

USARTL-TR-77-52



**DOCUMENTATION OF HELICOPTER AEROELASTIC STABILITY
ANALYSIS COMPUTER PROGRAM (HASTA)**

Lawrence R. Sutton
RASA Division
Systems Research Laboratories, Inc.
1055 J. Clyde Morris Blvd.
Newport News, Va. 23602

AD A 053022

December 1977

Final Report

AD No. _____
DDC FILE COPY



Approved for public release;
distribution unlimited.

Prepared for
APPLIED TECHNOLOGY LABORATORY
U. S. ARMY RESEARCH AND TECHNOLOGY LABORATORIES (AVRADCOM)
Fort Eustis, Va. 23604

BEST AVAILABLE COPY

5 - 7

APPLIED TECHNOLOGY LABORATORY POSITION STATEMENT

This user's manual has been reviewed by Applied Technology Laboratory (ATL) and is considered to be technically sound. The user's manual has been prepared to provide personnel unfamiliar with the HASTA computer program the ability to apply the procedure to helicopter stability analysis. Because the subject of helicopter stability is inherently very complicated, the manual is primarily directed to experienced dynamicists and mathematicians. However, the test is not difficult to read and a series of simple examples has been provided to help make understanding the program as simple as possible. Therefore, the manual should provide the potential for direct application of the HASTA computer program by a user familiar with the fundamental helicopter dynamic system.

Mr. William E. Nettles of the Aeromechanics Technical Area served as the Army project engineer for this effort.

DISCLAIMERS

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

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

Trade names cited in this report do not constitute an official endorsement or approval of the use of such commercial hardware or software.

DISPOSITION INSTRUCTIONS

Destroy this report when no longer needed. Do not return it to the originator.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER USARTL-TR-77-52	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) DOCUMENTATION OF HELICOPTER AEROELASTIC STABILITY ANALYSIS COMPUTER PROGRAM (HASTA)	5. TYPE OF REPORT & PERIOD COVERED Final Report	6. PERFORMING ORG. REPORT NUMBER RASA/SRL-14-77-04
7. AUTHOR(s) Lawrence R. Sutton	8. CONTRACT OR GRANT NUMBER(s)	9. SECURITY CLASS. (of this report) Unclassified
10. PERFORMING ORGANIZATION NAME AND ADDRESS RASA Division Systems Research Laboratories, Inc. 1055 J. Clyde Morris Blvd. Newport News, VA 23602	11. CONTROLLING OFFICE NAME AND ADDRESS Applied Technology Laboratory, Research and Technology Laboratories (AVRADCOM) Fort Eustis, VA 23604	12. REPORT DATE Dec 1977
13. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	14. NUMBER OF PAGES 411 P.	15. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Tail Rotor Air Resonance Main Rotor Stability Dynamics Rotor Helicopter Vibration		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This report is the documentation manual for the computer program Helicopter Aeroelastic Stability Analysis (HASTA) developed under Army Contract Number DAAJ02-76-C-0032. The HASTA program predicts the air resonance behavior of coupled rotor/helicopter (support structure) systems including rotor drive shaft torsional flexibility, anisotropic gearbox support flexibility, and control system anisotropic flexibility. The consideration of rigid, articulated,		

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

409 781

hc

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

20. gimballed, teetering, flexstrap, and bearingless rotor systems is allowed. This manual contains: a description of the method of solution, a list of program variables, a discussion of the equations, a listing of the program, instructions on the use of the program, sample input and output listing, and additional information pertaining to the computer program.

ACCESSION for	
NTIS	White Section <input checked="" type="checkbox"/>
DDC	Buff Section <input type="checkbox"/>
UNANNOUNCED	<input type="checkbox"/>
JUSTIFICATION _____	
BY _____	
DISTRIBUTION/AVAILABILITY CODES	
Dist. A/TAIL and/or SPECIAL	
A	

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

TABLE OF CONTENTS

	<u>Page</u>
LIST OF ILLUSTRATIONS	5
LIST OF TABLES	7
INTRODUCTION	8
OVERALL PROGRAM CAPABILITIES	11
HELICOPTER/ROTOR SYSTEM MODEL	15
Rotor Blade Model	15
Rotor Configuration Model	22
Control System Model	24
Control Rod Model	26
Rotor Elastic Support Structure Model	27
Gearbox or Transmission Mounting Model	28
Rotor Drive Shaft Torsional Flexibility Model	28
COORDINATE SYSTEMS	30
Rotor Hub Coordinate Systems	32
Blade Coordinate Systems	35
Pitch Control Structure Coordinate Systems	43
Control System Coordinate Systems	49
Control Rod Orientation	53
Support Structure Coordinate Systems	55
Shaft-Rotor Hub Interface Relationships	57
GENERAL DISCUSSION OF EQUATIONS	59
Transfer Matrix Operations	59
Final Governing Matrix	64
Eigenvalue and Eigenvector Solution Method	69
COMPUTER PROGRAM METHOD OF SOLUTION	73
INTERPRETATION OF COMPUTER PROGRAM RESULTS	79
DESCRIPTION OF SUBROUTINES AND FUNCTIONS	82
COMPUTER PROGRAM SYMBOL DICTIONARY	90

	<u>Page</u>
USER'S GUIDE	125
Dimensions of Computer Array Variables	125
Description of Input Data	145
Description of Program Output	198
Sample Input	214
Sample Output	227
Additional Sample Cases	262
PROGRAM LISTING	287
REFERENCES	411

LIST OF ILLUSTRATIONS

<u>Figure</u>		<u>Page</u>
1	A Representative System That Can Be Considered	16
2	A General Blade Section	18
3	Simplified Sectionalization of a Blade Structure	19
4	Pitch Control and Blade Retention Structure for Three Different Types of Blade Models . .	23
5	Swashplate Control System Model and Fixed Coordinate System for Counterclockwise Rotating Rotor	25
6	Six Possible Rotor System Configurations . . .	31
7	Fixed Hub Coordinate Systems for Two Directions of Rotor Rotation	33
8	Fixed Hub Coordinate System and Rotating Hub Coordinate System Corresponding to the First Blade for Two Directions of Rotor Rotation . .	35
9	General Description of the Rotating Local Blade Coordinate System for a jth Blade Section	36
10	Description of the Rotating Hub Coordinate System Rotations Required for Definition of the jth Section Local Blade Coordinate System Orientation	38
11	Relationship Between the Local Blade Aerodynamic and Local Blade Coordinate Systems for the jth Blade Section	41
12	Description of Rotating Hub Coordinate System Rotations Required for Definition of the Pitch Arm Coordinate System Orientation	46

<u>Figure</u>		<u>Page</u>
13	Description of Torque Tube Coordinate System Rotations Required for Definition of the Pitch Arm Coordinate System	50
14	Description of the Relationship Between the Fixed Hub Coordinate System and the Fixed Control System Coordinate System	51
15	Description of the Relationship Between the Rotating Control System Coordinate System Associated with First Blade and the Fixed Control System Coordinate System	52
16	Relationship Between Control Rod Axis and Rotating Hub Coordinate System	54
17	Fixed Fuselage Structure Coordinate System and Examples of Local Fuselage Structure Coordinate Systems	56
18	Relationship of the Local Support Structure Coordinate System and Fixed Hub Coordinate System Required at Shaft-Rotor Hub Interface	58
19	Overall Iterative System Flow	74

LIST OF TABLES

<u>Table</u>		<u>Page</u>
1	Values of Norm for Normalization to Various Blade Tip Deflections for Various System Models	157
2	Control Parameter Input Locations and Corresponding Program Variables	168
3	Section Data Input Locations and Corresponding Symbols	180

INTRODUCTION

The information presented in this documentation report is specifically directed at providing a User's Manual for the Helicopter Aeroelastic Stability Analysis computer program (HASTA) developed under Army Contract Number DAAJ02-76-C-0032. The primary purpose of this report is to inform the reader of the capabilities of HASTA and to provide the information required to construct an input data deck and successfully execute the program. Information pertaining to the construction and operation of HASTA is also presented to facilitate minor program alterations which may be desirable for specific HASTA applications; for example, redimensioning of variables.

The HASTA program was developed to provide a suitable analysis for representing the interaction of a rotor with its aerodynamic and support environment such that the air and ground resonance stability characteristics of fully coupled helicopter/rotor systems can be adequately predicted. HASTA can be used to predict the complex modal behavior of a main or tail rotor operating in vacuo or in air, with or without the effects of rotor support structure flexibilities included. These rotor support structure flexibilities include anisotropic gearbox or transmission support flexibility, anisotropic control system flexibility, rotor drive shaft torsional flexibility, and anisotropic landing gear flexibility, in addition to flexibilities associated with an elastic support structure such as the helicopter fuselage. The HASTA program is not limited to the consideration of a single type of rotor, but can be applied to a variety of rotor types including rigid to fully articulated, gimbaled, teetering, flexstrap, and bearingless, the last type having pitch control provided to the blades through torque tubes.

The HASTA program is primarily a FORTRAN IV program developed for use on IBM 360/65 computer systems. However, to reduce program running time, several matrix multiplication-related subroutines are in assembler language. The program in its present form has been developed to operate efficiently within the operational constraints of the USAAVRADCOM IBM 360/65 in St. Louis through the Applied Technology Laboratory terminal. Therefore, the present version is limited to the consideration of rotor systems consisting of identical blades equally spaced azimuthally. This restricts array dimensions such that in combination with the use of an overlay structure, the requirement for a core storage of less than 250K is satisfied. High core requirements result due to the necessity of employing

double precision real and complex variables to achieve satisfactory convergence and accurate solution eigenvalues, eigenvectors, and mode shapes on an IBM 360/65 system. On elimination of the present program array dimension restrictions, a rotor consisting of identical blades not equally spaced azimuthally can be considered at a cost of higher core storage requirements and longer running time. The mathematical analysis on which HASTA is based allows for consideration of a rotor having non-identical blades. The program modifications necessary for considering non-identical blades are not complicated and can be easily implemented if the capability is required.

The CPU time required for a HASTA run is primarily dependent on the number of iterations executed and the dimensional size of the final matrix for which the determinant must be obtained on each iteration. The final matrix size is dependent on both the degree of interharmonic coupling allowed and the complexity of the coupled rotor/helicopter system of interest. The CPU time for a run is also dependent upon the number and complexity of the blade and fuselage representational sections. This latter CPU time dependency is of secondary importance in the determination of CPU running time. A rough estimate for the CPU time required for a run, based on times encountered for program execution on the AVRADCOM IBM 360/65 in St. Louis, is

$$\text{time} = \frac{(\# \text{ iterations allowed}) * \text{MXQ} * \text{MXQ} \text{ minutes}}{4000}$$

where the number of iterations allowed is equal to the sum of the iterations allowed for each starting eigenvalue of each case of the run and MXQ is the number of rows in the final matrix.

Auxiliary equipment required for execution of the computer program consists of a card reader, a line printer, a tape or disk storage unit, and a card punch. Input data for the HASTA program is presently read in on cards. The program results, which are printed in a complex variable form, consist of the solution eigenvalues and their corresponding mode shapes defined relative to both the local blade coordinate systems and to the hub (or disk plane) coordinate systems. The disk plane mode shapes are printed in both complex variable and polar form, the latter allowing quick assessment as to the degree of coupling and phasing relationship occurring between blade motions. If desired, the resulting mode shapes can be punched on cards for subsequent use. The program can be readily modified to read aerodynamic coefficient input data from tape or disk data sets

as well as redimensioned for specific applications. It is recommended that the user consult his systems programmer if any program modifications are to be made.

A detailed discussion of the basic mathematical analysis on which the HASTA analysis largely is based is contained in Reference 1. Additional mathematical analysis was developed and used in the construction of the HASTA analysis to extend its capabilities. This analysis is discussed in Reference 2.

-
1. Sutton, Lawrence R., and Gangwani, Santu T., THE DEVELOPMENT AND APPLICATION OF AN ANALYSIS FOR THE DETERMINATION OF COUPLED TAIL ROTOR/HELICOPTER AIR RESONANCE BEHAVIOR, USAAMRDL-TR-75-35, Eustis Directorate, US Army Air Mobility Research and Development Laboratory, Fort Eustis, Virginia, August 1975.
 2. Sutton, Lawrence R., and White, Jr., Richard P., THE DEVELOPMENT AND APPLICATION OF AN ANALYSIS FOR THE STABILITY BEHAVIOR OF BEARINGLESS MAIN ROTOR SYSTEMS, Letter Report SRL 14-77-3, RASA Division, Systems Research Laboratories, Inc., 1055 J. Clyde Morris Blvd., Newport News, Virginia.

OVERALL PROGRAM CAPABILITIES

The HASTA program can predict the aeroelastic stability behavior of fully coupled helicopter/rotor systems of various degrees of complexity in hover or forward flight. The program is applicable to both tail rotor and main rotor systems with several different types of rotor support allowed. With a rotor support structure and aerodynamic effects included in the system model either air resonance or ground resonance stability results may be obtained depending on the rotor support structure model and its end conditions (free or cantilevered to ground). When a rotor support structure is not involved in the system model or only a rotor control system is involved, the rotor hub is assumed to be cantilevered to ground and rotor air resonance stability results are obtained if aerodynamic effects are included. If aerodynamic effects are not included, coupled or uncoupled normal mode results may be obtained. In all cases the results consist of solution eigenvalues in complex notation containing the damping and frequency characteristics of the modes and the corresponding mode shapes, i.e., the force, moment, deflection, and slope distributions along the blade span.

The rotor configurations which can be considered by the present form of the HASTA analysis, due to program dimensional and coding restrictions, consist of those having any number of identical flexible blades equally spaced azimuthally. Several different types of rotors can be considered. These include a rigid to fully articulated (hinged blade) rotor, a gimballed rotor, a teetering rotor, a flexstrap rotor, and a torque tube-type bearingless rotor. The orientation and amplitude of the gravitational field and flight velocity to which the rotor is exposed are treated as part of the rotor configuration and can be considered.

HASTA represents the blades comprising the rotor by a lumped-parameter approach in which up to 35 spanwise sections of the blade are allowed. This representation allows for the inclusion of all blade characteristics believed to be of significance. These characteristics, which are discussed in detail in the next section, include the blade geometric, structural, and aerodynamic properties. Regarding the aerodynamic representation, HASTA is capable of using airfoil coefficient data provided in series coefficient form and/or tabular form, one of the acceptable forms being that of the C-81 data tables. The blade lumped-parameter approach also allows consideration of blades having an applied control torque, a pitch bearing,

a flap hinge, and/or a lead-lag hinge. In addition, the applied control torque-imposed forces and moments acting on the blades of the more recent flexstrap and bearingless rotor concepts can be considered. By properly combining the blade representation with the rotor hub boundary conditions, most rotor system configurations can be represented and investigated by the HASTA program.

The HASTA program has the capability to include the effects on the rotor aeroelastic stability behavior of an anisotropically supported flexible swashplate-type control system. Basically, in this control system the flexible swashplate is anisotropically supported by linear spring-damper units, such that in combination with the control rod stiffness and damping characteristics any desired control system collective and cyclic stiffness and collective and cyclic damping may be obtained. The HASTA program is capable of correctly representing the control system effects on system modal behavior when an elastic support structure is included, in that the linear spring-damper units are taken to be attached to the elastic support structure, instead of ground. Regarding the control rod representation, the program is only capable of considering the control rods to act in a direction parallel to the rotor drive shaft in the case of articulated, gimbaled, or teetering rotor systems. For a flexstrap or bearingless rotor, the program is capable of considering the orientation of the control rods in that control rod angularity is usually more prevalent in these rotor types. The control rod for all rotor types is taken to be attached to the swashplate and pitch arm by swivel ball joints such that the control rods cannot carry moments or transverse loadings.

HASTA has the capability of including an elastic support structure. This elastic support structure is also represented by a lumped-parameter approach similar to that used for the blade representation, but it is simplified in that the elastic support structure is not a rotating member. A maximum of 15 elastic support structure sections are allowed in the representation of the elastic support structure unless there are more than 25 blade sections. Then the condition that the total number of blade and elastic support sections cannot exceed 40 must be satisfied. All elastic support structure geometric, structural, and aerodynamic properties believed to be of significance can be considered by this lumped-parameter representation. Elastic support structure aerodynamics, besides being based on airfoil data of the series or tabular form, can also be based on blockage (flat plate drag) effects or linearized NACA 0012 airfoil data. The

program is also capable of considering the magnitude and orientation of the flight velocity vector acting on sections of the elastic support structure. The consideration of torsional spring-damper units on the elastic support structure provides the capability of representing anisotropic gearbox or transmission mounting stiffnesses and anisotropic landing gear stiffness. The landing gear stiffness representation used in combination with the elastic support structure cantilevered to ground allows the capability to determine ground resonance characteristics.

A fully coupled helicopter/rotor system will experience interharmonic coupling of motions. That is, motions at a given frequency will couple with motions occurring at n/rev above and below the given frequency, where n is an integer. In particular, the aerodynamic environment acting on a blade will provide interharmonic coupling of motions of the blade as a result of aerodynamic damping effects. The inclusion of a swashplate control system having collective and cyclic stiffnesses will provide interharmonic coupling of the motions of one blade to the motions of another. The existence of an elastic support structure in the system will provide interharmonic coupling between blade motions and the support structure motions. HASTA is capable of including these interharmonic coupling effects.

A significant feature of the HASTA program is its capability to make use of a predefined phasing relationship between the motions of the blades of a rotor (blade phasing option). This option, however, is only applicable to rotor systems having the blades equally spaced azimuthally. By specification of the type of blade motion of interest; e.g., umbrella, forward cyclic, backward cyclic, and reactionless; and the type of rotor system of interest, the motion of each rotor blade can be defined in terms of a reference blade. This allows the analysis to consider only one blade instead of several blades so that the size of the problem is much smaller. Thus, the HASTA analysis running time is substantially shorter when this option is used. In addition, solution eigenvalues associated with blade motion relationships other than that specified are removed from the problem, which facilitates a more rapid convergence to the desired results. Use of this option, due to the present dimensional restrictions imposed to satisfy core requirements, is the only manner in which HASTA can be applied to investigations of the aeroelastic stability characteristics of rotor systems at the present time.

In addition to obtaining solution eigenvalues by iterating from inputted starting eigenvalues based on prior experience or engineering judgment, HASTA is also capable of performing a scanning procedure to facilitate the determination of solution eigenvalues. This eigenvalue scanning procedure, based on input, determines the final matrix determinant values in complex variable form for a rectangular grid of stability and frequency values over a range of stability (eigenvalue real part) and frequency (eigenvalue imaginary part) values. An interpolation scheme is then used to determine possible solution eigenvalues, which are then treated as starting eigenvalues. Generally, due to the number of grid points required to avoid missing possible roots, the efficiency of using this procedure to obtain solution eigenvalues is poor compared to the use of guessed starting eigenvalues. Thus, this procedure should only be used as a last resort.

HELICOPTER/ROTOR SYSTEM MODEL

The HASTA computer program was developed from an analysis which models helicopter/rotor dynamic systems of varying degrees of complexity. These systems, besides having a rotor comprised of flexible blades subjected to various restraint conditions imposed by hinges and the manner in which control torque is provided, may have a control system, various local rotor support conditions, and an elastic rotor support structure. A knowledge of the types of system configurations which might be treated is necessary to use the HASTA program properly. The system model configurations allowed may consist of several basic model components which can be categorized as:

1. Rotor blade structure
2. Rotor hub restraints
3. Control system
4. Control rod configurations
5. Elastic rotor support structure
6. Gearbox or transmission mounting flexibilities
7. Rotor drive shaft torsional flexibility

The computer program can mathematically model each of these basic model components individually and/or in combination with other basic model components. When all of the model components are included, the coupled and interdependent modal behavior of a total system such as that depicted in Figure 1 may be predicted. A description of the modeling of the individual system components is given in the following sections.

ROTOR BLADE MODEL

The basic blade model allows for the inclusion of all blade characteristics believed to be of significance in the determination of blade modal behavior. These are as follows:

1. Orientation of the blade shear center axis by pre-cone and presweep distributions
2. Location of the blade shear center axis (axis about which the blade cross section will rotate when perturbed)

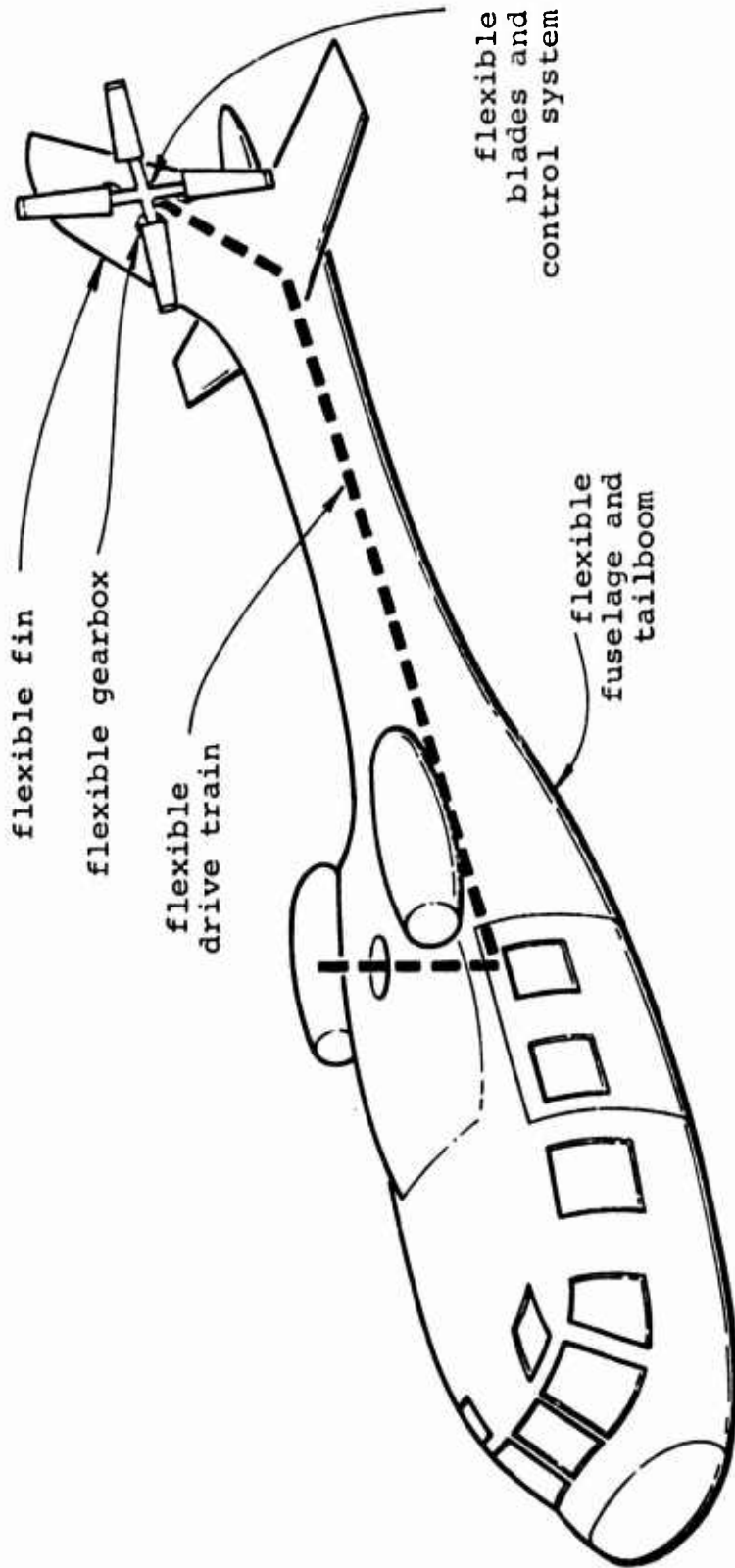


Figure 1. A Representative System That Can Be Considered.

3. Localized rigid offsets of the shear center axis in the flapwise, edgewise, and spanwise directions (a spanwise offset denotes that the blade is rigid over this offset distance)
4. Variable twist distribution about the blade shear center axis including collective pitch
5. Mass distribution and edgewise, flapwise, and torsional inertia distributions
6. Edgewise location of the center of mass relative to the blade shear center
7. Edgewise, flapwise, and torsional bending stiffness distributions and inclusion of centrifugal stiffening effects
8. Localized torsional spring-damper unit application about the flapwise, edgewise, and spanwise directions
9. Structural damping
10. Gravitational perturbation moment effects
11. Edgewise location of blade aerodynamic center axis relative to blade midchord
12. Chord length distribution
13. Aerodynamic effects including cyclic pitch, aerodynamic damping and Theodorsen's unsteady aerodynamic effects
14. Up to five different airfoil sections along blade span with the aerodynamic coefficients determined from either series coefficient data and/or airfoil table data

The blade characteristics listed above are modeled by utilizing a lumped-parameter approach in which the blade is represented by consecutive sections from the blade tip to root. In this sectional representation, the blade is divided into a finite number of sections. Each blade section is allowed a specified orientation (precone, presweep, and pitch) and is located in space by specification of the location of its inboard end. In addition, each section may consist of the following characteristic groups: shear center axis rigid offsets including a rigid

spanwise length, local bends in the shear center axis, a centrifugally stiffened elastic length or either flexstrap or bearingless rotor pitch control-imposed restraints, concentrated torsional spring-damper units, mass and inertias, and aerodynamics. A pictorial representation of these blade characteristic groups and the order in which they are considered, proceeding outboard to inboard along the blade, is given in Figure 2. The consideration of local shear center axis bends allows for the change in the shear center axis orientation from one section to another.

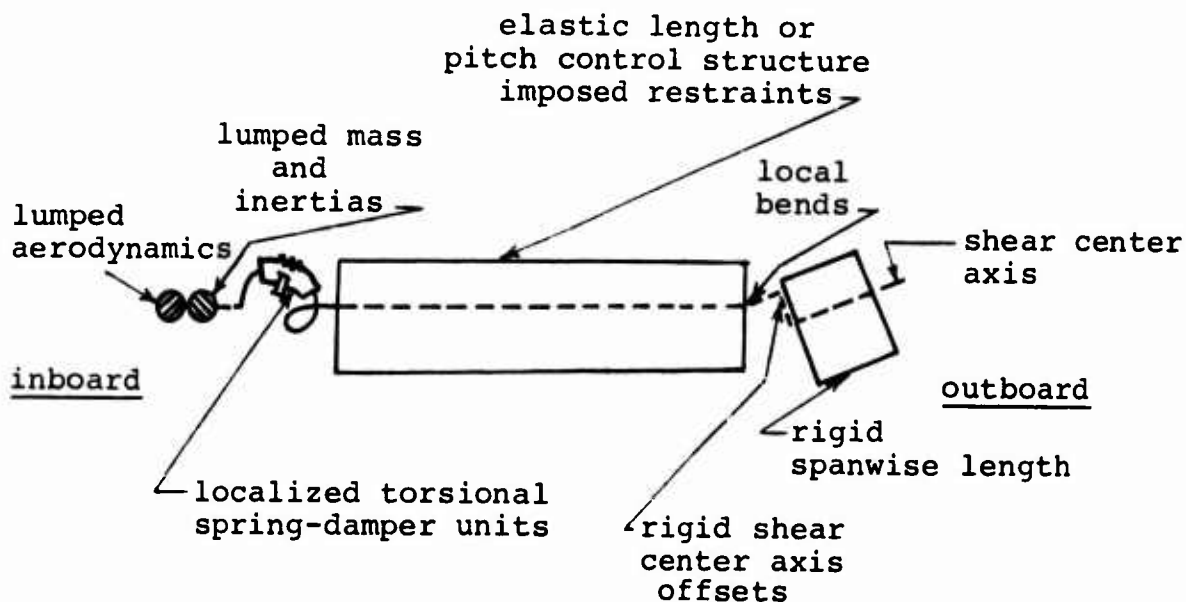


Figure 2. A General Blade Section.

The concentrated spring-damper units, shear center axis bends, and rigid offsets are allowed to occur relative to the local blade edgewise, flapwise, and spanwise directions at their point of application. Up to 35 blade sections may be used to represent the blade as a piecewise continuous structure. As an example of the lumped-parameter approach, a sectionalization of a blade having only mass, elastic, and coning properties is depicted in Figure 3.

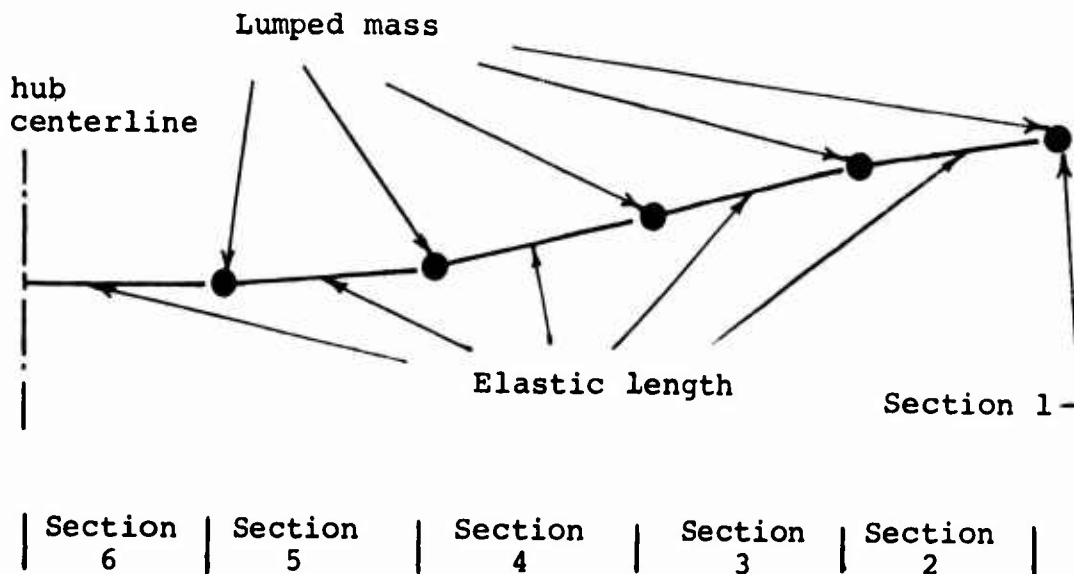


Figure 3. Simplified Sectionalization of a Blade Structure.

In addition to representing the basic blade geometric, structural, and aerodynamic characteristics, the blade model must also allow for the consideration of blade geometric and loading discontinuities which occur in the different rotor configurations allowed due to hinges on the blade and/or the manner of pitch control. An articulated rotor blade, besides having an applied control torque and pitch bearing, may have a flap hinge and/or a lead-lag hinge. The representation of a flap hinge, if required in the blade model, is accomplished in either of two ways. One way is to use a blade section with a torsional spring-damper unit (an allowed basic blade characteristic) having required torsional stiffness and damping value, with the condition that the axis of the spring-damper unit coincides with the flap hinge axis. This representation would be used in cases where the oscillatory flapping motion about the flap hinge is externally restrained (damped). The alignment of the torsional spring-damper unit axis with the flap hinge is achieved by proper location and orientation (through the use of section precone, presweep, and pitch angles) of the section containing the spring-damper unit.

The second way is to represent the flap hinge analytically by considering a discontinuity in the oscillatory flapping motion to occur at the flap hinge location and the condition that the local oscillatory flapwise moment must be zero at the hinge. For example, the oscillatory flapping motion of a rotating rigid blade attached to a rigid rotor hub by a flap hinge can be considered to be a flap angle discontinuity at the flap hinge. This method of representing a flap hinge does not allow inclusion of external damping of the oscillatory flapping motion. Both of these flap hinge models are included in the computer program to allow the user the choice of the most suitable model for particular program applications.

The representation of a lead-lag hinge, if required, is accomplished in either of two ways in a fashion similar to that of the flap hinge representation. The first way is to use a blade section with a torsional spring-damper unit having the required torsional stiffness and damping values and having its axis coincident with the lead-lag hinge axis. This representation would be used if the oscillatory lead-lag motion about the lead-lag hinge is externally damped. The second way is to represent the lead-lag hinge analytically by considering a discontinuity in the oscillatory lead-lag motion to occur at the lead-lag hinge location and the condition that the local oscillatory edgewise moment must be zero at the hinge. External damping of the oscillatory lead-lag motion is not allowed in this representation of the lead-lag hinge. The computer program includes both of these lead-lag hinge representational models to allow the user the choice of the most suitable model.

The representation of a pitch bearing, if required, is accomplished by considering a discontinuity in the oscillatory pitching motion to occur at the pitch bearing location and the condition that the local oscillatory torque at the pitch bearing must be zero. The concept of a pitch angle discontinuity is similar to that of flap and lead-lag angle discontinuities except that it occurs about the pitch bearing axis. The control rod effects on an articulated blade, if they are to be included, are represented by considering the control rod to apply an oscillatory torque (torque discontinuity) to the blade shear center axis at the effective spanwise application point of this torque. The torque discontinuity is normally considered to be applied outboard of the pitch bearing location.

The models for the flap and lead-lag hinges can represent large pitch-flap coupling δ_3 and pitch-lag coupling α_1 effects. This is possible since the flap and lead-lag hinge

axes may be placed in any desired orientation by the use of local section precone, presweep, and pretwist angles. An alternate model for representing pitch-flap and pitch-lag coupling is also allowed. In this model the flap and lead-lag hinges are taken to act about the local edgewise and flapwise axes, respectively, with the required coupling of blade motions taken into account on specification of pitch-flap and pitch-lag coupling factors.

A blade of the flexstrap or bearingless type does not have flap or lead-lag hinges or a pitch bearing. Instead, the flapwise, edgewise, and torsional motions inboard of the effective pitch control point are allowed by the flexibilities inherent in the blade retention structure. While these flexibilities can be represented by the basic blade model elastic properties, the relationship between the elastic blade motions is also dependent upon the oscillatory forces and moments in three mutually orthogonal directions acting on the blade shear center axis due to the restraint provided by the blade pitch control structure.

In the flexstrap-type blade model the control rod is assumed to be attached to a flexible pitch arm having length and having orientation in three mutually orthogonal directions. This pitch arm is considered to be rigidly attached to the blade. The oscillatory forces and moments acting on the blade shear center axis at the effective pitch control point can be defined in terms of the pitch arm properties, local blade perturbation (oscillatory) slopes and deflections, and the oscillatory displacements of the control rod attachment point to the pitch arm. Thus, instead of dealing with six force and moment discontinuities, the three control rod attachment point oscillatory displacements are considered as discontinuity quantities (restraint unknowns).

In the bearingless-type blade model that can be treated by the computer program, the control rod is assumed to be attached to a rigid pitch arm having length and having orientation in three mutually orthogonal directions. The pitch arm is considered to be rigidly attached by a fitting to the inboard end of a flexible torque tube having length and having orientation in three mutually orthogonal directions. The torque tube at its outboard end is taken to be rigidly attached to the blade at the blade shear center axis. The pitch arm-torque tube fitting also is considered to have a shaft (spur) extending inboard along the torque tube axis which is restrained by a spherical bearing attached to the rotor hub. This spur, which is allowed flexibility in three mutually orthogonal directions, is free to slide in the direction of its lengthwise axis and to rotate about three mutually orthogonal axes

at the spherical bearing. The oscillatory forces and moments acting on the blade due to this torque tube pitch control structure can be defined in terms of the properties of this structure, local blade perturbation (oscillatory) slopes and deflections, control rod forces, and spherical bearing translations. The three control rod forces and three spherical bearing translations constitute the discontinuity quantities or restraint unknowns for the bearingless-type blade model.

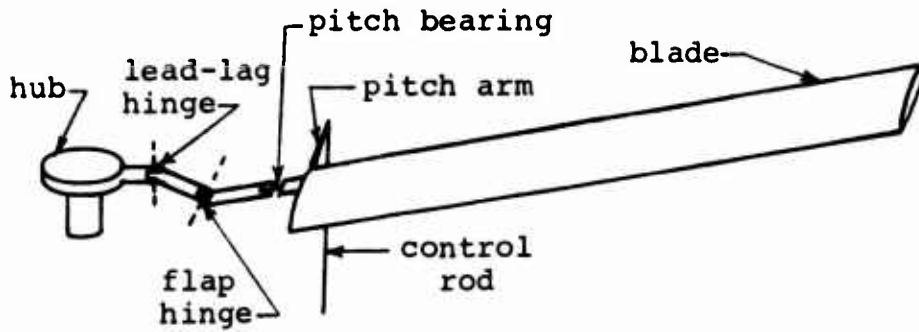
In the flexstrap or bearingless-type blade models, the pitch-flap δ_3 and pitch-lag α_1 couplings are automatically included and are directly related to the blade retention strap or beam and the pitch control structure representation. A significant amount of elastic coupling of flapwise, edgewise, and torsional deflections can occur with both of these blade models due to the flexibilities inherent in the retention strap or beam structure and the fact that the control rod does not remove all of the blade torque. Since the elastic couplings are very dependent on the rapidly changing orientation of the retention strap or beam stiffness parameters, a sufficient knowledge of the mean coning, lag, and pitch along the retention structure due to built-in and mean elastic deformation is required. Because of these elastic couplings, the retention strap or beam must be represented by more lumped-parameter sections for a given spanwise length than is required in the representation of the rest of the blade.

The pitch control and blade retention structure for the three types of blade models discussed above are depicted in Figure 4. The blade models necessary for considering a teetering or gimballed rotor are achieved by using the articulated blade model with the required representational options.

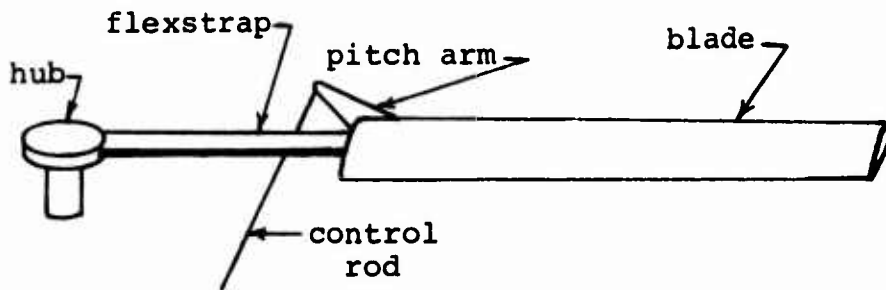
ROTOR CONFIGURATION MODEL

The rotor configuration model considered in the development of the representational analysis may consist of any number of flexible blades which can be arbitrarily located azimuthally. However, as noted in the introduction, the computer program was limited to rotor configurations consisting of identical flexible blades by the program coding and to rotor configurations consisting of any number of blades equally spaced azimuthally by restriction of array dimensions. These rotor configuration program limitations can be easily removed by increasing variable dimensions where required and by modifying the analytical coding.

Articulated blade



Flexstrap blade



Bearingless blade

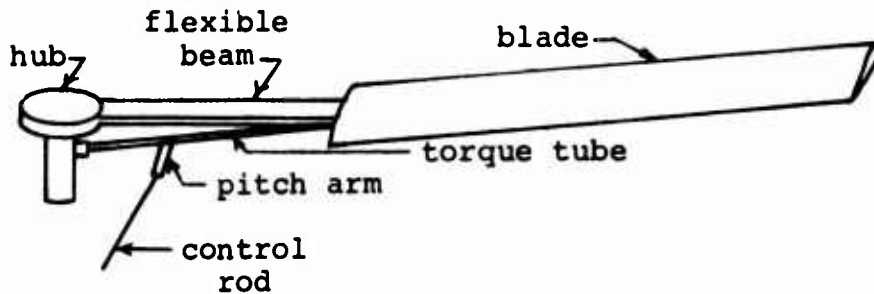


Figure 4. Pitch Control and Blade Retention Structure for Three Different Types of Blade Models.

The rotor configuration models allow consideration of the orientation and amplitude of the gravitational field and the rotor hub flight velocity. These orientations are specified relative to the reference (advancing) blade position. The rotor configuration models can be considered to be rotating in either a clockwise or counterclockwise (conventional) direction. The blades of all rotor configuration models which can be considered by the computer program can be considered to be cantilevered to a rigid rotor hub which may have degrees of freedom relative to its attachment to the rotor shaft. Types of rotor configuration models that can be considered are:

1. A rigid rotor model that is constructed with the articulated blade model (flap and lead-lag hinge not included)
2. A partial to fully articulated rotor model constructed with the articulated blade model
3. A flexstrap rotor model constructed with the flexstrap blade model
4. A bearingless rotor model constructed with the bearingless blade model
5. A gimbaled rotor (more than two blades) constructed with the articulated blade model
6. A teetering rotor (two blades only) constructed with the articulated blade model

The first four types of rotor models listed above require the rigid rotor hub to be cantilevered to the rotor shaft. The gimbaled rotor model assumes the rigid rotor hub to be attached to the rotor shaft such that it is free to rotate about two mutually orthogonal axes rotating in the plane perpendicular to the rotor shaft at the rotor hub attachment point. The teetering rotor model assumes the rigid rotor hub to be attached to the rotor shaft such that it is free to rotate about one axis rotating in the plane perpendicular to the rotor shaft at the rotor hub attachment point.

