
```

COMMON JOINTS FREE

COORDINATE NUMBERS FOR EACH Z DISPLACEMENT AT EACH UNRESTRAINED JOINT

JOINT NO.	COORD. NO.
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14

THE TIME ELAPSED FOR MATRIX INVERSION = 0.5062E 00 SECONDS

CKFJ MATRIX (STIFFNESS) - RELATES COMMON JOINTS TO FREE JOINTS - FOR COMPONENT MODE SYNTHESIS PROGRAM

ROW 1
-0.55124E 06 0.75664E 02 -0.17192E 07 0.21142E 03 0.34779E 06 0.21282E 04

ROW 2
0.39965E 03 -0.90842E 03 0.32339E 04 -0.20121E 04 0.12088E 05 -0.12391E 05

ROW 3
-0.25990E 03 -0.58494E 06 -0.86425E 03 -0.18237E 07 -0.19617E 04 0.82749E 05

ROW 4
0.11587E 06 -0.40859E 02 0.19294E 06 -0.67104E 02 -0.41419E 05 0.27574E 02

ROW 5
0.79401E 03 0.15147E 04 0.13391E 04 0.25185E 04 -0.20309E 03 -0.25073E 03

ROW 6
0.11264E 03 0.12266E 06 0.18421E 03 0.20442E 06 -0.56176E 02 -0.82026E 04

ROW 7
-0.28825E 05 0.10144E 02 -0.48005E 05 0.16809E 02 0.10265E 05 -0.31253E 01

ROW 8
-0.30027E 03 -0.50185E 03 -0.49714E 03 -0.83612E 03 0.12088E 03 0.40965E 02

ROW 9
-0.34461E 02 -0.30540E 05 -0.57766E 02 -0.50699E 05 0.10648E 02 0.20663E 04

ROW 10
0.48268E 04 -0.63727E 01 0.80[illegible] 04 -0.10[illegible] 02 -0.17192E 04 0.[illegible] 00

ROW 11
0.15645E 02 0.67471E 02 0.25613E 02 0.14573E 03 -0.76734E 01 -0.72663E 01

ROW 12
0.12894E 02 0.50987E 04 0.21526E 02 0.84842E 04 -0.43339E 01 -0.34426E 03

15

UPPER TRIANGLE OF CKIJ MATRIX (STIFFNESS) - COMMON JOINTS - FOR COMPONENT MODE SYNTHESIS PROGRAM

```
ROW   1
 0.45865E 06  -0.27458E 02   0.15629E 07  -0.47025E 02  -0.32491E 06  -0.29727E 02
ROW   2
 0.48753E 06  -0.46647E 02   0.16617E 07   0.53972E 01  -0.65784E 05
ROW   3
 0.65985E 07  -0.31749E 02  -0.13637E 07  -0.97478E 02
ROW   4
 0.70155E 07   0.29548E 00  -0.27628E 06
ROW   5
 0.36360E 06  -0.21785E 03
ROW   6
 0.91084E 05
```

CMFJ MATRIX (MASS) - RELATES COMMON JOINTS TO FREE JOINTS - FOR COMPONENT MODE SYNTHESIS PROGRAM

```
ROW   1
 0.         0.         0.         0.         0.         0.
ROW   2
 0.         0.         0.         0.         0.         0.
ROW   3
 0.         0.         0.         0.         0.         0.
ROW   4
 0.         0.         0.         0.         0.         0.
ROW   5
 0.         0.         0.         0.         0.         0.
ROW   6
 0.         0.         0.         0.         0.         0.
ROW   7
 0.         0.         0.         0.         0.         0.
ROW   8
 0.         0.         0.         0.         0.         0.
ROW   9
 0.         0.         0.         0.         0.         0.
ROW  10
 0.         0.         0.         0.         0.         0.
ROW  11
 0.         0.         0.         0.         0.         0.
ROW  12
 0.         0.         0.         0.         0.         0.
```

UPPER TRIANGLE OF CMIJ MATRIX (MASS) - COMMON JOINTS - FOR COMPONENT MODE SYNTHESIS PROGRAM

```
ROW    1
 0.12950E-02  0.           0.           0.           0.           0.
ROW    2
 0.12950E-02  0.           0.           0.           0.
ROW    3
 0.           0.           0.           0.
ROW    4
 0.           0.           0.
ROW    5
 0.           0.
ROW    6
 0.
```

SAMPLE PROBLEM-TYPICAL MISSILE
COMPONENT 3-CONTROL SURFACE

NJTS = 13 NR = 1 NBE = 18 NPE = 0 NMODE = 7 MKEY = 1 NLUMP = 13 NCUT = 1 NFUNJ = 0

M A T E R I A L   P R O P E R T I E S ********************************

NO.	YOUNG*S MODULUS	POISSON RATIO	MODULUS OF RIGIDITY	DENSITY
1	0.10000E 08	0.30000	0.38462E 07	0.

J O I N T   C O O R D I N A T E S

JOINT NO.	X COORD.	Y COORD.
1	81.66000	5.00000
2	85.00000	5.00000
3	91.66000	5.00000
4	81.66000	15.00000
5	85.00000	15.00000
6	91.66000	15.00000
7	81.66000	25.00000
8	85.00000	25.00000
9	91.66000	25.00000
10	81.66000	35.00000
11	85.00000	35.00000
12	91.66000	35.00000
13	85.00000	0.

-29-

J O I N T   R E S T R A I N T   C O D E ************************

JOINT NO.	Z DISPLACEMENT	ROTATION ABOUT X	ROTATION ABOUT Y
13	1	1	1

L U M P E D   W E I G H T S

JOINT NO.	WEIGHT
1	0.0840
2	0.1250
3	0.0410
4	0.0840
5	0.1250
6	0.0410
7	0.0840
8	0.1250
9	0.0410
10	0.0840
11	0.1250
12	0.0410
13	0.5000

B E A M   E L E M E N T   P R O P E R T I E S ****************************

ELEMENT NO.	A	I	J	MAT	JOINT 1	JOINT 2
1	0.5000	0.2660	0.0266	1	1	4
2	0.5000	0.2660	0.0266	1	4	7
3	0.5000	0.2660	0.0266	1	7	10
4	0.5000	0.2660	0.0266	1	13	2

```
   5        0.5000     0.2660     0.0266   1    2    5
   6        0.5000     0.2660     0.0266   1    5    8
   7        0.5000     0.2660     0.0266   1    8   11
   8        0.5000     0.2660     0.0266   1    3    6
   9        0.5000     0.2660     0.0266   1    6    9
  10        0.5000     0.2660     0.0266   1    9   12
  11        0.5000     0.2660     0.0266   1    1    2
  12        0.5000     0.2660     0.0266   1    2    3
  13        0.5000     0.2660     0.0266   1    4    5
  14        0.5000     0.2660     0.0266   1    5    6
  15        0.5000     0.2660     0.0266   1    7    8
  16        0.5000     0.2660     0.0266   1    8    9
  17        0.5000     0.2660     0.0266   1   10   11
  18        0.5000     0.2660     0.0266   1   11   12


COMMON JOINTS CONSTRAINED


COORDINATE NUMBERS FOR EACH Z DISPLACEMENT AT EACH UNRESTRAINED JOINT
JOINT NO.      COORD. NO.
   1              1
   2              2
   3              3
   4              4
   5              5
   6              6
   7              7
   8              8
   9              9
  10             10
  11             11
  12             12

 THE TIME ELAPSED FOR MATRIX INVERSION =  0.5036E 00 SECONDS


R E D U C E D   U P P E R   T R I A N G U L A R   S T I F F N E S S   M A T R I X

ROW   1
  0.78891E 05  -0.10951E 06   0.35856E 05  -0.11926E 05   0.21643E 04  -0.18297E 03   0.64524E 04   0.26454E 02  -0.26695E 01
 -0.10752E 04  -0.46201E 01   0.41586E 00

ROW   2
  0.35143E 06  -0.54112E 05   0.11619E 04  -0.55858E 05   0.22863E 03   0.40259E 03   0.16999E 05   0.20535E 03  -0.69370E 02
 -0.28302E 04  -0.34561E 02

ROW   3
  0.23014E 05  -0.18359E 03   0.74683E 03  -0.10321E 05  -0.26792E 01   0.13519E 02   0.64202E 04   0.41428E 00  -0.20433E 01
 -0.10702E 04

ROW   4
  0.10155E 06  -0.11174E 06   0.36241E 05  -0.24542E 05   0.22487E 04  -0.17898E 03   0.64495E 04  -0.39952E 02  -0.41564E 01

ROW   5
  0.23689E 06  -0.55349E 05   0.22017E 04  -0.29371E 05   0.75897E 03  -0.31346E 02   0.71091E 04  -0.16015E 02

ROW   6
  0.24906E 05  -0.17898E 03   0.78295E 03  -0.23014E 05  -0.41541E 01  -0.20333E 02   0.64194E 04
```

```
ROW  7
 0.10154E 06  -0.11219E 06   0.36234E 05  -0.11916E 05   0.25108E 04  -0.17785E 03
ROW  8
 0.19431E 06  -0.55578E 05   0.25107E 04  -0.13170E 05   0.91639E 03
ROW  9
 0.44903E 05  -0.17785E 03   0.91647E 03  -0.10318E 05
ROW 10
 0.78076E 05  -0.10987E 06   0.36056E 05
ROW 11
 0.16892E 06  -0.54759E 05
ROW 12
 0.22960E 05

THE TIME ELAPSED FOR MATRIX INVERSION =  0.7592E-01 SECONDS

R E D U C E D   U P P E R   T R I A N G U L A R   F L E X I B I L I T Y   M A T R I X

ROW  1
 0.56546E-03   0.15663E-04  -0.10716E-02   0.60939E-03   0.62344E-04  -0.10280E-02   0.65434E-03   0.10728E-03  -0.98373E-03
 0.69933E-03   0.15203E-03  -0.93927E-03
ROW  2
 0.15664E-04   0.15646E-04   0.62655E-04   0.62656E-04   0.62658E-04   0.10965E-03   0.10965E-03   0.10965E-03   0.15664E-03
 0.15664E-03   0.15644E-03
ROW  3
 0.22195E-02  -0.10346E-02   0.61461E-04   0.22529E-02  -0.99763E-03   0.10039E-03   0.22892E-02  -0.95850E-03   0.13856E-03
 0.23262E-02
ROW  4
 0.10738E-02   0.40721E-03  -0.90075E-03   0.15152E-02   0.77511E-03  -0.70252E-03   0.19322E-02   0.11309E-02  -0.46779E-03
ROW  5
 0.40757E-03   0.41383E-03   0.77607E-03   0.77779E-03   0.77797E-03   0.11460E-02   0.11454E-02   0.11444E-02
ROW  6
 0.30536E-02  -0.66772E-03   0.81239E-03   0.37623E-02  -0.37688E-03   0.12196E-02   0.44052E-02
ROW  7
 0.24975E-02   0.15846E-02  -0.22738E-03   0.34606E-02   0.24022E-02   0.28968E-03
ROW  8
 0.15977E-02   0.16049E-02   0.24293E-02   0.24264E-02   0.24201E-02
ROW  9
 0.52567E-02   0.36715E-03   0.24679E-02   0.66623E-02
ROW 10
 0.51436E-02   0.[illegible]303E-02   0.12146E-02
ROW 11
 0.35287E-02   0.[illegible]6139E-02
ROW 12
```

```
0.90097E-02

R E D U C E D   U P P E R   T R I A N G U L A R   W E I G H T   M A T R I X

ROW   1
 0.84000E-01  0.          0.          0.          0.          0.          0.          0.          0.
 0.           0.          0.
ROW   2
 0.12500E 00  0.          0.          0.          0.          0.          0.          0.          0.
 0.           0.
ROW   3
 0.41000E-01  0.          0.          0.          0.          0.          0.          0.          0.
 0.
ROW   4
 0.84000E-01  0.          0.          0.          0.          0.          0.          0.          0.
ROW   5
 0.12500E 00  0.          0.          0.          0.          0.          0.          0.
ROW   6
 0.41000E-01  0.          0.          0.          0.          0.          0.
ROW   7
 0.84000E-01  0.          0.          0.          0.          0.
ROW   8
 0.12500E 00  0.          0.          0.          0.
ROW   9
 0.41000E-01  0.          0.          0.
ROW  10
 0.84000E-01  0.          0.
ROW  11
 0.12500E 00  0.
ROW  12
 0.41000E-01
```

```
HERE ARE THE EIGENVALUES AND EIGENVECTORS

EIGENVECTOR NUMBER     1
          CORRESPONDING TO  2.6854951E 05
  4.3763326E-02  4.274280[illegible]E-02  3.4448653E-02  3.0708152E-01  3.0692197E-01  3.1847429E-01
  6.3822790E-01  6.3984079E-01  6.4111643E-01  1.0000000E 00  9.9457150E-01  9.833[illegible]533E-01

EIGENVECTOR NUMBER     2
          CORRESPONDING TO  5.[illegible]391095E 05
 -2.7535481E-01  7.6666640E-05  5.[illegible]87013E-01 -3.5570724E-01  8.3232501E-04  7.1198869E-01
 -4.2792436E-01  3.014926[illegible]E-03  [illegible]4216032E-01 -4.8971924E-01  7.327231[illegible]E-03  1.0000000E 00

EIGENVECTOR NUMBER     3
          CORRESPONDING TO  5.1420751E 06
 -4.9753423E-01 -1.2593402E-03  1.[illegible]000000E 00 -2.[illegible]E-01 -9.5247[illegible]E-03  4.4244901E-01
  6.9603764E-02 -4.1012022E-03 -[illegible]1383357E-01  3.7484759E-01  6.2506551E-04 -7.4971844E-01

EIGENVECTOR NUMBER     4
          CORRESPONDING TO  1.8352988E 07
  1.6583501E-01  2.4433891E-01  5.3619381E-01  1.0000000E 00  9.8970757E-01  7.2062401E-01
  6.7693762E-01  5.8472794E-01  3.[illegible]655256E-01 -7.5719614E-01 -6.6895014E-01 -4.7047089E-01

EIGENVECTOR NUMBER     5
          CORRESPONDING TO  6.4669574E 07
  5.7137201E-01  7.3544332E-02 -8.7916167E-01 -3.2508349E-01  1.2794445E-01  1.0000000E 00
 -4.3410221E-01  2.7828129E-02  9.3674911E-01  3.9024470E-01 -4.7569560E-02 -9.6388316E-01

EIGENVECTOR NUMBER     6
          CORRESPONDING TO  1.4151228E 08
 [illegible].3664673E-01  5.5973433E-01  1.0000000E 00  5.1609533E-01  4.2956161E-01  6.8872845E-02
 -6.9697038E-01 -6.8903154E-01 -5.1254781E-01  2.5540307E-01  2.8571143E-01  2.8195343E-01

EIGENVECTOR NUMBER     7
          CORRESPONDING TO  3.0554062E 08
  1.0000000E 00  4.5324107E-01  7.[illegible]5686395E-01 -6.0666910E-01 -4.5810373E-01  1.[illegible]E-01
  6.4351924E-01  3.2966876E-01 -3.6562495E-01 -2.0519272E-01 -8.8199578E-02  1.603[illegible]718E-01

HERE ARE THE NATURAL FREQUENCIES

THE NATURAL FREQUENCY NUMBER   1  IS      82.477  CPS
THE NATURAL FREQUENCY NUMBER   2  IS     112.976  CPS
THE NATURAL FREQUENCY NUMBER   3  IS     361.114  CPS
THE NATURAL FREQUENCY NUMBER   4  IS     681.924  CPS
THE NATURAL FREQUENCY NUMBER   5  IS    1279.883  CPS
THE NATURAL FREQUENCY NUMBER   6  IS    1891.952  CPS
THE NATURAL FREQUENCY NUMBER   7  IS    2781.984  CPS
```

COMMON JOINTS FREE

COORDINATE NUMBERS FOR EACH Z DISPLACEMENT AT EACH UNRESTRAINED JOINT

JOINT NO.	COORD. NO.
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13

THE TIME ELAPSED FOR MATRIX INVERSION = 0.5021E 00 SECONDS

CFFJ MATRIX (STIFFNESS) - RELATES COMMON JOINTS TO FREE JOINTS - FOR COMPONENT MODE SYNTHESIS PROGRAM

ROW 1
-0.69170E 03 -0.11558E 04 -0.40527E 04

ROW 2
-0.14801E 06 -0.45948E 06 0.30303E 04

ROW 3
-0.35255E 03 -0.58759E 03 0.10224E 04

ROW 4
0.96467E 03 0.16078E 04 -0.43128E 01

ROW 5
0.30499E 05 0.50832E 05 0.63536E 01

ROW 6
0.49191E 03 0.61985E 03 -0.20490E 01

ROW 7
-0.32931E 03 -0.54845E 03 -0.41756E-02

ROW 8
-0.74917E 04 -0.12446E 05 0.46206E-01

ROW 9
-0.16794E 03 -0.27990E 03 -0.42030E-01

ROW 10
0.56336E 02 0.93894E 02 -0.57775E-04

ROW 11
0.12466E 04 0.20774E 04 0.47140E-03

ROW 12
0.28563E 02 0.47638E 02 -0.41760E-03

```
UPPER TRIANGLE OF CKIJ MATRIX (STIFFNESS) - COMMON JOINTS - FOR COMPONENT MODE SYNTHESIS PROGRAM

ROW   1
 0.12376E 06   0.41906E 06   0.

ROW   2
 0.17624E 07   0.

ROW   3
 0.20346E 05

CMFJ MATRIX (MASS) - RELATES COMMON JOINTS TO FREE JOINTS - FOR COMPONENT MODE SYNTHESIS PROGRAM

ROW   1
 0.           0.           0.
ROW   2
 0.           0.           0.
ROW   3
 0.           0.           0.
ROW   4
 0.           0.           0.
ROW   5
 0.           0.           0.
ROW   6
 0.           0.           0.
ROW   7
 0.           0.           0.
ROW   8
 0.           0.           0.
ROW   9
 0.           0.           0.
ROW  10
 0.           0.           0.
ROW  11
 0.           0.           0.
ROW  12
 0.           0.           0.

UPPER TRIANGLE OF CMIJ MATRIX (MASS) - COMMON JOINTS - FOR COMPONENT MODE SYNTHESIS PROGRAM

ROW   1
 0.12950E-02  0.           0.
ROW   2
 0.           0.
```

2.6 Program Listing

```
$       FORTRAN DECK
CMAIN          PROGRAM FLUENC-FOR GENERATING STIFFNESS,FLEXIBILITY AND MASS
C              MATRICES FROM PLANE GRID BEAM AND TRIANG. PLATE ELEMENTS
C       FLUENC-100 C FOR 100 DEGREES OF FREEDOM OR LESS. GENERATES PUNCHED
C       OUTPUT TO BE USED IN THE COMPONENT MODE SYNTHESIS PROGRAM.
C
        DIMENSION TITLE(24),YM(10),PR(10),GF(10),DENS(10),X(200),Y(200),
       1NR1(200),NR2(200),NR3(200),N1(200),N2(200),N3(200),NOSC(9),DCS(2),
       2STM(6,6),SMM(6,6),PLTK(9,9),PLTM(9,9),SSTF(25050),SM(25050),
       3RSMASS(300),A(25050),VALU(9),TEMP(100),B(300),C(200),DUM3(300),
       4F(300,3),IDUM4(100),JMASS(300),JTN(100)
        INTEGER OUT
        EQUIVALENCE(SSTF(1),SM(1),A(1)),(STM(1,1),SMM(1,1),PLTK(1,1),
       1PLTM(1,1))
 1000 FORMAT(12A6)
 1001 FORMAT(16I5)
 1002 FORMAT(8F10.3)
 1003 FORMAT(10X,2F10.3)
 1004 FORMAT(3F10.3,3I5)
 1005 FORMAT(F10.3,5I5)
 1006 FORMAT(I5,5X,F10.3)
 5000 FORMAT(1H1,12A6/1X,12A6)
 5001 FORMAT(///6HNJTS =I4,5X,6H  NR =I4,5X,6H NBE =I4,5X,6H NPE =I4,5X,
      17HNMODE =I3,5X,6HMKEY =I3,5X,7H LUMP =I3,
      25X,6HNCJT =I3,5X,7HNPUNJ =I2)
 5002 FORMAT(///73HM A T E R I A L    P R O P E R T I E S **************
      1********************/73HNO.     YOUNG*S MODULUS    POISSON RATIO
      1 MODULUS OF RIGIDITY    DENSITY,10(/I2,6X,E12.5,9X,F7.5,10X,E12.5,
      16X,E12.5))
 5003 FORMAT(///34HJ O I N T    C O O R D I N A T E S/35HJOINT NO.     X
      1 COORD.      Y COORD.)
 5004 FORMAT(I5,7X,F10.5,3X,F10.5)
 5006 FORMAT(///67HJ O I N T    R E S T R A I N T    C O D E ***********
      1**************/67HJOINT NO.     Z DISPLACEMENT    ROTATION ABOUT X
      1    ROTATION ABOUT Y)
 5007 FORMAT(I5,I16,I19,I20)
 5008 FORMAT(///75HB E A M    E L E M E N T    P R O P E R T I E S *****
      1**********************/75HELEMENT NO.        A              I
      1     I        MAT    JOINT 1    JOINT 2)
 5009 FORMAT(I6,8X,F9.4,4X,F9.4,4X,F9.4,2X,I2,6X,I3,9X,I3)
 5010 FORMAT(///122HT R I A N G U L A R    P L A T E    E L E M E N T
      1 P R O P E R T I E S ****************************************
      1****/122HELEMENT NO.        T         MAT    JOINT 1    JOINT 2    JOINT
      13          DX               DY                D1              DXY         BETA)
 5011 FORMAT(I6,8X,F8.4,3X,I2,6X,I3,8X,I3,8X,I3,6X,F11.5,3X,F11.5,3X,
      1F11.5,3X,E11.5,3X,F6.2)
 5020 FORMAT(///69HCOORDINATE NUMBERS FOR EACH Z DISPLACEMENT AT EACH UN
      1RESTRAINED JOINT/25HJOINT NO.       COORD. NO.)
 5021 FORMAT(I5,I16)
 5022 FORMAT(///28HL U M P E D    W E I G H T S/23HJOINT NO.         WEIG
      1HT)
 5023 FORMAT(I5,6X,F10.4)
C       DISC ASSIGNMENTS
        IN=5
        OUT=6
        MDISC=7
        NDISC=8
        IDISC=9
        JDISC=10
        KDISC=11
```

```
      MMDISC=12
      NNDISC=13
      IIDISC=14
      JJDISC=15
C     BEGIN INPUT OF DATA
  100 READ(IN,1000) (TITLE(I),I=1,24)
      REWIND MDISC
      REWIND NDISC
      REWIND IDISC
      REWIND JDISC
      REWIND KDISC
      REWIND MMDISC
      REWIND NNDISC
      REWIND IIDISC
      REWIND JJDISC
      WRITE(OUT,5000) (TITLE(I),I=1,24)
      READ(IN,1001) NJTS,NR,NBE,NPE,NMODE,MKEY,NLUMP,NCJT,NPUNJ
C
C     NJTS=NO. OF JOINTS, NR=NO. OF JOINTS WITH RESTRAINTS
C     NBE=NO. OF BEAM ELEMENTS, NPE=NO. OF TRIANGULAR PLATE ELEMENTS
C     NMODE=NO. OF EIGENVALUES AND EIGENVECTORS DESIRED
C     MKEY = 1 DO NOT COMPUTE ELEMENTAL CONSISTENT MASS TERMS
C     MKEY = 2 COMPUTE ELEMENTAL CONSISTENT MASS TERMS
C     NLUMP = NO. OF LUMPED MASSES INPUT
C     NCJT  = 0 IF ONLY ONE COMPONENT IS CONSIDERED.
C     NCJT  = NO. OF COMMON JOINTS IF MORE THAN ONE COMPONENT IS CONSIDERED.
C     THE COMMON JOINTS MUST BE NUMBERED LAST.
C     NPUNJ =  0, BOTH MASS AND FLEXIBILITY MATRICES PUNCHED OUT
C     NPUNJ = -1, NO PUNCHED OUTPUT
C     NPUNJ =  1, ONLY MASS MATRIX AND ILOW,IHIGH CODE PUNCHED OUT
C     NPUNJ =  2, ONLY REDUCED FLEXIBILITY MATRIX PUNCHED OUT
C     IF NCJT IS GREATER THAN 0, NPUNJ MUST BE 0 SO THAT ALL OUTPUT WILL
C     BE PUNCHED INCLUDING THE STIFFNESS MATRIX, MODE SHAPES AND FREQ.
C     FOR THE COMPONENT MODE SYNTHESIS PROGRAM.
C
      WRITE(OUT,5001) NJTS,NR,NBE,NPE,NMODE,MKEY,NLUMP,NCJT,NPUNJ
C     INPUT MATERIAL PROPERTIES
      READ(IN,1001) NMAT
      DO 10 I=1,NMAT
      READ(IN,1002) YM(I),PR(I),GE(I),DENS(I)
C     YM=YOUNG'S MOD./10**6, PR=POISSON RATIO, GE=MOD. OF RIGIDITY
C     DENS=DENSITY
      IF(GE(I).EQ.0.) GE(I)=YM(I)/(2.*(1.+PR(I)))
      YM(I)=YM(I)*1.E6
   10 GE(I)=GE(I)*1.E6
      WRITE(OUT,5002) (I,YM(I),PR(I),GE(I),DENS(I),I=1,NMAT)
      DO 256 I=1,NMAT
  256 DENS(I)=DENS(I)/(32.174*12.)
C     INPUT JOINT COORDINATES
      READ(IN,1003) (X(M),Y(M),M=1,NJTS)
      WRITE(OUT,5003)
      WRITE(OUT,5004) (M,X(M),Y(M),M=1,NJTS)
      NNJTS=NJTS-1
      DO 500 I=1,NNJTS
      XCOR =X(I)
      MOR=I+1
      DO 499 II=MOR,NJTS
      XMOR=X(II)
      IF(XMOR-XCOR)499,495,499
  495 YCOR=Y(I)
```

```
      YMOR=Y(II)
      IF(YMOR-YCOR)499,485,499
  485 WRITE(OUT,5999) I,II
 5999 FORMAT(1H1,5X,31HA DATA ERROR HAS BEEN DETECTED./6X,34HTHE X AND Y
     1 COORDINATES OF JOINTS I3,1X,4HAND I3,1X,13HARE THE SAME.//6X,30HP
     2ROGRAM ENDED AND JOB DELETED.)
      CALL EXIT
  499 CONTINUE
  500 CONTINUE
C     INPUT JOINT RESTRAINT CODE
C     0=FREE
C     1=CLAMPED
      DO 12 I=1,NJTS
      NR1(I)=0
      NR2(I)=0
      NR3(I)=0
      N1(I)=0
      N2(I)=0
   12 N3(I)=0
      IF(NR.EQ.0) GO TO 80
      WRITE(OUT,5006)
      DO 11 I=1,NR
      READ(IN,1001) JT,M1,M2,M3
      NR1(JT)=M1
      NR2(JT)=M2
      NR3(JT)=M3
      WRITE(OUT,5007) JT,M1,M2,M3
      JTN(I)=JT
   11 CONTINUE
      IF(NR.EQ.1) GO TO 80
      NNR=NR-1
      DO 600 J=1,NNR
      JTT=JTN(J)
      JOT=J+1
      DO 599 JJ=JOT,NR
      JCT=JTN(JJ)
      IF(JCT-JTT)599,595,599
  595 WRITE(OUT,6999)JCT
 6999 FORMAT(1H1,5X,31HA DATA ERROR HAS BEEN DETECTED./6X,38HIN THE JOIN
     1T RESTRAINT LISTING, JOINT I3,1X,14HAPPEARS TWICE.//6X,30HPROGRAM
     2ENDED AND JOB DELETED.)
      CALL EXIT
  599 CONTINUE
  600 CONTINUE
   80 CONTINUE
C     INPUT LUMPED MASSES
      IF(NLUMP.EQ.0) GO TO 250
      READ(IN,1006) ((JMASS(I),RSMASS(I)),I=1,NLUMP)
      WRITE(OUT,5022)
      DO 251 I=1,NLUMP
      WRITE(OUT,5023) JMASS(I),RSMASS(I)
      RSMASS(I)=RSMASS(I)/(32.174*12.)
  251 CONTINUE
  250 CONTINUE
      IF(NBE.EQ.0) GO TO 202
      WRITE(OUT,5008)
      DO 650 NM=1,NBE
C     INPUT BEAM ELEMENT PROPERTIES
      READ(IN,1004) AR,XI,YJ,MAT,JTNR,JTFR
C     AR-AREA OF BEAM CROSS SECTION, XI=AREA MOMENT OF INERTIA,
```

```
C      YJ=EFFECTIVE TORSIONAL MOMENT OF INERTIA, MAT=MATERIAL CODE
C      JTNR,JTFR=JOINT NUMBERS AT ENDS
       WRITE(OUT,5009) NM,AR,XI,YJ,MAT,JTNR,JTFR
       IF(AR.EQ.0.0.OR.XI.EQ.0.0.OR.YJ.EQ.0.0.OR.MAT.EQ.0) GO TO 795
       IF(JTNR.EQ.JTFR) GO TO 796
       GO TO 799
  795  WRITE(OUT,7999)NM
 7999  FORMAT(1H1,5X,29HDATA ERROR HAS BEEN DETECTED./6X,39HA BEAM PROPE
      1RTY IS MISSING FOR ELEMENT    I3,1X,1H.//6X,30HPROGRAM ENDED AND JO
      2B DELETED.)
       CALL EXIT
  796  WRITE(OUT,7998)NM
 7998  FORMAT(1H1,5X,31HA DATA ERROR HAS BEEN DETECTED./6X,36HJOINT 1 AND
      1 JOINT 2 OF BEAM ELEMENT I3,1X,13HARE THE SAME.//6X,30HPROGRAM END
      2ED AND JOB DELETED.)
       CALL EXIT
  799  CONTINUE
       WRITE(IIDISC) AR,XI,YJ,MAT,JTNR,JTFR
  650  CONTINUE
  202  CONTINUE
       IF(NPE.EQ.0) GO TO 302
       WRITE(OUT,5010)
       DO 655 NM=1,NPE
C      INPUT TRIANGULAR PLATE ELEMENT PROPERTIES
       READ(IN,1005) PTH,MAT,JT1,JT2,JT3,NDX
C      PTH=PLATE THICKNESS, MAT=MATERIAL CODE,
C      JT1, JT2, JT3=JOINT NUMBERS AT CORNERS, ANGLE AT JT1 MUST NOT BE
C      90 DEGREES
C      DX,DY,D1,DXY,BETA - FLEXURAL RIGIDITY TERMS AND ANGLE OF MATERIAL
C      PRINCIPAL AXES W/O TRIANGLE LOCAL AXES
       IF(NDX.EQ.1) READ(IN,1002) DX,DY,D1,DXY,BETA
       IF(NDX.EQ.1) GO TO 18
       BETA=0.
       DX=(YM(MAT)*PTH**3)/(12.*(1.-PR(MAT)**2))
       DY=DX
       D1=PR(MAT)*DX
       DXY=((1.-PR(MAT))/2.)*DX
   18  BETA=BETA/57.2958
       WRITE(OUT,5011) NM,PTH,MAT,JT1,JT2,JT3,DX,DY,D1,DXY,BETA
       IF(PTH.EQ.0.0.OR.MAT.EQ.0) GO TO 895
       IF(JT1.EQ.JT2.OR.JT1.EQ.JT3.OR.JT2.EQ.JT3) GO TO 896
       GO TO 899
  895  WRITE(OUT,8999)NM
 8999  FORMAT(1H1,5X,29HDATA ERROR HAS BEEN DETECTED./6X,51HA TRIANGULAR
      1PLATE PROPERTY IS MISSING FOR ELEMENT I3,1X,1H.//6X,30HPROGRAM END
      2ED AND JOB DELETED.)
       CALL EXIT
  896  WRITE(OUT,8998)NM
 8998  FORMAT(1H1,5X,31HA DATA ERROR HAS BEEN DETECTED./6X,94HEITHER JOIN
      1TS 1 AND 2, OR JOINTS 1 AND 3, OR JOINTS 2 AND 3 DEFINING TRIANGUL
      2AR PLATE ELEMENT I3,1X,13HARE THE SAME.//6X,30HPROGRAM ENDED AND J
      3OB DELETED.)
       CALL EXIT
  899  CONTINUE
       WRITE(JJDISC) DX,DY,D1,DXY,BETA
       WRITE(IIDISC) PTH,MAT,JT1,JT2,JT3
  655  CONTINUE
  302  CONTINUE
       IF(NCJT) 657,656,657
  656  NK=1
```

```
      GO TO 658
  657 NK=2
      NNCJT = NJTS-NCJT+1
  658 DO 350 KKK=1,NK
      IF(KKK.EQ.1) GO TO 670
      REWIND IDISC
      REWIND MDISC
      REWIND MMDISC
      REWIND NDISC
      REWIND NNDISC
      DO 669 JT=NNCJT,NJTS
      NR1(JT)=0
      NR2(JT)=0
      NR3(JT)=0
  669 CONTINUE
      WRITE(OUT,659)
  659 FORMAT(1H1 41X,18HCOMMON JOINTS FREE)
C     GENERATE COORDINATE NUMBERS FOR EACH DEGREE OF FREEDOM, 0 IF
C     CLAMPED, NORMAL DISPLACEMENTS ARE NUMBERED FIRST
C     N1, N2, N3 CONTAIN COORD. NUMBERS FOR EACH JOINT
C     NREDU = NO. OF NORMAL DISPLACEMENTS
C     NDF = NO. OF DEGREES OF FREEDOM INCLUDING ROTATIONS
  670 CALL COORDN(NR1,NR2,NR3,N1,N2,N3,NJTS,NREDU,NDF,KKK,NNCJT)
      NOMASS=NDF-NREDU
      MM1=NREDU*NOMASS+(NREDU*(NREDU+1)/2)
      IF(NCJT.EQ.0) GO TO 671
      IF(KKK.EQ.2) GO TO 671
      WRITE(OUT,672)
  672 FORMAT(/// 25HCOMMON JOINTS CONSTRAINED)
  671 WRITE(OUT,5020)
      DO 50 I=1,NJTS
      IF(NR1(I).EQ.1) GO TO 50
      WRITE(OUT,5021) I,N1(I)
   50 CONTINUE
      NSSTF=NDF*(NDF+1)/2
      DO 13 I=1,MM1
   13 SSTF(I)=0.
      IK22=0
      REWIND IIDISC
      IF(NBE.EQ.0) GO TO 200
C     BEGIN TO GENERATE BEAM STIFFNESS TERMS
C     SET UP CODE NUMBERS FOR BEAM JOINTS
      DO 14 NM=1,NBE
      READ(IIDISC) AR,XI,YJ,MAT,JTNR,JTFR
      NOSC(1)=N1(JTNR)
      NOSC(2)=N2(JTNR)
      NOSC(3)=N3(JTNR)
      NOSC(4)=N1(JTFR)
      NOSC(5)=N2(JTFR)
      NOSC(6)=N3(JTFR)
      IF(MKEY.EQ.1) GO TO 253
C     STORE INFO. FOR LATER USE
      WRITE(IDISC) AR,XI,YJ,MAT,JTNR,JTFR,(NOSC(I),I=1,6)
  253 CONTINUE
      X1=X(JTNR)
      X2=X(JTFR)
      Y1=Y(JTNR)
      Y2=Y(JTFR)
      FLNTH=SQRT((X2-X1)**2+(Y2-Y1)** )
      CALL TRANS(X1,X2,Y1,Y2,FLNTH,DCS)
```

```
      E=YM(MAT)
      G=GI(MAT)
      CALL BEAMK(ELNTH,E,G,XI,YJ,STM,DCS)
      DO 15 K=1,6
      IF(NOSC(K).EQ.0) GO TO 15
      I=NOSC(K)
      DO 16 N=1,6
      IF(NOSC(N).EQ.0) GO TO 16
      J=NOSC(N)
      IF(J.LT.I) GO TO 16
      MM=(2*J+(I-1)*(2*NDF-I))/2
      MM2=MM-MM1
      IF(MM2)186,186,187
  186 SSTF(MM)=SSTF(MM)+STM(K,N)
      GO TO 16
  187 AS=STM(K,N)
      WRITE (MUDISC) MM2,AS
      IK22=IK22+1
   16 CONTINUE
   15 CONTINUE
   14 CONTINUE
  200 CONTINUE
      IF(NPE.EQ.0) GO TO 300
C     BEGIN TO GENERATE TRIANGULAR PLATE STIFFNESS TERMS
C     SET UP CODE NUMBERS FOR TRIANGULAR PLATE JOINTS
      REWIND JJDISC
      DO 17 NM=1,NPE
      READ(IIDISC) PTH,MAT,JT1,JT2,JT3
      READ(JJDISC) DX,DY,D1,DXY,BETA
      NOSC(1)=N1(JT1)
      NOSC(2)=N2(JT1)
      NOSC(3)=N3(JT1)
      NOSC(4)=N1(JT2)
      NOSC(5)=N2(JT2)
      NOSC(6)=N3(JT2)
      NOSC(7)=N1(JT3)
      NOSC(8)=N2(JT3)
      NOSC(9)=N3(JT3)
      IF(NKEY.EQ.1) GO TO 254
C     STORE INFO. FOR LATER USE
      WRITE(IDISC) PTH,MAT,JT1,JT2,JT3,(NOSC(I),I=1,9)
  254 CONTINUE
      BX1=X(JT1)
      BX2=X(JT2)
      BY1=Y(JT1)
      BY2=Y(JT2)
      Y2=SQRT((BX2-BX1)**2+(BY2-BY1)**2)
      CALL TRANS(BX1,BX2,BY1,BY2,Y2,DCS)
      X3=DCS(2)*(X(JT3)-BX1)-DCS(1)*(Y(JT3)-BY1)
      Y3=DCS(1)*(X(JT3)-BX1)+DCS(2)*(Y(JT3)-BY1)
      CALL PLATEK(Y2,X3,Y3,DX,DY,D1,DXY,BETA,DCS,PLTK)
      DO 19 K=1,9
      IF(NOSC(K).EQ.0) GO TO 19
      I=NOSC(K)
      DO 20 N=1,9
      IF(NOSC(N).EQ.0) GO TO 20
      J=NOSC(N)
      IF(J.LT.I) GO TO 20
      MM=(2*J+(I-1)*(2*NDF-I))/2
      MM2=MM-MM1
```

```
      IF(MM2)188,188,189
  188 SSTF(MM)=SSTF(MM)+PLTK(K,N)
      GO TO 20
  189 AS=PLTK(K,N)
      WRITE (MMDISC) MM2,AS
      IK22=IK22+1
 20   CONTINUE
   19 CONTINUE
   17 CONTINUE
  300 CONTINUE
C     STORE FOR REDUCTION
      DO 21 I=1,NREDU
      NS=(2*I+(I-1)*(2*NDF-I))/2
      NE=(2*NDF+(I-1)*(2*NDF-I))/2
   21 WRITE(MDISC) (SSTF(J), J=NS,NE)
      REWIND IDISC
      DO 22 I=1,MM1
   22 SM(I)=0.
      IM22=0
      IF(MKEY.EQ.1) GO TO 255
      IF(NBE.EQ.0) GO TO 201
C     GENERATE BEAM MASS MATRICES
      DO 23 NM=1,NBE
      READ(IDISC) AR,XI,YJ,MAT,JTER,JIER,(NOSC(I),I=1,6)
      X1=X(JTNR)
      X2=X(JTER)
      Y1=Y(JTNR)
      Y2=Y(JTER)
      FLNTH=SQRT((X2-X1)**2+(Y2-Y1)** )
      CALL TRANS(X1,X2,Y1,Y2,FLNTH,DCS)
      RHO=DENS(MAT)
      CALL BEAM3(FLNTH,RHO,AR,XI,YJ,SMM,DCS)
      DO 24 K=1,6
      IF(NOSC(K).EQ.0) GO TO 24
      I=NOSC(K)
      DO 25 N=1,6
      IF(NOSC(N).EQ.0) GO TO 25
      J=NOSC(N)
      IF(J.LT.I) GO TO 25
      MM=(2*J+(I-1)*(2*NDF-I))/2
      MM2=MM-MM1
      IF(MM2)190,190,191
  190 SM(MM)=SM(MM)+SMM(K,N)
      GO TO 25
  191 AS=SMM(K,N)
      WRITE (NMDISC)MM2,AS
      IM22=IM22+1
   25 CONTINUE
   24 CONTINUE
   23 CONTINUE
  201 CONTINUE
      IF(NPE.EQ.0) GO TO 301
C     GENERATE TRIANGULAR PLATE MASS MATRICES
      DO 26 NM=1,NPE
      READ(IDISC) PTH,MAT,JT1,JT2,JT3,(NOSC(I),I=1,9)
      BX1=X(JT1)
      BX2=X(JT2)
      BY1=Y(JT1)
      BY2=Y(JT2)
      X2=SQRT((BX2-BX1)**2+(BY2-BY1)**2)
```

```
      CALL TRANS(BX1,BX2,BY1,BY2,Y2,DCS)
      X3=DCS(2)*(X(JT3)-BX1)-DCS(1)*(Y(JT3)-BY1)
      Y3=DCS(1)*(X(JT3)-BX1)+DCS(2)*(Y(JT3)-BY1)
      PRHO=DENS(MAT)
      CALL PLATEM(Y2,X3,Y3,PRHO,PTH,DCS,PLTM)
      DO 27 K=1,9
      IF(NOSC(K).EQ.0) GO TO 27
      I=NOSC(K)
      DO 28 N=1,9
      IF(NOSC(N).EQ.0) GO TO 28
      J=NOSC(N)
      IF(J.LT.I) GO TO 28
      MM=(2*J+(I-1)*(2*NDF-I))/2
      MM2=MM-M01
      IF(MM2)192,192,193
  192 SM(MM)=SM(MM)+PLTM(K,N)
      GO TO 28
  193 AS=PLTM(K,N)
      WRITE (NMDISC)MM2,AS
      IM22=IM22+1
   28 CONTINUE
   27 CONTINUE
   26 CONTINUE
  301 CONTINUE
C     STORE FOR REDUCTION
  256 CONTINUE
      IF(NLUMP.EQ.0) GO TO 259
      DO 258 I=1,NLUMP
      NN=JMASS(I)
      IF(N1(NN).EQ.0) GO TO 258
      NNN=N1(NN)
      NS=(2*NNN+(NNN-1)*(2*NDF-NNN))/2
      SM(NS)=SM(NS)+RSMASS(I)
  258 CONTINUE
  259 CONTINUE
      DO 29 I=1,NREDU
      NS=(2*I+(I-1)*(2*NDF-I))/2
      NF=(2*NDF+(I-1)*(2*NDF-I))/2
   29 WRITE(NDISC) (SM(J),J=NS,NF)
      IF(KKEY.EQ.1) GO TO 305
      GO TO 324
  305 MM3=NOMASS*(NOMASS+1)/2
      AS=0.0
      DO 310 MM2=1,MM3
      WRITE (NMDISC)MM2,AS
      IM22=IM22+1
  310 CONTINUE
  324 IF(KKK.EQ.1) GO TO 325
      CALL DIVID (NREDU,NOMASS,MDISC,JDISC,IDISC,A,B,MMDISC,IK22)
      CALL ZROMAK(A,B,C,DUM3,NREDU,NOMASS,IDISC,JDISC,MDISC,KDISC,KKK)
      CALL ZROMAM(A,B,C,DUM3,NREDU,NOMASS,IDISC,JDISC,MDISC,KDISC)
      CALL DIVID (NREDU,NOMASS,NDISC,JDISC,IDISC,A,B,NNDISC,IM22)
      CALL ZROMAM(A,B,C,DUM3,NREDU,NOMASS,IDISC,JDISC,NDISC,KDISC)
      CALL COMS (NREDU,NCJT,MDISC,A,B,1)
      CALL COMS (NREDU,NCJT,NDISC,A,B,2)
      GO TO 350
  325 CALL EIGEN(A,VALU,TEMP,B,C,DUM3,F,IDUM4,IDISC,JDISC,KDISC,NDISC,
     1MDISC,NDF,NMODE,NMODE,NREDU,NOMASS,MMDISC,NNDISC,IK22,IM22,NPUNJ,
     2NCJT,KKK)
  350 CONTINUE
```

```
      GO TO 100
      END
$      FORTRAN DECK
CCOMS         FORMS, PRINTS, PUNCHES MASS OR STIFFNESS FJ AND JJ MATRICES
C                     FOR THE COMPONENT MODE SYNTHESIS PROGRAM
C     MDISC CONTAINS REDUCED MASS OR STIFFNESS MATRIX - COMMON JOINTS
C     FREE (INCLUDES ROTATION), ROTATION OF OTHER JOINTS ELIMINATED.
C     NP-PRINT AND PUNCH CODE, NP=1 FOR STIFFNESS, NP=2 FOR MASS
C
      SUBROUTINE COMS (NREDU,NCJT,MDISC,A,B,NP)
C
      DIMENSION A(1),B(1),CJI(1296)
C
    3 FORMAT(/2X,3HROW,I4/(9F14.5))
    4 FORMAT(///2X,101HCKFJ MATRIX (STIFFNESS) - RELATES COMMON JOINTS T
     1O FREE JOINTS - FOR COMPONENT MODE SYNTHESIS PROGRAM )
    8 FORMAT(/// 2X,96HCMFJ MATRIX (MASS) - RELATES COMMON JOINTS TO FRE
     1E JOINTS - FOR COMPONENT MODE SYNTHESIS PROGRAM )
   50 FORMAT(/// 2X,96HUPPER TRIANGLE OF CKJJ MATRIX (STIFFNESS) - COMMO
     1N JOINTS - FOR COMPONENT MODE SYNTHESIS PROGRAM )
   51 FORMAT(/// 2X,91HUPPER TRIANGLE OF CMJJ MATRIX (MASS) - COMMON JOI
     1NTS - FOR COMPONENT MODE SYNTHESIS PROGRAM )
      REWIND MDISC
      DATA 01/4HCKFJ/,02/4HCKJJ/,03/4HCMFJ/,04/4HCMJJ/
      MAX=NREDU*(NREDU+1)/2
      READ(MDISC)(A(I),I=1,MAX)
      IF(NP.EQ.2) GO TO 7
      WRITE(6,4)
      GO TO 11
    7 WRITE(6,8)
   11 IC=0
      N3=3*NCJT
      N=NREDU-3*NCJT
      DO 20 I=1,N
      NS=(2*I+(I-1)*(2*NREDU-I))/2
      NF=(2*NREDU+(I-1)*(2*NREDU-I))/2
      L=0
      NB=NS+N-I+1
      DO 15 K=NB,NF
      L=L+1
   15 B(L)=A(K)
      WRITE(6,3)I,(B(J),J=1,N3)
      IF(NP.EQ.2) GO TO 16
      CALL PUNC (B,1,N3,01,IC)
      GO TO 20
   16 CALL PUNC (B,1,N3,03,IC)
   20 CONTINUE
      IF(NP.EQ.2) GO TO 23
      WRITE(6,50)
      GO TO 24
   23 WRITE(6,51)
   24 IC=0
      N2=N+1
      L=0
      DO 30 I=N2,NREDU
      NS=(2*I+(I-1)*(2*NREDU-I))/2
      NF=(2*NREDU+(I-1)*(2*NREDU-I))/2
      N1=I-N
      WRITE(6,3)N1,(A(J),J=NS,NF)
      DO 25 K=NS,NF
```