CONTROL SYSTEM MODEL

The model of the rotor control system is based on the assumption of a swashplate-type control system. The main component of the swashplate control system model, depicted in Figure 5,

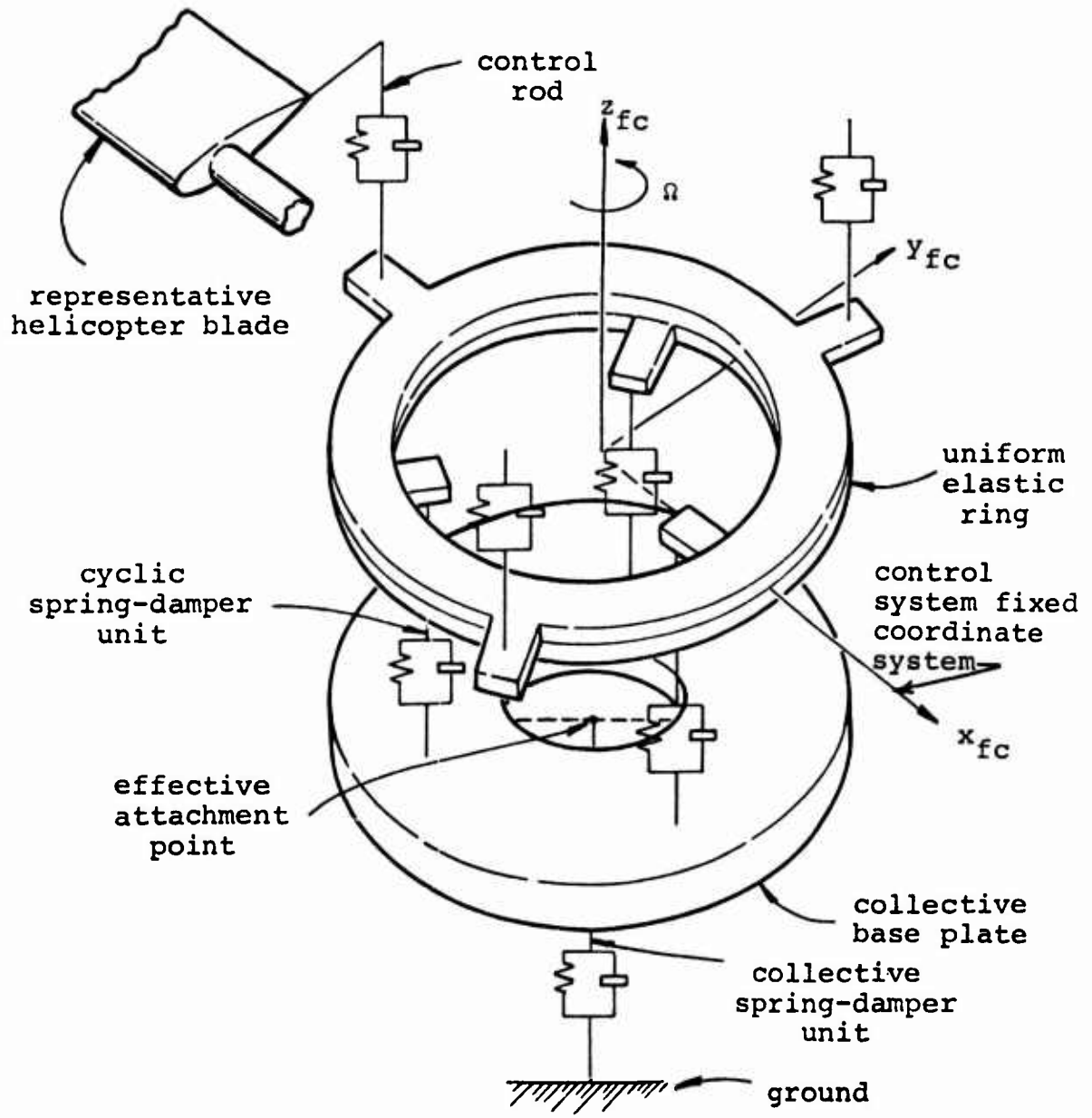


Figure 5. Swashplate Control System Model and Fixed Coordinate System for Counterclockwise Rotating Rotor.

is represented by a flexible ring having uniform mass distribution around its circumference and consisting of upper and lower portions. Both portions of the ring are allowed to translate along the rotor shaft axis (z_{fC} -axis) and rotate about two mutually orthogonal axes perpendicular to the rotor shaft axis. The upper portion of the ring also rotates with the blades about the rotor shaft axis. The lower portion of the ring, which does not rotate with the blades, is supported by a finite number of supports located azimuthally around the ring. These supports have linear stiffness and damping characteristics. The collective base plate to which the ring supports are attached is assumed to be attached to ground or to the rotor support structure, if included in the system model, by a linear support having an effective stiffness and damping value. The forces parallel to the rotor drive shaft axis acting on the swashplate from the control rods are passed through the swashplate control system model and are applied to rotor support structure, if included. Thus, by variation of the stiffness and damping characteristics and azimuthal location of the supports involved, any degree of control system anisotropic flexibility can be modeled.

CONTROL ROD MODEL

There are two types of control rod models which can be considered by the HASTA program. One type can only be used in conjunction with the articulated blade model. The other type can only be used with flexstrap and bearingless blade models. In both control rod models, the control rod is assumed to have axial stiffness and damping characteristics and to be connected to the pitch arm and swashplate (or ground, if there is no swashplate) by swivel ball joints. Attached in this manner, the control rods cannot carry moments or transverse loadings. In the control rod model used with an articulated blade model, the control rod is assumed to apply only control torque to the blade and to act along a line parallel to the rotor drive shaft axis.

The control rod model used with a flexstrap or bearingless blade model is much more complex than that which is used with an articulated blade model. The primary reasons for this model complexity are: (1) the large angularity of the control rod occurring in these rotor types; and (2) the strong elastic coupling created by the pitch control structure in the retention strap or beam flapwise, edgewise, and torsional degrees of freedom. In addition, because of the angularity of the control rod and the offset due to the pitch arm, blade motions in the inplane and flapwise directions are coupled

through the control system. Thus, in addition to control rod stiffness and damping characteristics, the orientation and location of the control rod must also be considered.

The stiffness and damping characteristics of the control rods are also included with those associated with the swashplate representation, discussed previously, in order to adequately represent the cyclic and collective stiffness and damping acting on the blades due to the control system and control rods.

ROTOR ELASTIC SUPPORT STRUCTURE MODEL

The rotor elastic support structure model allows for the inclusion of all geometric, structural, and aerodynamic characteristics believed to be of significance. These characteristics are essentially the same as those which were listed for the blade model. Since the support structure is not subjected to a constant rotational speed, the mass, inertia, and aerodynamic effects on the support structure behavior differ from those for a blade. In particular, the support structure mass and inertia distributions do not provide centrifugal stiffening or damping effects, and the support structure aerodynamic environment does not provide interharmonic coupling of support structure motions. The support structure aerodynamics can be based on blockage (flat plate drag) effects or a series representation (linearized aerodynamics) for a NACA 0012 airfoil section. The magnitude and orientation of the steady air velocity acting on the support structure is allowed to vary along the support structure length.

The rotor elastic support structure characteristics are modeled by using a lumped-parameter approach similar to the technique used to model the blade characteristics. The support structure is represented by consecutive sections from its tip to its attachment to the rotor hub. The tip of the support structure is generally that part of the support structure farthest from the rotor hub and is allowed to be either free in space or cantilevered to ground. Thus, for the fully coupled system depicted in Figure 1, the fuselage-tailboom-fin structure would be represented by consecutive sections from the front end of the fuselage, which would be treated as free in space, to the tail rotor hub.

The lumped-parameter sectional representation is used for the entire rotor support structure of interest, including the rotor gearbox or transmission mounting flexibilities and the rotor drive shaft, except for the drive shaft torsional

flexibility, which is treated in a different manner. Because of the similarity of the support structure and blade modeling techniques, Figure 2 is also valid for a support structure section on the condition that the outboard end of the section depicted is taken to correspond to the tipward end and the inboard end is taken to correspond to the end of the support structure section toward the rotor. Thus, the order in which the sectional characteristics can be considered in going outboard to inboard along a blade section is identical to the order that can be considered in going from the tipward end to the rotor attachment end of the support structure section. The section characteristics corresponding to the pitch control-imposed restraints which were allowed for a blade section are not allowed for a support structure section. A maximum of 15 support structure sections can be used to represent the entire support structure unless more than 25 blade sections are used in modeling the blade structure. Then the condition that the total number of blade and support structure sections cannot be greater than 40 must be satisfied.

GEARBOX OR TRANSMISSION MOUNTING MODEL

Rotor gearbox or transmission mounting flexibilities are represented by using the localized torsional spring-damper unit capabilities of a rotor support structure section. In particular, localized torsional spring-damper units having torsional stiffness and damping characteristics are applied about two mutually orthogonal axes at the attachment of the gearbox or transmission to its support structure (e.g., the fuselage). The specific orientation of the two axes relative to the support structure is accomplished through the use of the section orientation angles. By variation of the stiffness and damping characteristics of the two torsional spring-damper units, any degree of anisotropy in rotor gearbox or transmission mounting can be modeled.

ROTOR DRIVE SHAFT TORSIONAL FLEXIBILITY MODEL

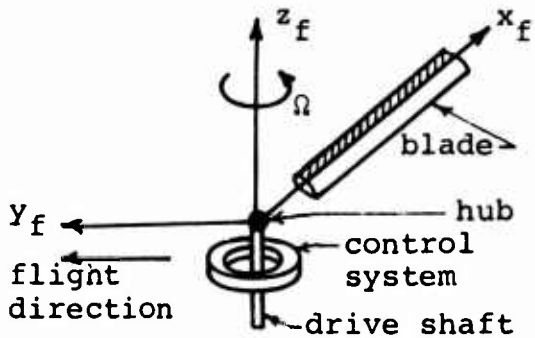
In representing the drive shaft torsional flexibility, the rotor drive shaft, whether it be for a main or tail rotor, is assumed to be constrained in torsion at the main rotor transmission. The model of the torsional characteristics of the drive shaft system was formulated by considering a localized torsional spring at the root of each blade whose in-plane flexibility is equivalent to the drive shaft torsional flexibility. The blade edgewise moments at the rotor hub are not allowed to provide torque to the shaft and thereby to the rest of the support structure since the torque on the rotor

shaft is assumed to be removed by the transmission. This torsional spring model is considered independent of the blade and support structure sectional representation and should not be construed to be modeled by use of blade or support structure section characteristics.

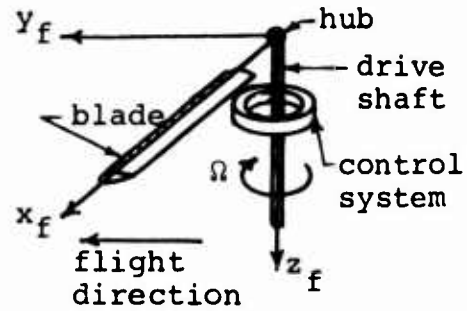
COORDINATE SYSTEMS

An understanding of the fundamental coordinate systems associated with the representation of the rotor/helicopter system model components and their relative orientation and use in defining program input variables is a prerequisite to proper use of the HASTA program. The information presented in this section is directed toward providing the user with this understanding. In particular, the coordinate systems pertaining to the representation of the rotor hub, blade structure, pitch control structure, control system, and rotor elastic support structure, and the manner in which they are related for main or tail rotors rotating clockwise or counterclockwise, are described. The coordinate system relationships required at the rotor shaft-rotor hub interface for system representational continuity are also discussed. The coordinate system-related definitions of the gravitational field and free stream velocity orientation variables are not provided here but are presented in the Description of Input Data section of this report.

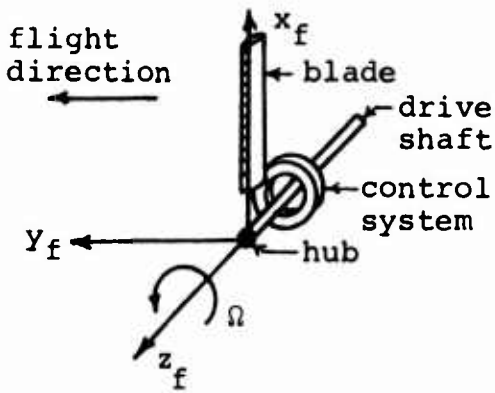
In the following discussion, two sets of coordinate systems are presented simultaneously for rotating system model components. One set of coordinate systems is pertinent to the consideration of a counterclockwise rotating rotor system. The second set of coordinate systems is associated with the consideration of a clockwise rotating rotor system. The direction of rotation of a rotor system is based on that observed from above (main rotor) or from the port side of a helicopter (tail rotor). Six fundamental rotor system arrangements (two main rotor and four tail rotor) which are determined by combinations of the rotor system rotational directions and the location of the rotor attachment are depicted in Figure 6 as viewed from the port side of a helicopter. In this figure, the rotor disk plane for a rotor system arrangement coincides with the corresponding $x_f - y_f$ plane where the (x_f, y_f, z_f) coordinate systems shown are the fixed (non-rotating) hub coordinate systems for the various rotor system arrangements. The detailed definition of fixed hub coordinate systems for counterclockwise and clockwise rotating rotor systems is provided below. In Figure 6 a rotor blade with its leading edge shaded is shown in the reference blade position (i.e., along the x_f axis) for each rotor system arrangement, to further clarify the direction of rotation. The control system, represented by a ring, and the rotor drive shaft are depicted for each rotor system arrangement to show that although the fixed hub coordinate systems may be identical for two rotor arrangements, the arrangements differ in the location of drive shaft and control system. Specifically, rotor system arrangements



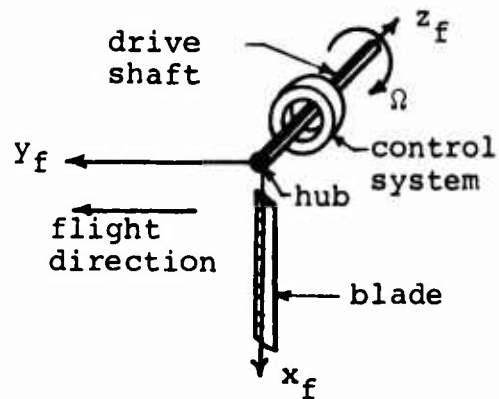
I. Main Rotor - Class A



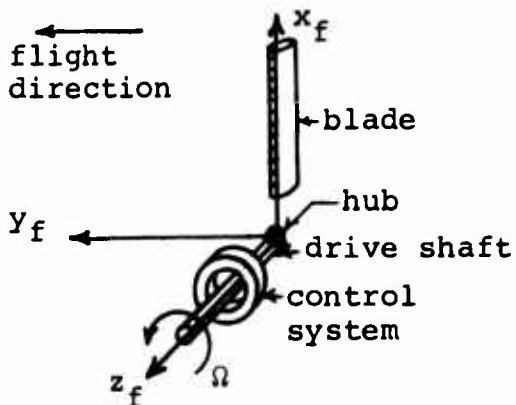
IV. Main Rotor - Class B



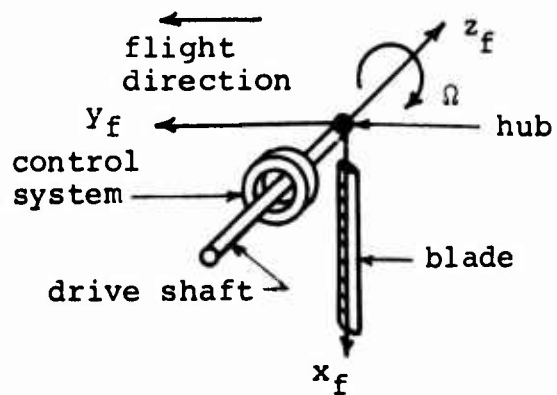
II. Tail Rotor on Port Side - Class A



V. Tail Rotor on Port Side - Class B



III. Tail Rotor on Starboard Side - Class B



VI. Tail Rotor on Starboard Side - Class A

Figure 6. Six Possible Rotor System Configurations.

having the drive shaft and control system in the $-z_f$ direction are denoted as class A arrangements (arrangements I, II, and VI) and rotor system arrangements having the drive shaft and control system in the z_f direction are denoted as class B arrangements (arrangements III, IV and V).

As originally developed, the HASTA program was based on a representational analysis for class A type arrangements. In this form, the analysis was intended to be used for a rotor rotating counterclockwise when viewed from the side of the disk opposite the drive shaft. However, since a clockwise rotating rotor can be viewed from a position and orientation in which the rotational direction appears counterclockwise, the HASTA program is applicable to clockwise rotating rotors, providing proper coordinate systems are used in defining program input variables. For class B type arrangements (arrangements III, IV, and V), the HASTA program is applicable if extra care is taken in representing the rotor hub-drive shaft interface conditions and if the signs of input parameters associated with the effects of the elastic support structure motions on the pitch control structure for flexstrap and bearingless type rotors are changed. Both of these requirements are discussed in detail below. Regardless of whether a counterclockwise or clockwise rotating rotor system is considered, the rotor thrust is always positive in the fixed hub coordinate system z_f direction and the control system (swashplate) displacement is always positive in the $-z_f$ direction. The coordinate systems for various system model components will now be discussed in detail.

ROTOR HUB COORDINATE SYSTEMS

The coordinate systems associated with the rotor hub consist of a fixed (non-rotating) hub coordinate system (used in Figure 6) and a rotating hub coordinate system corresponding to each blade of the rotor. The fixed hub coordinate system has its origin at the intersection of the rotor drive shaft centerline and the rotor disk plane, which is perpendicular to the shaft centerline.

The rotor disk plane is the plane in which the rotor blades rotate at drive shaft rotational speed if the rotor blades are considered to be unconed, unswept, and unpitched (no pitch due to either pitch control or twist). The fixed hub coordinate system z-axis coincides with the rotor drive shaft centerline and is positive in the direction of the rotor

rotational velocity vector. The x and y axis of the fixed hub coordinate system both lie in the rotor disk plane. The x -axis is defined to be located on the advancing side of the rotor disk plane and oriented in space perpendicular to the helicopter longitudinal axis. The azimuthal location of the fixed hub coordinate system x -axis coincides with the reference blade position used for defining the azimuthal location of all blades of the rotor. The y -axis is mutually orthogonal to the x and z axes producing a righthanded Cartesian coordinate system. The location of the fixed hub coordinate system axes; x_f , y_f , and z_f , was shown in Figure 6 for the different rotor system arrangements depicted.

To provide further clarification, the fixed hub coordinate system (x_f , y_f , z_f) is depicted in general in Figure 7 for the two directions of rotor rotation. The y_f -axis has not been shown parallel to the flight direction nor the z_f -axis

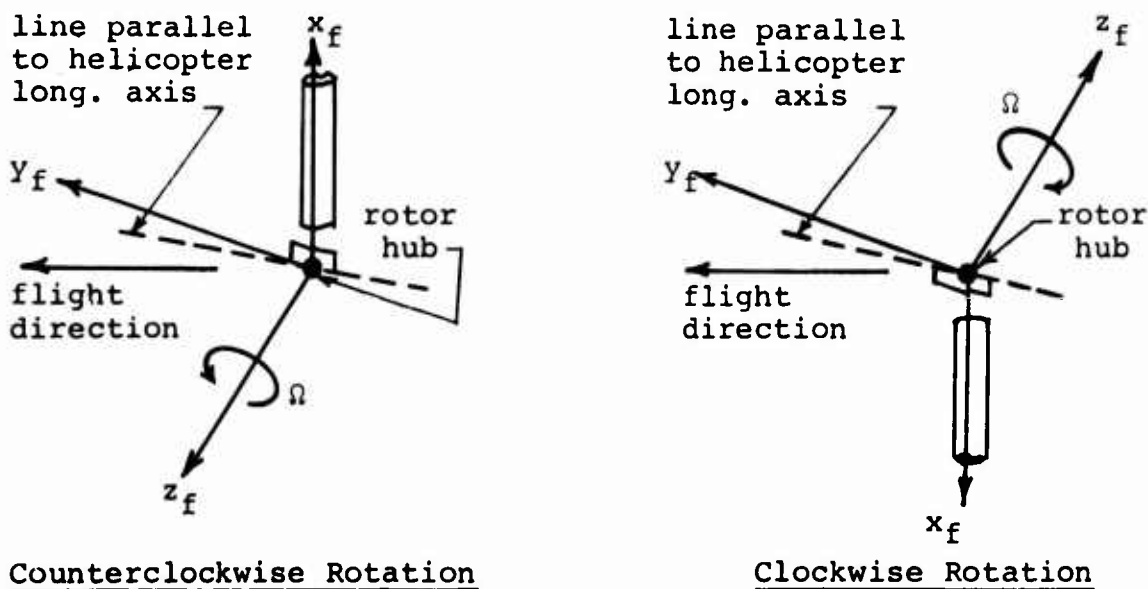


Figure 7. Fixed Hub Coordinate Systems for Two Directions of Rotor Rotation.

perpendicular to the flight direction since the rotor disk plane may be skewed with respect to the flight velocity vector. For an unconed, unpitched, and unswept blade at the reference blade position, as depicted in Figure 7, the y_f -axis is parallel to the blade chord and positive toward the blade leading edge. Also in Figure 7, a line parallel to the helicopter longitudinal axis has been drawn through the fixed hub coordinate system origin for the two directions of rotation. The x_f -axis, as previously defined, is perpendicular to this line. The y_f -axis, when viewed from a position on the z_f -axis, would appear to be coincident to this line.

The rotating hub coordinate systems, one corresponding to each blade, have the same origin and z-axes as the fixed shaft coordinate system, but their x- and y-related axes in the basic rotor plane are rotating about the z_f -axis with a constant rotational speed of Ω rad/sec. The rotating hub coordinate systems have their x-axis along the spanwise axis of the blades if the blades are unconed, unpitched, and unswept.

These coordinate systems can be related to the fixed hub coordinate system by the coordinate transformation

$$\begin{Bmatrix} \bar{i}_{rm} \\ \bar{j}_{rm} \\ \bar{k}_{rm} \end{Bmatrix} = \begin{bmatrix} \cos(\Omega t + \phi_m) & \sin(\Omega t + \phi_m) & 0 \\ -\sin(\Omega t + \phi_m) & \cos(\Omega t + \phi_m) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} \bar{i}_f \\ \bar{j}_f \\ \bar{k}_f \end{Bmatrix} \quad (1)$$

where t is time (sec) and ϕ_m is the angle between m th blade and the reference blade position at $t = 0$. The \bar{i} , \bar{j} , and \bar{k} variables are unit vectors, with rm subscripts denoting the rotating hub coordinate system for the m th blade and f subscripts denoting the fixed hub coordinate system. The rotating hub coordinate system (x_{rm} , y_{rm} , z_{rm}) for the first blade relative to the fixed hub coordinate system is depicted in Figure 8 for the two directions of rotor rotation.

Counterclockwise Rotation

Clockwise Rotation

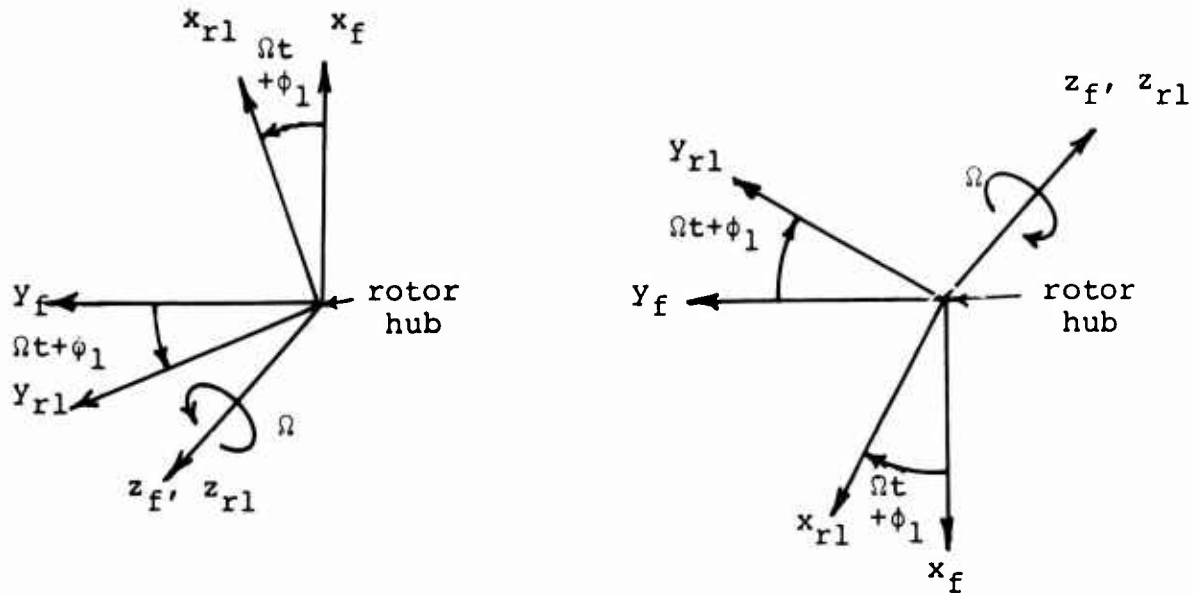


Figure 8. Fixed Hub Coordinate System and Rotating Hub Coordinate System Corresponding to the First Blade for Two Directions of Rotor Rotation.

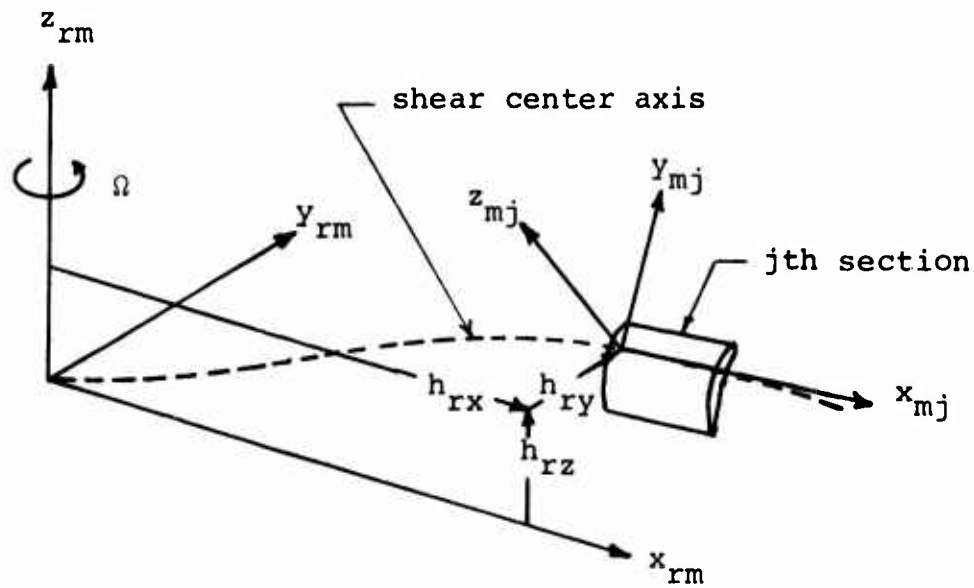
BLADE COORDINATE SYSTEMS

In addition to the hub coordinate systems (rotating and non-rotating) previously discussed, there are blade coordinate systems defined which account for the mean orientation and deflection of the blade structure along its span.

The fundamental coordinate systems associated with the blade structure consist of the rotating local blade coordinate systems, one for each blade section. The local blade coordinate systems are attached to the blade sections in their mean deformed position and rotate about the fixed hub coordinate system z-axis with the same rotational velocity as the rotating hub coordinate systems. Thus, the orientation and location of the local blade coordinate systems are defined relative to the associated rotating hub coordinate system. In the sectional modeling procedure for the lumped-parameter representation of a blade, the blade is subdivided into a number of straight sections that may be oriented relative to each other. One of these blade sections for a deformed blade

rotating either counterclockwise or clockwise is depicted in Figure 9 relative to its associated rotating hub coordinate system and with its local blade coordinate system shown. As can be noted in Figure 9, the origin of the local blade coordinate system for a blade section is located on the section shear center axis at the inboard end of the blade section.

Counterclockwise Rotation



Clockwise Rotation

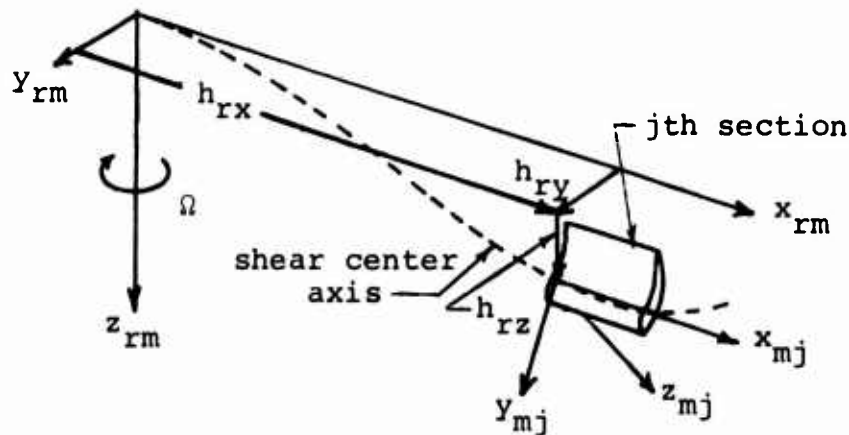


Figure 9. General Description of the Rotating Local Blade Coordinate System for a jth Blade Section.

The location of the origin of the local blade coordinate system for a blade section is defined relative to the associated rotating hub coordinate system axis by specification of the h_{rx} , h_{ry} , and h_{rz} values for the blade section. These three parameters, depicted in Figure 9, correspond to the three components of a position vector and are positive for a translation in the direction of the x_{rm} , y_{rm} , and z_{rm} axes, respectively. The x-axis of a local blade coordinate system is coincident with the blade section shear center axis and is positive in the outboard direction. The y-axis is perpendicular to the x-axis and parallel to the chordline of the airfoil section at the coordinate system origin and is positive toward the blade leading edge. The z-axis is perpendicular to both the x and y axes such as to produce a right-handed coordinate system. As depicted in Figure 9 by use of an airfoil shape having camber, the positive direction of the z-axis of a local blade coordinate system is toward the airfoil surface, normally considered to be the upper surface for a positive angle of attack.

The orientation of the local blade coordinate system (x_{mj} , y_{mj} , z_{mj}) for a blade section is defined relative to the associated rotating hub coordinate system (x_{rm} , y_{rm} , z_{rm}) by specification of the three angles; ϕ , θ , and ψ . These three angles, in the order listed, correspond to three consecutive rotations that must be applied to the rotating hub coordinate system (placed at the local blade coordinate system origin) to align it with the local blade coordinate system. In particular:

- ϕ corresponds to local blade section presweep about the z_{rm} -axis and is positive in the direction that the rotor rotates,
- θ corresponds to local blade section precone about the preswept y_{rm} -axis and is positive if the outboard end of the section is rotated in the $-z_{rm}$ direction, and
- ψ corresponds to local blade section prepitch about the preswept and preconed x_{rm} -axis and is positive if the leading edge of the blade section is rotated in the preconed z_{rm} direction.

The consecutive application of these angles to obtain the orientation of a local blade coordinate system relative to the associated rotating hub coordinate system orientation is depicted in Figure 10 for a counterclockwise and a clockwise rotating rotor system.

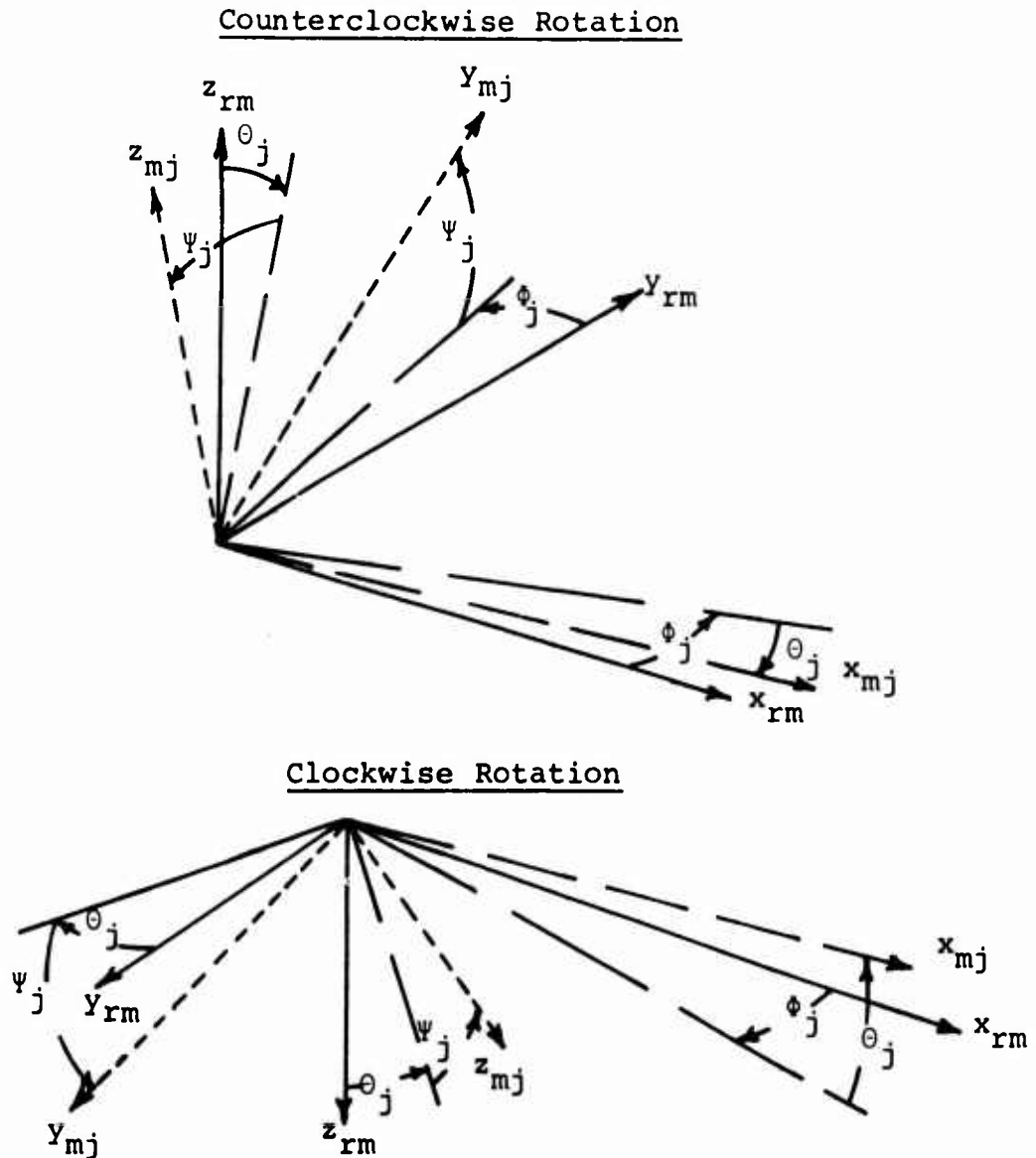


Figure 10. Description of the Rotating Hub Coordinate System Rotations Required for Definition of the j th Section Local Blade Coordinate System Orientation.

The orientation relationship between the rotating hub coordinate system and a local blade coordinate system is represented by the coordinate transformation

$$\begin{Bmatrix} \bar{i}_{mj} \\ \bar{j}_{mj} \\ \bar{k}_{mj} \end{Bmatrix} = \begin{bmatrix} C\phi C\theta & S\phi C\theta & -S\theta \\ -S\phi C\psi + C\phi S\theta S\psi & C\phi C\psi + S\phi S\theta S\psi & C\theta S\psi \\ S\phi S\psi + C\phi S\theta C\psi & -C\phi S\psi + S\phi S\theta C\psi & C\theta C\psi \end{bmatrix} \begin{Bmatrix} \bar{i}_{rm} \\ \bar{j}_{rm} \\ \bar{k}_{rm} \end{Bmatrix} \quad (2)$$

where a short form of denoting the sine and cosine functions has been utilized (e.g., $S\phi$ represents $\sin \phi_j$ and $C\phi$ represents $\cos \phi_j$). The \bar{i} , \bar{j} , and \bar{k} variables are unit vectors, with rm subscripts denoting the rotating hub coordinate system associated with the m th blade and mj subscripts denoting the local blade coordinate system for the j th section of the m th blade.

The local blade coordinate systems are the coordinate systems to which the blade slopes, deflections, forces and moments along the blade span are referenced. In particular, the blade state variables at the inboard end of the j th blade section act on the outboard end of the next inboard blade section in the positive axis directions of the j th section local blade coordinate system. These state variables are defined as follows:

- u_x, u_y, u_z are the blade deflections in the local blade coordinate system $x, y,$ and z directions, respectively,
- N, V_y, V_z are the axial, edgewise shear, and flapwise shear forces acting in the local blade coordinate system $x, y,$ and z directions, respectively,
- ϕ_x, ϕ_y, ϕ_z are the torsional, flapwise, and edgewise bending slopes about the local blade coordinate system $x, y,$ and z directions, respectively, and
- T, M_y, M_z are the torsional, flapwise bending, and edgewise bending moments about the local blade coordinate system $x, y,$ and z directions, respectively.

Since the blade state variables along the blade are defined relative to the local blade coordinate system, the blade state variables can also be defined relative to the rotating hub coordinate system by use of the coordinate transformation relationship specified in Equation (2).

The ϕ and θ values for a blade section having an elastic length but not shear center axis rigid offsets can be determined by consideration of the changes occurring in the h_{rx} , h_{ry} , and h_{rz} sectional parameters in going from the blade section of interest to the next outboard section, since the section shear center axis is a straight line. In particular, for the j th blade section

$$\phi_j = \arctan (\Delta h_{ry} / \Delta h_{rx}) \quad (3)$$

$$\theta_j = \arctan (-\Delta h_{rz} / \sqrt{\Delta h_{rx}^2 + \Delta h_{ry}^2}) \quad (4)$$

where

$$\Delta h_{rx} = h_{rx}^{j-1} - h_{rx}^j,$$

$$\Delta h_{ry} = h_{ry}^{j-1} - h_{ry}^j,$$

$$\Delta h_{rz} = h_{rz}^{j-1} - h_{rz}^j,$$

and the $j-1$ superscript denotes the blade section just outboard of the j th section. If shear center axis rigid offsets are involved in the blade section, the h_{rx} , h_{ry} , and h_{rz} parameters corresponding to the next outboard section in the previous expressions would be replaced by those corresponding to the location at which the rigid offsets occur on the shear center axis segment which passes through the local coordinate system origin. In the representation of the blade sectional characteristics (Figure 2), the localized change in the coordinate system orientation from that of the previous outboard section (local bends) can only be considered inboard of the section shear center axis rigid offset characteristics. Thus, the section shear center axis rigid offsets always are along the axis directions of the local coordinate system of the next outboard blade section. The use of the method described above for determining the blade section ϕ and θ values is particularly advantageous when blades having mean deformation due to steady loading effects in addition to built-in geometric angles are being considered.

The local blade aerodynamic coordinate system for a blade section having aerodynamic characteristic is oriented such that the sectional lift and drag forces act along its z and y axes, respectively. The origin and x -axis of the local blade aerodynamic coordinate system for a blade section coincide with those of the associated local blade coordinate system. The orientation of the local blade aerodynamic coordinate system is obtained by rotating the local coordinate system about its x -axis. The relationship between the local blade aerodynamic and local blade coordinate systems of a blade section and several aerodynamic-related parameters are depicted in Figure 11 for the two directions of rotor rotation, as viewed from a position on the positive x -axes of the coordinate systems.

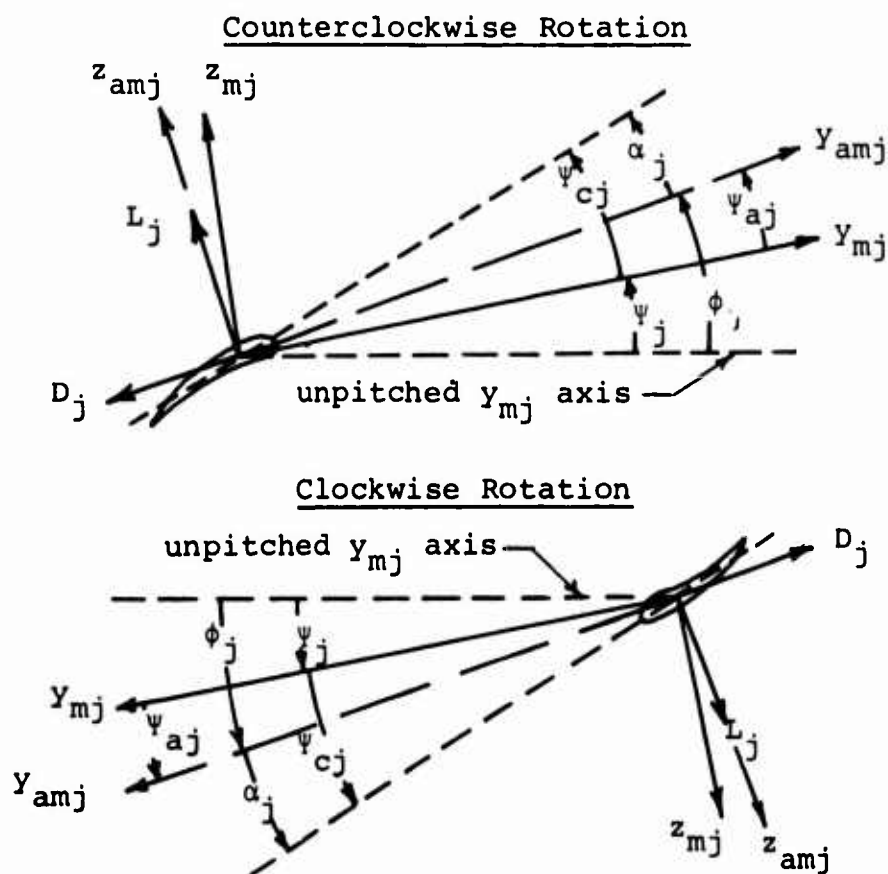


Figure 11. Relationship Between the Local Blade Aerodynamic and Local Blade Coordinate Systems for the j th Blade Section.

In Figure 11, Ψ_j is the previously discussed blade section prepitch which includes collective pitch, Ψ_{cj} is the blade cyclic pitch, ϕ_j is the section inflow angle, α_j is the section aerodynamic angle of attack, Ψ_{aj} is the angle of rotation between the local aerodynamic and local coordinate systems, and L_j and D_j are the section lift and drag forces, respectively. Since these parameters, except Ψ_j , are dependent on the blade azimuthal position in forward flight the relationship between these two blade section coordinate systems may vary azimuthally. This relationship is represented by the coordinate transformation

$$\begin{Bmatrix} \bar{i}_{amj} \\ \bar{j}_{amj} \\ \bar{k}_{amj} \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\Psi_a & S\Psi_a \\ 0 & -S\Psi_a & C\Psi_a \end{bmatrix} \begin{Bmatrix} \bar{i}_{mj} \\ \bar{j}_{mj} \\ \bar{k}_{mj} \end{Bmatrix} \quad (5)$$

where $S\Psi_a$ represents $\sin\Psi_{aj}$, $C\Psi_a$ represents $\cos\Psi_{aj}$, and the \bar{i} , \bar{j} , and \bar{k} variables are unit vectors, with amj subscripts denoting the local aerodynamic coordinate system and mj subscripts denoting the local coordinate system of the j th section of the m th blade. As can be noted in Figure 11, the rotation angle Ψ_{aj} is defined by $\Psi_{aj} = \phi_j - \Psi_j$.