```
      I=I+1
   25 CJJ(I)=A(K)
   30 CONTINUE
      DO 40 I=1,N3
      II=I-1
      IF(II.EQ.0) GO TO 32
      DO 31 J=1,II
      NU=(2*I+(J-1)*(2*I-J))/2+(J-1)*(N3-I)
   31 B(J)=CJJ(NU)
   32 CONTINUE
      NS=(2*I+(I-1)*(2*N3-I))/2
      NF=(2*N3+(I-1)*(2*N3-I))/2
      J=I
      DO 33 JJ=NS,NF
      B(J)=CJJ(JJ)
   33 J=J+1
      IF(NP.EQ.2) GO TO 35
      CALL PUNC (B,1,N3,02,IC)
      GO TO 40
   35 CALL PUNC (B,1,N3,04,IC)
   40 CONTINUE
      RETURN
      END
```

```
$      FORTRAN DECK
CZROMAK     GENERATES REDUCED STIFFNESS MATRIX FOR FLUENC-100 C
C
C      D IS A DUMMY VECTOR WITH STORAGE N OR M (LARGER)
C      A IS A DUMMY VECTOR WITH STORAGE  N*(N+1)/2  OR M*(M+1)/2 (LARGER)
C      B IS A DUMMY VECTOR WITH STORAGE  N OR M  (LARGER)
C      C IS A DUMMY VECTOR WITH STORAGE  N OR M  (LARGER)
C      N=NO. OF NORMAL DISPLACEMENTS
C      M=NO. OF ROTATIONAL  D.O.F.
C      NTPE CONTAINS  K11 MATRIX
C      MTPE CONTAINS  K12 MATRIX
C      ITPE SCRATCH  TAPE
C      KTPE STORES K12*K22**(-1)
C      A INITIALLY CONTAINS K22
C***   REDUCED STIFFNESS MATRIX IS STORED ON ITPE
       SUBROUTINE ZROMAK(A,B,C,D,N,M,NTPE,MTPE,ITPE,KTPE,KKK)
       DIMENSION A(1),B(1),C(1),D(1)
       DOUBLE PRECISION SUM,DP1,DP2
       NMAX=N*(N+1)/2
       MMAX=M*(M+1)/2
       IF(KKK.EQ.1) GO TO 5
       REWIND ITPE
       WRITE(ITPE) (A(I),I=1,MMAX)
     5 CALL SYMINV(A,M)
       REWIND MTPE
       REWIND ITPE
       REWIND NTPE
       REWIND KTPE
       DO 10   KK=1,N
       READ(MTPE) (B(I),I=1,M)
       ICNT=0
       DO 1000  IK=1,M
       JJ=IK
       JK=IK
       DO 20    I=JJ,M
       ICNT=ICNT+1
  20   C(I)=A(ICNT)
       JJ=JJ-1
       JA=M
       ID=IK
       DO 30 J=1,JJ
       IF(JJ.EQ.0) GO TO 30
       C(J)=A(ID)
       JA=JA-1
       ID=ID+JA
  30   CONTINUE
       SUM=0.0D0
       DO 50    J=1,M
       DP1=B(J)
       DP2=C(J)
  50   SUM=SUM+DP1*DP2
       D(IK)= SUM
  1000 CONTINUE
       IF(KKK.EQ.2) GO TO 11
       WRITE (ITPE)  (D(J),J=1,M)
    11 WRITE (KTPE)  (D(J),J=1,M)
  10   CONTINUE
       IF(KKK.EQ.1) GO TO 12
       READ(ITPE) (A(I),I=1,MMAX)
       GO TO 100
```

```
   12 REWIND ITPE
      REWIND MTPE
      REWIND NTPE
      REWIND KTPE
      READ (NTPE) (A(J),J=1,NMAX)
      ICNT=0
      DO 60  KK=1,N
      READ (ITPE) (D(J),J=1,M)
      KI=KK
      DO 70   KJ=1,N
      READ(MTPE)(C(J),J=1,M)
      KP=KJ
      IF(KP.LT.KI)  GO TO  70
      SUM=0.0D0
      DO 80   KR=1,M
      DP1=D(KR)
      DP2=C(KR)
 80   SUM=SUM +DP1*DP2
      ICNT=ICNT+1
      SM=SUM
      A(ICNT)=A(ICNT)-SM
 70   CONTINUE
      REWIND MTPE
 60   CONTINUE
      REWIND NTPE
      REWIND MTPE
      REWIND ITPE
      WRITE(ITPE) (A(I),I=1,NMAX)
      REWIND ITPE
  100 RETURN
      END
```

```
$       FORTRAN DECK
CEIGEN       REDUCES STIFFNESS MATRIX AND INVERTS IT, REDUCES MASS MATRIX
C            DETERMINES EIGENVALUES AND EIGENVECTORS FOR FLUENC-100 C
C       THE ARGUMENTS ARE=
C     A - VECTOR OF LENGTH NRDF*(NRDF+1)/2
C    VALU - VECTOR OF LENGTH NEIG
C       TEMP,B,C,DUM3, - VECTORS OF LENGTH NRDF OR NMASS (SMALLER)
C       E - MATRIX OF DIMENSION (NRDF,3)
C       IDUM4 - VECTOR OF LENGTH NRDF OR NMASS (SMALLER)
C       ITAPE,JTAPE, NTAPE, MTAPE, - THESE ARE VARIOUS TAPES
C       NRDF - NUMBER OF DEGREES OF FREEDOM OF THE SYSTEM
C       NEIG - NUMBER OF EIGENVALUES DESIRED
C       NVEC - NUMBER OF EIGENVECTORS DESIRED
C       NMASS=NO. OF NORMAL DISPLACEMENTS
C       NOMASS=NO. OF ROTATIONAL DEGREES OF FREEDOM
C       STIFF IS ON MTAPE IN COMPACT FORM
C       MASS IS ON NTAPE IN COMPACT FORM
        SUBROUTINE EIGEN(A,VALU,TEMP,B,C,DUM3,E,IDUM4,ITAPE,JTAPE,KTAPE,
       1NTAPE,MTAPE,NRDF,NEIG,NVEC,NMASS,NOMASS,MMTAPE,NNTAPE,IK22,IM22,
       2NPUNI,NCIT,KKK)
        DIMENSION DUM3(NRDF),IDUM4(1),A(1),VALU(1),B(1),C(1),E(NRDF,3),
       1TEMP(1)
        DIMENSION ILOW(100),IHIGH(100)
        INTEGER OUT
        DATA Q5/1HCKEL/,Q6/4HFLEX/,Q7/4HWGHT/,Q8/4HFREQ/
        DO 56 II=1,NMASS
        ILOW(II)=1
   56   IHIGH(II)=NMASS
        OUT=6
        REWIND MTAPE
        REWIND NTAPE
        NTEMP=NMASS
        CALL DIVID(NMASS,NOMASS,MTAPE,JTAPE,ITAPE,A,B,MMTAPE,IK22)
        CALL ZROMAK(A,B,C,DUM3,NMASS,NOMASS,ITAPE,JTAPE,MTAPE,KTAPE,KKK)
        CALL DIVID(NMASS,NOMASS,NTAPE,JTAPE,ITAPE,A,B,NNTAPE,IM22)
        CALL ZROMAM(A,B,C,DUM3,NMASS,NOMASS,ITAPE,JTAPE,NTAPE,KTAPE)
  345   CONTINUE
        REWIND MTAPE
        REWIND NTAPE
        NREDU=NMASS
        NRMX=NREDU*(NREDU+1)/2
C       READ IN THE STIFFNESS MATRIX
        READ(MTAPE) (A(I),I=1,NRMX)
        WRITE(OUT,5500)
 5500   FORMAT(///85HR E D U C E D    U P P E R    T R I A N G U L A R
       1S T I F F N E S S    M A T R I X)
        DO 5501 I=1,NREDU
        NS=(2*I+(I-1)*(2*NREDU-I))/2
        NF=(2*NREDU+(I-1)*(2*NREDU-I))/2
        WRITE(OUT,5502) I,(A(J),J=NS,NF)
 5502   FORMAT(/3HROW,I4/(9F14.5))
 5501   CONTINUE
        IF(NCIT.EQ.0) GO TO 375
        IC=0
        DO 370 I=1,NREDU
        II=I-1
        IF(II.EQ.0) GO TO 368
        DO 367 J=1,II
        NU=(2*I+(J-1)*(2*I-J))/2+(J-1)*(NREDU-I)
  367   B(J)=A(NU)
```

```
  368 CONTINUE
      NS=(2*I+(I-1)*(2*NREDU-I))/2
      NE=(2*NREDU+(I-1)*(2*NREDU-I))/2
      J=1
      DO 369 JJ=NS,NE
      B(J)=A(JJ)
  369 J=J+1
      CALL PUNC (B,1,NREDU,06,IC)
  370 CONTINUE
  375 CONTINUE
      CALL SYMINV(A,NREDU)
      WRITE(OUT,5503)
 5503 FORMAT(///89HR E D U C E D    U P P E R    T R I A N G U L A R
     1F L E X I B I L I T Y    M A T R I X)
C     PUNCH OPTION
      IF(NCUT.GT.0) GO TO 8003
      IF(NPUNJ.EQ.0) GO TO 8000
      IF(NPUNJ) 8001,8002,8002
 8001 GO TO 8003
 8002 IF(NPUNJ.EQ.1) GO TO 8000
      GO TO 8003
 8000 PUNCH 5602,((ILOW(K),IHIGH(K)),K=1,NREDU)
 8003 CONTINUE
 5602 FORMAT (18I4)
      DO 5504 I=1,NREDU
      NS=(2*I+(I-1)*(2*NREDU-I))/2
      NE=(2*NREDU+(I-1)*(2*NREDU-I))/2
 5504 WRITE(OUT,5502) I,(A(J),J=NS,NE)
      IF(NPUNJ) 803,802,802
  802 IF (NPUNJ.EQ.1) GO TO 803
      IC=0
      DO 5507 I=1,NREDU
      II=I-1
      IF(II.EQ.0) GO TO 5508
      DO 5509 J=1,II
      NU=(2*I+(J-1)*(2*I-J))/2+(J-1)*(NREDU-I)
 5509 B(J)=A(NU)
 5508 CONTINUE
      NS=(2*I+(I-1)*(2*NREDU-I))/2
      NE=(2*NREDU+(I-1)*(2*NREDU-I))/2
      J=1
      DO 5510 JJ=NS,NE
      B(J)=A(JJ)
 5510 J=J+1
      CALL PUNC (B,1,NREDU,06,IC)
 5507 CONTINUE
  803 CONTINUE
C     READ IN THE MASS MATRIX
      READ(NTAPE) (A(I),I=1,NRMX)
      DO 6012 I=1,NRMX
 6012 A(I)=A(I)*32.174*12.
      WRITE(OUT,5505)
 5505 FORMAT(///79HR E D U C E D    U P P E R    T R I A N G U L A R
     1W E I G H T    M A T R I X)
      DO 5506 I=1,NREDU
      NS=(2*I+(I-1)*(2*NREDU-I))/2
      NE=(2*NREDU+(I-1)*(2*NREDU-I))/2
 5506 WRITE(OUT,5502) I,(A(J),J=NS,NE)
      IF(NPUNJ.EQ.0) GO TO 700
      IF(NPUNJ) 701,702,702
```

```
 701 GO TO 703
 702 IF(NPUNJ.EQ.1)GO TO 700
     GO TO 703
 700 IC=0
     DO 5511 I=1,NREDU
     II=I-1
     IF(II.EQ.0) GO TO 5512
     DO 5513 J=1,II
     NU=(2*I+(J-1)*(2*I-J))/2+(J-1)*(NREDU-I)
5513 B(J)=A(NU)
5512 CONTINUE
     NS=(2*I+(I-1)*(2*NREDU-I))/2
     NE=(2*NREDU+(I-1)*(2*NREDU-I))/2
     J=I
     DO 5514 JJ=NS,NE
     B(J)=A(JJ)
5514 J=J+1
     CALL PUNC (B,1,NREDU,07,IC)
5511 CONTINUE
 703 CONTINUE
     IF(NEIG.EQ.0) RETURN
     CALL EIGMAT(NTEMP,A,VALU,TEMP,B,C,DUM3,E,IDUM4,MTAPE,NTAPE,JTAPE,
    1ITAPE,NEIG,NVEC,NCJT)
     DO 60 I=1,NEIG
     IF(VALU(I).LT.0.0) GO TO 59
     DUM3(I)=SQRT(VALU(I))/6.2831853
     GO TO 60
  59 DUM3(I)=0.0
  60 CONTINUE
     WRITE(OUT,9009)
     WRITE(OUT,9005) (I,DUM3(I),I=1, EIG)
     IF(NCJT.EQ.0) GO TO 380
     IC=0
     CALL PUNC (DUM3,1,NEIG,08,IC)
 380 CONTINUE
9009 FORMAT(/// 43X,33HHERE ARE THE NATURAL FREQUENCIES          ///)
9005 FORMAT(35X,29HTHE NATURAL FREQUENCY NUMBER  I3,2X,2HIS F12.3,2X,
    13HCPS)
     RETURN
     END
```

```
*      FORTRAN DECK
CCOORDN     ASSIGNS A COORD. NO. TO EACH DEGREE OF FREEDOM AT EACH JOINT
C           FOR FLUENC - 100 C
C      NR1,NR2,NR3 = ARRAYS CONTAINING RESTRAINT INFO. FOR EACH DEGREE
C      OF FREEDOM AT EACH JOINT (FREE=0, CLAMPED=1)
C      N1,N2,N3 = COORD. NO. FOR EACH DEGREE OF FREEDOM (NORMAL
C      DISPLACEMENTS ARE NUMBERED FIRST)
C      NJTS = NO. OF JOINTS
C      NRFDU = NO. OF NORMAL DISPLACEMENTS
C      NDF = TOTAL NO. OF DEGREES OF FREEDOM (INCLUDING ROTATIONS)
      SUBROUTINE COORDN(NR1,NR2,NR3,N1,N2,N3,NJTS,NRFDU,NDF,NT,NNCJT)
      DIMENSION NR1(1),NR2(1),NR3(1),N1(1),N2(1),N3(1)
      NO=1
      DO 10 I=1,NJTS
      IF(NR1(I).EQ.1) GO TO 10
      N1(I)=NO
      NO=NO+1
   10 CONTINUE
      IF(NT.EQ.2) GO TO 36
      NRFDU=NO-1
      DO 30 I=1,NJTS
      IF(NR2(I).EQ.1) GO TO 30
      N2(I)=NO
      NO=NO+1
   30 CONTINUE
      DO 35 I=1,NJTS
      IF(NR3(I).EQ.1) GO TO 35
      N3(I)=NO
      NO=NO+1
   35 CONTINUE
      GO TO 50
   36 DO 37 I=NNCJT,NJTS
      N2(I)=NO
      NO=NO+1
   37 CONTINUE
      DO 38 I=NNCJT,NJTS
      N3(I)=NO
      NO=NO+1
   38 CONTINUE
      NRFDU=NO-1
      NN=NNCJT-1
      DO 39 I=1,NN
      IF(NR2(I).EQ.1) GO TO 39
      N2(I)=NO
      NO=NO+1
   39 CONTINUE
      DO 40 I=1,NN
      IF(NR3(I).EQ.1) GO TO 40
      N3(I)=NO
      NO=NO+1
   40 CONTINUE
   50 NDF=NO-1
      RETURN
      END
```

```
$     FORTRAN DECK
CEIGMAT      FOR FLUENC - 100 C
C       THIS SUBROUTINE FINDS THE EIGENVALUES AND EIGENVECTORS FOR
C       SYMMETRIC MASS AND STIFFNESS MATRICES.
C       THE ARGUMENTS ARE--
C         N- ORDER OF MATRICES.
C         A-  DUMMY VECTOR WITH DIMENSION IN MAIN PROGRAM OF N*(N+1)/2
C         VALU-  STORAGE FOR EIGENVALUES. MUST BE DIMENSIONED IN THE MAIN
C                PROGRAM AS A VECTOR OF LENGTH NEIG.
C         TEMP,B,C,D,-  DUMMY VECTORS WITH DIMENSION OF N IN MAIN PROGRAM.
C         E-  DUMMY ARRAY WITH DIMENSIONS OF (N,3) IN MAIN PROGRAM.
C         IDUM-  DUMMY INTEGER VECTOR WITH DIMENSION OF N IN MAIN PROGRAM.
C         MTAPE-  TAPE WHERE STIFFNESS MATRIX IS STORED IN COMPACT FORM.
C         NTAPE-  TAPE WHERE MASS MATRIX IS STORED IN COMPACT FORM.
C         JTAPE,ITAPE-  SCRATCH TAPES.
C         NEIG-  NUMBER OF EIGRNVALUES DESIRED.
C         NVEC-  NUMBER OF EIGENVECTORS DESIRED. MUST BE EQUAL TO OR LESS
C                THAN NEIG.
C       THE MASS AND STIFFNESS MATRICES ARE STORED IN COMPACT FORM AS
C       VECTORS. ONLY THE UPPER TRIANGLE OF THESE MATRICES(BY ROWS) IS
C       STORED.
      SUBROUTINE EIGMAT(N,A,VALU,TEMP,B,C,D,E,IDUM,MTAPE,NTAPE,JTAPE,
     1ITAPE,NEIG,NVEC,NCJT)
      DIMENSION A(1),TEMP(1),VALU(1),B(1),C(1),D(1),E(N,3),IDUM(1)
      DOUBLE PRECISION SUM,SUM1
      INTEGER OUT
      OUT=6
      REWIND ITAPE
      REWIND JTAPE
      REWIND NTAPE
      REWIND MTAPE
      M=2*N
      NMAX=N*(N+1)/2
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C       STEP 1
C       READ IN M BY ROWS IN COMPACTED FORM
C       REPLACE M BY (L)TRANSPOSE, WHERE M=L*(L)TRANSPOSE
C       CALCULATE FIRST ROW
      READ (NTAPE) (A(I),I=1,NMAX)
      REWIND NTAPE
    5 CONTINUE
      A(1)=SQRT(A(1))
      DO 10 I=2,N
   10 A(I)=A(I)/A(1)
C     CALCULATE ALL THE OTHER ROWS
      IND=N
      DO 101 I=2,N
      IND=IND+1
      SUM=0.D0
      K1=I-1
      DO 50 JJ=1,K1
      MJ=(M-JJ)*(JJ-1)/2+I
   50 SUM=SUM+A(MJ)*A(MJ)
      A(IND)=DSQRT(A(IND)-SUM)
      IF(IND.EQ.NMAX) GO TO 100
      SUM1=A(IND)
      K1=I+1
      DO 99 J=K1,N
      IND=IND+1
      SUM=0.D0
```

```
      II=I-1
      DO 60 JJ=1,II
      K=(M-JJ)*(JJ-1)/2
      KI=K+I
      KJ=K+J
   60 SUM=SUM+A(KI)*A(KJ)
      A(IND)=(A(IND)-SUM)/SUM1
   99 CONTINUE
  100 CONTINUE
  101 CONTINUE
C     CHECK FOR SINGULAR MASS MATRIX
      DO 102 I=1,N
      KI=(M-I)*(I-1)/2+I
      IF(A(KI).EQ.0.) GO TO 1090
  102 CONTINUE
C     THIS COMPLETES STEP 1
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     STEP 2
C     WRITE (L)TRANSPOSE ON TAPE BY COLUMNS
C     PUT (L)TRANSPOSE INTO TEMPORARY STORAGE (TEMP--A VECTOR)
C     AND THEN WRITE TEMP ON TAPE
      KTAPE=NTAPE
  300 IND=0
      DO 340 J=1,N
      DO 330 I=1,J
      IND=IND+1
      MI1=(M-I)*(I-1)/2+J
      TEMP(IND)=A(MI1)
  330 CONTINUE
      WRITE(KTAPE) (TEMP(JJ),JJ=1,IND)
      IND=0
  340 CONTINUE
C     THIS COMPLETES STEP 2
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     STEP 3
C     ((L)TRANSPOSE) INVERSE REPLACES (L)TRANSPOSE IN CORE
C     REPLACEMENT IS DONE BY LAST COLUMN FIRST--WORKING UP THE COLUMN
      DO 410 I=1,N
      IND=(I*(M+3-I))/2-N
  410 A(IND)=1./A(IND)
      DO 499 J=2,N
      JJ=(N+2)-J
      DO 490 I=2,JJ
      IND=(N+J+I-3)*(JJ-I)/2
      SUM=0.D0
      K1=JJ-I+2
      DO 450 K=K1,JJ
      IDK=IND+K
      MK=(M-K)*(K-1)/2+JJ
  450 SUM=SUM+A(IDK)*A(MK)
      IND=IND+JJ
      ID1=IND-I+1
  490 A(IND)=-SUM*A(ID1)
  499 CONTINUE
C     END OF STEP 3
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     STEP 4
C     U=((L)TRANSPOSE)INVERSE
C      WRITE U ON TAPE BY ROWS
      WRITE(ITAPE) (A(I),I=1,NMAX)
```

```
C     FINISHED WITH STEP 4
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      STEP 5
C      WRITE U ON TAPE BY COLUMNS STARTING WITH THE LAST COLUMN FIRST
C      PUT U (LAST COLUMN FIRST) INTO TEMP AND THEN WRITE ON TAPE
       IND=0
       DO 555 K=1,N
       J=N-K+1
       DO 550 I=1,J
       IND=IND+1
       MI2=(M-I)*(I-1)/2+J
       TEMP(IND)=A(MI2)
  550  CONTINUE
       WRITE(JTAPE) (TEMP(JJ),JJ=1,IND)
       IND=0
  555  CONTINUE
C      END OF STEP 5
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      STEP 6
C      FORM KU
C      READ K INTO CORE
C      READ U INTO CORE A COLUMN AT A TIME IN REVERSE ORDER
C      REPLACE K BY KU COLUMN BY COLUMN STARTING WITH THE LAST COLUMN
C      AND WORKING UP THE COLUMN
       READ(MTAPE) (A(I),I=1,NMAX)
       REWIND JTAPE
       DO 690 JJ=1,N
       J=N+1-JJ
       READ(JTAPE) (TEMP(II),II=1,J)
       DO 690 II=1,J
       I=J+1-II
       SUM=0.D0
       DO 650 K=1,I
       MK1=(M-K)*(K-1)/2+I
  650  SUM=SUM+A(MK1)*TEMP(K)
       IND=(M-I)*(I-1)/2+J
       IF(I.EQ.J) GO TO 680
       K1=(M-I)*(I-1)/2
       I=I+1
       DO 660 K=I,J
       K1K=K1+K
  660  SUM=SUM+A(K1K)*TEMP(K)
  680  CONTINUE
       A(IND)=SUM
  690  CONTINUE
C      END OF STEP 6
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      STEP 7
C      FORM((L)INVERSE)*KU
C      KU IS IN CORE
C      READ IN L COLUMN BY COLUMN AND CALCULATE ((L)INVERSE)*KU
C      ROW BY ROW
C      CALCULATE THE FIRST ROW
       REWIND NTAPE
       READ(NTAPE) TEMP(1)
       DO 710 I=1,N
  710  A(I)=A(I)/TEMP(1)
C      NOW CALCULATE THE REST OF THE ROWS
       IND=N
       DO 799 I=2,N
```

```
      READ (NTAPE) (TEMP(JJ),JJ=1,I)
      DO 799 J=I,N
      IND=IND+1
      JJ=I-1
      SUM=0.D0
      DO 750 K=1,JJ
      MK2=(M-K)*(K-1)/2+J
  750 SUM=SUM+TEMP(K)*A(MK2)
  799 A(IND)=(A(IND)-SUM)/TEMP(I)
C     STEP 7 IS COMPLETE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     STEP 8
C     DETERMINE EIGENVALUES AND EIGENVECTORS OF THE NEW MATRIX
C      CHANGE THE SIGN OF A IN ORDER TO OBTAIN THE SMALLEST
C      EIGENVALUE FIRST
      DO 800 I=1,NMAX
  800 A(I)=-A(I)
      CALL EIGMAT(A,VALU,TEMP,B,C,D,F,IDUM,N,NEIG,NVEC,MTAPE)
C     CHANGE VALU BACK
      DO 850 I=1,NEIG
  850 VALU(I)=-VALU(I)
C     STEP 8 IS COMPLETE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      STEP 9
C     CHANGE EIGENVECTORS BACK
C     READ U INTO CORE BY ROWS
C     READ UNCHANGED EIGENVECTORS INTO CORE ONE AT A TIME
C      CHANGE AND PRINT EIGENVECTORS
      IF(NVEC.EQ.0) GO TO 2000
      WRITE(OUT,4001)
      REWIND ITAPE
      READ(ITAPE) (A(I),I=1,NMAX)
      REWIND MTAPE
      IF(NCJT.EQ.0) GO TO 860
      DATA U26/4HMODE/
      IC=0
  860 CONTINUE
      DO 999 JJ=1,NVEC
      READ(MTAPE) (TEMP(I),I=1,N)
      IND=0
      DO 910 I=1,N
      SUM=0.D0
      DO 909 J=I,N
      IND=IND+1
  909 SUM=SUM+A(IND)*TEMP(J)
  910 TEMP(I)=SUM
C     NORMALIZE THE EIGENVECTOR
      SUM=TEMP(1)
      DO 939 II=2,N
      IF(ABS(SUM)-ABS(TEMP(II))) 938,939,939
  938 SUM=TEMP(II)
  939 CONTINUE
      IF(SUM) 940,947,940
  940 CONTINUE
      DO 941 II=1,N
      TEMP(II)=TEMP(II)/SUM
  941 CONTINUE
  947 CONTINUE
      IF(NCJT.EQ.0) GO TO 999
      CALL PUNC (TEMP,1,N,U26,IC)
```

```
  999 WRITE(OUT,4000) JJ,VALU(JJ),(TEMP(I),I=1,N)
C     STEP 9 IS COMPLETE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      GO TO 2000
 4000 FORMAT (1H0,19H EIGENVECTOR NUMBER  I5/12X,17H CORRESPONDING TO
     11PE15.7/(1H 1P6E15.7))
 4001 FORMAT(1H1,38X,43HHERE ARE THE EIGENVALUES AND EIGENVECTORS   ///)
 4002 FORMAT(1H1,38X,27HTHE MASS MATRIX IS SINGULAR   ///)
 1090 WRITE(OUT,4002)
 2000 RETURN
      END
```

```
$       FORTRAN DECK
CCMAT
C       THIS SUBROUTINE FORMS THE C MATRIX RELATING THE CORNER
C       DISPLACEMENTS TO THE POLYNOMIAL DEFLECTION COEFFICIENTS
C       FOR THE TRIANGULAR PLATE ELEMENT
C       Y2,X3,Y3 = COORDS. OF PLATE CORNERS IN LOCAL COORDINATES
C       C = C MATRIX
        SUBROUTINE CMAT(Y2,X3,Y3,C)
        DIMENSION C(9,9)
        DO 10 I=1,9
        DO 10 J=1,9
     10 C(I,J)=0.
        C(1,1)=1.
        C(2,3)=1.
        C(3,2)=-1.
        C(4,1)=1.
        C(4,3)=Y2
        C(4,6)=Y2**2
        C(4,9)=Y2**3
        C(5,3)=1.
        C(5,6)=2.*Y2
        C(5,9)=3.*Y2**2
        C(6,2)=-1.
        C(6,5)=-Y2
        C(6,8)=-Y2**2
        C(7,1)=1.
        C(7,2)=X3
        C(7,3)=Y3
        C(7,4)=X3**2
        C(7,5)=X3*Y3
        C(7,6)=Y3**2
        C(7,7)=X3**3
        C(7,8)=X3*Y3**2+Y3*X3**2
        C(7,9)=Y3**3
        C(8,3)=1.
        C(8,5)=X3
        C(8,6)=2.*Y3
        C(8,8)=2.*X3*Y3+X3**2
        C(8,9)=3.*Y3**2
        C(9,2)=-1.
        C(9,4)=-2.*X3
        C(9,5)=-Y3
        C(9,7)=-3.*X3**2
        C(9,8)=-(Y3**2+2.*X3*Y3)
        RETURN
        END
```

```
$       FORTRAN DECK
CDIVID
C       N=NO. OF NORMAL DISPLACEMENTS
C       M=NO. OF ROTATIONAL  D.O.F.
C       NTPE-CONTAINS STIFFNESS (OR MASS) MATRIX
C       NNTPE-CONTAINS K22 (OR M22)
C       MTPE-K12 (M12) STORED
C       ITPE-K11 (M11) STORED
C       A- DUMMY  STORAGE VECTOR,LARGER  OF (N*(N+1)/2 OR M*(M+1)/2)
        SUBROUTINE DIVID (N,M,NTPE,MTPE,ITPE,A,B,NNTPE,I22)
        DIMENSION A(1),B(1)
        REWIND ITPE
        REWIND NTPE
        REWIND MTPE
        REWIND NNTPE
        NMAX=N*(N+1)/2
        MMAX=M*(M+1)/2
        NM=N+M
        ICNT=0
        DO 10 I=1,N
        II=NM-I+1
        READ(NTPE) (B(J),J=1,II)
        ID=II-M
        DO 20 J=1,ID
        ICNT=ICNT+1
 20     A(ICNT)=B(J)
        ID1=ID+1
        JCNT=0
        DO 30    J=ID1,II
        JCNT= JCNT+1
 30     B(JCNT)=B(J)
        WRITE(MTPE) (B(J),J=1,M)
 10     CONTINUE
        WRITE(ITPE) (A(J),J=1,NMAX)
        REWIND MTPE
        REWIND ITPE
        DO 65 I=1,MMAX
   65   A(I)=0.0
        DO 50 I=1,I22
        READ (NNTPE)MM2,AS
        A(MM2)=A(MM2)+AS
 50     CONTINUE
        RETURN
        END
$        FORTRAN DECK
CBEAMK        PLANE GRID BEAM ELEMENT STIFFNESS MATRIX IN SYSTEM COORDS.
C       FL = BEAM LENGTH
C       E = YOUNG*S MODULUS
C       G = MODULUS OF RIGIDITY
C       XI = AREA MOMENT OF INERTIA
C       YJ = EFFECTIVE TORSIONAL MOMENT OF INERTIA
C       STM = STIFFNESS MATRIX
C       DCS = DIRECTION COSINES
        SUBROUTINE BEAMK(FL,E,G,XI,YJ,STM,DCS)
        DIMENSION STM(6,6),DCS(2)
        Z1=E*XI/FL
        Z2=G*YJ/FL
        STM(1,1)=12.*Z1/(FL*FL)
        STM(2,1)=6.*Z1*DCS(2)/FL
        STM(2,2)=4.*Z1*DCS(2)*DCS(2)+Z2*DCS(1)*DCS(1)
```

```
      STM(3,1)=-6.*Z1*DCS(1)/FL
      STM(3,2)=(-4.*Z1+Z2)*DCS(1)*DCS(2)
      STM(3,3)=4.*Z1*DCS(1)*DCS(1)+Z2*DCS(2)*DCS(2)
      STM(4,1)=-STM(1,1)
      STM(4,2)=-STM(2,1)
      STM(4,3)=-STM(3,1)
      STM(4,4)=STM(1,1)
      STM(5,1)=STM(2,1)
      STM(5,2)=2.*Z1*DCS(2)*DCS(2)-Z2*DCS(1)*DCS(1)
      STM(5,3)=-(2.*Z1+Z2)*DCS(1)*DCS(2)
      STM(5,4)=-STM(2,1)
      STM(5,5)=STM(2,2)
      STM(6,1)=STM(3,1)
      STM(6,2)=STM(5,3)
      STM(6,3)=2.*Z1*DCS(1)*DCS(1)-Z2*DCS(2)*DCS(2)
      STM(6,4)=-STM(3,1)
      STM(6,5)=STM(3,2)
      STM(6,6)=STM(3,3)
      DO 10 I=2,6
      N=I-1
      DO 10 J=1,N
   10 STM(J,I)=STM(I,J)
      RETURN
      END
```

```
$       FORTRAN DECK
CBEAMM        PLANE GRID-BEAM ELEMENT MASS MATRIX IN SYSTEM COORDS.
C       FL = BEAM LENGTH
C       RHO = DENSITY
C       A = CROSS SECTIONAL AREA
C       XI = AREA MOMENT OF INERTIA
C       XJ = EFFECTIVE TORSIONAL MOMENT OF INERTIA
C       SMM = MASS MATRIX
C       DCS = DIRECTION COSINES
        SUBROUTINE BEAMM(FL,RHO,A,XI,YJ,SMM,DCS)
        DIMENSION SMM(6,6),DCS(2)
        Z1=RHO*A*FL
        Z2=FL**2
        Z3=XI/A
        DD=Z1*(13./35.+(6.*Z3)/(5.*Z2))
        CC=Z1*(11.*FL/210.+Z3/(10.*FL))
        AA=Z1*(Z2/105.+2.*Z3/15.)
        TT=Z1*YJ/(3.*A)
        RR=Z1*(9./70.-(6.*Z3)/(5.*Z2))
        QQ=Z1*(13.*FL/420.-Z3/(10.*FL))
        SS=-Z1*(Z2/140.+Z3/30.)
        PP=Z1*YJ/(6.*A)
        SMM(1,1)=DD
        SMM(2,1)=CC*DCS(2)
        SMM(2,2)=AA*DCS(2)*DCS(2)+TT*DCS(1)*DCS(1)
        SMM(3,1)=-CC*DCS(1)
        SMM(3,2)=(-AA+TT)*DCS(1)*DCS(2)
        SMM(3,3)=AA*DCS(1)*DCS(1)+TT*DCS(2)*DCS(2)
        SMM(4,1)=RR
        SMM(4,2)=QQ*DCS(2)
        SMM(4,3)=-QQ*DCS(1)
        SMM(4,4)=SMM(1,1)
        SMM(5,1)=-SMM(4,2)
        SMM(5,2)=SS*DCS(2)*DCS(2)+PP*DCS(1)*DCS(1)
        SMM(5,3)=(-SS+PP)*DCS(1)*DCS(2)
        SMM(5,4)=-SMM(2,1)
        SMM(5,5)=SMM(2,2)
        SMM(6,1)=-SMM(4,3)
        SMM(6,2)=SMM(5,3)
        SMM(6,3)=SS*DCS(1)*DCS(1)+PP*DCS(2)*DCS(2)
        SMM(6,4)=-SMM(3,1)
        SMM(6,5)=SMM(3,2)
        SMM(6,6)=SMM(3,3)
        DO 10 I=2,6
        K=I-1
        DO 10 J=1,N
     10 SMM(J,I)=SMM(I,J)
        RETURN
        END
$        FORTRAN DECK
CPLATEK
C       THIS SUBROUTINE DETERMINES THE STIFFNESS MATRIX OF A
C       TRIANGLE PLATE ELEMENT IN SYSTE  COORDS.
C       Y2,X3,Y3 = COORDS. OF PLATE CORNERS IN LOCAL COORDINATES
C       DX,DY,D1,DXY,BETA = FLEXURAL RIGIDITY TERMS AND ANGLE OF MATERIAL
C       PRINCIPAL AXES W/O TRIANGLE LOCAL AXES
C       DCS = DIRECTION COSINES
C       PLTK = STIFFNESS MATRIX
        SUBROUTINE PLATEK(Y2,X3,Y3,DX,DY,D1,DXY,BETA,DCS,PLTK)
        DIMENSION PLTK(9,9),C(9,9),CINV(9,9),P(9,9),R(9,9)
```

```
      DIMENSION T(9,9),STIFF(9,9),DCS(2)
      EQUIVALENCE(P(1,1),STIFF(1,1)),        (R(1,1),T(1,1))
      CALL CMAT(Y2,X3,Y3,C)
      CALL MINV(C,CINV,9)
      CALL BINMAT(Y2,X3,Y3,DX,DY,D1,DXY,BETA,P)
      CALL MATMPY(P,CINV,R,9)
      DO 10 I=2,9
      N=I-1
      DO 10 J=1,N
      ZZ1=CINV(I,J)
      ZZ2=CINV(J,I)
      CINV(I,J)=ZZ2
      CINV(J,I)=ZZ1
   10 CONTINUE
      CALL MATMPY(CINV,R,STIFF,9)
      DO 400 I=1,9
      DO 400 J=1,9
  400 T(I,J)=0.
      T(1,1)=1.
      T(4,4)=1.
      T(7,7)=1.
      T(2,2)=DCS(2)
      T(3,3)=DCS(2)
      T(5,5)=DCS(2)
      T(6,6)=DCS(2)
      T(8,8)=DCS(2)
      T(9,9)=DCS(2)
      T(2,3)=-DCS(1)
      T(5,6)=-DCS(1)
      T(8,9)=-DCS(1)
      T(3,2)=DCS(1)
      T(6,5)=DCS(1)
      T(9,8)=DCS(1)
      CALL MATMPY(STIFF,T,C,9)
      T(2,3)=DCS(1)
      T(5,6)=DCS(1)
      T(8,9)=DCS(1)
      T(3,2)=-DCS(1)
      T(6,5)=-DCS(1)
      T(9,8)=-DCS(1)
      CALL MATMPY(T,C,PLTK,9)
      RETURN
      END
```

```
$       FORTRAN DECK
CMINV        MATRIX INVERSION SUBROUTINE
C       A = MATRIX TO BE INVERTED
C       U = INVERTED MATRIX
C       NM = ORDER OF MATRIX (.LE.9)
        SUBROUTINE MINV(A,U,NM)
        DIMENSION A(9,9),U(9,9)
        DO 9001 I=1,NM
        DO 9001 J=1,NM
        U(I,J)=0.0
        IF (I.EQ.J) U(I,J)=1.0
  9001  CONTINUE
        EPS=0.00000001
        DO 9015 I=1,NM
        K=I
        IF (I-NM) 9021,9007,9021
  9021  IF (A(I,I)-EPS) 9005,9006,9007
  9005  IF (-A(I,I)-EPS) 9006,9006,9007
  9006  K=K+1
        DO 9023 J=1,NM
        U(I,J)=U(I,J)+U(K,J)
  9023  A(I,J)=A(I,J)+A(K,J)
        GO TO 9021
  9007  DIV=A(I,I)
        DO 9009 J=1,NM
        U(I,J)=U(I,J)/DIV
  9009  A(I,J)=A(I,J)/DIV
        DO 9015 MM=1,NM
        DELT=A(MM,I)
        IF (ABS(DELT)-EPS) 9015,9015,9016
  9016  IF (MM-I) 9010,9015,9010
  9010  DO 9011 J=1,NM
        U(MM,J)=U(MM,J)-U(I,J)*DELT
  9011  A(MM,J)=A(MM,J)-A(I,J)*DELT
  9015  CONTINUE
        DO 9033 I=1,NM
        DO 9033 J=1,NM
  9033  A(I,J)=U(I,J)
        RETURN
        END
```

```
$       FORTRAN DECK
CDINMAT
C       THIS SUBROUTINE DETERMINES THE DOUBLE INTEGRAL MATRIX FOR
C       THE K EQUATION FOR THE TRIANGULAR PLATE ELEMENT
C       Y2,X3,Y3 = COORDS. OF PLATE CORNERS IN LOCAL COORDINATES
C       DX,DY,D1,DXY,BETA = FLEXURAL RIGIDITY TERMS AND ANGLE OF MATERIAL
C       PRINCIPAL AXES W/O TRIANGLE LOCAL AXES
C       P = DOUBLE INTEGRAL MATRIX
        SUBROUTINE DINMAT(Y2,X3,Y3,DX,DY,D1,DXY,BETA,P)
        DIMENSION P(9,9),D(3,3)
        DO 10 I=1,9
        DO 10 J=1,9
    10  P(I,J)=0.
        CALL DMAT(DX,DY,D1,DXY,BETA,D)
        A1=DBLINT(Y2,X3,Y3,0,0)
        A2=DBLINT(Y2,X3,Y3,1,0)
        A3=DBLINT(Y2,X3,Y3,2,0)
        A4=DBLINT(Y2,X3,Y3,0,1)
        A5=DBLINT(Y2,X3,Y3,0,2)
        A6=DBLINT(Y2,X3,Y3,1,1)
        P(4,4)=4.*D(1,1)*A1
        P(4,5)= 4.*D(1,3)*A1
        P(4,6)=4.*D(1,2)*A1
        P(4,7)=12.*D(1,1)*A2
        P(4,8)=4.*(D(1,1)*A4+D(1,2)*A2+2.*D(1,3)*(A2+A4))
        P(4,9)=12.*D(1,2)*A4
        P(5,5)=4.*D(3,3)*A1
        P(5,6)= 4.*D(3,2)*A1
        P(5,7)= 12.*D(3,1)*A2
        P(5,8)= 4.*(D(3,1)*A4+D(3,2)*A2+2.*D(3,3)*(A2+A4))
        P(5,9)= 12.*D(3,2)*A4
        P(6,6)=4.*D(2,2)*A1
        P(6,7)=12.*D(2,1)*A2
        P(6,8)=4.*(D(2,1)*A4+D(2,2)*A2+2.*D(2,3)*(A2+A4))
        P(6,9)=12.*D(2,2)*A4
        P(7,7)=36.*D(1,1)*A3
        P(7,8)=12.*(D(1,1)*A6+D(1,2)*A3+2.*D(1,3)*(A3+A6))
        P(7,9)=36.*D(1,2)*A6
        P(8,8)=4.*(D(1,1)*A5+D(1,2)*A6+2.*D(1,3)*(A6+A5))
      1        +4.*(D(2,1)*A6+D(2,2)*A3+2.*D(2,3)*(A3+A6))
      1        +8.*(D(3,1)*A6+D(3,2)*A3+2.*D(3,3)*(A3+A6))
      1        +8.*(D(3,1)*A5+D(3,2)*A6+2.*D(3,3)*(A6+A5))
        P(8,9)=12.*(D(1,2)*A5+D(2,2)*A6+2.*D(3,2)*(A6+A5))
        P(9,9)=36.*D(2,2)*A5
        DO 20 I=1,8
        K=I+1
        DO 20 J=K,9
    20  P(J,I)=P(I,J)
        RETURN
        END
```

```
$       FORTRAN DECK
CMATMPY
C       MULTIPLIES MATRICES A AND B TO GET C, ALL OF ORDER N*N
        SUBROUTINE MATMPY(A,B,C,N)
        DIMENSION A(9,9),B(9,9),C(9,9)
        DO 10 I=1,N
        DO 10 J=1,N
        C(I,J)=0.
        DO 10 K=1,N
   10   C(I,J)=C(I,J)+A(I,K)*B(K,J)
        RETURN
        END
```

```
$       FORTRAN DECK
CDMAT
C       THIS SUBROUTINE DETERMINES THE FLEXURAL RIGIDITY MATRIX IN
C       TRIANGLE LOCAL COORDINATES
C       DX,DY,D1,DXY,BETA = FLEXURAL RIGIDITY TERMS AND ANGLE OF MATERIAL
C       PRINCIPAL AXES W/O TRIANGLE LOCAL AXES
C       D - FLEXURAL RIGIDITY MATRIX IN TRIANGLE LOCAL COORDS.
        SUBROUTINE DMAT(DX,DY,D1,DXY,BETA,D)
        DIMENSION D(3,3)
        T11=(COS(BETA))**2
        T12=(SIN(BETA))**2
        T13=SIN(BETA)*COS(BETA)
        T21=T12
        T22=T11
        T23=-T13
        T31=-2.*SIN(BETA)*COS(BETA)
        T32=-T31
        T33=T11-T12
        Z11=DX*T11+D1*T12
        Z12=DX*T21+D1*T22
        Z13=DX*T31+D1*T32
        Z21=D1*T11+DY*T12
        Z22=D1*T21+DY*T22
        Z23=D1*T31+DY*T32
        Z31=DXY*T13
        Z32=DXY*T23
        Z33=DXY*T33
        D(1,1)=T11*Z11+T12*Z21+T13*Z31
        D(1,2)=T11*Z12+T12*Z22+T13*Z32
        D(1,3)=T11*Z13+T12*Z23+T13*Z33
        D(2,1)=T21*Z11+T22*Z21+T23*Z31
        D(2,2)=T21*Z12+T22*Z22+T23*Z32
        D(2,3)=T21*Z13+T22*Z23+T23*Z33
        D(3,1)=T31*Z11+T32*Z21+T33*Z31
        D(3,2)=T31*Z12+T32*Z22+T33*Z32
        D(3,3)=T31*Z13+T32*Z23+T33*Z33
        RETURN
        END
```

```
$       FORTRAN DECK
CDBLINT
C       THIS SUBROUTINE EVALUATES THE DOUBLE INTEGRALS APPEARING IN THE
C       EQUATIONS FOR K AND M FOR THE TRIANGULAR PLATE ELEMENT
C       Y2,X3,Y3 = COORDS. OF PLATE CORNERS IN LOCAL COORDINATES
C       M,N = POWER OF X-AND Y,RESPECTIVELY, PRZEMIENIECKI, PAGE 305
        FUNCTION DBLINT(Y2,X3,Y3,M,N)
        DIMENSION A1(2),B1(7),P1(7),P2(7),P3(7)
        EQUIVALENCE(B1(1),P3(1))
        IF(M-1) 40,41,42
   40   P1(1)=1.0
        N1=0
        GO TO 43
   41   P1(1)=-1.0
        P1(2)=1.0
        N1=1
        GO TO 43
   42   CONTINUE
        A1(1)=-1.0
        A1(2)=1.0
        B1(1)=-1.0
        B1(2)=1.0
        M1=1
        MM=M-1
        DO 10 J=1,MM
        CALL PLYMP(A1,1,B1,M1,P1,N1)
        NN1=N1+1
        DO 10 I=1,NN1
        B1(I)=P1(I)
        M1=N1
   10   CONTINUE
   43   CONTINUE
        IF(N-1) 50,51,52
   50   P2(1)=1.0
        N2=0
        GO TO 53
   51   P2(1)=-Y3+Y2
        P2(2)=Y3
        N2=1
        GO TO 53
   52   CONTINUE
        A1(1)=-Y3+Y2
        A1(2)=Y3
        B1(1)=-Y3+Y2
        B1(2)=Y3
        M1=1
        NN=N-1
        DO 20 J=1,NN
        CALL PLYMP(A1,1,B1,M1,P2,N2)
        NN2=N2+1
        DO 20 I=1,NN2
        B1(I)=P2(I)
        M1=N2
   20   CONTINUE
   53   CONTINUE
        CALL PLYMP(P1,N1,P2,N2,P3,N3)
        NN3=N3+1
        SOL=0.
        DO 30 I=1,NN3
        SOL=SOL+(X3**(M+1))*Y2*(1./FLOAT(M+N+2))* P3(I)*(1./FLOAT(N3+2-I))
```

```
30 CONTINUE
   DBLINT=SOL
   RETURN
   END
```

```
$       FORTRAN DECK
CPLYMP            12/65                                                      PLYMP
C                 POLYNOMIAL MULTIPLY                                        PLYMP
      SUBROUTINE PLYMP(A,L,B,M,C,N)                                          PLYMP
C     C1-0301         9-3-64                                                 PLYMP
      DIMENSION A(1),B(1),C(1)                                               PLYMP
      N = L + M                                                              PLYMP
      L1 = N + 1                                                             PLYMP
      DO 8 I = 1,L1                                                          PLYMP
 8    C(I) = 0                                                               PLYMP
      L2 = L + 1                                                             PLYMP
      M2 = M + 1                                                             PLYMP
      DO 1 I = 1,L2                                                          PLYMP
      DO 1 J = 1,M2                                                          PLYMP
      K = I+J                                                                PLYMP
 1    C(K-1) = C(K-1) + A(I)*B(J)                                            PLYMP
      DO 2 K = 1,L1                                                          PLYMP
      I1 = K                                                                 PLYMP
      IF(C(K)) 3,2,3                                                         PLYMP
 2    CONTINUE                                                               PLYMP
 3    IF(I1 - 1) 5,5,6                                                       PLYMP
 6    N = L1 - I1                                                            PLYMP
      N2 = N + 1                                                             PLYMP
      DO 7 J = 1,N2                                                          PLYMP
      N1 = J + I1 - 1                                                        PLYMP
 7    C(J) = C(N1)                                                           PLYMP
 5    RETURN                                                                 PLYMP
      END                                                                    PLYMP

$       FORTRAN DECK                                                             5
CPUNC         PUNCHES FULL MATRIX IN (1P6E12.5) FORMAT AND SEQUENCES CARDS       5
C
C     THE CALL PUNC STATEMENT MUST BE IN A LOOP.
C     EACH ROW STARTS ON A NEW CARD.
C     A IS THE ROW VECTOR TO BE PUNCHED.
C     NS IS THE FIRST ELEMENT OF A TO BE PUNCHED.
C     NE IS THE LAST ELEMENT OF A TO BE PUNCHED.
C     Q IS THE ALPHANUMERIC IDENTIFICATION CODE FOR THE MATRIX.
C     IC IS THE SEQUENCE NUMBER FOR THE FIRST CARD LESS ONE.
C     IC = 0, IF THE FIRST CARD FOR THE MATRIX IS TO BE SEQUENCED 1.             5
C                                                                                5
      SUBROUTINE PUNC (A,NS,NE,Q,IC)                                             5
      DIMENSION A(1)                                                             5
C                                                                                5
    1 FORMAT(1P1E12.5,60X,1A4,I4)                                                5
    2 FORMAT(1P2E12.5,48X,1A4,I4)                                                5
    3 FORMAT(1P3E12.5,36X,1A4,I4)                                                5
    4 FORMAT(1P4E12.5,24X,1A4,I4)                                                5
    5 FORMAT(1P5E12.5,12X,1A4,I4)                                                5
    6 FORMAT(1P6E12.5,1A4,I4)                                                    5
C                                                                                5
      NT=NE-NS+1                                                                 5
      N6=NT/6                                                                    5
      NC=N6*6                                                                    5
      N1=NS                                                                      5
      N2=N1+5                                                                    5
      IF(NT.LT.6) GO TO 20                                                       5
      DO 10 J=1,N6                                                               5
      IC=IC+1                                                                    5
      PUNCH6,(A(I),I=N1,N2),Q,IC
```

```
      N1=N2+1                                                           5
   10 N2=N2+6                                                           5
      IF(NT.EQ.NC) GO TO 50                                             5
   20 NO=NT-NC                                                          5
      IC=IC+1                                                           5
      GO TO(21,22,23,24,25),NO                                          5
   21 PUNCH 1,(A(I),I=N1,NE),Q,IC                                       5
      GO TO 50                                                          5
   22 PUNCH 2,(A(I),I=N1,NE),Q,IC                                       5
      GO TO 50                                                          5
   23 PUNCH 3,(A(I),I=N1,NE),Q,IC                                       5
      GO TO 50                                                          5
   24 PUNCH 4,(A(I),I=N1,NE),Q,IC                                       5
      GO TO 50                                                          5
   25 PUNCH 5,(A(I),I=N1,NE),Q,IC                                       5
   50 RETURN                                                            5
      END
```

```
$       FORTRAN DECK
CPLATEM
C       THIS SUBROUTINE DETERMINES THE MASS MATRIX OF A
C       TRIANGLE PLATE ELEMENT IN SYSTEM COORDS.
C       Y2,X3,Y3 = COORDS. OF PLATE CORNERS IN LOCAL COORDINATES
C       PRHO = DENSITY
C       PTH = PLATE THICKNESS
C       DCS = DIRECTION COSINES
C       PLTM = MASS MATRIX
        SUBROUTINE PLATEM(Y2,X3,Y3,PRHO,PTH,DCS,PLTM)
        DIMENSION PLTM(9,9),C(9,9),CINV(9,9),P(9,9),R(9,9)
        DIMENSION T(9,9),FMASS(9,9),DCS(2)
        EQUIVALENCE(P(1,1),FMASS(1,1)),          (R(1,1),T(1,1))
        CALL CMAT(Y2,X3,Y3,C)
        CALL MINV(C,CINV,9)
        CALL DINMTM(Y2,X3,Y3,PRHO,PTH,P)
        CALL MATMPY(P,CINV,R,9)
        DO 10 I=2,9
        N=I-1
        DO 10 J=1,N
        ZZ1=CINV(I,J)
        ZZ2=CINV(J,I)
        CINV(I,J)=ZZ2
        CINV(J,I)=ZZ1
   10   CONTINUE
        CALL MATMPY(CINV,R,FMASS,9)
        DO 400 I=1,9
        DO 400 J=1,9
  400   T(I,J)=0.
        T(1,1)=1.
        T(4,4)=1.
        T(7,7)=1.
        T(2,2)=DCS(2)
        T(3,3)=DCS(2)
        T(5,5)=DCS(2)
        T(6,6)=DCS(2)
        T(8,8)=DCS(2)
        T(9,9)=DCS(2)
        T(2,3)=-DCS(1)
        T(5,6)=-DCS(1)
        T(8,9)=-DCS(1)
        T(3,2)=DCS(1)
        T(6,5)=DCS(1)
        T(9,8)=DCS(1)
        CALL MATMPY(FMASS,T,C,9)
        T(2,3)=DCS(1)
        T(5,6)=DCS(1)
        T(8,9)=DCS(1)
        T(3,2)=-DCS(1)
        T(6,5)=-DCS(1)
        T(9,8)=-DCS(1)
        CALL MATMPY(T,C,PLTM,9)
        RETURN
        END

$       FORTRAN DECK
CDINMTM
C       THIS SUBROUTINE DETERMINES THE DOUBLE INTEGRAL MATRIX FOR
C       THE TRIANGULAR PLATE M MATRIX - PRZEMIENIECKI, PAGE 304
C       Y2,X3,Y3 = COORDS. OF PLATE CORNERS IN LOCAL COORDINATES
```

```
C       PRHO = DENSITY
C       PTH = PLATE THICKNESS
C       P = DOUBLE INTEGRAL MATRIX
        SUBROUTINE DINMTH(Y2,X3,Y3,PRHO,PTH,P)
        DIMENSION P(9,9)
        P(1,1)=DBLINT(Y2,X3,Y3,0,0)
        P(2,1)=DBLINT(Y2,X3,Y3,1,0)
        P(2,2)=DBLINT(Y2,X3,Y3,2,0)
        P(3,1)=DBLINT(Y2,X3,Y3,0,1)
        P(3,2)=DBLINT(Y2,X3,Y3,1,1)
        P(3,3)=DBLINT(Y2,X3,Y3,0,2)
        P(4,1)=P(2,2)
        P(4,2)=DBLINT(Y2,X3,Y3,3,0)
        P(4,3)=DBLINT(Y2,X3,Y3,2,1)
        P(4,4)=DBLINT(Y2,X3,Y3,4,0)
        P(5,1)=P(3,2)
        P(5,2)=P(4,3)
        P(5,3)=DBLINT(Y2,X3,Y3,1,2)
        P(5,4)=DBLINT(Y2,X3,Y3,3,1)
        P(5,5)=DBLINT(Y2,X3,Y3,2,2)
        P(6,1)=P(3,3)
        P(6,2)=P(5,3)
        P(6,3)=DBLINT(Y2,X3,Y3,0,3)
        P(6,4)=P(5,5)
        P(6,5)=DBLINT(Y2,X3,Y3,1,3)
        P(6,6)=DBLINT(Y2,X3,Y3,0,4)
        P(7,1)=P(4,2)
        P(7,2)=P(4,4)
        P(7,3)=P(5,4)
        P(7,4)=DBLINT(Y2,X3,Y3,5,0)
        P(7,5)=DBLINT(Y2,X3,Y3,4,1)
        P(7,6)=DBLINT(Y2,X3,Y3,3,2)
        P(7,7)=DBLINT(Y2,X3,Y3,6,0)
        P(8,1)=P(5,3)+P(4,3)
        P(8,2)=P(6,4)+P(5,4)
        P(8,3)=P(6,5)+P(5,5)
        P(8,4)=P(7,6)+P(7,5)
        P(8,5)=DBLINT(Y2,X3,Y3,2,3)+P(7,6)
        P(8,6)=DBLINT(Y2,X3,Y3,1,4)+DBLINT(Y2,X3,Y3,2,3)
        P(8,7)=DBLINT(Y2,X3,Y3,4,2)+DBLINT(Y2,X3,Y3,5,1)
        P(8,8)=DBLINT(Y2,X3,Y3,2,4)+DBLINT(Y2,X3,Y3,4,2)
       1    +2.*DBLINT(Y2,X3,Y3,3,3)
        P(9,1)=P(6,3)
        P(9,2)=P(6,5)
        P(9,3)=P(6,6)
        P(9,4)=DBLINT(Y2,X3,Y3,2,3)
        P(9,5)=DBLINT(Y2,X3,Y3,1,4)
        P(9,6)=DBLINT(Y2,X3,Y3,0,5)
        P(9,7)=DBLINT(Y2,X3,Y3,3,3)
        P(9,8)=DBLINT(Y2,X3,Y3,1,5)+DBLINT(Y2,X3,Y3,2,4)
        P(9,9)=DBLINT(Y2,X3,Y3,0,6)
        DO 10 I=1,9
        DO 10 J=1,I
     10 P(I,J)=P(I,J)*PRHO*PTH
        DO 20 I=2,9
        N=I-1
        DO 20 J=1,N
        P(J,I)=P(I,J)
     20 CONTINUE
        RETURN
```

```
$       FORTRAN DECK
C7ROMAM
C      N=NO. OF NORMAL DISPLACEMENTS
C      M=NO. OF ROTATIONAL  D.O.F.
C      NTPE CONTAINS M11 MATRIX
C      MTPE CONTAINS M12 MATRIX
C      ITPE SCRATCH TAPE
C      KTPE CONTAINS K12*K22**(-1)
C***   REDUCED MASS MATRIX IS STORED ON ITPE
       SUBROUTINE ZROMAM(A,B,C,D,N,M,NTPE,MTPE,ITPE,KTPE)
       DIMENSION A(1),B(1),C(1),D(1)
       DOUBLE PRECISION SUM1,SUM2,DP1,DP2,DP3
       NMASS=N
       REWIND MTPE
       REWIND ITPE
       REWIND NTPE
       REWIND KTPE
       NMAX=N*(N+1)/2
       DO 10 KK=1,N
       READ(KTPE) (B(I),I=1,M)
       ICNT=0
       DO 1000 IK=1,M
       JJ=IK
       JK=IK
       DO 20 J=JJ,M
       ICNT=ICNT+1
    20 C(J)=A(ICNT)
       JJ=JJ-1
       JA=M
       ID=IK
       DO 30 J=1,JJ
       IF(JJ.EQ.0) GO TO 30
       C(J)=A(ID)
       JA=JA-1
       ID=ID+JA
    30 CONTINUE
       SUM1=0.D0
       DO 50 J=1,M
       DP1=C(J)
       DP2=C(J)
    50 SUM1=SUM1+DP1*DP2
       D(IK)=SUM1
  1000 CONTINUE
       WRITE(ITPE) (D(J),J=1,M)
    10 CONTINUE
       REWIND ITPE
       REWIND MTPE
       REWIND NTPE
       REWIND KTPE
       READ(NTPE) (A(J),J=1,NMAX)
       DO 60 KK=1,N
       READ(MTPE) (B(J),J=1,M)
       READ(ITPE) (D(J),J=1,M)
       DO 70 KJ=1,N
       READ(KTPE) (C(J),J=1,M)
       SUM1=0.D0
       SUM2=0.D0
       DO 80 KR=1,M
       DP1=B(KR)
       DP2=D(KR)
```

```
      DP3=C(KR)
      SUM1=SUM1+DP1*DP3
   80 SUM2=SUM2+DP2*DP3
      SM1=SUM1
      SM2=SUM2
      IF(KJ.GE.KK) MM=(2*KJ+(KK-1)*(2*NMASS-KK))/2
      IF(KJ.GE.KK) A(MM)=A(MM)-SM1*SM
      IF(KJ.LE.KK) MM=(2*KK+(KJ-1)*(2*NMASS-KJ))/2
      IF(KJ.LE.KK) A(MM)=A(MM)-SM1
   70 CONTINUE
      REWIND KTPE
   60 CONTINUE
      REWIND NTPE
      REWIND MTPE
      REWIND ITPE
      REWIND KTPE
      WRITE(ITPE) (A(I),I=1,NMAX)
      REWIND ITPE
      RETURN
      END
```

```
*      FORTRAN DECK
CBIGMAT
C PROG.AUTHORS M.ELSON AND R.E.FUNDERLIC,CENTRAL DATA PROCESSING,4.1.65 BIGM0
      SUBROUTINE BIGMAT(A,VALU,VALL,UPPERD,DIAG,V,T,INTER,NN,NEIG,NVEC,
     1MTAPE)
      DIMENSION A(1),VALU(1),VALL(1),UPPERD(1),DIAG(1),V(1),T(NN,3),
     1INTER(1)
      REWIND MTAPE
      N7=0
      N=NN
      IF(N.LE.2)GO TO 49
      NP1=N+1
      NM1=N-1
      NM2=N-2
      NTNP1=N*N+1
      IX=0
      DO 10 I=1,NM2
      SIGMA2=0.
      IP1=I+1
      DO 1 J=IP1,N
      IJ=IX+J
    1 SIGMA2=SIGMA2+A(IJ)**2
      SIGMA=SQRT(SIGMA2)
      II=IX+I
      DIAG(I)=A(II)
      IIP1=IX+I+1
      UPPERD(I)=-SIGN(SIGMA,A(IIP1))
      T(I,2)=SIGMA2
      IF(ABS(SIGMA).GT.ABS(A(IIP1)))GO TO 2
      UPPERD(I)=A(IIP1)
      A(IIP1)=0.
      GO TO 10
    2 A(IIP1)=SQRT(1.+ABS(A(IIP1))/SIGMA)
      SQTGAM=-SIGN(SIGMA*A(IIP1),UPPERD(I))
      IP2=I+2
      DO 3 J=IP2,N
      IJ=IX+J
    3 A(IJ)=A(IJ)/SQTGAM
      JK1=I*(2*N-I-1)/2
      JX=JK1
      IIX=JK1
      DO 5 J=IP1,N
      VALL(J)=0.
      JK=JK1+J
      DO 4 K=IP1,J
      IK=IX+K
      VALL(J)=VALL(J)+A(JK)*A(IK)
    4 JK=JK+N-K
      IF(J.EQ.N)GO TO 6
      CALL LOOP1(J+2,NP1,VALL(J),A(JX),A(IX))
    5 JX=JX+N-J
    6 DELGAM=0.
      DO 7 J=IP1,N
      IJ=IX+J
    7 DELGAM=DELGAM+A(IJ)*VALL(J)
      DG02=.5*DELGAM
      DO 8 J=IP1,N
      IJ=IX+J
    8 T(J,1)=VALL(J)-DG02*A(IJ)
```

```
      DO 9 II=IP1,N                                                      BIGMO
      III=IX+II                                                          BIGMO
      CALL LOOP2(A(IIX),A(IX),T(NZ,1),T(II,1),A(III),II+1,NP1)
    9 IIX=IIX+N-II                                                       BIGMO
   10 IX=IX+N-I                                                          BIGMO
      M=N*(N+1)/2                                                        BIGMO
      UPPERD(NM1)=A(M-1)                                                 BIGMO
      T(NM1,2)=UPPERD(NM1)**2
      DIAG(NM1)=A(M-2)                                                   BIGMO
      DIAG(N)=A(M)                                                       BIGMO
      ENORM=AMAX1(ABS(DIAG)+ABS(UPPERD),ABS(DIAG(N))+ABS(UPPERD(NM1)))   BIGMO
      DO 11 I=2,NM1                                                      BIGMO
      ENRTMP=ABS(DIAG(I))+ABS(UPPERD(I))+ABS(UPPERD(I-1))                BIGMO
   11 IF(ENRTMP.GT.ENORM)ENORM=ENRTMP                                    BIGMO
      DO 12 I=1,NEIG                                                     BIGMO
      VALU(I)=ENORM                                                      BIGMO
   12 VALL(I)=-ENORM                                                     BIGMO
      DO 24 I=1,NEIG                                                     BIGMO
   13 ROOT=.5*(VALU(I)+VALL(I))                                          BIGMO
      IF(ROOT.EQ.VALL(I).OR.ROOT.EQ.VALU(I))GO TO 24                     BIGMO
      NAGREE=0                                                           BIGMO
      PM2=0.                                                             BIGMO
      PM1=1.                                                             BIGMO
      DO 21 J=1,N                                                        BIGMO
      IF(PM2.NE.0.)GO TO 15                                              BIGMO
   14 PM1=SIGN(1.,PM1)                                                   BIGMO
      GO TO 17                                                           BIGMO
   15 IF(PM1.NE.0.0) GO TO 17
   16 P=-SIGN(1.,PM2)                                                    BIGMO
      PM2=0.                                                             BIGMO
      IF(T(J-1,2)) 18,14,18
   17 P=DIAG(J)-ROOT-T(J-1,2)*PM2/PM1
      PM2=1.                                                             BIGMO
   18 IF(P)21,19,20                                                      BIGMO
   19 PM2=PM1                                                            BIGMO
      IF(PM2)21,20,20                                                    BIGMO
   20 NAGREE=NAGREE+1                                                    BIGMO
   21 PM1=P                                                              BIGMO
      DO 23 J=I,NEIG                                                     BIGMO
      IF(J.LE.NAGREE)GO TO 22                                            BIGMO
      IF(VALU(J).LE.ROOT)GO TO 13                                        BIGMO
      VALU(J)=ROOT                                                       BIGMO
      GO TO 23                                                           BIGMO
   22 VALL(J)=ROOT                                                       BIGMO
   23 CONTINUE                                                           BIGMO
      GO TO 13                                                           BIGMO
   24 CONTINUE                                                           BIGMO
      IF(NVEC.EQ.0)GO TO 49
      EPSLON=ENORM*1.E-8                                                 BIGMO
      COMPL1=COMPL(1)                                                    BIGMO
      DO 48 I=1,NVEC
      DO 25 J=1,N                                                        BIGMO
      V(J)=1.                                                            BIGMO
      T(J,2)=DIAG(J)-VALU(I)                                             BIGMO
      IF(J.EQ.N)GO TO 26                                                 BIGMO
      T(J,3)=UPPERD(J)                                                   BIGMO
   25 T(J+1,1)=UPPERD(J)                                                 BIGMO
   26 T(N,3)=0.                                                          BIGMO
      DO 29 J=1,N                                                        BIGMO
      IF(ABS(T(J,2)).LT.1.E-17)T(J,2)=EPSLON                             BIGMO
```

```
      T(J,1)=T(J,2)                                                     BIGMO
      T(J,2)=T(J,3)                                                     BIGMO
      T(J,3)=0.                                                         BIGMO
      IF(J.EQ.M)GO TO 30                                                BIGMO
      INTER(J)=0                                                        BIGMO
      JP1=J+1                                                           BIGMO
      IF(ABS(T(JP1,1)).LE.ABS(T(J,1)))GO TO 28                          BIGMO
      INTER(J)=1                                                        BIGMO
      DO 27 K=1,3                                                       BIGMO
      TEMP=T(J,K)                                                       BIGMO
      T(J,K)=T(JP1,K)                                                   BIGMO
   27 T(JP1,K)=TEMP                                                     BIGMO
   28 TMULTP=T(JP1,1)/T(J,1)                                            BIGMO
      VALI(J)=OR(INTER(J),AND(TMULTP,COMPL1))
      T(JP1,2)=T(JP1,2)-TMULTP*T(J,2)                                   BIGMO
   29 T(JP1,3)=T(JP1,3)-TMULTP*T(J,3)                                   BIGMO
   30 ITER=1                                                            BIGMO
   31 DO 32 J=1,N                                                       BIGMO
      L=N+1-J                                                           BIGMO
   32 V(L)=(V(L)-T(L,2)*V(L+1)-T(L,3)*V(L+2))/T(L,1)                    BIGMO
      VNORM=0.                                                          BIGMO
      DO 33 L=1,N                                                       BIGMO
   33 VNORM=VNORM+V(L)**2                                               BIGMO
      VNORM=SQRT(VNORM)                                                 BIGMO
      DO 34 J=1,N                                                       BIGMO
   34 V(J)=V(J)/VNORM                                                   BIGMO
      IF(ITER.EQ.2)GO TO 36                                             BIGMO
      ITER=2                                                            BIGMO
      DO 35 L=2,N                                                       BIGMO
      LM1=L-1                                                           BIGMO
      TRY=VALI(LM1)
      IF(AND(TRY,1).EQ.0) GO TO 35
      VTEMP=V(LM1)                                                      BIGMO
      V(LM1)=V(L)                                                       BIGMO
      V(L)=VTEMP                                                        BIGMO
   35 V(L)=V(L)-VALI(LM1)*V(LM1)
      GO TO 31                                                          BIGMO
   36 IF(VNORM.EQ.0.)V(1)=1.                                            BIGMO
      IIX=(N*N-N-6)/2                                                   BIGMO
      DO 37 KK=1,NM2                                                    BIGMO
      IIP1=N-KK                                                         BIGMO
      UTV=0.                                                            BIGMO
      CALL LOOP3(UTV,A(IIX),V(NZ),IIP1+1,NP1)                           BIGMO
      CALL LOOP4(A(IIX),V(NZ),NP1,IIP1+1,UTV)                           BIGMO
   37 IIX=IIX+IIP1-N-2                                                  BIGMO
      WRITE(MTAPE) (V(ICH),ICH=1,N)
   48 CONTINUE
   49 RETURN
      END
```

```
$       FORTRAN DECK
CLOOP1
        SUBROUTINE LOOP1(JP2,NP1,SGAMPJ,AJX,AIX)                               BIGMO
        DIMENSION AJX(1), AIX(1)                                               BIGMO
        DO 1 L=JP2,NP1                                                         BIGMO
      1 SGAMPJ=SGAMPJ+AJX(L)*AIX(L)                                            BIGMO
        RETURN                                                                 BIGMO
        END
$       FORTRAN DECK
CLOOP2
        SUBROUTINE LOOP2(AIIX,AIX,S,SI,AIII,IP1,NP1)                           BIGMO
        DIMENSION AIIX(1),AIX(1),S(1)                                          BIGMO
        DO 2 JJ=IP1,NP1                                                        BIGMO
      2 AIIX(JJ)=AIIX(JJ)-AIII*S(JJ)-SI*AIX(JJ)                                BIGMO
        RETURN                                                                 BIGMO
        END
$       FORTRAN DECK
CLOOP3
        SUBROUTINE LOOP3(UTV,AIIX,V,IIP2,NP1)                                  BIGMO
        DIMENSION AIIX(1), V(1)                                                BIGMO
        DO 3 J=IIP2,NP1                                                        BIGMO
      3 UTV=UTV+AIIX(J)*V(J)                                                   BIGMO
        RETURN                                                                 BIGMO
        END
$       FORTRAN DECK
CLOOP4
        SUBROUTINE LOOP4(AIIX,V,NP1,IIP2,UTV)                                  BIGMO
        DIMENSION AIIX(1),V(1)                                                 BIGMO
        DO 4 K=IIP2,NP1                                                        BIGMO
      4 V(K)=V(K)-AIIX(K)*UTV                                                  BIGMO
        RETURN                                                                 BIGMO
        END
$       FORTRAN DECK
CTRANS        TRANSFORMATION DIRECTION COSINES
C       X1,Y1 = COORDS. OF POINT 1
C       X2,Y2 = COORDS. OF POINT 2
C       FL = DISTANCE BETWEEN POINTS 1 AND 2
C       DCS = DIRECTION COSINES OF VECTOR FROM POINT 1 TO POINT 2
        SUBROUTINE TRANS(X1,X2 Y1,Y2,FL,DCS)
        DIMENSION DCS(2)
        DCS(1)=(X2-X1)/FL
        DCS(2)=(Y2-Y1)/FL
        RETURN
        END
```

```
$       FORTRAN DECK
CSYMINV
C       A  IS THE UPPER TRIANGLE OF THE SYMMETRIC MATRIX TO BE INVERTED.   SYMV
C       ELEMENTS ARE STORED ROWWISE.                                        SYMV
C       N - ORDER OF MATRIX                                                 SYMV
C       PROGRAM INVERTS IN PLACE.                                           SYMV
        SUBROUTINE SYMINV(A,N)                                              SYMV
        DIMENSION A(1)                                                      SYMV
        CALL ECLOCK(III)
        NMAX=N*(N+1)/2
        IF(A(1).LT.0.0) GO TO 25
        GO TO 99
   25   II=1
   26   WRITE(6,27)II
   27   FORMAT(1H1,5X,36HA NEGATIVE VALUE APPEARS IN ELEMENT        ,I5,1X,
       125HOF VECTOR TO BE INVERTED./6X,65HSINCE ELEMENT FALLS ON DIAGONAL
       2, MATRIX IS NOT POSITIVE DEFINITE.//6X,30HPROGRAM ENDED AND JOB DE
       3LETED.)
        DO 45 I=1,N
        NS=(2*I+(I-1)*(2*N-I))/2
        NE=(2*N+(I-1)*(2*N-I))/2
        WRITE(6,28)I,(A(J),J=NS,NE)
   28   FORMAT(/3HROW,I4/(9F14.5))
   45   CONTINUE
        CALL EXIT
   99   CONTINUE
        A(1)=SQRT(A(1))                                                     SYMV
        DO 100 IJ=2,N                                                       SYMV
  100   A(IJ)=A(IJ)/A(1)                                                    SYMV
        A(1)=1.0/A(1)                                                       SYMV
        IM1=1                                                               SYMV
        IJ=N                                                                SYMV
        DO 1000 I=2,N                                                       SYMV
        II=IJ+1                                                             SYMV
        IJ=II                                                               SYMV
        DO 200 J=I,N                                                        SYMV
        JM1=J-I                                                             SYMV
        LI=I                                                                SYMV
        LJ=J                                                                SYMV
        DO 120 L=1,IM1                                                      SYMV
        A(IJ)=A(IJ)-A(LI)*A(LJ)                                             SYMV
        LI=LI+N-L                                                           SYMV
  120   LJ=LI+JM1                                                           SYMV
  200   IJ=IJ+1                                                             SYMV
        IF(A(II).LT.0.0) GO TO 26
        A(II)=SQRT(A(II))                                                   SYMV
        JI=I                                                                SYMV
        JJ=1                                                                SYMV
        DO 500 J=1,IM1                                                      SYMV
        A(JI)=A(JJ)*A(II)                                                   SYMV
        IF(J-IM1)300,420,420                                                SYMV
  300   JP1=J+1                                                             SYMV
        JL=JJ                                                               SYMV
        LI=JI                                                               SYMV
        DO 400 L=JP1,IM1                                                    SYMV
        JL=JL+1                                                             SYMV
        LI=LI+N-L+1                                                         SYMV
  400   A(JI)=A(JI)+A(JL)*A(LI)                                             SYMV
  420   A(JI)=-A(JI)/A(II)                                                  SYMV
        JI=JI+N-J                                                           SYMV
```

```
 500 JJ=JJ+N-J+1                                                        SYMV
     IF(I-N)600,900,900                                                 SYMV
 600 IP1=I+1                                                            SYMV
     II=II
     DO 700 J=IP1,N                                                     SYMV
     IJ=IJ+1                                                            SYMV
 700 A(IJ)=A(IJ)/A(II)                                                  SYMV
 900 A(II)=1.0/A(II)                                                    SYMV
1000 IM1=I                                                              SYMV
     II=1                                                               SYMV
     DO 2000 I=1,N                                                      SYMV
     JJ=II                                                              SYMV
     IJ=II                                                              SYMV
     DO 1400 J=I,N                                                      SYMV
     A(IJ)=A(IJ)*A(JJ)                                                  SYMV
     JP1=J+1                                                            SYMV
     IF(JP1-N)1100,1100,1400                                            SYMV
1100 IL=IJ                                                              SYMV
     JL=JJ                                                              SYMV
     DO 1200 L=JP1,N                                                    SYMV
     IL=IL+1                                                            SYMV
     JL=JL+1                                                            SYMV
1200 A(IJ)=A(IJ)+A(IL)*A(JL)                                            SYMV
     JJ=JL+1                                                            SYMV
1400 IJ=IJ+1                                                            SYMV
2000 II=IJ                                                              SYMV
     CALL ICLOCK(IT2)
     TIME = FLOAT(IT2-IT1)/60000.
     WRITE(6,3000) TIME
3000 FORMAT(1H0 39HTHE TIME ELAPSED FOR MATRIX INVERSION = F12.4,1X,7HS
    1ECONDS)
     RETURN                                                             SYMV
     END
```

3.0 COMSYN ~ COMPONENT MODE SYNTHESIS PROGRAM

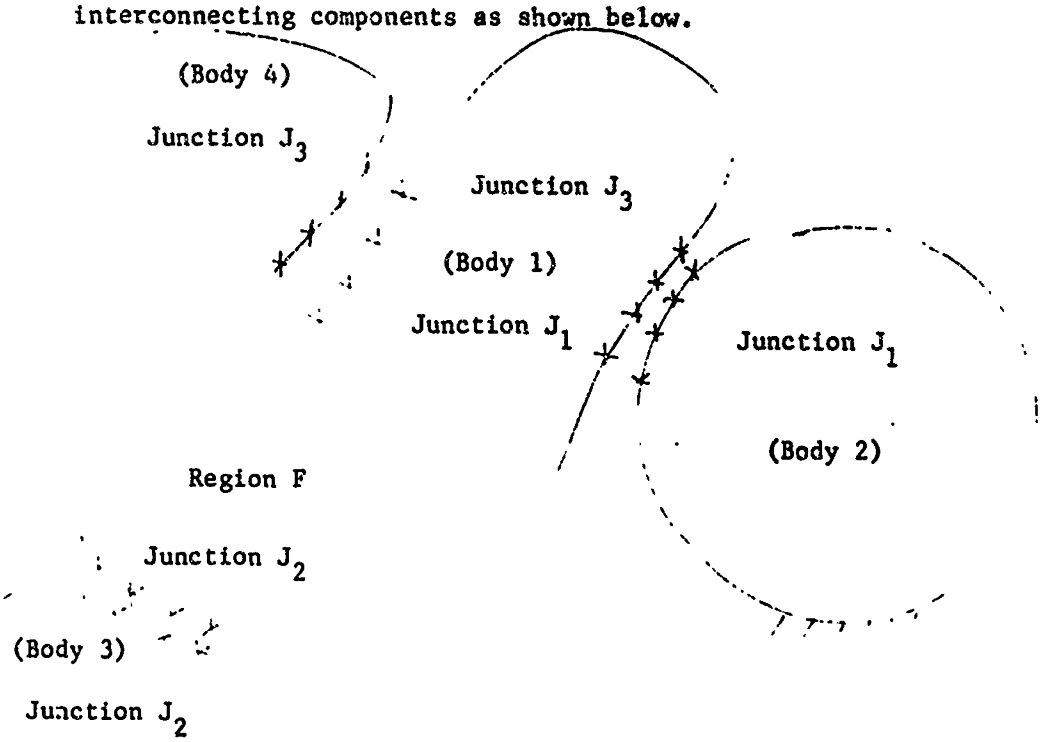
3.1 Introduction

The component mode synthesis technique provides the structural engineer with a valuable analytical tool for obtaining the dynamic response of large complex structures. The basic approach requires that the structure be divided into a number of smaller interconnected components each of which can be analyzed using a small number of degrees of freedom. The total system response is then obtained by coupling the component modal data. The principal advantage of the approach is that the order of the final system of equations to be solved is substantially smaller than the total number of degrees of freedom of the system. The order of the final system matrix depends on the number of component modes selected, and the number of common joints.

The program presented here is orientated for use in flutter analyses. As such, only planar structures may be analyzed. Consequently, only three degrees of freedom are used in the analysis; they are the translation normal to the plane of the structure and the two rotations in the plane of the structure. The program yields the natural vibration modes and frequencies for the composite structure, and generalized mass values for each mode. In addition, there is an option to calculate generalized aerodynamic forces when AICs are entered into the program. Note: When the generalized aerodynamic forces calculated by COMSYN are used in a flutter analysis, it is necessary to use the mode shapes and generalized masses calculated by COMSYN in order that the magnitude of all the parameters be consistent.

3.2 Theoretical Derivation

Assume that the structure under consideration is subdivided into interconnecting components as shown below.



Each component may be attached to one or more components by junction nodes. Junction nodes (common joints) are structural points that exist on two or more components. The components may also contain physically restrained nodes (boundary attach points), and nodes that are free to move. It is noted, that when dividing a structure into components, common joints cannot be boundary attach points.

The basic approach to the solution is as follows:

1. The structure is divided into components.

2. The stiffness and mass matrices are derived for each component with the common joints restrained; also a vibration analysis is performed for this condition.

3. The stiffness and mass matrices are derived for each component with the common joints free.

4. The absolute displacements of each component is expanded in terms of the fixed modes, and the rigid body and constraint modes. The fixed modes are calculated in Step 2 above. The rigid body and constraint modes are derived from information calculated in Step 3 above. These modes are defined as the displacements, X_F, in Region F, due to motion of the junction displacements, X_J's, individually, when no external forces are applied in Region F.

5. Using Step 4 a transformation is established that takes the component from physical coordinates (absolute displacements) to the system component mode coordinates (the generalized coordinates and junction displacements).

6. The transformation matrix derived in Step 5 is used to transform the component stiffness and mass matrices to the system component mode coordinates.

7. The dynamical matrix for the entire structure in system component mode coordinates is assembled by combining the component stiffness and mass matrices of Step 6.

8. A vibration analysis is performed for the entire structure in system component mode coordinates.

9. The system component modes are transformed to the absolute displacements which are the mode shapes for the entire structure.

10. Using the mode shapes the generalized mass and generalized aerodynamic forces are calculated.

Steps 1 through 3 are performed by the program FLUENC-100C. Steps 4 through 10 are performed by the program COMSYN.

The equilibrium equation for a typical component can be written in the form

$$\begin{Bmatrix} P_F \\ P_J \\ 0 \end{Bmatrix} = \begin{bmatrix} K_{FF} & K_{FJ} & K_{F\theta} \\ K_{JF} & K_{JJ} & K_{J\theta} \\ K_{\theta F} & K_{\theta J} & K_{\theta\theta} \end{bmatrix} \begin{Bmatrix} X_F \\ X_J \\ \theta \end{Bmatrix} \qquad (1)$$

where the matrices have been partitioned such that P_F and X_F refer to the forces and displacements in region F, P_J and X_J refer to the forces and displacements of the nodes in junctions J, and θ refers to the rotations in region F. If the common joint normal displacements are restrained, the above matrix equation can be written as

$$\begin{Bmatrix} P_F \\ 0 \\ P_J \end{Bmatrix} = \begin{bmatrix} K_{FF} & K_{F\theta} & K_{FJ} \\ K_{\theta F} & K_{\theta\theta} & K_{\theta J} \\ K_{JF} & K_{J\theta} & K_{JJ} \end{bmatrix} \begin{Bmatrix} X_F \\ \theta \\ 0 \end{Bmatrix} \qquad (2)$$

which is equivalent to the partitioned matrix equation

$$\begin{Bmatrix} P_F \\ 0 \end{Bmatrix} = \begin{bmatrix} K_{FF} & K_{F\theta} \\ K_{\theta F} & K_{\theta\theta} \end{bmatrix} \begin{Bmatrix} X_F \\ \theta \end{Bmatrix} \qquad (3a)$$

and

$$\begin{Bmatrix} P_J \end{Bmatrix} = \begin{bmatrix} K_{JF} & K_{J\theta} \end{bmatrix} \begin{Bmatrix} X_F \\ \theta \end{Bmatrix} \qquad (3b)$$

The second row of Eq (3a) can be written as

$$0 = K_{\theta F} X_F + K_{\theta\theta} \theta$$

$$\text{or} \quad \theta = -K_{\theta\theta}^{-1} K_{\theta F} X_F \tag{4}$$

Substituting back into the first row of Eq (2) yields

$$P_F = \left[K_{FF} - K_{F\theta} K_{\theta\theta}^{-1} K_{\theta F}\right] \left\{X_F\right\} \tag{5}$$

The reduced component stiffness matrix with joint nodes restrained is

$$\left[K\right]_R = \left[K_{FF} - K_{F\theta} K_{\theta\theta}^{-1} K_{\theta F}\right] \tag{6}$$

The component consistent mass matrix can be written as

$$\begin{bmatrix} M_{FF} & M_{FJ} & M_{F\theta} \\ M_{JF} & M_{JJ} & M_{J\theta} \\ M_{\theta F} & M_{\theta J} & M_{\theta\theta} \end{bmatrix} \tag{7}$$

If the common joint normal displacements are restrained, Eq (7) becomes

$$\begin{bmatrix} M_{FF} & M_{F\theta} \\ M_{\theta F} & M_{\theta\theta} \end{bmatrix} \tag{8}$$

Using Eq (4) we can write

$$\begin{Bmatrix} X_F \\ \theta \end{Bmatrix} = \begin{bmatrix} I \\ -K_{\theta\theta}^{-1} K_{\theta F} \end{bmatrix} \begin{Bmatrix} X_F \end{Bmatrix} \tag{9}$$

and the reduced mass matrix becomes

$$\left[M\right]_R = \left[R_1\right]^T \begin{bmatrix} M_{FF} & M_{F\theta} \\ M_{\theta F} & M_{\theta\theta} \end{bmatrix} \left[R_1\right] \qquad (10)$$

where

$$\left[R_1\right] = \begin{bmatrix} I \\ -K_{\theta\theta}^{-1} \; K_{\theta F} \end{bmatrix} \qquad (11)$$

In order to perform a component mode synthesis analysis, it is necessary to expand the absolute displacements of each component in terms of the fixed modes and the rigid body and constraint modes. The fixed modes, $\overline{X}$, are given in the form

$$\left\{\overline{X}_F\right\} = \left[\phi\right]\left\{q\right\} \qquad (12)$$

where $\left\{q\right\}$ are generalized coordinates and the modes $\left\{\phi\right\}$, stored columnwise in the modal matrix $\left[\phi\right]$ satisfy the following equations of motion

$$\left[K\right]_R\left\{y\right\} - \omega^2 \left[M\right]_R \left\{y\right\} = \left\{0\right\} \qquad (13)$$

Equation (13) gives the modes of the component with the junction nodes fixed.

The rigid body and constraint modes are determined simultaneously from Eq (3) by defining these modes as the displacements in region F, X_F, due to the motion of the junction displacements, X_J, when no external forces are applied in region F. This condition is represented by the equations

$$\begin{Bmatrix} 0 \\ P_J \\ 0 \end{Bmatrix} = \begin{bmatrix} K_{FF} & K_{FJ} & K_{F\theta} \\ K_{JF} & K_{JJ} & K_{J\theta} \\ K_{\theta F} & K_{\theta J} & K_{\theta\theta} \end{bmatrix} \begin{Bmatrix} X_F \\ X_J \\ \theta \end{Bmatrix} \qquad (14)$$

where X_J are the junction displacements including the rational degrees of freedom. The third equation of matrix Eq (13) yields

$$\begin{bmatrix} K_{\theta F} & K_{\theta J} \end{bmatrix} \begin{Bmatrix} X_F \\ X_J \end{Bmatrix} + \begin{bmatrix} K_{\theta\theta} \end{bmatrix} \begin{Bmatrix} \theta \end{Bmatrix} = \begin{Bmatrix} 0 \end{Bmatrix}$$

or $$\theta = - \begin{bmatrix} K_{\theta\theta} \end{bmatrix}^{-1} \begin{bmatrix} K_{\theta F} & K_{\theta C} \end{bmatrix} \begin{Bmatrix} X_F \\ X_J \end{Bmatrix} \tag{15}$$

Substituting into the first two equations of Eq (14) yields

$$\begin{Bmatrix} 0 \\ \\ P_J \end{Bmatrix} = \begin{bmatrix} (K_{FF} - K_{F\theta} K_{\theta\theta}^{-1} K_{\theta F}) & (K_{FJ} - K_{F\theta} K_{\theta\theta}^{-1} K_{\theta J}) \\ \\ (K_{JF} - K_{J\theta} K_{\theta\theta}^{-1} K_{\theta F}) & (K_{JJ} - K_{J\theta} K_{\theta\theta}^{-1} K_{\theta J}) \end{bmatrix} \begin{Bmatrix} X_F \\ \\ X_J \end{Bmatrix}$$

or $$\begin{Bmatrix} 0 \\ P_J \end{Bmatrix} = \begin{bmatrix} \bar{K}_{FF} & \bar{K}_{FJ} \\ \bar{K}_{JF} & \bar{K}_{JJ} \end{bmatrix} \begin{Bmatrix} \bar{\bar{X}}_F \\ X_J \end{Bmatrix} \tag{16}$$

it is noted $\begin{bmatrix} \bar{K}_{FF} \end{bmatrix} = \begin{bmatrix} K \end{bmatrix}_R$

where
$$\begin{aligned} \bar{K}_{FF} &= K_{FF} - K_{F\theta} \; K_{\theta\theta}^{-1} \; K_{\theta F} \\ \bar{K}_{FJ} &= K_{FJ} - K_{F\theta} \; K_{\theta\theta}^{-1} \; K_{\theta J} \\ \bar{K}_{JF} &= K_{JF} - K_{J\theta} \; K_{\theta\theta}^{-1} \; K_{\theta F} \\ \bar{K}_{JJ} &= K_{JJ} - K_{J\theta} \; K_{\theta\theta}^{-1} \; K_{\theta J} \end{aligned} \tag{17}$$

$\bar{\bar{X}}_F$ = displacements in region F due to motion of joint displacements

The first equation of Eq (16) yields

$$\left[\bar{K}_{FF}\right]\left\{\bar{\bar{X}}_F\right\} + \left[K_{FJ}\right]\left\{X_J\right\} = 0$$

or

$$\left\{\bar{\bar{X}}_F\right\} = -\left[K_{FF}\right]^{-1}\left[K_{FJ}\right]\left\{X_J\right\}$$

or

$$\left\{\bar{\bar{X}}_F\right\} = \left[T\right]\left\{X_J\right\} \tag{18}$$

where

$$\left[T\right] = -\left[K_{FF}\right]^{-1}\left[K_{FJ}\right]$$

The absolute displacements of each component may be written in the form

$$\left\{X\right\} = \left\{\bar{X}\right\} + \left\{\bar{\bar{X}}\right\} \tag{19}$$

or

$$\begin{Bmatrix} X_F \\ X_J \end{Bmatrix} = \begin{bmatrix} \phi & T \\ 0 & I \end{bmatrix} \begin{Bmatrix} q \\ X_J \end{Bmatrix} \tag{20}$$

Let

$$\left[\beta\right] = \begin{bmatrix} \phi & T \\ 0 & I \end{bmatrix} \tag{21}$$

which is the transformation matrix which takes the component from physical coordinates (absolute displacements) to system component mode coordinates (generalized coordinates and junction displacements). The transformation matrix is exact if the modal matrix $\left[\phi\right]$ is complete; however, for a component mode synthesis analysis the modal matrix is truncated to reduce the number of degrees of freedom. To this extent the $\left[\beta\right]$ matrix is approximate.

The $\left[\beta\right]$ matrix is now used to obtain the transformed stiffness in the following manner

$$\left[\bar{K}_C\right] = \left[\beta\right]^T \left[K_C\right] \left[\beta\right] \tag{22}$$

where

$$\left[K_C\right] = \begin{bmatrix} \bar{K}_{FF} & \bar{K}_{FJ} \\ \bar{K}_{JF} & \bar{K}_{JJ} \end{bmatrix}$$

The mass matrix for a typical component is given by Eq (7). Likewise, using the transformation matrix

$$\left[R_2\right] = \begin{bmatrix} I \\ \left[-K_{\theta\theta}\right]^{-1} \left[K_{\theta F} \quad K_{\theta J}\right] \end{bmatrix} \tag{23}$$

The rotational degrees of freedom can be eliminated in the free region

$$\left[M_C\right] = \left[R_2\right]^T \begin{bmatrix} M_{FF} & M_{FJ} & M_{F\theta} \\ M_{JF} & M_{JJ} & M_{J\theta} \\ M_{\theta F} & M_{\theta J} & M_{\theta\theta} \end{bmatrix} \left[R_2\right] \tag{24}$$

$$\left[M_C\right] = \begin{bmatrix} \bar{M}_{FF} & \bar{M}_{FJ} \\ \bar{M}_{JF} & \bar{M}_{JJ} \end{bmatrix}$$

The $\left[\beta\right]$ matrix can again be used to obtain the transformed mass matrix

$$\left[\bar{M}_C\right] = \left[\beta\right]^T \left[M_C\right] \left[\beta\right] \tag{25}$$

Combining the system component stiffness matrix (Eq 22) and system component mass matrix (Eq 25) for all the components, one finally forms an overall stiffness and mass matrix

$$\begin{bmatrix} K_{qq} & K_{qJ} & K_{q\theta} \\ K_{Jq} & K_{JJ} & K_{J\theta} \\ K_{\theta q} & K_{\theta J} & K_{\theta\theta} \end{bmatrix} \begin{Bmatrix} q \\ \omega_J \\ \theta_J \end{Bmatrix} = \omega_S^2 \begin{bmatrix} M_{qq} & M_{qJ} & M_{q\theta} \\ M_{Jq} & M_{JJ} & M_{J\theta} \\ M_{\theta q} & M_{\theta J} & M_{\theta\theta} \end{bmatrix} \begin{Bmatrix} q \\ \omega_J \\ \theta_J \end{Bmatrix} \tag{26}$$

or $$\left[K_S\right]\left\{X_S\right\} = \omega_S^2 \left[M_S\right]\left\{X_S\right\}$$

where q - component modes

ω_J - junction normal displacements

θ_J - junction rotations

The system stiffness and mass matrices can be reduced by eliminating the joint rotational degrees of freedom. This can be done by using the transformation matrix

$$\left[R_3\right] = \begin{bmatrix} [I] \\ -\left[K_{\theta\theta}\right]^{-1}\left[K_{\theta q} \quad K_{\theta J}\right] \end{bmatrix} \tag{27}$$

in the following way

$$\left[\bar{K}_S\right] = \left[R_3\right]^T \left[K_S\right] \left[R_3\right] \tag{28}$$

$$\left[\bar{M}_S\right] = \left[R_3\right]^T \left[M_S\right] \left[R_3\right] \tag{29}$$

and the frequencies and mode shapes $\left[\gamma\right]$ of the entire structural system is given by

$$\left[\bar{K}_S\right] \left\{X_S\right\} = \omega_S^{\,2} \left[\bar{M}_S\right] \left\{X_S\right\}$$

where

$$\left\{X_S\right\} = \begin{Bmatrix} q \\ \omega_J \end{Bmatrix}$$

The deflections in each component can be calculated by using $\left\{X_S\right\}$ in conjunction with Eq (12) for each component.

Generalized Mass and Aerodynamic Forces

The generalized mass matrix is formed by utilizing the system modal matrix $\left[\bar{\gamma}\right]$, which is composed of the system modal shapes $\left\{X_S\right\}$ stored columnwise, as a transformation matrix. This can be written in matrix form as follows

$$\left[M\right]_G = \left[\bar{\gamma}\right]^T \left[\bar{M}_S\right] \left[\bar{\gamma}\right] \tag{30}$$

The aerodynamic forces are computed from the AIC matrix. For the uncoupled aero case the AIC matrix is in the form

$$\left[C_h\right] = \begin{bmatrix} \left[C_{h_1}\right] & & & 0 \\ & \left[C_{h_2}\right] & & \\ & & \left[C_{h_3}\right] & \\ 0 & & & \ddots \end{bmatrix} \tag{31}$$

where the diagonal terms are the AIC matrices for each component. The displacements $\left\{X_F\right\}$ must be computed for each component by using Eq (19) and the system modal matrix $\left[\bar{\gamma}\right]$. The modal matrix $\left[\gamma\right]$ was reduced by eliminating any unneccessary degrees of freedom when X_F was determined for an individual component. The $\left[X_F\right]$ for a particular component is computed from

$$\left[X_F\right]_i = \left[\phi\right] \left[q\right] + \left[T\right] \left[X_S\right]_i \tag{32}$$

where $\left[q\right]$ is the upper partition of the reduced modal matrix and $\left[X_S\right]_i$ is the junction displacement (translations and rotations) matrix for component i. Once $\left[X_F\right]_i$ is known for a particular component the equation

$$\left[Q\right]_{unc} = \sum_{i=1}^{N} \left[X_F\right]_i \left[C_{h_i}\right] \left[X_F\right]_i \tag{33}$$

yields the aerodynamic forces for N components. If the AIC matrix is coupled between components a and b, then

$$\left[C_h\right]_{a,b} = \begin{bmatrix} C_{h_{aa}} & C_{h_{ab}} \\ C_{h_{ba}} & C_{h_{bb}} \end{bmatrix} \tag{34}$$

then for a two component system the generalized forces are computed from the matrix equation

$$\begin{aligned} \begin{bmatrix} Q \end{bmatrix}_{coupled} = & \begin{bmatrix} X_{F_a} \end{bmatrix}^T \begin{bmatrix} C_{h_{aa}} \end{bmatrix} \begin{bmatrix} X_{F_a} \end{bmatrix} \\ & + \begin{bmatrix} X_{F_a} \end{bmatrix}^T \begin{bmatrix} C_{h_{ab}} \end{bmatrix} \begin{bmatrix} X_{F_b} \end{bmatrix} \\ & + \begin{bmatrix} X_{F_b} \end{bmatrix}^T \begin{bmatrix} C_{h_{ba}} \end{bmatrix} \begin{bmatrix} X_{F_a} \end{bmatrix} + \begin{bmatrix} X_{F_b} \end{bmatrix}^T \begin{bmatrix} C_{h_{bb}} \end{bmatrix} \begin{bmatrix} X_{F_b} \end{bmatrix} \end{aligned} \tag{35}$$

The final aerodynamic force matrix is obtained by adding $\begin{bmatrix} Q \end{bmatrix}$ uncoupled and $\begin{bmatrix} Q \end{bmatrix}$ coupled

$$\begin{bmatrix} Q \end{bmatrix} = \begin{bmatrix} Q \end{bmatrix}_{\substack{uncoupled \\ components}} + \begin{bmatrix} Q \end{bmatrix}_{\substack{coupled \\ components}} \tag{36}$$

The aerodynamic generalized forces as calculated have not been multiplied by any nondimensionalizing factors used in calculating $\begin{bmatrix} C_h \end{bmatrix}$.

3.3 Program Description

The computer program COMSYN written in Fortran IV performs a vibration analysis for a planar structural system using the component mode synthesis technique. A complex structure can be divided into as many as five components; each component may have as many as 12 common joints. The order of the system eigenvalue solution is determined by the number of components times the number of component modes plus the number of unique common joints times three. Common joints may exist in two or more components; however, they are counted once in determining the order of the system eigenvalue solution. The input requirements of COMSYN are a modal representation of each component and related data concerning the junction points (common joints). This input data is available as punched output from the FLUENC-100C program.

The program as such is oriented to facilitate flutter analyses by the normal mode (modal) method. A modal flutter analysis requires the vibration frequencies, the generalized masses, and the generalized aerodynamic forces. The vibration frequencies and generalized masses are the normal output of a component mode synthesis vibration analysis. A subroutine was added to the program to calculate generalized aerodynamic forces when aerodynamic influence coefficients are supplied. The aerodynamic influence coefficients may be uncoupled for each component or any two of the five components may be coupled. The only restriction being that the two coupled components must be entered serially as input data.

3.3.1 Processing Information

A. Operation -- Standard FORTRAN IV processor system. Operable on the GE635 computer.

B. Core Storage -- The program COMSY requires a minimum of 65,000 memory units for execution.

C. Tape Units -- Standard input, output, and punch tape units, and 11 scratch tape units.

3.4 Input Instructions

1. Title Cards, Format (12A6) Two cards required.

Column	1 - - - - - - - - - - 72
Name	Any Alphanumeric Statement

Column	1 - - - - - - - - - - 72
Name	Any Alphanumeric Statement

2. Control Card, Format (6I5) NCOUP = NAT = NV = 0 for vibration analysis only.

Column	1-5	6-10	11-15	16-20	21-25	26-30
Name	NCOMP	MODE	NCOM	NCOUP	NAT	NV

NCOMP = Number of components used in the analysis ≤ 5.

MODE = Number of Modes requested for total system ≤ 9.

NCOM = Number of rows in the matrix that relates the common joints between two components (See instruction No. 6) ≤ 48.

NCOUP = 0, no aerodynamic coupling exists between components.

= The lowest number of the two components which are aerodynamically coupled (coupled components must be in sequence).

NAT = 1, AICs are entered as non-zero partitions (strip or piston theories).

= 2 AICs are full matrices (kernel function or Mach box)

NV = Number of reduced velocities (1/k's)

NOTE: When NAT=1 then it is required that NCOUP=0; and when NAT=2 then it is required that NCOUP $\geq 0 < 5$

3. Information Card, Format (9I5)

Column	1-5	6-10	11-15	
	NREDU(1)	NREDU(2)	NREDU(3)	NREDU(i)

NREDU(i) = Number of translational degrees of freedom for component "i" when the common joints are restrained.

4. Information Card, Format (9I5)

Column	1-5	6-10	11-15	
Name	NMODE(i)	NMODE(2)	NMODE(3)	NMODE(i)

NMODE(i) = Number of modes for component "i" used in the analysis.

5. Information Card, Format (9I5)

Column	1-5	6-10	11-15		
Name	NCJT(1)	NCJT(2)	NCJT(3)	...	NCJT(i)

NCJT(i) = Number of common joints for component "i" used in the analysis ≤ 12.

6. Information Card, Format (36I2) enter data continuously used as many cards as required. This card/series of cards describes a correlation matrix that relates the common joints of each component to the overall structure. Four numbers describe each row of the matrix as follows:

Common Joint No. of Component No. is the same as Common Joint No. of Component No.
1st No. 2nd No. 3rd No. 4th No.

EXAMPLE: If the third junction point of Component 1 is common to the fifth junction point of Component 2, then the set of four numbers are:

Column	1	2	3	4	5	6	7	8
		3		1		5		2

Restrictions

1. Correlation must always be made to a lower numbered component;i.e., in each set, the 2nd number must always be smaller than the 4th number.

2. If a joint is common to more than two components, all possible correlations must be made to the lowest numbered component in which it appears. Additional correlations may be made to higher numbered components but are unnecessary.

7. Information Card, Format (6E12.8). If NV = 0 omit this card/s.

Column	1-12	13-24	25-36		
Name	VEL(1)	VEL(2)	VEL(3)	...	VEL(NV)

VEL(i) = Reduced velocity series, i = 1, NV

NV $\leq$ 20 (continue on next card if necessary)

8. The following input is available as punched output from the program FLUENC or can be derived from any other structural analysis computer program. The output from FLUENC is punched in Format (1P6E12.5), but any card than can be read in Format (6E12.8) can be used. All data is presented as full matrices, and each row begins on a new card.

The data is stacked in the following order by component:

1. Mode Shapes
2. Frequencies, Hz
3. CKFF - Stiffness Matrix
4. CKFJ
5. CKJJ
6. Flexibility Matrix
7. Weight Matrix
8. CMFJ
9. CMJJ

9. When generalized aerodynamic forces are desired, NV > 0, aerodynamic influence coefficients, AICs, are required input. The AIC matrix or partition is entered by row; each row begins on a new card. Format (6E12.8) is used.

When NAT = 1 the AICs are entered as non-zero partitions. The punched output from the computer programs STRIP and PISTON (Vol.I) are compatible input for this option. Repeat the following information for each component.

9a. Control Card, Format (2I4)

Column	1-4	5-8	
Name	NSIZE	NPART	

NSIZE = Size of complete AIC matrix for component "i"

NPART = Number of non-zero partitions for component "i" (number of strips)

Repeat the following instructions NPART times

9b. Control Card, Format (I4)

Column	1-4	
Name	NS	

NS = Size of partition

9c. Data Card Format (6E12.8) Start each row on a new card.

Column	1-12	13-24	25-36	37-48	
Name	$ARe_{1,1}$	$AIm_{1,1}$	$ARe_{1,2}$	$AIm_{1,2}$	

ARe - Real part of $AIC_{i,j}$

AIm - Imaginary part of $AIC_{i,j}$

Repeat data items 9a, 9b, and 9c for each 1/k value.

When NAT = 2, the AICs are entered as full matrices for each component. The punched output from the AIC computer programs subsonic, sonic, and supersonic (Vol. III, Ref. 1) are compatible input for this option. Stack the matrices for each component sequentially as they are entered in Items 4 through 8. Enter each matrix as follows:

9d. Data Card Format (6E12.5) Start each row on a new card.

Column	1-12	13-24	25-36	37-48
Name	$ARe_{1,1}$	$AIm_{1,1}$	$ARe_{1,2}$	$AIm_{1,2}$

ARe ~ Real part of $AIC_{i,j}$

AIm ~ Imaginary part of $AIC_{i,j}$

When two components are coupled, the coupled matrix is inserted in its proper sequence in the stack of AIC matrices. The maximum size for any AIC matrix is 40x80.

Repeat data item 9d for each 1/k value.

NOTE: For all input, reference to the common joints, free joints, and components must be consistent between the structural analysis program and the aerodynamic influence coefficient programs. The sequence order of the above must be maintained when input into COMSYN.

3.5 DESCRIPTION OF PROGRAM OUTPUT

I. Printed Output

A. For each component

1. All input data except flexibility matrix
2. Transformed stiffness and mass matrices (component mode coordinates - generalized coordinates and junction displacements)
3. Normalized mode shapes (for orthogonality)
4. Mass matrix for orthogonality check (diagonal elements = 1.0)

B. Results for the total system

1. Relative locations of the degrees of freedom of each component when combined to form the system matrix by the NCODE METHOD.
2. Reduced stiffness and mass matrices (rotational degrees of freedom eliminated)
3. Eigenvalues and eigenvectors
4. Natural frequencies (CPS)
5. Mode shapes representing free joints on each component - printed columnwise.
6. Generalized mass matrix
7. Generalized aerodynamic forces for each 1/k if NV > 0.

II. Punched Output

A. Generalized mass matrix, GENM

B. Generalized aerodynamic force matrix for each 1/k, GENA

In both cases full matrices are punched out. Format (1P6E12.5) is used. The cards, which are sequenced and indentified with the names given above, are compatible with the input requirements for MOFA, the Modal Flutter Analysis Program.

3.6 SAMPLE PROBLEM

The sample problem of Section 2.5 will be used to demonstrate COMSYM. The structure is divided into three components, the fuselage, the wing, and the control surface. There are three common joints: two attach the wing to the fuselage and one attaches the control surface to the fuselage. The punched output from FLUENC-100C is used as input to the program.

COMPONENT MODE SYNTHESIS ANALYSIS

3 COMPONENTS CONSIDERED

TYPICAL MISSILE

COMSYN CHECK CASE 1

INPUT DATA FOR COMPONENT 1

13 DEGREES OF FREEDOM
7 MODES
3 COMMON JOINTS

MODE SHAPES

MODE 1 - FREQUENCY = 67.697 CPS

1.00000E 00	4.09777E-01	2.24870E-01	5.28821E-02	1.40428E-21	2.63921E-21	1.52588E-21	2.58298E-21
4.85164E-21	2.80639E-21	9.79013E-19	3.38824E-18	6.47293E-18			

MODE 2 - FREQUENCY = 310.743 CPS

-9.01140E-01	8.17430E-01	1.00000E 00	4.05314E-01	3.27313E-20	6.14870E-20	3.54938E-20	6.11248E-20
1.14776E-19	6.62411E-20	9.24783E-17	3.19965E-16	6.11126E-16			

MODE 3 - FREQUENCY = 429.813 CPS

-3.94167E-15	2.96710E-15	3.97479E-15	1.69699E-15	4.91140E-19	9.22218E-19	5.31596E-19	9.31452E-19
1.74785E-18	1.00659E-18	1.51349E-01	5.23604E-01	1.00000E 00			

MODE 4 - FREQUENCY = 932.061 CPS

-3.03670E-01	1.00000E 00	-2.84708E-01	-6.43529E-01	-1.06615E-18	-1.99517E-18	-1.13784E-18	-2.32725E-18
-4.34504E-18	-2.46343E-18	-3.78158E-17	-1.31628E-16	-2.52580E-16			

MODE 5 - FREQUENCY = 1669.060 CPS

-1.44262E-01	7.91290E-01	-7.93726E-01	1.00000E 00	-4.54991E-18	-8.44006E-18	-4.69242E-18	-2.28876E-17
-4.22278E-17	-2.31778E-17	-3.80695E-18	-1.50044E-17	-3.13932E-17			

MODE 6 - FREQUENCY = 2063.090 CPS

1.82859E-15	-1.86274E-15	-3.46040E-17	3.98753E-15	2.33168E-15	4.30228E-15	2.35918E-15	5.44044E-01
1.00000E 00	5.44044E-01	1.42494E-16	8.97373E-16	2.30905E-15			

MODE 7 - FREQUENCY = 2527.250 CPS

7.51469E-16	-3.76243E-16	3.03547E-16	9.01854E-16	5.43863E-01	1.00000E 00	5.43863E-01	1.95646E-15
3.64100E-15	2.02904E-15	-1.27880E-16	1.00389E-16	1.00667E-16			

UPPER TRIANGLE OF REDUCED STIFFNESS MATRIX FOR COMPONENT (COMMON JOINTS RESTRAINED) K-FF