Regardless of whether a counterclockwise or clockwise rotating rotor is being considered, the lift force generated by the rotor is positive if it acts on the rotor hub in the positive fixed hub coordinate system z -axis direction. This definition is consistent with the direction of the positive section lift force resulting from a positive blade section angle of attack, as depicted in Figure 11. For some rotor systems (for example, the rotor system depicted in Figure 6 as Arrangement IV) a negative lift force as defined relative to the rotor's fixed hub coordinate system is desirable. For a symmetrical airfoil this can be achieved by defining the Ψ_j , Ψ_{cj} , and ϕ_j blade section parameters such that the α_j (blade section aerodynamic angle of attack) parameters would be for the most part negative in value. In particular, the Ψ_j and Ψ_{cj} blade parameters would have values opposite in

sign to that required for positive rotor lift. In that the ϕ_j parameter is dependent on the magnitude and orientation of the hub flight velocity and the induced velocity field, the value of ϕ_j need not be negative, but the induced velocity should be negative in value (defined as positive acting on the rotor disk in the $-z_{rm}$ axis direction).

The above comments for a symmetrical airfoil do not suffice for a cambered airfoil producing negative rotor lift. In addition, the airfoil coefficient data must be defined such that the cambered airfoil represented has a profile which is inverted compared to the cambered airfoil profile depicted in Figure 11. This is necessary since a cambered airfoil having its lower surface on the z_{mj} -axis side of the airfoil chord-line would normally be used to generate rotor lift in the $-z_{rm}$ direction. The lift, drag and moment coefficients as a function of angle of attack, α , and Mach number, M , necessary to construct the airfoil coefficient data deck for this inverted cambered airfoil profile (inverted relative to the local aerodynamic coordinate system) can be obtained by use of the expressions:

$$\begin{aligned}
 C_L^I(\alpha, M) &= -C_L^N(-\alpha, M) \\
 C_D^I(\alpha, M) &= C_D^N(-\alpha, M) \\
 C_M^I(\alpha, M) &= C_M^N(-\alpha, M)
 \end{aligned}
 \tag{6}$$

In these expressions C_L , C_D , and C_M correspond to the airfoil lift, drag, and moment coefficients, respectively, with the I superscripts denoting the coefficients for the inverted airfoil profile and the N superscripts denoting the coefficients as normally defined. As an example, the inverted cambered airfoil lift coefficient for an angle of attack of -6 degrees and a given Mach number would have the same magnitude as but be opposite in sign to the standard cambered airfoil lift coefficient for an angle of attack of 6 degrees and the same Mach number. The airfoil coefficient data for a symmetrical airfoil inherently satisfies the expressions of Equation (6).

PITCH CONTROL STRUCTURE COORDINATE SYSTEMS

The pitch control structure, as previously defined, consists

of the structure connecting the control rod to the blade. The HASTA analysis can represent three different types of pitch control structure, the use of which is dependent on the type of rotor being considered. These are:

1. an arbitrarily oriented rigid pitch arm which applies only torque about the blade shear center axis at the pitch arm-blade attachment point (articulated teetering, or gimbaled rotor),
2. an arbitrarily oriented flexible pitch arm which applies forces and moments in three directions to the blade (flexstrap rotor), and
3. a torque tube structure which applies forces and moments in three directions to the blade and consists of a flexible torque tube and spur, both of which are allowed the same arbitrary orientation, and an arbitrarily oriented rigid pitch arm (bearingless rotor);

all of which have been depicted in Figure 4. The coordinate systems associated with the different pitch control structures consist of those necessary to define the orientation of the pitch arm, torque tube, and spur components and the axes about which the stiffness characteristics of a component are taken to act.

The articulated rotor type of pitch control structure, consisting of a rigid pitch arm of length L_{pa} , has the origin of its associated coordinate system located at the pitch arm-blade attachment point. The x-axis of the pitch arm coordinate system coincides with the lengthwise axis of the pitch arm and is positive in the direction toward the pitch arm-control rod attachment point. The y and z axes of the pitch arm coordinate system are both perpendicular to the pitch arm coordinate system x-axis as well as perpendicular to each other. The azimuthal orientation of these two axes relative to the pitch arm coordinate system x-axis is arbitrary. However, the positive directions of the x and y axes must be chosen such that a right-handed Cartesian coordinate system is produced.

The orientation of the pitch arm coordinate system (x_p, y_p, z_p) is defined relative to the associated rotating hub coordinate system (x_{rm}, y_{rm}, z_{rm}) by specification of the three angles, α , β , and γ . These angles, in the order listed, correspond to three consecutive rotations, which must be applied to the rotor

hub coordinate system (placed at the pitch arm-blade attachment point) to align it with the pitch arm coordinate system. The application of the α , β , and γ angles is conceptually identical to that of the ϕ , θ , and ψ angles used to define the orientation of a local blade coordinate system. The use of these angular rotations in defining the orientation of the pitch arm coordinate system relative to the rotating hub coordinate system is depicted in Figure 12 for both directions of rotor rotation.

The relationship between the pitch arm coordinate system (x_{pm}, y_{pm}, z_{pm}) is represented by the coordinate transformation

$$\begin{Bmatrix} \bar{i}_p \\ \bar{j}_p \\ \bar{k}_p \end{Bmatrix} = \begin{bmatrix} C\alpha C\beta & S\alpha C\beta & -S\beta \\ -S\alpha C\gamma + C\alpha S\beta S\gamma & C\alpha C\gamma + S\alpha S\beta S\gamma & C\beta S\gamma \\ S\alpha S\gamma + C\alpha S\beta C\gamma & -C\alpha S\gamma + S\alpha S\beta C\gamma & C\beta C\gamma \end{bmatrix} \begin{Bmatrix} \bar{i}_{rm} \\ j_{rm} \\ \bar{k}_{rm} \end{Bmatrix} \quad (7)$$

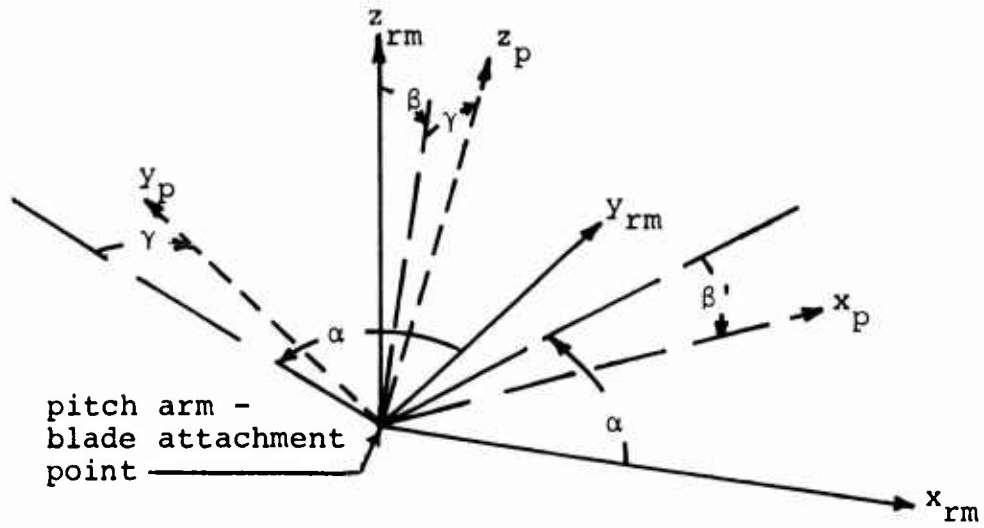
where a short form of denoting the sine and cosine functions has been used and the \bar{i} , \bar{j} , and \bar{k} variables are unit vectors, with the subscripts denoting the coordinate system to which unit vectors correspond.

This rigid pitch arm pitch control structure is only allowed to apply torque to the blade about the local blade shear center axis (x_{mj} -axis) at the pitch arm-blade attachment point. This torque can be defined in terms of the control rod force applied to the pitch arm and an effective moment arm. The value of this effective moment arm is, in fact, the only data required by HASTA to represent this pitch control structure used with articulated, gimballed, or teetering rotors, since the control rod applied force in this case is considered to act in a direction parallel to the rotating hub coordinate system z-axis. This effective moment arm as required by the HASTA program is defined by

$$a = -L_p (S\alpha C\phi - C\alpha S\phi) C\beta C\theta, \quad (8)$$

the terms on the right hand side having been previously defined. As can be noted from Equation (8) for a blade without presweep, precone, and prepitch at the pitch arm-blade attachment point, a is negative in value if the pitch arm is directed toward the blade leading edge ($0^\circ < \alpha < 180^\circ$) and positive if the pitch arm is directed aft ($180^\circ < \alpha < 360^\circ$).

Counterclockwise Rotation



Clockwise Rotation

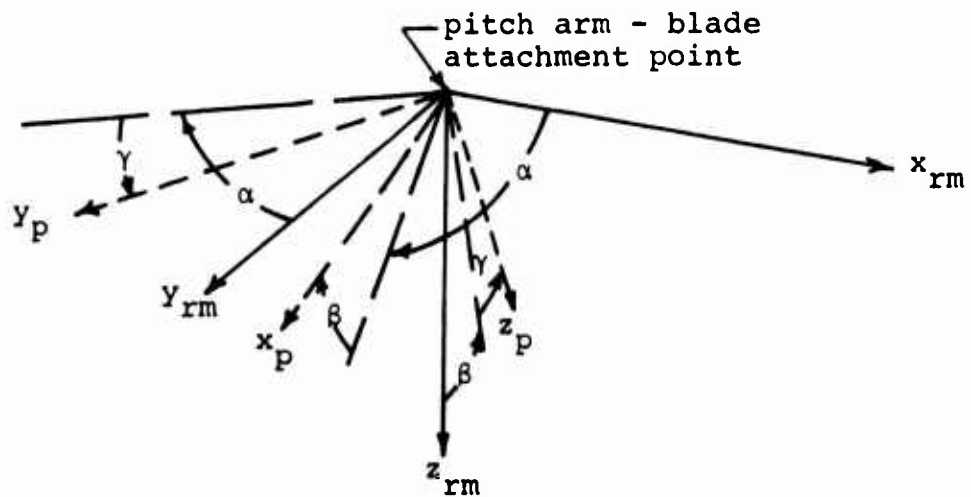


Figure 12. Description of Rotating Hub Coordinate System Rotations Required for Definition of the Pitch Arm Coordinate System Orientation.

The pitch arm of the flexstrap pitch control structure differs from that discussed above in that the pitch arm is allowed bending and axial flexibility and applies restraining forces and moments to the blade in three mutually orthogonal directions. The definition of the flexstrap pitch arm coordinate system orientation is identical to that presented above for the rigid pitch arm. Thus, the orientation angles (α , β , and γ) and the coordinate system relationships depicted in Figure 12 and represented by the coordinate transformation of Equation (7) are directly applicable to the flexstrap pitch arm. Since the flexstrap pitch arm representation requires bending stiffness information about the pitch arm coordinate system y and z axes (EI_{yy} and EI_{zz} , respectively), the azimuthal orientation of these two axes relative to the pitch arm coordinate system x -axis should be based on the axes about which these stiffness parameters are known. The data required by HASTA to represent the flexstrap pitch arm, in addition to the bending stiffnesses, consists of the pitch arm length L_p , the axial stiffness EA_c , and the three orientation angles α , β , and γ . The flexstrap pitch arm representation requires also that the shear center axis at the pitch arm-blade attachment point be parallel to the x_{rm} -axis (i.e., unswept, unconed, and un pitched). The positional parameters h_{ry} and h_{rz} for the pitch arm-blade attachment point need not be zero in value and, in fact, would not normally be zero due to the presweep, precone, and prepitch distributions occurring inboard of the attachment point.

The torque tube pitch control structure coordinate systems consist of the torque tube, spur, and pitch arm coordinate systems. The torque tube of length L_t has the origin of its associated coordinate system located on the blade shear center axis at the torque tube-blade attachment point. The x -axis of the torque tube coordinate system coincides with the lengthwise axis of the torque tube and is positive in the direction toward the torque tube-spur-pitch arm attachment point. The y and z axes of the torque tube coordinate system are defined in the same manner as the y and z axes of the flexstrap pitch arm coordinate system. The definition of the torque tube coordinate system orientation relative to the rotating hub coordinate system is identical to the definition of the rigid pitch arm orientation. Thus, as was also noted for the flexstrap arm, the α , β , and γ orientation angles and the coordinate system relationships depicted in Figure 12 and represented by Equation (7) are directly applicable to the torque tube representation. For the torque tube coordinate system, the value

of α in degrees should be close to 180 degrees since the x-axis of the torque tube should be primarily in the $-x_{rm}$ direction. The data required by the HASTA program to represent the torque tube portion of the torque tube pitch control structure consists of data similar to that required for the flexstrap pitch arm and the torsional stiffness of the torque tube. In addition, the blade structure must be modeled such that the blade shear center axis at the torque tube-blade attachment point is unswept, unconed, and unpitched regardless of the location of the attachment point. This condition is identical to that required for the flexstrap pitch arm.

The spur (extension shaft) of length L_s has the origin of its associated coordinate system located at the torque tube-spur-pitch arm attachment point. The orientation of the spur coordinate system is identical to the orientation of the torque tube coordinate system and as such is defined by the α , β , and γ orientation angles used in the representation of the torque tube. Thus, the numerical forms of the coordinate transformation matrix presented in Equation (7) for the torque tube and the spur are always identical. Since the spur is attached to the hub structure by a spherical bearing such that the torque and axial force cannot be applied to the spur and the orientation angles are provided via torque tube data, data required by HASTA for the representation of the spur consists of its length and bending stiffnesses. If the rotor hub is allowed rotational and translational degrees of freedom due to its support structure, an additional length parameter is required to define the transverse motions of the spherical bearing relative to the rotating hub coordinate system z_{rm} -axis. This length, L_b , is the perpendicular distance from the rotor disk plane (x_{rm} - y_{rm} plane) to the spherical bearing (i.e., measured along a line parallel to the z_{rm} -axis). This length is positive in value if the spherical bearing is in the $-z_{rm}$ direction from the rotor disk plane regardless of the direction of rotor rotation. For example, if the spherical bearing is on the same side of the rotor disk plane as the control system, the length L_b is positive for the Class A rotors and negative for the Class B rotors depicted in Figure 6.

The torque tube pitch control structure pitch arm of length L_p has the origin of its associated coordinate system also located at the torque tube-spur-pitch arm attachment point. The

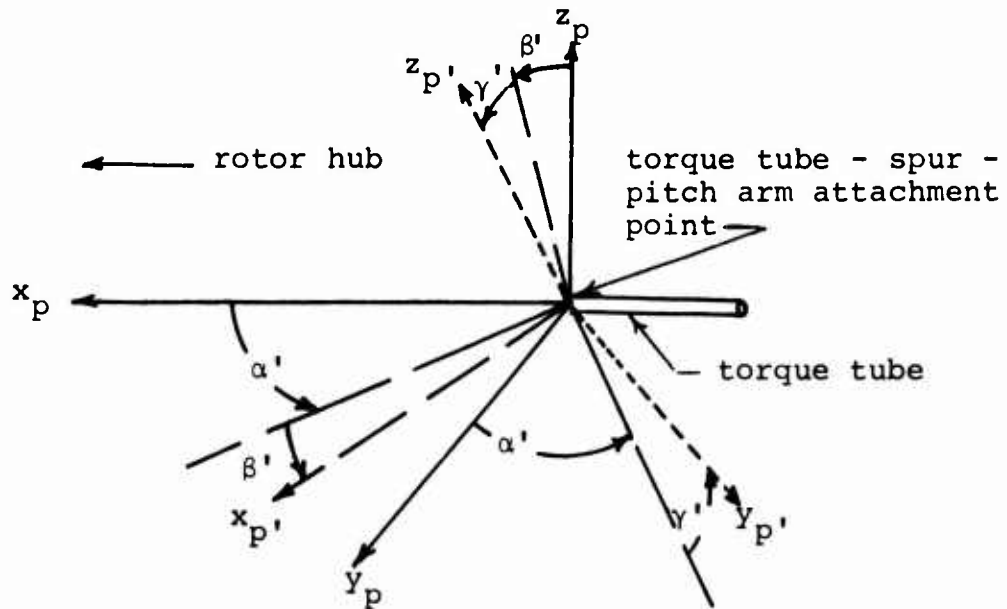
axes of this rigid pitch arm coordinate system are located relative to the pitch arm structure in the same manner as for the rigid pitch arm used with articulated rotor blades. The orientation of this rigid pitch arm coordinate system, however, is not defined relative to the rotating hub coordinate system but rather to the torque tube coordinate system. This orientation is specified by the three angles α' , β' , and γ' which are applied conceptually in the same manner as α , β , and γ . The use of these three angular rotations in defining the orientation of this rigid pitch arm coordinate system relative to the torque tube coordinate system (placed at the torque tube-spur-pitch arm attachment point) is depicted in Figure 13 for the two directions of rotor rotation. In this figure, the torque tube orientation angles α , β , and γ have been assumed to have values (in degrees) of 180, 0, and 0, respectively.

The relationship between this rigid pitch arm coordinate system ($x_{p'}$, $y_{p'}$, $z_{p'}$) and the torque tube coordinate system (x_p , y_p , z_p) is represented by Equation (7) with the p and rm subscripts replaced by p' and p , respectively, and α , β , and γ replaced by α' , β' , and γ' , respectively. Since the pitch arm is rigid, only the correct orientation of the pitch arm coordinate system x-axis is required. This orientation can be achieved by use of only α and β angular rotations. The data required by the HASTA program for the representation of the pitch arm component of the torque tube pitch control structure consists of the pitch arm length and orientation parameters.

CONTROL SYSTEM COORDINATE SYSTEMS

The fundamental coordinate systems required for the representation of the control system consist of a fixed coordinate system and a rotating coordinate system corresponding to each rotor blade. The origin of the fixed control system coordinate system is located on the rotor drive shaft axis (same as the z_f -axis of the fixed hub coordinate system) where the plane of the control system ring intersects it. The x and y axes of this coordinate system lie in the unperturbed plane of the control system ring which is also parallel to the rotor disk plane. The x-axis is defined to be along the line from the coordinate system origin to the location on the control ring at which the control rod of the first blade is attached when this blade is in the reference blade position (i.e., the position at which the spanwise axis of the unswept, uncone, and unpitched blade coincides with the x_f -axis). The orientation and positive direction of the y-axis is defined as required to produce a right-handed Cartesian coordinate system.

Counterclockwise Rotation



Clockwise Rotation

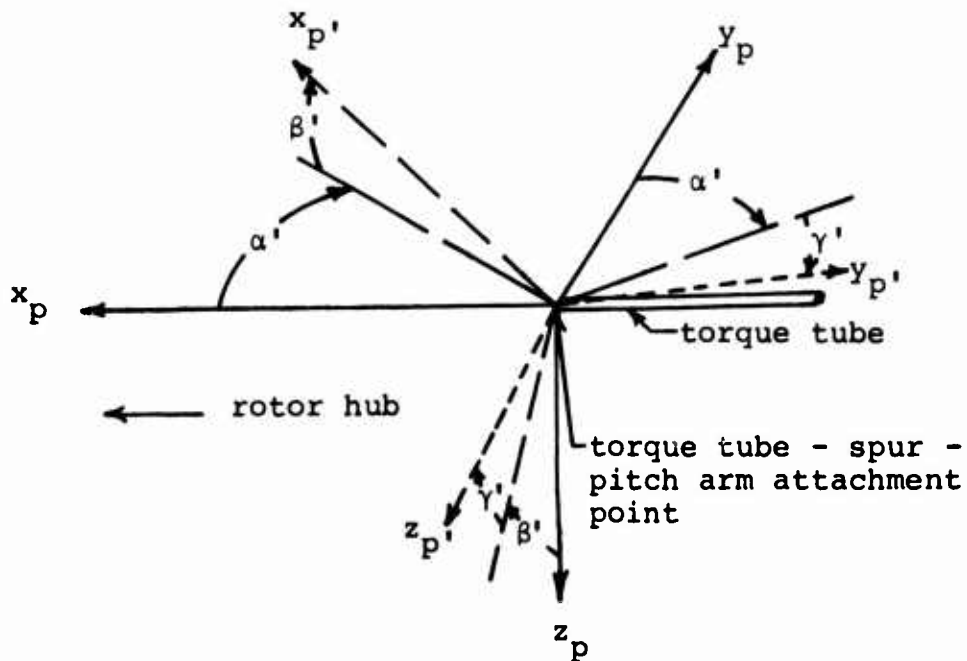


Figure 13. Description of Torque Tube Coordinate System Rotations Required for Definition of the Pitch Arm Coordinate System.

The orientation of the fixed control system coordinate system (x_{fc}, y_{fc}, z_{fc}) relative to the fixed hub coordinate system (x_f, y_f, z_f) is depicted in Figure 14 for both directions of rotor rotation.

Counterclockwise Rotation

Clockwise Rotation

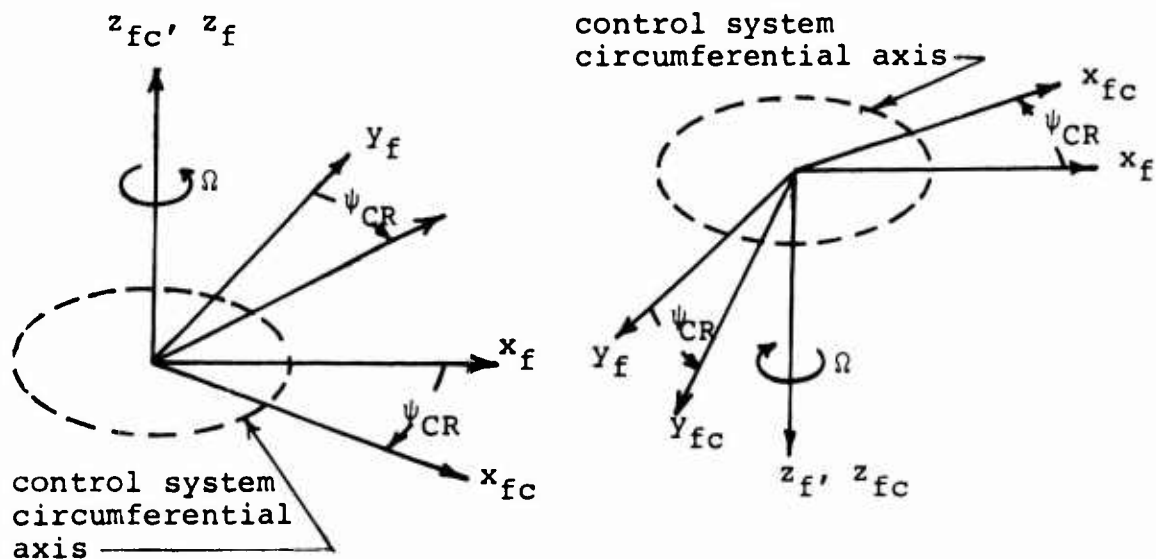


Figure 14. Description of the Relationship Between the Fixed Hub Coordinate System and the Fixed Control System Coordinate System.

The angle ψ_{CR} denotes the degree of azimuthal shift occurring between the two fixed coordinate systems that results from the control rods being attached to the control system ring at locations not in their associated blade-related rotating hub coordinate system x - z planes. The value of ψ_{CR} is not used by the HASTA program. Instead, the azimuthal location (χ_j) of the cyclic spring-damper units depicted in Figure 5 are defined relative to the fixed control system coordinate system. In addition, the control system displacement results obtained from HASTA application must be interpreted relative to the fixed and rotating coordinate systems associated with the control system and not those associated with the rotor hub.

The rotating control system coordinate systems, one for each blade but associated with the azimuthal location of the related control rod-control system ring attachment point, have the same origin and z-axes as the fixed control system coordinate system. Their x- and y-related axes are in the same plane as the x- and y-axes of the fixed coordinate system but are rotating about the z_{fc} -axis with a constant rotational speed of Ω rad/sec. The relationship between the rotating control system coordinate system (x_{mc}, y_{mc}, z_{mc}) associated with the first blade and the fixed control system coordinate system is depicted in Figure 15. As can be noted in this figure, the angular relationship includes ϕ_1 . This is

Counterclockwise Rotation

Clockwise Rotation

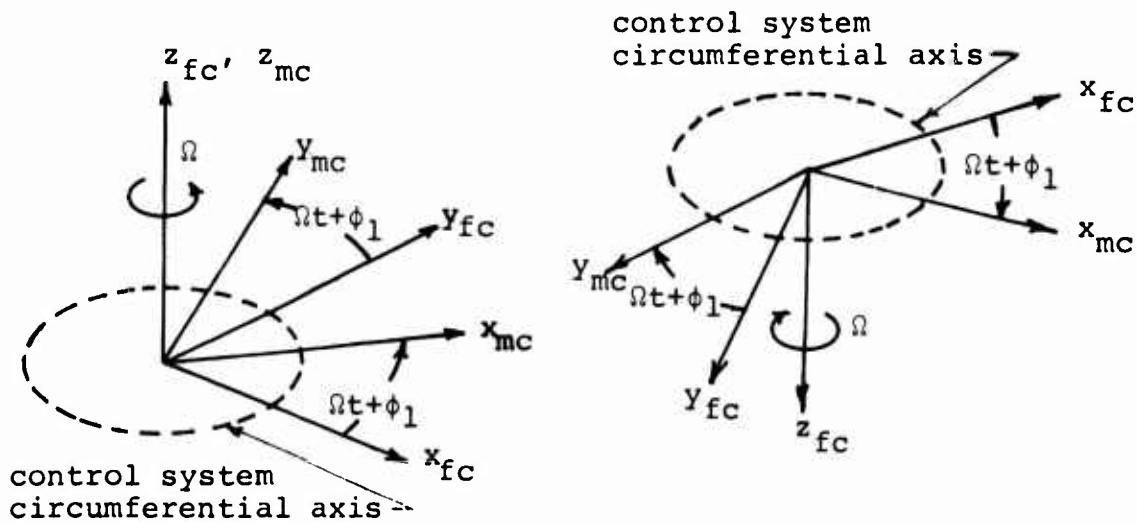


Figure 15. Description of the Relationship Between the Rotating Control System Coordinate System Associated with First Blade and the Fixed Control System Coordinate System.

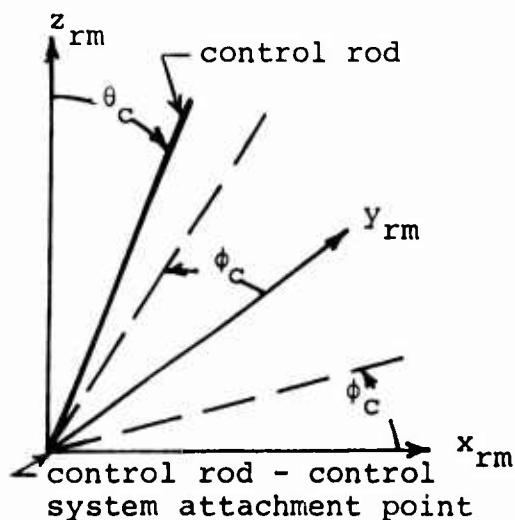
necessary since ϕ_1 , if non-zero, shifts not only the azimuthal location of the first blade but also shifts its control rod-control system attachment point. Due to the azimuthal shift occurring between the fixed control system coordinate system and the fixed hub coordinate system, the orientation of a rotating control system coordinate system will lag by ψ_{CR} that of the corresponding rotating hub

coordinate system. The relationship between the rotating and fixed control system coordinate systems is defined by Equation (1) with the rm and f subscripts replaced by mc and fc , respectively. The above discussion is valid for both Class A and Class B control systems.

CONTROL ROD ORIENTATION

The control rod representation does not require coordinate system definition per se. Only the orientation of the control rod lengthwise axis need be defined, since the control rods are only subjected to axial loading. There are two types of control rod representation, the use of which is dependent on the type of rotor being considered. The first type is that used for the articulated, gimballed, or teetering rotor. In this representation the control rod is assumed to act only in the rotating hub coordinate system z_{rm} -axis direction (i.e., perpendicular to the rotor disk plane), although it actually may or not act only in that direction. The second type is that used for the flexstrap or bearingless rotor. This control rod representation allows for arbitrary orientation of the control rod relative to the rotating hub coordinate system (placed at the control rod-control system attachment point). The orientation of the control rod lengthwise axis is specified by the orientation angles ϕ_c and θ_c , which correspond to the consecutive angular rotations that must be applied to the rotating hub coordinate system to align its z-axis with the control rod axis. The application of the ϕ_c and θ_c angles is identical in concept to that of the ϕ and θ angles used for specification of the local blade coordinate system orientation. The relationship between the control rod axis and the rotating hub coordinate system is depicted in Figure 16 for the two directions of rotor rotation. If a Class B rotor system (i.e., control system located in the $+z_{rm}$ direction relative to the rotor disk plane) is being considered, the value of θ_c will be greater than 90 degrees.

Counterclockwise Rotation



Clockwise Rotation

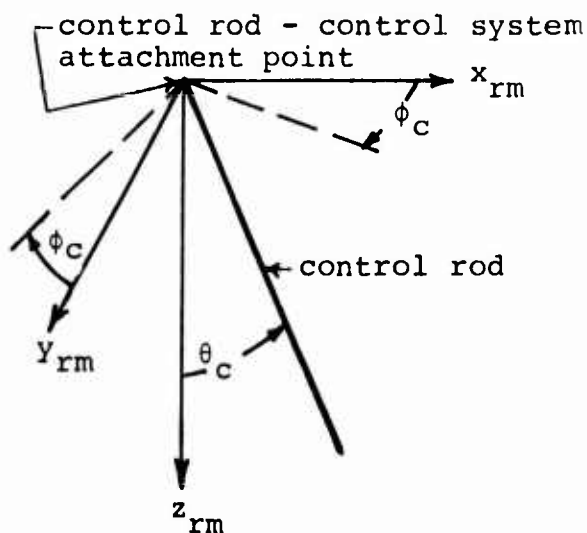


Figure 16. Relationship Between Control Rod Axis and Rotating Hub Coordinate System.

The data required by the HASTA program for the first type of control rod representation, that associated with articulated, gimballed, and teetering rotor systems, consists of the effective control rod stiffness and damping characteristics in the z_{rm} axis direction. The primary data required for the control rod representation associated with a flexstrap or bearingless rotor system consists of the orientation angles, ϕ_c and θ_c , and the control rod axial stiffness and damping characteristics. In the latter case, an additional length parameter, similar in concept to the length L_b , is required if the rotor hub is allowed rotational and translational degrees of freedom due to its support structure. This length parameter is necessary to define the transverse motions of the control rod-control system attachment point relative to the rotating hub coordinate system z_{rm} axis. This length, L_{cc} , is defined as the perpendicular distance from the rotor disk plane (x_{rm} - y_{rm} plane) to the control rod-

control system attachment point. This length is positive in value for Class A rotors and negative for Class B rotors.

SUPPORT STRUCTURE COORDINATE SYSTEMS

The coordinate systems associated with the rotor elastic support structure consist of a fixed coordinate system and local section coordinate systems which are conceptually similar to the rotating hub and local blade coordinate systems, respectively. The primary difference is that the support structure-related coordinate systems are not rotating. The fixed (reference) support structure coordinate system has its origin located on the shear center axis at the tip end of the support structure (the end of the support structure furthest from its attachment to the rotor hub). In that the orientation of the fixed coordinate system axes is in general arbitrary except when gravitational effects on the support structure are included, two different possible fixed coordinate systems are discussed below. These coordinate systems are defined on the basis of the elastic support structure being a helicopter fuselage-tailboom-fin structure but are also valid for any other elastic support structure.

The inclusion of the effects of gravity requires the fixed coordinate system y-axis to be parallel to but opposite in direction to the gravity force (weight) vector acting on the fuselage structure. Thus, the fixed coordinate system y-axis is positive in the general direction toward the top of the helicopter fuselage. The x- and z-axes of this fixed coordinate system must lie in a plane perpendicular to the y-axis at the fixed coordinate system origin. The azimuthal orientation of the x- and z-axes relative to the y-axis is arbitrary providing that the positive directions of these axes are such that a right-handed Cartesian coordinate system is produced. However, for convenience in generating input data, the positive direction of the fixed coordinate system x- and y-axes should be toward the tailboom (aft) and to the port side of the helicopter, respectively.

When gravity effects are not to be included, the orientation and positive direction of the fixed support structure coordinate system axes are arbitrary providing that a right-handed Cartesian coordinate system is produced. However, a fixed support structure coordinate system can be defined that is convenient to the determination of support structure input data. In this fixed coordinate system, the x-axis is defined to be parallel to the forward flight velocity of the helicopter in level forward flight and to be positive directed aft. The y-axis is defined to be perpendicular to the x-axis, to

lie in the vertical plane of the helicopter fuselage, and to be positive toward the top of the helicopter. The z-axis is mutually orthogonal to the x- and y-related axes such as to produce a right-handed coordinate system. The two fixed support structure coordinate systems described above are depicted in Figure 17 as they relate to a fuselage-tailboom-fin support structure. In regard to the fixed coordinate system depicted for the case of inclusion of gravitational effects, the gravity vector and therefore the y-axis have been assumed to be in the vertical plane of the fuselage. Examples of local support structure coordinate systems are also shown in Figure 17.

The local support structure coordinate systems are obtained by application of the orientation angles ϕ , θ , and ψ in the same manner as is used in defining the orientation of the rotating local blade coordinate systems. Thus, the relationship between the local support structure coordinate systems (x_s , y_s , z_s) and the fixed structure coordinate system (x_{fs} , y_{fs} , z_{fs}) is represented by Equation (2) with the m_j and r_m subscripts replaced by s and fs subscripts, respectively. It should be noted that the local coordinate system y-axis for a support structure section is the axis taken as the chordline for aerodynamic purposes.

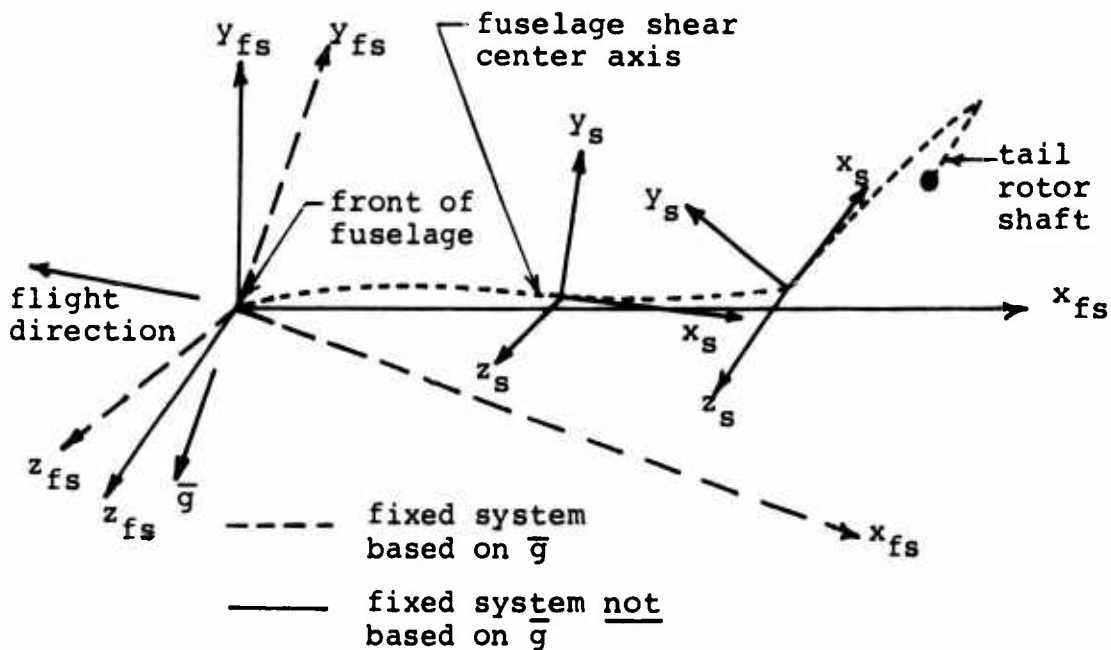


Figure 17. Fixed Fuselage Structure Coordinate System and Examples of Local Fuselage Structure Coordinate Systems.

SHAFT-ROTOR HUB INTERFACE RELATIONSHIPS

The matching of boundary conditions at the interface of the rotor drive shaft and the rotor hub requires a defined relationship between the local support structure coordinate system at the end of the rotor drive shaft and the fixed hub coordinate system. The orientation of the local support structure coordinate system at the end of the rotor drive shaft is accomplished through the use of the ϕ , θ , and ψ angles. In that the blade state variables involved in the boundary condition equations must be defined relative to the rotating hub coordinate system, the most inboard blade section should end at the hub centerline and have no built-in presweep, precone, or prepitch; i.e., ϕ , θ , and ψ should be zero in value.

The relationship between the two coordinate systems used for the matching of boundary conditions is that at the rotor drive shaft-rotor hub interface, in terms of unit vectors

$$\begin{aligned}\bar{i}_s &= \bar{k}_f \\ \bar{j}_s &= \bar{j}_f \\ \bar{k}_s &= -\bar{i}_f\end{aligned}\tag{9}$$

Thus, the required relationship between the support structure coordinate system and the fixed hub coordinate system at this interface is as depicted in Figure 18 for the two directions of rotor rotation. Note: This coordinate system relationship which is dependent upon the direction of rotor rotation, must be satisfied whether the rotor drive shaft is modeled on the positive or negative z_f -axis side of the rotor disk plane.

If this required relationship is not satisfied by the modeling of the support structure, the computer program will yield erroneous results except for the case of reactionless blade modes. The necessary orientation of the hub end of the rotor drive shaft can be easily specified by taking the last section of the support structure to consist of only the localized bending lumped-parameter characteristics ϕ , θ , and ψ , which must be properly defined.

Counterclockwise Rotation

Clockwise Rotation

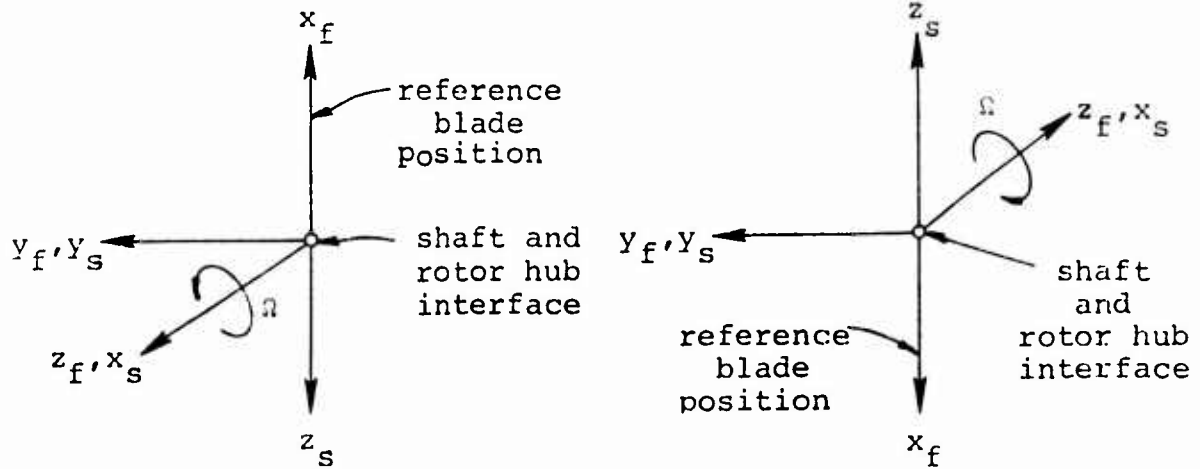


Figure 18. Relationship of the Local Support Structure Coordinate System and Fixed Hub Coordinate System Required at Shaft-Rotor Hub Interface.

GENERAL DISCUSSION OF EQUATIONS

Most of the mathematical representation of the various coupled rotor-helicopter systems that can be analyzed was developed rigorously in Reference 1. Additional mathematical analysis developed to extend the capabilities of the HASTA program was discussed in Reference 2. Thus, the mathematical analysis involved will not be discussed in detail herein. The intent of this section of the report is to provide a background concerning the primary computational procedures utilized by the HASTA program. In particular, the discussion herein is concerned with the numerical application of transfer matrix procedures, the numerical construction of the final governing matrix, and the eigenvalue and eigenvector solution method. This information should provide the user with a sufficient knowledge of the overall concept of the computer program.

It is to be noted that the representational equations (i.e., control system equations, rotor support structure equations, and blade equations) and thereby the final governing matrix equation are in a Laplace transformed form involving the Laplace transform variable s and frequency-shifted Laplace transform variables such as $s - ik\Omega$, where k has a non-zero integer value. The application of Laplace transformation techniques provides an effective means of converting time-differential equations having first and second order time-derivative terms, as well as periodically varying coefficients, to an algebraic form which can be efficiently manipulated. In addition, a characteristic of Laplace transformed equations allows the development of the additional equations necessary to define the harmonically shifted variables required for the consideration of interharmonic coupling due to the control system and/or rotor hub motion and to periodically varying aerodynamic coefficients. Because of the nature of the rotor/helicopter problem, the resulting algebraic equations involve complex variable notation such that the Laplace transform variables must be allowed to be complex variables.