```
ROW  1
     1.97531E 05   -5.75309E 05   4.88889E 05   -1.55556E 05   0.            0.            0.            0.
     0.            0.             0.            0.             0.

ROW  2
     2.25309E 06   -2.78889E 06   1.55556E 06   0.             0.            0.            0.            0.
     0.            0.             0.            0.

ROW  3
     5.20000E 06   -4.90000E 06   0.            0.             0.            0.            0.            0.
     0.            0.             0.

ROW  4
     9.10000E 06   0.             0.            0.             0.            0.            0.            0.
     0.            0.

ROW  5
     1.05600E 08   -6.72000E 07   2.38000E 07   0.             0.            0.            0.            0.
     0.

ROW  6
     8.40000E 07   -6.72000E 07   0.            0.             0.            0.            0.            0.

ROW  7
     1.05600E 08   0.             0.            0.             0.            0.            0.

ROW  8
     1.05600E 08   -6.72000E 07   2.38000E 07   0.             0.            0.

ROW  9
     8.40000E 07   -6.72000E 07   0.            0.             0.

ROW 10
     1.05600E 08   0.             0.            0.

ROW 11
     5.16923E 07   -2.97231E 07   7.75385E 06

ROW 12
     2.84308E 07   -1.03385E 07

ROW 13
     4.52308E 06


STIFFNESS MATRIX - RELATES COMMON JOINTS TO FREE JOINTS (COMMON JOINTS FREE)   K-FJ

ROW  1
     4.44444E 04   0.             0.            0.             0.            0.            1.48148E 05   0.
     0.

ROW  2
     -4.44444E 04  0.             0.            0.             0.            0.            -1.48148E 06  0.
     0.

ROW  3
```

```
   2.00000E 06   0.            0.            0.            0.            0.            6.66667E 06   0.
   0.

ROW  4
  -5.60000E 06   0.            0.            0.            0.            0.           -3.26667E 07   0.
   0.

ROW  5
  -6.00000E 07  -7.20000E 06   0.            0.            0.            0.            1.56000E 08  -1.20000E 07
   0.

ROW  6
   2.52000E 07   2.52000E 07   0.            0.            0.            0.           -4.20000E 07   4.20000E 07
   0.

ROW  7
  -7.20000E 06  -6.00000E 07   0.            0.            0.            0.            1.20000E 07  -1.56000E 08
   0.

ROW  8
   0.           -6.00000E 07  -7.20000E 06   0.            0.            0.            0.            1.56000E 08
  -1.20000E 07

ROW  9
   0.            2.52000E 07   2.52000E 07   0.            0.            0.            0.           -4.20000E 07
   4.20000E 07

ROW 10
   0.           -7.20000E 06  -6.00000E 07   0.            0.            0.            0.            1.20000E 07
  -1.56000E 08

ROW 11
   0.            0.           -2.97231E 07   0.            0.            0.            0.            0.
   7.75385E 07

ROW 12
   0.            0.            1.16308E 07   0.            0.            0.            0.            0.
  -1.93846E 07

ROW 13
   0.            0.           -1.93846E 06   0.            0.            0.            0.            0.
   3.23077E 06


UPPER TRIANGLE OF STIFFNESS MATRIX - COMMON JOINTS   K-JJ

ROW  1
   4.42000E 07   1.80000E 06   0.            0.            0.            0.           -9.56667E 07   3.00000E 06
   0.

ROW  2
   8.04000E 07   1.80000E 06   0.            0.            0.           -3.00000E 06  -5.82484E-01   3.00000E 06

ROW  3
   6.02308E 07   0.            0.            0.            0.           -3.00000E 06   6.16154E 07

ROW  4
   1.61538E 08   0.            0.            0.            0.            0.
```

```
ROW  5
    2.15385E 08   0.          0.          0.          0.

ROW  6
    1.61538E 08   0.          0.          0.

ROW  7
    7.16111E 08  -5.00000E 06  0.

ROW  8
    9.70000E 08  -5.00000E 06

ROW  9
    7.27308E 08

UPPER TRIANGLE OF TRANSFORMED STIFFNESS MATRIX FOR COMPONENT 1

ROW  1
    1.80927E 05   0.          0.          0.          0.          0.          0.          0.
    0.            0.          0.          0.          0.          0.          0.          0.

ROW  2
    3.81208E 06   0.          0.          0.          0.          0.          0.          0.
    0.            0.          0.          0.          0.          0.          0.

ROW  3
    7.29321E 06   0.          0.          0.          0.          0.          0.          0.
    0.            0.          0.          0.          0.          0.

ROW  4
    3.42964E 07   0.          0.          0.          0.          0.          0.          0.
    0.            0.          0.          0.          0.

ROW  5
    1.09977E 08   0.          0.          0.          0.          0.          0.          0.
    0.            0.          0.          0.

ROW  6
    1.68034E 08   0.          0.          0.          0.          0.          0.          0.
    0.            0.          0.

ROW  7
    2.52108E 08   0.          0.          0.          0.          0.          0.          0.
    0.            0.

ROW  8
    1.04988E 06  -1.05010E 06  0.          0.          0.          0.         -1.05005E 07 -1.05003E 07
    0.

ROW  9
    2.09990E 06  -1.05010E 06  0.          0.          0.          1.05003E 07  1.06925E 00 -1.05003E 07

ROW 10
    1.04986E 06   0.          0.          0.          0.          1.05003E 07  1.05005E 07

ROW 11
```

```
   1.61538E 08   0.          0.          0.          0.

ROW 12
   2.15385E 08   0.          0.          0.          0.

ROW 13
   1.61538E 08   0.          0.          0.

ROW 14
   1.39997E 08   7.00007E 07   0.

ROW 15
   2.80000E 08   7.00007E 07

ROW 16
   1.39998E 08

UPPER TRIANGLE OF REDUCED WEIGHT MATRIX FOR COMPONENT (COMMON JOINTS RESTRAINED)   M-FF

ROW  1
   2.50000E 01   0.          0.          0.          0.          0.          0.          0.
   0.            0.          0.          0.          0.

ROW  2
   2.50000E 01   0.          0.          0.          0.          0.          0.          0.
   0.            0.          0.          0.

ROW  3
   5.00000E 01   0.          0.          0.          0.          0.          0.          0.
   0.            0.          0.

ROW  4
   5.00000E 01   0.          0.          0.          0.          0.          0.          0.
   0.            0.

ROW  5
   1.66000E 01   0.          0.          0.          0.          0.          0.          0.
   0.

ROW  6
   1.67000E 01   0.          0.          0.          0.          0.          0.          0.

ROW  7
   1.66000E 01   0.          0.          0.          0.          0.          0.

ROW  8
   2.50000E 01   0.          0.          0.          0.          0.

ROW  9
   2.50000E 01   0.          0.          0.          0.

ROW 10
   2.50000E 01   0.          0.          0.

ROW 11
   5.00000E 00   0.          0.
```

ROW 12
5.00000E 00 0.

ROW 13
1.50000E 01

MASS MATRIX - RELATES COMMON JOINTS TO FREE JOINTS (COMMON JOINTS FREE) M-FJ

ROW 1
0. 0. 0. 0. 0. 0. 0. 0.
0.

ROW 2
0. 0. 0. 0. 0. 0. 0. 0.
0.

ROW 3
0. 0. 0. 0. 0. 0. 0.
0.

ROW 4
0. 0. 0. 0. 0. 0. 0. 0.
0.

ROW 5
0. 0. 0. 0. 0. 0. 0. 0.
0.

ROW 6
0. 0. 0. 0. 0. 0. 0. 0.
0.

ROW 7
0. 0. 0. 0. 0. 0. 0. 0.
0.

ROW 8
0. 0. 0. 0. 0. 0. 0. 0.
0.

ROW 9
0. 0. 0. 0. 0. 0. 0. 0.
0.

ROW 10
0. 0. 0. 0. 0. 0. 0. 0.
0.

ROW 11
0. 0. 0. 0. 0. 0. 0. 0.
0.

ROW 12
0. 0. 0. 0. 0. 0. 0. 0.
0.

ROW 13

```
   0.        0.        0.        0.        0.        0.        0.        0.
   0.

UPPER TRIANGLE OF MASS MATRIX - COMMON JOINTS   M-JJ

ROW  1
   1.28209E-01   0.        0.        0.        0.        0.        0.        0.
   0.
ROW  2
   1.92961E-01   0.        0.        0.        0.        0.        0.        0.
ROW  3
   1.92961E-01   0.        0.        0.        0.        0.        0.
ROW  4
   0.            0.        0.        0.        0.        0.
ROW  5
   0.            0.        0.        0.        0.
ROW  6
   0.            0.        0.        0.
ROW  7
   0.            0.        0.
ROW  8
   0.            0.
ROW  9
   0.

NORMALIZED MODE SHAPES FOR ORTHOGONALITY

MODE 1 - FREQUENCY =     67.697 CPS

   3.37414E 00   1.68632E 00   7.58742E-01   1.78431E-01   4.73823E-21   8.98586E-21   5.14826E-21   8.71282E-21
   1.63701E-20   9.46915E-21   3.38332E-18   1.14324E-17   2.18406E-17

MODE 2 - FREQUENCY =    310.743 CPS

  -1.81456E 00   1.64460E 00   2.01363E 00   8.16151E-01   6.59086E-20   1.23812E-19   7.14712E-20   1.23882E-19
   2.31116E-19   1.33345E-19   1.86217E-16   6.44290E-16   1.23058E-15

MODE 3 - FREQUENCY =    420.613 CPS

  -1.90754E-14   1.44591E-14   1.92357E-14   8.21246E-15   2.37684E-18   4.46301E-18   2.57262E-18   4.50770E-18
   8.45860E-18   4.87132E-18   7.32443E-01   2.53394E 00   4.83943E 00

MODE 4 - FREQUENCY =    932.861 CPS

  -9.26938E-01   2.72315E 00  -7.75302E-01  -1.75242E 00  -2.90328E-18  -5.43314E-18  -3.09851E-18  -6.33744E-18
  -1.18372E-17  -6.70828E-18  -1.02978E-16  -3.58421E-16  -6.87812E-16
```

```
MODE 5 - FREQUENCY =     1669.060 CPS

   -2.86818E-01   1.57322E 00  -1.57806E 00   1.08817E 00  -9.04601E-18  -1.67803E-17  -9.32934E-18  -4.55845E-17
   -8.39541E-17  -4.60815E-17  -7.56887E-18  -2.08313E-17  -6.24151E-17

MODE 6 - FREQUENCY =     2063.090 CPS

    5.69538E-15  -5.80174E-15  -1.07779E-16   1.24197E-14   7.26231E-15   1.34000E-14   7.34797E-15   1.69449E 00
    3.11463E 00   1.69449E 00   4.43816E-16   2.79498E-15   7.19183E-15

MODE 7 - FREQUENCY =     2527.050 CPS

    2.86726E-15  -3.34334E-15   1.50159E-15   3.44105E-15   2.07513E 00   3.81553E 00   2.07513E 00   7.46494E-15
    1.38958E-14   7.73806E-15  -4.87961E-16   4.17377E-16   3.84098E-15


CHECK FOR ORTHOGONALITY

UPPER TRIANGLE OF THE GENERALIZED MASS MATRIX IS NOW

ROW  1
    1.00000E 00   1.89236E-07  -5.15354E-16   5.72974E-08   1.79013E-07   8.87237E-16   4.88442E-16

ROW  2
    1.00000E 00   9.91016E-15   2.00079E-07   1.19888E-07  -2.88159E-18   6.20782E-17

ROW  3
    1.00000E 00  -3.83966E-16  -1.23738E-17   1.45086E-15   7.32420E-16

ROW  4
    1.00000E 00   1.16364E-07  -4.13957E-15  -1.67619E-15

ROW  5
    1.00000E 00   2.49602E-15   1.60874E-16

ROW  6
    1.00000E 00   7.98564E-15

ROW  7
    1.00000E 00


UPPER TRIANGLE OF TRANSFORMED MASS MATRIX FOR COMPONENT 1

ROW  1
    1.00000E 00   0.            0.            0.            0.            0.            0.            4.49026E-01
    1.51297E-21   1.04942E-18   0.            0.            0.            1.53037E 01  -3.03492E-21  -1.44175E-17

ROW  2
    1.00000E 00   0.            0.            0.            0.            0.            3.55549E-01   2.12700E-20
    5.85765E-17   0.            0.            0.            4.18249E 00  -4.32377E-20  -8.12568E-16

ROW  3
    1.00000E 00   0.            0.            0.            0.            3.24943E-15   7.75236E-19   2.30301E-01
    0.            0.            0.            3.27670E-14  -1.59640E-18  -3.19585E 00
```

```
ROW  4
    1.00000E 00   0.             0.            0.            -2.04566E-01   -1.04661E-18   -3.35089E-17   0.
    0.            0.            -1.39723E 00   2.39754E-18    4.50398E-16

ROW  5
    1.00000E 00   0.             0.            1.36416E-01   -6.43265E-18   -8.60523E-18    0.             0.
    0.            7.07828E-01    2.13680E-17   1.59815E-17

ROW  6
    1.00000E 00   0.             2.19018E-15   2.10561E-01    2.10561E-01    0.             0.             0.
    1.85067E-14  -9.15654E-01    9.15654E-01

ROW  7
    1.00000E 00   1.71740E-01    1.71740E-01   1.09648E-15    0.             0.             0.            -7.47175E-01
    7.47175E-01   1.85097E-15

ROW  8
    5.59177E-01   2.21504E-02    0.            0.             0.             0.             8.57883E 00    1.06973E-01
    0.

ROW  9
    2.99300E-01   3.32615E-02    0.            0.             0.            -1.06973E-01   -8.16888E-02    1.60616E-01

ROW 10
    3.21572E-01   0.             0.            0.             0.            -1.60616E-01   -5.32884E-01

ROW 11
    0.            0.             0.            0.             0.             0.

ROW 12
    0.            0.             0.            0.             0.

ROW 13
    0.            0.             0.            0.

ROW 14
    2.54798E 02  -4.97074E-01    0.

ROW 15
    1.62204E 00  -7.46168E-01

ROW 16
    1.13341E 01
```

```
                              INPUT DATA FOR COMPONENT 2

                              12 DEGREES OF FREEDOM
                               7 MODES
                               2 COMMON JOINTS

MODE SHAPES

MODE 1 - FREQUENCY =     126.157 CPS

   3.60023E-02   2.46591E-02   3.45409E-02   2.73388E-01   2.82589E-01   2.62472E-01   6.19440E-01   6.89906E-01
   5.96171E-01   1.00000E 00   9.81827E-01   9.54697E-01

MODE 2 - FREQUENCY =     207.478 CPS

  -3.78869E-02  -7.32831E-04   3.79890E-02  -2.78665E-01   7.86185E-04   2.81802E-01  -6.20722E-01   5.43402E-03
   6.32633E-01  -9.71042E-01   1.3988E-02    1.00000E 00

MODE 3 - FREQUENCY =     823.410 CPS

   2.41634E-01   5.22065E-01   1.99279E-01   9.99413E-01   1.00000E 00   8.17415E-03   6.29004E-01   5.67109E-01
   5.17529E-01  -8.50269E-01  -7.69006E-01  -6.80419E-01

MODE 4 - FREQUENCY =     954.061 CPS

  -1.96793E-01   3.06234E-02   2.24483E-01  -8.83013E-01   6.63081E-02   1.00000E 00  -5.75936E-01   4.26279E-02
   6.52360E-01   7.63008E-01  -4.95302E-02  -8.66648E-01

MODE 5 - FREQUENCY =    1553.690 CPS

   6.37686E-02   1.00000E 00   5.44835E-02  -6.77355E-02   7.41860E-02  -7.41683E-02  -2.10200E-01  -2.33321E-01
  -1.94885E-01   1.52293E-01   1.41241E-01   1.43256E-01

MODE 6 - FREQUENCY =    2511.320 CPS

   4.32600E-01  -7.75638E-02   1.32571E-01   1.00000E 00  -8.52817E-01   5.13808E-01  -2.75323E-01  -9.36194E-01
   3.83021E-01   2.74410E-01   8.95907E-02  -1.46114E-02

MODE 7 - FREQUENCY =    2562.280 CPS

  -3.16680E-01   7.69478E-02   4.12077E-01  -4.17428E-01  -4.70103E-01   7.80171E-01   1.00000E 00  -2.95102E-01
  -6.03074E-01  -3.43330E-01  -3.89376E-02   3.49338E-01


UPPER TRIANGLE OF REDUCED STIFFNESS MATRIX FOR COMPONENT (COMMON JOINTS RESTRAINED)   K-FF

ROW  1
   7.19287E 05  -4.00747E 04   1.91324E 04  -2.01988E 05   8.39914E 02  -4.03173E 02   6.49635E 04   1.77001E 02
   8.77766E 01  -1.08793E 04   4.03252E 01  -3.12795E 01

ROW  2
   9.66117E 04  -3.81519E 04   1.33528E 05  -4.11915E 04   1.63782E 03   1.71266E 02   2.51283E 04  -7.52511E 01
  -2.25039E 01  -4.20366E 03   2.32634E 01
```

```
ROW  3
   7.61415E 05  -5.39768E 01  -4.04133E 02  -2.14028E 05  -5.34760E 01   6.99542E 02   6.89133E 04   1.88368E 01
  -1.23204E 02  -1.14385E 04

ROW  4
   1.84366E 05  -5.10494E 04   2.42436E 04  -1.01075E 05   2.29922E 03  -3.42962E 02   2.67977E 04  -4.37147E 02
   7.45661E 01

ROW  5
   2.02939E 05  -5.06656E 04   1.60505E 03  -9.14928E 04   2.01162E 03   2.54565E 01   2.52710E 04  -1.97542E 02

ROW  6
   1.93765E 05  -7.77366E 01   1.59706E 03  -1.07054E 05  -6.54436E 01   1.04716E 01   2.82717E 04

ROW  7
   1.36388E 05  -6.91289E 04   3.26452E 04  -3.95690E 04   3.41291E 03  -4.67244E 02

ROW  8
   2.37957E 05  -6.89752E 04   2.74167E 03  -4.31669E 04   2.96646E 03

ROW  9
   1.42239E 05  -1.07932E 02   2.55424E 03  -4.13208E 04

ROW 10
   6.42241E 04  -9.32779E 04   4.52939E 04

ROW 11
   2.02609E 05  -9.28015E 04

ROW 12
   6.45728E 04


STIFFNESS MATRIX - RELATES COMMON JOINTS TO FREE JOINTS (COMMON JOINTS FREE)   K-FJ

ROW  1
  -5.51237E 05   7.56539E 01  -1.71924E 06   2.11420E 02   3.47788E 05   2.12822E 03

ROW  2
   3.99647E 02  -9.08418E 02   3.23390E 03  -2.01208E 03   1.20878E 04  -1.23908E 04

ROW  3
  -2.59904E 02  -5.84937E 05  -8.64233E 02  -1.82366E 06  -1.96170E 03   8.27486E 04

ROW  4
   1.15871E 05  -4.08586E 01   1.02936E 05  -6.71036E 01  -4.14194E 04   2.75737E 01

ROW  5
   7.94014E 02   1.51468E 03   1.33997E 03   2.51848E 03  -2.03090E 02  -2.50729E 02

ROW  6
   1.12636E 02   1.22655E 05   1.84209E 02   2.04425E 05  -5.61756E 01  -8.20262E 03

ROW  7
  -2.88251E 04   1.01441E 01  -4.80049E 04   1.68088E 01   1.02651E 04  -3.12535E 00

ROW  8
```

```
  -3.00275E 02   -5.01851E 02   -4.97143E 02   -8.36118E 02    1.20883E 02    4.09649E 01

 ROW  9
  -3.44807E 01   -3.05400E 04   -5.77662E 01   -5.08987E 04    1.06477E 01    2.06634E 03

 ROW 10
   4.82680E 03   -6.37270E 00    8.03841E 03   -1.06053E 01   -1.71920E 03    8.21345E-01

 ROW 11
   1.56446E 01    8.74708E 01    2.56128E 01    1.45727E 02   -7.67339E 00   -7.26627E 00

 ROW 12
   1.28943E 01    5.09067E 03    2.15280E 01    8.48425E 03   -4.33392E 00   -3.44263E 02


 UPPER TRIANGLE OF STIFFNESS MATRIX - COMMON JOINTS   K-J

 ROW  1
   4.58651E 05   -2.74576E 01    1.56293E 06   -4.70249E 01   -3.24906E 05   -2.97270E 01

 ROW  2
   4.87528E 05   -4.66473E 01    1.66173E 06    5.39717E 00   -6.57840E 04

 ROW  3
   6.59853E 06   -8.17691E 01   -1.36372E 06   -9.74784E 01

 ROW  4
   7.01554E 06    2.95689E-01   -2.76276E 05

 ROW  5
   3.63602E 05   -2.17850E 02

 ROW  6
   9.10844E 04


 UPPER TRIANGLE OF TRANSFORMED STIFFNESS MATRIX FOR COMPONENT 2

 ROW  1
   6.28322E 05   0.          0.          0.          0.          0.          0.          0.
   0.            0.          0.          0.          0.

 ROW  2
   1.69943E 06   0.          0.          0.          0.          0.          0.          0.
   0.            0.          0.          0.

 ROW  3
   2.67665E 07   0.          0.          0.          0.          0.          0.          0.
   0.            0.          0.

 ROW  4
   3.59345E 07   0.          0.          0.          0.          0.          0.          0.
   0.            0.

 ROW  5
   9.52990E 07   0.          0.          0.          0.          0.          0.          0.
```

```
          0.

     ROW  6
          2.48980E 08   0.            0.            0.            0.            0.            0.            0.

     ROW  7
          2.59187E 08   0.            0.            0.            0.            0.            0.

     ROW  8
          4.44797E 02  -4.47645E 02   3.49453E 03  -3.50216E 03  -4.93587E 03  -4.01242E 03

     ROW  9
          4.42832E 02  -3.50338E 03   3.48564E 03   4.03750E 03   4.01320E 03

     ROW 10
          1.90796E 05  -1.90837E 05  -5.79529E 04  -1.20631E 04

     ROW 11
          1.90880E 05   5.79827E 04   1.20623E 04

     ROW 12
          9.55068E 04   3.24085E 03

     ROW 13
          7.70080E 04


     UPPER TRIANGLE OF REDUCED WEIGHT MATRIX FOR COMPONENT (COMMON JOINTS RESTRAINED)    M-FF

     ROW  1
          3.33000E-01   0.            0.            0.            0.            0.            0.            0.
          0.            0.            0.            0.

     ROW  2
          3.33000E-01   0.            0.            0.            0.            0.            0.            0.
          0.            0.            0.

     ROW  3
          3.33000E-01   0.            0.            0.            0.            0.            0.            0.
          0.            0.

     ROW  4
          2.88000E-01   0.            0.            0.            0.            0.            0.            0.
          0.

     ROW  5
          2.88000E-01   0.            0.            0.            0.            0.            0.            0.

     ROW  6
          2.88000E-01   0.            0.            0.            0.            0.            0.

     ROW  7
          2.55000E-01   0.            0.            0.            0.            0.

     ROW  8
          2.55000E-01   0.            0.            0.            0.

     ROW  9
```

```
      2.55000E-01    0.          0.          0.

   ROW 10
      2.20000E-01    0.          0.

   ROW 11
      2.20000E-01    0.

   ROW 12
      2.20000E-01


   MASS MATRIX - RELATES COMMON JOINTS TO FREE JOINTS (COMMON JOINTS FREE)    M-FJ

   ROW  1
      0.             0.          0.          0.          0.          0.
   ROW  2
      0.             0.          0.          0.          0.          0.
   ROW  3
      0.             0.          0.          0.          0.          0.
   ROW  4
      0.             0.          0.          0.          0.          0.
   ROW  5
      0.             0.          0.          0.          0.          0.
   ROW  6
      0.             0.          0.          0.          0.          0.
   ROW  7
      0.             0.          0.          0.          0.          0.
   ROW  8
      0.             0.          0.          0.          0.          0.
   ROW  9
      0.             0.          0.          0.          0.          0.
   ROW 10
      0.             0.          0.          0.          0.          0.
   ROW 11
      0.             0.          0.          0.          0.          0.
   ROW 12
      0.             0.          0.          0.          0.          0.


   UPPER TRIANGLE OF MASS MATRIX - COMMON JOINTS    M-JJ

   ROW  1
      1.29504E-03    0.          0.          0.          0.          0.
```

```
ROW  2
   1.29504E-03    0.             0.             0.             0.

ROW  3
   0.             0.             0.             0.

ROW  4
   0.             0.             0.

ROW  5
   0.             0.

ROW  6
   0.


NORMALIZED MODE SHAPES FOR ORTHOGONALITY

MODE 1 - FREQUENCY =      126.157 CPS

   7.12582E-01    4.88070E-01    6.83657E-01    5.41108E 00    5.59319E 00    5.19502E 00    1.22604E 01    1.20717E 01
   1.17998E 01    1.97927E 01    1.94330E 01    1.99939E 01

MODE 2 - FREQUENCY =      207.478 CPS

  -9.06784E-01   -1.75396E-02    9.00228E-01   -6.66956E 00    1.88165E-02    6.74464E 00   -1.48563E 01    1.30858E-01
   1.51414E 01   -2.32409E 01    3.34807E-01    2.39340E 01

MODE 3 - FREQUENCY =      823.410 CPS

   3.83378E 00    8.28312E 00    3.10177E 00    1.58568E 01    1.58661E 01    1.29692E 01    9.97982E 00    8.99779E 00
   8.21115E 00   -1.34904E 01   -1.22011E 01   -1.07956E 01

MODE 4 - FREQUENCY =      954.061 CPS

  -3.80779E 00    5.92539E-01    4.34357E 00   -1.70856E 01    1.28301E 00    1.93492E 01   -1.11439E 01    8.24817E-01
   1.26227E 01    1.47636E 01   -9.58371E-01   -1.67690E 01

MODE 5 - FREQUENCY =     1553.690 CPS

   2.00982E 00    3.15174E 01    1.71712E 00   -2.13485E 00    2.33815E 00   -2.33759E 00   -6.62496E 00   -7.35366E 00
  -6.14227E 00    4.79988E 00    4.45155E 00    4.51506E 00

MODE 6 - FREQUENCY =     2511.320 CPS

   8.75667E 00   -1.57004E 00    2.68351E 00    2.92420E 01   -1.72627E 01    1.04005E 01   -5.57308E 00   -1.09504E 01
   7.75309E 00    5.55459E 00    1.81349E 00   -2.95763E-01

MODE 7 - FREQUENCY =     2572.280 CPS

  -6.93920E 00    1.68301E 00    9.02957E 00   -9.14682E 00   -1.03011E 01    1.70954E 01    2.19123E 01   -6.46637E 00
  -1.32148E 01   -7.52316E 00   -8.53214E-01    7.65481E 00


CHECK FOR ORTHOGONALITY

UPPER TRIANGLE OF THE GENERALIZED MASS MATRIX IS NOW
```

```
ROW 1
  1.00000E 00   2.97555E-07   5.36412E-08   1.35301E-07  -1.80098E-07   9.62603E-08   3.79607E-07
ROW 2
  1.00000E 00   8.14212E-08   2.21995E-07  -2.97788E-07   4.75223E-07  -1.10137E-07
ROW 3
  1.00000E 00   3.43849E-07   3.33076E-07   3.48765E-08   7.30441E-08
ROW 4
  1.00000E 00   6.73325E-08   6.32944E-08  -7.24075E-09
ROW 5
  1.00000E 00  -2.70145E-07  -1.00436E-07
ROW 6
  1.00000E 00   3.51801E-08
ROW 7
  1.00000E 00

UPPER TRIANGLE OF TRANSFORMED MASS MATRIX FOR COMPONENT 2

ROW 1
  1.00000E 00   0.            0.            0.            0.            0.            0.            3.52117E-02
  3.55963E-02   9.65917E-01   9.83159E-01  -1.91009E-01  -4.18817E-02
ROW 2
  1.00000E 00   0.            0.            0.            0.            0.           -2.55277E-02   2.64476E-02
 -4.69888E-01   4.98221E-01   9.06024E-02  -2.41790E-02
ROW 3
  1.00000E 00   0.            0.            0.            0.            2.29964E-02   2.06855E-02   1.52324E-01
  1.34880E-01  -3.33809E-02  -2.92657E-03
ROW 4
  1.00000E 00   0.            0.            0.           -1.42799E-02   1.77319E-02  -1.08688E-01   1.32165E-01
  2.11161E-02  -6.02249E-03
ROW 5
  1.00000E 00   0.            0.            1.18032E-02   1.15579E-02   3.57115E-02   3.47199E-02  -1.14106E-02
  2.52033E-03
ROW 6
  1.00000E 00   0.            9.24745E-03   2.19900E-03   4.36038E-02   1.28064E-02  -8.50144E-03  -7.83479E-04
ROW 7
  1.00000E 00  -8.07169E-03   1.06337E-02  -3.48948E-02   4.72971E-02   6.92118E-03  -2.09644E-03
ROW 8
  4.62535E-03   8.84921E-04   5.44367E-02   2.33402E-02  -1.96327E-02  -8.05818E-04
ROW 9
  4.71324E-03   2.34044E-02   5.56957E-02  -4.80866E-03  -2.36153E-03
```

ROW 10
1.19711E 00 7.22331E-01 -2.36256E-01 -2.87870E-02

ROW 11
1.25885E 00 -1.44613E-01 -5.46648E-02

ROW 12
4.0616E-02 5.74829E-03

ROW 13
2.48617E-03

INPUT DATA FOR COMPONENT 3

12 DEGREES OF FREEDOM
7 MODES
1 COMMON JOINTS

MODE SHAPES

```
MODE 1 - FREQUENCY =      82.477 CPS

   4.37633E-02   4.27628E-02   3.44499E-02   3.07082E-01   3.06972E-01   3.18474E-01   6.38228E-01   6.39841E-01
   6.41116E-01   1.00000E 00   9.94571E-01   9.83305E-01

MODE 2 - FREQUENCY =     112.979 CPS

  -2.75355E-01   7.66566E-05   5.52870E-01  -3.55707E-01   8.32325E-04   7.11989E-01  -4.27924E-01   3.01599E-03
   8.62180E-01  -4.89719E-01   7.52720E-03   1.00000E 00

MODE 3 - FREQUENCY =     361.313 CPS

  -4.97534E-01  -1.25534E-03   1.00000E 00  -2.30311E-01  -5.52473E-03   4.42450E-01   6.96038E-02  -4.10120E-03
  -1.50834E-01   3.74840E-01   6.25066E-04  -7.49210E-01

MODE 4 - FREQUENCY =     681.926 CPS

   1.65835E-01   2.44339E-01   5.36194E-01   1.00000E 00   9.09780E-01   7.20624E-01   6.76938E-01   5.84728E-01
   3.76553E-01  -7.57196E-01  -6.68956E-01  -4.70477E-01

MODE 5 - FREQUENCY =    1279.880 CPS

   5.71372E-01   7.35543E-02  -8.29162E-01  -3.25083E-01   1.27944E-01   1.00000E 00  -4.34102E-01   2.78281E-02
   9.36749E-01   3.96245E-01  -6.75696E-02  -9.63883E-01

MODE 6 - FREQUENCY =    1891.950 CPS

   7.36647E-01   5.59734E-01   1.00000E 00   5.16095E-01   4.29582E-01   6.88228E-02  -6.96970E-01  -6.89002E-03
  -5.19580E-01   2.55403E-01   2.85712E-01   2.81553E-01

MODE 7 - FREQUENCY =    2781.980 CPS

   1.00000E 00   4.63241E-01   7.05656E-01  -8.08685E-01  -4.58104E-01   1.38029E-01   6.43519E-01   3.29389E-01
  -3.65825E-01  -2.05193E-01  -8.81996E-02   1.60157E-01
```

UPPER TRIANGLE OF REDUCED STIFFNESS MATRIX FOR COMPONENT (COMMON JOINTS RESTRAINED)   K-FF

```
ROW  1
   7.88907E 04  -1.09507E 05   3.58562E 04  -1.10263E 04   2.16435E 03  -1.82966E 02   6.45244E 03   2.64544E 01
  -2.66948E 00  -1.07517E 03  -4.62009E 00   4.15884E-01

ROW  2
   3.51425E 05  -5.41123E 04   1.16190E 03  -5.58576E 04   2.28626E 02   4.02588E 02   1.69988E 04   2.05353E 02
```

```
   -6.93702E 01   -2.83023E 03   -3.48611E 01

 ROW  3
   2.30135E 04   -1.83587E 02    7.40825E 02   -1.03213E 04   -2.67917E 00    1.35188E 01    6.42017E 03    4.14279E-01
  -2.04333E 00   -1.07021E 03

 ROW  4
   1.01554E 05   -1.11744E 05    3.62407E 04   -2.45424E 04    2.24874E 03   -1.78984E 02    6.44955E 03   -3.99523E 01
  -4.15644E 00

 ROW  5
   2.08895E 05   -5.53486E 04    2.20173E 03   -2.93712E 04    7.58970E 02   -3.13457E 01    7.10912E 03   -1.60154E 01

 ROW  6
   4.49063E 04   -1.78984E 02    7.82950E 02   -2.30136E 04   -4.15410E 00   -2.03304E 01    6.41942E 03

 ROW  7
   1.01540E 05   -1.12194E 05    3.62337E 04   -1.19163E 04    2.51085E 03   -1.77851E 02

 ROW  8
   1.94307E 05   -5.55780E 04    2.51066E 03   -1.31695E 04    9.16389E 02

 ROW  9
   4.49027E 04   -1.77852E 02    9.16469E 02   -1.03183E 04

 ROW 10
   7.80755E 04   -1.09874E 05    3.60558E 04

 ROW 11
   1.68917E 05   -5.47593E 04

 ROW 12
   2.29601E 04


 STIFFNESS MATRIX - RELATES COMMON JOINTS TO FREE JOINTS (COMMON JOINTS FREE)    K-FJ

 ROW  1
  -6.91696E 02   -1.15283E 03   -4.05270E 03

 ROW  2
  -1.48011E 05   -4.59485E 05    3.83026E 03

 ROW  3
  -3.52552E 02   -5.87586E 02    1.02244E 03

 ROW  4
   9.64669E 02    1.60778E 03   -4.31283E 00

 ROW  5
   3.04991E 04    5.08318E 04    6.35361E 00

 ROW  6
   4.91911E 02    8.19851E 02   -2.04077E 00

 ROW  7
  -3.29309E 02   -5.48848E 02   -4.17559E-03
```

```
ROW  8
  -7.49147E 03   -1.24861E 04    4.62058E-02

ROW  9
  -1.67942E 02   -2.79903E 02   -4.20302E-02

ROW 10
   5.63363E 01    9.38939E 01   -5.77948E-05

ROW 11
   1.24657E 03    2.07761E 03    4.70399E-04

ROW 12
   2.85825E 01    4.76375E 01   -4.12604E-04
```

UPPER TRIANGLE OF STIFFNESS MATRIX - COMMON JOINTS   K-JJ

```
ROW  1
   1.23757E 05    4.19061E 05    0.