TRANSFER MATRIX OPERATIONS

The HASTA program operation required for the transfer matrix procedure representation of a blade will be initially presented, followed by the operational modifications required for the rotor support structure representation. By considering the blade to be divided into sections representing the lumped-parameter characteristics of the blade, the k -frequency-shifted state

variable column vector at the inboard end of the i th section of the m th blade can be shown to be defined in general by the expression

$$\begin{aligned} \left\{ \bar{S}_k \right\}_m^i = & \sum_{n=-Nf}^{Nf} \left[\left[\bar{B}_{k,n} \right]_m^i \left\{ \bar{S}_n \right\}_m^* - \left\{ \bar{b}_{k,n} \right\}_m^i (\bar{\Delta 1}_n)_m - \left\{ \bar{c}_{k,n} \right\}_m^i (\bar{\Delta 2}_n)_m \right. \\ & - \left\{ \bar{d}_{k,n} \right\}_m^i (\bar{\Delta 3}_n)_m - \left\{ \bar{e}_{k,n} \right\}_m^i (\bar{\Delta 4}_n)_m \\ & \left. - \left\{ \bar{f}_{k,n} \right\}_m^i (\bar{\Delta 5}_n)_m - \left\{ \bar{g}_{k,n} \right\}_m^i (\bar{\Delta 6}_n)_m \right] . \end{aligned} \quad (10)$$

In this expression, the state variable column vector is defined by the form

$$\left\{ \bar{S} \right\} = \left\{ \overline{u_x} \quad \overline{N} \quad \overline{\phi_x} \quad \overline{T} \quad \overline{u_y} \quad \overline{\phi_z} \quad \overline{M_z} \quad -\overline{V_y} \quad -\overline{u_z} \quad \overline{\phi_y} \quad \overline{M_y} \quad \overline{V_z} \right\}$$

where the state variables were defined previously in the discussion following Figure 10 and the n -frequency-shifted blade tip unknowns column vector is defined by

$$\left\{ \bar{S}_n \right\}_m^* = \left\{ \overline{u_x} \quad \overline{\phi_x} \quad \overline{u_y} \quad \overline{\phi_z} \quad -\overline{u_z} \quad \overline{\phi_y} \right\}_{n,m}^*$$

The $(\bar{\Delta 1}_n)_m$, $(\bar{\Delta 2}_n)_m$, $(\bar{\Delta 3}_n)_m$, $(\bar{\Delta 4}_n)_m$, $(\bar{\Delta 5}_n)_m$, and $(\bar{\Delta 6}_n)_m$ quantities are the n -frequency-shifted blade discontinuity unknowns. The specific definition of these unknowns and the inclusion of their contribution to Equation (10) is dependent upon the type of rotor system being considered. The

associated blade matrices $\left[\bar{B}_{k,n} \right]_m^i$ denote the contribution of the n -frequency-shifted m th blade tip unknowns to the k -frequency-shifted state variables at the i th section of the m th blade. These matrices are 12×6 arrays due to the implementation of the blade tip boundary conditions (i.e., blade tip shears and moments are zero in value). The blade discontinuity column vectors

$$\{\bar{b}_{k,n}\}_m^i, \{\bar{c}_{k,n}\}_m^i, \{\bar{d}_{k,n}\}_m^i, \{\bar{e}_{k,n}\}_m^i, \{\bar{f}_{k,n}\}_m^i, \text{ and } \{\bar{g}_{k,n}\}_m^i$$

denote the contribution of the n-frequency-shifted mth blade discontinuity unknowns to the k-frequency state variables at the ith section of the mth blade. The quantity N_f represents the number of harmonics above and below the main eigenvalue of interest s which are allowed to couple with the main eigenvalue-related behavior. Thus, if N_f is 1, the $1/\text{rev}$ and $-1/\text{rev}$ shifted frequencies will be allowed to couple with the 0 shifted frequencies.

The computer program during a normal iteration does not calculate the state variables at the inboard end of the ith blade section. Instead, the computer program determines the associated blade transfer matrices and the necessary blade discontinuity vectors at the inboard end of each section. This is accomplished by consecutively multiplying, in matrix form, the associated blade transfer matrices and the necessary blade discontinuity vectors for the inboard end of the previous section by the individual transfer matrices for the section lumped-parameters, as these parameters are crossed in going inboard along the section. The associated blade transfer matrices and blade discontinuity vectors are not stored for each section but are updated by successive transfer matrix multiplication such that their final numerical values for an iteration correspond to those defining the blade state variables at the rotor hub centerline as required for construction of the final matrix. The construction of the final matrix also requires information concerning the numerical values occurring in specific rows of these matrices and vectors at the blade discontinuity locations. This information is saved in storage arrays as these discontinuities are crossed. After eigenvalue convergence is obtained the blade transfer matrix procedure is used to obtain the values of the k-frequency-shifted state variables at the inboard end of each blade section. These state variables are determined relative to both their local coordinate systems and the disk plane (rotating hub coordinate system).

The general manner in which the associated blade transfer matrices and discontinuity vectors are obtained will be briefly discussed. At the start of the transfer matrix procedure the off-diagonal associated blade transfer matrices ($k \neq n$) are initialized to 12×6 arrays consisting of zeroes.

The diagonal associated blade transfer matrices ($k=n$) are initialized to zeroes except for a value of unity in each of the six columns such that on multiplying the blade tip unknown column vector by the 12×6 array the same vector results. As each lumped-parameter characteristic is crossed, the associated blade transfer matrices inboard of the lumped-parameter characteristic are obtained in terms of those outboard by carrying out the numerical operation specified by

$$\left[\bar{B}_{k,n} \right]_m^j = \left[\bar{G}_k \right]_m^j \left[\bar{B}_{k,n} \right]_m^{j-1} \quad (11)$$

if a non-aerodynamic lumped-parameter characteristic is being considered and

$$\left[\bar{B}_{k,n} \right]_m^j = \sum_{\ell=-m}^{Nf} \left[\bar{G}_{k,\ell} \right]_m^j \left[\bar{B}_{\ell,n} \right]_m^{j-1} \quad (12)$$

if an aerodynamic lumped-parameter characteristic is being considered. In these expressions j is used to denote the lumped-parameter characteristic index and $\left[\bar{G}_k \right]_m^j$ may represent a k -frequency-shifted mass or spring transfer matrix or an unshifted (not involving s) elastic, bend, restraint, or rigid transfer matrix. These transfer matrices specify the effect of a particular lumped-parameter characteristic on the k -frequency-shifted state variables as the parameter is crossed.

The required blade discontinuity vectors, diagonal ($k=n$) and off-diagonal ($k \neq n$), initially consist of all zeroes. When a discontinuity is encountered the corresponding diagonal blade discontinuity vectors ($k=n$) are defined as necessary for the type of discontinuity involved. After a discontinuity has been crossed the corresponding blade discontinuity vectors are updated in the identical manner as exemplified by Equations (11) and (12) for the associated blade transfer matrices.

The rotor support structure is not exposed to any representational discontinuities or self-induced interharmonic coupling. Thus the k -frequency-shifted support structure state variables just beyond the i th section can be defined by the expression

$$\left\{ \bar{s}_k \right\}_s^i = \left[\bar{s}_k \right]_s^i \left\{ \bar{s}_k \right\}_s^* , \quad (13)$$

where the support structure state variables and the k-shifted support structure state variable vector are defined in the same manner as that for the blades except that they are related to the rotor support structure. In this expression

$\left[\bar{s}_k \right]_s^i$ is a 12 x 6 array representing the k-frequency-shifted associated support structure transfer matrix for the i th support structure section. The k-frequency-shifted support structure unknowns vector may consist of either the rotor support structure tip slope and deflection unknowns similar to that for the blades and corresponding free and conditions, or rotor support structure tip moments and shears corresponding to cantilevered end conditions such that

$$\left\{ \bar{s}_k \right\}_s^* = \left\{ \bar{N}_s \quad \bar{T}_s \quad \bar{Mz}_s \quad - \quad \bar{V}_y_s \quad \bar{M}_y_s \quad \bar{Vz}_s \right\}_k^* .$$

The support structure transfer matrix procedure is carried out by the computer program in a manner similar to that used in the blade transfer matrix procedure. The starting k-frequency-shifted associated support structure transfer matrices are all initialized to consist of zeroes except for a value of unity in each of the six columns such that on multiplying the support structure unknowns vector by the initial 12 x 6 matrices the same vector results. It is to be noted that there are no off-diagonal associate transfer matrices for this structure and the initial form of the associate transfer matrices is dependent upon the form of the support structure unknowns vector. As each lumped-parameter characteristic is crossed, the associated support structure transfer matrices just beyond the lumped-parameter characteristic are obtained in terms of those previous by use of the form specified in Equation (11) with

$$\left[\bar{B}_{k,n} \right]_m^i \text{ replaced by } \left[\bar{s}_k \right]_s^i .$$

After eigenvalue convergence is obtained, the k-frequency-shifted support structure state variables are then determined at each section relative to the local and fixed coordinate

systems of the rotor support structure by use of the support structure transfer matrix procedure.

FINAL GOVERNING MATRIX

The system representational equations (Equations (20), (21), (29), (30), (31), (32), (33), and (35) of Reference 1 and the additional blade discontinuity-related equations discussed in Reference 2), as required for the coupled rotor/helicopter system of interest, can be used in conjunction with Equations (10) and (13) to define the k-frequency-shifted governing equations. These k-frequency-shifted governing equations can be expressed in the simple matrix form

$$\sum_{n=Nf}^{Nf} [\bar{T}_{k,n}] \{\bar{q}_n\}^* = 0 \quad (14)$$

where $[\bar{T}_{k,n}]$ specifies the contribution of n-frequency-shifted unknowns contained in $\{\bar{q}_n\}^*$ to the k-frequency-shifted governing equations.

As an example of the construction of $[\bar{T}_{k,n}]$, the general form of this matrix for a coupled three-bladed rotor/helicopter system is given by the expression

$$[\bar{T}_{k,n}] = \begin{array}{c|c|c|c|c} \begin{array}{c} [\bar{S}W_{k,n}] \\ 0 \\ [\bar{B}S_k]_1 \delta_n^k \\ [\bar{B}S_k]_2 \delta_n^k \\ [\bar{B}S_k]_3 \delta_n^k \end{array} & \begin{array}{c} [\bar{S}F_{k,n}] \\ [\bar{F}U_k] \delta_n^k \\ [\bar{B}F_{k,n}]_1 \\ [\bar{B}F_{k,n}]_2 \\ [\bar{B}F_{k,n}]_3 \end{array} & \begin{array}{c} [\bar{S}B_{k,n}]_1 \\ [\bar{F}B_{k,n}]_1 \\ [\bar{B}B_{k,n}]_1 \\ [\bar{B}B_{k,n}]_2 \\ [\bar{B}B_{k,n}]_3 \end{array} & \begin{array}{c} [\bar{S}B_{k,n}]_2 \\ [\bar{F}B_{k,n}]_2 \\ [\bar{B}B_{k,n}]_2 \\ [\bar{B}B_{k,n}]_3 \end{array} & \begin{array}{c} [\bar{S}B_{k,n}]_3 \\ [\bar{F}B_{k,n}]_3 \\ [\bar{B}B_{k,n}]_3 \end{array} \end{array} \quad (15)$$

where integers have been used for blade subscripts and superscripts. The top row of submatrices are obtained from the control system governing equations. The second row of submatrices are obtained from the moment and shear boundary condition equations at shaft-rotor hub interface. The third, fourth, and fifth rows of submatrices are obtained from the slope and displacement boundary condition equations at the shaft-rotor hub interface and from the blade discontinuity equations for the first, second, and third blades, respectively. These submatrices have been defined in detail in Reference 1 for a fully coupled flexstrap, articulated, gimballed, or teetering rotor/helicopter system. For a fully coupled bearingless rotor/helicopter system the submatrix definitions are different than those of Reference 1 due to the different discontinuity unknowns and discontinuity equations which are involved. It is to be noted that the content and dimensions of some of the submatrices for a particular system are dependent upon the degree of inclusion of control system spatial harmonic effects, the number of blade discontinuities, and the type of rotor system.

The n-frequency-shifted unknowns vector is defined by

$$\left\{ \bar{q}_n \right\}^* = \left\{ \begin{array}{l} \left\{ \bar{r}_n \right\} \\ \left\{ \bar{s}_n \right\}_s^* \\ \left\{ \bar{p}_n \right\}_1 \\ \left\{ \bar{p}_n \right\}_2 \\ \left\{ \bar{p}_n \right\}_3 \end{array} \right\} \quad (16)$$

where the n-frequency-shifted control system unknowns are given by the general form

$$\left\{ \bar{r}_n \right\} = \left\{ \begin{array}{l} \bar{w}_{-N_{\max}}(s-in\Omega) \\ \cdot \\ \cdot \\ \bar{w}_{-1}(s-in\Omega) \\ \bar{w}_0(s-in\Omega) \\ \bar{w}_1(s-in\Omega) \\ \cdot \\ \cdot \\ \bar{w}_{N_{\max}}(s-in\Omega) \end{array} \right\}$$

in which N_{\max} is the limit on the number of spatial harmonics retained in the control system representation, the n-frequency-shifted support structure unknowns are given by

$\left\{ \bar{s}_n \right\}_s^*$, and the n-frequency-shifted mth blade unknowns are given by the general form

$$\left\{ \bar{p}_n \right\}_m = \left\{ \begin{array}{l} \left\{ \bar{s}_n \right\}_m^* \\ (\bar{\Delta 1}_n)_m \\ (\bar{\Delta 2}_n)_m \\ (\bar{\Delta 3}_n)_m \\ (\bar{\Delta 4}_n)_m \\ (\bar{\Delta 5}_n)_m \\ (\bar{\Delta 6}_n)_m \end{array} \right\}$$

Blade discontinuity unknowns that are not to be included for a particular system of interest are omitted from the blade unknowns vectors. In addition, the individual columns of the last three columns of submatrices in Equation (15),

which in the general form of these submatrices would multiply the discontinuity unknowns not included, would also be omitted.

The definitions provided by Equations (15) and (16) assume a coupled rotor/helicopter system having a control system and a rotor support structure in addition to the rotor blades. With alterations in the coupled rotor/helicopter system complexity the construction of this matrix and the unknowns column vector is modified. As an example, if a particular coupled tail rotor/helicopter system does not have a control system, the first row of submatrices and the first column of submatrices are omitted from the matrix and the control system unknowns are omitted from the unknowns vector. Similarly, if a particular system does not have a rotor support structure the second row of submatrices and the second column of submatrices are omitted from the matrix and the support structure unknowns are omitted from the unknowns vector. In addition, the removal of a blade requires the omission of the corresponding blade unknowns (including discontinuity unknowns) from the unknowns vector and the omission of the row and column of submatrices containing the associated diagonal blade

array. Thus, analytically the $\begin{bmatrix} \bar{T}_{k,n} \end{bmatrix}$ matrices can be as complex as shown in Equation (15), more complex due to additional blades, or as simple as $\begin{bmatrix} \bar{B}B_{k,n} \end{bmatrix}_1^1$.

The use of the blade phasing assumption for which all blades of the rotor are identical and equally spaced azimuthally

allows the size of the $\begin{bmatrix} \bar{T}_{k,n} \end{bmatrix}$ matrices to be significantly

reduced by the removal of the submatrices associated with blades other than the first blade. With the use of this

assumption the submatrices $\begin{bmatrix} \bar{S}B_{k,n} \end{bmatrix}_1$ and $\begin{bmatrix} \bar{F}B_{k,n} \end{bmatrix}_1$ have an

altered form to include the contributions of the additional blades. In the case of a gimbaled or teetering rotor, the

submatrices $\begin{bmatrix} \bar{B}B_{k,n} \end{bmatrix}_1^1$ also are modified. This option allows

the user to specify the type of blade modes to be obtained

(i.e., umbrella, reactionless, forward cyclic, or backward cyclic modes).

The k-frequency-shifted matrix equations represented by Equation (14) involve n-frequency-shifted unknowns which can be defined by considering values of k from -Nf to +Nf such that the final governing matrix equation required for solution may be obtained in the form

$$\begin{bmatrix}
 \boxed{\bar{T}_{-1,-1}} & \boxed{\bar{T}_{-1,0}} & \boxed{\bar{T}_{-1,1}} \\
 \boxed{\bar{T}_{0,-1}} & \boxed{\bar{T}_{0,0}} & \boxed{\bar{T}_{0,1}} \\
 \boxed{\bar{T}_{1,-1}} & \boxed{\bar{T}_{1,0}} & \boxed{\bar{T}_{1,1}}
 \end{bmatrix}
 \begin{Bmatrix}
 \{\bar{q}_{-1}\}^* \\
 \{\bar{q}_0\}^* \\
 \{\bar{q}_1\}^*
 \end{Bmatrix}
 = 0 \quad (17)$$

for Nf=1. In general, the number of $\boxed{\bar{T}_{k,n}}$ matrices required to construct the final governing matrix is defined by $(2Nf+1)^2$. Thus, the size of the final governing matrix for a particular coupled tail rotor/helicopter system is dependent on the value of Nf and the size of the individual $\boxed{\bar{T}_{k,n}}$ matrices. The use of the blade phasing assumption can be seen to significantly decrease the size of the final governing matrix which must be solved.

The operational procedure that the computer program uses to obtain the final governing matrix in numerical form is to determine the numerical form of the individual $\boxed{\bar{T}_{k,n}}$ matrices involved in the final governing matrix and to store this information in a consistent order in a temporary storage unit (disk or tape). The final governing matrix in numerical form as required for the determination of the determinant and assumed mode shape correction values is obtained by reading the stored values of the $\boxed{\bar{T}_{k,n}}$ matrix in the proper sequential order. It is to be noted that when

the numerical content of each $\left[\bar{T}_{k,n} \right]$ matrix is fully determined, the data is dumped off to tape so that the matrix storage array can be used for the next matrix.

The numerical construction of a $\left[\bar{T}_{k,n} \right]$ matrix is achieved by determination of the numerical content of each submatrix which is required to construct the matrix and proper placement of this information in the matrix storage array. In the majority of submatrices, the required numerical information is obtained by use of the final associated transfer matrices for the blade and rotor elastic support structure and the stored information required at blade discontinuity locations. Although the analytical representation uses matrix operators in addition to numerical operators to specify the particular information required for a submatrix, the computer program achieves the same representation by use of numerical operators, whose values can be obtained from input parameters, and proper indexing procedures. Some of the control system-related submatrices are directly constructed with only manipulation of the control system input parameters and proper indexing control. Due to the allowance of several different program representational options such as the type of rotor system, the type of blades, and the use of the blade phasing assumption, the construction of submatrices may have several different forms. Thus, the computer program coding for the construction of a submatrix may consist of several parts, of which only one part may be used for a particular coupled rotor/helicopter system.

EIGENVALUE AND EIGENVECTOR SOLUTION METHOD

The solution method used to obtain the eigenvalues and eigenvectors is based on a modified transfer-matrix method. This method avoids the numerical accuracy problems due to the taking of small differences of large numbers that generally occurs in transfer matrix techniques, particularly when the frequency determinant is computed for higher natural frequencies. In the solution procedure the eigenvector corresponding to the system unknowns and the eigenvalue are iterated simultaneously until the desired degree of eigenvalue convergence is achieved. This is accomplished by considering the eigenvector to consist of the sum of a trial eigenvector and a correction eigenvector such that as the trial eigenvalues are used in the iteration procedure to obtain a solution eigenvalue the trial eigenvector is being continuously updated.

The determination of the correction eigenvector for a given trial eigenvalue requires a modification of the final matrix equation for the trial eigenvalue. The final governing matrix equation, as represented by Equation (17), can be expressed in the form $[\bar{A}] \{\bar{X}\} = 0$. The $\{\bar{X}\}$ column vector can be defined in general as $\{\bar{X}\} = \{\bar{\lambda}\} + \{\bar{\epsilon}\}$ where $\{\bar{\lambda}\}$ is the trial eigenvector for an iteration and $\{\bar{\epsilon}\}$ is the correction eigenvector. By substitution of the $\{\bar{X}\}$ definition into the final governing matrix, the expression $[\bar{A}] \{\bar{\epsilon}\} = \{\bar{F}\}$ where $\{\bar{F}\} = [\bar{A}] \{\bar{\lambda}\}$ can be obtained. Premultiplication of this equation by the inverse of $[\bar{A}]$ results in $\{\bar{\epsilon}\} = 0$. This problem of indeterminate equations is avoided by normalizing a particular element \bar{x}_j of $\{\bar{X}\}$ to unity such that the corresponding $\{\bar{\lambda}\}$ and $\{\bar{\epsilon}\}$ elements, $\bar{\lambda}_j$ and $\bar{\epsilon}_j$, can be defined to always be unity and zero in value. Since $\bar{\epsilon}_j$ will always be zero by definition, the j th column of $[\bar{A}]$ can be removed from $[\bar{A}]$ and the $\bar{\epsilon}_j$ element removed from $\{\bar{\epsilon}\}$ such that the modified matrix equation $[\bar{A}'] \{\bar{\epsilon}'\} = -\{\bar{F}\}$ results where $[\bar{A}']$ is a $n \times (n-1)$ matrix and $\{\bar{\epsilon}'\}$ consists of $n-1$ elements. Thus, in this form the modified matrix equation represents one more equation than there are unknowns. Although, in general, any k th row of the modified matrix equation can be removed, best results for the coupled rotor/helicopter problem are obtained when λ_k is a variable closely associated with the λ_j defined as unity. For example, if the blade flapwise tip deflection is normalized to unity, the λ_k should correspond to the blade flapwise tip slope. On removal of the k th row from the modified equation, the equation $[\bar{A}''] \{\bar{\epsilon}'\} = -\{\bar{F}'\}$ is obtained, where $[\bar{A}'']$ results due to removal of the k th row of elements from $[\bar{A}']$ and $\{\bar{F}'\}$ results due to removal of the k th element from $\{\bar{F}\}$. By multiplying both sides of this equation by the inverse of $[\bar{A}'']$, the $n-1$ element correction vector is defined by

$$\{\bar{\epsilon}'\} = [\bar{A}'']^{-1} \{\bar{F}'\} . \quad (18)$$

The computer program for each trial eigenvalue, as the individual $[\bar{T}_{k,n}]$ matrices are constructed, also carries out the matrix multiplications necessary to determine $\{\bar{F}\}$. The determinant and correction eigenvector are determined on the basis of the $[\bar{T}_{k,n}]$ matrices and $\{\bar{F}\}$ by performing several steps.

First, the final matrix is constructed numerically by reading the $\left[\bar{T}_{k,n} \right]$ information in proper sequence from the auxiliary storage unit. The $[\bar{A}]$ matrix is then reordered by switching the j th column and last column of $[\bar{A}]$, which technically switches the j th variable and last variable of $\{\bar{\epsilon}\}$. Then the k th row and last row of the resulting matrix are switched and the k th element and last element of $\{\bar{F}\}$ are switched. A procedure is then applied which obtains the determinant of the modified $n \times n$ matrix and obtains the numerical $\{\bar{\epsilon}'\}$ correction factor using the modified $n \times n$ matrix with the last row and column omitted and the last element of the modified $\{\bar{F}\}$ vector omitted. In this procedure the $\{\bar{\epsilon}'\}$ correction vector is returned to the inputted $\{\bar{F}\}$. The $\{\bar{\epsilon}\}$ correction vector is obtained by defining the last element of what was $\{\bar{F}\}$ to have the value of ϵ_j as zero in value. This program operation accomplishes the manipulation required by Equation (18).

For the solution method iterative scheme, the program considers trial eigenvalues until the convergence criteria based on the changes in trial eigenvalues are satisfied or the number of allowed iterations have occurred. In particular, for a given starting trial eigenvalue and a set of parameters required for the analysis, the determinant of $[\bar{A}]$ and the correction vector $\{\bar{\epsilon}\}$ by use of Equation (18) are determined based on all quantities of the trial eigenvector $\{\bar{\lambda}\}$ being equal to unity. The $\{\bar{\epsilon}\}$ quantities are then added to the $\{\bar{\lambda}\}$ quantities to define a new $\{\bar{\lambda}\}$. The new trial eigenvalue is obtained by increasing the starting trial eigenvalue by a specified percentage. The new trial eigenvalue and new $\{\bar{\lambda}\}$ are used to determine a new determinant of $[\bar{A}]$ and a new $\{\bar{\epsilon}\}$ which is used to update $\{\bar{\lambda}\}$. From this point on the new trial eigenvalues are based upon a slope interpolation scheme using the previous two eigenvalues and corresponding determinant values of the form

$$s_{NE} = s_{CU} - \bar{\Delta}_{CU} (s_{CU} - s_{LA}) / (\bar{\Delta}_{CU} - \bar{\Delta}_{LA}).$$

In this expression, $\bar{\Delta}_{CU}$ and $\bar{\Delta}_{LA}$ denote the determinant values of $[\bar{A}]$ for the iteration just completed and the iteration just prior to the one just completed, respectively, and s_{CU} , s_{LA} , and s_{NE} denote the trial eigenvalues for the iteration just completed, the iteration just prior to the one just completed, and the next iteration, respectively. For each new eigenvalue the determinant and $\{\bar{\epsilon}\}$ are calculated with the latter used to update the $\{\bar{\lambda}\}$. After each iteration after the

first two trial eigenvalues have been used, a convergence criteria test is applied such that a sufficient eigenvalue and eigenvector have been obtained if

$\left| (s_{NE} - s_{CU})/s_{CU} \right| \leq . 1^{Mer}$, where $\left| \right|$ denotes the complex absolute value and Mer is an integer defining the convergence limit desired. As the program iteration proceeds, the determinant of $[\bar{A}]$ and the $\{\bar{E}\}$ quantities should both approach zero.

COMPUTER PROGRAM METHOD OF SOLUTION

The overall execution of the computer program is controlled by the main program. The main program initially determines, on the basis of input parameters, whether or not the scanning procedure is to be employed to obtain starting eigenvalues, and if so, specifies the operational procedure required. The main program then, on the basis of the number of starting eigenvalues inputted or obtained from the scanning procedure, the convergence criteria, and the iteration limit, directs the operational procedure of the program. If more than one case is to be considered in a computer run, the main program repeats the process for the additional runs. The overall iterative system flow for the computer program, assuming that starting eigenvalues are provided, is depicted in Figure 19.

The basic operational steps involved in the program method of solution are:

1. the input of system parameters,
2. the determination of intermediate terms,
3. the eigenvalue scanning procedure,
4. the blade transfer matrix procedure,
5. the rotor support structure transfer matrix procedure,
6. the construction of the $\left[\bar{T}_{k,n} \right]$ matrices and the trial eigenvector forcing function vectors $\{\bar{F}\}$,
7. the determination of the final matrix determinant and eigenvector correction array $\{\bar{\epsilon}\}$.

Each of these operational components will be discussed to various degrees.

The system model parameters, program logic parameters, and, if needed, aerodynamic coefficient data are required as input to the program. The method of input of data consists of reading into a storage array all data (except aerodynamic data) using a single format which requires specification of

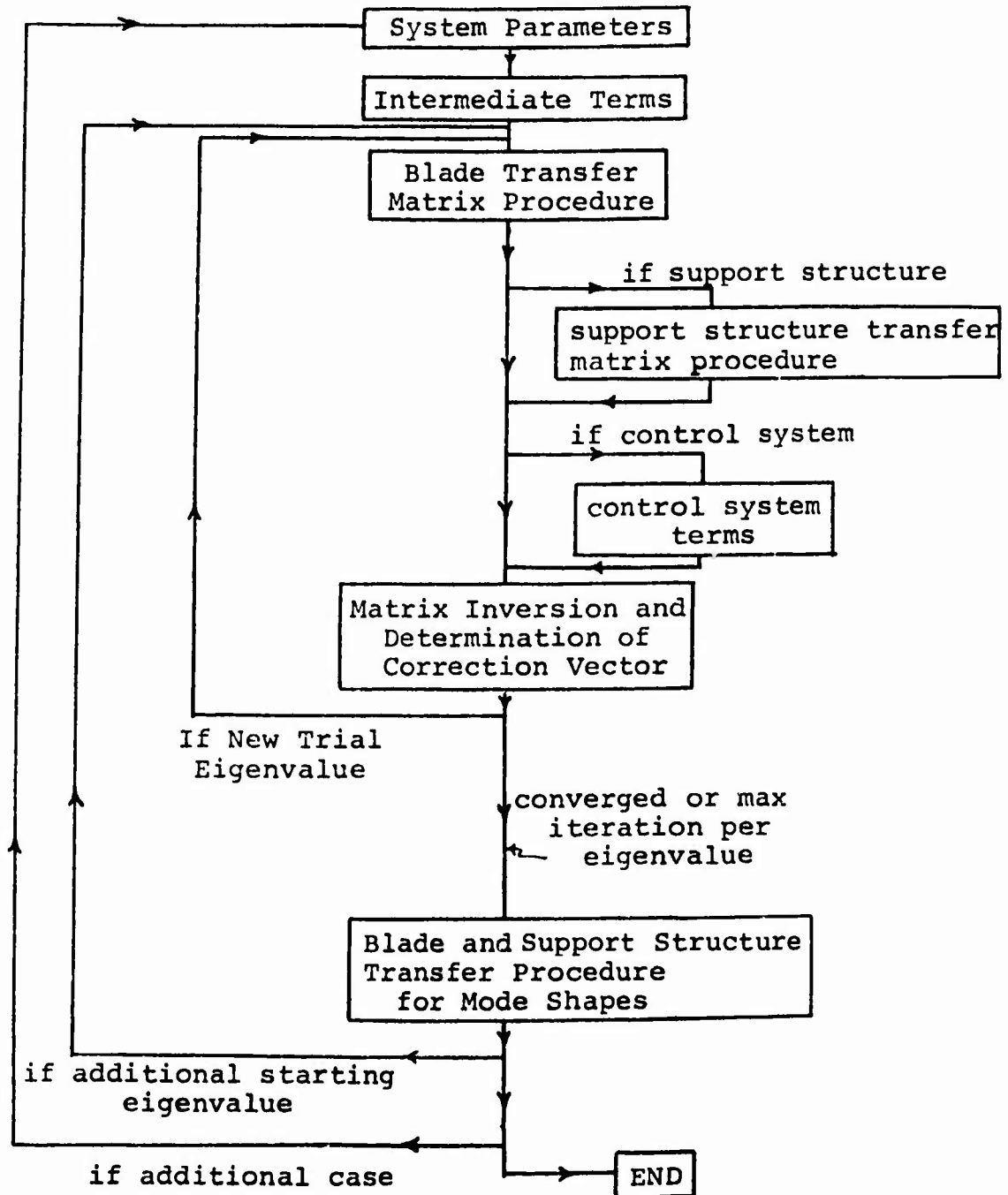


Figure 19. Overall Iterative System Flow.

the variable array location and the variable value in floating point form. The storage array input in floating point form corresponding to program integer variables is used to define the respective integer variables. The storage array consists of control and system parameter input stored in the first 200 locations, blade and rotor support structure section data in the next 2000 locations (50 locations per section), and a radial and azimuthally varying induced velocity distribution, if desired, in the last 600 locations. The reason for the use of this input form was to eliminate the necessity of a defined order of input so that input cards out of order would not result in erroneous results or aborted computer runs. This method of input also allows several model configurations to be considered consecutively, with only system values inputted for those that are altered from the previous configuration. In addition, a variable can be modified to a new value after being previously defined in the same set of input, such that a basic input deck followed by input modifications can be used for the specific configurations of interest. The aerodynamic coefficient data, if required, must be read in for each case of a run. The input of this data allows the use of several input formats.

Since many terms and multiplying coefficients involved in the analysis are not a function of the Laplace transform variables and thereby the trial eigenvalue, these terms and coefficients (intermediate terms) are computed and stored prior to the first iteration of a case instead of being recalculated on every iteration of the iteration procedure for every initial eigenvalue. Intermediate terms required for the representation of the blade and rotor support structure sectional characteristics, except for aerodynamics, are stored in the SD vector which is contained in the labelled common SAIN. Intermediate terms for the representation of the blade and rotor support structure aerodynamic sectional characteristics are stored in the AMA vector contained in the labelled common AMAT and the AFA vector contained in the labelled common FUSA. In addition, intermediate terms associated with the control system representation and the blade discontinuities are also stored. This procedure of calculating and storing intermediate terms was instituted to reduce the program running time for a system configuration.

The eigenvalue scanning procedure was incorporated into the program mainline as an independent control segment to allow the determination of the possible solution eigenvalues in a specified range of stability (real part of the eigenvalue) and frequency (imaginary part of the eigenvalue) values. If

this procedure is utilized, the program operation equivalent to a single iteration is executed for each eigenvalue of a grid of values defined by the specification of the lowest stability and frequency values to be considered, the stability and frequency step sizes, and the number of stability and frequency steps to be taken. For each trial eigenvalue of the grid, the determinant of the final governing matrix is determined. An interpolation scheme is applied to the resulting information to determine possible solution eigenvalues, which are then used as starting eigenvalues for the normal iterative procedure to determine the solution eigenvalues and corresponding eigenvectors and mode shapes for the problem of interest. This procedure is not foolproof, however, since reasonable step sizes in both frequency and stability values are required to avoid missing possible eigenvalues.

The blade and rotor support structure transfer matrix procedures are discussed in a previous section; therefore, only the overall concept of these procedures is reviewed here. The blade transfer matrix procedure consists of two basic forms, that used during the iterative procedure and that used to obtain the state variables (mode shapes) at the end of each section. During the iterative procedure, the representation of the blade variables at the rotor hub - in terms of the blade tip and discontinuity unknowns - is obtained for each trial eigenvalue by successive multiplication of the initial blade tip-associated transfer matrices and discontinuity columns by the individual lumped-parameter characteristic transfer arrays as each characteristic is crossed in transferring from blade tip to rotor hub. The individual blade transfer matrices are obtained by use of the stored intermediate terms and from the value of the trial eigenvalue. Also, during the application of the blade transfer matrices, information required for the blade discontinuity equations is stored as the discontinuities are crossed. When a solution eigenvalue is obtained (converged eigenvalue) the blade transfer matrix procedure is repeated for the solution eigenvalue in the same manner as during an iteration, except that at the inboard end of each blade section the associated blade transfer matrices and discontinuity columns corresponding to this location are applied to pertinent blade solution eigenvectors to obtain the frequency-shifted and unshifted blade section state variables. The solution eigenvector used to determine the values for the blade section state variables is equivalent to the sum of the trial eigenvector and the correction eigenvector associated with the iteration involving the solution eigenvalue.

The rotor support structure transfer matrix procedure is similar in concept to the blade transfer matrix procedure, except that due to the lack of structural discontinuities and aerodynamic interharmonic coupling the form of the representation is simpler. This transfer matrix procedure also consists of two basic forms; that used during the iterative procedure and that used to obtain the state variables (mode shapes) at each section of the rotor support structure when the eigenvalue has converged. These procedures are carried out in the same manner as for the blade representation but involve rotor support structure variables, unknowns, and intermediate parameters.

The representation of the frequency-shifted and unshifted blade and rotor support structure state variables for each trial eigenvalue at the rotor hub and discontinuity locations, in terms of the blade tip unknowns, discontinuity unknowns, and the rotor support structure unknowns, provides the majority of the terms necessary for the construction of the

$\bar{T}_{k,n}$ matrices as defined by Equation (15). The remaining terms, such as control system governing equation terms, can be obtained by direct use of the system parameters or a combination of system parameters and blade and rotor support structure variable representations. As previously discussed, the complexity of these matrices is dependent on the degree of interharmonic coupling involved, the degrees of freedom of the rotor shaft-rotor hub interface, the number of rotor blades, the inclusion of control system representation, the inclusion of a rotor support structure, the use of the blade phasing assumption, etc. Through the use of subroutines, the program for each trial eigenvalue determines the numerical construction of each $\bar{T}_{k,n}$ matrix as required by the value of N_f and stores this information on an auxiliary storage input and also obtains the trial eigenvector forcing function $\{\bar{F}\}$.

The determination of the final governing matrix determinant and the correction eigenvector for each trial eigenvalue, and the iteration procedure used to obtain the solution eigenvalue, eigenvector, and mode shapes, is discussed in detail in the previous section. It should be noted that the determinant of the final governing matrix and the correction vector for each trial eigenvalue are obtained simultaneously by use of a sophisticated triangularization technique. This technique operates on the total final governing matrix and trial eigenvector forcing function in a manner such that an iterative procedure to obtain the determinant and a matrix

inversion for the correction eigenvector are not required. This technique has proven to be much faster in execution time and more accurate than previously used matrix manipulation procedures.

The roles which the various subroutines perform in the overall operation of the computer program to obtain the solution for the eigenvalues, eigenvectors, and mode shapes of a coupled rotor/helicopter system can be noted in the section pertaining to the description of the functionality of the various subroutines.

INTERPRETATION OF COMPUTER PROGRAM RESULTS

The resultant eigenvalue and eigenvector values obtained by the use of the HASTA program represent the behavior of the total system investigated. This program provides the fundamental and harmonic coefficients in a complex variable form of the real-time control system, blade, and rotor support structure state variables. A rotor support structure state variable at the hubward end of the i th section can be expressed as a function of time in the general form

$$f(t)^{(i)} = \sum_{p=-\infty}^{\infty} (X_p^{(i)} + iY_p^{(i)}) e^{\sigma t} e^{i(\omega t + p\Omega t)}. \quad (19)$$

In this expression, $X_p^{(i)}$ and $Y_p^{(i)}$ are the real and imaginary part, respectively, of the state variable of interest, σ and ω are the real and imaginary part, respectively, of the solution eigenvalue, and p is an integer that when positive in value denotes an oscillatory behavior at a frequency $p\Omega$ radians per second higher than that corresponding to ω . In the HASTA program output the state variables are printed in the order of p , going from positive to negative. By conversion of the exponential function with the imaginary argument in Equation (19) to a sine and cosine equivalent representation, the form of this equation can be modified to

$$f(t)^{(i)} = \sum_{p=-\infty}^{\infty} R_p^{(i)} e^{\sigma t} \cos(\omega t + p\Omega t + \beta_p^{(i)}) \quad (20)$$

where $R_p^{(i)} = \sqrt{(X_p^{(i)})^2 + (Y_p^{(i)})^2}$ and $\beta_p^{(i)} = \arctan(Y_p^{(i)}/X_p^{(i)})$.

To be practical, the summation can be truncated to the range $-N_f$ to N_f .

This state variable representation can be modified to represent a blade state variable at the inboard end of the i th section when the blade phasing assumption is not employed, by adding m subscripts to all blade-related variables. The behavior of the m th blade is relative to the rotating coordinate system of the m th blade on an azimuthal basis. Thus, at t equal to zero, the value of a state variable for the m th blade defined by the form of Equation (20) is for the

blade ϕ_m radians azimuthally ahead of the fixed reference blade location. The variable ϕ_m is not involved in the non-phased blade state variable definition of the form of Equation (20) since the contribution of this variable has been included in the blade tip state variables by the analysis coded.

When blade phasing is used, state variable coefficients are obtained for the first blade for which ϕ_1 is defined to be 0 in value, such that the definition for additional blades is obtained by use of the phasing relationship

$$\left\{ \bar{S}_n \right\}_m^{(i)} = \left\{ \bar{S}_n \right\}_1^{(i)} e^{i(-n-Nps)\phi_m} \quad \text{where } Nps \text{ is an integer}$$

denoting the type of phasing specified. From this relationship the p -related coefficients for a state variable of the m th blade can be defined as the corresponding state variable of the first blade multiplied by $e^{i(p-Nps)\phi_m}$. Thus, for blade phasing the representation used above for a support structure state variable would be modified, with the blade-related quantities having a subscript of 1 added, except for $f(t)^{(i)}$, which would have an m subscript added and would have $p\Omega t$ replaced with $p(\Omega + \phi_m) - Nps\phi_m$. The resulting m th blade state variables are azimuthally referenced to their own rotating hub coordinate system.

The fundamental and harmonic coefficients of the blade state variables defined both relative to the local blade coordinate system axes and to the rotating hub coordinate system (disk plane) axes can be provided in complex variable form by the HASTA program. Thus, the required form of Equations (19) and (20) can be used to determine the time-dependent behavior of the blade state variables as defined relative to either set of coordinate system axes, depending on which state variable output is used. The disk plane-related blade state variable fundamental and harmonic coefficients, if provided in complex variable form, are also provided in polar form such that only Equation (20) need be used to define the time-dependent behavior of the blade state variable as defined relative to the disk plane coordinate system axes. Similar comments are appropriate to the support structure state variables in that the fundamental and harmonic coefficients of these state variables can be provided relative to the support structure

fixed coordinate system in both complex variable and polar form, in addition to those of the standard complex variable form defined relative to the local support structure coordinate systems. The control system rotating frame displacement variable $w_\ell(t)$ can be defined in the same form of representation as for the rotor support structure, with $f(t)^{(i)}$ replaced by $w_\ell(t)$ and $X_p^{(i)}$ and $Y_p^{(i)}$ replaced by $X_{\ell,p}$ and $Y_{\ell,p}$, respectively. The latter corresponds to the coefficients of the ℓ th spatial harmonic of control system ring motion in the rotating control system coordinate system at a frequency of $\omega + p\Omega$.

DESCRIPTION OF SUBROUTINE AND FUNCTIONS

<u>Subroutine or function</u>	<u>Description</u>
ACOEFF	called from AERO and computes the two-dimensional lift, drag, and moment coefficients and their slopes based on an airfoil angle of attack, Mach number, and specified type of aerodynamic coefficient representation
AERO	called from BAERO and FAERO and computes aerodynamic angle of attack, Mach number, and preliminary aerodynamic terms (including Theodorsen aerodynamics on blade) required to determine the intermediate terms necessary for the aerodynamic transfer matrices at an aerodynamic load point
BAERO	called from SECPAR and computes the intermediate terms necessary for the construction of the aerodynamic transfer matrices at a blade aerodynamic load point by performing a harmonic analysis of functions
BARRAY	called from the main program and performs the transfer matrix procedure for successive blade sections to obtain the associated blade transfer matrices and discontinuity column vectors with the necessary information at discontinuity locations being stored for subsequent use during an iteration and the blade state variables at each section (mode shapes) being determined with respect to the local section and disk plane coordinate system after eigenvalue convergence
BEND	called from BARRAY and SARRAY and constructs the bend transfer matrix necessary to account for changes in the local blade or rotor support structure section coning, pitch, and sweep angles

<u>Subroutine or function</u>	<u>Description</u>
BLARO	called from BARRAY and constructs the aerodynamic transfer matrices necessary to account for aerodynamic effects on a blade section
BMASS	called from BARRAY and constructs the mass transfer matrices necessary to account for mass and inertia effects on a blade section
CDOT	called from MLCC2 and is a FORTRAN-callable complex function written in assembler language that obtains the dot product of two complex vectors
DCMAT	called from SOLVE and computes the determinant of a nxn complex matrix and solves the nxn complex matrix equation, returning the matrix inverse to the input matrix and the solution to the input column
ELAST	called from BARRAY and SARRAY and constructs the elastic transfer matrix necessary to account for flexibility of a blade or rotor support structure section or constructs a transfer matrix associated with flexstrap or torque tube blade restraints (blade section only)
EPSOLN	called from the main program and controls the order of determination of the $\begin{bmatrix} T_{k,n} \end{bmatrix}$ matrices and their transferral to auxiliary storage (tape or disk) and constructs the $\begin{bmatrix} \bar{F} \end{bmatrix}$ vector
ERRSET	called from SOLVE and is an IBM-supplied subroutine for handling errors
EXCHI	called from ZLN, XNLQ, and ULN and computes exponential functions of the general form $e^{i(L-Q)\chi_j}$ where L and Q are both integers

<u>Subroutine or function</u>	<u>Description</u>
EXPON	called from SUBMA, SUBMB, SUBMF, SWA, SWB, and ZTEGI; computes the exponential functions of the general form $e^{iL\phi_m}$ where L is an integer
FAERO	called from SETUP and computes from the preliminary aerodynamic terms the intermediate terms necessary for the construction of the aerodynamic transfer matrices at a rotor support structure aerodynamic load point
FMASS	called from SARRAY and constructs the mass transfer matrices necessary to account for mass and inertia effects on a rotor support structure section
FUARO	called from SARRAY and constructs the aerodynamic transfer matrices necessary to account for aerodynamic effects on a rotor support structure section
LOADIN	called from SETUP and reads in the program control parameters, blade and fuselage section data, and control system data
MLCC2	called from BARRAY and SARRAY and multiplies, in string mode, a complex 12xn matrix by a complex 12x12 matrix, returning the complex results in the input 12xn matrix
MLRC2	called from BARRAY and SARRAY and multiplies, in string mode, a complex 12xn matrix by a real 12x12 matrix, returning the complex results in the input matrix
POLAR	called from BARRAY and SARRAY and converts state variable harmonic coefficients to a polar form
RCDOT	called from MLRC2; obtains dot products of a real and a complex vector and is a FORTRAN-callable complex function written in assembler language

<u>Subroutine or function</u>	<u>Description</u>
RIGID	called from BARRAY and SARRAY and constructs the rigid transfer matrix necessary to account for rigid shear center axis offsets on a blade or rotor support structure section
ROWSUM	called from DCMAT and is a FORTRAN-callable subroutine written in assembler language which performs matrix row operations
SARRAY	called from the main program and performs the transfer matrix procedure for successive rotor support structure sections to obtain the associated rotor support structure transfer matrices with the rotor support structure state variables at each section (mode shapes) being determined with respect to the local and fixed rotor support structure coordinate systems after eigenvalue convergence
SECPAR	called from SETUP and computes the intermediate transfer matrix-related terms for both blade and rotor support structure sections and stores this information in the SD storage array, constructs the aerodynamic AMA storage array, and also outputs the section input data for all sections and the blade section steady forces and moments
SETUP	called from the main program and calls LOADIN and TABLU for the reading of system input parameters and aerodynamic data, respectively; defines integer control variables based on their corresponding floating point form variables, calculates intermediate terms including the construction of the aerodynamic AFA storage array, calculates program control parameters including indexing parameters, and outputs input data with descriptive headings

<u>Subroutine or function</u>	<u>Description</u>
SOLVE	called from the main program; reads the final matrix from the auxiliary storage unit and obtains the determinant of the final matrix and the correction eigenvector on each iteration
STIFF	called from BARRAY and SARRAY and constructs the transfer matrices necessary to account for torsional spring-damper units on a blade or rotor support structure section; can be used to account for a flap or lead-lag hinge between two blade sections
SUBMA	called from TKNS; for a value of k and n constructs the elements of the $\begin{bmatrix} \bar{T}_{k,n} \end{bmatrix}$ matrix corresponding to the $\begin{bmatrix} \bar{F}U_k \end{bmatrix}$ and $\begin{bmatrix} \bar{B}F_{k,n} \end{bmatrix}_m$ submatrices of Equation (15), which specify the contribution of the rotor support structure unknowns to the blade and rotor support structure related governing equations
SUBMB	called from TKNS; for a value of k and n constructs the elements of the $\begin{bmatrix} \bar{T}_{k,n} \end{bmatrix}$ matrix corresponding to the $\begin{bmatrix} \bar{F}B_{k,n} \end{bmatrix}_m$ submatrices of Equation (15), which specify the contribution of the blade unknowns to the rotor support structure-related governing equations; constructs the first six rows of the $\begin{bmatrix} \bar{B}B_{k,n} \end{bmatrix}_1^1$ submatrix of Equation (15), which specify the contribution of the blade unknowns of the first blade to the blade-related governing equations of the first blade (not including discontinuity equations)

<u>Subroutine or function</u>	<u>Description</u>
SUBMF	called from TKNS and for a value of k and n constructs the elements of the $\overline{T}_{k,n}$ matrix corresponding to the $\overline{BB}_{k,n}^m$ submatrices of Equation (15) for values of m other than unity
SUBMG	called from TKNS and for a value of k and n constructs the elements of the $\overline{T}_{k,n}$ matrix corresponding to discontinuity rows of the $\overline{BB}_{k,n}^1$ submatrix of Equation (15), which specify the contribution of the blade unknowns of the first blade to the blade discontinuity equations of the first blade
SWA	called from TKNS and for a value of k and n constructs the elements of the $\overline{T}_{k,n}$ matrix corresponding to the $\overline{SW}_{k,n}$ and \overline{BS}_k^m submatrices of Equation (15), which specify the contribution of the control system unknowns to the control system governing equations and to the blade-related governing equations, respectively
SWAPS	called from SOLVE; swaps the column designated by NORM with the last column of the final matrix and then swaps the row designated by IREM with the last row of the final matrix (note: NORM and IREM are associated with the location of a column and row in the $\overline{T}_{0,0}$ matrix

<u>Subroutine or function</u>	<u>Description</u>
SWB	called from TKNS and for a value of k and n constructs the elements of the $\overline{T}_{k,n}$ matrix corresponding to the $\overline{SF}_{k,n}$ and $\overline{SB}_{k,n}$ submatrices of Equation (15), which specify contributions of the rotor support structure and blade unknowns to the control system governing equations
TABLU	called from SETUP and reads in aerodynamic data, if required
TKNS	called from EPSOLN and for each value of k and n calls the subroutines necessary for the construction of $\overline{T}_{k,n}$
ULN	called from SWB and computes multiplier terms involved in the determination of the elements of the $\overline{T}_{k,n}$ matrix corresponding to the $\overline{SF}_{k,n}$ submatrix
XNLQ	called from SWA and computes the elements of the $\overline{T}_{k,n}$ matrix corresponding to the diagonal elements of $\overline{SW}_{k,n}$
ZLN	called from SWA and computes the elements of the $\overline{T}_{k,n}$ matrix corresponding to the off-diagonal elements of $\overline{SW}_{k,n}$

Subroutine
or function

Description

ZTEGI

called from TKNS when the rotor system is gimballed or teetering and computes the elements of the $\overline{\mathbf{T}}_{k,n}$ matrix corresponding to the $\overline{\mathbf{B}}_{k,n}^j$ submatrices of Equation (15) where $j \neq m$, redefines some of the elements corresponding to $\overline{\mathbf{B}}_{k,n}^m$ as required for these two types of rotor systems

COMPUTER PROGRAM SYMBOL DICTIONARY

Due to the quantity of computer program variables involved in the HASTA program, the program variables to be included in this section are restricted to those which are more significant. Thus, input variables which are only used to define other program variables on a one-to-one basis (such as floating point variables which define integer variables), some of the intermediate variables that are used to define other intermediate and final variables, and indexing variables are not included in the following definitions. The number of indices involved in array variables is denoted by using the general form exemplified by AA(I,J,K,L). The dimensional requirements and present dimensions of the program arrays are specified as part of the user's manual portion of this report. The algebraic symbols used in Reference 1 and this report corresponding to the computer program variables are provided where applicable. It is to be noted that although a program name may represent more than one variable there is no conflict since an identical program name may be used in a subroutine. Where applicable, units are given in English units but equivalent SI units or other systems of units (e.g., inches instead of feet) may be used if applied to all definitions of program variables.

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
A(I)	$\left[\bar{A}_k \right]_m^i$	array representing a blade mass transfer matrix in string mode by row
A(I,J)	$\left[\bar{A}_{ij} \right], \left[\bar{A} \right]$	final governing matrix array
AC(I)	c_j	damping coefficient of the control system ring jth cyclic spring-damper unit support, lb-sec/ft
AC(I)	$\left[\bar{C}_{n-k} \right]_m^i$	array specifying the contribution of aerodynamic damping terms to a blade section aerodynamic transfer matrix in string mode by row

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
ACP(I)	$c\phi_j$	torsional damping coefficient of control system ring jth torsional spring-damper unit counteracting local ring longitudinal rotation, ft-lb-sec/rad
ACT(I)	$c\theta_j$	torsional damping coefficient of control system ring jth torsional spring-damper unit counteracting local ring lateral rotation, ft-lb-sec/rad
AD(I)	$\left[\bar{D}_{n-k} \right]_m^i$	array specifying the contribution of non-damping aerodynamic terms to a blade section aerodynamic transfer matrix in string mode by row
AEXT	L_s	length of torque tube pitch control structure spur or extension shaft, ft
AFA(I)		storage array for intermediate support structure section aerodynamic terms
AG	g	gravitational acceleration constant, ft/sec ²
AIRD	ρ	air density, lb-sec ² /ft ⁴
AK(I)	k_j	linear stiffness of the control system ring jth cyclic spring-damper unit support, lb/ft
AKCI(I)	$1/k_m$	reciprocal of the mth blade control rod axial stiffness for an articulated type blade, ft/lb

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
AKP(I)	$k\phi_j$	torsional stiffness of control system ring jth torsional spring-damper unit counteracting local ring longitudinal rotation, ft-lb/rad
AKT(I)	$k\theta_j$	torsional stiffness of control system ring jth torsional spring-damper unit counteracting local ring lateral rotation, ft-lb/rad
AL(I,J)		trigonometric functions of control rod orientation angles for the jth flexstrap or bearingless blade
AL12(I)	$L12_m, L_{cc}$	perpendicular distance between rotor disk plane and mth blade control rod-control system attachment point, positive if attachment point is in $-z_{rm}$ direction from disk plane, ft (for flexstrap or bearingless blade)
ALM(I)	$\{\lambda\}$	trial eigenvector column array
ALPHS(I,J,K,L)	α_i	storage array for the angles of attack inputted for each Mach number for each type of aerodynamic coefficient for each aerodynamic table number
ALPHZ	α_j	calculated angle of attack at an aerodynamic load point, rad (but usually converted to degrees, dependent on aerodynamic coefficient representation used)

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
ALSTH(I)	L_b	perpendicular distance between rotor disk plane and spur spherical bearing attachment point, positive if attachment point is in $-z_{rm}$ direction from disk plane, ft (for bearingless blade)
AMA(I)		storage array for intermediate blade section aerodynamic terms
AMACH(I,J,K)	M_i	storage array for the Mach number inputted for each type of aerodynamic coefficient for each aerodynamic table number
AMACZ		calculated Mach number at an aerodynamic load point
AMKC(I)	$1/K_{c,m}$	reciprocal of the mth blade control rod axial stiffness for a flexstrap or bearingless blade, ft/lb
AMY(I,J,K), AMZ(I,J,K)	$(\overline{M_y}_m)_k^i, (\overline{M_z}_m)_k^i$	arrays containing the shifted and unshifted flapwise and edgewise bending moments, respectively, acting on each section of each blade
AMY(I,J) AMZ(I,J)	$(\overline{M_y}_s)_k^i, (\overline{M_z}_s)_k^i$	arrays containing the shifted and unshifted lateral and vertical bending moments, respectively, acting on each support structure section
AN(I,J,K)	$(\overline{N}_m)_k^i$	array containing the shifted and unshifted radial forces acting on each section of each blade

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
AN(I,J)	$(\bar{N}_s)_k^i$	array containing the shifted and unshifted axial forces acting on each support structure section
ARMO, ARVZ, ARVY		steady torque, flapwise force, and edgewise force, respectively acting on a blade section due to aerodynamic effects
ATAB		number of the aerodynamic data table to be used for an aerodynamic section
AZER		velocity of sound, ft/sec
B(I)	$\left[\overline{BE} \right]_m^i$	array representing a blade or support structure bend transfer matrix in string mode by row
B(I)	$\left[\overline{B}_{k,n} \right]_m^i$	array representing all k and n subscripted associated blade transfer matrices at a section where each matrix is stored in string mode by column and the order of storage of these matrices in the array is accomplished by varying n from -Nf to +Nf for each value of k from -Nf to +Nf
BJ(I)	b_j	offset of the control system jth cyclic spring-damper unit attachment point from the circumferential axis of the control system ring, positive toward center of ring, ft
BSAVE(I)		storage array for the blade-associated transfer matrices B(I) at an aerodynamic characteristic; required to properly transfer across this type of characteristic because of interharmonic coupling effects

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
BVLI, BVLJ, BVLK		calculated velocities of an aerodynamic load point with respect to the first blade-associated rotating hub coordinate system \bar{I} , \bar{J} , and \bar{K} directions, respectively, ft/sec
C(I)	$\begin{bmatrix} C_{n-k} \end{bmatrix}_m^i$	aerodynamic transfer matrix array identical to AC(I)
CAPC	C	damping coefficient of control system base plate spring-damper unit support, lb-sec/ft
CAPFY, CAPFZ, CAPM		steady two-dimensional local aerodynamic edgewise force, flapwise force, and moment, respectively, at an aerodynamic load point
CAPK	K	linear stiffness of control system base plate spring-damper unit support, lb/ft
CAPP		effective rotational velocity of the blade around its spanwise axis due to coning, rad/sec
CCS(I, J, K)	$C1_i, C2_i, C3_i$	storage array for series representation coefficients inputted for each type of aerodynamic coefficient for each aerodynamic table number
CFORC	P_T	centrifugal force acting on a blade section, lb
CHI	χ	one of two angles which specify orientation of the helicopter velocity vector with respect to the fixed hub coordinate system, rad

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
CI,CR		frequency (rad/sec) and stability (1/sec) parts of a trial eigenvalue used during the search option procedure
CIS,CRS		frequency (rad/sec) and stability (1/sec) parts of the initial eigenvalue used during the search option procedure
CLALP,CDALP, CMALP		calculated lift, drag, and moment coefficient slopes, respectively, at an aerodynamic load point
CLDM(I,J,K,L)	CL_i, CD_i, CM_i	storage array for the values of the aerodynamic coefficients for each angle of attack for each Mach number for each type of aerodynamic coefficient for each aerodynamic table number
CM1	i	imaginary number, $\sqrt{-1}$
CMS	s	Laplace transform variable
COE1		steady blade torque at pitch control structure - blade attachment point not removed by pitch control structure, ft-lb (for articulated blade without pitch bearing, flexstrap blade, or bearingless blade)
COE2		effective distance, which in conjunction with COE1 defines steady flapwise shear force acting on blade as a result of steady blade torque removed by pitch control structure, ft
COE3		percentage of steady blade torque not removed by pitch control structure (for articulated blades with a pitch bearing)

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
COLL	θ_{coll}	collective pitch of the blades, rad, if not included in the blade section twist distribution
CPSI(I,J)		array containing terms of the form $\cos(L(K-1)2\pi/NAS)$
CS(I,J)		array containing terms of the form $\cos L\phi_m$
CS1,CS2		k-frequency-shifted Laplace transform variable $s-ik\Omega$ and $(s-ik\Omega)^2$, respectively
CSA(I,J)		array containing terms of the form $\cos L\chi_j$
CSUBL,CSUBD,CSUBM		calculated lift, drag, and moment coefficients at an aerodynamic load point
CTAU(I)	$C_{\tau_{c,m}}$	damping coefficient of the mth blade control rod for a flex-strap or bearingless blade, lb-sec/ft
CTB(I)		array representing contributions of the blade tip state variables to the control torque discontinuity equations for an articulated blade, to the forces acting on the control rod end of the pitch arm in the rotating hub coordinate system x-axis direction for a flex-strap blade, and to the deflection of the control rod end of the pitch arm in the rotating hub coordinate system x-axis direction for a bearingless blade, where in all cases the information is stored by varying n from -Nf to +Nf for each value of k from -Nf to +Nf

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
CTB1(I), CTB2(I), CTB3(I), CTB4(I)		arrays containing the elements required to denote the blade torque, pitch bearing, flap, and lead-lag discontinuity contributions, respectively, to the control torque discontinuity equations for an articulated blade where the order of storage is the same as for CTB(I)
CX(I)	$C_{i,m}$	influence coefficients used to construct the restraint transfer matrices associated with the pitch control structure of a flexstrap or bearingless blade
CYFA		fore and aft blade cyclic pitch corresponding to the value of blade cyclic pitch when blade is located in the fixed hub coordinate system -y-axis direction, rad
CYLA		lateral blade cyclic pitch corresponding to the value of blade cyclic pitch when blade is located in the fixed hub coordinate system x-axis direction, rad
D(I)	$\begin{bmatrix} D_{n-k} \end{bmatrix}_m^i$	aerodynamic transfer matrix array identical to AD(I)
DACO		control variable such that if greater than zero in value, mass and inertia-related damping are not included in blade mass transfer matrix
DAT(I)		input data storage array which is equivalenced to labelled common PLOAD

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
DB(I,J)	$\bar{A}_{i,j}$, \bar{A}	final governing matrix array-identical to A(I,J)
DETR(I)		array storing the real part of the determinant for a zero-valued imaginary part based on interpolation at the first change in sign of the imaginary part during scan of stability values for each frequency
DETR2(I)		same concept as DETR(I) but pertains to a second change in sign of the imaginary part of the determinant
DETSV		value of the determinant of the final governing matrix with a factor of $10^{(n*20)}$ removed to avoid overflow problems
DFAC		corresponds to the power of 10 (10^{DFAC}) which is removed on calculating DETSV and is an integer multiple of 20
DFYDP,DFZDP, DMDP		change in the total local aerodynamic edgewise force, flapwise force, and moment, respectively, with respect to a change in the blade local torsional angle at an aerodynamic load point
DFYDVZ,DFZDVZ, DMDVZ		change in the total local aerodynamic edgewise force, flapwise force, and moment, respectively, with respect to a change in the blade local flapwise velocity at an aerodynamic load point

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
DMS(I)	d_m	rigid offset of the mth blade control rod attachment point from the circumferential axis of the control system ring, positive outward, ft
DPHAS, DPIVOT		complex numbers which when multiplied together provide the determinant value of the determinant matrix modified to have diagonal elements of a complex absolute value of unity
DTLG10		factor removed from the determinant in the process of reducing the determinant matrix to one which has diagonal elements with a complex absolute value of unity
E(I)	$\begin{bmatrix} E \end{bmatrix}_m^i$	array representing a blade or support structure elastic transfer matrix in string mode by row where this matrix can also represent an elastic restraint applied to the blade as in the case of a flexstrap or bearingless blade
EC(I)		array storing the interpolated eigenvalues associated with the data stored in DETR(I)
EC(2)		array storing the interpolated eigenvalues associated with the data stored in DETR2(I)
ECHI(I)	χ_j	azimuthal angle of the jth control system cyclic spring-damper unit support relative to the fixed shaft coordinate system, rad

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
EE(I)		integer data block whose six values specify the elements of the initial associated support structure transfer matrices that are set to unity for free support structure conditions
EGNC, EGNN, EGNL		trial eigenvalue for the iteration just completed, the next iteration, and previous iteration, respectively
EGNS(I)		starting eigenvalues for the solution iteration procedure (five values allowed for each case)
EGNSL(I)		array containing final eigenvalues obtained for each starting eigenvalue
EM(I)		integer data block used to define initial associated transfer matrices corresponding to initial section end conditions - in BARRAY specifies free end conditions - in SARRAY specifies cantilevered end conditions (similar in concept to EE(I))
EPS(I)	{ \bar{E} } and { \bar{F} }	used as both correction eigenvector column and trial eigenvector forcing function column array
EYEX		flapwise bending stiffness of the torque tube pitch control structure spur about its local coordinate system y-axis, lb-ft ²

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
EZEX		edgewise bending stiffness of the torque tube pitch control structure spur about its local coordinate system z-axis, lb-ft ²
FAB(I)		same concept as for CTB(I) except that the contributions specified are for the pitch bearing discontinuity equations for an articulated blade, for the forces acting on the control rod end of the pitch arm in the rotating hub coordinate system y-axis direction for a flexstrap blade, and for the deflection of the control rod end of the pitch arm in the rotating hub coordinate system y-axis direction for a bearingless blade
FAB1(I), FAB3(I), FAB4(I)		arrays containing the elements required to denote the blade torque, flap, and lead-lag discontinuity contributions, respectively, to the pitch bearing discontinuity equations for an articulated blade (similar in concept to CTB1(I), CTB3(I), and CTB4(I))
FACC, FACL		value of DFAC on iteration just completed and previous iteration, respectively
FACT		correction factor to account for a change in DFAC on successive iterations, required for proper interpolation to obtain the next trial eigenvalue
FGRA		gravitational acceleration constant, ft/sec - identical to AG

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
FLB(I)		same concept as for CTB(I) except that the contributions specified are for the flap discontinuity equations for an articulated blade and for the forces acting on the control rod end of the pitch arm in the rotating hub coordinate system z-axis direction for a flexstrap or bearingless blade
FLB1(I), FLB3(I), FLB4(I)		arrays containing the elements required to denote the blade torque, pitch bearing, and lead-lag discontinuity contributions, respectively, to the flap discontinuity equations for an articulated blade (similar in concept to CTB1(I), CTB2(I), and CTB4(I))
FORCE(I)	$\{\bar{F}\}, \{\bar{\epsilon}\}$	trial eigenvectors forcing function column array in SOLVE which on solution is converted to the correction eigenvector column array
FSB(I)		same concept as for CTB(I) except that the contributions specified are for the lead-lag discontinuity equations for an articulated blade and for the deflection of the control rod end of the pitch arm in the rotating hub coordinate system z-axis direction for a bearingless blade (not involved with flexstrap blade representation)

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
FSB1(I), FSB2(I), FSB3(I)		array containing the elements required to denote the blade torque, pitch bearing, and flap discontinuity contributions, respectively, to the lead-lag discontinuity equations for an articulated blade (similar in concept to CTB1(I), CTB2(I), and CTB3(I))
FVEL		helicopter velocity with respect to a stationary reference coordinate system, ft/sec
FVLA, FVLP, FVLT		helicopter velocity in the x, z, and y axis directions of the fixed hub coordinate system, respectively, ft/sec
GCTCP		component of gravitational acceleration acting perpendicular to the rotor disk plane, ft/sec ²
GP, GT	ϕ_g and θ_g	two angles that specify orientation of the gravity vector relative to the fixed hub coordinate system, rad
IALP		integer data block whose six values specify the rows of the associated transfer matrices at the shaft-hub interface that correspond to slope and deflection definitions
IBW		internal control variable allowing use of blockage (MTAB=7) or simplified NACA 0012 linear aerodynamics (MTAB=6) for the support structure

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
ICHG (I)		operational integer array utilized in triangularization of the final governing matrix
ICOL, IROW		number of the column and row, respectively, of the final governing matrix to be switched with the last column and row of the matrix for determination of the correction eigenvector (ICOL=NHC*NRIFC+NORM and IROW=NHC*NRIFC+IREM)
IF, IG		number of frequency and stability steps, respectively, to be used in the search option solution method
ILL		index denoting aerodynamic load point number for use of azimuthal and radially variable induced velocity field
IPCT		number denoting the percentage of a 1-percent increment of the starting trial eigenvalue to be added to obtain the second trial eigenvalue
IPU		control parameter which controls punched shape vector output (0-no punched output, 1-punched output)
IREM		the number of the row in the set of unshifted equations of the final governing matrix which is to be removed for obtaining the correction eigenvector (see definition of IROW)

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
IROW1 (I)		integer data block whose six values specify the rows of the associated transfer matrices at shaft-hub shaft interface that correspond to force and moment definitions
IROW3 (I)		integer data block whose two values specify the rows of the associated transfer matrices at the shaft-hub interface that correspond to the blade flapwise and edge-wise force definitions
ISER		control parameter that specifies whether aerodynamic coefficient data corresponding to a data table number is to be inputted in the standard tabular form, C-81 tabular form, or a series representation form (0-standard tabular data, 81- C-81 tabular data, 1-series data)
ITI		iteration index with ITI equal to zero on the first iteration
KSML	k	integer denoting the frequency shift of the equations and Laplace transform variables
M1-M7		section control parameters where M1, M2, M3, M4, M5, M6, and M7 correspond to mass and inertia, bend, aerodynamic, elastic, rigid offset, torsional stiffness, and elastic restraint lumped-parameter characteristics of a section, respectively (the section control parameter associated with a lumped-parameter characteristic is 0 if characteristic

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
		is not to be included, 1 if included in a blade section, and 2 if included in a support structure section) Note: M7 must = 0 if MFLEX = 0
MAER		aerodynamic control parameter (0-no aerodynamics, 1-blade and/or support structure aerodynamics included)
MAXN	Nmax	highest azimuthal (spatial) harmonic of control system ring displacement to be included
MCON	Mr	control parameter for the inclusion of the effects of the pitch control structure restraint unknowns on the flexstrap or bearingless blade behavior (0-these effects not included, 1-these effects included)
MCT,MFEA, MFLAP,MLEL	Mct,Mfea, Mflap,Mlel	control torque, pitch bearing, flap, and lead-lag discontinuity control parameters for an articulated blade, respectively (0-discontinuity not included, 1-discontinuity included)
MCTY		control parameter used in conjunction with MFLEX=1 to denote whether a flexstrap or bearingless rotor is of interest (0-flexstrap, 1-bearingless)
MER	Mer	number of repeatable significant figures required for convergence of the iteration procedure for each starting eigenvalue

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
MFASB		number of elements required in storing discontinuity contributions to the discontinuity equations (e.g., CTB1(I))
MFLEX		blade type control parameter (0-articulated, 1-flexstrap or bearingless)
MFUS		support structure control parameter (0-no support structure, 1-support structure is included)
MODE		internal iteration control parameter (0-iteration procedure to continue, 1-mode shapes to be determined for solution eigenvalue)
MSC		control system collective spring-damper unit control parameter (0-no unit, 1-collective spring-damper unit included)
MTAB		number of the aerodynamic data table to be used for an aerodynamic section
MXCPK		number of elements required in the associated blade transfer matrix storage arrays B(I) and BSAVE(I)
MXCPL		number of combinations of k and n values required (MXSMI*MXSMI)
MXCPM		number of elements in an individual associated transfer matrix array (72)
MXCSB		number of elements in an individual discontinuity column array (12)

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
MXFAB		number of elements required in the arrays used for storing the blade tip variable contributions to the discontinuity equations (i.e., CTB(I))
MXKQ		number of elements required to store the n-related discontinuity column arrays for a particular value of k
MXQ		number of equations in the final governing matrix (MXQ x MXQ)
MXSMB		number of elements required in the arrays used for storing the discontinuity column arrays (e.g., SMLB(I))
MXSMI		number of sets of k-frequency-shifted equations (2*NHC+1)
MXT2P1		number of control system equations for a particular value of k
MXTKN		number of elements in TKN(I)
NAMA		number of blade aerodynamic intermediate terms stored for each blade section having aerodynamic lumped-parameter characteristics
NAS	NAS	number of azimuthal steps to be used for the blade aerodynamic harmonic computations
NB	Nb	number of blades unless the blade phasing assumption is used (NBP≠0) where NB must be equal to 1

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
NBC		number specifying type of rotor hub conditions (0-cantilevered, 1-gimballed, 2-teetering)
NBP		number of blades when the blade phasing option is used (otherwise equal to zero)
NBSEC	NS	number of blade sections (cannot be greater than 35)
NCACF, NCACG		internal logic control parameters that specify application of the transfer matrix procedures to SMLF(I) and SMLG(I), respectively, after the pitch control structure restraint application point is crossed for a bearingless blade
NCCT, NCFFA, NCFLP		internal logic control parameters that specify application of transfer matrix procedures to the control torque, pitch bearing, and flap discontinuity column arrays (SMLB(I), SMLC(I), SMLD(I)), respectively, after the respective discontinuities are crossed for an articulated blade; specifies application of the transfer matrix procedures to the SMLB(I), SMLC(I), and SMLD(I) arrays after the pitch control structure restraint application point is crossed for a flexstrap or bearingless blade

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
NCLEL		internal logic control parameter which specifies application of the transfer matrix procedures to SMLE(I) after the lead-lag discontinuity is crossed for an articulated blade or the pitch control structure restraint application point is crossed for a bearingless blade
NCLS (I, J, K)		storage array for the number of angles of attack for each Mach number for each type of aerodynamic coefficient for each aerodynamic table number
NCOLS		number of blade tip variable unknowns for a particular value of k (6)
NCON	Nr	blade section number of section at which the pitch control structure restraint is applied for a flexstrap or bearingless blade
NCT, NFEA, NFLAP, NLEL	Nct, Nfea, Nflap, Nlel	blade section number just inboard of which the control torque, pitch bearing, flap, and lead-lag discontinuities, respectively, are applied for an articulated rotor (Note: if MFEA=0, NFEA still denotes the location at and outboard of which the collective pitch is added to the sectional twist distribution)
NDAT		number of elements in DAT(I) array, defined internally in SETUP as 2800

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
NDIM		number of rows dimensioned for the final governing matrix array as defined in DCMAT; must be altered if the dimensions of the arrays in this subroutine are altered
NEBC		number of elements required to store the n-related associated blade transfer matrices for a particular value of k
NEGN		number of starting eigenvalues to be considered for a model
NEIFC		number of elements required to represent the first column of submatrices in Equation (15) (0 if no control system)
NEISC		number of elements required to represent the second column of submatrices in Equation (15) (0 if no rotor support structure)
NEITC		number of elements required to represent the third column of submatrices in Equation (15)
NES	Nes	number of azimuthally located control system cyclic spring-damper unit supports
NESBC		number of elements required to store the n-related discontinuity column arrays for a particular value of k
NEX		must have same value as IREM in the present solution technique

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
NFFB		control parameter specifying boundary conditions at the first section of the support structure (0-cantilevered to ground, 1-free)
NFPI		NHC+1
NFSEC	NSF	number of support structure sections (NFSEC+NBSEC cannot be greater than 10)
NFUS		number of support structure-related governing equations for a particular value of k (6*MFUS)
NHC	Nf	maximum number of harmonics included for interharmonic coupling
NIT		maximum number of iterations, in addition to the initial iteration, which are allowed for each starting eigenvalue
NMACH(I,J)		storage array for the number of Mach numbers for each type of aerodynamic coefficient for each aerodynamic table number
NOGNS		number of starting eigenvalues obtained from use of the search option procedure or set equal to NEGN
NORM		the number of the column in the set of the (n=0) related columns of the final governing matrix which is to be removed for obtaining the correction eigenvector (corresponds to the row number of the unknown in the unshifted solution eigenvector which is normalized to unity; see definition of ICOL)

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
NPD		control parameter for the determination of the disk plane blade shape vectors and support structure shape vectors relative to the support structure fixed coordinate system in both complex variable and polar form. (0-shape vectors only with respect to local axis systems, 1-shape vectors calculated and outputted relative to the additional coordinate systems in complex variable and polar form)
NPRS		control parameter for output of control parameter input in labelled form (0-no output, 1-output)
NPS		number denoting the type of blade phasing to be considered (-1 backward cyclic mode, 0-umbrella mode, 1-forward cyclic mode, 2-reactionless mode)
NRBD		number of blade-related governing equations including blade discontinuity equations for each blade for a particular value of k
NRIFC		number of rows (or columns) in the matrix represented by TKN(I)
NSCH		control parameter for use of search option solution method (0-no search option use, 1-search option used)

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
NSK		control parameter for use of support structure aerodynamics (0-no support structure aerodynamics, 1-support structure aerodynamics included if MAER≠0)
NSP		control parameter for use of control system representation (0-no control system equations, 1-control system equations included)
NSY		number of element in Y(I) (100 by definition in SETUP)
NTAB		number of aerodynamic tables (or series form representations) to be inputted
NTOT		total number of lumped-parameter sections (NBSEC+NFSEC) and cannot exceed 40
NVI		control parameter for use of radial and azimuthally variant induced velocity distribution (0-section input induced velocity used, 1-radial and azimuthally variant distribution used)
OM,OM1	Ω	rotor rotational speed, rad/sec
OM2	Ω^2	square of the rotor rotational speed, $\text{rad}^2/\text{sec}^2$
P(I)		storage array for all blade and support structure section data with 50 elements to each section

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
PENOM(I)		array containing each final frequency obtained, in non-dimensional form, for each starting eigenvalue
PFC		pitch-flap coupling factor representing pitch-flap coupling for an articulated blade, if this coupling is not represented by blade modeling (see definition of input Loc. 9 in the Description of Input Data section of this report)
PHIC(I)	$\phi_{c,m}$	one of two angles defining the orientation of the mth flex-strap or bearingless blade control rod with respect to the blade's rotating hub coordinate system, rad
PHIM(I)	ϕ_m	angular position of the mth blade relative to the reference blade position (azimuthal location of fixed hub coordinate system x-axis) at time equal to zero, rad
PHIX(I,J,K), PHIY(I,J,K), PHIZ(I,J,K)	$(\overline{\phi x}_m)_k^i, (\overline{\phi y}_m)_k^i,$ $(\overline{\phi z}_m)_k^i$	arrays containing the shifted and unshifted torsional, flapwise, and edgewise bending slopes, respectively, for each section of each blade
PHIX(I,J), PHIY(I,J), PHIZ(I,J)	$(\overline{\phi x}_s)_k^i, (\overline{\phi y}_s)_k^i,$ $(\overline{\phi z}_s)_k^i$	arrays containing the shifted and unshifted torsional, lateral, and vertical bending slopes, respectively, for each support structure section
PHOTT(I)		array containing length, stiffness, and orientation parameters associated with the flex-strap pitch arm (MCTY=0) or with the torque tube (MCTY=1)

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
PHWTT (I)		array containing length, stiffness and orientation parameters associated with the pitch arm of the torque tube pitch control structure (MCTY=1)
PINO	π	pi, 3.141592654
PLC		pitch-lag coupling factor representing pitch-lag coupling for an articulated blade, if this coupling is not represented by blade modeling (see definition of input Loc. 10 in the Description of Input Data section of this report)
QBAR (I)		temporary storage array of 12 elements for the shifted and unshifted support structure state variable vectors at a section as each is determined
R (I)	$\left[\bar{R} \right]_m^i$	array representing a blade or support structure rigid offset transfer matrix in string mode by row
R (I)		operational complex variable array used in BARRAY and SARRAY which takes on the values of complex variable transfer matrix arrays generated in transfer matrix subroutines, except for the blade aerodynamic transfer matrices
REMC, REML		value of the determinant on the iteration just completed and the previous iteration, respectively

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
RREAL(I)		operational real variable array used in BARRAY and SARARRAY which takes on the values of real variable transfer matrix arrays generated in transfer matrix subroutines
S(I)	$\left[\begin{array}{c} \bar{s} \\ k \end{array} \right]_s^i$	array representing all k-subscripted associated support structure transfer matrices at a section where each matrix is stored in string mode by column and the order of storage of the matrices is for k from -Nf to +Nf
SA(I,J)		used to represent coordinate system transformation arrays for a bend lumped-parameter characteristic and also used in the determination of terms associated with the representation of the flexstrap or torque tube restraints
SAT(I,J)		used to represent a coordinate system transformation array for the next section outboard of a section having a bend lumped-parameter characteristic (multiplication of this array by the SA(I,J) array of the next inboard section with a change in orientation provides the intermediate terms required for a bend transfer matrix) and also used in the determination of terms associated with the representation of the torque tube restraints

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
SD (I)		storage array for the Y(I) arrays of each blade and support structure section
SDCO		structural damping coefficient factor
SDVEL		velocity of sound, ft/sec
SHAPE (I)		temporary storage array of 12 elements for the shifted and unshifted blade state variable vectors at a section of a blade as each is determined
SI (I)	ω	array of the frequency components of the starting eigenvalues, rad/sec
SK (I)	$\left[\overline{SK}_k \right]_m^i$	array representing a blade or support structure torsional spring-damper transfer matrix in string mode by row
SMLA (I)	a_m	effective torque arm length of the mth blade pitch arm for an articulated blade, positive if pitch arm is aft of blade shear center, ft
SMLB (I), SMLC (I), SMLD (I), SMLE (I), SMLF (F), SMLG (G)	$\left\{ \overline{b}_{k,n} \right\}_m^i, \left\{ \overline{c}_{k,n} \right\}_m^i$ $\left\{ \overline{d}_{k,n} \right\}_m^i, \left\{ \overline{e}_{k,n} \right\}_m^i$ $\left\{ \overline{f}_{k,n} \right\}_m^i, \left\{ \overline{g}_{k,n} \right\}_m^i$	arrays representing all k- and n-subscripted associated blade discontinuity vectors corresponding to the contributions of specific discontinuity unknowns, where each type of column array is stored in its respective array in the same manner as the associated blade transfer matrices are stored in B(I)

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
SMLBSV(I), SMLCSV(I), SMLDSV(I), SMLESV(I), SMLFSV(I), SMLGSV(I)		storage arrays for SMLB(I), SMLC(I), SMLD(I), SMLE(I), SMLF(I), and SMLG(I), respectively, at an aerodynamic characteristic; required to properly transfer across this characteristic due to inter-harmonic coupling effects
SN(I,J)		array containing terms of the form $\sin L\phi_m$
SNA(I,J)		array containing terms of the form $\sin L\chi_j$
SPSI(I,J)		array containing terms of the form $\sin(L(K-1)2\pi/NAS)$
SR(I)	σ	array of the stability components of the starting eigenvalues, rad/sec
STF,STG		stepsize of frequency, rad/sec, and stability (growth rate), rad/sec, respectively, to be used if the search option method is to be used
SWEI	EI	control system ring local bending stiffness about a radial axis perpendicular to its circumferential axis, lb-ft ²
SWGJ	GJ	control system ring local torsional stiffness about its circumferential axis, lb-ft ²
SWM	M	mass of the control system ring, lb-sec ² /ft
SWR	R	control system ring radius, ft

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
T(I)		operational array for passing an associated transfer matrix or discontinuity column into a matrix multiplication subroutine and returning the resultant matrix
TAU(I)	τ_m	damping retardation time of the mth articulated blade control rod spring-damper representation, sec
TC(I)		temporary array containing intermediate aerodynamic terms at an aerodynamic load point for use in the generation of AMA(I)
TEA(I,J,K)	$(\bar{T}_m)_k^i$	array containing the shifted and unshifted torque for each section of each blade
TEA(I,J)	$(\bar{T}_s)_k^i$	array containing the shifted and unshifted torque for each support structure section
TEMPN, TEMVY, TEMVZ		temporary parameters for the steady axial, edgewise, and flapwise forces, respectively, acting on the blade and which are updated as each of the section lumped-parameter characteristics are crossed
TEMPT, TEMMY, TEMMZ		temporary parameters for the steady torque, flapwise bending moment, and edgewise bending moment, respectively, acting on the blade and which are updated as each of the section lumped-parameter characteristics are crossed
THC		real part of Theodorsen's coefficient for unsteady aerodynamics

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
THCK		real part of Theodorsen's coefficient for unsteady aerodynamics (identical to THC)
THEC (I)	$\theta_{c,m}$	one of two angles defining the orientation of the mth flex-strap blade control rod with respect to its reference rotating coordinate system, rad
THLD (I)		resulting associated transfer matrix or discontinuity column obtained in multiplication subroutines which is passed back to the calling subroutine by T(I)
TKN	$\left[\bar{T}_{k,n} \right]$	array representing the contribution of all n-frequency-shifted unknowns to all k-frequency-shifted governing equations for a particular value of k and n where placement of data is in string mode by column
TORFLX	1/kd	reciprocal of drive shaft torsional stiffness, rad/(ft-lb)
UBX (I, J, K), UBY (I, J, K), UBZ (I, J, K)	$(\bar{u}x_m)_k^i, (\bar{u}y_m)_k^i,$ $(\bar{u}z_m)_k^i$	arrays containing the shifted and unshifted radial, edgewise, and flapwise deflections, respectively, for each section of each blade
UBX (I, J), UBY (I, J), UBZ (I, J)	$(\bar{u}x_s)_k^i, (\bar{u}y_s)_k^i,$ $(\bar{u}z_s)_k^i$	arrays containing the shifted and unshifted axial, vertical, and lateral deflections, respectively, for each support structure section
V (I, J)		radially and azimuthally variant induced velocity distribution array defined by input, ft/sec

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
VELIK		induced velocity at an aerodynamic load point (radially and azimuthally located) that may be defined by use of $V(I,J)$, if $NVI=1$; or the blade section inputted induced velocities, ft/sec
VLLX,VLLY, VLLZ		calculated velocities of an aerodynamic load point with respect to the local blade rotating coordinate system x, y, and z axes, respectively, ft/sec
VY(I,J,K), VZ(I,J,K)	$(\overline{VY}_m)_k^i, (\overline{VZ}_m)_k^i$	arrays containing the shifted and unshifted edgewise and flapwise shear forces, respectively, for each section of each blade
VY(I,J) VZ(I,J)	$(\overline{VY}_s)_k^i, (\overline{VZ}_s)_k^i$	arrays containing the shifted and unshifted vertical and lateral shear forces, respectively, for each support structure
VZERO		calculated resultant velocity acting on an aerodynamic load point, ft/sec
W(I)	$\left[\overline{W}_k \right]_s^i$	array representing the support structure mass transfer matrix in string mode by row
X(I)		operational array in DCMAT, required for solution of the correction eigenvector
Y(I)		storage array for the intermediate terms associated with a section and required for construction section transfer matrices

<u>Computer Name</u>	<u>Algebraic Symbol</u>	<u>Definition or Description</u>
Y(I)		array in DCMAT corresponding to FORCE(I)
YKM(I,J)		temporary storage array for terms required in specifying the contribution of the control system unknowns to the blade-related governing equations
ZETA	ζ	one of two angles which specify orientation of the helicopter velocity vector with respect to the fixed hub coordinate system, rad

USER'S GUIDE

This section of the report describes in detail the input data and its format required for execution of the HASTA computer program and the output which results. As an aid to possible redimensioning of the program for specific applications, the dimension sizes of the arrays contained in the program are presented. A listing of a sample input deck for a flexstrap rotor in forward flight with aerodynamics included is provided. Some of the output produced by use of this input deck is also presented.

DIMENSIONS OF COMPUTER ARRAY VARIABLES

The HASTA program in its present form is restricted to the investigation of the aeroelastic stability behavior of coupled rotor/helicopter systems in which the rotor is taken to consist of an arbitrary number of identical blades equally spaced azimuthally. This restriction is due to the dimensions of some of the larger arrays being based upon the use of the blade phasing option, which provides an adequate representation of system behavior without the penalty of higher core requirements and running time. The program coding allows the consideration of identical blades arbitrarily spaced azimuthally if the restrictive array sizes are increased, and provides the basis for program modifications required to represent nonidentical blades. The program coding also allows for use of up to five different aerodynamic data tables. However, the present associated storage array dimensions are such that only one aerodynamic data table can be inputted. To facilitate possible program redimensioning, the dimensions of the HASTA program arrays are presented in terms of program or defining variables in addition to their present (maximum) numerical dimensions. The variables necessary to specify the required array dimension sizes are defined below in a tabular form.