ROW  2
   1.76244E 06    0.

ROW  3
   2.03460E 04
```

UPPER TRIANGLE OF TRANSFORMED STIFFNESS MATRIX FOR COMPONENT 3

```
ROW  1
   2.68549E 05    0.             0.             0.             0.             0.             0.             0.
   0.             0.

ROW  2
   5.03913E 05    0.             0.             0.             0.             0.             0.             0.
   0.

ROW  3
   5.14809E 06    0.             0.             0.             0.             0.             0.             0.

ROW  4
   1.83530E 07    0.             0.             0.             0.             0.             0.

ROW  5
   6.46693E 07    0.             0.             0.             0.             0.

ROW  6
   1.41312E 08    0.             0.             0.             0.

ROW  7
   3.05540E 08    0.             0.             0.

ROW  8
   1.52951E 00    7.73437E 00    5.31563E-04
```

```
ROW  9
   2.09219E 01    9.56793E-01

ROW 10
   5.29297E-01

UPPER TRIANGLE OF REDUCED WEIGHT MATRIX FOR COMPONENT (COMMON JOINTS RESTRAINED)  , M-FF

ROW  1
   8.40000E-02    0.             0.             0.             0.             0.             0.             0.
   0.             0.             0.             0.

ROW  2
   1.25000E-01    0.             0.             0.             0.             0.             0.             0.
   0.             0.             0.

ROW  3
   4.10000E-02    0.             0.             0.             0.             0.             0.             0.
   0.             0.

ROW  4
   8.40000E-02    0.             0.             0.             0.             0.             0.             0.
   0.

ROW  5
   1.25000E-01    0.             0.             0.             0.             0.             0.             0.

ROW  6
   4.10000E-02    0.             0.             0.             0.             0.             0.

ROW  7
   8.40000E-02    0.             0.             0.             0.             0.

ROW  8
   1.25000E-01    0.             0.             0.             0.

ROW  9
   4.10000E-02    0.             0.             0.

ROW 10
   8.40000E-02    0.             0.

ROW 11
   1.25000E-01    0.

ROW 12
   4.10000E-02

MASS MATRIX - RELATES COMMON JOINTS TO FREE JOINTS (COMMON JOINTS FREE)    M-FJ

ROW  1
   0.             0.             0.

ROW  2
```

```
      0.                  0.                   0.
    ROW  3
      0.                  0.                   0.
    ROW  4
      0.                  0.                   0.
    ROW  5
      0.                  0.                   0.
    ROW  6
      0.                  0.                   0.
    ROW  7
      0.                  0.                   0.
    ROW  8
      0.                  0.                   0.
    ROW  9
      0.                  0.                   0.
    ROW 10
      0.                  0.                   0.
    ROW 11
      0.                  0.                   0.
    ROW 12
      0.                  0.                   0.

    UPPER TRIANGLE OF MASS MATRIX - COMMON JOINTS   M-JJ

    ROW  1
      1.29584E-03         0.                   0.
    ROW  2
      0.                  0.
    ROW  3
      0.

    NORMALIZED MODE SHAPES FOR ORTHOGONALITY

    MODE 1 - FREQUENCY =      22.477 CPS

      1.40643E 00   1.37428E 00   1.10709E 00   9.86878E 00   9.86364E 00   1.02349E 01   2.05109E 01   2.05628E 01
      2.06037E 01   3.21373E 01   3.19628E 01   3.16008E 01

    MODE 2 - FREQUENCY =     112.979 CPS

     -1.36496E 01   3.79745E-03   2.73883E 01  -1.76212E 01   4.12320E-07   3.52708E 01  -2.11987E 01   1.49407E-01
      4.27110E 01  -2.42599E 01   3.72885E-01   4.95384E 01
```

```
MODE 3 - FREQUENCY =    361.113 CPS

  -2.94176E 01   -7.42742E-02    5.91267E 01   -1.36175E 01   -3.26659E-01    2.61606E 01    4.11545E 00   -2.42491E-01
  -8.91832E 00    2.21635E 01    3.69581E-02   -4.42983E 01

MODE 4 - FREQUENCY =    681.826 CPS

   4.96610E 00    7.31698E 00    1.60569E 01    2.99460E 01    2.72445E 01    2.15798E 01    2.02716E 01    1.75103E 01
   1.12763E 01   -2.26750E 01   -2.00326E 01   -1.40889E 01

MODE 5 - FREQUENCY =   1279.880 CPS

   2.44082E 01    3.14214E 00   -3.54206E 01   -1.38871E 01    5.46559E 00    4.27186E 01   -1.85442E 01    1.18878E 00
   4.00166E 01    1.66707E 01   -2.88648E 00   -4.11757E 01

MODE 6 - FREQUENCY =   1891.950 CPS

   2.63596E 01    2.00291E 01    3.57832E 01    1.84675E 01    1.53718E 01    2.46270E 00   -2.49398E 01   -2.46547E 01
  -1.85925E 01    9.13914E 00    1.02237E 01    1.00749E 01

MODE 7 - FREQUENCY =   2781.980 CPS

   3.76352E 01    1.74342E 01    2.65575E 01   -3.04350E 01   -1.72408E 01    5.19475E 00    2.42189E 01    1.23966E 01
  -1.37679E 01   -7.72247E 00   -3.31941E 00    6.02754E 00


CHECK FOR ORTHOGONALITY

UPPER TRIANGLE OF THE GENERALIZED MASS MATRIX IS NOW

ROW  1
   1.00000E 00   -6.53587E-08    2.92860E-07    4.42412E-07   -1.75867E-08   -5.74257E-08   -1.59785E-07
ROW  2
   1.00000E 00    2.31455E-08    5.50551E-08   -3.10518E-07   -2.47876E-07   -7.91508E-08
ROW  3
   1.00000E 00    3.12898E-08   -1.25820E-08    2.38010E-07   -2.32745E-08
ROW  4
   1.00000E 00   -3.65153E-08   -1.06387E-07   -2.37192E-08
ROW  5
   1.00000E 00   -5.36830E-08   -1.89469E-08
ROW  6
   1.00000E 00   -9.39489E-08
ROW  7
   1.00000E 00


UPPER TRIANGLE OF TRANSFORMED MASS MATRIX FOR COMPONENT 3

ROW  1
```

```
   1.00000E 00     0.              0.              0.              0.              0.              0.              4.13003E-02
   1.15780E 00     1.50496E-03

ROW  2
   1.00000E 00     0.              0.              0.              0.              0.              -5.68605E-05    1.51083E-03
  -1.65303E-01

ROW  3
   1.00000E 00     0.              0.              0.              0.              -4.36581E-04    -3.79528E-03    -3.48582E-02

ROW  4
   1.00000E 00     0.              0.              0.              2.11424E-02     1.20076E-01     -1.00606E-03

ROW  5
   1.00000E 00     0.              0.              4.76994E-03     1.82893E-02     1.94233E-03

ROW  6
   1.00000E 00     0.              1.62625E-02     5.72662E-02     6.74802E-05

ROW  7
   1.00000E 00     1.07067E-02     2.99866E-02     2.37416E-04

ROW  8
   3.88511E-03     5.18012E-02     7.74068E-05

ROW  9
   1.35975E 00     1.54764E-03

ROW 10
   2.85472E-02
```

AIC MATRICES FOR 1/K = 1.0000E 01

COMPONENT 1

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.	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.	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.						

COUPLED AIC MATRIX FOR COMPONENTS 2 AND 3

0.962072E 02	-0.711336E 01	-0.136578E 03	0.538727E 01	0.403241E 02	-0.194996E 01	-0.464903E 02	0.464217E 01
0.655664E 02	-0.410689E 01	-0.190006E 02	0.129620E 01	0.180261E 02	-0.174219E 01	-0.255697E 02	0.154332E 01
0.750913E 01	-0.503517E 00	-0.202693E 01	0.197447E 00	0.287261E 01	-0.175362E 00	-0.842724E 00	0.568409E-01
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.485666E 02	-0.400282E 01	-0.486746E 02	0.262153E 01	0.652877E-01	-0.178181E 01	-0.232823E 01	0.146993E 01
-0.345536E 01	-0.889725E 00	0.597476E 01	0.185707E-01	0.156679E 01	0.159017E 00	0.418460E 00	-0.482892E 00
-0.198114E 01	0.625623E-01	-0.121427E 00	-0.114538E-01	-0.361200E-01	0.458419E-01	0.217313E 00	-0.506601E-02
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
-0.827010E 01	-0.391798E 00	0.482483E 02	0.579542E 00	-0.399601E 02	-0.224724E 01	0.326383E 02	-0.276557E 01
-0.553635E 02	0.422826E 01	0.227763E 02	-0.199818E 01	-0.119510E 02	0.342330E 01	0.203791E 02	-0.410577E 01
-0.832068E 01	0.853400E 00	0.110850E 01	-0.292606E 00	-0.195981E 01	0.352903E 00	0.842501E 00	-0.708684E-01
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.237461E 01	-0.240628E 00	-0.864962E 01	0.136255E 01	0.633449E 01	-0.839521E 00	0.353606E 02	-0.304079E 01
-0.376757E 02	0.125605E 01	0.219520E 01	-0.181503E-01	0.546823E 01	0.536060E-02	-0.968245E 01	0.161877E 00
0.424325E 01	-0.194463E 00	-0.117057E 01	0.170819E 00	0.188624E 01	-0.243009E 00	-0.714728E 00	0.906594E-01
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.

0.	0.	0.	0.	0.	0.	0.	0.
0.168855E 02	-0.418128E 01	-0.190799E 02	0.431332E 01	0.203425E 01	-0.109589E 01	-0.401291E 01	0.216736E 01
0.332615E 02	-0.409272E 01	-0.292018E 02	0.478358E 00	0.170488E 02	-0.272991E 01	-0.190001E 02	0.250934E 01
0.186192E 01	-0.597616E 00	0.709056E 00	-0.150249E 00	-0.125172E 01	0.199711E 00	0.540006E 00	-0.545471E-01
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.139454E 01	-0.152078E 01	0.159613E 02	-0.940097E 00	-0.175687E 02	0.122913E 01	-0.416257E 02	0.223986E 01
0.868191E 02	0.136299E 00	-0.450536E 02	-0.252026E 01	0.136908E 01	-0.115448E 01	0.140472E 02	-0.472751E 00
-0.155394E 02	0.714178E 00	0.315446E 01	-0.642883E 00	-0.435310E 01	0.675036E 00	0.117929E 01	-0.126366E 00
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.452193E 01	-0.128340E 01	-0.319832E 01	0.570416E 00	-0.142237E 01	0.285208E 00	-0.125925E 02	0.297766E 01
0.123335E 02	-0.217565E 01	0.420268E 00	0.562223E-01	0.573327E 02	-0.465575E 01	-0.705286E 02	0.321727E 01
0.130902E 02	-0.953061E 00	0.643023E 01	-0.409490E 00	-0.884009E 01	0.396322E 00	0.240225E 01	-0.175724E 00
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.190075E 01	-0.951616E 00	-0.402569E 01	0.153886E 01	0.210928E 01	-0.577768E 00	0.670689E 01	-0.101672E 01
-0.382909E 01	0.370596E 00	-0.293523E 01	0.771456E-01	-0.518129E 01	0.966797E 00	0.362163E 02	-0.251080E 01
-0.310444E 02	0.196552E 00	-0.272816E 00	0.160440E 00	0.448275E 01	-0.440136E 00	-0.421167E 01	0.860971E-01
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.106445E 02	-0.384639E 01	-0.154224E 02	0.505143E 01	0.466596E 01	-0.159339E 01	-0.149577E 01	0.322807E 00
0.161363E 02	-0.252612E 01	-0.147337E 02	0.142310E 01	-0.510498E 02	0.311102E 01	0.113539E 03	-0.245698E 01
-0.624165E 02	-0.123253E 01	-0.741605E 01	0.598361E 00	0.170024E 02	-0.723109E 00	-0.957855E 01	0.184666E-01
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.243334E 01	-0.366361E 00	-0.272576E 01	0.297330E 00	0.279849E 00	-0.628608E-01	-0.741922E 01	0.631087E 00
0.860013E 01	-0.435127E 00	-0.115994E 01	0.147331E 00	0.194062E 02	-0.120409E 01	-0.234113E 02	0.730467E 00
0.399233E 01	-0.316081E 00	0.291521E 01	-0.543143E-01	-0.245496E 01	-0.126525E 00	-0.460743E 00	0.118781E-01
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.804467E 00	-0.268829E 00	-0.535334E 00	0.191138E 00	-0.286645E 00	-0.773407E-03	0.110278E 01	-0.324231E-01
-0.337682E 01	0.171989E 00	0.223442E 01	-0.553443E-01	-0.241754E 01	0.301808E 00	0.138168E 02	-0.925784E 00
-0.114124E 02	0.144252E 00	-0.252234E 01	0.301901E 00	0.634654E 01	-0.353244E 00	-0.381615E 01	-0.104397E-01
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
-0.230373E 01	0.503576E 00	0.144439E 01	0.144372E 00	0.919305E 00	-0.428580E 00	0.106946E 02	-0.230105E 01

-0.129149E 02	0.156103E 01	0.204862E 01	0.204730E 00	-0.200053E 02	0.202712E 01	0.420825E 02	-0.178197E 01
-0.212110E 02	-0.242586E 00	-0.492382E 01	0.103975E 00	0.102361E 02	0.209361E 00	-0.529779E 01	-0.334110E 00
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.507538E 00	-0.289592E-01	-0.154879E 02	0.373353E 01	0.151917E 02	-0.276436E 01	-0.585934E 01	0.197317E 01
0.172382E 02	-0.497545E 01	-0.114579E 02	0.264336E 01	0.285158E 01	-0.158824E 01	-0.114520E 00	0.439236E 00
-0.285938E 01	0.875411E 00	0.644752E 00	-0.326170E 00	-0.347101E 00	0.202736E 00	-0.337556E 00	0.705746E-01
0.129658E 02	-0.206705E 01	0.580621E 01	0.936155E 00	-0.188336E 02	0.585500E-02	0.210288E 01	-0.446357E-01
-0.559266E 01	0.149558E 00	0.349260E 01	0.325272E-02	0.187303E 01	-0.262483E 00	0.265448E-01	0.134954E 00
-0.190688E 01	0.113209E-02	-0.208114E 00	0.291647E-01	-0.294959E-02	-0.149949E-01	0.211876E 00	-0.125787E-03
0.849194E 00	-0.294488E-01	-0.253194E 02	0.565289E 01	0.247873E 02	-0.407674E 01	-0.958980E 01	0.304883E 01
0.282043E 02	-0.762293E 01	-0.187332E 02	0.398709E 01	0.469575E 01	-0.251049E 01	-0.203752E 00	0.714433E 00
-0.468361E 01	0.134645E 01	0.109170E 01	-0.512769E 00	-0.571472E 00	0.320694E 00	-0.551176E 00	0.105471E 00
0.212051E 02	-0.300715E 01	0.939400E 01	0.168574E 01	-0.306682E 02	-0.513061E 00	0.341907E 01	-0.141409E-01
-0.910463E 01	0.883028E-01	0.568716E 01	0.102209E 00	0.306078E 01	-0.375545E 00	0.359697E-01	0.220513E 00
-0.310514E 01	-0.510688E-01	-0.346086E 00	0.417272E-01	-0.399692E-02	-0.245014E-01	0.345016E 00	0.567430E-02
0.102297E 01	-0.253184E-01	-0.294264E 02	0.573621E 01	0.287193E 02	-0.391965E 01	-0.111658E 02	0.321615E 01
0.328226E 02	-0.791440E 01	-0.217757E 02	0.401555E 01	0.552038E 01	-0.275870E 01	-0.267392E 00	0.823613E 00
-0.546080E 01	0.140824E 01	0.127799E 01	-0.558114E 00	-0.671632E 00	0.352853E 00	-0.639366E 00	0.104268E 00
0.246610E 02	-0.280670E 01	0.167367E 02	0.224650E 01	-0.354196E 02	-0.156307E 01	0.393910E 01	0.923897E-01
-0.105108E 02	-0.106307E 00	0.656803E 01	0.298023E 00	0.355499E 01	-0.337388E 00	0.280685E-01	0.256081E 00
-0.358624E 01	-0.157245E 00	-0.394998E 00	0.374875E-01	-0.311903E-02	-0.284534E-01	0.398471E 00	0.174717E-01
-0.867103E 01	0.346927E 01	0.179654E 02	-0.604380E 01	-0.927260E 01	0.254603E 01	0.843327E 01	-0.332424E 01
-0.263200E 02	0.746085E 01	0.179474E 02	-0.359159E 01	-0.619437E 01	0.221457E 01	0.114488E 02	-0.354132E 01
-0.522552E 01	0.138451E 01	-0.947195E 00	0.329207E 00	0.264413E 01	-0.770451E 00	-0.170406E 01	0.404324E 00
0.656700E 01	-0.663977E 00	-0.567007E 01	0.471155E 00	-0.911247E 00	0.600948E-02	0.340511E 01	-0.996072E 00
0.117152E 02	0.249402E 00	-0.151565E 02	-0.205407E-02	0.701864E 01	-0.714750E 00	-0.594505E 01	0.503598E 00
-0.108913E 01	0.637264E-02	-0.257193E 00	0.297856E-01	0.137041E 00	-0.184821E-01	0.120880E 00	-0.205038E-03
-0.141962E 02	0.540007E 01	0.294005E 02	-0.933645E 01	-0.151775E 02	0.388877E 01	0.138299E 02	-0.518031E 01
-0.430568E 02	0.114164E 02	0.293100E 02	-0.534804E 01	-0.103351E 02	0.342953E 01	0.187350E 02	-0.544599E 01
-0.855475E 01	0.211020E 01	-0.156091E 01	0.509647E 00	0.432659E 01	-0.118117E 01	-0.278627E 01	0.611169E 00
0.107136E 02	-0.899009E 00	-0.924773E 01	0.709831E 00	-0.148401E 01	-0.155143E-01	0.560032E 01	-0.152830E 01
0.190419E 02	0.771382E 00	-0.246864E 02	-0.423008E 00	0.114508E 02	-0.969171E 00	-0.969671E 01	0.655083E 00
-0.177369E 01	-0.198441E-01	-0.410855E 00	0.413714E-01	0.223888E 00	-0.262942E-01	0.196843E 00	0.302031E-02
-0.165376E 02	0.577391E 01	0.342535E 02	-0.984481E 01	-0.176742E 02	0.401373E 01	0.161533E 02	-0.555051E 01
-0.500941E 02	0.118245E 02	0.340083E 02	-0.524722E 01	-0.117951E 02	0.363283E 01	0.218086E 02	-0.569426E 01
-0.996483E 01	0.216911E 01	-0.180701E 01	0.539528E 00	0.503586E 01	-0.122781E 01	-0.323913E 01	0.618157E 00
0.124106E 02	-0.659941E 00	-0.107077E 02	0.412174E 00	-0.171422E 01	-0.648759E-01	0.657105E 01	-0.159109E 01
0.219271E 02	0.144874E 01	-0.285037E 02	-0.127062E 01	0.132652E 02	-0.757715E 00	-0.112286E 02	0.450211E 00
-0.204879E 01	-0.790429E-01	-0.486945E 00	0.345382E-01	0.259938E 00	-0.233078E-01	0.227348E 00	0.971737E-02
-0.330944E-01	-0.315478E 00	0.658805E 01	-0.200798E 01	-0.671885E 01	0.198450E 01	-0.306047E 01	0.945063E 00
0.418214E 01	-0.978619E 00	-0.107543E 01	0.160696E 00	-0.108978E 01	0.620370E 00	-0.145844E 02	0.339244E 01
0.159969E 02	-0.314251E 01	-0.267341E 00	0.106762E 00	-0.232089E 01	0.619166E 00	0.262166E 01	-0.582964E 00
-0.259183E 01	0.273149E-01	0.198354E 00	-0.186040E-01	0.609425E-01	-0.228037E-03	0.729615E 01	-0.726886E 00
-0.640770E 01	0.523409E 00	-0.513980E 00	0.673624E-02	0.408945E 01	-0.110952E 01	0.121195E 02	0.298847E 00

-0.162486E 02	-0.177952E-02	0.813461E 00	-0.167225E 00	0.120834E 01	0.590273E-01	-0.202732E 01	0.698360E-04
-0.362576E-01	-0.519259E 00	0.107764E 02	-0.322905E 01	-0.109941E 02	0.304463E 01	-0.500612E 01	0.145291E 01
0.684214E 01	-0.146220E 01	-0.176467E 01	0.233133E 00	-0.177562E 01	0.974983E 00	-0.238458E 02	0.512109E 01
0.259416E 02	-0.467461E 01	-0.429604E 00	0.165982E 00	-0.379742E 01	0.944055E 00	0.428396E 01	-0.876366E 00
-0.422915E 00	0.372883E-01	0.324437E 00	-0.247766E-01	0.992432E-01	0.131970E-02	0.119026E 02	-0.984474E 00
-0.105967E 02	0.671988E 00	-0.132565E 01	-0.116167E-01	0.671984E 01	-0.169411E 01	0.196968E 02	0.823126E 00
-0.264586E 02	-0.453763E 00	0.133299E 01	-0.249840E 00	0.196226E 01	0.129672E 00	-0.330124E 01	-0.561485E-01
-0.640695E-02	-0.609908E 00	0.125335E 02	-0.338156E 01	-0.127987E 02	0.316925E 01	-0.582353E 01	0.151827E 01
0.796165E 01	-0.147505E 01	-0.205306E 01	0.216398E 00	-0.205265E 01	0.106064E 01	-0.277202E 02	0.516591E 01
0.300972E 02	-0.457689E 01	-0.497620E 00	0.177085E 00	-0.441940E 01	0.971122E 00	0.497535E 01	-0.876717E 00
-0.490048E 00	0.297140E-01	0.375364E 00	-0.183683E-01	0.114629E 00	0.466462E-02	0.137868E 02	-0.761059E 00
-0.122689E 02	0.441248E 00	-0.153136E 01	-0.553627E-01	0.787358E 01	-0.174743E 01	0.226772E 02	0.157556E 01
-0.305574E 02	-0.136138E 01	0.155594E 01	-0.246825E 00	0.221627E 01	0.212074E 00	-0.381266E 01	-0.169309E 00
0.804804E 00	-0.386486E 00	-0.192019E-01	0.554395E-02	-0.827999E 00	0.275767E 00	-0.132279E 01	0.335491E 00
0.399572E 01	-0.974704E 00	-0.269224E 01	0.565233E 00	-0.512686E 00	0.155832E 00	-0.379982E 01	0.104908E 01
0.437509E 01	-0.954266E 00	-0.150265E 00	-0.117018E-01	-0.683556E 00	0.294900E 00	0.845719E 00	-0.233897E 00
-0.178325E 00	0.109110E-01	0.314036E 00	-0.127384E-01	-0.135646E 00	-0.219359E-03	0.160492E 01	-0.981991E-01
-0.282632E 01	0.114645E 00	0.122073E 01	0.197423E-02	0.111113E 01	-0.345971E 00	0.429366E 01	0.815470E-01
-0.541752E 01	-0.836218E-03	-0.209096E 01	0.101350E 00	0.420007E 01	-0.149156E 00	-0.218941E 01	-0.278282E-02
0.132391E 01	-0.608319E 00	-0.345402E-01	0.981327E-02	-0.135556E 01	0.425913E 00	-0.216342E 01	0.509796E 00
0.653556E 01	-0.147694E 01	-0.440154E 01	0.846052E 00	-0.833953E 00	0.238018E 00	-0.621893E 01	0.160350E 01
0.714759E 01	-0.143207E 01	-0.243231E 00	-0.234342E-01	-0.112267E 01	0.461398E 00	0.138390E 01	-0.357395E 00
-0.290432E 00	0.128114E-01	0.511470E 00	-0.120268E-01	-0.220860E 00	-0.412084E-02	0.261388E 01	-0.115302E 00
-0.460323E 01	0.108242E 00	0.198774E 01	0.370875E-01	0.182891E 01	-0.532827E 00	0.697943E 01	0.251997E 00
-0.882171E 01	-0.151687E 00	-0.340363E 01	0.106893E 00	0.696965E 01	-0.124086E 00	-0.356509E 01	-0.652833E-01
0.155387E 01	-0.663515E 00	-0.459682E-01	0.127926E-01	-0.157905E 01	0.448922E 00	-0.251608E 01	0.520972E 00
0.760198E 01	-0.150103E 01	-0.511606E 01	0.639029E 00	-0.961495E 00	0.245669E 00	-0.724069E 01	0.165728E 01
0.829846E 01	-0.142767E 01	-0.273206E 00	-0.351976E-01	-0.131439E 01	0.497947E 00	0.161083E 01	-0.369164E 00
-0.335523E 00	0.559254E-02	0.590898E 00	0.229664E-02	-0.259064E 00	-0.117408E-01	0.301979E 01	-0.583328E-01
-0.531808E 01	-0.206698E-01	0.229557E 01	0.105739E 00	0.214859E 01	-0.558651E 00	0.803793E 01	0.512421E 00
-0.101883E 02	-0.454360E 00	-0.322865E 01	0.154759E-01	0.804956E 01	0.773091E-01	-0.411724E 01	-0.188221E 00

```
EFFECTIVE LOCATION OF COMPONENT D.O.F. IN THE SYSTEM MATRIX
                     q's

NCODE FOR COMPONENT 1
                                          ω1   ω2   ω3   θx1  θx2  θx3  θy1  θy2  θy3
 1    2    3    4    5    6    7          22   23   24   25   26   27   28   29   30

NCODE FOR COMPONENT 2
                                          ω1   ω2   θx1  θx2  θy1  θy2
 8    9    10   11   12   13   14         22   23   25   26   28   29

NCODE FOR COMPONENT 3
                                          ω3   θx3  θy3
 15   16   17   18   19   20   21         24   27   30

THE TIME ELAPSED FOR MATRIX INVERSION =  0.1320E-01 SECONDS

UPPER TRIANGLE OF REDUCED STIFFNESS MATRIX FOR THE TOTAL SYSTEM

ROW   1
   1.80927E 05    0.     0.     0.     0.     0.     0.     0.
   0.             0.     0.     0.     0.     0.     0.     0.
   0.             0.     0.     0.     0.     0.     0.     0.

ROW   2
   3.81208E 06    0.     0.     0.     0.     0.     0.     0.
   0.             0.     0.     0.     0.     0.     0.     0.
   0.             0.     0.     0.     0.     0.     0.

ROW   3
   7.29321E 06    0.     0.     0.     0.     0.     0.     0.
   0.             0.     0.     0.     0.     0.     0.     0.
   0.             0.     0.     0.     0.     0.

ROW   4
   3.42964E 07    0.     0.     0.     0.     0.     0.     0.
   0.             0.     0.     0.     0.     0.     0.     0.
   0.             0.     0.     0.     0.

ROW   5
   1.09977E 08    0.     0.     0.     0.     0.     0.     0.
   0.             0.     0.     0.     0.     0.     0.     0.
   0.             0.     0.     0.

ROW   6
   1.68034E 08    0.     0.     0.     0.     0.     0.     0.
   0.             0.     0.     0.     0.     0.     0.     0.
   0.             0.     0.

ROW   7
   2.52108E 08    0.     0.     0.     0.     0.     0.     0.
   0.             0.     0.     0.     0.     0.     0.     0.
   0.             0.

ROW   8
   6.28322E 05    0.     0.     0.     0.     0.     0.     0.
   0.             0.     0.     0.     0.     0.     0.     0.
```

```
 0.
ROW   9
 1.69943E 06   0.   0.   0.   0.   0.   0.   0.
 0.            0.   0.   0.   0.   0.   0.   0.
ROW  10
 2.67665E 07   0.   0.   0.   0.   0.   0.   0.
 0.            0.   0.   0.   0.   0.   0.
ROW  11
 3.59345E 07   0.   0.   0.   0.   0.   0.   0.
 0.            0.   0.   0.   0.   0.
ROW  12
 9.52990E 07   0.   0.   0.   0.   0.   0.   0.
 0.            0.   0.   0.   0.
ROW  13
 2.48980E 08   0.   0.   0.   0.   0.   0.   0.
 0.            0.   0.   0.
ROW  14
 2.59187E 08   0.   0.   0.   0.   0.   0.   0.
 0.            0.   0.
ROW  15
 2.68549E 05   0.   0.   0.   0.   0.   0.   0.
 0.            0.
ROW  16
 5.03913E 05   0.   0.   0.   0.   0.   0.   0.
 0.
ROW  17
 5.14809E 06   0.   0.   0.   0.   0.   0.   0.
ROW  18
 1.83530E 07   0.   0.   0.   0.   0.   0.
ROW  19
 6.46693E 07   0.   0.   0.   0.   0.
ROW  20
 1.41312E 08   0.   0.   0.   0.
ROW  21
 3.05540E 08   0.   0.   0.
ROW  22
 1.31103E 05   -2.62648E 05   1.31306E 05
ROW  23
 5.25122E 05   -2.62654E 05
ROW  24
 1.31091E 05
```

NOT REPRODUCIBLE

UPPER TRIANGLE OF REDUCED WEIGHT MATRIX FOR THE TOTAL SYSTEM

```
ROW  1
  3.86088E 02   0.            0.            0.            0.            0.            0.            0.
  0.            0.            0.            0.            0.            0.            0.            0.
  0.            0.            0.            0.            0.            5.42598E 02  -4.43000E 02   7.37773E 01

ROW  2
  3.86088E 02   0.            0.            0.            0.            0.            0.            0.
  0.            0.            0.            0.            0.            0.            0.            0.
  0.            0.            0.            0.            2.38185E 02  -1.21071E 02   2.01633E 01

ROW  3
  3.86088E 02   0.            0.            0.            0.            0.            0.            0.
  0.            0.            0.            0.            0.            0.            0.            0.
  0.            0.            0.            1.54308E 01  -9.25589E 01   1.66046E 02

ROW  4
  3.86088E 02   0.            0.            0.            0.            0.            0.            0.
  0.            0.            0.            0.            0.            0.            0.            0.
  0.            0.           -1.12692E 02   4.04461E 01  -6.73590E 00

ROW  5
  3.86088E 02   0.            0.            0.            0.            0.            0.            0.
  0.            0.            0.            0.            0.            0.            0.            0.
  0.            6.97466E 01  -2.04897E 01   3.41236E 00

ROW  6
  3.86088E 02   0.            0.            0.            0.            0.            0.            0.
  0.            0.            0.            0.            0.            0.            0.            0.
 -1.32632E 01   1.07823E 02   6.80393E 01

ROW  7
  3.86088E 02   0.            0.            0.            0.            0.            0.            0.
  0.            0.            0.            0.            0.            0.            0.            5.54948E 01
  8.79288E 01  -1.08105E 01

ROW  8
  3.86088E 02   0.            0.            0.            0.            0.            0.            0.
  0.            0.            0.            0.            0.            0.            8.58214E 00   1.92724E 01
 -5.16540E-01

ROW  9
  3.86088E 02   0.            0.            0.            0.            0.            0.            0.
  0.            0.            0.            0.            0.           -7.90430E 00   7.59034E 00   6.69202E-01

ROW 10
  3.86088E 02   0.            0.            0.            0.            0.            0.            0.
  0.            0.            0.            0.            8.04506E 00   8.95263E 00  -1.32639E-01

ROW 11
  3.86088E 02   0.            0.            0.            0.            0.            0.            0.
  0.            0.            0.           -5.06221E 00   6.23531E 00   1.59691E-01

ROW 12
  3.86088E 02   0.            0.            0.            0.            0.            0.            0.
  0.            0.            4.30612E 00   4.79264E 00  -7.93144E-02

ROW 13
```

```
   3.86088E 02   0.            0.            0.            0.            0.            0.            0.
   0.            3.35768E 00   1.09594E 00   -3.33915E-02

ROW  14
   3.86088E 02   0.            0.            0.            0.            0.            0.            0.
   -2.96971E 00  3.90538E 00   5.35206E-02

ROW  15
   3.86088E 02   0.            0.            0.            0.            0.            0.            -7.26649E-03
   4.35869E-02   1.59092E 01

ROW  16
   3.86088E 02   0.            0.            0.            0.            0.            7.98144E-01   -4.78754E 00
   3.96753E 00

ROW  17
   3.86088E 02   0.            0.            0.            0.            1.68309E-01   -1.00957E 00  6.72725E-01

ROW  18
   3.86088E 02   0.            0.            0.            4.85764E-03   -2.91378E-02  8.18710E 00

ROW  19
   3.86088E 02   0.            0.            -9.37832E-03  5.62544E-02   1.79474E 00

ROW  20
   3.86088E 02   0.            -3.25818E-04  1.05437E-03   6.27712E 00

ROW  21
   3.86088E 02   -1.14633E-03  6.87608E-03   4.12801E 00

ROW  22
   1.01809E 03   -7.08720E 02  1.21714E 02

ROW  23
   7.10939E 02   -1.18692E 02

ROW  24
   1.86565E 02
```

```
EIGENVALUES AND EIGENVECTORS OF THE SYSTEM

EIGENVECTOR NUMBER     1
          CORRESPONDING TO -6.3258670E 02
 -3.7208169E-05 -1.8709838E-05 -2.3470081E-05  1.4724591E-06 -3.0600923E-07 -1.3784224E-06
 -5.2504176E-07 -4.7527122E-05 -2.6302303E-06 -6.3205589E-07 -9.9839752E-08 -9.8329854E-08
 -1.4745202E-08 -1.0910318E-08 -9.6842141E-05 -2.6179774E-06 -3.3256147E-09 -7.2484729E-07
 -4.8184085E-06 -7.3597301E-08 -1.9169941E-08  4.7294854E-01  7.3651303E-01  1.0000000E 00

EIGENVECTOR NUMBER     2
          CORRESPONDING TO -1.5689356E 02
 -9.2144612E-04 -2.1485534E-05  5.6694550E-06  1.1973125E-06 -2.3356066E-07  8.2029550E-08
 -1.2797515E-07 -6.7024921E-06  1.8454090E-06 -1.7987434E-07  5.0308674E-08 -2.2353880E-08
 -4.0493505E-09  4.4363803E-09  1.4758188E-05  2.1408526E-06  3.8463833E-08  1.1788146E-07
  3.3589408E-09  1.2202844E-08  4.6424112E-10  1.0000000E 00  1.0739885E-01 -6.0608661E-01

EIGENVECTOR NUMBER     3
          CORRESPONDING TO  2.6765938E 05
  2.2182124E-01 -5.1423549E-03  2.6439347E-03  3.1935046E-04 -6.6826148E-05 -4.5444503E-05
 -7.7465425E-05 -1.0135206E-02  1.6570637E-04 -8.5936646E-05 -9.3566233E-07 -1.2834755E-05
 -2.5089466E-06 -2.0230314E-07  1.0000000E 00 -2.9674909E-03  2.8401699E-05  2.5597889E-05
  1.4835263E-06  2.5000664E-06  7.5751750E-07 -2.0809982E-01 -1.7874054E-01  8.1066148E-02

EIGENVECTOR NUMBER     4
          CORRESPONDING TO  3.0084021E 05
  1.0000000E 00 -2.6111626E-02  1.7164690E-02  1.5394537E-03 -3.1736843E-04 -1.4398041E-05
 -3.2200641E-04 -4.4062016E-02  1.5509027E-03 -3.5154289E-04  1.6086173E-05 -5.2471786E-05
 -1.0872807E-05  9.5521760E-07 -2.6458073E-01  1.8364347E-02  1.4404671E-04  2.4449739E-04
  1.4712188E-05  2.3943836E-05  7.2708834E-06 -8.4106243E-01 -5.6779247E-01  6.9037414E-01

EIGENVECTOR NUMBER     5
          CORRESPONDING TO  5.0404064E 05
 -4.2586513E-02  2.1240036E-03 -7.4831970E-04 -1.0514532E-04  2.0651792E-05 -9.0760390E-06
  1.2335295E-05  3.9905379E-03 -2.0148686E-04  1.4286858E-05 -3.7978661E-06  2.1178681E-06
  5.3440947E-07 -2.9204080E-07  1.9658195E-03  1.0000000E 00 -4.8001654E-06 -1.3354815E-05
 -6.0683177E-07 -1.2978063E-06 -3.9388785E-07  2.7886958E-02  6.6955203E-03 -2.2295621E-02

EIGENVECTOR NUMBER     6
          CORRESPONDING TO  6.3252272E 05
  1.4288496E-01 -1.1652481E-02  6.9465195E-04  5.8021741E-04 -1.1549123E-04 -6.9855285E-05
 -8.9545530E-05  1.0000000E 00  7.1663602E-04 -1.0824931E-04  9.2813400E-06 -1.6009387E-05
 -3.4572550E-06  6.3568081E-07  8.9405524E-04 -2.5868953E-03  1.6283239E-05 -9.2013557E-06
 -6.4584620E-07 -9.0241689E-07 -2.7606330E-07 -1.3183919E-01 -7.4626703E-02 -1.2339023E-02

EIGENVECTOR NUMBER     7
          CORRESPONDING TO  1.4263243E 06
  1.0000000E 00 -1.5707811E-01 -5.3190519E-02  4.1941144E-03 -6.8799205E-04  1.2406296E-03
  7.2802495E-04 -5.0465177E-02  7.1086306E-02  6.7448529E-04  4.5884480E-04  9.7527725E-05
  4.8216252E-06  3.8282274E-05  8.0605882E-03  1.4528377E-02 -7.3319500E-04 -2.9075512E-04
 -1.4771717E-05 -2.6572500E-05 -7.9978015E-06 -1.0353612E-01  6.0743568E-01 -1.6049711E-01

NATURAL FREQUENCIES OF THE SYSTEM

THE NATURAL FREQUENCY NUMBER  1  IS       0.       CPS
THE NATURAL FREQUENCY NUMBER  2  IS       0.       CPS
```

THE NATURAL FREQUENCY NUMBER 3 IS 82.340 CPS
THE NATURAL FREQUENCY NUMBER 4 IS 87.295 CPS
THE NATURAL FREQUENCY NUMBER 5 IS 112.993 CPS
THE NATURAL FREQUENCY NUMBER 6 IS 126.578 CPS
THE NATURAL FREQUENCY NUMBER 7 IS 190.877 CPS

SYSTEM MODE SHAPES FOR FREE JOINTS ON COMPONENT 1

MODE 1	MODE 2	MODE 3	MODE 4	MODE 5	MODE 6	MODE 7	MODE 8	MODE 9
-0.12022E 00	0.28024E 01	0.61286E 00	0.25178E 01	-0.76007E-01	0.24508E 00	0.11216E 01		
0.77483E-01	0.22021E 01	0.20053E 00	0.76494E 00	-0.11682E-01	0.74004E-02	-0.28594E 00		
0.20929E 00	0.18017E 01	-0.22105E-01	-0.16292E 00	0.19706E-01	-0.10311E 00	-0.74358E 00		
0.34114E 00	0.14011E 01	-0.15932E 00	-0.70096E 00	0.31805E-01	-0.14508E 00	-0.60228E 00		
0.53884E 00	0.79939E 00	-0.21441E 00	-0.83107E 00	0.23071E-01	-0.11802E 00	0.16231E 00		
0.60474E 00	0.59877E 00	-0.21530E 00	-0.79792E 00	0.18069E-01	-0.10405E 00	0.39329E 00		
0.67063E 00	0.39812E 00	-0.20513E 00	-0.71752E 00	0.12658E-01	-0.89531E-01	0.55245E 00		
0.80239E 00	-0.33961E-02	-0.13278E 00	-0.33411E 00	0.72599E-04	-0.59590E-01	0.53893E 00		
0.86825E 00	-0.20425E 00	-0.70596E-01	-0.31144E-01	-0.70964E-02	-0.44177E-01	0.36605E 00		
0.93413E 00	-0.40515E 00	0.25256E-02	0.31806E 00	-0.14606E-01	-0.28327E-01	0.12029E 00		
0.10659E 01	-0.80703E 00	0.16237E 00	0.10791E 01	-0.33580E-01	0.40599E-02	-0.48393E 00		
0.11317E 01	-0.10079E 01	0.24650E 00	0.14862E 01	-0.39664E-01	0.21202E-01	-0.86422E 00		
0.11974E 01	-0.12088E 01	0.33195E 00	0.17019E 01	-0.49123E-01	0.38694E-01	-0.12713E 01		

SYSTEM MODE SHAPES FOR FREE JOINTS ON COMPONENT 2

MODE 1	MODE 2	MODE 3	MODE 4	MODE 5	MODE 6	MODE 7	MODE 8	MODE 9
0.48675E 00	0.95787E 00	-0.21712E 00	-0.87320E 00	0.29976E-01	0.58252E 00	-0.14855E 00		
0.61294E 00	0.57362E 00	-0.20075E 00	-0.73129E 00	0.18836E-01	0.38518E 00	0.27129E 00		
0.73912E 00	0.18937E 00	-0.18454E 00	-0.59065E 00	0.90260E-02	0.60994E 00	0.64376E 00		
0.51395E 00	0.87435E 00	-0.26817E 00	-0.10845E 01	0.48056E-01	0.52813E 01	-0.69585E 00		
0.62918E 00	0.52342E 00	-0.25151E 00	-0.94271E 00	0.37936E-01	0.54936E 01	0.66746E-01		
0.74450E 00	0.17229E 00	-0.22933E 00	-0.77258E 00	0.25430E-01	0.51259E 01	0.85723E 00		
0.54098E 00	0.79099E 00	-0.33861E 00	-0.13822E 01	0.74812E-01	0.12132E 02	-0.15363E 01		
0.64540E 00	0.47325E 00	-0.31574E 00	-0.12128E 01	0.62362E-01	0.11977E 02	-0.20614E 00		
0.74968E 00	0.15562E 00	-0.29222E 00	-0.10406E 01	0.49602E-01	0.11738E 02	0.11316E 01		
0.56815E 00	0.70727E 00	-0.41418E 00	-0.17024E 01	0.10405E 00	0.10666E 02	-0.24323E 01		
0.66155E 00	0.42305E 00	-0.38815E 00	-0.15187E 01	0.90154E-01	0.10344E 02	-0.52381E 00		
0.75480E 00	0.13892E 00	-0.36233E 00	-0.13360E 01	0.76343E-01	0.10042E 02	0.13853E 01		

SYSTEM MODE SHAPES FOR FREE JOINTS ON COMPONENT 3

MODE 1	MODE 2	MODE 3	MODE 4	MODE 5	MODE 6	MODE 7	MODE 8	MODE 9
0.95589E 00	-0.47187E 00	0.13934E 01	-0.18526E 00	-0.13655E 02	0.13017E-01	-0.13855E 00		
0.99965E 00	-0.60606E 00	0.14556E 01	0.32926E 00	-0.15929E-01	-0.11213E-01	-0.15215E 00		
0.10876E 01	-0.87367E 00	0.13773E 01	0.14144E 01	0.27357E 02	-0.60237E-01	-0.18123E 00		
0.95504E 00	-0.47174E 00	0.98449E 01	-0.24902E 01	-0.17619E 02	0.30958E-01	-0.14570E 00		
0.99901E 00	-0.60593E 00	0.99455E 01	-0.19117E 01	0.37947E-01	-0.38955E-02	-0.88425E-01		
0.10867E 01	-0.87351E 00	0.10528E 02	-0.85906E 00	0.35258E 02	-0.73069E-01	0.29332E-01		

0.95400E 00	-0.47157E 00	0.20477E 02	-0.53722E 01	-0.21176E 02	0.50133E-01	-0.12129E 00
0.99794E 00	-0.60575E 00	0.20645E 02	-0.47437E 01	0.16733E 00	0.55128E-02	0.30525E-02
0.10856E 01	-0.87332E 00	0.20917E 02	-0.34741E 01	0.42719E 02	-0.83496E-01	0.25048E 00
0.95294E 00	-0.47143E 00	0.32093E 02	-0.85113E 01	-0.24214E 02	0.69091E-01	-0.73979E-01
0.99702E 00	-0.60568E 00	0.32045E 02	-0.77642E 01	0.41368E 00	0.15450E-01	0.10813E 00
0.10847E 01	-0.87323E 00	0.31933E 02	-0.62699E 01	0.49568E 02	-0.91649E-01	0.47189E 00

GENERALIZED MASS MATRIX FOR THE TOTAL SYSTEM

```
 6.38403E-01  -1.54503E-06  -1.41649E-09   3.17956E-09   1.07419E-08  -9.59838E-09  -5.99677E-09
-1.56183E-06   1.85091E 00  -8.24498E-09  -8.34970E-07   1.06865E-08  -5.34498E-08  -2.93679E-07
-9.31323E-09  -2.45025E-08   1.06343E 00   1.51258E-06   4.27862E-09   2.57922E-08   2.20675E-08
 2.60770E-08  -8.39026E-07   1.49186E-06   1.10718E 00  -6.56597E-09   1.77566E-08   1.92939E-07
 1.08848E-08   1.17178E-08   4.64632E-09  -7.18208E-09   1.00047E 00   1.06992E-08  -5.45822E-09
-7.77072E-09  -5.70089E-08   2.39628E-08   1.99978E-08   1.06937E-08   1.00085E 00   3.10904E-08
 1.67638E-08  -2.54159E-07   1.72297E-08   1.49520E-07  -4.23197E-09   2.56165E-08   4.16718E-01
```

GENERALIZED AERODYNAMIC FORCES FOR 1/K = 1.0000E 01

```
-0.47111833E 02 -0.21968463E 02   0.14384303E 03  0.89207724E 01   -0.83685180E 02 -0.19839139E 03   -0.46236642E 03  0.51130739E 02
-0.13671253E 05  0.21562985E 03   0.68124125E 02 -0.11282143E 02   -0.11102205E 03 -0.23611010E 02

-0.11039279E 02  0.26916291E 01   0.32244561E 02 -0.21655729E 02    0.46166780E 02  0.13898841E 03    0.25158353E 03 -0.11399768E 02
 0.91751227E 04 -0.11502524E 03   0.65813721E 02 -0.12168104E 03   -0.24987570E 03  0.19053188E 02

-0.17134386E 03 -0.15608943E 03   0.54969952E 03  0.25825980E 03   -0.12344262E 04 -0.36633762E 04   -0.52933951E 04  0.64779885E 03
-0.18678106E 06  0.21700655E 04  -0.23056684E 03  0.21579465E 04    0.29505081E 04 -0.77319611E 03

 0.73529346E 02  0.40605642E 02  -0.22821368E 03 -0.32045514E 02    0.25269205E 03  0.74117373E 03    0.18163469E 04 -0.16132885E 03
 0.34749794E 05 -0.51331863E 03  -0.53472567E 02 -0.31632940E 03   -0.25492326E 03  0.17973287E 03

-0.19832149E 03 -0.15276939E 03   0.60348285E 03  0.22326951E 03   -0.84949744E 03 -0.25264967E 04   -0.46784853E 04  0.29479513E 03
-0.15415803E 06 -0.30892515E 04  -0.47654415E 02  0.10984352E 04    0.15057917E 04 -0.29934659E 03