<u>Defining Variable</u>	<u>Definition</u>
MAFA	16*NAEF
MAMB	NAMA*NAERB
MAXN	highest harmonic of control system to be considered

Defining VariableDefinition

MCON	1 - effects of restraint related unknowns included for flexstrap or bearingless blades, 0 - effects of restraint-related unknowns not included for flexstrap or bearingless blades
MCT	1 - control torque discontinuity considered for articulated blades, 0 - no control torque discontinuity
MCTY	1 - bearingless blade, 0 - flexstrap blade
MFASB	NCSB*MXCPL
MFEA	1 - pitch bearing discontinuity included for articulated blades, 0 - no pitch bearing discontinuity
MFLAP	1 - flap hinge discontinuity included for articulated blades, 0 - flap hinge discontinuity not included
MFLEX	1 - flexstrap or bearingless blades, 0 - articulated blades
MFUS	1 - support structure is included, 0 - no support structure
MLEL	1 - lead-lag hinge discontinuity included for articulated blades, 0 - lead-lag hinge discontinuity not included
MXCPK	MXCPL*MXCPM
MXCPL	MXSMI*MXSMI
MXCPM	12*NCOLS
MXFAB	NCOLS*MXCPL
MXQ	NRIFC*MXSMI

<u>Defining Variable</u>	<u>Definition</u>
MXSMI	$2*NHC+1$
MXTKN	$NRIFC*NRIFC$
MXT2P1	$(2*MAXN+1)*NSP$
NAEF	number of support structure sections having aerodynamic characteristics included
NAERB	number of radial aerodynamic load points (number of blade sections having aerodynamic characteristics included)
NAMA	$34*MXSMI$
NAS	maximum number of azimuthal locations considered for a radial aerodynamic load point in one rotor revolution
NB	number of blades on the rotor; Note: $NB=1$ if $NBP \neq 0$
NBP	0 - if blade phasing option not to be used; equals the number of blades on rotor if blade phasing option is to be used
NBSEC	number of blade sections
NCOLS	6
NCSB	1
NDAT**	$200+50*(NBSEC+NFSEC)+NAERB*NAS$
NEBC	$MXCPM*MXSMI$
NESMI	$MXSMI$ or $MXT2P1$; whichever is larger in value

**NDAT is equivalent to the number of variables in labelled common PLOAD and is set equal to 2800 in sub-routine SETUP

<u>Defining Variable</u>	<u>Definition</u>
NEI	number of starting eigenvalues
NEIT	number of crossover eigenvalues to be stored during use of search option
NES	number of control system elastic supports
NESBC	$12 * MXSMI$
NFSEC	number of support structure sections
NHC	highest harmonic of system motion to be included for interharmonic coupling - corresponds to Nf
NRBD	$NCOLS + 3 * MCON * MFLEX * (1 + MCTY) + (MCT + MFEA + MFLAP + MLEL) * (1 - MFLEX)$
NRIFC	$MXT2P1 + 6 * MFUS + NB * NRBD$
NSD	$NSY * (NBSEC + NFSEC)$
NSP	1 - control system to be included, 0 - no control system
NSY	100
NTAB	number of aerodynamic data sets
NTAL	maximum number of angle of attack values for a Mach number
NTC	maximum number of series representation coefficients
NTM	maximum number of Mach number values for an aerodynamic coefficient table

The dimensions of the arrays involved in the computer program are given by subroutines as follows: where the arrays of each subroutine are grouped by type of variable (i.e.,

integer, real, or complex), an asterisk (*) following an array variable is used to denote that the variable is in double precision; the letter p following an array variable denotes that the array is involved in the labelled common PLOAD such that any change in dimension size requires modification of the data input format.

<u>Variable</u>		<u>Dimensions</u>	<u>Maximum Size</u>
<u>ACOEFF</u>			
NCC		NTAB,3	1,3
NCLS	(Integer)	NTM,NTAB,3	11,1,3
NMACH		NTAB,3	1,3
ALPHS		NTM,NTAL,NTAB,3	11,50,1,3
AMACH	(Real)	NTM,NTAB,3	11,1,3
CCS		NTC,NTAB,3	25,1,3
CLDM		NTM,NTAL,NTAB,3	11,50,1,3
<u>BAERO</u>			
ACE p		NES	4
AKC p		NB	6
AKE p		NES	4
ATAU p		NB	6
BJE p		NES	4
BLI2 p		NB	6
BTAU p		NB	6
CLESP p		NB	6
CPSI		NAS, 2*MXSMI	24,6
DBS p		NB	6
ECHI p		NES	4
P p	(Real)	(NBSEC+NFSEC)*50	2000
PAC p		NES	4
PAK p		NES	4
PHIC p		NB	6
PHIM p		NB	6
PHOTT p		8	8
PHWTT p		8	8
SD		NSD	4000
SI p		NEI	5
SMA p		NB	6
SPSI		NAS, 2*MXSMI	24,6
SR p		NEI	5

<u>Variable</u>		<u>Dimensions</u>	<u>Maximum Size</u>
STKC p		NB	6
TAC p		NES	4
TAK p		NES	4
TC*		34	34
THEC p		NB	6
V p		NAERB,NAS	25,24
Y		NSY	100
AMA*	(Complex)	MAMB	2550

Note: The AMA array is presently initialized to zero in this subroutine by a use of a loop from 1 to 2550, such that if the dimensions of this array are altered, the upper limit of this loop must be modified.

BARRAY

EM	(Integer)	6	6
CX*		75	75
RREAL*	(Real)	144	144
SD		NSD	4000
Y		NSY	100
ALM*		MXQ	63
AMY		NB,NBSEC,MXSMI	1,35,3
AMZ		NB,NBSEC,MXSMI	1,35,3
AN		NB,NBSEC,MXSMI	1,35,3
B*		MXCPK	648
BSAVE*		MXCPK	648
C*		144	144
CTB*		MXFAB	54
CTB1*		MFASB	9
CTB2*		MFASB	9
CTB3*		MFASB	9
CTB4*		MFASB	9
D*		144	144
EPS*		MXQ	63
FAB*		MXFAB	54
FAB1*		MFASB	9
FAB3*		MFASB	9
FAB3*		MFASB	9
FLB*	(Complex)	MXFAB	54

<u>Variable</u>	<u>Dimensions</u>	<u>Maximum Size</u>
FLB1*	MFASB	9
FLB2*	MFASB	9
FLB4*	MFASB	9
FSB*	MXFAB	54
FSB1*	MFASB	9
FSB2*	MFASB	9
FSB3*	MFASB	9
PHIX	NB,NBSEC,MXSMI	1,35,3
PHIY	NB,NBSEC,MXSMI	1,35,3
PHIZ	NB,NBSEC,MXSMI	1,35,3
R*	144	144
SHAPE*	12	12
SMLB*	NESBC	108
SMLBSV*	NESBC	108
SMLC*	NESBC	108
SMLCSV*	NESBC	108
SMLD*	NESBC	108
SMLDSV*	NESBC	108
SMLE*	NESBC	108
SMLESV*	NESBC	108
SMLF*	NESBC	108
SMLFSV*	NESBC	108
SMLG*	NESBC	108
SMLGSV*	NESBC	108
T*	72	72
TEA	NB,NBSEC,MXSMI	1,35,3
UBX	NB,NBSEC,MXSMI	1,35,3
UBY	NB,NBSEC,MXSMI	1,35,3
UBZ	NB,NBSEC,MXSMI	1,35,3
VY	NB,NBSEC,MXSMI	1,35,3
VZ	NB,NBSEC,MXSMI	1,35,3
<u>BEND</u>		
Y	NSY	100
B*	144	144
<u>BLARO</u>		
AC*	144	144
AD*	(Complex) 144	144
AMA*	MAMB	2550

<u>Variable</u>		<u>Dimensions</u>	<u>Maximum Size</u>
<u>BMASS</u>			
Y	(Real)	NSY	100
A*	(Complex)	144	144
<u>DCMAT</u>			
ICHG	(Integer)	MXQ	63
A*		MXQ, MXQ	63, 63
X*	(Complex)	MXQ	63
Y*		MXQ	63

Note: When dimensions are changed, change the statement defining NDIM in this subroutine such that NDIM is equal to the magnitude of the variable MXQ, i.e., if MXQ is to be 135, put NDIM=135.

<u>ELAST</u>			
CX*		75	75
E*	(Real)	144	144
Y		NSY	100

<u>EPSOLN</u>			
ALM*		MXQ	63
EPS*	(Complex)	MXQ	63
FTEMP*		NRIFC	21
TKN*		MXTKN	441

<u>EXCHI</u>			
CSA		NES, MXT2P1	4, 24
SNA	(Real)	NES, MXT2P1	4, 24

<u>EXPON</u>			
CS		NB, NESMI	4, 6
SN	(Real)	NB, NESMI	4, 6

<u>FAERO</u>			
ACE p		NES	4

<u>Variable</u>		<u>Dimensions</u>	<u>Maximum Size</u>
AFA		MAFA	240
AKC p		NB	6
AKE p		NES	4
ATAU p		NB	6
BJE p		NES	4
BL12 p		NB	6
BTAU p		NB	6
CLESP p		NB	6
DBS p		NB	6
ECHI p		NES	4
P p	(Real)	(NBSEC+NFSEC)*50	2000
PAC p		NES	4
PAK p		NES	4
PHIC p		NB	6
PHIM p		NB	6
PHOTT p		8	8
PHWTT p		8	8
SD		NSD	4000
SI p		NEI	5
SMA p		NB	6
SR p		NEI	5
STKC p		NB	6
TAC p		NES	4
TAK p		NES	4
THEC p		NB	6
V p		NAERB,NAS	25,24
Y		NSY	100

Note: The AFA array is presently initialized to zero in this subroutine by a use of a loop from 1 to 240, such that if the dimensions of this array are altered, the upper limit of this loop must be modified.

FMASS

Y	(Real)	NSY	100
W*	(Complex)	144	144

FUARO

AFA		MAFA	240
Y	(Real)	NSY	100

<u>Variable</u>		<u>Dimensions</u>	<u>Maximum Size</u>
C*	(Complex)	144	144
	<u>LOADIN</u>		
T		5	5
X	(Real)	1	1
	<u>MAIN</u>		
NBIGMD	(Integer)	NEI	5
AC		NES	4
ACE p		NES	4
ACP		NES	4
ACT		NES	4
AFA		MAFA	240
AK		NES	4
AKC p		NB	6
AKCI		NB	6
AKE p		NES	4
AKP		NES	4
AKT		NES	4
AL		6,NB	6,6
ALSTH		NB	6
AL12		NB	6
AMKC		NB	6
ARRAY*		3	3
ATAU p		NB	6
BJ	(Real)	NES	4
BJE p		NES	4
BL12 p		NB	6
BTAU p		NB	6
CLESP p		NB	6
CONV		NEI	5
CPSI		NAS, 2*MXSMI	24,6
CS		NB,NESMI	4,6
CSA		NES,MXT2P1	4,24
CTAU		NB	6
CX*		75	75
DBS p		NB	6
DETR		25	25
DETR2		25	25
DMS		NB	6
ECHI p		NES	4

<u>Variable</u>	<u>Dimensions</u>	<u>Maximum Size</u>
P p	(NBSEC+NFSEC) *50	2000
PAC p	NES	4
PAK p	NES	4
PENOM	NEI	5
PHIC p	NB	6
PHIM p	NB	6
PHOTT p	8	8
PHWTT p	8	8
SD	NSD	4000
SI p	NEI	5
SMA p	NB	6
SMLA	NB	6
SN	NB,NESMI	4,6
SNA	NES,MXT2P1	4,24
SPSI	NAS, 2*MXSMI	24,6
SR p	NEI	5
STKC p	NB	6
TAC p	NES	4
TAK p	NES	4
TAU	NB	6
THEC p	NB	6
V p	NAERB,NAS	25,24
Y	NSY	100
ALM*	MXQ	63
AMA*	MAMB	2550
EC	25	25
EC2 (Complex)	25	25
EGNS	NEI	5
EGNSL*	NEI	5
EPS*	MXQ	63
<u>MLCC2</u>		
R*	144	144
T* (Complex)	84	84
THLD*	84	84
<u>MLRC2</u>		
R* (Real)	144	144
T*	84	84
THLD* (Complex)	84	84

<u>Variable</u>		<u>Dimensions</u>	<u>Maximum Size</u>
<u>RIGID</u>			
R*		144	144
Y	(Real)	NSY	100
<u>SARRAY</u>			
EE		6	6
EM	(Integer)	6	6
RREAL*		144	144
SD	(Real)	NSD	4000
Y		NSY	100
ALM*		MXQ	63
AMY		NFSEC, MXSMI	25, 3
AMZ		NFSEC, MXSMI	25, 3
AN		NFSEC, MXSMI	25, 3
EPS*		MXQ	63
PHIX		NFSEC, MXSMI	25, 3
PHIY		NFSEC, MXSMI	25, 3
PHIZ	(Complex)	NFSEC, MXSMI	25, 3
QBAR*		12	12
R*		144	144
S*		NEBC	216
T*		72	72
TEA		NFSEC, MXSMI	25, 3
UBX		NFSEC, MXSMI	25, 3
UBY		NFSEC, MXSMI	25, 3
UBZ		NFSEC, MXSMI	25, 3
VY		NFSEC, MXSMI	25, 3
VZ		NFSEC, MXSMI	25, 3
<u>SECPAR</u>			
ACE p		NES	4
AKC p		NB	6
AKE p		NES	4
ATAU p		NB	6
BC		3	3
BJE p	(Real)	NES	4
BL12 p		NB	6
BTAU p		NB	6
CLESP p		NB	6
CX*		75	75

<u>Variable</u>	<u>Dimensions</u>	<u>Maximum Size</u>
DBS p	NB	6
ECHI p	NES	4
P p	(NBSEC+NFSEC) *50	2000
PAC p	NES	4
PAK p	NES	4
PHIC p	NB	6
PHIM p	NB	6
PHOTT p	8	8
PHWTT p	8	8
SA	3,3	3,3
SAT	3,3	3,3
SBT	3,3	3,3
SD	NSD	4000
SI p	NEI	5
SMA p	NB	6
SR p	NEI	5
STKC p	NB	6
TAC p	NES	4
TAK p	NES	4
THEC p	NB	6
V p	NAERB ,NAS	25,24
Y	NSY	100

SETUP

AC	NES	4
ACE p	NES	4
ACP	NES	4
ACT	NES	4
AK	NES	4
AKC p	NB	6
AKCI	NB	6
AKE p	NES	4
AKP	NES	4
AKT	NES	4
AL	6,NB	6,6
ALSTH	NB	6
AL12	NB	6
AMKC	NB	6
ATAU p	NB	6
BJ (Real)	NES	4
BJE p	NES	4
BL12 p	NB	6
BTAU p	NB	6

<u>Variable</u>	<u>Dimensions</u>	<u>Maximum Size</u>
CLESP p	NB	6
CPSI	NAS,2*MXSMI	24,6
**CS	NB,NESMI	4,6
**CSA	NES,MXT2P1	4,24
CTAU	NB	6
DAT p	NDAT	2800
DBS p	NB	6
DMS	NB	6
ECHI p	NES	4
P p	(NBSEC+NFSEC) *50	2000
PAC p	NES	4
PAK p	NES	4
PHIC p	NB	6
PHIM p	NB	6
PHOTT p	8	8
PHWTT p	8	8
SI p	NEI	5
SMA p	NB	6
SMLA	NB	6
**SN	NB,NESMI	4,6
**SNA	NES,MXT2P1	4,24
SPSI	NAS,2*MXSMI	24,6
SR p	NEI	5
STKC p	NB	6
TAC p	NES	4
TAK p	NES	4
TAU	NB	6
THEC p	NB	6
V p	NAERB,NAS	25,24
EGNS	NEI	5
AMA* (Complex)	MAMB	2550

**Note: At present the program in subroutine SETUP determines values of CS and SN arrays for values of the second index up to and including a value of six, and values of CSA and SNA arrays for values of the second index up to and including a value of 24 due to internal numerical definitions of the upper values on two loops. Dimensional modifications involving these arrays requires the internal numerical definitions of the upper values on these two loops to be such that the values for these arrays must be defined at least for values of the second index up to that specified by the dimensional definitions of these arrays.

<u>Variable</u>		<u>Dimensions</u>	<u>Maximum Size</u>
<u>SOLVE</u>			
DB*		MXQ, MXQ	63, 63
EPS*	(Complex)	MXQ	63
FORCE*		MXQ	63
<u>STIFF</u>			
Y	(Real)	NSY	100
SK*	(Complex)	144	144
<u>SUBMA</u>			
IROW1	(Integer)	6	6
S*	(Complex)	NEBC	216
TKN*		MXTKN	441
<u>SUBMB</u>			
IALP		6	6
IROWC	(Integer)	2	2
IROW3		2	2
CX*	(Real)	75	75
B*		MXCPK	648
CTB*		MXFAB	54
CTB1*		MFASB	9
CTB2*		MFASB	9
CTB3*		MFASB	9
CTB4*		MFASB	9
FAB*		MXFAB	54
FAB1*		MFASB	9
FAB3*	(Complex)	MFASB	9
FAB4*		MFASB	9
FLB*		MXFAB	54
FLB1*		MFASB	9
FLB2*		MFASB	9
FLB4*		MFASB	9
FSB*		MXFAB	54
FSB1*		MFASB	9
FSB2*		MFASB	9
FSB3*		MFASB	9

<u>Variable</u>		<u>Dimensions</u>	<u>Maximum Size</u>
SMLB*		NESBC	108
SMLC*		NESBC	108
SMLD*		NESBC	108
SMLE*		NESBC	108
SMLF*		NESBC	108
SMLG*		NESBC	108
TKN*		MXTKN	441
	<u>SUBMF</u>		
TKN*	(Complex)	MXTKN	441
	<u>SUBMG</u>		
AC		NES	4
AK		NES	4
AKCI		NB	6
AL		6, NB	6, 6
ALSTH		NB	6
AL12		NB	6
AL436		NB	6
AL536		NB	6
AMKC	(Real)	NB	6
BJ		NES	4
CTAU		NB	6
CX*		75	75
DMS		NB	6
SMLA		NB	6
TAU		NB	6
B*		MXCPK	648
CTB*		MXFAB	54
CTB1*		MFASB	9
CTB2*		MFASB	9
CTB3*		MFASB	9
CTB4*		MFASB	9
FAB*		MXFAB	54
FAB1*		MFASB	9
FAB3*		MFASB	9
FAB4*		MFASB	9
FLB*	(Complex)	MXFAB	54
FLB1*		MFASB	9
FLB2*		MFASB	9

<u>Variable</u>		<u>Dimensions</u>	<u>Maximum Size</u>
FLB4*		MFASB	54
FSB*		MXFAB	9
FSB1		MFASB	9
FSB2		MFASB	9
FSB3		MFASB	9
SMLB*		NESBC	108
SMLC*		NESBC	108
SMLD*		NEGBC	108
SMLE*		NESBC	108
SMLF*		NESBC	108
SMLG*		NESBC	108
TKN*		MXTKN	441
YKM*		MXSMI,NB	3,6
<u>SWA</u>			
AC		NES	4
AK		NES	4
AKCI		NB	6
BJ	(Real)	NES	4
DMS		NB	6
SMLA		NB	6
TAU		NB	6
TKN*		MXTKN	441
YKM*	(Complex)	MXSMI,NB	3,4
<u>SWAPS</u>			
DB*	(Complex)	MXQ,MXQ	63,63
<u>SWB</u>			
AC		NES	4
AK		NES	4
AKC1		NB	6
BJ		NES	4
CX*	(Real)	75	75
DMS		NB	6
SMLA		NB	6
TAU		NB	6
B*		MXCPK	648
CTB*		MXFAB	54

<u>Variable</u>		<u>Dimensions</u>	<u>Maximum Size</u>
CTB1*		MFASB	9
CTB2*		MFASB	9
CTB3*		MFASB	9
CTB4*		MFASB	9
FAB*		MXFAB	54
FAB1*		MFASB	9
FAB3*		MFASB	9
FAB4*		MFASB	9
FLB*	(Complex)	MXFAB	54
FLB1*		MFASB	9
FLB2*		MFASB	9
FLB4*		MFASB	9
FSB*		MXFAB	54
FSB1*		MFASB	9
FSB2*		MFASB	9
FSB3*		MFASB	9
S*		NEBC	216
SMLB*		NESBC	108
SMLC*		NESBC	108
SMLD*		NESBC	108
SMLE*		NESBC	108
SMLF*		NESBC	108
SMLG*		NESBC	108
TKN*		MXTKN	441

TABLU

NCC		NTA, 3	5, 3
NCLS	(Integer)	NTM, NTA, 3	10, 5, 3
NMACH		NTA, 3	5, 3
ALPHS		NTM, NTAL, NTA, 3	10, 15, 5, 3
AMACH	(Real)	NTM, NTA, 3	10, 5, 3
CCS		NTC, NTA, 3	25, 5, 3
CLDM		NTM, NTAL, NTA, 3	10, 15, 5, 3

TKNS

TKN*	(Complex)	MXTKN	441
------	-----------	-------	-----

ULN

AC		NES	4
----	--	-----	---

<u>Variable</u>		<u>Dimensions</u>	<u>Maximum Size</u>
AK		NES	4
AKCI		NB	6
BJ	(Real)	NES	4
DMS		NB	6
SMLA		NB	6
TAU		NB	6
<u>XNLQ</u>			
AC		NES	4
ACP		NES	4
ACT		NES	4
AK		NES	4
AKCI		NB	6
AKP	(Real)	NES	4
AKT		NES	4
BJ		NES	4
DMS		NB	6
SMLA		NB	6
TAU		NB	6
<u>ZLN</u>			
AC		NES	4
ACP		NES	4
ACT		NES	4
AK		NES	4
AKCI		NB	6
AKP	(Real)	NES	4
AKT		NES	4
BJ		NES	4
DMS		NB	6
SMLA		NB	6
TAU		NB	6
<u>ZTEGI</u>			
B*		MXCPK	648
CTB*		MXFAB	54
CTB1*		MFASB	9
CTB2*		MFASB	9
CTB3*		MFASB	9
CTB4*		MFASB	9
FAB*		MXFAB	54

<u>Variable</u>		<u>Dimensions</u>	<u>Maximum Size</u>
FAB1*		MFASB	9
FAB3*		MFASB	9
FAB4*		MFASB	9
FLB*	(Complex)	MXFAB	54
FLB1*		MFASB	9
FLB2*		MFASB	9
FLB4*		MFASB	9
FSB*		MXFAB	54
FSB1*		MFASB	9
FSB2*		MFASB	9
FSB3*		MFASB	9
SMLB*		NESBC	108
SMLC*		NESBC	108
SMLD*		NESBC	108
SMLE*		NESBC	108
SMLF*		NESBC	108
SMLG*		NESBC	108
TKN*		MXTKN	441

DESCRIPTION OF INPUT DATA

The input required for the computer program consists of values for (1) system model and program logic parameters and, if required, (2) aerodynamic coefficient tables. The system model and program logic parameters are inputted directly into a single storage array equivalent to the labeled common PLOAD by specification of their associated array locations and input values. In that all elements of this storage array are initialized to zero when the program is entered, only values for parameters which are non-zero in value need to be read in for the first set of input to be used in a program run. For convenience, the storage array is divided into three parts containing values for specific types of input parameters. The array location numbers 2 - 200 correspond to control parameter input consisting of values for program logic control, flight condition, rotor configuration, and control system parameters. The array location numbers 201 - 2200 correspond to blade and support structure section input consisting of values for the blade and support structure sectional parameters. Array location numbers 2201 - 2800 correspond to values defining a radially and azimuthally varying induced velocity distribution if this program option is to be used.

The value of parameters represented by the storage array are inputted by use of punched data cards with an I4, 11, 5E14.7 format. The first four columns contain the right-justified array location number specifying the array location in which the first parametric value on the card is to be stored. The fifth column contains the number of parametric values to be read from the data card (up to five values may be read). The remaining columns, consisting of five fields of 14 columns each, contain consecutive array-located parametric values. Thus, columns 6 - 19 contain the parametric value to be stored in the array location specified on the card, columns 20 - 33 contain the parametric value to be stored in the next array location, and so on until the specified number of values to be read are contained on the card. For example, the input of a card punched 101620.01bbbbbbbbb1.57, where b denotes a blank space, will set the value of array location 1016 and array location 1017 to 0.01 and 1.57, respectively.

Data cards of the same format are used to denote the end of storage array input for an input case and to terminate the

program execution. In particular, a card with an 8 punched in the fifth column must be placed at the end of the storage array input for each case and a card with a 9 punched in the fifth column must be placed at the end of all input for a run. These cards, in combination with storage array input and aerodynamic coefficient tables (if required), are used to construct the input for a run. The input for running more than one case in a run is exemplified by the input for two cases per run which has the form:

1. Storage array input for Case I
2. Card with 8 punched in fifth column
3. Aerodynamic tables (if required)
4. Storage array input for Case II
5. Card with 8 punched in fifth column
6. Aerodynamic tables (if required)
7. Card with 9 punched in fifth column

The aerodynamic coefficient tables must be included in the input for a case if in the same case the storage array location corresponding to the parameter MAER is to be non-zero.

Since the storage array is only initialized to zero prior to reading storage array input for the first input case, the input for a subsequent case needs only to consist of the data necessary to update the values already in the storage array. For example, if the input parameter values for Case I and Case II of a run are identical except for the helicopter velocity, then the storage array input for Case II consists of a single card, which when read updates the value of the storage array location associated with helicopter velocity to that desired. The remainder of the storage array for Case II is automatically taken as that for Case I. This input procedure of updating the existing storage array reduces drastically the number of input cards that have to be read for additional cases in the same run. The value of a storage array location can also be defined more than once in the storage array input for a case with the last values read being that used for the case. This input capability allows the use of a basic storage array input deck in combination with a correction (update) input deck to construct the input deck for the initial case of a run. The correction deck is added to the end of the basic storage array input deck.

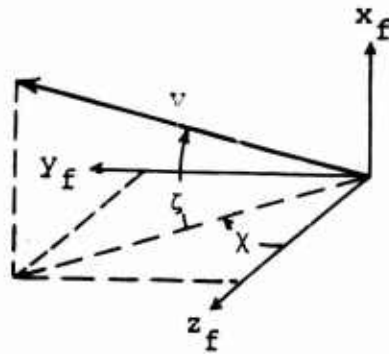
The program storage array input parameters and their associated array location numbers, and also the aerodynamic coefficient table input, are discussed in detail in the following sections. Prior to the generation of input data for a coupled rotor/helicopter system of interest, the inexperienced HASTA program user should be cognizant of the information presented in this report regarding the capabilities of the program, the coupled rotor/helicopter models, and the representational coordinate systems including definitions of various system parameters.

Control Parameter Input

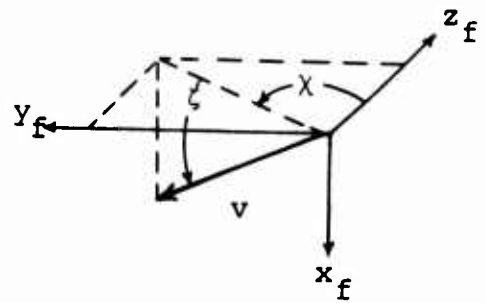
As mentioned previously, control parameter input information is stored in the first 200 elements of the storage array corresponding to array location numbers 2 - 200. The definition of the specific input information stored in each of these 200 storage array elements is provided on this and the following pages where Loc. is used to denote array location number. Table 2 at the end of this section lists the array location numbers for the storage array elements defined herein and the computer program variables to which they correspond. It is to be noted that although an element of the input data storage array may correspond to a system parameter which is an integer, the value inputted must be provided in decimal form.

- Loc. 2: Rotational speed of the rotor, rad/sec (must always be positive in value)
- Loc. 3: Velocity of sound, ft/sec, is used to determine the Mach number at aerodynamic stations when aerodynamics are included
- Loc. 4: Air density, $\text{lb-sec}^2/\text{ft}^4$, is used when aerodynamics are included
- Loc. 5: Helicopter velocity, ft/sec
Note: This is the total velocity at which the craft is moving with respect to a stationary reference coordinate system

Loc. 6 Two angles χ and ζ , respectively, in radians,
 and
 Loc. 7 velocity vector relative to the fixed rotor hub
 coordinate system as shown in the following
 sketch for two directions of rotation:

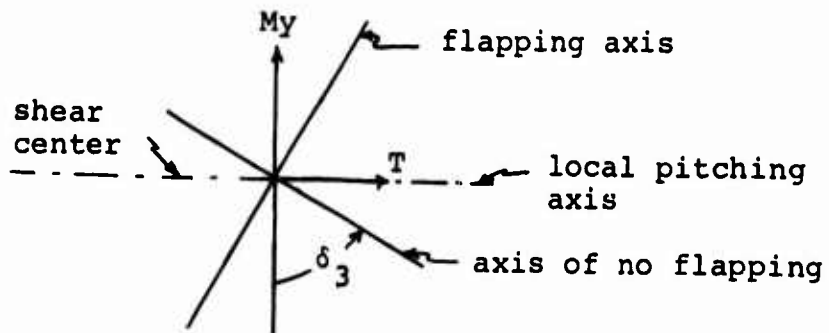


Counterclockwise
Rotation

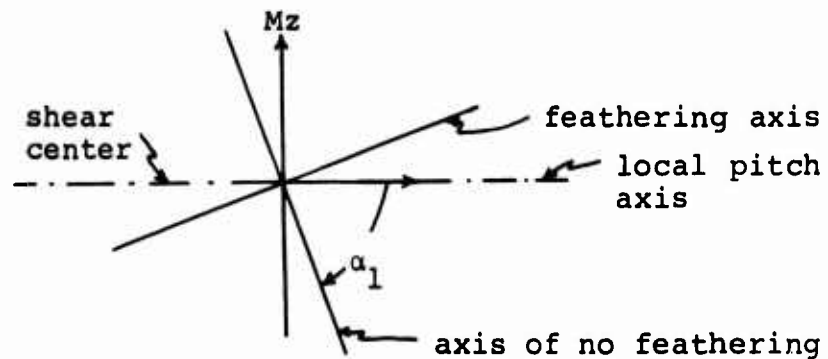


Clockwise Rotation

- Loc. 8: Real part of Theodorsen's coefficient for unsteady aerodynamics, normally taken equal to 1.0
- Loc. 9: Pitch-flap coupling factor representing pitch-flap coupling for an articulated blade with a flap hinge, if this coupling is not represented through blade modeling, defined as the $\cot \delta_3$ where δ_3 is the angle between the no-flapping axis and the line perpendicular to the local pitching axis at the flap hinge location as shown in the following sketch



Loc. 10: Pitch-lag coupling factor representing the pitch-lag coupling for an articulated blade with a pitch bearing, if this coupling is not represented through blade modeling, defined as the $\cot \alpha_1$ where α_1 is the angle between the no feathering axis and local pitching axis at the pitch bearing location as shown in the following sketch



Loc. 11: Collective pitch of the blades, if not included in the blade section pitch angles specified in the sectional input (Locs. seventeen), in radians (the collective pitch value is added to the section pitch angle of each blade section for a flexstrap or bearingless blade, and added to the sectional pitch angle at each blade section outboard of, and including, the blade section at the inboard end of which the pitch bearing is located for an articulated blade).

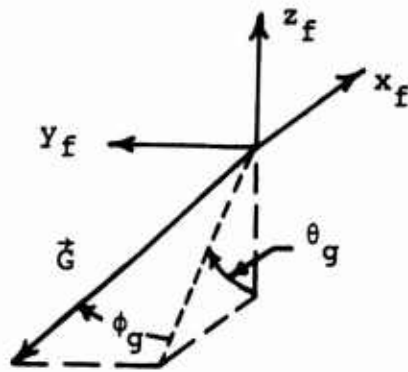
Note: If collective pitch is inputted, care must be taken to insure that the total geometric pitch distribution is that desired. Also, the condition that the blade total geometric pitch must be zero at the inboard end of the blade (shaft-hub interface) and at the pitch control structure-blade attachment point, if a flexstrap or bearingless blade is involved, must be satisfied.

Loc. 12: Gravitational acceleration constant, ft/sec^2 , inputted if gravitational effects are to be included

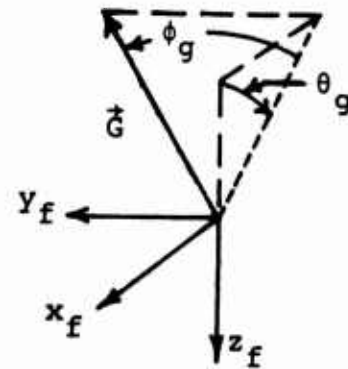
Note: As discussed in the section pertaining

to coordinate systems, the fixed support structure coordinate system y-axis orientation is dependent on the direction of the gravity field vector if gravitational effects are included.

Loc. 13 and Loc. 14: Two angles θ_g and ϕ_g , respectively, in radians, which specify the orientation of the gravity vector relative to the fixed rotor hub coordinate system and are required if gravity effects are to be included. They are depicted in the following sketch for two directions of rotation



Counterclockwise Rotation



Clockwise Rotation

Loc. 15: Logic control parameter, which equals 1.0 if the blade shape vectors relative to the rotor disk plane and the support structure vectors relative to the fixed support structure coordinate system are to be determined in complex variable and polar form; equals 0.0 if these shape vectors are not to be determined

Loc. 16: Logic control parameter, which equals 1.0 if support structure aerodynamics are to be included and equals 0.0 if support structure aerodynamics are not to be included

Note: For support structure aerodynamics to be included Loc. 26 (MAER) cannot be 0.0.

- Loc. 17: Number of rotor blades (NB) if the blade phasing option is not to be used, in which case Loc. 18 (NBP) equals 0.0; equals 1.0 if the blade phasing option is to be used
 Important Note: The present program is dimensioned such that only the blade phasing option can be used to represent rotors having more than one blade. Thus, Loc. 17 must always equal 1.0 for the present version of the HASTA program.
- Loc. 18: Number of rotor blades (NBP) if the blade phasing option is to be used, in which case Loc. 17 (NB) must equal 1.0; equals 0.0 if the blade phasing option is not to be used
 Note: The blade phasing option takes advantage of defined relationships between the motions of the blades, which are assumed identical and equally spaced azimuthally such that core requirements and running time are significantly reduced. To use this option Loc. 18 must be defined as greater than or equal to 2.0 and the assumed relationship of blade motions specified by Loc. 65 (NPS).
- Loc. 19: Number of blade sections (NBSEC) representing the blade with a maximum value of 35.0 allowed; NBSEC+NFSEC (Loc. 20) cannot exceed 40.0
- Loc. 20: Number of support structure sections (NFSEC) representing the support structure with a maximum value of 15.0 allowed; NBSEC (Loc. 19) + NFSEC cannot exceed 40.0
- Loc. 21: Logic control parameter (MFLAP), which equals 1.0 if a flap hinge is to be included for an articulated blade and equals 0.0 if a flap hinge is not to be included
- Loc. 22: Logic control parameter (MFEA), which equals 1.0 if a pitch bearing is to be included for an articulated blade and equals 0.0 if a pitch bearing is not to be included
- Loc. 23: Logic control parameter (MCT), which equals 1.0 if control torque is to be applied for an articulated blade and equals 0.0 if control torque is not to be applied

- Loc. 24: Logic control parameter (MCON), which equals 1.0 if the restraint applied to the flexstrap or bearingless blade includes the effects of the pitch control structure-related discontinuity unknowns and equals 0.0 if these effects are not to be included
 Note: Normally would be 1.0 if Loc. 25 (MFLEX)=1.0.
- Loc. 25: Logic control parameter (MFLEX), which equals 1.0 if the rotor consists of flexstrap or bearingless blades and equals 0.0 if the rotor consists of articulated blades
- Loc. 26: Logic control parameter (MAER), which equals 1.0 if blade and/or support structure aerodynamics are to be included and equals 0.0 if aerodynamics are not to be included
 Note: For support structure aerodynamics to be included, Loc. 16 (NSK) must also be 1.0.
- Loc. 27: Logic control parameter (NFFB), which equals 1.0 if the first section of the support structure lumped-parameter model is to be considered as unrestrained (free) and equals 0.0 if the first section of the support structure is to be cantilevered (to ground)
- Loc. 28: Number of blade section (NFLAP), counting from the blade tip inboard, at the inboard end of which the flap hinge is to be located for an articulated blade if a flap hinge is to be included; i.e., Loc. 21 (MFLAP)≠0.0
- Loc. 29: Number of blade section (NFEA), counting from the blade tip inboard, at the inboard end of which the pitch bearing is to be located for an articulated blade if a pitch bearing is to be included; i.e., Loc. 22 (MFEA)≠0.0
 Note: Even if Loc. 22 (MFEA)=0.0, the collective pitch (Loc. 11) is added to the sectional twist distribution of an articulated blade at and outboard of the blade section specified by NFEA.

- Loc. 30: Number of blade section (NCT), counting from the blade tip inboard, at the inboard end of which the control torque application point is to be located for an articulated blade if control torque is to be included; i.e., Loc. 23 (MCT)≠0.0
- Loc. 31: Number of blade section (NCON), counting from the blade tip inboard, at which the pitch control structure restraint is to be applied to a flex-strap or bearingless blade if the effects of pitch control structure discontinuity unknowns are to be included; i.e., Loc. 24 (MCON)≠0.0
- Loc. 32: Logic control parameter (NBC), which specifies the type of blade root conditions at the shaft-rotor hub interface such that a value of 0.0 denotes cantilevered blade root conditions as in the case of an articulated rotor, a value of 1.0 denotes gimballed rotor blade root conditions, and a value of 2.0 denotes teetering rotor blade root conditions
- Loc. 33: Logic control parameter (MFUS), which equals 1.0 if a support structure is to be included and equals 0.0 if a support structure is not to be included
Note: If a support structure is to be included Loc. 20 (NFSEC) must be 1.0 or greater.
- Loc. 34: Number of aerodynamic load point azimuthal steps in one blade revolution (NAS), used for the blade aerodynamics harmonic computations
Note: NAS should be an integer multiple of the number of blades on the rotor and must be greater than 2*NHC (Loc. 35) for the necessary number of harmonics to be determined. However, in that the representative capability of the harmonic analysis is dependent on the number of azimuthal steps, a value of at least 12.0 is recommended.
- Loc. 35: Maximum number of harmonics above and below the primary frequency to be included for interharmonic coupling (NHC)
Important Note: The program is presently dimensional such that NHC cannot exceed 1.0. The

program should use $NHC = 1.0$ when the system has a forward velocity and/or an elastic support structure and/or an elastically supported control system. For cases where interharmonic coupling is not possible (simple axisymmetric problem NHC may be set equal to 0.0

Loc. 36: Logic control parameter (NVI), which equals 1.0 if the radial and azimuthal induced velocity distribution as inputted in Locs. 2201 - 2800 is to be used, and equals 0.0 if the induced velocity distribution is to be taken from the blade section data (Locs. 201 - 2200) for each blade section and is assumed to be constant with azimuth

Loc. 37: Logic control parameter (NSP), which equals 1.0 if control system equations are to be included and equals 0.0 if control system equations are not to be included

**Loc. 38: Maximum number of harmonics of the control system ring deflection above and below the primary frequency to be included for interharmonic coupling (MAXN)

Note: If the control system ring is rigid, the value for MAXN need not be more than 1.0 . If the control system ring is rigid and supported isotopically, MAXN should still be taken as 1.0 since cyclic deflection of the ring can still occur. Present program dimensions restrict MAXN to 1.0 unless there is either no support structure included (i.e., Loc. 33 (MFUS)= 0.0) or no blade discontinuities.

**Loc. 39: Number of concentrated cyclic spring-damper units supporting the control system ring (NES) (azimuthal and radial positions and stiffness and damping characteristics for these supports defined in Locs. 89 - 96 and 109 - 132)

Note: Present program dimensions restrict the maximum number of spring-damper support units to no more than 4.

- **Loc. 40: Logic control parameter (MSC), which equals 1.0 if a collective spring-damper unit supporting the control system base plate is to be included, in which case the stiffness and damping characteristics of the spring-damper unit are defined in Locs. 133 - 134; equals 0.0 if a collective spring-damper unit is not to be included
- **Loc. 41: Mass of the control system ring, lb-sec²/ft
- **Loc. 42: Radius of the control system ring, ft
- **Loc. 43: Local torsional stiffness of a segment of the control system ring about its circumferential axis, lb-ft²
- **Loc. 44: Local bending stiffness of a segment of the control system ring about a radially oriented axis, lb-ft²

*Note: If Loc. 37 (NSP) equals 0.0 a control system is not being included. Therefore, the control system information in Locs. 38 - 44 need not be included.

- Loc. 45: Effective drive shaft torsional flexibility, rad/(ft-lb), defined as the reciprocal of kd where kd is the torsional stiffness of an effective torsional spring
- Loc. 46: Logic control parameter (NPRS), which equals 1.0 if control parameter input is to be printed as output and equals 0.0 if control parameter input is not to be printed
- Loc. 47: Not used
- Loc. 48: Number of starting eigenvalues (NEGN) to be considered for a particular case where a maximum of five starting values is allowed for a given case
- Loc. 49 thru
Loc. 53: Real part (growth rate, rad/sec) of the starting eigenvalues, where the number of array locations to be defined is specified by Loc. 48 (NEGN)
Note: As an example, if Loc. 48 is defined to be 2.0 only Loc. 49 and 50 need to be defined.

- Loc. 54 thru 58: Imaginary part (frequency, rad/sec) of the starting eigenvalues where the number of array locations to be defined is specified by Loc. 48 (NEGN)
 Note: As an example, if Loc. 48 is to be 2.0, only Loc. 54 and 55 need to be defined.
- Loc. 59: Percentage of a 1-percent increment of starting eigenvalue (IPCT) which is to be used to compute the second trial eigenvalue to be used in the next step in the iteration procedure (recommended value is 50.0, which results in a second trial eigenvalue of 1.005 times the starting eigenvalue)
- Loc. 60: Maximum number of iterations to be performed for each starting eigenvalue (NIT)
- Loc. 61: Number of repeatable significant figures (MER) required in consecutive trial eigenvalues for satisfactory convergence of the iteration procedure for each starting eigenvalue (a value of 7.0 is recommended)
 Note: Whether convergence has actually been obtained can be noted on comparison of the last two eigenvalues and their determinant values and the degree to which the shaft-rotor hub interface conditions are satisfied.
- Loc. 62: Logic control parameter (NORM), which equals the number of the unknown in the unshifted unknowns vector $\{\bar{q}_0\}^*$ counting from the top, which is to be used as the basis for normalizing the mode shape (i.e., normalized to unity)
 Note: Mode shapes may be normalized to any unknown expected to be non-zero in value. For normalization based on a specific blade tip unknown being set to unity,

$$\text{NORM} = \text{NSP} * (2 * \text{MAXN} + 1) + 6 * \text{MFUS} + \text{the location number of the specific unknown in the blade tip vector.}$$
 In this expression the first and second terms represent the number of control system and support structure unknowns in the unshifted unknowns vector, respectively. NSP is contained in Loc. 37, MAXN is contained in Loc. 38, and MFUS is contained in Loc. 33. From the above

expression, values of NORM required for normalization based on various blade tip deflections for various system configurations are as shown in Table 1.

TABLE 1.			
VALUES OF NORM FOR NORMALIZATION TO VARIOUS BLADE TIP DEFLECTIONS FOR VARIOUS SYSTEM MODELS			
System Model	Flapwise Deflection	Edgewise Deflection	Torsional Deflection
MFUS=0, NSP=0	5	3	2
MFUS=1, NSP=0	11	9	8
MFUS=0, NSP=1*	6	4	3
MFUS=1, NSP=1*	12	10	9

*MAXN taken as zero in value

If MAXN has been taken to have a value of one, the numbers in the last two rows of Table 1 would be increased by two. For normalization based on a support structure tip unknown the expression for NORM is similar to that for a blade tip unknown except that the term $6 \cdot MFUS$ is not included. Thus, if the system model includes a support structure, MFUS=1, and has a free end, NFFB (Loc. 27)=1, the values of NORM for setting a support structure tip deflection to unity are obtainable by subtracting 6 from the value of NORM required for the same type of blade tip deflection. If the support structure is cantilevered (NFFB=0) the support structure tip unknowns are force and moment unknowns to which the mode shapes can be normalized if desired. It should be noted that the support structure mode shapes, when reactionless blade behavior is assumed, can only be obtained if normalization is based on a support structure unknown.

Loc. 63: Logic control parameter (IREM), which equals the number of the unshifted equation to be removed from the matrix procedure to obtain correction values for updating the value of the unknowns (with one unknown being set to unity there is

one equation too many)

Note: The value IREM is dependent on the value of NORM (Loc. 62), particularly in the case of uncoupled modes. Basically, the value to take for IREM corresponds to the location number in the unshifted unknowns column vector of an unknown other than that being normalized to, which will also be non-zero in value. For example, if normalization is based on setting the blade tip flapwise deflection to unity, the blade tip flapwise slope will be non-zero such that IREM is taken as $NORM + 1$. Thus, on this basis, for tip flapwise or edgewise deflections (either blade or support structure) of unity use $IREM = NORM + 1$ and for a blade tip torsional deflection of unity use $IREM = NORM + 5$ if blade discontinuity equations are involved. For a blade tip torsional deflection of unity without discontinuity equations or a support structure tip torsion deflection of unity use $IREM = NORM - 1$. Relationships between NORM and IREM for normalization to support structure tip force or moment unknowns, if the support structure is cantilevered, can be obtained in a similar manner.

Note: If the system model has sufficient modal coupling, all modes except those corresponding to reactionless blade behavior when a support structure is included may be obtained by normalizing the blade tip flapwise deflection to unity. In order to avoid singular matrix difficulties with an articulated rotor using the previous criteria for NORM and IREM, a stiff elastic blade section, even if of short length, should be modeled inboard of the most inboard hinge or bearing of an articulated blade.

Loc. 64: Logic control parameter (NEX), which has the same value as Loc. 63

Loc. 65: Logic control parameter (NPS), which denotes the type of blade phasing to be used where 0.0 specifies the blades to be in an umbrella or collective mode, 2.0 specifies a reactionless or scissors mode, 1.0 specifies a forward cyclic mode, and -1.0 specifies a backward cyclic mode
Note: The value of NPS is significant only

when the phasing option of the program is to be used; that is, when Loc. 18 (NBP) is non-zero.

Loc. 66: Logic control parameter (NSCH), which equals 1.0 if the search option is to be used for estimating the starting eigenvalues and equals 0.0 if the search option is not to be used
Note: If NSCH = 1, then the real and imaginary parts of the starting eigenvalue for the scanning procedure must be inputted in Loc. 49 and 54, respectively. These input values should represent the beginning of the range within which the starting eigenvalues for the iteration procedure are to be estimated.

**Loc. 67: Number of steps in growth rate to be considered if the search option is to be used

**Loc. 68: Number of steps in frequency to be considered if the search option is to be used (should not be greater than 25.0)

**Loc. 69: Stepsize of the growth rate increment if the search option is to be used

**Loc. 70: Stepsize of the frequency increment if the search option is to be used

**Note: If Loc. 66 (NSCH) = 0.0, the search option is not specified and the numbers in Locs. 67 - 70 need not be defined. If NSCH = 1, the ranges of growth rate and frequency should be selected such that no more than five starting eigenvalues are obtained.

Loc. 71 Angular position, ϕ_m , of the mth blade ahead
 thru
Loc. 76: of the reference blade position (fixed hub coordinate system x-axis) in the direction of rotor rotation at time = 0 for values of m from 1 to NB, sequentially, in radians
Note: For a four-bladed rotor with equal azimuthal blade spacing the values for Loc. 71 - 74 would be 0.0, $\pi/2$, π , and $3\pi/2$, respectively, assuming ϕ to be 0.0 as is the normal case. When the blade phasing option is being used Loc. 71 must be defined to be 0.0. It is to be noted that due to present program dimensions only one

blade can be considered unless the blade phasing option is used.

- Loc. 77 thru Loc. 82: Reciprocal of the axial stiffness of the control rod attached to the m th blade for values of m from 1 to NB, sequentially, ft/lb (inputted only for rotor systems having articulated type blades - for flexstrap bearingless rotors see Locs. 135 - 140)
- Loc. 83 thru Loc. 88: Damping coefficient to axial stiffness ratio of the form c/k associated with the control rod attached to the m th blade, τ_m , for values of m from 1 to NB, sequentially, sec (inputted only for rotor systems having articulated type blades - for flexstraps or bearingless rotors, see Locs. 141 - 146)
- **Loc. 89 thru Loc. 92: Torsional stiffness of the j th torsional spring-damper unit counteracting local lateral (out of plane) control system ring rotation, $k\theta_j$, for values of j from 1 to NES, sequentially, ft-lb/rad
Note: At the control system cyclic spring-damper support unit locations specified by Locs. 117 - 120, torsional spring-damper units attaching the control system ring to ground such that local lateral ring rotations are restrained can be considered. Specifically, each of these torsional spring-damper units counteracts out-of-plane rotation of the local segment of the ring at their point of attachment about a radially oriented axis thru their point of attachment.
- **Loc. 93 thru Loc. 96: Torsional damping coefficient of the j th torsional spring-damper unit counteracting local lateral control system ring rotation, $c\theta_j$, for values of j from 1 to NES, sequentially, ft-lb-sec/rad

**Note: If Loc. 37 (NSP) equals 0.0, a control system is not being included, so that the control system information in Locs. 89 - 96 need not be provided.

- Loc. 97 Effective torque arm length of the m th blade
thru pitch arm, a_m , for values of m from 1 to NB,
Loc. 102: sequentially, as defined by Equation (8);
positive if pitch arm is aft of blade shear
center, ft (required only for articulated blades)
- Loc. 103 Rigid offset distance of m th control rod attach-
thru ment point from the circumferential axis of the
Loc. 108: control system ring, d_m , for values of m from 1
to NB, sequentially, ft (positive if control
rod attachment point is located outward from the
ring circumferential axis)
- **Loc. 109 Torsional stiffness of the j th torsional spring-
thru damper unit counteracting local control system
Loc. 112: ring longitudinal (torsional) rotation, $k\phi_j$, for
values of j from 1 to NES, sequentially,
ft-lb/rad
Note: At the control system cyclic spring-damper
support unit locations specified by Locs. 117 -
120, torsional spring-damper units attaching the
control system ring to ground such that local
longitudinal ring rotations are restrained can
be considered. Specifically, each of these
torsional spring-damper units counteracts tor-
sional rotation of the local segment of the
ring at their point of attachment about the local
segment's circumferential axis.
- * *Loc. 113 Torsional damping coefficient of the j th tor-
thru sional spring-damper unit counteracting local
Loc. 116: control system ring longitudinal (torsional)
rotation, $c\phi_j$, for values of j from 1 to NES,
sequentially, ft-lb-sec/rad
- * *Loc. 117 Angle specifying azimuthal location of the j th
thru control system cyclic spring-damper unit ahead
Loc. 120: of the control system fixed coordinate system
x-axis in the direction of rotor rotation, χ_j ,
for values of j from 1 to NES, sequentially,
radians

- **Loc. 121 thru Loc. 124: Linear stiffness of the j th cyclic spring-damper unit supporting control system ring, k_j , for values of j from 1 to NES, sequentially, lbs/ft
- **Loc. 125 thru Loc. 128: Damping coefficient of the j th cyclic spring-damper unit supporting the control system ring, c_j , for values of j from 1 to NES, sequentially, lb-sec/ft
- **Loc. 129 thru Loc. 132: Rigid offset distance of the j th cyclic spring-damper unit attachment point from the circumferential axis of control system ring, b_j , for values of j from 1 to NES, sequentially, ft (positive if cyclic spring-damper unit attachment point is located inward from the ring circumferential axis)
- **Loc. 133: Linear stiffness of the collective spring-damper unit supporting the control system base plate, K , lb/ft (required only when Loc. 40 (MSC) = 1.0)
- **Loc. 134: Damping coefficient of the collective spring-damper unit supporting the control system base plate, C , lb-sec/ft (required only when Loc. 40 (MSC) = 1.0)

*Note: If Loc. 37 (NSP) equals 0.0, a control system is not being included such that the control system information in Locs. 93 - 134 is not required.

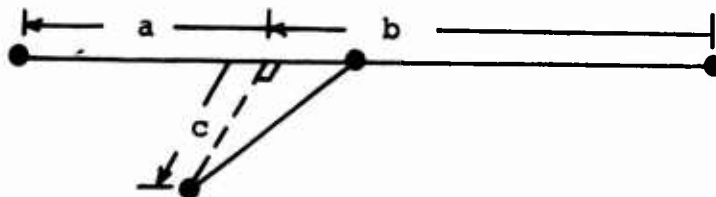
Loc. 135 thru Loc. 140: Reciprocal of the axial stiffness of the control rod attached to the m th blade for values of m from 1 to NB, sequentially, ft/lb (required for flexstrap and bearingless rotors - for rotors having articulated blades, see Loc. 77 - 82).

Loc. 141 thru Loc. 146: Damping coefficient of the spring-damper representation of control rod attached to the m th blade for values of m from 1 to NB, sequentially, lb-sec/ft (required for flexstrap and bearingless rotors-for rotors having articulated blades, see Locs. 83 - 88)

- Loc. 147 thru
Loc. 152: First of two angles, ϕ_c , (in radians) needed to define the orientation of the control rod of the m th flexstrap or bearingless blade with respect to its rotating hub coordinate system at the control rod-control system attachment point required for values of m from 1 to NB, sequentially, and depicted in Figure 16
- Loc. 153 thru
Loc. 158: Second of two angles, ϕ_c , (in radians) needed to define the orientation of the control rod of the m th flexstrap or bearingless blade with respect to its rotating hub coordinate system at the control rod-control system attachment point required for values of m from 1 to NB, sequentially, and depicted in Figure 16
- Loc. 159 thru
Loc. 164: Perpendicular distance (in feet) from the rotor disk plane to the control rod-control system attachment point associated with the m th flexstrap or bearingless blade, required for values of m from 1 to NB, sequentially; positive in value if the control rod-control system attachment point is located on the fixed hub coordinate system $-z$ -axis side of the rotor disk plane
- Loc. 165: Structural damping coefficient for the blade structural damping
- Loc. 166: Logic control parameter, which equals 1.0 if all shape vector information determined is to be outputted in punched form (in addition to printed form) and equals 0.0 if no punched shape vector output is to be generated
- Loc. 167: Fore and aft blade cyclic pitch, which equals the value of the blade cyclic pitch (in radians) when the blade is located in the fixed hub coordinate system $-y$ -axis direction, i.e., located aft
- Loc. 168: Lateral blade cyclic pitch, which equals the value of the blade cyclic pitch (in radians) when the blade is located in the fixed hub coordinate system x -axis direction; i.e., located in the reference blade position

- Loc. 169: Logical control parameter (MLEL), which equals 1.0 if a lead-lag hinge is to be included for an articulated blade and equals 0.0 if a lead-lag hinge is not to be included
- Loc. 170: Number of blade section (NLEL) counting from the blade tip inboard, at the inboard end of which the lead-lag hinge is to be located for an articulated blade if a lead-lag hinge is to be included; i.e., Loc. 169 (MLEL) \neq 0.0
- Loc. 171: Logical control parameter which equals 1.0 if the blade section mass and inertia-related damping effects on the system behavior are not to be included and equals 0.0 if these damping effects are to be included
Note: Normally a value of 0.0 would be used.
- Loc. 172: Logical control parameter (MCTY) used in conjunction with MFLEX (Loc. 25) = 1.0 such that when equal to 0.0 a flexstrap rotor system is to be considered and when equal to 1.0 a bearingless rotor system is to be considered
Note: When MFLEX (Loc. 25) = 1.0, MCON (Loc. 24) is normally defined to be 1.0
- Loc. 173: Length of torque tube pitch control structure spur or extension shaft, ft, which is required for a bearingless rotor system
- Loc. 174: Flapwise bending stiffness of the torque tube pitch control structure spur about its local coordinate system y-axis, lb-ft², which is required for a bearingless rotor system
- Loc. 175: Edgewise bending stiffness of the torque tube pitch control structure spur about its local coordinate system z-axis, lb-ft², which is required for a bearingless rotor system
- Loc. 176: Steady blade torque (COE1) at the pitch control structure-blade attachment point not removed by the pitch control structure, ft-lb (applicable for articulated blades not having a pitch bearing, for flexstrap blades, and for bearingless blades)

Loc. 177: An effective distance (COE2) which is used in conjunction with COE1 (Loc. 176) to define the steady flapwise shear force acting on the blade as a result of the steady blade torque removed by the pitch control structure, ft (applicable to same blades specified for Loc. 176)
 Note: To provide clarification to the definition of this system parameter the following discussion is provided. In the case of a torque tube pitch control structure lying in the rotor disk plane, as depicted in the following sketch, the value of COE2 can be defined as equal to $c(a+b)/a$ where, as depicted, c is the perpendicular distance from the control rod-pitch arm attachment point to the torque tube axis, a is the distance along the torque tube axis from the dropped perpendicular to the spur spherical bearing, and b is the distance along the torque tube axis from the dropped perpendicular to the torque tube-blade attachment point. This definition of COE2 is based on the torsional and



flapwise moment equilibrium equations about the torque tube axis and the spherical bearing, respectively. The above definition of COE2 is valid for a pitch arm oriented toward the blade leading edge if c is taken to have a negative value and for the dropped perpendicular intersecting the torque tube axis inboard of the spur spherical bearing if a is taken to be negative. For a torque tube pitch control structure not lying in the rotor disk plane the distances required can be determined from the projection of the three attachment points onto the disk plane.

In the case of an articulated or flexstrap rotor, a and b go to zero such that COE2 is equal to c.

- Loc. 178: Percentage of steady blade torque (COE3) in decimal form (i.e., less than 1.0) not removed by the pitch control structure in the case of an articulated blade having a pitch bearing (normally would be 0.0 in value for a frictionless pitch bearing)
- **Loc. 179: Length of the flexstrap blade pitch arm along its coordinate system x-axis if Loc. 172 (MCTY) = 0.0 and length of the torque tube (from blade attachment point to pitch arm attachment point) if Loc. 172 (MCTY) = 1.0, ft
- **Loc. 180: Product of Young's modulus and the average cross-sectional area (EA) of the flexstrap blade pitch arm if Loc. 172 (MCTY) = 0.0 and product of Young's modulus and the average cross-sectional area of the torque tube if Loc. 172 (MCTY) = 1.0, lb
- **Loc. 181: Flapwise bending stiffness of the flexstrap blade pitch arm about its coordinate system y-axis if Loc. 172 (MCTY) = 0.0 and of the torque tube about its coordinate system y-axis if Loc. 172 (MCTY) = 1.0, lb-ft²
- **Loc. 182: Edgewise bending stiffness of the flexstrap blade pitch arm about its coordinate system z-axis if Loc. 172 (MCTY) = 0.0 and of the torque tube about its coordinate system z-axis if Loc. 172 (MCTY) = 1.0, lb-ft²
- **Loc. 183 thru
Loc. 185: Three angles; α , β , and γ , respectively, (in radians) which define the orientation of the flexstrap blade pitch arm (Loc. 172 (MCTY) = 0.0) or torque tube (Loc. 172 (MCTY) = 1.0) relative to the rotating hub coordinate system as depicted in Figure 12

**Note: If a flexstrap or bearingless rotor is not to be considered the information of Locs. 179 - 185 is not needed.

- Loc. 186: Torsional stiffness (GJ) of the torque tube if a bearingless rotor is to be considered, lb-ft²
- Loc. 187: Length of bearingless blade pitch arm along its coordinate system x-axis, ft
- Loc. 188 Not used
thru
Loc. 190:
- Loc. 191 Three angles, α' , β' , and γ' , respectively, (in
thru radians) which define the orientation of the
Loc. 193: bearingless blade pitch arm relative to the torque tube coordinate system as depicted in Figure 13
- Loc. 194: Not used
- Loc. 195 Perpendicular distance (in feet) from the rotor
thru disk plane to the mth bearingless blade spheri-
Loc. 200: cal bearing-spur attachment point for values of m from 1 to NB, sequentially, and is positive in value if the spherical bearing is located on the fixed hub coordinate system -z-axis side of the rotor disk plane

TABLE 2.
CONTROL PARAMETER INPUT LOCATIONS
AND CORRESPONDING PROGRAM VARIABLES

Loc.	program symbol	Loc.	program symbol	Loc.	program symbol	Loc.	program symbol
0002	OM	0030	NCT	0064	NEX	0135-	AMKC(I)
0003	SDVEL	0031	NCON	0065	NPS	0140	I=1,NB
0004	AIRD	0032	NBC	0066	NSCH	0141-	CTAU(I)
0005	FVEL	0033	MFUS	0067	IG	0146	I=1,NB
0006	CHI	0034	NAS	0068	IF	0147-	PHIC(I)
0007	ZETA	0035	NHC	0069	STG	0152	I=1,NB
0008	THC	0036	NVI	0070	STF	0153-	THEC(I)
0009	PFC	0037	NSP	0071-	PHIM(I)	0158	I=1,NB
0010	PLC	0038	MAXN	0076	I=1,NB	0159-	AL12(I)
0011	COLL	0039	NES	0077-	AKCI(I)	0164	I=1,NB
0012	AG	0040	MSC	0082	I=1,NB	0165	SDCO
0013	GT	0041	SWM	0083-	TAU(I)	0166	PUNI
0014	GP	0042	SWR	0088	I=1,NB	0167	CYFA
0015	NPD	0043	SWGJ	0089-	AKT(J)	0168	CYLA
0016	NSK	0044	SWEI	0092	J=1,NES	0169	MLEL
0017	NB	0045	TORFLX	0093-	ACT(J)	0170	NLEL
0018	NBP	0046	NPRS	0096	J=1,NES	0171	DACO
0019	NBSEC	0047	not used	0097-	SMLA(I)	0172	MCTY
0020	NFSEC	0048	NEGN	0102	I=1,NB	0173	AEXT
0021	MFLAP	0049-	SR(I),	0103-	DMS(I)	0174	EYEX
0022	MFEA	0053	I=1,NEGN	0108	I=1,NB	0175	EZEX
0023	MCT	0054-	SI(I),	0109-	AKP(J)	0176	COE1
0024	MCON	0058	I=1,NEGN	0112	J=1,NES	0177	COE2
0025	MFLEX	0059	IPCT	0113-	ACP(J)	0178	COE3
0026	MAER	0060	NIT	0116	J=1,NES	0179-	PHOTT(I)
0027	NFFB	0061	MER	0117-	ECHI(J)	0186	I=1,8
0028	NFLAP	0062	NORM	0120	J=1,NES	0187-	PHWTT(I)
0029	NFEA	0063	IREM	0121-	AK(J)	0194	I=1,8
				0124	J=1,NES	0195-	ALSTH(I)
				0125-	AC(J)	0200	I-1,NB
				0128	J=1,NES		
				0129-	BJ(J)		
				0132	J=1,NES		
				0133	CAPK		
				0134	CAPC		

Blade/Support Structure Section Input

As mentioned earlier, location numbers 201 - 2200 contain section input for blade and support structure sections. Each section has 50 locations associated with it. Thus, locations 201 - 250 are associated with the first section, locations 251 - 300 with the second section, and so on. The maximum number of sections is 40 and the 40th section would be associated with locations 2151 - 2200. However, it is not necessary that this many sections be used and in such case the locations corresponding to unused sections would be left with values of 0.0 as initialized by the program.

The order in which the particular sections are taken is very important. Section number 1 is always at the blade tip, section number 2 is the next inboard section on the blade, and so on until the hub section is reached. The most inboard blade section corresponds to the rotor hub. The number of this section by definition is equal to NBSEC. The next section, number NBSEC+1, always is the first section of the support structure. This is the support structure section furthest from the rotor hub (at the support structure tip). This section is also allowed to have either free or cantilevered end conditions at its end most distant from the rotor hub. The next section, number NBSEC+2, is the next most distant section of the support structure, and so on. This numbering order continues until section number NBSEC + NFSEC is reached, which corresponds to the last section of the support structure. The total section data for all sections is placed in the storage array such that the section data is stored in the single array P(2000) of the labelled common PLOAD. Thus, each section is not denoted by an independent program name.

The definitions of the 50 input locations associated with each blade or support structure section are provided in the following discussion. In addition, for the more experienced HASTA program user these input locations and their corresponding mathematical symbols are given in Table 3 at the end of this discussion of section input data.

As described previously, each general section (blade or support structure) may consist of concentrated mass and

inertia, aerodynamics, torsional spring-damper units, an elastic section with centrifugal stiffening or localized elastic restraint (this localized elastic restraint is used for only one particular section for a flexstrap or bearingless rotor), bends in shear center axis and/or a rotation of local coordinates to account for blade twist and collective pitch, and rigid offsets in the shear center axis. Of the 50 locations provided for each section, the first seven are the section control indicators; M1, M2, M3, M4, M5, M6, and M7. Each of these control indicators specifies whether or not a particular transfer matrix is to be used for the section. The section control indicators and their related transfer matrices are as follows:

<u>Location</u>	<u>Parameter</u>	<u>Related Transfer Matrix</u>
one	M1	Concentrated Mass and Inertia
two	M2	Bend
three	M3	Aerodynamics
four	M4	Elastic Section
five	M5	Rigid Offset
six	M6	Torsional Stiffness
seven	M7	Elastic Restraint (for flexstrap or bearingless rotors only)

It is to be noted that the alpha location number used here is with regard to the position in the block of 50 locations given to each section. This convention will be followed throughout the remainder of this discussion. In using the control indicator a value of 0.0 indicates that the particular transfer matrix will not be used. For a blade section a control indicator value of 1.0 specifies that the particular transfer matrix is to be used. For a support structure section, a value of 2.0 specifies that the particular transfer matrix is to be used. For example, a blade section may have a concentrated mass and an elastic

length associated with it. Thus, M1 and M4 for that section would have a value of 1.0. If, in addition, the section includes aerodynamics, then M3 would be 1.0. The remaining values for M2, M5, M6, and M7 would be 0.0. For a support structure section with mass, an elastic length, and aerodynamics, the values of M1, M4, and M3 should be 2.0 and the values of M2, M5, M6, and M7 would be 0.0.

In developing the input values for the seven control indicators, there are two sets of special conditions. The first set of conditions is in regard to the use of M7. Only one blade section is allowed to have a non-zero value for M7 and all other control indicators for this section should be 0.0. Thus, a bend transfer matrix should be used in the next outboard section to align the local blade coordinate system of the section having non-zero M7 with the rotating hub coordinate system, as required in the program. A further condition is that the blade section having a non-zero value for M7 must be that specified by NCON (Loc. 31) if MCON (Loc. 24) equals 1.0. A support structure section is not allowed to have a non-zero value for M7. The second set of conditions is in regard to the first blade section (section number 1) and the first support structure section (section number NBSEC+1) when a support structure is included. These two sections must consist of only mass and inertia properties. Thus, only M1 can be non-zero for these two sections.

Of the remaining section array locations (locations eight thru fifty), locations fifteen thru seventeen must be defined for every section; whether or not other locations must be defined is dependent on the values for the section control indicators. This dependency is given below.

If a section has a non-zero M1, locations eight thru fourteen for that section must be defined (only non-zero numbers need to be inputted).

If a section has a non-zero M2, no additional locations other than locations fifteen thru seventeen must be defined.

If a section has a non-zero M3, locations ten, eleven, and eighteen thru twenty-four must be defined.

If a section has a non-zero M4, locations twenty-five thru twenty-nine must be defined.

If a section has a non-zero M5, locations thirty-eight thru forty must be defined.

If a section has a non-zero M6, locations forty-four thru fifty must be defined.

If a blade section has a value of 1.0 for M7, no additional information other than that provided in array locations 2 thru 200 is required.

The various section properties and corresponding locations are described in the following:

Loc. eight:	Lumped mass of section, m , lb-sec ² /ft
Loc. nine:	Offset of the section center of gravity in the local section y-axis direction from the local x-axis (shear center axis), ϵ_{ry} , ft (e.g., positive for a blade section if cg is forward of the local x-axis)
Loc. ten:	x-coordinate of the origin of the local blade or support structure section coordinate system in the rotating hub or fixed support structure coordinate system as depicted in Figure 9 for a blade section, h_{rx} , ft
Loc. eleven:	y-coordinate of the origin of the local blade or support structure section coordinate system in the rotating hub or fixed support structure coordinate system as depicted in Figure 9 for a blade section, h_{ry} , ft
Loc. twelve:	Local blade or support structure section polar mass moment of inertia I_x , lb-sec ² -ft
Loc. thirteen:	Local blade or support structure section flapwise mass moment of inertia I_y , lb-sec ² -ft

Loc. Local blade or support structure section
fourteen: edgewise mass moment of inertia I_z , lb-sec²-ft

Note: The mass moments of inertia are with respect to axes passing thru the section center of gravity and parallel to the local section coordinate system axes.

Section locations fifteen thru seventeen are angles which describe the orientation of the local blade or support structure section coordinate system with respect to the rotating hub or fixed support structure coordinate system. These have been discussed in detail in the section of this report pertaining to coordinate systems and are depicted in Figure 10.

Loc. Local blade or support structure section
fifteen: lead-lag angle, ϕ , positive for a right-handed presweep rotation about the reference coordinate system z-axis (for a blade, the outboard end of the section forward), rad

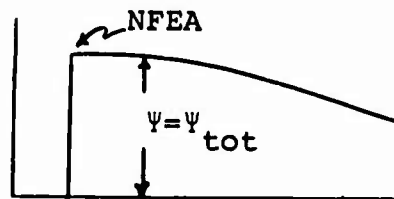
Loc. Local blade or support structure section
sixteen: coning angle, θ , positive for a right-handed precon rotation about the preswept reference coordinate system y-axis (for a blade, the outboard end of the section in the -z-axis direction), rad

Loc. Local blade or support structure section
seventeen: pitch angle, ψ , positive for a right-handed rotation about the preswept and precon reference coordinate system x-axis (for a blade, leading edge up), rad

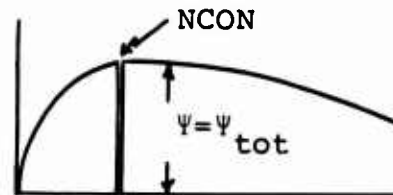
Note: The value read for a blade section can be taken as the total local section pitch angle (collective + twist contribution) with the collective pitch value of Loc. 11 equal to 0.0 or can be taken as only the local twist contribution. In the latter case this value is superimposed on the collective value of Loc. 11 in a manner dependent on the MFLEX (Loc. 25) and NFEA (Loc. 24) values (see Loc. 11). The twist contribution should include steady twist due to blade loads. As examples of the possibilities for the determination of ψ input values,

Loc. 11 = 0.0 and Loc. 11 \neq 0.0 cases can be considered for articulated and flexstrap or bearingless blades. If Loc. 11 is 0.0 then the value of Ψ for a section should be inputted as the total of collective and twist contribution at the section such that the Ψ distribution over the blade span might be as shown in the following sketch, where NFEA (Loc. 24) and NCON (Loc. 31) denote location of pitch bearing and flexstrap or bearingless pitch control structure restraint respectively.

Loc. 11 = 0.0



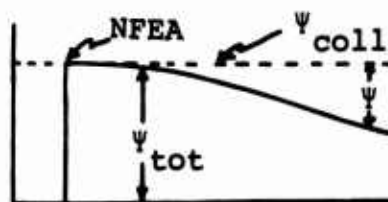
articulated blade



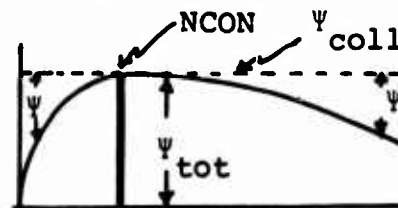
flexstrap or bearingless blade

But if a collective input, Ψ_{coll} , is used, then the Ψ for each section should be defined by utilizing the expression $\Psi = \Psi_{tot} - \Psi_{coll}$ at each section, as depicted in the following sketch where Ψ of a section, if other than zero, is negative.

Loc. 11 = Ψ_{coll}



articulated blade



flexstrap or bearingless blade

It should be noted that at the flexstrap or bearingless pitch control structure restraint location, ϕ , θ , and ψ_{tot} must be zero for consistency with the pitch control structure restraint representations.

Section locations eighteen thru twenty-four describe the aerodynamics of the section and need not be inputted if $M_3 = 0.0$

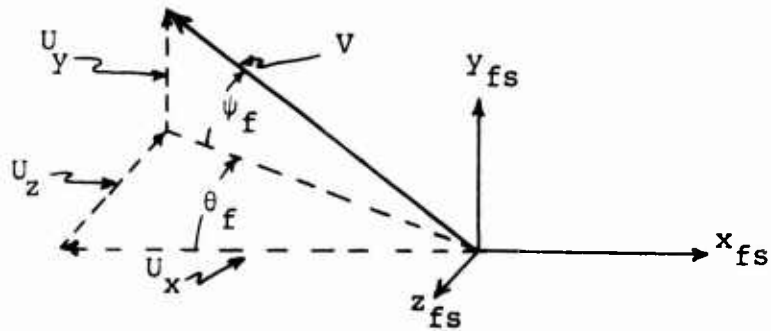
Loc. Aerodynamic table number (MTAB) which denotes the aerodynamic coefficient table to be used to determine the lift, drag, and moment coefficients and their slopes as a function of angle of attack and Mach number for this particular section

eighteen:

Note: The aerodynamic coefficient tables are discussed in further detail in the presentation of their input. Basically, the program was developed to use up to five different spanwise airfoil shapes. The representation of any of these shapes may involve the use of table look-up or analytical expressions for nonlinear aerodynamic coefficients. Thus, for a blade section, MTAB may equal any number from 1 to 5. However, for a support structure section, MTAB may also be equal to 6 and 7, where for MTAB = 6, the aerodynamic section is a NACA 0012 airfoil and for MTAB = 7, flat-plate crossflow aerodynamics are used. Due to present program dimensions only one aerodynamic coefficient table corresponding to MTAB equal to 1 can be used. These dimensional restrictions can be easily removed if more than one set of aerodynamic coefficient data must be used. These dimensional restrictions do not affect the use of MTAB = 6 or 7.

Loc. Effective aerodynamic length of the blade
nineteen: or support structure section, L_a , ft

Locations twenty and twenty-one are required for a support structure sections only. These are the two angles which describe the velocity of the helicopter with respect to the fixed support structure coordinate system, as depicted in the following sketch.



Loc. twenty: $\psi_f = \arcsin (U_y/V)$ where U_y is the climb rate and V is the resultant velocity of the helicopter, rad

Loc. twenty-one: $\theta_f = \arctan (U_z/U_x)$ where U_z is the lateral velocity and U_x is the forward velocity of the helicopter, rad

Note: U_x is positive if the helicopter is moving forward (negative x_{fs} direction), U_y is positive if the helicopter is climbing (positive y_{fs} direction), and U_z is positive if the helicopter is moving laterally to the starboard (negative z_{fs} direction). In such case that V is zero, both ψ_f and θ_f are indeterminate. In this case input these angles as zero. Similarly, if $U_x = U_z = 0$, but V is not zero (i.e., $U_y = V$) then input $\psi_f = +\pi/2$ if U_y is positive and $\psi_f = -\pi/2$ if U_y is negative. In either case input θ_f as zero.

Loc. twenty-two: Distance that the blade or support structure section shear center axis is forward of mid-chord along the local coordinate system y -axis, δ , ft

Loc. twenty-three: Chord length of the blade or support structure section, c , ft

Loc. twenty-four: Local induced velocity of the blade section, v_i , ft/sec (not required for a support structure section)

Note: This induced velocity needs to be inputted only when the control parameter NVI (Loc. 36) is 0.0. If NVI is greater than zero, the induced velocity as a function of radial and azimuthal position of the blade section inputted in locations 2201 - 2800 is used.

Locations twenty-five thru twenty-nine must be defined if M4 is non-zero.

Loc. twenty-five: A parameter used in combination with the centrifugal force acting on a section to specify the centrifugal stiffening effects on the torsional stiffness of a flexible strap or beam section inboard of the flex-strap or bearingless pitch control structure restraint application; defined as the effective value of $.5 (EI_y + EI_z)A_s / (I_f + I_c)$ (in pounds) for the section where EI_y and EI_z are the section flapwise and edgewise bending stiffness, respectively, A_s is the total cross-sectional area of the section, and I_f and I_c are the flapwise and edgewise area moments of inertia, respectively
Note: Only required for sections in which this effect is significant. Not required for a support structure section.

Loc. twenty-six: Elastic length of the blade or support structure section, L_e , ft

Loc. twenty-seven: Local blade or support structure section torsional stiffness, EI_x , lb-ft²

Loc. twenty-eight: Local blade or support structure section flapwise bending stiffness, EI_y , lb-ft²

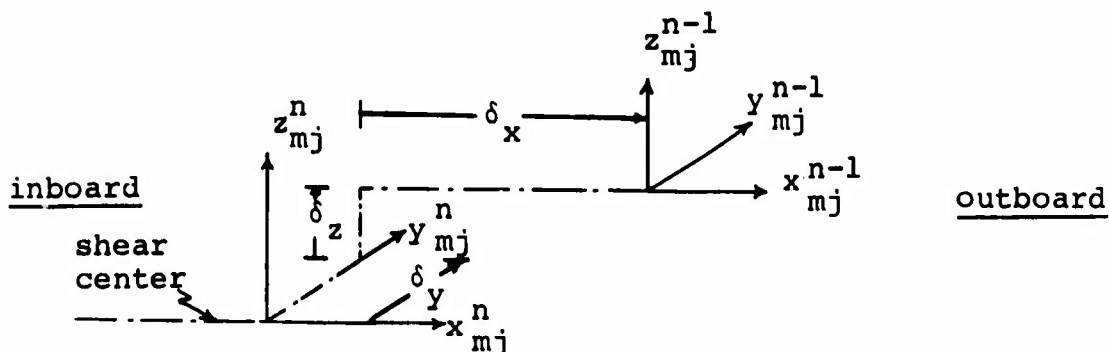
Loc. twenty-nine: Local blade or support structure section edgewise bending stiffness, EI_z , lb-ft²

Loc. thirty: Centrifugal force acting on the blade section that is estimated inside the program using mass distribution and rotational

speed, and for output purposes is stored in this location, lb (no input needed)

Locations thirty-one thru thirty-seven are not used.

Locations thirty-eight thru forty need only be described if M5 is non-zero. The pertinent parameters are depicted in the following sketch for a blade section where n and $n-1$ superscripts denote a local coordinate system inboard and outboard of the rigid offset characteristics, respectively.



Loc. thirty-eight: Rigid blade or support structure section shear center axis offset in the local coordinate system x -axis direction, δ_x , ft

Loc. thirty-nine: Rigid blade or support structure section shear center axis offset in the local coordinate system y -axis direction, δ_y , ft

Loc. forty: Rigid blade or support structure section shear center axis offset in the local coordinate system z -axis direction, δ_z , ft

Locations forty-one thru forty-three are not used.

Locations forty-four thru fifty, which pertain to consideration of localized torsional spring-damper units in the blade or support structure section, need be described only when M6 is non-zero. Also, locations forty-four thru forty-nine cannot be defined non-zero if location fifty is non-zero.

- Loc.
forty-four: Inverse of the localized torsional spring-damper unit stiffness about the local section coordinate system x-axis,
 K_x^{-1} , rad/(ft-lb)
- Loc.
forty-five: Inverse of the localized torsional spring-damper unit stiffness about the local section coordinate system y-axis,
 K_y^{-1} , rad/(ft-lb)
- Loc.
forty-six: Inverse of the localized torsional spring-damper unit stiffness about the local section coordinate system z-axis,
 K_z^{-1} , rad/(ft-lb)
- Loc.
forty-seven: Damper time constant (c/k) associated with the localized torsional spring-damper unit acting about the local section coordinate system x-axis, τ_x , sec
- Loc.
forty-eight: Damper time constant (c/k) associated with the localized torsional spring-damper unit acting about the local section coordinate system y-axis, τ_y , sec
- Loc.
forty-nine: Damper time constant (c/k) associated with the localized torsional spring-damper unit acting about the local section coordinate system z-axis, τ_z , sec
- Loc.
fifty: Control system spring constant, K_s , which may be used to represent the applied control torque as a linear function of local torsional rotation. (However, use of the previously described representations of the control system effects are recommended instead)

TABLE 3. SECTION DATA INPUT LOCATIONS AND CORRESPONDING SYMBOLS					
Location	Symbol	Location	Symbol	Location	Symbol
X* + 0001	M1	X + 0018	MTAB	X + 0035	↑
X + 0002	M2	X + 0019	L _a	X + 0036	not used
X + 0003	M3	X + 0020	ψ _f	X + 0037	↓
X + 0004	M4	X + 0021	θ _f	X + 0038	δ _x
X + 0005	M5	X + 0022	δ	X + 0039	δ _y
X + 0006	M6	X + 0023	c	X + 0040	δ _z
X + 0007	M7	X + 0024	v _i	X + 0041	↑
X + 0008	m	X + 0025	GJ _{cs}	X + 0042	not used
X + 0009	ε _{ry}	X + 0026	L _e	X + 0043	↓
X + 0010	h _{rx}	X + 0027	EI _x	X + 0044	K _x ⁻¹
X + 0011	h _{ry}	X + 0028	EI _y	X + 0045	K _y ⁻¹
X + 0012	I _x	X + 0029	EI _z	X + 0046	K _z ⁻¹
X + 0013	I _y	X + 0030	P _T	X + 0047	τ _x
X + 0014	I _z	X + 0031	↑	X + 0048	τ _y
X + 0015	φ	X + 0032	not used	X + 0049	τ _z
X + 0016	θ	X + 0033	↓	X + 0050	K _s
X + 0017	ψ	X + 0034			

*X represents the last data input location of the previous section which is factorable by 50.

Optional Induced Velocity Distribution Input

The computer program can use a rotor induced velocity distribution which varies both azimuthally and radially, determined by use of an independent analysis, instead of a radially varying distribution inputted through blade section data. The use of this optional induced velocity distribution is controlled by the parameter NVI (Loc. 36). If the value of this array location is equal to 1.0 this optional induced velocity distribution is used and must be defined for Locations 2201 - 2800. This set of induced velocities is stored in the array V(25,24) in common block PLOAD. The first index of this array corresponds to blade aerodynamic section number (radial position), and the second index corresponds to the azimuthal position of the blade. The blade aerodynamic section numbers are determined in the same manner as the blade section numbers except only blade sections having aerodynamic characteristics included are counted. The induced velocity values for each aerodynamic blade section are those occurring at the radial position of the aerodynamic load (lumped mass) point on the section.

The induced velocities are inputted for each radial aerodynamic section going from the blade tip section to the most inboard blade aerodynamic section, while maintaining a fixed blade azimuthal position. Then, the induced velocities are inputted for each radial aerodynamic section, taken in the same order as above, at a fixed blade azimuthal position one azimuthal step in the direction of rotor rotation ahead of the previous azimuthal position. This process is repeated until the induced velocities at each blade aerodynamic section (radial position) and blade azimuthal position have been defined. Due to the dimensions of the induced velocity storage array, 25 input locations corresponding to aerodynamic radial position are reserved for each azimuthal position of the blade and up to 24 blade azimuthal positions are allowed, which are equally spaced azimuthally. The azimuthal blade positions allowed are dependent on the value of the control input parameter NAS (Loc. 34) and defined by $\psi_k = 2\pi(1-l)/NAS$ where ψ_k is the kth azimuthal blade position, in radians. This azimuthal blade position is defined relative to the fixed hub coordinate system x-axis (reference blade position) and k is an integer having values of 1 through

NAS. Thus, location numbers 2201 - 2225 are associated with the blade azimuthal position denoted by ψ_1 , location numbers 2226 - 2250 are associated with the blade azimuthal position denoted by ψ_2 , and so on.

In general, the location number for the induced velocity of the nth aerodynamic radial position at the kth blade azimuthal location is defined as $2200 + 25(k-1) + n$. Since k cannot be greater than NAS and n cannot be greater than the number of blade aerodynamic sections, the largest location number to be defined would be $2200 + 25(NAS-1) +$ the number of aerodynamic sections.

It should be noted that if the induced velocity distribution to be used does not vary azimuthally but is either uniform or only varies radially, this distribution can be more conveniently inputted by the definition of v_i (location twenty-four of the blade section data) with the logic control parameter NVI equal to 0.0.