-0.29459288E 03 -0.87568346E 02   0.88325631E 03 -0.92717748E 02   -0.58574428E 02  0.47866358E 02   -0.44131865E 03  0.16832638E 03
 0.34630491E 03 -0.28094733E 01   0.72953255E 03 -0.14281007E 04   -0.31544211E 04 -0.66987806E 01

 0.14503499E 01 -0.65069821E 00  -0.43387116E 01  0.36982166E 01   -0.54134724E 00 -0.11367353E 02    0.11693199E 02 -0.35949843E 01
 0.58903664E 02 -0.34508222E 02  -0.10078286E 02  0.67480598E 02    0.10552976E 03 -0.26879910E 02
```

```
$       FORTRAN DECK
CCOMSYN      COMPONENT MODE SYNTHESIS PROGRAM
C
C      NCOMP = NO. OF COMPONENTS IN THE TOTAL SYSTEM (LIMITED TO 5).
C      MODE = NO. OF MODES DESIRED IN THE ANALYSIS OF THE TOTAL SYSTEM.
C      NCOM = NO. OF SETS OF COMMON JOINTS AMONG COMPONENTS IN SYSTEM.
C      NREDU = SIZE OF REDUCED MASS AND STIFFNESS MATRICES - NUMBER OF
C      DEGREES OF FREEDOM IN THE ANALYSIS OF EACH COMPONENT.
C      NMODE = NO. OF MODES IN EIGENVALUE SOLUTION OF EACH COMPONENT.
C      NCJT = NO. OF COMMON JOINTS ON EACH COMPONENT (LIMITED TO 12).
C      THE FOLLOWING INFORMATION IS NEEDED IF THERE IS AERODYNAMIC INPUT.
C      NCOUP = 0 IF NO AERODYNAMIC COUPLING EXISTS BETWEEN THE COMPONENTS
C      NCOUP = THE LOWER NUMBERED OF THE TWO COMPONENTS FOR WHICH THERE
C      IS AERODYNAMIC COUPLING (COUPLE  COMPONENTS MUST BE IN SEQUENCE)
C      NAT = AERODYNAMIC THEORY USED IN THE COMPONENT ANALYSIS
C      NAT = 1, AIC-S ARE FORMED BY NON-ZERO PARTITIONS (STRIP OR PISTON)
C      NAT = 2, AIC-S ARE FULL MATRICE  (KERNAL FUNCTION OR MACH BOX)
C      NV = NO. OF REDUCED VELOCITIES CONSIDERED FOR THE AERODYNAMICS.
C
       DIMENSION TITLE(24),NREDU(5),NCJT(5),NMODE(5),CKFF(97,97),
      1CMFF(97,97),CK12(97,36),CM12(97,36),CK22(36,36),CM22(36,36),
      2XMODE(97,9),XMODEN(97,9),T(97,36),XM(9,97),XMX(9,9),TTK1(36,36),
      3XKC(1035),TTM1(36,36),XMC(1035),W1(9,97),PMT(9,36),PM12(9,36),
      4TM(36,97),TMT(36,36),NCODE(8,45),XKS (9180),XMS (9180),A (9180),
      5ROOT(9),VALU(9),TEMP( 75),B( 97),C( 97),DUM3(135),F(135,3),
      6IDUM4( 75),PKT(9,36),W2(9,9),CKFFI(97,97),CMT(36,36)
      7,FREQ(9),ONE(9),GM(9,9),
      8AP(4,8),AIC(40,80),XF(97,9),XFT(9,97),VEL(20)
       INTEGER COM(48,4)
C
       EQUIVALENCE (CKFF(1,1),CKFFI(1,1),CMFF(1,1)),(CK12(1,1),CM12(1,1))
      1,(CK22(1,1),CM22(1,1)),(XMODE(1,1),XMODEN(1,1), XF(1,1)),
      2(TTK1(1,1),TTM1(1,1)),(XKS(1),XMS(1),A(1),           AIC(1,1))
      3,(PKT(1,1),PMT(1,1)),(XKC(1),XMC(1)),(XMX(1,1),W2(1,1),GM(1,1))
      4,(XM(1,1),W1(1,1),XFT(1,1))
C                               FORMAT
     1 FORMAT(1H0 25X,12A6//26X,12A6///)
     2 FORMAT(9I5)
     3 FORMAT(1H1 42X,33HCOMPONENT MODE SYNTHESIS ANALYSIS//48X,I1,22H CO
      1MPONENTS CONSIDERED//)
     4 FORMAT(1H0 46X,24HINPUT DATA FOR COMPONENT,I2///49X,I2,19H DEGREES
      1 OF FREEDOM/50X,I1,6H MODES/49X,I2,14H COMMON JOINTS//)
     5 FORMAT(1H0 2X,11HMODE SHAPES//)
     6 FORMAT(1H0 2X,4HMODE,I2,14H - F EQUENCY = F12.3,4H CPS//(3X,
      11P8E15.5))
     7 FORMAT(/// 3X,90HUPPER TRIANGLE OF REDUCED STIFFNESS MATRIX FOR CO
      1MPONENT (COMMON JOINTS RESTRAINED)    K-FF//)
     8 FORMAT(1H0 2X,3HROW I3 /(3X,1P8E15.5))
    11 FORMAT(/// 3X,83HSTIFFNESS MATRIX - RELATES COMMON JOINTS TO FREE
      1JOINTS (COMMON JOINTS FREE)    K-FJ//)
    13 FORMAT(/// 3X,57HUPPER TRIANGLE OF STIFFNESS MATRIX - COMMON JOINT
      1S    K-JJ//)
    16 FORMAT(/// 3X,87HUPPER TRIANGLE OF REDUCED WEIGHT MATRIX FOR COMPO
      1NENT (COMMON JOINTS RESTRAINED)    M-FF//)
    18 FORMAT(/// 3X,78HMASS MATRIX - RELATES COMMON JOINTS TO FREE JOINT
      1S (COMMON JOINTS FREE)    M-FJ//)
    21 FORMAT(/// 3X,52HUPPER TRIANGLE OF MASS MATRIX - COMMON JOINTS    M
      1-JJ//)
    23 FORMAT(/// 3X,40HNORMALIZED MODE SHAPES FOR ORTHOGONALITY//)
```

```
   25 FORMAT(/// 3X,23HCHECK FOR ORTHOGONALITY//3X,52HUPPER TRIANGLE OF
     1THE GENERALIZED MASS MATRIX IS NOW//)
   27 FORMAT(/// 3X,60HUPPER TRIANGLE OF TRANSFORMED STIFFNESS MATRIX FO
     1R COMPONENT I2//)
   29 FORMAT(/// 3X,55HUPPER TRIANGLE OF TRANSFORMED MASS MATRIX FOR COM
     1PONENT I2//)
   30 FORMAT(6F12.8)
   33 FORMAT(12A6)
   34 FORMAT(1H1)
   37 FORMAT (36I2)
   38 FORMAT(1H1 40X,22HAIC MATRICES FOR 1/K = 1P1E11.4)
   39 FORMAT(2I4)
   40 FORMAT(///  9HCOMPONENT I2 //)
   41 FORMAT(1H0 5HSTRIP I3 //)
   42 FORMAT(1H0 /(3X,2E14.6,2X,2E14.6,2X,2E14.6,2X,2E14.6))
   45 FORMAT(/// 33HCOUPLED AIC MATRIX FOR COMPONENTS I2,4H AND I2 //)
C                                 DISC ASSIGNMENTS
      KDISC=7
      MDISC=8
      IDISC=9
      JDISC=10
      KKDISC=11
      IKDISC=12
      IMDISC=13
      NMDISC=14
      MCDISC=15
      MTDISC=16
      MADISC=17
C                              READ INPUT DATA AND PRINT
 1000 READ(5,33)(TITLE(I),I=1,24)
      REWIND KDISC
      REWIND MDISC
      REWIND IDISC
      REWIND JDISC
      REWIND KKDISC
      REWIND IKDISC
      REWIND IMDISC
      REWIND NMDISC
      REWIND MCDISC
      REWIND MTDISC
      REWIND MADISC
      READ(5,2) NCOMP,MODE,NCOM,NCOUP,NAT,NV
      READ(5,2) (NREDU(I),I=1,NCOMP)
      READ(5,2) (NMODE(I),I=1,NCOMP)
      READ(5,2) (NCJT(I),I=1,NCOMP)
      READ(5,37)((COM(I,J),J=1,4),I=1,NCOM)
      IF(NV.EQ.0) GO TO 199
      READ(5,30)(VEL(I),I=1,NV)
  199 WRITE(6,3) NCOMP
      WRITE(6,1) (TITLE(I),I=1,24)
      DO 100 I=1,NCOMP
      N=NREDU(I)
      NC=NCJT(I)*3
      NM=NMODE(I)
      DO 200 K=1,NM
  200 READ(5,30) (XMODE(J,K),J=1,N)
      READ(5,30)(FREQ(L),L=1,NM)
      DO 210 J=1,N
  210 READ(5,30) (CKFF(J,K),K=1,N)
      DO 220 J=1,N
```

```
  220 READ(5,30) (CK12(J,K),K=1,NC)
      DO 230 J=1,NC
  230 READ(5,30) (CK22(J,K),K=1,NC)
      IF(I.EQ.1)GO TO 231
      WRITE(6,34)
  231 WRITE(6,4) I,N,NM,NCJT(I)
      WRITE(6,5)
      DO 9 K=1,NM
    9 WRITE(6,6)K,FREQ(K),(XMODE(J,K),J=1,N)
      WRITE(6,7)
      DO 12 L=1,N
   12 WRITE(6,8)L,(CKFF(L,J),J=L,N)
      WRITE(6,11)
      DO 14 L=1,N
   14 WRITE(6,8) L,(CK12(L,J),J=1,NC)
      WRITE(6,13)
      DO 15 L=1,NC
   15 WRITE(6,8) L,(CK22(L,J),J=L,NC)
      DO 240 J=1,N
  240 READ(5,30) (CKFFI(J,K),K=1,N)
C     GENERATE TRANSFORMED STIFFNESS MATRIX
      CALL MATMPL(CKFFI,CK12,T,97,97,97,36,97,36,N,NC,N,1)
      DO 46 J=1,N
      DO 10 K=1,NC
   10 T(J,K)=-1.*T(J,K)
   46 WRITE(MTDISC) (T(J,K),K=1,NC)
      CALL MATMPL(T,CK12,TTK1,97,36,97,36,36,36,NC,NC,N,2)
      DO 35 J=1,NC
      DO 35 K=1,NC
   35 TTK1(J,K)=TTK1(J,K)+CK22(J,K)
      NMC=NM+NC
      DO 32 L=1,NM
   32 ROOT(L)=(FREQ(L)*6.2831853)**2
      DO 36 J=1,NM
      DO 36 K=1,NC
   36 PKT(J,K)=0.0
      CALL GENC(NM,NMC,NC,XKC,TTK1,PKT,ROOT,IKDISC)
C     PRINT TRANSFORMED STIFFNESS MATRIX
      WRITE(6,27) I
      DO 28 L=1,NMC
      NS=(2*L+(L-1)*(2*NMC-L))/2
      NE=(2*NMC+(L-1)*(2*NMC-L))/2
   28 WRITE(6,8)L,(XKC(J)  ,J=NS,NE)
      DO 250 J=1,N
  250 READ(5,30) (CMFF(J,K),K=1,N)
      DO 260 J=1,N
  260 READ(5,30) (CM12(J,K),K=1,NC)
      DO 270 J=1,NC
  270 READ(5,30) (CM22(J,K),K=1,NC)
      WRITE(6,16)
      DO 17 L=1,N
   17 WRITE(6,8)L,(CMFF(L,J),J=L,N)
      WRITE(6,18)
      DO 19 L=1,N
   19 WRITE(6,8)L,(CM12(L,J),J=1,NC)
      WRITE(6,21)
      DO 22 L=1,NC
   22 WRITE(6,8) L,(CM22(L,J),J=L,NC)
C     NORMALIZE MODE SHAPES FOR ORTHOGONALITY
      DO 51 K=1,N
```

```
      DO 51 J=1,N
   51 CMFF(K,J)=CMFF(K,J)/(32.174*12.)
      CALL MATMPL(XMODE,CMFF,XM,97,9,97,97,9,97,NM,N,N,2)
      CALL MATMPL(XM,XMODE,XMX,9,97,97,9,9,9,NM,NM,N,1)
      DO 47 J=1,N
      DO 20 K=1,NM
   20 XMODEN(J,K)=XMODE(J,K)/SQRT(XMX(K,K))
   47 WRITE(MCDISC) (XMODEN(J,K),K=1,NM)
      CALL MATMPL(XMODEN,CMFF,W1,97,9,97,97,9,97,NM,N,N,2)
      CALL MATMPL(W1,XMODEN,W2,9,97,97,9,9,9,NM,NM,N,1)
      WRITE(6,23)
      DO 24 K=1,NM
   24 WRITE(6,6) K,FREQ(K),(XMODEN(J,K),J=1,N)
      WRITE(6,25)
      DO 26 L=1,NM
   26 WRITE(6,8) L,(W2(L,J),J=L,NM)
C     GENERATE TRANSFORMED MASS MATRIX
      CALL MATMPL(W1,T,PMT,9,97,97,36,9,36,NM,NC,N,1)
      CALL MATMPL(XMODEN,CM12,PM12,97,9,97,36,9,36,NM,NC,N,2)
      DO 55 J=1,NM
      DO 55 K=1,NC
   55 PMT(J,K)=PMT(J,K)+PM12(J,K)
      CALL MATMPL(T,CMFF,TM,97,36,97,97,36,97,NC,N,N,2)
      CALL MATMPL(TM,T,TMT,36,97,97,36,36,36,NC,NC,N,1)
      CALL MATMPL(T,CM12,TTM1,97,36,97,36,36,36,NC,NC,N,2)
      CALL MATMPL(CM12,T,CMT,97,36,97,36,36,36,NC,NC,N,2)
      DO 60 J=1,NC
      DO 60 K=1,NC
   60 TTM1(J,K)=TMT(J,K)+CM22(J,K)+TTM1(J,K)+CMT(J,K)
      DO 65 M=1,NM
   65 ONE(M)=1.0
      CALL GENC(NM,NMC,NC,XMC,TTM1,PMT,ONE,IMDISC)
C     PRINT TRANSFORMED MASS MATRIX
      WRITE(6,29) I
      DO 31 L=1,NMC
      NS=(2*L+(L-1)*(2*NMC-L))/2
      NE=(2*NMC+(L-1)*(2*NMC-L))/2
   31 WRITE(6,8)L,(XMC(J)   ,J=NS,NE)
  100 CONTINUE
      IF(NV.EQ.0) GO TO 101
C     FOR EACH REDUCED VELOCITY, READ AIC MATRIX FOR EACH COMPONENT
      DO 109 K=1,NV
      WRITE(MADISC) VEL(K)
      WRITE(6,38) VEL(K)
      II=0
      DO 108 I=1,NCOMP
      IF(II.EQ.I) GO TO 108
      N=NREDU(I)
      N2=2*N
      IF(NAT.EQ.2) GO TO 105
      READ(5,39) NSIZE,NPART
      WRITE(MADISC) NPART
      WRITE(6,40)I
      DO 104 J=1,NPART
      READ(5,39) NS
      NS2=2*NS
      WRITE(6,41)J
      DO 103 JJ=1,NS
      READ(5,30)(AP(JJ,KK),KK=1,NS2)
  103 WRITE(6,42)(AP(JJ,KK),KK=1,NS2)
```

```
      WRITE(MADISC)NS,NS2
  104 WRITE(MADISC)((AP(JJ,KK),KK=1,NS2),JJ=1,NS)
      GO TO 108
  105 IF(NCOUP.NE.I) GO TO 107
      II=I+1
      N=NREDU(I)+NREDU(II)
      N2=2*N
      WRITE(6,45) I,II
      GO TO 121
  107 WRITE(6,40) I
  121 DO 106 JA=1,N
      READ(5,30)(AIC(JA,KA),KA=1,N2)
      WRITE(MADISC)(AIC(JA,KA),KA=1,N2)
  106 WRITE(6,42)(AIC(JA,KA),KA=1,N2)
  108 CONTINUE
  109 CONTINUE
C     GENERATE THE NCODE MATRIX
  101 KK=0
      DO 110 I=1,NCOMP
      NM=NMODE(I)
      DO 110 J=1,NM
      KK=KK+1
  110 NCODE(I,J)=KK
      NTM=KK
      DO 120 I=1,NCOMP
      JM=NMODE(I)+1
      NCC=NMODE(I)+NCJT(I)
      DO 119 J=JM,NCC
      IF(I.EQ.1) GO TO 118
      DO 117 K=1,NCOM
      L=COM(K,4)
      IF(L.EQ.I) GO TO 115
      GO TO 117
  115 JC=J-JM+1
      LL=COM(K,3)
      IF(LL.EQ.JC) GO TO 116
      GO TO 117
  116 II=COM(K,2)
      JJ=COM(K,1)+NMODE(II)
      NCODE(I,J)=NCODE(II,JJ)
      GO TO 119
  117 CONTINUE
  118 KK=KK+1
      NCODE(I,J)=KK
  119 CONTINUE
  120 CONTINUE
      DO 130 I=1,NCOMP
      JM=NMODE(I)+NCJT(I)+1
      NCC=NMODE(I)+2*NCJT(I)
      DO 129 J=JM,NCC
      IF(I.EQ.1) GO TO 128
      DO 127 K=1,NCOM
      L=COM(K,4)
      IF(L.EQ.I) GO TO 125
      GO TO 127
  125 JC=J-JM+1
      LL=COM(K,3)
      IF(LL.EQ.JC) GO TO 126
      GO TO 127
  126 II=COM(K,2)
```

```
      JJ=COM(K,1)+NMODE(II)+NCJT(II)
      NCODE(I,J)=NCODE(II,JJ)
      GO TO 129
  127 CONTINUE
  128 KK=KK+1
      NCODE(I,J)=KK
  129 CONTINUE
  130 CONTINUE
      WRITE(6,131)
  131 FORMAT(1H1 2X,59HEFFECTIVE LOCATION OF COMPONENT D.O.F. IN THE SYS
     1TEM MATRIX//)
      DO 140 I=1,NCOMP
      JM=NMODE(I)+2*NCJT(I)+1
      NCC=NMODE(I)+3*NCJT(I)
      DO 139 J=JM,NCC
      IF(I.EQ.1) GO TO 138
      DO 137 K=1,NCOM
      L=COM(K,4)
      IF(L.EQ.I) GO TO 135
      GO TO 137
  135 JC=J-JM+1
      LL=COM(K,3)
      IF(LL.EQ.JC) GO TO 136
      GO TO 137
  136 II=COM(K,2)
      JJ=COM(K,1)+NMODE(II)+2*NCJT(II)
      NCODE(I,J)=NCODE(II,JJ)
      GO TO 139
  137 CONTINUE
  138 KK=KK+1
      NCODE(I,J)=KK
  139 CONTINUE
      WRITE(6,150) I,(NCODE(I,J),J=1,NCC)
  150 FORMAT(1H0 2X,19HNCODE FOR COMPONENT I2//(25I5))
  140 CONTINUE
C     THE FINAL KK BECOMES THE ORDER OF THE SYSTEM MATRICES-XKS AND XMS
C     GENERATE AND REDUCE SYSTEM MATRICES AND SOLVE EIGENVALUE PROBLEM
      CALL GENS(KK,NCOMP,NMODE,NCJT,NCODE,XKS,XKC,KDISC,IKDISC)
      CALL GENS(KK,NCOMP,NMODE,NCJT,NCODE,XMS,XMC,MDISC,IMDISC)
      NTOT=KK-NTM
      NREDUS=NTM+NTOT/3
      NROT=KK-NREDUS
      CALL EIGEN(A,VALU,TEMP,B,C,DUM3,F,IDUM4,IDISC,JDISC,KKDISC,MDISC,
     1KDISC,KK,MODE,MODE,NREDUS,NROT,NMDISC)
C     TRANSFORM SYSTEM MODE SHAPES BACK TO COMPONENTS
      CALL TMODE (KKDISC,MCDISC,MTDISC, IDISC,NMODE,NCJT,NREDU,NCOMP,
     1MODE,NV,NCODE,NTM,NREDUS,NROT,XF,B,C,DUM3)
C     GENERATE GENERALIZED MASS MATRIX FOR SYSTEM
      CALL GENM (IDISC,NMDISC,NREDUS,MODE,A,B,C,JDISC,DUM3,GM)
      IF(NV.EQ.0) GO TO 999
C     GENERATE GENERALIZED AERODYNAMIC FORCES FROM AIC MATRICES IF INPUT
      CALL GENA (KKDISC,MADISC,MTDISC,VEL,NCOMP,MODE,NCOUP,NAT,NV,NREDU,
     1AP,AIC,XF,XFT,B)
  999 GO TO 1000
      END
```

```
$      FORTRAN DECK
CEIGEN       REDUCES STIFFNESS MATRIX AND INVERTS IT, REDUCES MASS MATRIX
C            DETERMINES EIGENVALUES AND EIGENVECTORS FOR COMSYN
C        THE ARGUMENTS ARE=
C     A - VECTOR OF LENGTH NRDF*(NRDF+1)/2
C     VALU - VECTOR OF LENGTH NEIG
C        TEMP,B,C,DUM3, - VECTORS OF LENGTH NRDF OR NMASS (SMALLER)
C        E - MATRIX OF DIMENSION (NRDF,3)
C        IDUM4 - VECTOR OF LENGTH NRDF OR NMASS (SMALLER)
C        ITAPE,JTAPE, NTAPE, MTAPE, - THESE ARE VARIOUS TAPES
C        NRDF - NUMBER OF DEGREES OF FREEDOM OF THE SYSTEM
C        NEIG - NUMBER OF EIGENVALUES DESIRED
C        NVEC - NUMBER OF EIGENVECTORS DESIRED
C        NMASS=NO. OF NORMAL DISPLACEMENTS
C        NOMASS=NO. OF ROTATIONAL DEGREES OF FREEDOM
C        STIFF IS ON MTAPE IN COMPACT FORM
C        MASS IS ON NTAPE IN COMPACT FORM
         SUBROUTINE EIGEN(A,VALU,TEMP,B,C,DUM3,E,IDUM4,ITAPE,JTAPE,KTAPE,
        1NTAPE,MTAPE,NRDF,NEIG,NVEC,NMASS,NOMASS,NMTAPE)
         DIMENSION DUM3(NRDF),IDUM4(1),A(1),VALU(1),B(1),C(1),E(NRDF,3),
        1TEMP(1)
         INTEGER OUT
         OUT=6
         REWIND MTAPE
         REWIND NTAPE
         NTEMP=NMASS
         CALL DIVID(NMASS,NOMASS,MTAPE,JTAPE,ITAPE,A,B)
         CALL ZROMAK(A,B,C,DUM3,NMASS,NOMASS,ITAPE,JTAPE,MTAPE,KTAPE)
         CALL DIVID(NMASS,NOMASS,NTAPE,JTAPE,ITAPE,A,B)
         CALL ZROMAM(A,B,C,DUM3,NMASS,NOMASS,ITAPE,JTAPE,NTAPE,KTAPE)
         REWIND MTAPE
         REWIND NTAPE
         NREDU=NMASS
         NRMX=NREDU*(NREDU+1)/2
C        READ IN STIFFNESS MATRIX
         READ(MTAPE) (A(I),I=1,NRMX)
         WRITE(OUT,5500)
  5500   FORMAT(/// 3X,63HUPPER TRIANGLE OF REDUCED STIFFNESS MATRIX FOR TH
        1E TOTAL SYSTEM//)
         DO 5501 I=1,NREDU
         NS=(2*I+(I-1)*(2*NREDU-I))/2
         NE=(2*NREDU+(I-1)*(2*NREDU-I))/2
         WRITE(OUT,5502) I,(A(J),J=NS,NE)
  5502   FORMAT(1H0 2X,3HROW I4 /(3X,1P8E15.5))
  5501   CONTINUE
C        READ IN THE MASS MATRIX
         READ(NTAPE) (A(I),I=1,NRMX)
         DO 6012 I=1,NRMX
  6012   A(I)=A(I)*32.174*12.
         WRITE(OUT,5505)
  5505   FORMAT(/// 3X,60HUPPER TRIANGLE OF REDUCED WEIGHT MATRIX FOR THE T
        1OTAL SYSTEM//)
         DO 5506 I=1,NREDU
         NS=(2*I+(I-1)*(2*NREDU-I))/2
         NE=(2*NREDU+(I-1)*(2*NREDU-I))/2
  5506   WRITE(OUT,5502) I,(A(J),J=NS,NE)
         IF(NEIG.EQ.0) RETURN
         CALL EIGMAT(NTEMP,A,VALU,TEMP,B,C,DUM3,E,IDUM4,MTAPE,NTAPE,JTAPE,
        1ITAPE,NEIG,NVEC,NMTAPE)
         DO 60 I=1,NEIG
```

```
      IF(VALU(I).LT.0.0) GO TO 59
      DUM3(I)=SQRT(VALU(I))/6.2831853
      GO TO 60
   59 DUM3(I)=0.0
   60 CONTINUE
      WRITE(OUT,9009)
      WRITE(OUT,9005) (I,DUM3(I),I=1,NEIG)
 9009 FORMAT(/// 3X,33HNATURAL FREQUENCIES OF THE SYSTEM ///)
 9005 FORMAT( 3X,29HTHE NATURAL FREQUENCY NUMBER  I3,2X,2HIS F12.3,2X,
     13HCPS)
      RETURN
      END
```

```
*      FORTRAN DECK
CMATMPL     MATRIX MULTIPLICATION (REAL AND TWO-DIMENSIONAL)
C
C     MATRIX A DIMENSION (MA,NA) IN MAIN PROGRAM
C            B               (MB,NB)
C            C               (MC,NC)
C     M = NO. OF ROWS IN PRODUCT MATRIX C
C     N = NO. OF COLUMNS IN C
C     L = COMMON DIMENSION OF A AND B
C     IOP = 1, A X B = C
C           2, A(TRANSPOSE) X B = C
C           3, A X B(TRANSPOSE) = C
C
      SUBROUTINE MATMPL (A,B,C,MA,NA,MB,NB,MC,NC,M,N,L,IOP)
      DIMENSION A(MA,NA),B(MB,NB),C(MC,NC)
C
      GO TO (100,200,300),IOP
  100 DO 175 I=1,M
      DO 150 J=1,N
      C(I,J)=0.0
      DO 125 K=1,L
      C(I,J)=C(I,J)+A(I,K)*B(K,J)
  125 CONTINUE
  150 CONTINUE
  175 CONTINUE
      GO TO 400
  200 DO 275 I=1,M
      DO 250 J=1,N
      C(I,J)=0.0
      DO 225 K=1,L
      C(I,J)=C(I,J)+A(K,I)*B(K,J)
  225 CONTINUE
  250 CONTINUE
  275 CONTINUE
      GO TO 400
  300 DO 375 I=1,M
      DO 350 J=1,N
      C(I,J)=0.0
      DO 325 K=1,L
      C(I,J)=C(I,J)+A(I,K)*B(J,K)
  325 CONTINUE
  350 CONTINUE
  375 CONTINUE
  400 RETURN
      END
```

```
$      FORTRAN DECK
CCMULT       PRE AND POST MULTIPLIES A COMPLEX MATRIX BY REAL MATRICES
C     A IS THE PRE-MULTIPLIER (REAL) OF SIZE NR X NC
C     B IS THE COMPLEX MATRIX OF SIZE NC X ND IN REAL NOTATION
C     C IS THE POST-MULTIPIER (REAL) OF SIZE ND/2 X NR
C     D IS RESULTANT (COMPLEX) OF SIZE NR X 2NR IN REAL NOTATION
C     MR = NO. OF COLUMNS IN A AS DIMENSIONED IN MAIN PROGRAM
C        = NO  OF ROWS IN C AS DIMENSIONED IN MAIN PROGRAM
C     MA = NO. OF ROWS IN B AS DIMENSIONED IN MAIN PROGRAM
C     MB = NO. OF COLUMNS IN B AS DIMENSIONED IN MAIN PROGRAM
C
      SUBROUTINE CMULT(A,B,C,D,MR,MA,MB,NR,NC,ND)
C
      DIMENSION  A(9,MR),B(MA,MB),C(MR,9),D(9,18),E(9,80)
C
      NR2=2*NR
      NDH=ND/2
      DO 20 K=1,NR
      DO 20 L=1,ND
      E(K,L)=0.0
      DO 20 J=1,NC
   20 E(K,L)=E(K,L)+A(K,J)*B(J,L)
      DO 30 K=1,NR
      DO 25 L=1,NR2,2
      MM=(L+1)/2
      D(K,L)=0.0
      DO 25 J=1,NDH
   25 D(K,L)=D(K,L)+E(K,2*J-1)*C(J,MM)
      DO 30 L=2,NR2,2
      MM=L/2
      D(K,L)=0.0
      DO 30 J=1,NDH
   30 D(K,L)=D(K,L)+E(K,2*J)*C(J,MM)
      RETURN
      END
```

```
$     FORTRAN DECK
CGENC        GENERATES THE TRANSFORMED MASS OR STIFFNESS MATRIX
C                   FOR EACH COMPONENT
C      NMC = ORDER OF THE FINAL MATRIX (XC).
C      THE FINAL MATRIX IS STORED ON IDISC IN COMPACT FORM(BY ROWS)
C
       SUBROUTINE GENC(NM,NMC,NC,XC,TT1,PT,DIAG,IDISC)
       DIMENSION XC(1),      TT1(36,36),PT(9,36),DIAG(1)
C
       NMCT=NMC*(NMC+1)/2
       DO 40 J=1,NMCT
   40  XC(J)=0.0
       DO 45 M=1,NM
       J=(2*M+(M-1)*(2*NMC-M))/2
       MM=M-1
       MMM=J+NM-MM
       DO 44 K=1,NC
       XC(MMM)=PT(M,K)
   44  MMM=MMM+1
   45  XC(J)=DIAG(M)
       L=(2*(NM+1)+NM*(2*NMC-(NM+1)))/2
       DO 50 J=1,NC
       DO 50 K=J,NC
       XC(L)    = TT1(J,K)
       L=L+1
   50  CONTINUE
       WRITE(IDISC) (XC(I),I=1,NMCT)
       RETURN
       END
```

```
$       FORTRAN DECK
CGENS         GENERATES THE MASS OR STIFFNESS MATRIX FOR THE TOTAL SYSTEM
C                   BY THE NCODE METHOD.
C         KK = ORDER OF THE SYSTEM MATRIX (XS).
C         IDISC CONTAINS THE COMPONENT MATRIX IN COMPACT FORM BY ROWS (XC).
C         THE SYSTEM MATRIX IS STORED ON NDISC IN COMPACT FORM BY ROWS (XS).
C
          SUBROUTINE GENS (KK,NCOMP,NMODE,NCJT,NCODE,XS,XC,NDISC,IDISC)
          DIMENSION NMODE(1),NCJT(1),NCODE(5,45),XS(1),XC(1)
C
          REWIND NDISC
          REWIND IDISC
          KKK=KK*(KK+1)/2
          DO 200 I=1,KKK
      200 XS(I) = 0.0
          DO 500 I=1,NCOMP
          NMC=NMODE(I)+3*NCJT(I)
          NMCT=NMC*(NMC+1)/2
          READ(IDISC)(XC(K),K=1,NMCT)
          DO 400 II=1,NMC
          KI=NCODE(I,II)
          DO 375 JJ=II,NMC
          LI=NCODE(I,JJ)
          KA=KI
          LA=LI
          IF(LI.GE.KI) GO TO 370
          KI=LA
          LI=KA
      370 MO=(2*JJ+(II-1)*(2*NMC-II))/2
          NO=(2*LI+(KI-1)*(2*KK-KI))/2
          XS(NO)=XS(NO)+XC(MO)
          KI=KA
      375 CONTINUE
      400 CONTINUE
      500 CONTINUE
          DO 510 I=1,KK
          NS=(2*I+(I-1)*(2*KK-I))/2
          NE=(2*KK+(I-1)*(2*KK-I))/2
      510 WRITE(NDISC)(XS(J),J=NS,NE)
          RETURN
```

```
      END
$      FORTRAN DECK
CGENM        GENERATES THE GENERALIZED MASS MATRIX FOR THE TOTAL SYSTEM
C                    USED IN THE MODAL FLUTTER PROGRAM
C     N = SIZE OF REDUCED MASS MATRIX FOR THE SYSTEM
C     NVEC = NO. OF MODES
C     MSDISC - SYSTEM MODE SHAPES STORED
C     NDISC CONTAINS REDUCED MASS MATRIX OF THE SYSTEM
C     F = GENERALIZED MASS MATRIX - FORMED, PRINTED AND PUNCHED
C
      SUBROUTINE GENM (MSDISC,NDISC,N,NVEC,A,B,C,LLDISC,D,E)
      DIMENSION A(1),B(1),C(1),D(1),E(9,9),G(9)
C
   10 FORMAT(1H1 2X,44HGENERALIZED MASS MATRIX FOR THE TOTAL SYSTEM //)
   11 FORMAT (1H  1P9E14.5)
      REWIND NDISC
      REWIND MSDISC
      REWIND LLDISC
      NMAX = N*(N+1)/2
      READ(NDISC)(A(I),I=1,NMAX)
      DO 900  K=1,NVEC
      READ(MSDISC)(C(L),L=1,N)
      DO 800 I=1,N
      II=I-1
      IF(II.EQ.0) GO TO 600
      DO 595  J=1,II
      NU=(2*I+(J-1)*(2*I-J))/2+(J-1)*(N-I)
  595 B(J)=A(NU)
  600 CONTINUE
      NS=(2*I+(I-1)*(2*N-I))/2
      NE=(2*N+(I-1)*(2*N-I))/2
      J=I
      DO 650 JJ=NS,NE
      B(J)=A(JJ)
  650 J=J+1
      D(I)=0.0
      DO 750 LL=1,N
  750 D(I)=D(I)+C(LL)*B(LL)
  800 CONTINUE
      WRITE(LLDISC)(D(I),I=1,N)
  900 CONTINUE
      REWIND LLDISC
      DO 1000 K=1,NVEC
      REWIND MSDISC
      READ(LLDISC)(D(I),I=1,N)
      DO 950 KK=1,NVEC
      READ(MSDISC)(C(L),L=1,N)
      E(K,KK)=0.0
      DO 930 LL=1,N
  930 E(K,KK)=E(K,KK)+D(LL)*C(LL)
  950 CONTINUE
 1000 CONTINUE
      WRITE(6,10)
      DATA Q1/4HGENM/
      IC=0
      DO 20 I=1,NVEC
      WRITE (6,11)(E(I,J),J=1,NVEC)
      DO 15 J=1,NVEC
   15 G(J)=E(I,J)
      CALL PUNC (G,1,NVEC,Q1,IC)
   20 CONTINUE
```

```
      RETURN
      END
```

```
$      FORTRAN DECK
CTMODE        TRANSFORMS SYSTEM MODE SHAPES TO EACH COMPONENT
C
C      KKDISC CONTAINS (K-NN) X (K-MM) INVERSE
C      MCDISC CONTAINS THE MODE SHAPES FOR EACH COMPONENT
C      MTDISC CONTAINS THE T MATRIX FOR EACH COMPONENT
C      MSDISC CONTAINS THE MODE SHAPES FOR THE TOTAL SYSTEM
C      SM MATRIX - SYSTEM MODE SHAPES INVOLVING MODAL DEGREES OF FREEDOM.
C      STR MATRIX - SYSTEM MODE SHAPES INVOLVING TRANSLATIONAL AND
C      ROTATIONAL DEGREES OF FREEDOM.
C      XF - TRANSFORMED SYSTEM MODE SHAPES FOR EACH COMPONENT.
C      XF STORED ON KKDISC FOR SUBROUTINE GENA IF AERO INPUT
C
       SUBROUTINE TMODE (KKDISC,MCDISC,MTDISC,MSDISC,NMODE,NCJT,NREDU,
      1NCOMP,MODE,NV,NCODE,NTM,NREDUS,NROT,XF,B,C,D)
C
       DIMENSION NMODE(1),NCJT(1),NREDU(1),SM(45,9),STR(90,9),NCODE(5,45)
      1,XF(97,9),B(1),C(1),D(1)
C
       DATA Q5/4HSYSM/
  300 FORMAT(/// 3X,47HSYSTEM MODE SHAPES FOR FREE JOINTS ON COMPONENT
      1I2//)
  301 FORMAT(1H 9E14.5)
  302 FORMAT(// 6X,6HMODE 1,8X,6HMODE 2,8X,6HMODE 3,8X,6HMODE 4,8X,
      16HMODE 5,8X,6HMODE 6,8X,6HMODE 7,8X,6HMODE 8,8X,6HMODE 9//)
       REWIND KKDISC
       REWIND MCDISC
       REWIND MSDISC
       REWIND MTDISC
C      GENERATE XF, TRANFORMED SYSTEM MODE SHAPES TO EACH COMPONENT
       NT=NTM+1
       DO 20 J=1,MODE
       READ(MSDISC)(B(K),K=1,NREDUS)
       DO 11 K=1,NTM
   11  SM(K,J)=B(K)
       L=0
       DO 12 K=NT,NREDUS
       L=L+1
   12  STR(L,J)=B(K)
       DO 15 M=1,NROT
       D(M)=0.0
       DO 14 N=1,NREDUS
       READ(KKDISC)(C(KK),KK=1,NROT)
   14  D(M)=D(M)+B(N)*C(M)
       REWIND KKDISC
       L=L+1
       STR(L,J)=-D(M)
   15  CONTINUE
   20  CONTINUE
       REWIND KKDISC
       NNM=0
       DO 100 I=1,NCOMP
       N=NREDU(I)
       NM=NMODE(I)
       NO=NNM+1
       NNM=NO+NM-1
       NC=3*NCJT(I)
       JJ=NM+1
       JK=NM+NC
       DO 30 K=1,N
```

```
      READ(MCDISC)(B(KK),KK=1,NM)
      DO 26 L=1,MODE
      D(K)=0.0
      DO 25 J=NO,NNM
      M=J-NO+1
   25 D(K)=D(K)+B(M)*SM(J,L)
      XF(K,L)=D(K)
   26 CONTINUE
      READ(MTDISC)(B(KK),KK=1,NC)
      DO 35 L=1,MODE
      C(K)=0.0
      DO 29 JL=JJ,JK
      M=JL-NM
      JM=NCODE(I,JL)-NTM
   29 C(K)=C(K)+B(M)*STR(JM,L)
      XF(K,L)=XF(K,L)+C(K)
   35 CONTINUE
   30 CONTINUE
      IF(NV.EQ.0) GO TO 39
      WRITE(KKDISC)((XF(K,L),L=1,MODE),K=1,N)
   39 WRITE(6,300)I
      WRITE(6,302)
      DO 40 K=1,N
   40 WRITE(6,301)(XF(K,L),L=1,MODE)
  100 CONTINUE
      RETURN
      END
```

```
$      FORTRAN DECK
CSYMINV
C      A  IS THE UPPER TRIANGLE OF THE SYMMETRIC MATRIX TO BE INVERTED.  SYMV  CC
C      ELEMENTS ARE STORED ROWWISE.
C      N = ORDER OF MATRIX                                               SYMV  CC
C      PROGRAM INVERTS IN PLACE.                                         SYMV  CC
       SUBROUTINE SYMINV(A,N)                                            SYMV  08
       DIMENSION A(1)                                                    SYMV  18
       CALL ECLOCK(IT1)
       NMAX=N*(N+1)/2
       IF(A(1).LT.0.0) GO TO 25
       GO TO 99
   25  II=1
   26  WRITE(6,27)II
   27  FORMAT(1H1,5X,36HA NEGATIVE VALUE APPEARS IN ELEMENT       ,I5,1X,
      125HOF VECTOR TO BE INVERTED,/6X 69HSINCE ELEMENT FALLS ON DIAGONAL
      2, MATRIX IS NOT POSITIVE DEFINITE,//6X,38HPROGRAM ENDED AND JOB DE
      3LETED.)
       DO 45 I=1,N
       NS=(2*I+(I-1)*(2*N-I))/2
       NE=(2*N+(I-1)*(2*N-I))/2
       WRITE(6,28)I,(A(J),J=NS,NE)
   28  FORMAT(/3HROW,I4/(9F14.5))
   45  CONTINUE
       CALL EXIT
   99  CONTINUE
       A(1)=SQRT(A(1))
       DO 100 IJ=2,N                                                     SYMV  30
  100  A(IJ)=A(IJ)/A(1)                                                  SYMV  40
       A(1)=1.0/A(1)                                                     SYMV  50
       IM1=1                                                             SYMV  60
       IJ=N
       DO 1000 I=2,N
       II=IJ+1
       IJ=II
       DO 200 J=I,N
       JM1=J-I
       LI=I
       LJ=J
       DO 120 L=1,IM1
       A(IJ)=A(IJ)-A(LI)*A(LJ)
       II=LI+N-I
  120  LJ=LI+JM1
  200  IJ=IJ+1
       IF(A(II).LT.0.0) GO TO 26
       A(II)=SQRT(A(II))
       JI=I
       JJ=1
       DO 500 J=1,IM1
       A(JI)=A(JJ)*A(JI)
       IF(J-IM1)300,420,420
  300  JP1=J+1
       JL=JJ
       LI=JI
       DO 400 L=JP1,IM1
       JL=JL+1
       LI=LI+N-I+1
  400  A(JI)=A(JI)+A(JL)*A(LI)
  420  A(JI)=-A(JI)/A(II)
```

```
      JI=JI+N-J
  500 JJ=JJ+N-J+1
      IF(I-N)600,900,900
  600 IP1=I+1
      IJ=II
      DO 700 J=IP1,N
      IJ=IJ+1
  700 A(IJ)=A(IJ)/A(II)
  900 A(II)=1.0/A(II)
 1000 IM1=I
      II=1
      DO 2000 I=1,N
      JJ=II
      IJ=II
      DO 1400 J=I,N
      A(IJ)=A(IJ)*A(JJ)
      JP1=J+1
      IF(JP1-N)1100,1100,1400
 1100 IL=IJ
      JL=JJ
      DO 1200 I=JP1,N
      IL=IL+1
      JL=JL+1
 1200 A(IJ)=A(IJ)+A(IL)*A(JL)
      JJ=JL+1
 1400 IJ=IJ+1
 2000 II=IJ
      CALL ECLOCK(IT2)
      TIME = FLOAT(IT2-IT1)/64000.
      WRITE(6,3000) TIME
 3000 FORMAT(1H0 39HTHE TIME ELAPSED FOR MATRIX INVERSION = E12.4,1X,7HS
     1ECONDS)
      RETURN
      END
```

```
$      FORTRAN DECK
CDIVID
C      N=NO. OF NORMAL DISPLACEMENTS
C      M=NO. OF ROTATIONAL  D.O.F.
C      NTPE-CONTAINS STIFFNESS (OR MASS) MATRIX
C      MTPE-K12 (M12) STORED
C      ITPE-K11 (M11) STORED
C      A- DUMMY  STORAGE VECTOR,LARGER  OF (N*(N+1)/2 OR M*(M+1)/2)
       SUBROUTINE DIVID (N,M,NTPE,MTPE,ITPE,A,B)
       DIMENSION A(1),B(1)
       REWIND ITPE
       REWIND NTPE
       REWIND MTPE
       NMAX=N*(N+1)/2
       MMAX=M*(M+1)/2
       NM=N+M
       ICNT=0
       DO 10 I=1,N
       II=NM-I+1
       READ(NTPE) (B(J),J=1,II)
       ID=II-M
       DO 20 J=1,ID
       ICNT=ICNT+1
 20    A(ICNT)=B(J)
       ID1=ID+1
       JCNT=0
       DO 30    J=ID1,II
       JCNT=JCNT+1
 30    B(JCNT)=B(J)
       WRITE(MTPE) (B(J),J=1,M)
 10    CONTINUE
       WRITE(ITPE) (A(J),J=1,NMAX)
       REWIND MTPE
       REWIND ITPE
       ID=0
       ICNT=0
       DO 50    I=1,M
       II=M-ICNT
       READ(NTPE) (B(J),J=1,II)
       ICNT=ICNT+1
       DO 60  J=1,II
       ID=ID+1
 60    A(ID)=B(J)
 50    CONTINUE
       RETURN
       END
```

```
$       FORTRAN DECK
CEIGMAT      FOR CONSYN
C       THIS SUBROUTINE FINDS THE EIGENVALUES AND EIGENVECTORS FOR
C       SYMMETRIC MASS AND STIFFNESS MATRICES.
C       THE ARGUMENTS ARE--
C         N- ORDER OF MATRICES.
C         A-  DUMMY VECTOR WITH DIMENSION IN MAIN PROGRAM OF N*(N+1)/2
C         VALU-  STORAGE FOR EIGENVALUES. MUST BE DIMENSIONED IN THE MAIN
C                PROGRAM AS A VECTOR OF LENGTH NEIG.
C         TEMP,B,C,D,-  DUMMY VECTORS WITH DIMENSION OF N IN MAIN PROGRAM.
C         E-  DUMMY ARRAY WITH DIMENSIONS OF (N,3) IN MAIN PROGRAM.
C         IDUM-  DUMMY INTEGER VECTOR WITH DIMENSION OF N IN MAIN PROGRAM.
C         MTAPE-  TAPE WHERE STIFFNESS MATRIX IS STORED IN COMPACT FORM.
C         NTAPE-  TAPE WHERE MASS MATRIX IS STORED IN COMPACT FORM.
C         JTAPE,ITAPE-  SCRATCH TAPES.
C         NEIG-  NUMBER OF EIGENVALUES DESIRED.
C         NVEC-  NUMBER OF EIGENVECTORS DESIRED. MUST BE EQUAL TO OR LESS
C                THAN NEIG.
C       THE MASS AND STIFFNESS MATRICES ARE STORED IN COMPACT FORM AS
C       VECTORS. ONLY THE UPPER TRIANGLE OF THESE MATRICES(BY ROWS) IS
C       STORED.
        SUBROUTINE EIGMAT(N,A,VALU,TEMP,B,C,D,E,IDUM,MTAPE,NTAPE,JTAPE,
       1ITAPE,NEIG,NVEC,NMTAPE)
        DIMENSION A(1),TEMP(1),VALU(1),B(1),C(1),D(1),E(N,3),IDUM(1)
        DOUBLE PRECISION SUM,SUM1
        INTEGER OUT
        OUT=6
        REWIND ITAPE
        REWIND JTAPE
        REWIND NTAPE
        REWIND MTAPE
        REWIND NMTAPE
        M=2*N
        NMAX=N*(N+1)/2
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C       STEP 1
C       READ IN M BY ROWS IN COMPACTED FORM
C       REPLACE M BY (L)TRANSPOSE, WHERE M=L*(L)TRANSPOSE, SAVE M ON NMTAPE
C       CALCULATE FIRST ROW
        READ (NTAPE) (A(I),I=1,NMAX)
        WRITE (NMTAPE)(A(I),I=1,NMAX)
        REWIND NTAPE
      5 CONTINUE
        A(1)=SQRT(A(1))
        DO 10 I=2,N
     10 A(I)=A(I)/A(1)
C     CALCULATE ALL THE OTHER ROWS
        IND=N
        DO 101 I=2,N
        IND=IND+1
        SUM=0.D0
        K1=I-1
        DO 50 JJ=1,K1
        MJ=(M-JJ)*(JJ-1)/2+I
     50 SUM=SUM+A(MJ)*A(MJ)
        A(IND)=DSQRT(A(IND)-SUM)
        IF(IND.EQ.NMAX) GO TO 100
        SUM1=A(IND)
        K1=I+1
        DO 99 J=K1,N
```

```
      IND=IND+1
      SUM=0.D0
      II=I-1
      DO 60 JJ=1,II
      K=(M-JJ)*(JJ-1)/2
      KI=K+I
      KJ=K+J
   60 SUM=SUM+A(KI)*A(KJ)
      A(IND)=(A(IND)-SUM)/SUM1
   99 CONTINUE
  100 CONTINUE
  101 CONTINUE
C     CHECK FOR SINGULAR MASS MATRIX
      DO 102 I=1,N
      KI=(M-I)*(I-1)/2+I
      IF(A(KI).EQ.0.) GO TO 1090
  102 CONTINUE
C     THIS COMPLETES STEP 1
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     STEP 2
C     WRITE (L)TRANSPOSE ON TAPE BY COLUMNS
C     PUT (L)TRANSPOSE INTO TEMPORARY STORAGE (TEMP--A VECTOR)
C     AND THEN WRITE TEMP ON TAPE
      KTAPE=NTAPE
  300 IND=0
      DO 340 J=1,N
      DO 330 I=1,J
      IND=IND+1
      MI1=(M-I)*(I-1)/2+J
      TEMP(IND)=A(MI1)
  330 CONTINUE
      WRITE(KTAPE) (TEMP(JJ),JJ=1,IND)
      IND=0
  340 CONTINUE
C     THIS COMPLETES STEP 2
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     STEP 3
C     ((L)TRANSPOSE) INVERSE REPLACES (L)TRANSPOSE IN CORE
C     REPLACEMENT IS DONE BY LAST COLUMN FIRST--WORKING UP THE COLUMN
      DO 410 I=1,N
      IND=(I*(M+3-I))/2-N
  410 A(IND)=1./A(IND)
      DO 499 J=2,N
      JJ=(N+2)-J
      DO 490 I=2,JJ
      IND=(N+J+I-3)*(JJ-I)/2
      SUM=0.D0
      K1=JJ-I+2
      DO 450 K=K1,JJ
      IDK=IND+K
      MK=(M-K)*(K-1)/2+JJ
  450 SUM=SUM+A(IDK)*A(MK)
      IND=IND+JJ
      IDI=IND-I+1
  490 A(IND)=-SUM*A(IDI)
  499 CONTINUE
C     END OF STEP 3
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     STEP 4
C     U=((L)TRANSPOSE)INVERSE
```

```
C     WRITE U ON TAPE BY ROWS
      WRITE(ITAPE) (A(I),I=1,NMAX)
C    FINISHED WITH STEP 4
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     STEP 5
C     WRITE U ON TAPE BY COLUMNS STARTING WITH THE LAST COLUMN FIRST
C     PUT U (LAST COLUMN FIRST) INTO TEMP AND THEN WRITE ON TAPE
      IND=0
      DO 555 K=1,N
      J=N-K+1
      DO 550 I=1,J
      IND=IND+1
      MI2=(M-I)*(I-1)/2+J
      TEMP(IND)=A(MI2)
  550 CONTINUE
      WRITE(JTAPE) (TEMP(JJ),JJ=1,IND)
      IND=0
  555 CONTINUE
C    END OF STEP 5
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C    STEP 6
C    FORM KU
C    READ K INTO CORE
C    READ U INTO CORE A COLUMN AT A TIME IN REVERSE ORDER
C    REPLACE K BY KU COLUMN BY COLUMN STARTING WITH THE LAST COLUMN
C    AND WORKING UP THE COLUMN
      READ(MTAPE) (A(I),I=1,NMAX)
      REWIND JTAPE
      DO 690 JJ=1,N
      J=N+1-JJ
      READ(JTAPE) (TEMP(II),II=1,J)
      DO 690 II=1,J
      I=J+1-II
      SUM=0.D0
      DO 650 K=1,I
      MK1=(M-K)*(K-1)/2+I
  650 SUM=SUM+A(MK1)*TEMP(K)
      IND=(M-I)*(I-1)/2+J
      IF(I.EQ.J) GO TO 680
      K1=(M-I)*(I-1)/2
      I=I+1
      DO 660 K=I,J
      K1K=K1+K
  660 SUM=SUM+A(K1K)*TEMP(K)
  680 CONTINUE
      A(IND)=SUM
  690 CONTINUE
C    END OF STEP 6
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C    STEP 7
C     FORM((L)INVERSE)*KU
C     KU IS IN CORE
C     READ IN L COLUMN BY COLUMN AND CALCULATE ((L)INVERSE)*KU
C     ROW BY ROW
C    CALCULATE THE FIRST ROW
      REWIND NTAPE
      READ(NTAPE) TEMP(1)
      DO 710 I=1,N
  710 A(I)=A(I)/TEMP(1)
C     NOW CALCULATE THE REST OF THE ROWS
```

```
      IND=N
      DO 799 I=2,N
      READ (NTAPE) (TEMP(JJ),JJ=1,I)
      DO 799 J=I,N
      IND=IND+1
      JJ=I-1
      SUM=0.D0
      DO 750 K=1,JJ
      MK2=(M-K)*(K-1)/2+J
  750 SUM=SUM+TEMP(K)*A(MK2)
  799 A(IND)=(A(IND)-SUM)/TEMP(I)
C     STEP 7 IS COMPLETE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     STEP 8
C     DETERMINE EIGENVALUES AND EIGENVECTORS OF THE NEW MATRIX
C      CHANGE THE SIGN OF A IN ORDER TO OBTAIN THE SMALLEST
C      EIGENVALUE FIRST
      DO 800 I=1,NMAX
  800 A(I)=-A(I)
      CALL BIGMAT(A,VALU,TEMP,B,C,D,E,IDUM,N,NEIG,NVEC,MTAPE)
C      CHANGE VALU BACK
      DO 850 I=1,NEIG
  850 VALU(I)=-VALU(I)
C     STEP 8 IS COMPLETE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      STEP 9
C     CHANGE EIGENVECTORS BACK
C     READ U INTO CORE BY ROWS
C     READ UNCHANGED EIGENVECTORS INTO CORE ONE AT A TIME
C      CHANGE AND PRINT EIGENVECTORS
      IF(NVEC.EQ.0) GO TO 2000
      WRITE(OUT,4001)
      REWIND ITAPE
      READ(ITAPE) (A(I),I=1,NMAX)
      REWIND MTAPE
      REWIND ITAPE
      DO 999 JJ=1,NVEC
      READ(MTAPE) (TEMP(I),I=1,N)
      IND=0
      DO 910 I=1,N
      SUM=0.D0
      DO 909 J=I,N
      IND=IND+1
  909 SUM=SUM+A(IND)*TEMP(J)
  910 TEMP(I)=SUM
C      NORMALIZE THE EIGENVECTOR
      SUM=TEMP(1)
      DO 939 II=2,N
      IF(ABS(SUM)-ABS(TEMP(II))) 938,939,939
  938 SUM=TEMP(II)
  939 CONTINUE
      IF(SUM) 940,947,940
  940 CONTINUE
      DO 941 II=1,N
      TEMP(II)=TEMP(II)/SUM
  941 CONTINUE
  947 CONTINUE
      WRITE (ITAPE)(TEMP(I),I=1,N)
  999 WRITE(OUT,4000) JJ,VALU(JJ),(TEMP(I),I=1,N)
C     STEP 9 IS COMPLETE
```

```
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      GO TO 2000
 4000 FORMAT(1H0 2X,18HEIGENVECTOR NUMBER I5/12X,17H CORRESPONDING TO
     11PE15.7/(1H  2X,1P6E15.7))
 4001 FORMAT(1H1 2X,42HEIGENVALUES AND EIGENVECTORS OF THE SYSTEM ///)
 4002 FORMAT(1H1,38X,27HTHE MASS MATRIX IS SINGULAR   ///)
 1090 WRITE(OUT,4002)
 2000 RETURN
      END
```

```
$       FORTRAN DECK
CZROMAK      GENERATES REDUCED STIFFNESS MATRIX FOR CONSYN
C
C       D IS A DUMMY VECTOR WITH STORAGE N OR M (LARGER)
C       A IS A DUMMY VECTOR WITH STORAGE  N*(N+1)/2  OR M*(M+1)/2 (LARGER)
C       B IS A DUMMY VECTOR WITH STORAGE  N OR M  (LARGER)
C       C IS A DUMMY VECTOR WITH STORAGE  N OR M  (LARGER)
C       N=NO. OF NORMAL DISPLACEMENTS
C       M=NO. OF ROTATIONAL  D.O.F.
C       NTPE CONTAINS  K11 MATRIX
C       MTPE CONTAINS  K12 MATRIX
C       ITPE SCRATCH  TAPE
C       KTPE STORES K12*K22**(-1)
C       A INITIALLY CONTAINS K22
C***    REDUCED STIFFNESS MATRIX IS STORED ON ITPE
        SUBROUTINE ZROMAK(A,B,C,D,N,M,NTPE,MTPE,ITPE,KTPE)
        DIMENSION A(1),B(1),C(1),D(1)
        DOUBLE PRECISION SUM,DP1,DP2
        CALL SYMINV( A,M)
        REWIND MTPE
        REWIND ITPE
        REWIND NTPE
        REWIND KTPE
        NMAX=N*(N+1)/2
        MMAX=M*(M+1)/2
        DO 10   KK=1,N
        READ(MTPE) (B(I),I=1,M)
        ICNT=0
        DO 1000  IK=1,M
        JJ=IK
        JK=IK
        DO 20    J=JJ,M
        ICNT=ICNT+1
 20     C(J)=A(ICNT)
        JJ=JJ-1
        JA=M
        ID=IK
        DO 30 J=1,JJ
        IF(JJ.EQ.0) GO TO 30
        C(J)=A(ID)
        JA=JA-1
        ID=ID+JA
 30     CONTINUE
        SUM=0.0D0
        DO 50    J=1,M
        DP1=B(J)
        DP2=C(J)
 50     SUM=SUM+DP1*DP2
        D(JK)= SUM
 1000   CONTINUE
        WRITE (ITPE)  (D(J),J=1,M)
        WRITE (KTPE) (D(J),J=1,M)
 10     CONTINUE
        REWIND ITPE
        REWIND MTPE
        REWIND NTPE
        REWIND KTPE
        READ (NTPE) (A(J),J=1,NMAX)
        ICNT=0
        DO 60  KK=1,N
```

```
      READ (ITPE) (D(J),J=1,M)
      KI=KK
      DO 70   KJ=1,N
      READ(MTPE)(C(J),J=1,M)
      KP=KJ
      IF(KP.LT.KI)  GO TO  70
      SUM=0.0D0
      DO 80   KR=1,M
      DP1=D(KR)
      DP2=C(KR)
   80 SUM=SUM +DP1*DP2
      ICNT=ICNT+1
      SM=SUM
      A(ICNT)=A(ICNT)-SM
   70 CONTINUE
      REWIND MTPE
   60 CONTINUE
      REWIND NTPE
      REWIND MTPE
      REWIND ITPE
      WRITE(ITPE) (A(I),I=1,NMAX)
      REWIND ITPE
      RETURN
      END
```

```
$       FORTRAN DECK
CZROMAM
C      N=NO. OF NORMAL DISPLACEMENTS
C      M=NO. OF ROTATIONAL  D.O.F.
C      NTPE CONTAINS M11 MATRIX
C      MTPE CONTAINS M12 MATRIX
C      ITPE SCRATCH TAPE
C      KTPE CONTAINS K12*K22**(-1)
C***   REDUCED MASS MATRIX IS STORED ON ITPE
       SUBROUTINE ZROMAM(A,B,C,D,N,M,NTPE,MTPE,ITPE,KTPE)
       DIMENSION A(1),B(1),C(1),D(1)
       DOUBLE PRECISION SUM1,SUM2,DP1,DP2,DP3
       NMASS=N
       REWIND MTPE
       REWIND ITPE
       REWIND NTPE
       REWIND KTPE
       NMAX=N*(N+1)/2
       DO 10 KK=1,N
       READ(KTPE) (B(I),I=1,M)
       ICNT=0
       DO 1000 IK=1,M
       JJ=IK
       JK=IK
       DO 20 J=JJ,M
       ICNT=ICNT+1
   20  C(J)=A(ICNT)
       JJ=JJ-1
       JA=M
       ID=IK
       DO 30 J=1,JJ
       IF(JJ.EQ.0) GO TO 30
       C(J)=A(ID)
       JA=JA-1
       ID=ID+JA
   30  CONTINUE
       SUM1=0.D0
       DO 50 J=1,M
       DP1=B(J)
       DP2=C(J)
   50  SUM1=SUM1+DP1*DP2
       D(JK)=SUM1
 1000  CONTINUE
       WRITE(ITPE) (D(J),J=1,M)
   10  CONTINUE
       REWIND ITPE
       REWIND MTPE
       REWIND NTPE
       REWIND KTPE
       READ(NTPE) (A(J),J=1,NMAX)
       DO 60 KK=1,N
       READ(MTPE) (B(J),J=1,M)
       READ(ITPE) (D(J),J=1,M)
       DO 70 KJ=1,N
       READ(KTPE) (C(J),J=1,M)
       SUM1=0.D0
       SUM2=0.D0
       DO 80 KR=1,M
       DP1=B(KR)
       DP2=D(KR)
```

```
      DP3=C(KR)
      SUM1=SUM1+DP1*DP3
   80 SUM2=SUM2+DP2*DP3
      SM1=SUM1
      SM2=SUM2
      IF(KJ.GE.KK) MM=(2*KJ+(KK-1)*(2*NMASS-KK))/2
      IF(KJ.GE.KK) A(MM)=A(MM)-SM1+SM2
      IF(KJ.LE.KK) MM=(2*KK+(KJ-1)*(2*NMASS-KJ))/2
      IF(KJ.LE.KK) A(MM)=A(MM)-SM1
   70 CONTINUE
      REWIND KTPE
   60 CONTINUE
      REWIND NTPE
      REWIND MTPE
      REWIND ITPE
      REWIND KTPE
      WRITE(ITPE) (A(I),I=1,NMAX)
      REWIND ITPE
      RETURN
      END
```

```
$       FORTRAN DECK
CPUNC         PUNCHES FULL MATRIX IN (1P6E12.5) FORMAT AND SEQUENCES CARDS
C
C      THE CALL PUNC STATEMENT MUST BE IN A LOOP.
C      EACH ROW STARTS ON A NEW CARD.
C      A IS THE ROW VECTOR TO BE PUNCHED.
C      NS IS THE FIRST ELEMENT OF A TO BE PUNCHED.
C      NE IS THE LAST ELEMENT OF A TO BE PUNCHED.
C      Q IS THE ALPHANUMERIC IDENTIFICATION CODE FOR THE MATRIX.
C      IC IS THE SEQUENCE NUMBER FOR THE FIRST CARD LESS ONE.
C      IC = 0, IF THE FIRST CARD FOR THE MATRIX IS TO BE SEQUENCED 1.
C
      SUBROUTINE PUNC (A,NS,NE,Q,IC)
      DIMENSION A(1)
C
    1 FORMAT(1P1E12.5,60X,1A4,I4)
    2 FORMAT(1P2E12.5,48X,1A4,I4)
    3 FORMAT(1P3E12.5,36X,1A4,I4)
    4 FORMAT(1P4E12.5,24X,1A4,I4)
    5 FORMAT(1P5E12.5,12X,1A4,I4)
    6 FORMAT(1P6E12.5,1A4,I4)
C
      NT=NE-NS+1
      N6=NT/6
      NC=N6*6
      N1=NS
      N2=N1+5
      IF(NT.LT.6) GO TO 20
      DO 10 J=1,N6
      IC=IC+1
      PUNCH6,(A(I),I=N1,N2),Q,IC
      N1=N2+1
   10 N2=N2+6
      IF(NT.EQ.NC) GO TO 50
   20 NO=NT-NC
      IC=IC+1
      GO TO(21,22,23,24,25),NO
   21 PUNCH 1,(A(I),I=N1,NE),Q,IC
      GO TO 50
   22 PUNCH 2,(A(I),I=N1,NE),Q,IC
      GO TO 50
   23 PUNCH 3,(A(I),I=N1,NE),Q,IC
      GO TO 50
   24 PUNCH 4,(A(I),I=N1,NE),Q,IC
      GO TO 50
   25 PUNCH 5,(A(I),I=N1,NE),Q,IC
   50 RETURN
      END
```

```
$      FORTRAN DECK
CBIGMAT
C PROG.AUTHORS M.ELSON AND R.E.FUNDERLIC,CENTRAL DATA PROCESSING,4,1,65 BIGM0003
      SUBROUTINE BIGMAT(A,VALU,VALL,UPPERD,DIAG,V,T,INTER,NN,NEIG,NVEC,
     1MTAPE)
      DIMENSION A(1),VALU(1),VALL(1),UPPERD(1),DIAG(1),V(1),T(NN,3),
     1INTER(1)
      REWIND MTAPE
      NZ=0                                                              BIGM0007
      N=NN                                                              BIGM0008
      IF(N.LE.2)GO TO 49
      NP1=N+1                                                           BIGM0010
      NM1=N-1                                                           BIGM0011
      NM2=N-2                                                           BIGM0012
      NT2P1=N*2+1                                                       BIGM0013
      IX=0                                                              BIGM0014
      DO 10 I=1,NM2                                                     BIGM0015
      SIGMA2=0.                                                         BIGM0016
      IP1=I+1                                                           BIGM0017
      DO 1 J=IP1,N                                                      BIGM0018
      IJ=IX+J                                                           BIGM0019
    1 SIGMA2=SIGMA2+A(IJ)**2                                            BIGM0020
      SIGMA=SQRT(SIGMA2)                                                BIGM0021
      II=IX+I                                                           BIGM0022
      DIAG(I)=A(II)                                                     BIGM0023
      IIP1=IX+I+1                                                       BIGM0024
      UPPERD(I)=-SIGN(SIGMA,A(IIP1))                                    BIGM0025
      T(I,2)=SIGMA2
      IF(ABS(SIGMA).GT.ABS(A(IIP1)))GO TO 2                             BIGM0027
      UPPERD(I)=A(IIP1)                                                 BIGM0028
      A(IIP1)=0.                                                        BIGM0029
      GO TO 10                                                          BIGM0030
    2 A(IIP1)=SQRT(1.+ABS(A(IIP1))/SIGMA)                               BIGM0031
      SQTGAM=-SIGN(SIGMA*A(IIP1),UPPERD(I))                             BIGM0032
      IP2=I+2                                                           BIGM0033
      DO 3 J=IP2,N                                                      BIGM0034
      IJ=IX+J                                                           BIGM0035
    3 A(IJ)=A(IJ)/SQTGAM                                                BIGM0036
      JK1=I*(2*N-I-1)/2                                                 BIGM0037
      JX=JK1                                                            BIGM0038
      IIX=JK1                                                           BIGM0039
      DO 5 J=IP1,N                                                      BIGM0040
      VALL(J)=0.
      JK=JK1+J                                                          BIGM0042
      DO 4 K=IP1,J                                                      BIGM0043
      IK=IX+K                                                           BIGM0044
      VALL(J)=VALL(J)+A(JK)*A(IK)
    4 JK=JK+N-K                                                         BIGM0046
      IF(J.EQ.N)GO TO 6                                                 BIGM0047
      CALL LOOP1(J+2,NP1,VALL(J),A(JX),A(IX))
    5 JX=JX+N-J                                                         BIGM0049
    6 DELGAM=0.                                                         BIGM0050
      DO 7 J=IP1,N                                                      BIGM0051
      IJ=IX+J                                                           BIGM0052
    7 DELGAM=DELGAM+A(IJ)*VALL(J)
      DG02=.5*DELGAM                                                    BIGM0054
      DO 8 J=IP1,N                                                      BIGM0055
      IJ=IX+J                                                           BIGM0056
    8 T(J,1)=VALL(J)-DG02*A(IJ)
      DO 9 II=IP1,N                                                     BIGM0058
```

```
      III=IX+II
      CALL LOOP2(A(IIX),A(IX),T(NZ,1),T(II,1),A(III),II+1,NP1)
    9 IIX=IIX+N-II
   10 IX=IX+N-I
      M=N*(N+1)/2
      UPPERD(NM1)=A(M-1)
      T(NM1,2)=UPPERD(NM1)**2
      DIAG(NM1)=A(M-2)
      DIAG(N)=A(M)
      ENORM=AMAX1(ABS(DIAG)+ABS(UPPERD),ABS(DIAG(N))+ABS(UPPERD(NM1)))
      DO 11 I=2,NM1
      ENRTMP=ABS(DIAG(I))+ABS(UPPERD(I))+ABS(UPPERD(I-1))
   11 IF(ENRTMP.GT.ENORM)ENORM=ENRTMP
      DO 12 I=1,NEIG
      VALU(I)=ENORM
   12 VALL(I)=-ENORM
      DO 24 I=1,NEIG
   13 ROOT=.5*(VALU(I)+VALL(I))
      IF(ROOT.EQ.VALL(I).OR.ROOT.EQ.VALU(I))GO TO 24
      NAGREE=0
      PM2=0.
      PM1=1.
      DO 21 J=1,N
      IF(PM2.NE.0.)GO TO 15
   14 PM1=SIGN(1.,PM1)
      GO TO 17
   15 IF(PM1.NE.0.0) GO TO 17
   16 P=-SIGN(1.,PM2)
      PM2=0.
      IF(T(J-1,2)) 18,14,18
   17 P=DIAG(J)-ROOT-T(J-1,2)*PM2/PM1
      PM2=1.
   18 IF(P)21,19,20
   19 PM2=PM1
      IF(PM2)21,20,20
   20 NAGREE=NAGREE+1
   21 PM1=P
      DO 23 J=I,NEIG
      IF(J.LE.NAGREE)GO TO 22
      IF(VALU(J).LE.ROOT)GO TO 13
      VALU(J)=ROOT
      GO TO 23
   22 VALL(J)=ROOT
   23 CONTINUE
      GO TO 13
   24 CONTINUE
      IF(NVEC.EQ.0)GO TO 49
      EPSLON=ENORM*1.E-8
      COMPL1=COMPL(1)
      DO 48 I=1,NVEC
      DO 25 J=1,N
      V(J)=1.
      T(J,2)=DIAG(J)-VALU(I)
      IF(J.EQ.N)GO TO 26
      T(J,3)=UPPERD(J)
   25 T(J+1,1)=UPPERD(J)
   26 T(N,3)=0.
      DO 29 J=1,N
      IF(ABS(T(J,2)).LT.1.E-17)T(J,2)=EPSLON
      T(J,1)=T(J,2)
```

```
      T(J,2)=T(J,3)                                                     BIGM0119
      T(J,3)=0.                                                         BIGM0120
      IF(J.EQ.N)GO TO 30                                                BIGM0121
      INTER(J)=0                                                        BIGM0122
      JP1=J+1                                                           BIGM0123
      IF(ABS(T(JP1,1)).LE.ABS(T(J,1)))GO TO 28                          BIGM0124
      INTER(J)=1                                                        BIGM0125
      DO 27 K=1,3                                                       BIGM0126
      TEMP=T(J,K)                                                       BIGM0127
      T(J,K)=T(JP1,K)                                                   BIGM0128
   27 T(JP1,K)=TEMP                                                     BIGM0129
   28 TMULTP=T(JP1,1)/T(J,1)                                            BIGM0130
      VALL(J)=OR(INTER(J),AND(TMULTP,COMPL1))
      T(JP1,2)=T(JP1,2)-TMULTP*T(J,2)                                   BIGM0132
   29 T(JP1,3)=T(JP1,3)-TMULTP*T(J,3)                                   BIGM0133
   30 ITER=1                                                            BIGM0134
   31 DO 32 J=1,N                                                       BIGM0135
      L=N+1-J                                                           BIGM0136
   32 V(L)=(V(L)-T(L,2)*V(L+1)-T(L,3)*V(L+2))/T(L,1)                    BIGM0137
      VNORM=0.                                                          BIGM0138
      DO 33 L=1,N                                                       BIGM0139
   33 VNORM=VNORM+V(L)**2                                               BIGM0140
      VNORM=SQRT(VNORM)                                                 BIGM0141
      DO 34 J=1,N                                                       BIGM0142
   34 V(J)=V(J)/VNORM                                                   BIGM0143
      IF(ITER.EQ.2)GO TO 36                                             BIGM0144
      ITER=2                                                            BIGM0145
      DO 35 L=2,N                                                       BIGM0146
      LM1=L-1                                                           BIGM0147
      TRY=VALL(LM1)
      IF(AND(TRY,1).EQ.0) GO TO 35
      VTEMP=V(LM1)                                                      BIGM0149
      V(LM1)=V(L)                                                       BIGM0150
      V(L)=VTEMP                                                        BIGM0151
   35 V(L)=V(L)-VALL(LM1)*V(LM1)
      GO TO 31                                                          BIGM0153
   36 IF(VNORM.EQ.0.)V(1)=1.                                            BIGM0154
      IIX=(N*N-N-6)/2                                                   BIGM0155
      DO 37 KK=1,NM2                                                    BIGM0156
      IIP1=N-KK                                                         BIGM0157
      UTV=0.                                                            BIGM0158
      CALL LOOP3(UTV,A(IIX),V(NZ),IIP1+1,NP1)                           BIGM0159
      CALL LOOP4(A(IIX),V(NZ),NP1,IIP1+1,UTV)                           BIGM0160
   37 IIX=IIX+IIP1-N-2                                                  BIGM0161
      WRITE(MTAPE) (V(ICH),ICH=1,N)
   48 CONTINUE
   49 RETURN
      END
```

```
$       FORTRAN DECK
CLOOP1
        SUBROUTINE LOOP1(JP2,NP1,SGAMPJ,AJX,AIX)                        BIGM0167
        DIMENSION AJX(1), AIX(1)                                        BIGM0168
        DO 1 L=JP2,NP1                                                  BIGM0169
      1 SGAMPJ=SGAMPJ+AJX(L)*AIX(L)                                     BIGM0170
        RETURN                                                          BIGM0171
        END
$       FORTRAN DECK
CLOOP2
        SUBROUTINE LOOP2(AIIX,AIX,S,SI,AIII,IP1,NP1)                    BIGM0174
        DIMENSION AIIX(1),AIX(1),S(1)                                   BIGM0175
        DO 2 JJ=IP1,NP1                                                 BIGM0176
      2 AIIX(JJ)=AIIX(JJ)-AIII*S(JJ)-SI*AIX(JJ)                         BIGM0177
        RETURN                                                          BIGM0178
        END
$       FORTRAN DECK
CLOOP3
        SUBROUTINE LOOP3(UTV,AIIX,V,IIP2,NP1)                           BIGM0181
        DIMENSION AIIX(1), V(1)                                         BIGM0182
        DO 3 J=IIP2,NP1                                                 BIGM0183
      3 UTV=UTV+AIIX(J)*V(J)                                            BIGM0184
        RETURN                                                          BIGM0185
        END
$       FORTRAN DECK
CLOOP4
        SUBROUTINE LOOP4(AIIX,V,NP1,IIP2,UTV)                           BIGM0188
        DIMENSION AIIX(1),V(1)                                          BIGM0189
        DO 4 K=IIP2,NP1                                                 BIGM0190
      4 V(K)=V(K)-AIIX(K)*UTV                                           BIGM0191
        RETURN                                                          BIGM0192
        END
```

```
$       FORTRAN DECK
CGENA          GENERATES THE GENERALIZED AERODYNAMIC FORCES
C                    FOR THE SYSTEM FROM AIC MATRICES
C       KKDISC CONTAINS XF - TRANSFORMED SYSTEM MODE SHAPES FOR COMPONENTS
C       MADISC CONTAINS THE AIC MATRIX FOR EACH COMPONENT
C       MTDISC - SCRATCH TAPE
C       GAC - GENERALIZED AERODYNAMIC FORCES FOR EACH COMPONENT.
C       GA - GENERALIZED AERODYNAMIC FORCES FOR THE SYSTEM.
C
        SUBROUTINE GENA (KKDISC,MADISC,MTDISC,VEL,NCOMP,MODE,NCOUP,NAT,NV,
       1NREDU,AP,AIC,XF,XFT,B)
C
        DIMENSION NREDU(1),XFPT(9,4),XFP(4,9),XF(97,9),GAC(9,18),GA(9,18),
       1B(1),VEL(1),XFA(9,18),AIC(46,80),XFT(9,97),AP(4,8),AC(40,80)
C
      1 FORMAT(/// 1X,40HGENERALIZED AERODYNAMIC FORCES FOR 1/K = 1P1E11.4
       1/)
      2 FORMAT(1H0 /(2E16.8,1X,2E16.8,1X,2E16.8,1X,2E16.8))
        M2=2*MODE
        DATA Q1/4HGENA/
        REWIND MADISC
        DO 200 K=1,NV
        REWIND KKDISC
        READ(MADISC)VEL(K)
        WRITE(6,1)VEL(K)
        DO 175 JA=1,MODE
        DO 175 KA=1,M2
    175 GA(JA,KA)=0.0
        DO 198 I=1,NCOMP
        IF(I.EQ.NCOUP+1)GO TO 198
        N=NREDU(I)
        N2=2*N
        DO 176 JA=1,MODE
        DO 176 KA=1,M2
    176 GAC(JA,KA)=0.0
        READ(KKDISC)((XF(M,L),L=1,MODE),M=1,N)
        IF(NAT.EQ.2) GO TO 191
C       PARTITIONED AIC MATRICES (FROM STRIP OR PISTON THEORIES)
        READ(MADISC) NPART
        NSE=0
        NSS=1
        DO 190 J=1,NPART
        READ(MADISC) NS,NS2
        READ(MADISC)((AP(JJ,KK),KK=1,NS ),JJ=1,NS)
        NSE=NSE+NS
        DO 185  NF=1,MODE
        LF=0
        DO 185  MF=NSS,NSE
        LF=LF+1
    185 XFP(LF,NF)=XF(MF,NF)
        NSS=NSE+1
        DO 186 JA=1,NS
        DO 186 KA=1,MODE
    186 XFPT(KA,JA)=XFP(JA,KA)
        CALL CMULT(XFPT,AP,XFP,XFA,4,4,8,MODE,NS,NS2)
        DO 180 JA=1,MODE
        DO 180 KA=1,M2
    180 GAC(JA,KA)=GAC(JA,KA)+XFA(JA,KA)
    190 CONTINUE
        GO TO 196
```

```
  191 IF(NCOUP.EQ.1) GO TO 250
C     FULL AIC MATRICES (FROM KERNEL FUNCTION OR MACH BOX THEORIES)
      DO 193 JA=1,N
  193 READ(MADISC)(AIC(JA,KA),KA=1,N2)
      DO 194 JA=1,N
      DO 194 KA=1,MODE
  194 XFT(KA,JA)=XF(JA,KA)
      CALL CMULT (XFT,AIC,XF,GAC,97,40,80,MODE,N,N2)
      GO TO 196
C     OPTION FOR AERODYNAMIC COUPLING BETWEEN TWO COMPONENTS
  250 II=I+1
      N=NREDU(I)+NREDU(II)
      N2=2*N
      DO 251 JA=1,N
  251 READ(MADISC) (AIC(JA,KA),KA=1,N2)
      NC=NREDU(I)
      NC2=2*NC
      NCC=NREDU(II)
      NCC2=2*NCC
      NCC3=NC2+1
      NCC4=NC+1
      REWIND MTDISC
      DO 275 JK=1,4
      GO TO(352,355,357,361),JK
  352 DO 252 JA=1,NC
      DO 252 KA=1,NC2
  252 AC(JA,KA)=AIC(JA,KA)
      DO 253 JA=1,NC
      DO 253 KA=1,MODE
  253 XFT(KA,JA)=XF(JA,KA)
      CALL CMULT(XFT,AC,XF,XFA,97,40,80,MODE,NC,NC2)
      WRITE(MTDISC)((XF(JA,KA),KA=1,MODE),JA=1,NC)
      GO TO 362
  355 DO 255 JA=1,NC
      DO 255 KA=NCC3,N2
      LA=KA-NCC3+1
  255 AC(JA,LA)=AIC(JA,KA)
      READ(KKDISC)((XF(M,L),L=1,MODE),M=1,NCC)
      CALL CMULT (XFT,AC,XF,XFA,97,40,80,MODE,NC,NCC2)
      WRITE(MTDISC)((XF(JA,KA),KA=1,MODE),JA=1,NCC)
      GO TO 362
  357 REWIND MTDISC
      DO 258 JA=1,NCC
      DO 258 KA=1,MODE
  258 XFT(KA,JA)=XF(JA,KA)
      READ(MTDISC)((XF(JA,KA),KA=1,MODE),JA=1,NC)
      DO 259 JA=NCC4,N
      LA=JA-NCC4+1
      DO 259 KA=1,NC2
  259 AC(LA,KA)=AIC(JA,KA)
      CALL CMULT(XFT,AC,XF,  XFA,97,40,80,MODE,NCC,NC2)
      GO TO 362
  361 READ(MTDISC)((XF(JA,KA),KA=1,MODE),JA=1,NCC)
      DO 261 JA=NCC4,N
      LA=JA-NCC4+1
      DO 261 KA=NCC3,N2
      LB=KA-NCC3+1
  261 AC(LA,LB)=AIC(JA,KA)
      CALL CMULT (XFT,AC,XF,XFA,97,40,80,MODE,NCC,NCC2)
  362 DO 262 JA=1,MODF
```

```
      DO 262 KA=1,M2
  262 GAC(JA,KA)=GAC(JA,KA)+XFA(JA,KA)
  275 CONTINUE
  196 DO 197 JA=1,MODE
      DO 197 KA=1,M2
  197 GA(JA,KA)=GA(JA,KA)+GAC(JA,KA)
  198 CONTINUE
      IC=0
      DO 199 JA=1,MODE
      WRITE(6,2)(GA(JA,KA),KA=1,M2)
      DO 192 KA=1,M2
  192 B(KA)=GA(JA,KA)
      CALL PUNC (B,1,M2,Q1,IC)
  199 CONTINUE
  200 CONTINUE
      RETURN
      END
```

4.0 MOFA - MODAL FLUTTER ANALYSIS PROGRAM

4.1 Theoretical Development

The flutter problem can be solved with either a collocation or normal-mode formulation. The collocation approach is attractive if an accurate stiffness and aerodynamic influence coefficient matrix can be generated for the system. The normal-mode method merits consideration when mode shapes and natural frequencies for the structure are known.

The equations of motion, in matrix notation, for a lumped parameter, linear system acted upon by aerodynamic forces is

$$[m]\{\ddot{h}\} + [k]\{h\} = \{F_a\} \tag{4.1.1}$$

where $[m]$ = symmetric mass matrix

$[k]$ = symmetric stiffness matrix

$\{h\}$ = control point deflection

$\{F_a\}$ = aerodynamic force

The aerodynamic force matrix can be defined as a complex matrix of oscillatory aerodynamic influence coefficients such that

$$\{F_a\} = \rho\, \omega^2\, b_r^2\, s\, [C_h]\{h\} \tag{4.1.2}$$

where ρ = air density

ω = oscillatory frequency, rad/sec

b_r = reference semi-chord

s = reference semi-span

$[C_h]$ = aerodynamic influence coefficient matrix

Assuming harmonic motion, $\{h\} = \{\bar{h}\}\, e^{i\omega t}$, the equations of motion become

$$-\omega^2 [m]\{\bar{h}\} + [k]\{\bar{h}\} = \rho\, \omega^2 b_r^2\, s\, [C_h]\{\bar{h}\} \tag{4.1.3}$$

The collocation method solves the flutter problem with the equations of motion cast in this form.

For the modal approach, the equations of motion can be uncoupled with the linear transformation

$$\{\bar{h}\} = \left[\Phi\right] \{\psi\} \qquad (4.1.4)$$

where $\left[\Phi\right]$ is the modal matrix and $\{\psi\}$ the normal coordinates. Substituting Equation (4.1.4) into Equation (4.1.3) and premultiplying both sides of the resulting equation by $\left[\Phi\right]^T$, we obtain

$$-\omega^2 \left[\Phi\right]^T \left[m\right]\left[\Phi\right] \{\psi\} + \left[\Phi\right]^T \left[k\right] \left[\Phi\right] \{\psi\} = \rho\ \omega^2\ b_r^2\ s\left[\Phi\right]^T \left[C_h\right]\left[\Phi\right] \{\psi\} \quad (4.1.5)$$

By virtue of the orthogonality of the modal matrix with respect to the mass and stiffness matrix, $\left[\Phi\right]^T \left[m\right] \left[\Phi\right]$ and $\left[\Phi\right]^T \left[k\right]\left[\Phi\right]$ become diagonal matrices. With the following definitions

$$\left[\bar{M}\right] = \left[\Phi\right]^T \left[m\right]\left[\Phi\right] = \text{generalized mass matrix}$$

$$\left[\bar{K}\right] = \left[\Phi\right]^T \left[k\right] \left[\Phi\right] = \text{generalized stiffness matrix}$$

$$\left[Q\right] = \rho\ \omega^2\ b_r^2\ s \left[\Phi\right]^T \left[C_h\right]\left[\Phi\right] = \text{generalized force matrix}$$

$$\left[\bar{Q}\right] = \rho\ b_r^2 s \left[\Phi\right]^T \left[C_h\right]\left[\Phi\right]$$

Equation (4.1.5) can be written as

$$\left(\left[\bar{M}\right] + \left[\bar{Q}\right] - \frac{1}{\omega^2} \left[\bar{K}\right]\right) \{\psi\} = 0 \qquad (4.1.6)$$

Equation (4.1.6) can be written in a slightly different form by noting the generalized stiffness matrix can be expressed in terms of the generalized masses and natural frequencies of the structure. If $\left[\omega_n^2\right]$ is a diagonal matrix of natural frequencies squared, rad^2/sec^2, then

$$\left[\bar{K}\right] = \left[\omega_n^2\right] \left[\bar{M}\right] \qquad (4.1.7)$$

and equation (3.2.6) becomes

$$\left(\left[\bar{M}\right] + \left[\bar{Q}\right] - \frac{1}{\omega^2} \left[\omega_n^2\right] \left[\bar{M}\right]\right)\{\psi\} = 0 \qquad (4.1.8)$$

Adding artificial structural damping, g, to the system of equations, we arrive at the classical normal-mode formulation of the fluter problem:

$$\left(\left[\bar{M}\right] + \left[\bar{Q}\right] - \frac{1+ig}{\omega^2} \left[\omega_n^2\right] \left[\bar{M}\right]\right)\{\psi\} = 0 \qquad (4.1.9)$$

or

$$\left(\left[\bar{M}\right] + \left[\bar{Q}\right] - \lambda\left[\omega_n^2\right]\left[\bar{M}\right]\right) \{\psi\} = 0 \qquad (4.1.10)$$

Equation (4.1.10) can be solved for the complex eigenvalue λ and complex eigenvector $\{\psi\}$. From the complex eigenvalue $(\lambda = \lambda_{real} \pm i\,\lambda_{imag})$ the flutter frequency is

$$\omega_f = \frac{1}{\sqrt{\lambda_{real}}} \tag{4.1.11}$$

and the artificial structural damping is

$$g = \frac{\lambda_{imag}}{\lambda_{real}} \tag{4.1.12}$$

4.2 Program Description

The notation used in the computer program HOFA for Equation 4.1.9 or 4.1.10 is:

$$\left[M_{ij} + F\, B_{ij}) - \lambda_i \; P_{ii}\right] \left\{Q_i\right\} = 0 \qquad 4.2.1$$

or

$$\left[A_{ij} - \lambda_i\, P_{ii}\right] \left\{Q_i\right\} = 0 \qquad 4.2.2$$

where:

n = number of generalized coordinates (Max 20)

i = row index, $1 \leq i \leq n$

j = column index, $1 \leq j \leq n$

M_{ij} = generalized mass matrix

B_{ij} = generalized aerodynamic force matrix

A_{ij} = sum of mass and aero-matrices

P_{ii} = diagonal stiffness parameter matrix = $M_{ii} \left(\frac{W_i}{W_r}\right)^2$

F = factor for aero matrix

W_i = modal frequency

W_r = reference frequency, usually W_1 or W_n

The sum of mass and aero-matrices, A_{ij} <u>or</u> the generalized mass M_{ij}, and the generalized force B_{ij}, must be input. The stiffness parameter P_{ii} may be input or computed by inputing ω_i, the modal frequencies and the diagonal generalized mass matrix M_{ii} (if the generalized mass M_{ij} is not input). In addition to the complex Eigenvalues, the program computes the following:

λ = Eigenvalues = $\lambda_{real} + i\ \lambda_{imag}$

ω_f = flutter frequencies = $W_r / \sqrt{(\lambda_{real})}$

g = artificial structural damping = $\lambda_{imag} / \lambda_{real}$

V = flutter velocity = ω_fx ref. chord/Strouhal No.

A vibration analysis may be performed by entering zero aerodynamic forces, and zeros for all parameters associated with determining the aerodynamic forces.

4.2.1 PROCESSING INFORMATION

A. Operation-Standard FORTRAN IV Processor System operable on the GE 635 computer.

B. Core Storage - The program MOFA requires a minimum of 20,000 memory units for execution.

C. Tape Units - Standard input and output tapes.

4.3 Input Instructions

The following instructions describe the input data, the physical units and the input format to be used.

Title Cards - 2 title cards are required at beginning of each computer run.

<u>Item 1</u> Control Card Format (15I3)

Column	1-3	4-6	7-9	10-12	13-15	16-18	19-21
Name	W_r	b_r	NDEGA	M_{ij}	M_{ii}	W_i	P_{ii}
Field	(1)	(2)	(3)	(4)	(5)	(6)	(7)

Column	22-24	25-27	28-30	31-33	34-36	37-39	40-42	42-45
Name	NOK	F	B_{ij}	A_{ij}				
Field	(8)	(9)	(10)	(11)				

(1) Input Reference Frequency
W_r = 1 ~ Input
0 ~ No Input

(2) Input Reference Semi-Chord
b_r = 1 ~ Input
0 ~ No Input

(3) NDEGA Number of modes to be used in the analysis, generalized coordinates (Max 20)

(4) Input Coupled Generalized Mass
M_{ij} = 2 ~ Input, all real elements
1 ~ Input, both real and imaginary elements
0 ~ No Input

(5) Input Uncoupled Generalized Mass
M_{ii} = 2 ~ Input, all real elements
1 ~ Input, both real and imaginary elements
0 ~ No Input
M_{ii} must equal zero if M_{ij} = 1 or 2

(6) Input Modal Frequencies
W_i = 1 ~ Input
0 ~ No Input

(7) Input Stiffness Parameter
P_{ii} = 2 ~ to be computed. In Item 1, (1), (4) or (5), and (6) must be input or available in core from previous problem
= 1 ~ Input
= 0 ~ No Input

(8) Input Reduced Velocity
NOK = 1 ~ Input
0 ~ No Input (Vibration Analysis)

(9) Input Factor for Aero Matrix
F = 1 ~ Input
0 ~ No Input

(10) Input Aero Matrix
2 ~ Input, by rows loaded in a continuous manner (to accommodate the output from programs in Vol. II)
B_{ij} = 1 ~ Input, by rows - each row starts on a new card
0 ~ No Input

(11) Input Sum of Mass & Aerodynamic Matricies
2 ~ to be computed. In Item 1, (4) or (5) (9), and (10) must be input of available in care from previous problem
A_{ij} = 1 ~ Input
0 ~ No Input

Item 2. Reference Frequency, W_r, Format 1E12.8

Column	1-12	
Name	ω_r	
Field	(1)	

ω_r must equal the value used when computing the unsteady generalized aerodynamic forces (RAD/SEC).

Item 3. Reference semi-chord, b_r, Format 1E12.8

Column	1-12	
Name	b_r	
Field	(1)	

b_r must be same units used in calculating aerodynamics

Item 4. Coupled Generalized Mass Format 6E12.8

If in Item 1, M_{ij} = 1, use the following format

Column	1-12	13-24	25-36	37-48	49-60	61-72	
Name	$ReM_{1,1}$	$IM_{1,1}$	$ReM_{1,2}$	$IM_{1,2}$		$IM_{1,NDEGA}$	
Field	(1)	(2)	(3)	(4)	(5)	(6)	

Continued on next card if NDEGA > 6
Start each row on a new card

$ReM_{i,j}$ = The Real Part of $M_{i,j}$

$IM_{i,j}$ = The Imaginary Part of $M_{i,j}$ (pseudo-IM = 0.0 in all cases)

If in Item 1, M_{ij} = 2, use the following format

Column	1-12	13-24	25-36		61-72	
Name	$M_{1,1}$	$M_{1,2}$	$M_{1,3}$		$M_{1,NDEGA}$	

All mass elements are real numbers. Continue on next card if NDEGA>6. Start each row on a new card.

Item 5. Uncoupled Generalized Mass Format 6E12.8
If in Item 1 M_{ii} = 1, use the following format

Column	1-12	13-24	25-36	37-48	49-60	61-72	
Name	$ReM_{1,1}$	$IM_{1,1}$	$ReM_{2,2}$	$IM_{2,2}$	'''	$IM_{i,i}$	
Field	(1)	(2)	(3)	(4)	(5)	(6)	

i = 1,2, ... NDEGA

Continued on next card if necessary

$ReM_{i,i}$ = The Real Park of $M_{i,i}$

$IM_{i,i}$ = The Imaginary Part of $M_{i,i}$ (pseudo-IM = 0.0 in all cases)

If in Item 1, $M_{i,i}$ = 2, use the following format

Column	1-12	13-24	25-36		61-72	
Name	$M_{1,1}$	$M_{2,2}$	$M_{3,3}$	'''	$M_{i,i}$	

Item 6. Modal Frequencies Format 6E12.8 (RAD/SEC)

Column	1-12	13-24	25-36	37-48	49-60	61-72	
Name	W_1	W_2	'''	'''	'''	W_{NDEGA}	
Field	(1)	(2)	(3)	(4)	(5)	(6)	

Continued on next card if necessary

Item 7.

Column	1-12	13-24	25-36	37-48	49-60	61-72	
Name	$ReP_{1,1}$	$IP_{1.1}$	$ReP_{2,2}$	$IP_{2,2}$	'''	IP_{ii}	
Field	(1)	(2)	(3)	(4)	(5)	(6)	

$ReP_{i,i}$ is the Real Part of $P_{i,i}$

$IP_{i,i}$ is the Imaginary Part of $P_{i,i}$ (pseudo-IP = 0.0 in all cases)

where i = 1,2, ... NDEGA

Continue on next card if necessary until $IP_{1.NDEGA}$ is entered

Item 8. Reduced Velocity, Format 1E12.8

Column	1-12	
Name	1/K	
Field	(1)	

1/K = Reciprocal of reduced frequency used in calculating the unsteady aerodynamic generalized forces

Item 9. Aero Matrix Factor Format 1E12.8
F = Nondimensionalizing factor, $F \neq 0$

Column	1-12	
Name	F	
Field	(1)	

F depends on method used to calculate aerodynamics.
If the programs presented in Volume II are used, $F = 1/n\omega^{-2}$

where n = 1 when the generalized mass matrix is calculated for the entire vehicle (nonsymmetrical mode shapes)

n = 2 when the generalized mass matrix is calculated for one half the vehicle (symmetrical or anti-symmetrical mode shapes)

n = 4 when the generalized mass matrix is calculated for 1/4 the vehicle (anti-symmetrical mode shapes of crusiform planforms).

If the generalized aerodynamic forces are obtained from COMSYN, F = the non-dimensionalizing factor for the aerodynamics, equals

$b_r^2 s$, where s is the semi-span.

NOTE: The air density, ρ, should be included in F for altitude variation, if not already considered in the aerodynamics. (Length units should be consistent with b_r.)

Item 10. Generalized Aerodynamic Force Matrix Format 6E12.8
Input by Rows

Column	1-12	13-24	25-36	37-48	49-60	61-72	
Name	$ReB_{1,1}$	$IB_{1,1}$	$ReB_{1,2}$	$IB_{1,2}$	'''	'''	
Field	(1)	(2)	(3)	(4)	(5)	(6)	

$ReB_{i,j}$ = The Real Part of $B_{i,j}$
$IB_{i,j}$ = The Imaginary Part of $B_{i,j}$

Continue on next card if necessary until $IB_{1,NDEGA}$ is entered
Start each row on a new card if in item 1, $B_{ij} = 1$

Item 11. Sum of Mass and Aerodynamic Force Matrices Format 6E12.8 Input By Rows.

Column	1-12	13-24	25-36	37-48	49-60	61-72	
Name	$ReA_{1,1}$	$IA_{1,1}$	$ReA_{1,2}$	$IA_{1,2}$	'''	'''	
Field	(1)	(2)	(3)	(4)	(5)	(6)	

$ReA_{i,j}$ = The Real Part of $A_{i,j}$

$IA_{i,j}$ = The Imaginary Part of $A_{i,j}$

Continue on next card if necessary until $IA_{1,NDEGA}$ is entered, Start each row on a new card.

The basic use of the control card is to minimize input when stacking cases. Once program has been initialized, you may use information from previous case by entering a zero or leave blank on the control card the quantity you want to use from the previous case. Any new input, indicate input as such on the control card and input that quantity. Any number of cases may be stacked. A control card must be present for each and every case.

4.4 Program Output

All input, computed data and results are printed.

The complex eigenvalue problem is solved by the subroutine ALLMAT.

The flutter results are presented in tabular form, the units being consistent with the reference semi-chord used in the input. The output frequency is in radians per second (as required for input), and the velocity is in inches or feet per second depending upon the units of b_r.

4.5 Sample Problem

A sample problem is presented; the input data is presented on the first page of the computer output. The control card input is shown below:

1-3	4-6	7-9	10-12	13-15	16-18	19-21	22-24	25-27	28-30	31-33	34-36	37-39	40-42	43-45
W_r	b_r	NDEGA	M_{ij}	M_{ii}	W_i	P_{ii}	NOK	F	B_{ij}	A_{ij}				
1	1	7	2	0	1	2	1	1	1	2				

The punched output of the sample problem in Section 3.6 is used as input in MOFA.

MODAL FLUTTER PROGRAM MODIFICATION CHECK CASE FOR COFA II

TYPICAL MISSILE DIVIDED INTO THREE COMPONENTS

MODAL FLUTTER ANALYSIS

INPUT DATA

REFERENCE FREQUENCY = 0.10000000E 01 RPS

REFERENCE SEMI-CHORD = 0.15000000E 02

NUMBER OF MODES = 7

GENERALIZED MASS MATRIX

1	1	0.63840300E 00	0.
1	2	-0.15450300E-05	0.
1	3	-0.14164900E-08	0.
1	4	0.31795600E-08	0.
1	5	0.10741900E-07	0.
1	6	-0.95983800E-08	0.
1	7	-0.59987700E-08	0.
2	1	-0.15618300E-05	0.
2	2	0.18509100E 01	0.
2	3	-0.82449800E-08	0.
2	4	-0.83497000E-06	0.
2	5	0.10686500E-07	0.
2	6	-0.53449000E-07	0.
2	7	-0.29367900E-06	0.
3	1	-0.93132300E-08	0.
3	2	-0.24502500E-07	0.
3	3	0.10634300E 01	0.
3	4	0.15125800E-05	0.
3	5	0.42786200E-08	0.
3	6	0.25792200E-07	0.
3	7	0.22967500E-07	0.
4	1	0.26977000E-07	0.
4	2	-0.83902600E-06	0.
4	3	0.14918600E-05	0.
4	4	0.11071800E 01	0.
4	5	-0.65659700E-08	0.
4	6	0.17756600E-07	0.
4	7	0.19203900E-06	0.
5	1	0.10884800E-07	0.
5	2	0.11717000E-07	0.
5	3	0.46463200E-08	0.
5	4	-0.71920800E-08	0.
5	5	0.10004700E 01	0.
5	6	0.10609200E-07	0.
5	7	-0.54582200E-08	0.
6	1	-0.77707200E-08	0.
6	2	-0.57098900E-07	0.
6	3	0.23062800E-07	0.

```
 6  4  0.19997800E-07  0.
 6  5  0.19693700E-07  0.
 6  6  0.10000500E 01  0.
 6  7  0.31090400E-07  0.
 7  1  0.16763800E-07  0.
 7  2 -0.25415900E-06  0.
 7  3  0.17229700E-07  0.
 7  4  0.14952000E-06  0.
 7  5 -0.42319700E-08  0.
 7  6  0.25616500E-07  0.
 7  7  0.41671800E 09  0.
```

MODAL FREQUENCIES (RPS)

```
 0.          0.          0.51709500E 03  0.54821300E 03  0.70959600E 03  0.79491000E 03  0.11936800E 04
```

DIAGONAL STIFFNESS PARAMETER MATRIX

```
 0.               0.          0.               0.          0.28434761E 06  0.
 0.33274910E 06   0.          0.50376314E 06   0.          0.63191350E 06  0.
 0.59376977E 06   0.
```

```
1/K =  0.10000000E 02
```

```
FACTOR FOR AERODYNAMIC FORCES =  0.90247500E-03
```

GENERALIZED AERODYNAMIC FORCES

181

```
 1  1 -0.47111800E 02 -0.21968500E 02
 1  2  0.14384300E 03  0.89207700E 01
 1  3 -0.83685200E 02 -0.19839100E 03
 1  4 -0.46236500E 03  0.51138700E 02
 1  5 -0.13671300E 05  0.21563000E 03
 1  6  0.68124100E 02 -0.11202200E 02
 1  7 -0.11102200E 03 -0.23611000E 02
 2  1 -0.11039300E 02  0.26916300E 01
 2  2  0.32244600E 02 -0.21655700E 02
 2  3  0.48166800E 02  0.13890800E 03
 2  4  0.25158400E 03 -0.11399800E 02
 2  5  0.91751200E 04 -0.11502500E 03
 2  6  0.65013700E 02 -0.12168100E 03
 2  7 -0.24987600E 03  0.10053200E 02
 3  1 -0.17134300E 03 -0.15608900E 03
 3  2  0.54970000E 03  0.25826000E 03
 3  3 -0.12344300E 04 -0.36633800E 04
 3  4 -0.52934000E 04  0.64779100E 03
 3  5 -0.18678100E 06  0.21700700E 04
 3  6 -0.23556700E 03  0.21579500E 04
 3  7  0.29505100E 04 -0.77310600E 03
 4  1  0.73529300E 02  0.49605600E 02
 4  2 -0.22921400E 03 -0.32045500E 02
 4  3  0.25269200E 03  0.74117400E 03
 4  4  0.10163500E 04 -0.16132900E 03
 4  5  0.34749400E 05 -0.51331900E 03
 4  6 -0.53172600E 02 -0.31632900E 03
 4  7 -0.25102300E 03  0.17973300E 03
 5  1 -0.19332100E 03 -0.15276900E 03
 5  2  0.61148300E 03  0.22327000E 03
 5  3 -0.84949700E 03 -0.25265000E 04
```

```
5 4  [illegible]46704900E 04  0.29479500E 03
5 5 -0.15415000E 06 -0.30802500E 04
5 6 -0.47654400E 02  0.10904400E 04
5 7  0.15157900E 04 -0.29934700E 03
6 1 -0.29459200E 03 -0.87568300E 02
6 2  0.88125600E 03 -0.92717700E 02
6 3 -0.58574400E 02  0.47866400E 02
6 4 -0.44131900E 03  0.16832600E 03
6 5  0.34630500E 03 -0.28094700E 01
6 6  0.72953300E 03 -0.14281000E 04
6 7 -0.31544200E 04 -0.66987800E 01
7 1  0.14503500E 01 -0.65069800E 00
7 2 -0.43187100E 01  0.36982200E 01
7 3 -0.54134700E 00 -0.11367400E 02
7 4  0.11693200E 02 -0.35940800E 01
7 5  0.58903700E 02 -0.34508200E 02
7 6 -0.10078300E 02  0.67400600E 02
7 7  0.10553000E 03 -0.26879900E 02
```

COMPUTED SUM OF MASS AND AERODYNAMIC MATRICES

```
 5.9588578E-01 -1.9826022E-02   1.2981317E-01  8.0507720E-03   -7.5523803E-02 -1.7904292E-01   -4.1727326E-01  4.6151399E-02
-1.2338007E 01  1.9460068E-01   6.1480287E-02 -1.0109706E-02   -1.0019459E-01 -2.1308337E-02

-9.9642541E-03  2.4291288E-03   1.8800099E 00 -1.9543728E-02    4.3469325E-02  1.2536100E-01    2.2704744E-01 -1.0288034E-02
 8.2803165E 00 -1.0380719E-01   5.9395165E-02 -1.0981406E-01   -2.2550714E-01  1.7195037E-02

-1.5463278E-01 -1.4086642E-01   4.9609048E-01  2.3307319E-01   -5.0612211E-02 -3.3061089E 00   -4.7771597E 00  5.8461529E-01
-1.6856518E 02  1.9584339E 00  -2.0808093E-01  1.9474959E 00    2.6627616E 00 -6.9779006E-01

 6.6358381E-02  3.6645539E-02  -2.0595827E-01 -2.8920263E-02    2.2804970E-01  6.6889101E-01    2.0244105E 00 -1.4559539E-01
 3.1360826E 01 -4.6325757E-01  -4.8257667E-02 -2.8547901E-01   -2.3006144E-01  1.6220454E-01

-1.7175993E-01 -1.3787020E-01   5.4462834E-01  2.0149560E-01   -7.6664981E-01 -2.2801031E 00   -4.2150005E 00  2.6604512E-01
-1.3812327E 02 -2.7879709E 00  -4.3006894E-02  9.8409484E-01    1.3589378E 00 -2.7015318E-01

-2.6586192E-01 -7.9028202E-02   7.9711640E-01 -8.3675407E-02   -5.2861908E-02  4.3198229E-02   -3.9827935E-01  1.5191001E-01
 3.1253112E-01 -2.5354764E-03   1.6584353E 00 -1.2888246E 00   -2.8467852E 00 -6.0454815E-03

 1.3089214E-03 -5.8723868E-04  -3.9158314E-03  3.3375511E-03   -4.8853490E-04 -1.0258794E-02    1.0552970E-02 -3.2443796E-03
 5.3159113E-02 -3.1142788E-02  -9.0953882E-03  6.0827357E-02    5.1195619E-01 -2.4258438E-02
```

REDUCED SUM OF MASS AND AERODYNAMIC MATRICES

```
-1.9597798E-02 -3.4097399E 00  -4.9557877E 00  4.6445660E-01   -1.7396934E 02 -2.1649887E 00   -2.2291091E-01  1.9819342E 00
 2.7085352E 00 -6.9956306E-01

 2.2683313E-01  7.0853409E-01   2.0987933E 00 -1.1964862E-01    3.3662077E 01  4.8642254E-01   -4.6221413E-02 -2.9991025E-01
-2.4734406E-01  1.6816027E-01

-7.4174740E-01 -2.3913354E 00  -4.4115158E 00  1.5280500E-01   -1.4410780E 02 -6.7196652E 00   -5.5958394E-02  1.0220785E 00
 1.4078026E 00 -2.7639621E-01

-2.2055489E-02 -1.1130895E-01  -6.8885560E-01  1.2053894E-01   -8.8207973E 00 -1.4851908E 00    1.6627027E 00 -1.2325751E 00
-2.7552458E 00 -4.4281395E-02
```

```
1.5767918E-04 -9.7271201E-03   1.1922584E-02 -4.1425370E-03   9.8575315E-02 -5.7503633E-02   -9.2854379E-03  6.0548313E-02
5.1171678E-01 -2.3853809E-02

DYNAMICAL MATRIX

-6.8921969E-08 -1.1991449E-05  -1.7428624E-05  1.6334113E-06  -6.1181922E-04 -7.6138803E-06  -7.8393805E-07  6.9761101E-06
9.5254368E-06 -2.4602389E-06

6.0169420E-07  2.1293343E-06   6.3074350E-06 -3.5957608E-07   1.0116354E-04  1.4618298E-06   -1.3890770E-07 -9.0131048E-07
-7.4333502E-07  5.0536659E-07

-1.4724130E-06 -4.7469439E-06  -8.7571229E-06  3.0332708E-07  -2.8606261E-04 -1.3338938E-05   -1.1108076E-07  2.0288870E-06
2.7945725E-06 -5.4866303E-07

-1.2985241E-07 -1.6046334E-07  -1.0901106E-06  1.9075227E-07  -1.3958868E-05 -2.3503072E-06    2.6312189E-06 -1.9505440E-06
-4.4076378E-06 -7.0074948E-08

2.6555608E-10 -1.6381972E-08   2.0079472E-08 -6.9766720E-09   1.6601605E-07 -9.6844999E-08   -1.5638111E-08  1.0197271E-07
8.6181009E-07 -4.0173498E-08
```

```
NUMBER OF EIGENVALUES AND EIGENVECTORS CALCULATED =  5

EIGENVECTOR CORRESPONDING TO EIGENVALUE =  7.2213784E-07  4.5314527E-08

  3.9214671E-02 -1.6127506E-01   -6.8672888E-02  6.5037549E-02    1.9977601E-03 -3.4525222E-04    9.9999999E-01  0.
  4.2620574E-01 -4.7529974E-01

EIGENVECTOR CORRESPONDING TO EIGENVALUE =  2.8087757E-04 -2.4446720E-06

  9.9999998E-01  0.              -2.1714868E-01  3.1638493E-02    2.7546624E-05 -1.8360252E-02   -8.4956662E-02  1.1154428E-01
  3.7636015E-03 -9.5741354E-03

EIGENVECTOR CORRESPONDING TO EIGENVALUE =  3.3482892E-04 -1.7909358E-07

  8.8454765E-02 -4.2602004E-02    9.9999999E-01  0.              -3.3833316E-02  1.2686323E-03    1.0258012E-02  3.4126832E-01
 -8.2322250E-03 -4.1762769E-03

EIGENVECTOR CORRESPONDING TO EIGENVALUE =  3.1833040E-06 -1.7404286E-06

  9.9999999E-01  0.              -3.0652155E-01  4.0514051E-02    4.1752065E-03 -1.7193443E-02    1.2755683E-01 -1.8779222E-01
  1.0878949E-03 -6.2904709E-05

EIGENVECTOR CORRESPONDING TO EIGENVALUE = -2.8639364E-04 -2.3361805E-05

  9.9999999E-01  0.              -1.6574362E-01 -8.7316478E-04    4.7279767E-01  1.2542876E-02    2.2805968E-02  3.4255458E-03
 -2.4669210E-04  2.1733336E-04
```

184

FLUTTER SOLUTION FOR 1/K = 1.000000E 01

EIGENVALUE-R	EIGENVALUE-I	FREQUENCY	G	V
-2.863936E-04	-2.336180E-05	IMAGINARY	ZERO	ZERO
3.348289E-06	-1.790936E-07	5.464979E 02	-5.348809E-02	8.197469E 04
3.183304E-06	-1.740429E-06	5.604811E 02	-5.467365E-01	8.407216E 04
2.808776E-06	-2.444672E-06	5.966800E 02	-8.703692E-01	8.950200E 04
7.221378E-07	4.531453E-08	1.176766E 03	6.275052E-02	1.765148E 05

4.6 PROGRAM LISTINGS

```
9          FORTRAN DECK
CMCFA            MODAL FLUTTER ANALYSIS PROGRAM
C
C        TAPE DESIGNATIONS -- IRTAPE = STANDARD INPUT TAPE
C                             IWTAPE = STANDARD OUTPUT TAPE
C        NCONT(1)  = REFERENCE TO INPUT WR
C        NCONT(2)  = REFERENCE TO INPUT BR
C        NCONT(3)  = REFERENCE TO INPUT DEGREE OF MATRIX A
C        NCONT(4)  = REFERENCE TO INPUT MIJ(I,J)
C        NCONT(5)  = REFERENCE TO INPUT MIJ(I,I)
C        NCONT(6)  = REFERENCE TO INPUT WI(I)
C        NCONT(7)  = REFERENCE TO INPUT PII(I,J)
C        NCONT(8)  = REFERENCE TO INPUT 1/K
C        NCONT(9)  = REFERENCE TO INPUT F, FACTOR FOR AERO MATRIX
C        NCONT(10) = REFERENCE TO INPUT BIJ(I,J)
C        NCONT(11) = REFERENCE TO INPUT AIJ(I,J)
C
         COMMON/ALL/IRTAPE,IWTAPE
         COMMON/M2IN12/AMTRX(20,40),ONEK,NEWA,WR,BR
         DIMENSION NCONT(11),CMIJ(20,40),WI(20),PII(20,2),AIJ(20,40),
      1            TEMP(6),T(20,40),PIISAV(20,2),TITLE(24),BIJ(20,40)
C
         IRTAPE = 5
         IWTAPE = 6
         KEK=
         READ(IRTAPE,5) (TITLE(I),I=1,24)
       5 FORMAT(12A6)
         WRITE(IWTAPE,10) (TITLE(I),I=1,24)
      10 FORMAT(1H1 25X,12A6// 25X,12A6////)
    1000 READ(IRTAPE,70)(NCONT(I),I=1,11)
      70 FORMAT (14I3)
         IF(KEK.EQ.0) GO TO 75
         WRITE(IWTAPE,74)
      74 FORMAT(1H1)
      75 WRITE (IWTAPE,71)
      71 FORMAT(1H0 48X,22HMODAL FLUTTER ANALYSIS//55X,10HINPUT DATA/0/)
C
         DO 300 I=1,9
         IF ( NCONT(I)) 80,300,80
      80 GO TO (90,110,120,125,140,190,200,240,290),I
      90 READ (IRTAPE,100) WR
         WRITE (IWTAPE,91) WR
      91 FORMAT(1H0 21HREFERENCE FREQUENCY = F16.8,4H RPS)
     100 FORMAT (6E12.8)
         GO TO 300
     110 READ (IRTAPE,100) BR
         WRITE (IWTAPE,111) BR
     111 FORMAT(1H0 22HREFERENCE SEMI-CHORD = F16.8)
         GO TO 300
     120 NDEGA   = NCONT(I)
         N2A     = 2*NDEGA
         WRITE (IWTAPE,121) NDEGA
     121 FORMAT(1H0 17HNUMBER OF MODES = I3//)
         GO TO 300
     125 IF(NCONT(I).EQ.1) GO TO 129
         DO 126 I=1,NDEGA
         DO 126 J=1,N2A
     126 CMIJ(I,J)=0.0
         DO 129 I=1,NDEGA
     129 READ(IRTAPE,100) (CMIJ(I,J),J=1,N2A,2)
```

```
      GO TO 131
  120 DO 130 I=1,NDEGA
  130 READ(IRTAPE,100) (CMIJ(I,J),J=1,N2A)
  131 WRITE (IWTAPE,132)
  132 FORMAT(1H0 23HGENERALIZED MASS MATRIX //)
      DO 134 I=1,NDEGA
      K       = 1
      DO 134 J=1,N2A,2
      WRITE (IWTAPE,133) I, K, CMIJ(I,J), CMIJ(I,J+1)
  133 FORMAT (2I3,2F16.8)
  134 K       = K+1
      GO TO 300
  140 CONTINUE
      DO 150 I=1,NDEGA
      DO 150 J=1,N2A
  150 CMIJ(I,J) = 0
      IF(ICONT(I).EQ.1) GO TO 155
      READ(IRTAPE,100) (CMIJ(I,2*I-1),I=1,NDEGA)
      GO TO 131
  155 J = 1
      I1 = 1
      I2 = 3
  160 READ (IRTAPE,100) (TEMP(NT),NT=1,6)
      NT = 1
      DO 180 I=I1,I2
      CMIJ(I,J) = TEMP(NT)
      CMIJ(I,J+1) = TEMP(NT+1)
      IF (N2A - (J+1)) 131,131,170
  170 J  = J+2
      NT = NT+2
  180 CONTINUE
      I1 = I2+1
      I2 = I1+2
      GO TO 160
  190 READ (IRTAPE,100) (WI(I),I=1,NDEGA)
      WRITE (IWTAPE,191) (WI(I),I=1,NDEGA)
  191 FORMAT(1H0 23HMODAL FREQUENCIES (RPS) //(8F16.8))
      GO TO 300
  200 I    = NCONT(I)
      GO TO (210,220), I
  210 READ (IRTAPE,100) ((PII(I,J),J=1,2),I=1,NDEGA)
      WRITE (IWTAPE,211) (I, I, (PII(I,J),J=1,2),I=1,NDEGA)
  211 FORMAT(1H0 35HDIAGONAL STIFFNESS PARAMETER MATRIX //(3(I2,I3,2F16.
     18,4X)))
      GO TO 300
  220 I1 = 1
      DO 230 I=1,NDEGA
      X   = WI(I)/WR
      PII(I,1) = CMIJ(I,I1)*X*X
      PII(I,2) = CMIJ(I,I1+1)*X*X
      I1 = I1+2
  230 CONTINUE
      WRITE (IWTAPE,2230) ((PII(I,J),J=1,2),I=1,NDEGA)
 2230 FORMAT(1H0 35HDIAGONAL STIFFNESS PARAMETER MATRIX //(3(2F16.8,4X))
     1)
      GO TO 300
  240 READ(IRTAPE,100) ONEK
      WRITE(IWTAPE,241) ONEK
  241 FORMAT(1H0 5H1/K = F16.8)
      GO TO 300
```

```
  290 READ(IRTAPE,100) F
      WRITE(IWTAPE,291) F
  291 FORMAT(1H0 31HFACTOR FOR AERODYNAMIC FORCES = F16.8//)
  300 CONTINUE
C
C
      IF(NCONT(10)) 329,350,329
  329 IF(NCONT(10).EQ.2) GO TO 339
      DO 330 I=1,NDEGA
  330 READ(IRTAPE,100)(BIJ(I,J),J=1,N2A)
      GO TO 340
  339 READ(IRTAPE,100)((BIJ(I,J),J=1,N2A),I=1,NDEGA)
  340 WRITE(IWTAPE,331)
  331 FORMAT(1H0 30HGENERALIZED AERODYNAMIC FORCES //)
      DO 333 I=1,NDEGA
      K       = 1
      DO 333 J=1,N2A,2
      WRITE (IWTAPE,332) I, K, BIJ(I,J), BIJ(I,J+1)
  332 FORMAT (2I3,2E16.8)
  333 K       = K+1
  350 IF(NCONT(11)) 370,410,370
  370 IF(NCONT(11).EQ.2) GO TO 390
      DO 380 I=1,NDEGA
  380 READ(IRTAPE,100)(AIJ(I,J),J=1,N2A)
      WRITE (IWTAPE,381)
  381 FORMAT(1H0 36HSUM OF MASS AND AERODYNAMIC MATRICES)
      DO 383 I=1,NDEGA
      K       = 1
      DO 383 J=1,N2A,2
      WRITE (IWTAPE,382) I, K, AIJ(I,J), AIJ(I,J+1)
  382 FORMAT (2I3,2E16.8)
  383 K       = K+1
      GO TO 410
  390 CONTINUE
      DO 400 I=1,NDEGA
      DO 400 J=1,N2A
  400 T(I,J) = F*BIJ(I,J)
      CALL [illegible](CMIJ,T,20,40,AIJ)
      WRITE(IWTAPE,2400)
 2400 FORMAT(/// 1X,45HCOMPUTED SUM OF MASS AND AERODYNAMIC MATRICES)
      DO 405 I=1,NDEGA
  405 WRITE (IWTAPE,2405)  (AIJ(I,J),J=1,N2A)
 2405 FORMAT(1H  /(1P2E15.7,2X,1P2E15.7,2X,1P2E15.7,2X,1P2E15.7))
  410 NEWA    = NDEGA
      NOLDA   = NDEGA
      NEW2A   = N2A
      DO 415 I1=1,NDEGA
      BIISAV(I1,1) = BII(I1,1)
  415 BIISAV(I1,2) = BII(I1,2)
      KR=0
      DO 510 I1=1,NDEGA
  420 IF (BIISAV(I1,1)) 510,430,510
  430 IF (BIISAV(I1,2)) 510,440,510
  440 K        = 2*I1 - 1
      KP1      = K+1
      KR=KR+1
      DO 4[illegible]0 I=1,NEWA
      DO 4[illegible]0 J=1,NEW2A,2
      T1       = AIJ(I,K)*AIJ(I1,J) - AIJ(I,KP1)*AIJ(I1,J+1)
      T2       = AIJ(I,K)*AIJ(I1,J+1) + AIJ(I,KP1)*AIJ(I1,J)
```

```
      T3     = AIJ(I1,K)*AIJ(I1,K)  + AIJ(I1,KP1)*AIJ(I1,KP1)
      T(I,J)     = AIJ(I,J)   - (AIJ(I1,K)*T1 + AIJ(I1,KP1)*T2)/T3
      T(I,J+1)   = AIJ(I,J+1) - (AIJ(I1,K)*T2 - AIJ(I1,KP1)*T1)/T3
  480 CONTINUE
      DO 481 I=1,NEWA
      DO 481 J=1,NEW2A
  481 AIJ(I,J) = T(I,J)
      NOLDA  = NEWA
      NOLD2A = 2*NOLDA
      NEWA   = NEWA - 1
      NEW2A  = 2*NEWA
      DO 490 I=I1,NEWA
      PIISAV(I,1) = PIISAV(I+1,1)
      PIISAV(I,2) = PIISAV(I+1,2)
      DO 490 J=1,NOLD2A
      AIJ(I,J) = AIJ(I+1,J)
  490 CONTINUE
      DO 500 J=K,NEW2A
      DO 500 I=1,NEWA
  500 AIJ(I,J) = AIJ(I,J+2)
      IF (I1-NOLDA) 420,510,510
  510 CONTINUE
      IF(KK.EQ.0) GO TO 506
      WRITE(IWTAPE,504)
  504 FORMAT(/// 1X,44HREDUCED SUM OF MASS AND AERODYNAMIC MATRICES)
      DO 505 I=1,NEWA
  505 WRITE (IWTAPE,2405)  (AIJ(I,J),J=1,NEW2A)
  506 DO 520 I=1,NEWA
      T1      = PIISAV(I,1)*PIISAV(I,1) + PIISAV(I,2)*PIISAV(I,2)
      DO 520 J=1,NEW2A,2
      AMTRX(I,J)    = (AIJ(I,J)*PIISAV(I,1) + AIJ(I,J+1)*PIISAV(I,2))/T1
      AMTRX(I,J+1) =(-AIJ(I,J)*PIISAV(I,2) + AIJ(I,J+1)*PIISAV(I,1))/T1
  520 CONTINUE
      WRITE(IWTAPE,2520)
      DO 525 I=1,NEWA
  525 WRITE (IWTAPE,2405) (AMTRX(I,J),J=1,NEW2A)
 2520 FORMAT(/// 1X,16HDYNAMICAL MATRIX)
      CALL MAIL2
      KKK=KKK+1
      GO TO 1000
      END
```

```
*       FORTRAN DECK
CMAIN2        COMPLEX EIGENVALUE SOLUTION AND FLUTTER RESULTS
C
      SUBROUTINE MAIN2
      COMMON/ALL/IRTAPE,IWTAPE
      COMMON/MAIN12/AMTRX(20,40),ONEK,NEWA,WR,BR
      DIMENSION EIGVAL(40)
      COMPLEX AMC(20,20),EIGC(20)
C
      DATA Q1/6HIMAGIN/,Q2/6HARY   /,Q3/6HINFINI/,Q4/6HTE    /,Q5/6HZERO
     1  /
      N = 2*NEWA
      DO 10 I=1,NEWA
      DO 10 J=1,N,2
      JJ=(J+1)/2
   10 AMC(I,JJ)=CMPLX(AMTRX(I,J),AMTRX(I,J+1))
      NCAL=NEWA
      CALL ALLMAT(AMC,EIGC,NEWA,20,NCAL)
      IF(NCAL.LT.NEWA) GO TO 12
      WRITE(IWTAPE,11) NCAL
   11 FORMAT(1H1 3X,51HNUMBER OF EIGENVALUES AND EIGENVECTORS CALCULATED
     1 = I3//)
      GO TO 14
   12 WRITE (IWTAPE,13) NCAL
   13 FORMAT(1H1 3X,51HNUMBER OF EIGENVALUES AND EIGENVECTORS CALCULATED
     1 = I3//4X,72HCONVERGENCE DID NOT OCCUR WITHIN TEN ITERATIONS FOR T
     2HE NEXT EIGENVALUE.//)
      IF(NCAL.EQ.0) RETURN
   14 DO 25 J=1,NCAL
      WRITE (IWTAPE,20) EIGC(J)
   20 FORMAT(1H0 3X,41HEIGENVECTOR CORRESPONDING TO EIGENVALUE = 1P2E15.
     17//)
      WRITE(IWTAPE,21) (AMC(I,J),I=1,NCAL)
   21 FORMAT(1H  2X,1P2E15.7,2X,1P2E15.7,2X,1P2E15.7,2X,1P2E15.7)
   25 CONTINUE
      WRITE(IWTAPE,15) ONEK
   15 FORMAT(//// 4X,26HFLUTTER SOLUTION FOR 1/K = 1P1E13.6 )
      WRITE(IWTAPE,16)
   16 FORMAT(1H0 3X,12HEIGENVALUE-R,6X,12HEIGENVALUE-I,7X,9HFREQUENCY,13
     1X,1HG,17X,1HV//)
C
      NCAL2=2*NCAL
      DO 30 I=1,NCAL2,2
      II=(I+1)/2
      EIGVAL(I)=REAL(EIGC(II))
      EIGVAL(I+1)=AIMAG(EIGC(II))
   30 CONTINUE
      DO 40 I=1,NCAL2,2
      DO 35 J=I,NCAL2,2
      SAX=EIGVAL(I)
      IF(ABS(SAX).GE.ABS(EIGVAL(J))) GO TO 35
      SAX=EIGVAL(J)
      SAXI=EIGVAL(J+1)
      EIGVAL(J)=EIGVAL(I)
      EIGVAL(J+1)=EIGVAL(I+1)
      EIGVAL(I)=SAX
      EIGVAL(I+1)=SAXI
   35 CONTINUE
   40 CONTINUE
      DO 170 I=1,NCAL2,2
```

```
      IF (EIGVAL(I)) 60,80,100
   60 WRITE(IWTAPE,70) EIGVAL(I),EIGVAL(I+1),Q1,Q2,Q5,Q5
   70 FORMAT(1H  1PE15.6,1P1E18.6,7X,2A6,8X,1A6,12X,1A6)
      GO TO 170
   80 WRITE(IWTAPE,90)EIGVAL(I),EIGVAL(I+1),Q3,Q4,Q3,Q4,Q3,Q4
   90 FORMAT(1H  1PE15.6,1P1E18.6,7X,2A6,6X,2A6,6X,2A6)
      GO TO 170
  100 WF     = WR/SQRT(EIGVAL(I))
  120 G      = EIGVAL(I+1)/EIGVAL(I)
      V = WF*RR*ONEK
      WRITE(IWTAPE,130) EIGVAL(I),EIGVAL(I+1),WF,G,V
  130 FORMAT(1H  1P1E15.6,1P4E18.6)
  170 CONTINUE
      RETURN
       END
```

```
*       FORTRAN DECK
CALLMAT
      SUBROUTINE ALLMAT(A,LAMBDA,M,IA,NCAL)                             ALLMO
C                                                                       ALLMO
C PROG.AUTHORS JOHN RINZEL,R.E.FUNDERLIC,UNION CARBIDE CORP.            ALLMO
C NUCLEAR DIVISION,CENTRAL DATA PROCESSING FACILITY,                    ALLMO
C OAK RIDGE TENNESSEE                                                   ALLMO
C                                                                       ALLMO
      COMPLEX A(IA,1), H(30,30) , HL(30,30), LAMBDA(1),VECT(30),        ALLMO
     1MULT(30),SHIFT(3),TEMP,SIN,COS,TEMP1,TEMP2                        ALLMO
      LOGICAL INTH(30),TWICE                                            ALLMO
      INTEGER INT(30),R,RP1,RP2                                         ALLMO
      NVEC = NCAL                                                       ALLMO
      N=M                                                               ALLMO
      NCAL=N                                                            ALLMO
      IF(N.NE.1)GO TO 1                                                 ALLMO
      LAMBDA(1)=A(1,1)                                                  ALLMO
      A(1,1)=1.                                                         ALLMO
      GO TO 57                                                          ALLMO
    1 ICOUNT=0                                                          ALLMO
      SHIFT(1) = 0.                                                     ALLMO
      IF(N.NE.2)GO TO 4                                                 ALLMO
    2 TEMP=(A(1,1)+A(2,2)+CSQRT((A(1,1)+A(2,2))**2-                     ALLMO
     14.*(A(2,2)*A(1,1)-A(2,1)*A(1,2))))/2.                             ALLMO
      IF(REAL(TEMP).NE.0..OR.AIMAG(TEMP).NE.0.)GO TO 3                  ALLMO
      LAMBDA(M)= SHIFT(1)                                               ALLMO
      LAMBDA(M-1) = A(1,1) + A(2,2)  + SHIFT(1)                         ALLMO
      GO TO 37                                                          ALLMO
    3 LAMBDA(M) =  TEMP + SHIFT(1)                                      ALLMO
      LAMBDA(M-1)=(A(2,2)*A(1,1)-A(2,1)*A(1,2))/(LAMBDA(M)-SHIFT(1))    ALLMO
     A              + SHIFT(1)                                          ALLMO
      GO TO 37                                                          ALLMO
C                                                                       ALLMO
C     REDUCE MATRIX A TO HESSENBERG FORM                                ALLMO
C                                                                       ALLMO
    4 NM2=N-2                                                           ALLMO
      DO 15 R=1,NM2                                                     ALLMO
      RP1=R+1                                                           ALLMO
      RP2=R+2                                                           ALLMO
      ABIG=0.                                                           ALLMO
      INT(R)=RP1                                                        ALLMO
      DO 5 I=RP1,N                                                      ALLMO
      ABSSQ=REAL(A(I,R))**2+AIMAG(A(I,R))**2                            ALLMO
      IF(ABSSQ.LE.ABIG)GO TO 5                                          ALLMO
      INT(R)=I                                                          ALLMO
      ABIG=ABSSQ                                                        ALLMO
    5 CONTINUE                                                          ALLMO
      INTER=INT(R)                                                      ALLMO
      IF(INTER.EQ.RP1)GO TO 8                                           ALLMO
      IF(ABIG.EQ.0.)GO TO 15                                            ALLMO
      DO 6 I=R,N                                                        ALLMO
      TEMP=A(RP1,I)                                                     ALLMO
      A(RP1,I)=A(INTER,I)                                               ALLMO
    6 A(INTER,I)=TEMP                                                   ALLMO
      DO 7 I=1,N                                                        ALLMO
      TEMP=A(I,RP1)                                                     ALLMO
      A(I,RP1)=A(I,INTER)                                               ALLMO
    7 A(I,INTER)=TEMP                                                   ALLMO
    8 DO 9 I=RP2,N                                                      ALLMO
      MULT(I)=A(I,R)/A(RP1,R)                                           ALLMO
```

```
    9 A(I,[illegible])=MULT(I)
      DO 11 I=1,RP1
      TEMP=0.
      DO 10 J=RP2,N
   10 TEMP=TEMP+A(I,J)*MULT(J)
   11 A(I,RP1)=A(I,RP1)+TEMP
      DO 13 I=RP2,N
      TEMP=0.
      DO 12 J=RP2,N
   12 TEMP=TEMP+A(I,J)*MULT(J)
   13 A(I,RP1)=A(I,RP1)+TEMP-MULT(I)*A(RP1,RP1)
      DO 14 I=RP2,N
      DO 14 J=RP2,N
   14 A(I,J)=A(I,J)-MULT(I)*A(RP1,J)
   15 CONTINUE
C
C     CALCULATE EPSILON
C
      EPS=0.
      DO 16 I=1,N
   16 EPS=EPS+CABS(A(1,I))
      DO 18 I=2,N
      SUM=0.
      IM1=I-1
      DO 17 J=IM1,N
   17 SUM=SUM+CABS(A(I,J))
   18 IF(SUM.GT.EPS)EPS=SUM
      EPS=SQRT(FLOAT(N))*EPS*1.E-12
      IF(EPS.EQ.0.)EPS=1.E-12
      DO 19 I=1,N
      DO 19 J=1,N
   19 H(I,J)=A(I,J)
   20 IF(N.NE.1)GO TO 21
      LAMBDA(M) = A(1,1) + SHIFT(1)
      GO TO 37
   21 IF(N.EQ.2)GO TO 2
   22 MN1=M-N+1
      IF(REAL(A(N,N)).NE.0..OR.AIMAG(A(N,N)).NE.0.)
     1 IF(ABS(REAL(A(N,N-1)/A(N,N)))+ABS(AIMAG(A(N,N-1)/A(N,N)))-1.E-9)
     2 24,24,23
   23 IF(ABS(REAL(A(N,N-1)))+ABS(AIMAG(A(N,N-1))).GE.EPS)GO TO 25
   24 LAMBDA(MN1) = A(N,N) + SHIFT(1)
      ICOUNT=0
      N=N-1
      GO TO 21
C
C     DETERMINE SHIFT
C
   25 SHIFT(2)=(A(N-1,N-1)+A(N,N)+CSQRT((A(N-1,N-1)+A(N,N))**2
     1 -4.*(A(N,N)*A(N-1,N-1)-A(N,N-1)*A(N-1,N))))/2.
      IF(REAL(SHIFT(2)).NE.0..OR.AIMAG(SHIFT(2)).NE.0.)GO TO 26
      SHIFT(3)=A(N-1,N-1)+A(N,N)
      GO TO 27
   26 SHIFT(3)=(A(N,N)*A(N-1,N-1)-A(N,N-1)*A(N-1,N))/SHIFT(2)
   27 IF(CABS(SHIFT(2)-A(N,N)).LT.CABS(SHIFT(3)-A(N,N)))GO TO 28
      INDEX=3
      GO TO 29
   28 INDEX=2
   29 IF(CABS(A(N-1,N-2)).GE.EPS)GO TO 30
      LAMBDA(MN1) = SHIFT(2) + SHIFT(1)
```

```
      LAMBDA(NM1+1) = SHIFT(3) + SHIFT(1)
      ICOUNT=0
      K=K-2
      GO TO 20
   30 SHIFT(1) = SHIFT(1) + SHIFT(INDEX)
      DO 31 I=1,K
   31 A(I,I)=A(I,I)-SHIFT(INDEX)
C
C     PERFORM GIVENS ROTATIONS, OR ITERATES
C
      IF(ICOUNT.LE.10)GO TO 32
      NCAL=M-N
      GO TO 37
   32 KM1=K-1
      TEMP1=A(1,1)
      TEMP2=A(2,1)
      DO 36 P=1,KM1
      RP1=P+1
      RHO=SQRT(REAL(TEMP1)**2+AIMAG(TEMP1)**2+
     1 REAL(TEMP2)**2+AIMAG(TEMP2)**2)
      IF(RHO.EQ.0.)GO TO 36
      COS=TEMP1/RHO
      SIN=TEMP2/RHO
      INDEX=MAX0(P-1,1)
      DO 33 I=INDEX,N
      TEMP=CONJG(COS)*A(P,I)+CONJG(SIN)*A(RP1,I)
      A(RP1,I)=-SIN*A(P,I)+COS*A(RP1,I)
   33 A(P,I)=TEMP
      TEMP1=A(RP1,RP1)
      TEMP2=A(P+2,P+1)
      DO 34 I=1,P
      TEMP=COS*A(I,P)+SIN*A(I,RP1)
      A(I,RP1)=-CONJG(SIN)*A(I,P)+CONJG(COS)*A(I,RP1)
   34 A(I,P)=TEMP
      INDEX=MIN0(P+2,K)
      DO 35 I=RP1,INDEX
      A(I,P)=SIN*A(I,RP1)
   35 A(I,RP1)=CONJG(COS)*A(I,RP1)
   36 CONTINUE
      ICOUNT=ICOUNT+1
      GO TO 22
C
C     CALCULATE VECTORS
C
   37 IF(NCAL .EQ. 0  .OR. NVEC .EQ. 0 ) GO TO 57
      K=N
      KM1=K-1
      IF(K.NE.2)GO TO 38
      EPS=AMAX1(CABS(LAMBDA(1)),CABS(LAMBDA(2)))*1.E-8
      IF(EPS.EQ.0.)EPS=1.E-12
      H(1,1)=A(1,1)
      H(1,2)=A(1,2)
      H(2,1)=A(2,1)
      H(2,2)=A(2,2)
   38 DO 56 L=1,NCAL
      DO 40 I=1,N
      DO 39 J=1,N
   39 HL(I,J)=H(I,J)
   40 HL(I,I)=HL(I,I)-LAMBDA(L)
      DO 44 I=1,KM1
```

NOT REPRODUCIBLE

```
      MULT(I)=0.                                                     ALLM
      INTH(I)=.FALSE.                                                ALLM
      IP1=I+1                                                        ALLM1
      IF(CABS(HL(I+1,I)).LE.CABS(HL(I,I)))GO TO 42                   ALLM1
      INTH(I)=.TRUE.                                                 ALLM
      DO 41 J=I,N                                                    ALLM1
      TEMP=HL(I+1,J)                                                 ALLM1
      HL(I+1,J)=HL(I,J)                                              ALLM
   41 HL(I,J)=TEMP                                                   ALLM
   42 IF(REAL(HL(I,I)).EQ.0..AND.AIMAG(HL(I,I)).EQ.0.)GO TO 44        ALLM1
      MULT(I)=-HL(I+1,I)/HL(I,I)                                     ALLM
      DO 43 J=IP1,N                                                  ALLM
   43 HL(I+1,J)=HL(I+1,J)+MULT(I)*HL(I,J)                            ALLM1
   44 CONTINUE                                                       ALLM1
      DO 45 I=1,N                                                    ALLM
   45 VECT(I)=1.                                                     ALLM1
      TWICE=.FALSE.                                                  ALLM1
   46 IF(REAL(HL(N,N)).EQ.0..AND.AIMAG(HL(N,N)).EQ.0.)HL(N,N)=EPS    ALLM
      VECT(N)=VECT(N)/HL(N,N)                                        ALLM
      DO 48 I=1,NM1                                                  ALLM1
      K=N-I                                                          ALLM
      DO 47 J=K,NM1                                                  ALLM
   47 VECT(K)=VECT(K)-HL(K,J+1)*VECT(J+1)                            ALLM2
      IF(REAL(HL(K,K)).EQ.0..AND.AIMAG(HL(K,K)).EQ.0.)HL(K,K)=EPS    ALLM2
                 0Y28/BL(K,K)                                        ALLM
      BIG=0.                                                         ALLM2
      DO 49 I=1,N                                                    ALLM2
      SUM=ABS(REAL(VECT(I)))+ABS(AIMAG(VECT(I)))                     ALLM
   49 IF(SUM.GT.BIG)BIG=SUM                                          ALLM
      DO 50 I=1,N                                                    ALLM2
   50 VECT(I)=VECT(I)/BIG                                            ALLM
      IF(TWICE)GO TO 52                                              ALLM
      DO 51 I=1,NM1                                                  ALLM2
      IF(.NOT.INTH(I))GO TO 51                                       ALLM2
      TEMP=VECT(I)                                                   ALLM
      VECT(I)=VECT(I+1)                                              ALLM2
      VECT(I+1)=TEMP                                                 ALLM2
   51 VECT(I+1)=VECT(I+1)+MULT(I)*VECT(I)                            ALLM
      TWICE=.TRUE.                                                   ALLM
      GO TO 46                                                       ALLM2
   52 IF(N.EQ.2)GO TO 55                                             ALLM
      NM2=N-2                                                        ALLM
      DO 54 I=1,NM2                                                  ALLM2
      N1I=N-1-I                                                      ALLM2
      N11=N-I+1                                                      ALLM
      DO 53 J=N11,N                                                  ALLM2
   53 VECT(J)=H(J,N1I)*VECT(N1I+1)+VECT(J)                           ALLM2
      INDEX=INT(N1I)                                                 ALLM
      TEMP=VECT(N1I+1)                                               ALLM
      VECT(N1I+1)=VECT(INDEX)                                        ALLM2
   54 VECT(INDEX)=TEMP                                               ALLM2
   55 DO 56 I=1,N                                                    ALLM
   56 A(I,L)=VECT(I)                                                 ALLM2
      DO 61   I = 1,NCAL                                             ALLM2
      TE      = 0.                                                   ALLM
      DO 59   I = 1,N                                                ALLM
      TEM     = CABS(A(I,J))                                         ALLM2
      IF (TE     .GT. TEM   ) GO TO 59                               ALLM
      I  = I                                                         ALLM
      TE      = TEM                                                  ALLM2
```

```
58 CONTINUE                                  ALLM2
   TEMP1 = A(L,J)                            ALLM2
   DO 60  I = 1,N                            ALLM2
60 A(I,J) = A(I,J) / TEMP1                   ALLM2
61 CONTINUE                                  ALLM2
62 RETURN                                    ALLM2
   END                                       ALLM2
```

REFERENCES

1. Hughes Aircraft Co. MSD-P69-144, Collocation Flutter Analysis Program, Vol. I-IV, NASC Contract 00019-68-C-0274, dated April 1969.

2. Hughes Aircraft Co. SSD80319R, Research Report No. 32-MARS-COMOSY and DYNOPT Computer Programs, dated August 1968.

3. Hurty, W. C., "Dynamic Analysis of Structural Systems Using Component Modes" AIAA Journal, Vol. 3, No. 4, April 1965, pp 678-685.

4. TRW Inc. Project Appollo Interim Rpt Structural Dynamics Characteristics Document, Contract No. NAS 9-4810, MSC-TRW Task ASPO-17, dated 1 March 1966.

5. J.P.L. Tech Rpt. 32-530, Dynamic Analysis of Structural Systems by Component Mode Synthesis, date January 1964.

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Hughes Aircraft Company, Missile Systems Division Fallbrook and Roscoe Boulevards Canoga Park, California 91304	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

Collocation Flutter Analysis Study II

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Final Report (April 1969 through April 1970)

5. AUTHOR(S) (First name, middle initial, last name)

Dynamics and Environment Section, D. R. Ulbrich

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
April 1970		

8a. CONTRACT OR GRANT NO. N00019-69-C-0427 b. THIS DOCUMENT IS SUBJECT TO SPECIAL EXPORT CONTROLS AND EACH TRANSMITTAL TO FOREIGN GOVERNMENTS OR FOREIGN NATIONALS MAY BE MADE ONLY WITH THE PRIOR APPROVAL OF COMMANDER, NAVAL AIR SYSTEMS c. d. COMMAND, AIR ____, WASHINGTON, D.C. 20360.	9a. ORIGINATOR'S REPORT NUMBER(S) 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

10. DISTRIBUTION STATEMENT

~~In addition to security requirements which apply to this document and must be met, each transmittal of this document outside the agencies of the U.S. Government must have prior approval of the commander NASC.~~

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Naval Air Systems Commands Department of the Navy Washington, D.C.

13. ABSTRACT

This study covers the development of a set of computer programs to perform flutter analysis. These programs supplement those of collocation flutter Analysis-Study I Contract No. 00019-68-C-0274. This study is presented in three volumes. Volume I contains a subsonic strip theory unsteady aerodynamics program and a supersonic pison theory unsteady aerodynamics program. Volume II contains unsteady aerodynamic generalized force programs for subsonic, transonic, and supersonic flight regimes. Volume III contains the structural analysis programs, FLUENC-100C and COMSYN, and the modal flutter analysis program.

DD FORM 1 NOV 65 1473

UNCLASSIFIED

Security Classification

UNCLASSIFIED

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
FLUTTER						
Vibration						
Onsteady aerodynamic forces						

UNCLASSIFIED

Security Classification