Aerodynamic Coefficient Input

The airfoil aerodynamic coefficient data is to be included for a case only when the control parameter MAER (corresponding to location 26) of the case is equal to 1 and must be included after the number 8 card following the storage array data for the case. This data, if required, is inputted with an entirely different format than that utilized for the previously discussed storage array data.

The aerodynamic coefficient data for an airfoil shape can consist of any of three types, corresponding to the form of representation to be used to determine the aerodynamic coefficients. The first type consists of data describing aerodynamic coefficients in a tabular form for a lower and upper range of angle of attack values, and of data defining coefficients to be used in a series representation of aerodynamic coefficients for angles of attack between the two tabular data ranges. For example, tabular data may be provided for angle of attack values of 0 to 20 degrees and 350 to 360 degrees, with series coefficient data provided for angle of attack values between 20 to 350 degrees. The second type consists of only data defining coefficients to be used in a series representation of the aerodynamic coefficients of a symmetrical airfoil. The third type consists primarily of aerodynamic coefficient data in the C-81 computer program form.

Initially, the input data required for the first and second type of aerodynamic coefficient representation is presented. This is followed by a detailed discussion of the series representation used in both of these types of aerodynamic coefficient representation. The input data required for the third type of representation is then briefly discussed in regard to the modifications required, in that the C-81 aerodynamic table format has been fully documented elsewhere (e.g., Reference 3). All aerodynamic coefficient input is read in and stored in common block COEFFS for later use in subroutine ACOEFF.

Although the HASTA program is presently limited to the consideration of only one set of aerodynamic coefficient data due to dimensional restrictions, the program was developed to consider up to five different sets of aerodynamic coefficient data. Thus, the construction of the aerodynamic data described below will be based on five different sets:

First Card - Format 14I5

NTAB, a number from one to five defining the number of airfoil aerodynamic coefficient data sets to be inputted.

Note: The following cards are included for each set of aerodynamic coefficient data.

Card #1 - Format 14I5

ISER, a control parameter such that if equal to 0, the aerodynamic coefficient data consisting of both tabular and series coefficient data (corresponding to the first type of data) is to be inputted; if equal to 1, aerodynamic coefficient data consisting of only series coefficient data (corresponding to the second type of data) is to be inputted; and if equal to 81, aerodynamic coefficient data based on the C-81 aerodynamic table format is to be inputted (corresponding to the third type of data).

-
3. Davis, John M., ROTORCRAFT FLIGHT SIMULATION WITH AERO-ELASTIC ROTOR AND IMPROVED AERODYNAMIC REPRESENTATION, Volume II - User's Manual, Bell Helicopter Company; USAAMRDL-TR-74-10B, Eustis Directorate, US Army Air Mobility Research and Development Laboratory, Fort Eustis, Virginia, June 1974, AD 782756

Note: As an example, if aerodynamic coefficient data set number two is designated to read in only the series coefficients of the second type (i.e., $ISER = 1$), then any aerodynamic section with its section input variable $MTAB$ (corresponding to location eighteen) = 2 will use the series representation with the inputted coefficients for determination of lift, drag, and moment coefficients.

Card # 2 - Format 14I5

MT , a number from one to five assigned to this set of aerodynamic coefficient data to distinguish it from other sets such that the input between this card and the next $ISEK$ card is associated with a data set designated by the number MT .

Note: Due to present program dimensional restrictions, MT must = 1.

An aerodynamic coefficient data set is constructed of the data necessary for the determination of the two-dimensional lift, drag, and moment coefficients about quarter chord (i.e., about a point on the airfoil chord located $1/4$ chord-length aft of the leading edge of the airfoil). The data necessary for each of these three types of two-dimensional coefficients is presented in the following discussion in the required order of input. The data for each type of two-dimensional coefficient is constructed independent of the other two-dimensional coefficients.

Lift Coefficient Data:

Card #3.1 = Format 14I5

$NCC1$ (equivalent to $NCC(MT,1)$), number of series coefficients for lift coefficient determination where if $ISER = 0$ a value of 15 is required and if $ISER = 1$ a value of 20 is required

Cards #3.1a - Format 5E14.7 (more than one card)

$CCS(K,MT,1)$ for $K = 1$, $NCC1$, coefficients (constants) of the series representations for determination of lift coefficients (dependent on type of series to be used) such that

Cols. 1 - 14 $C1_1$, the first constant
 15 - 28 $C1_2$, the second constant
 29 - 42 $C1_3$, the third constant
 43 - 56 $C1_4$, the fourth constant
 57 - 70 $C1_5$, the fifth constant

Next card

Cols. 1 - 14 $C1_6$, the sixth constant
 15 - 28 $C1_7$, the seventh constant
 etc., until $C1_{NCC1}$ has been reached
 where the form $C1_k$ has been used to
 denote $CCS(K,MT,1)$

Note: Cards #3.2, etc., are not inputted if $ISER = 1$.

Card #3.2 - Format 14I5

$NMACHS$ (equivalent to $NMACH(MT,1)$), the number
 of Mach numbers involved in the lift coefficient
 table where $NMACHS$ cannot be greater than 11

Card #3.2a - Format 7F10.0 (may be more than one card)

$AMACH(K,MT,1)$ for $K = 1, NMACHS$, values of Mach
 numbers for the construction of the lift coefficient
 table such that

Cols. 1 - 10 M_1 , lowest Mach number
 11 - 20 M_2 , next highest Mach number
 21 - 30 M_3 , next highest Mach number
 31 - 40 M_4 , next highest Mach number
 41 - 50 M_5 , next highest Mach number
 etc., until M_{NMACHS} has been reached
 where M_k has been used to denote $AMACH(K,MT,1)$
 Note: The value of M_1 should be zero and
 the highest Mach number should be equal to 1.0.

Card #3.3 - Format 14I5

NCL (equivalent to NCLS(1,MT,1)) - first index being the Mach number index, the number of angles of attack (with corresponding lift coefficients) for Mach number M_1 where NCL cannot be greater than 50 for any Mach number

Card #3.3a - Format 7F10.0 (may be more than one card)

ALPHS(1,K,MT,1) for $K = 1$, NCL - first index being the Mach number index, values of angles of attack, deg, for the first Mach number such that

Cols. 1 - 10 α_1 , first angle of attack for M_1
11 - 20 α_2 , second angle of attack for M_1
21 - 30 α_3 , third angle of attack for M_1
31 - 40 α_4 , fourth angle of attack for M_1
etc., until α_{NCL} has been reached

where α_K has been used to denote ALPHS(1,K,MT,1)

Note: Normally the angle of attack distribution is such that it represents a lower and an upper range of angle of attack with a series representation utilized between the two ranges.

Card #3.3b - Format 7F10.0 (same number of cards as Card #3.3a)

CLDM(1,K,MT,1) for $K = 1$, NCL - first index being the Mach number index, values of the lift coefficient for the first Mach number and each angle of attack such that

Cols. 1 - 10 CL_1 , lift coefficient at M_1 , α_1
11 - 20 CL_2 , lift coefficient at M_1 , α_2
21 - 30 CL_3 , lift coefficient at M_1 , α_3
31 - 40 CL_4 , lift coefficient at M_1 , α_4
etc., until CL_{NCL} has been reached

where CL_K has been used to denote
 $CLDM(1,K,MT,1)$

Card #3.4, Card #3.4a, and Card #3.4b are the next set of input cards; they are similar to Card #3.3, Card #3.3a, and Card #3.3b, respectively, except that the input numbers are those corresponding to M_2 .

Similar sets of cards are added corresponding to M_3 thru M_{NMACHS} to complete the lift coefficient data required to construct an aerodynamic coefficient data set. It is not necessary to have the same number of or values of angles of attack for each Mach number, since NCL is read for each Mach number. However, the angle of attack defining the upper end of the lower angle of attack range and the angle of attack defining the lower end of the upper angle of attack range should be the same for all Mach numbers used in constructing an aerodynamic coefficient table, (e.g., lift coefficient table).

Drag Coefficient Data:

Input for the drag coefficient data is carried out in the same fashion as that for the lift coefficient data. Thus,

Card #4.1 - Format 14I5

NCC1 (equivalent to $NCC(MT,2)$), the number of series coefficients for drag coefficient determination where if $ISER = 0$ a value of 10 is required and if $ISER = 1$ a value of 7 is required

Card #4.1a - Format 5E14.7 (more than one card)

$CCS(K,MT,2)$ for $K = 1$, NCC1, coefficients (constants) of the series representations for the determination of drag coefficients (dependent on type of series to be used)

Note: Cards #4.2, etc., for the drag coefficient table are not inputted if $ISER = 1$.

Card #4.2 - Format 14I5

NMACHS (equivalent to NMACH(MT,2)), the number of Mach numbers involved in the drag coefficient table where NMACHS cannot be greater than 10

Card #4.2a - Format 7F10.0 (may be more than one card)

AMACH(K,MT,2) for $K = 1, \text{NMACHS}$, values of Mach numbers for the construction of the drag coefficient table where M_1 should be zero and the highest Mach number should be 1.0

Card #4.3 - Format 14I5

NCL (equivalent to NCLS (1,MT,2)) - first index being the Mach number index), the number of angles of attack (with corresponding drag coefficients) for Mach number M_1 where NCL cannot be greater than 15 for any Mach number

Card #4.3a - Format 7F10.0 (may be more than one card)

ALPHS(1,K,MT,2) for $K = 1, \text{NCL}$ - first index being the Mach number index, values of angles of attack, deg, for the first Mach number

Card #4.3b - Format 7F10.0 (may be more than one card)

CLDM(1,K,MT,2) for $K = 1, \text{NCL}$ - first index being the Mach number index, values of the drag coefficients for the first Mach number and each angle of attack

Cards 4.3, 4.3a, and 4.3b are followed by similar sets of cards with values corresponding to drag coefficient table Mach numbers M_2 thru M_{NMACHS} to complete the drag coefficient data required to construct an aerodynamic coefficient data set.

Moment Coefficient Data:

Input of the moment coefficient data is carried out in the same manner as that for the lift coefficient data or the drag coefficient data. These cards may be

numbered starting with Card #5.1. For the moment coefficient data, if ISER = 0 a value of 22 is required for the NCC1 card and if ISER = 1 a value of 16 is required. The moment coefficient data must correspond to the data necessary to define the two-dimensional moment coefficient about quarter chord (i.e., about a point on the airfoil chord line located a distance of 1/4 of a chordlength aft of the airfoil leading edge).

The construction of the preceding data completes the input necessary for the first data set, which is designated as the MTth set of aerodynamic coefficient data by the number inputted for MT.

The input of the next set of aerodynamic coefficient data (if NTAB>1) must be constructed in the same fashion as that given for the previous data set, starting with input for variable ISER, followed by input for MT, NCC(MT,1), etc. The same procedure is repeated for the third, fourth, and fifth sets of data, if required. Thus, additional sets of data would be inputted in the form represented below.

Aerodynamic Coefficient Data Set No. 2 (Include only if NTAB₂):

- Cards #6 Data set type card (either ISER = 0 or ISER = 1)
- Cards #7 Data set identification card (MT = any number from 1 - 5 that has not been used for the other data sets)
- Cards #8 Lift coefficient data
- Cards #9 Drag coefficient data
- Cards #10 Moment coefficient data

Aerodynamic Coefficient Data Set No. 3 (Include only if NTAB₃):

- Cards #11 Data set type card (either ISER = 0 or ISER = 1)

Cards #12 Data set identification card (MT = any number from 1 - 5 that has not been used for the other data sets)

Cards #13 Lift coefficient data

Cards #14 Drag coefficient data

Cards #15 Moment coefficient data

Aerodynamic Coefficient Data Set No. 4 (Include only if $NTAB > 4$):

Cards #16 Data set type card (either $ISER = 0$ or $ISER = 1$)

Cards #17 Data set identification card (MT = any number from 1 - 5 that has not been used for the other data sets)

Cards #18 Lift coefficient data

Cards #19 Drag coefficient data

Cards #20 Moment coefficient data

Aerodynamic Coefficient Data Set No. 5 (Include only if $NTAB = 5$):

Cards #21 Data set type card (either $ISER = 0$ or $ISER = 1$)

Cards #22 Data set identification card (MT = any number from 1 - 5 that has not been used for the other data sets)

Cards #23 Lift coefficient data

Cards #24 Drag coefficient data

Cards #25 Moment coefficient data

The series coefficients required for an aerodynamic coefficient data set are dependent upon the type of representation being utilized to determine the two-dimensional lift, drag, and moment coefficients and their slopes about airfoil quarter chord. For the first type of representation (tables plus series), $ISER = 0$, a series representation is utilized if the angle of attack α is not in the ranges represented by the tabular data. For the second type of representation, $ISER = 1$, a series representation is always used. The various coefficients (constants) required for each of these series representations is presented in the following discussion where Cl_K , $C2_K$, and $C3_K$ correspond to lift, drag, and moment series constants, respectively.

Lift Coefficient Series Representation ($ISER = 0$):

This representation is used when the condition $Cl_1 < \alpha < Cl_2$ is satisfied, where Cl_1 , which must be ≥ 0 , is the maximum value of the angle of attack, deg, for which the lower angle of attack range of the lift coefficient tabular data can be used and Cl_2 is the minimum value of the angle of attack, deg, for which the upper angle of attack range of the lift coefficient tabular data can be used. If the maximum value of the angle of attack in the lower range of the tabular data is not the same for all Mach numbers of the table, then the lowest maximum angle of attack defining the lower angle of attack range of all Mach numbers must be used for Cl_1 .

Similarly, if the minimum value of the angle of attack in the upper range of the tabular data is not the same for all Mach numbers, then the highest minimum angle of attack defining the upper angle of attack range of all Mach numbers must be used for Cl_2 .

As an aid to defining in series form the lift coefficient as a function of angle of attack α , two intermediate functions can be defined. These functions are in general form:

$$L1(x) = Cl_3 + Cl_4x + Cl_5x^2 + Cl_6x^3 + Cl_7x^4$$

$$L2(x) = Cl_8 + Cl_9x + Cl_{10}x^2$$

Then, the series representation for the two-dimensional lift coefficient CL for various ranges of angle of attack is defined as:

$$CL = Cl_{11}L1(\alpha) \quad \text{for} \quad Cl_1 < \alpha \leq 90^\circ$$

$$CL = Cl_{12}L1(\alpha) \quad \text{for} \quad 90^\circ < \alpha \leq 160^\circ$$

$$CL = Cl_{12}L2(\alpha) + Cl_{13}(\alpha-160)/20 \quad \text{for} \quad 160^\circ < \alpha \leq 180^\circ$$

$$CL = Cl_{14}L2(\Delta) + Cl_{13}(200-\alpha)/20 \quad \text{for} \quad 180^\circ < \alpha \leq 200^\circ$$

$$CL = -Cl_{14}L1(\Delta) \quad \text{for} \quad 200^\circ < \alpha \leq 270^\circ$$

$$CL = -Cl_{15}L1(\Delta) \quad \text{for} \quad 270^\circ < \alpha < Cl_2$$

where $\Delta = 360^\circ - \alpha$.

Drag Coefficient Series Representation (ISER = 0):

This representation is used when the condition $C2_1 < \alpha < C2_2$ is satisfied, where the $C2_1$ and $C2_2$ angles, deg, are based on the upper and lower angle of attack limits of the lower and upper angle of attack ranges for the drag coefficient tabular data, respectively, in the same manner as the Cl_1 and Cl_2 angles were determined. As an aid to defining in series form the drag coefficient as a function of the angle of attack α , two intermediate functions can be defined. These functions in a general form are:

$$D1(x) = C2_3 + C2_4x + C2_5x^2 + C2_6x^3 + C2_7x^4$$

$$D2(x) = C2_8 + C2_9x + C2_{10}x^2$$

The series representation for the two-dimensional drag coefficient CD for various ranges of angle of attack is defined as:

$$CD = D1(\gamma_1) \quad \text{for} \quad C2_1 < \alpha \leq 90^\circ$$

$$CD = D1(\gamma_2) \quad \text{for} \quad 90^\circ < \alpha \leq 166^\circ$$

$$CD = D2(\alpha) \quad \text{for} \quad 166^\circ < \alpha \leq 180^\circ$$

$$C_D = D_2(\Delta) \quad \text{for} \quad 180^\circ < \alpha < 194^\circ$$

$$C_D = D_1(-\gamma_2) \quad \text{for} \quad 194^\circ \leq \alpha < 270^\circ$$

$$C_D = D_1(\gamma_3) \quad \text{for} \quad 270^\circ \leq \alpha < C_{2_2}$$

where

$$\gamma_1 = \alpha + (15 - C_{2_1})(90 - \alpha) / (90 - C_{2_1})$$

$$\gamma_2 = 180 - \alpha, \text{ and}$$

$$\gamma_3 = 360 - \alpha - (345 - C_{2_2})(\alpha - 270) / (C_{2_2} - 270).$$

Moment Coefficient Series Representation (ISER = 0):

This representation is used when the condition $C_{3_1} < \alpha < C_{3_2}$ is satisfied where the C_{3_1} and C_{3_2} angles, deg, are based on the upper and lower angle of attack limits of the lower and upper angle of attack ranges for the moment coefficient tabular data, determined in the same manner as the C_{1_1} and C_{1_2} angles were determined. As an aid to defining in series form the moment coefficient about airfoil quarter chord as a function of the angle of attack, five functions can be defined. These functions in a general form are:

$$M_1(x) = C_{3_3}(30-x)/14 + C_{3_4} + C_{3_5}x + C_{3_6}x^2 + C_{3_7}x^3$$

$$M_2(x) = C_{3_8} + C_{3_9}x + C_{3_{10}}x^2 + C_{3_{11}}x^3 + C_{3_{12}}x^4$$

$$M_3(x) = C_{3_{13}} + C_{3_{14}}x + C_{3_{15}}x^2 + C_{3_{16}}x^3$$

$$M_4(x) = C_{3_{17}} + C_{3_{18}}x + C_{3_{19}}x^2 + C_{3_{20}}x^3 + C_{3_{21}}x^4$$

$$M_5(x) = C_{3_{22}}(\alpha - 330) / (C_{3_2} - 330) - C_{3_4} - C_{3_5}x \\ - C_{3_6}x^2 - C_{3_7}x^3$$

The series representation for the two-dimensional moment coefficient about the airfoil quarter chord for various ranges of angle of attack is then defined as:

$$C_M = M_1(\alpha) \quad \text{for} \quad C_{3_1} < \alpha \leq 30^\circ$$

$$C_M = M_2(\alpha) \quad \text{for} \quad 30^\circ < \alpha \leq 150^\circ$$

$$\begin{aligned}
CM &= M3(\alpha) && \text{for } 150^\circ < \alpha < 168^\circ \\
CM &= M4(\alpha) && \text{for } 168^\circ < \alpha < 180^\circ \\
CM &= -M4(\Delta) && \text{for } 180^\circ < \alpha < 192^\circ \\
CM &= -M3(\Delta) && \text{for } 192^\circ < \alpha < 210^\circ \\
CM &= -M2(\Delta) && \text{for } 210^\circ < \alpha < 330^\circ \\
CM &= M5(\Delta) && \text{for } 330^\circ < \alpha < 360^\circ
\end{aligned}$$

For the first type of aerodynamic coefficient representation (tabular plus series form) the angle of attack α has the units of degrees, and the coefficients required in the series representation are based on this fact. For the second type of aerodynamic coefficient representation, the angle of attack α has the units of radians in regard to the symmetrical airfoil series representation and, thus, the series coefficients for this type must be based on this fact. In addition to the condition that ISER = 1 for the second type (series only), the value of Cl_1 must be less than zero.

If the combination of ISER = 1 and $Cl_1 \geq 0$ is used, the program will terminate execution. Since a symmetrical airfoil is being represented, only the expressions for the lift, drag, and moment coefficients about quarter chord for α between 0 and π radians are required to define the necessary constants. If α is greater than π the aerodynamic coefficients are determined using the expressions, with α being redefined as $2\pi - \alpha$ and the sign changed on the two-dimensional aerodynamic coefficients, as required.

Lift Coefficient Representation (ISER = 1, $Cl_1 < 0$):

As an aid to defining the lift coefficient the following terms can be defined:

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

$$LL = Cl_2(1-M)(1+Cl_3)$$

$$LM_1 = [(1-M)Cl_8 + (Cl_9M + Cl_{10})\alpha] / (Cl_{11} + Cl_{12}M)$$

$$LM_2 = (Cl_{13}\sin\alpha + Cl_{14}\sin2\alpha + Cl_{15}\sin3\alpha + Cl_{16}\sin4\alpha)$$

where M is the Mach number. The term LL, with $Cl_3 = 0$, is the angle in radians at which the lift coefficient curve normally becomes nonlinear. The purpose of Cl_3 is to

allow extension of the normal range of linear aerodynamics to a higher angle of attack. The representation for the two-dimensional lift coefficient for various angle of attack ranges is defined as:

$$\begin{aligned}
 CL &= Cl_7 \alpha / SQT & \text{for } 0 < \alpha < LL \\
 CL &= LM_1 / SQT & \text{for } LL < \alpha < Cl_4 \\
 CL &= LM_2 / SQT & \text{for } Cl_4 < \alpha < Cl_5 \\
 CL &= (Cl_{17} + Cl_{18} \alpha) / SQT & \text{for } Cl_5 < \alpha < Cl_6 \\
 CL &= (Cl_{19} + Cl_{20} \alpha) / SQT & \text{for } Cl_6 < \alpha < \pi
 \end{aligned}$$

Drag Coefficient Representation (ISER = 1, $Cl_1 < 0$):

As an aid to defining the drag coefficient the following terms can be defined:

$$\begin{aligned}
 DM_1 &= C2_3 + C2_4 \cos \alpha + C2_5 \cos 2\alpha + C2_6 \cos 3\alpha \\
 &\quad + C2_7 \cos 4\alpha
 \end{aligned}$$

The representation for the two-dimensional drag coefficient for various angles of attack ranges is defined as:

$$\begin{aligned}
 CD &= C2_1 + C2_2 \alpha^2 & \text{for } 0 < \alpha < LL \\
 CD &= DM_1 / SQT & \text{for } LL < \alpha < \pi
 \end{aligned}$$

Moment Coefficient Representation (ISER = 1, $Cl_1 < 0$):

As an aid to defining the moment coefficient the following terms can be defined:

$$\begin{aligned}
 MM_1 &= C3_6 (C3_7 - (\alpha + C3_8) / C3_9) \\
 MM_2 &= C3_1 \sin \alpha + C3_2 \sin 2\alpha + C3_3 \sin 3\alpha + C3_4 \sin 4\alpha \\
 &\quad - .25 (LM_2 \cos \alpha + DM_1 \sin \alpha)
 \end{aligned}$$

where the first four terms of the last expression above define the moment coefficient about midchord. The representation for the two-dimensional moment coefficient for various angles of attack ranges is defined as:

$$\begin{aligned}
 CM &= 0 && \text{for } 0 \leq \alpha \leq Cl_4 \\
 CM &= MM_2/SQT && \text{for } Cl_4 < \alpha \leq Cl_5 \\
 CM &= C3_5/SQT && \text{for } Cl_5 < \alpha \leq Cl_6 \\
 CM &= MM_1/SQT && \text{for } Cl_6 < \alpha \leq \pi
 \end{aligned}$$

For this type of aerodynamic coefficient representation the lift, drag, and moment coefficient slopes are normally determined by use of analytical expressions obtained by taking the derivative of the expressions for the two-dimensional aerodynamic coefficients with respect to α , such that additional constants are not required to define the slopes. However, this is not the case for the moment coefficient slope for $Cl_4 < \alpha \leq Cl_5$ where the moment coefficient expression directly uses the values obtained for CL and CD for the lift and drag contribution to the moment coefficient about quarter chord. In this case, the moment coefficient slope expression, which is obtained by taking the derivative with respect to α of the moment coefficient expression with the lift and drag coefficient effects represented in series form, requires the definition of additional constants. These constants can be defined in terms of previously defined constants as:

$$\begin{aligned}
 C3_{10} &= C3_1 - Cl_{14}/2 - C2_3/4 - C2_5/4 \\
 C3_{11} &= 2C3_2 - Cl_{13}/4 - 3Cl_{15}/4 - C2_4/4 - C2_6/4 \\
 C3_{12} &= 3C3_3 - Cl_{16} - C2_7/4 \\
 C3_{13} &= 4C3_4 \\
 C3_{14} &= 3(Cl_{14} + C2_5)/4 \\
 C3_{15} &= Cl_{15} + C2_6 \\
 C3_{16} &= 5(Cl_{16} + C2_7)/4
 \end{aligned}$$

The modification required to the C-81 format aerodynamics data table for use in the HASTA program is discussed below. Initially, the table identification card and the title and control card are removed from an existing C-81 data deck such that only a basic lift coefficient table, drag coefficient table, and a moment coefficient table (in that order) remain. Six cards must be added to the front of the resulting C-81 deck. These are, in terms of previously defined data cards,

1. First Card - Format 14I5 - (NTAB)
2. Card #1 - Format 14I5 - (ISER)
3. Card #2 - Format 14I5 - (MT)
4. Card #3.1 - Format 14I5 - (NCC1)
5. Card #3.1a - Format 5E14.7 - (CCS(K, MT,1))
6. Card #3.2 - Format 14I5 - (NMACHS, NCL)

which must be added in the order listed. For C-81 format tables ISER (Card #1) must be equal to 81. The First Card and Card #2 are as previously defined. NCC1 (Card #3.1) is defined equal to 2 and CCS (1, MT, 1) and CCS (2, MT, 1) (Card #3.1a) are defined equal to 180.0 and -180.0, respectively, corresponding to the highest and lowest angle of attack included in the lift coefficient table. Two integer values corresponding to the number of Mach numbers involved in the lift coefficient table and the number of angles of attack involved in the same table, in the order mentioned, must be provided on Card #3.2 instead of the previously defined value. After the last card of the lift coefficient table but before the first card of the drag coefficient table, three cards must be added in the following order: Card #4.1 (NCC2), Card #4.1a (CCS (1, MT, 2), CCS (2, MT, 2)), and Card #4.2 (NMACHS, NCL). These three cards are identical in concept to Cards #3.1, #3.1a, and #3.2 except that they are defined in regard to the drag coefficient table. In fact, NCC2 must be defined as 2 and CCS (1, MT, 2) and CCS (2, MT, 2) must be defined as 180.0 and -180.0, respectively. Similarly, after the last card of the drag coefficient table but before the first card of the moment coefficient table, three cards containing NCC3; CCS (1, MT, 3) and CCS (2, MT, 3); NMACH and NCL must be added. These are defined as 2; 180.0 and -180.0;

the number of Mach numbers involved in the moment coefficient table and the number of angles of attack involved in the same table, respectively.

If more than one C-81 data set is to be used the additional sets would be constructed in the same manner but without the First Card, which contains NTAB, and added to the back of the first set. In fact, the three different types of aerodynamic coefficient representations can be utilized for one case by noting that the departure path of input generation is initiated at Card #1 (ISER).

DESCRIPTION OF PROGRAM OUTPUT

The output obtained from the execution of the program is dependent upon the type of run, the type of data included in the input deck, and the program options activated by the various input parameters. This output may consist of both printed and punched output. For purposes of discussion the output can be divided into several basic output groups, which are discussed individually in the following sections. Each of these groups may not be outputted or may have various parts which may or may not be outputted depending on the values used for the input parameters. The dependency of output on input parameters is also presented in the discussion of the various output groups.

The general sequence in which the basic output groups are outputted (overall output flow) for a general computer run can be represented by the following steps where the output is in printed form unless otherwise noted:

1. Output of storage array data
2. Output of program control parameters
3. Output of aerodynamic coefficient tables
4. Output of sectional aerodynamic coefficients and steady loads
5. Output of section data
6. Output of search option information, if used
7. Output of iteration scheme steps

8. Shape vector output (printed)
9. Shape vector output (punched)
10. Output groups 7, 8, and 9 are repeated for each additional starting eigenvalue inputted or obtained from use of the search option for the first system configuration (case) of a computer run
11. Output specified by 1 - 10 is repeated for each additional system configuration (case) of a computer run

where output groups 1 - 9 are the basic output groups that will be discussed in the following sections.

Output of Storage Array Data

The first numerical output for a case of a run consists of the printout of the basic input deck, which defines values for variables of the 2800-element storage array. This output is preceded by a title heading, PRINTOUT OF INPUT DATA, and by column headings of the form LOC NUM VAL1 VAL2 VAL3 VAL4 VAL5. The numerical printout of all storage location input cards follows this last heading in the same order in which they are read. Each printed line consists of the initial storage array location number (under LOC), the number of variable values on the card to be read (under NUM), and up to five variable values, depending on the value of NUM (under VAL1, VAL2, VAL3, VAL4, and VAL5, as required). The variable values are printed in a 5(G14.5,4X) format to aid in the determination of whether or not these values are properly aligned on the input cards.

The main reason for this output is to allow the input data to be easily checked to ascertain whether or not all of these data cards are properly punched. It should be noted that number 8 and number 9 punched control cards are not included in this output. For cases in a run other than the first case, this output may be short since only a few modification cards may need to be inputted.

Output of Program Control Parameters

The next group of output for a case of a run, which may be printed, is that pertaining to the program control parameters. The output of this group is preceded by the heading, HELICOPTER AEROELASTIC STABILITY ANALYSIS INPUT PARAMETERS, and consists of one page of output. This output is not printed if the input number in Location 46 (corresponding to NPRS) of the control parameter input is equal to 0.0.

The first four lines of this output contain the values of various important integer parameters. The first line prints out the parameters NB, NBP, NBSEC, NFSEC, NBC, NHC, NFFB, and NAS - corresponding to the input in Locations 17, 18, 19, 20, 32, 35, 27, and 34, respectively. It should be noted that the integer parameters are obtained by truncation of the corresponding input numbers in floating point form. The second line contains the values of parameters NFLAP, NFEA, NCT, NCON, NES, MAXN, and FUA - corresponding to the input in Locations 28, 29, 30, 31, 39, 38, and 16, respectively. Also in this line, the value of parameter NSY - which is always equal to 100 - is printed. The next two lines provide the values of parameters NEGN, IPCT, NIT, MER, NORM, IREM, NEX, NPS, IG, IF, NLEL and MCTY - corresponding to input in Locations 48, 59, 60, 61, 62, 63, 64, 65, 67, 68, 170, and 172, respectively.

The next four lines of this output give the general description of the rotor configuration being considered. This output is self-explanatory, in that 17 short statements, which can be thought of as questions regarding the configuration, are followed by "yes" or "no" answers. For example, if ARTICULATED SYS NO is printed, the configuration has a flex-strap or bearingless rotor system with or without the effect of constraint unknowns included - depending on whether CONSTRAINT APP is followed by YES or NO, respectively. The "yes" or "no" answers are based upon particular input parameters. For example: BLADE PHASING YES corresponds to NBP input (in Location 18) being non-zero; DISK PLAN CALC YES corresponds to NPD (input in Location 15) being non-zero; and COLL SPRING IN is followed by YES when the input in Location 40 (MSC) is 1.0.

The statements answered by YES or NO are FLAP HINGE, CONTROL TORQUE EQ, PITCH BEARING, ARTICULATED SYS, CONSTRAINT

APPLIED, FUSELAGE, BLADE PHASING, AERODYNAMICS, GIMBALLED SYSM., VARIABLE OUTPUT, DISK PLANE CALCULATIONS, FUSELAGE AERO., SWASHPLATE IN, COLLECTIVE SPRING IN, VI (induced velocity) DISTRIBUTION USED, SEARCH OPTION, SHAPE PUNCHED, and LEAD-LAG HINGE. Lower case letters contained above have been added for clarification of the statements and are not included in the output. It should be noted that GIMBALLED SYSM. YES corresponds to either a gimballed or a teetering rotor system, with the specific type determined by the value printed for NBC.

The next set of output in this group consists of seven lines of output describing ROTOR SPEED, VEL. OF SOUND, AIR DENSITY, CRAFT VEL., ANGLE CHI, ANGLE ZETA, THETA GRAV., PSI GRAVITY, COEF. C(K), DELTA3, ALPHA1, COLL. ANGLE, GR RATE STEP, FREQ. STEP, GRAV. ACC., SHAFT FLEX., STR. DAMPING, F/A CYCLIC, LONG CYCLIC, DAMPING OUT, SPUR LENGTH, SPUR EIY, SPUR EIZ, COE1, COE2, and COE3 - corresponding to input in Locations 2, 3, 4, 5, 6, 7, 13, 14, 8, 9, 10, 11, 69, 70, 12, 45, 165, 167, 168, 171, 173, 174, 175, 176, 177, and 178, respectively.

The next few lines of output give the information regarding the control rod and its attachment to the blade. If the rotor configuration is an articulated rotor system (MFLEX=0) this output consists of the parameters PHIM(MS), AKCI(MS), TAU(MS), SMLA(MS), and DMS(MS) for values of MS from 1 to NB - corresponding to input in Locations 71-76, 77-82, 83-88, 97-102, 103-108, respectively. But if the rotor configuration is a flexstrap rotor system (MFLEX = 1 and MCTY = 0), this output describes the parameters PHIM(MS), AMKC(MS), CTAU(MS), PHIC(MS), THEC(MS), AL12(MS), and ALSTH(MS), for values of MS from 1 to NB - corresponding to input in Locations 71-76, 135-140, 141-146, 147-152, 153-158, 159-164, and 195-200, respectively, and the eight parameters of the PHOTT(I) array corresponding to Locations 179-186. If the rotor configuration is a bearingless rotor system (MFLEX = 1 and MCTY = 1) the same parameters described for a flexstrap configuration are described plus the eight parameters of the PHWTT(I) array corresponding to Locations 187-194.

Information concerning the control system ring and its support system is printed next if there is a control system included (NSP \neq 0). The first line of this part of the output describes the control system ring; torsional stiffness, bending stiffness, mass, and radius; in that order. The next output consists of the spring-damper unit support

parameters ECHI(J), AK(J), AC(J), and BJ(J) for values of J from 1 to NES - corresponding to the input in Locations 117-120, 121-124, 125-128, and 129-132, respectively. In the last line of this part of the output, the collective spring-damper unit parameters CAPK and CAPC are printed, which correspond to the input in Location 133 and 134, respectively. It should be noted that these last two parameters are outputted independent of the value of MSC, which specifies whether or not the collective base plate support exists.

The last lines of output on this page depict the starting eigenvalues (both real and imaginary parts) for the normal iteration procedure and correspond to input in Locations 49-58. One eigenvalue is printed on each line, such that the number of lines printed is equivalent to the values of parameter NEGN. If the search option is employed, only one eigenvalue is outputted, corresponding to the starting eigenvalue for the range of search.

Output of Aerodynamic Coefficient Tables

If C-81 aerodynamic coefficient tables are inputted into the computer program, these tables, including the cards denoting the number of Mach numbers and angles of attack, are printed out essentially the same as inputted. The only difference is that the angle of attack value is outputted in a lX, F6.1 format instead of F7.3 (as inputted) to avoid dropping the first character of the angle of attack value. It is to be noted that if several aerodynamic coefficient data tables are inputted only those of the C-81 format will be outputted. Preceding this output is a message FOR ISER EQUAL 1 ONLY CCS ARRAYS ARE INPUTTED which is printed if ISER is equal 1 or 81. If ISER is equal 81 this message should be ignored.

Output of Sectional Aerodynamic Coefficients and Steady Loads

This output follows the previous group and consists of the printout of two groups of intermixed output. The first group consists of the steady forces and moments acting on the next inboard section from the section for which they are outputted. Prior to the first force and moment output the title STEADY FORCES AND MOMENTS is outputted followed by the heading SECTION, N, VY, VZ, T, MZ, MY spaced such that

the corresponding values fall directly underneath. The positive direction of these variables corresponds to that defined for similar variables in the blade shape vector. The information is outputted for every blade section. The second group consists of the aerodynamic variables for each aerodynamic load point position. This output is not printed out if aerodynamics are not included; that is, if the input variable MAER (Location 26) is equal to zero. The size of this output for each aerodynamic blade section is dependent on the number of azimuthal steps (NAS) used for representing the blade aerodynamics. The aerodynamic variables printed for each blade aerodynamic section at various azimuthal positions of the reference blade and for each fuselage aerodynamic section are, in the order of printing: angle of attack (ALPHZ), Mach number (AMACZ), lift coefficient (CSUBL), drag coefficient (CSUBD), pitching moment coefficient (CSUBL), lift curve slope (CLALP), drag curve slope (CDALP), and pitching moment curve slope (CMALP) where the aerodynamic coefficients printed are two-dimensional. The heading, AERO COEFFICIENTS, and the column headings corresponding to the variables specified above in parentheses precede the numerical output in tabular form. This aerodynamic output for a blade section precedes that of the blade section steady loads and moments.

The order in which these numbers are outputted for a blade section will be briefly noted. The first line of aerodynamic output is for the blade aerodynamic section at the first azimuthal position (reference position) of the blade (corresponding to ψ_1 which equals zero). The next line of aerodynamic output is for the blade aerodynamic section at the next azimuthal position of the blade (corresponding to ψ_2 which equals $2\pi/\text{NAS}$). Additional lines are printed corresponding to the stepped azimuthal positions of the blade aerodynamic section until all azimuthal positions have been considered. It should be noted that aerodynamic coefficients are printed only for those blade sections having aerodynamics associated with them. Similar aerodynamic coefficient information for support structure aerodynamic sections is outputted (there being only one line per support structure aerodynamic section).

Output of Section Data

The input for the blade and support structure sections is printed on the next four pages of output in the form of four tables. The numbers printed in these tables correspond to that stored in locations 201 - 2200 as described in the section pertaining to blade/support structure section input. Preceding the first table, the heading, SECTION INPUT DATA, is printed followed by another heading specifying the order of the application of structural sections of the blade and fuselage - which is 'ORDER OF SECTIONS: BLADE TIP (SECTION 1) TO HUB, THEN FROM FREE END OF FUSELAGE (SECTION 1) TO HUB'.

On the first page of this output, the section number, the values of parameters M1 - M7 which denote the transfer matrices associated with a section, and the values of the parameters m , ϵ_{ry} , h_{rx} , h_{ry} , I_x , and I_y for each section are outputted under similar column headings. Section parameters described on page two for each section are, in order: I_z , ϕ , θ , ψ , L_a , ψ_f , θ_f , and δ , where these parameters are outputted under similar column headings. Page three of the output describes c , v_i , xxxx (GJcs), L_e , EI_x , EI_y , EI_z , and P_T for each section where these parameters are outputted under similar column headings. P_T is the steady centrifugal force acting on the section due to rotation of the blade and depends upon the magnitude and distribution of masses outboard of the blade section. It may be noted that P_T for fuselage sections is zero. The fourth group of output describes, in order: δ_x , δ_y , δ_z , N_0 , Vy_0 , Vz_0 , K_x^{-1} , K_y^{-1} , K_z^{-1} , τ_x , τ_y , τ_z , and K_s for each section with the values given under column headings similar to the variables. The values for N_0 , Vy_0 , and Vz_0 will always be 0.0 since these are the steady forces now determined internally in the program.

This tabular form of output of blade and support structure section data provides a convenient form for checking the sectional input to ascertain if the blade and support structure representation is as desired.

Output of Search Option Information

This set of output provides information regarding the value of the final matrix determinant for each trial eigenvalue specified in the use of the search option procedure by input of the initial stability and frequency values, the size of the stability and frequency steps, and the number of stability and frequency steps.

For each trial eigenvalue, a set of three numbers in double precision (first and last numbers being complex) are printed, which are components of the resulting final governing matrix determinant. The numbers printed with descriptive labels are the program variables in the order DTPHAS, DTLG10, and DPIVOT: where the determinant $\bar{\Delta}$ is defined as

$$\bar{\Delta} = \text{DTPHAS} * \text{DPIVOT} * 10 \quad (\text{DTLG10})$$

The quantity DTLG10 is representative of a factor removed from the determinant in the process of reducing the determinant matrix to the one which has diagonal elements with a complex absolute value of 1 and DTPHAS and DPIVOT are complex numbers resulting from obtaining the determinant of this resultant matrix.

This information is followed by the output of the trial (initial) search option eigenvalue and the value of the determinant with descriptive labels. It is to be noted that due to size limitations, a factor (multiples of 10^{**20}) is removed from the calculation of the determinant to prevent overflow problems. Thus, for purposes of output the matrix determinant is defined by the equation

$$\bar{\Delta} = \text{DTPHAS} * \text{DPIVOT} * 10 \quad (\text{DTLG10} - \text{IFAC} * 20.)$$

where IFAC is the integer number of times 20 can be divided into DTLG10. Thus, a factor $10^{**(\text{IFAC} * 20.)}$ has been removed from the determinant before printout. When the above information has been outputted for all trial eigenvalues the possible starting eigenvalues that are obtained will be used in the iteration procedure. However, if there are not any possible starting eigenvalues in the region scanned, then a message that there are no eigenvalues in that region is outputted.

The main purpose of the output provided for the search option procedure is to allow the user to ascertain if possible

roots may have been missed or if there may be an eigenvalue in a neighboring area based on trends observed in the determinant values as a function of stability and frequency values.

Output of Iteration Scheme Steps

This set of output is the pertinent information regarding the iterational procedure such that the program behavior from one iteration to the next can be observed.

The first line that is printed under this group of output by the program is as follows:

xxx VALUES OF UPDATED LAMBDA MATRIX

where in place of the xxx the actual number of values is printed. The numbers printed after this heading are the trial eigenvector quantities which are the approximations to the non-zero solution unknowns. It is to be noted that these are initially assumed to be unity and are updated by the correction quantities after each iteration and printed. Once the program has converged to an eigenvalue, there should be no significant change in these quantities. Therefore, the printout of this information provides an additional check on the program convergence to an eigenvalue. The initial set of trial eigenvector quantities is followed by the printout of the three determinant components, DTPHAS, DTLG10, and DPIVOT, for the starting eigenvalue. The functionality of these variables was discussed in detail in the previous output section. After the output of this information on the first iteration (ITI = 0), the initial eigenvalue and corresponding modified (i.e., a factor removed as discussed in the previous section) determinant value are printed.

The updated LAMBDA MATRIX for the second iteration (ITI = 1) is then printed, followed by the listing of determinant components (DTPHAS, DTLG10, and DPIVOT) for the second trial eigenvalue. The eigenvalue and modified determinant value on the second iteration are not printed, but are outputted as the LAST values on the third iteration.

The output for each iteration above the second iteration (ITI > 1) has the following form:

1. Updated LAMBDA MATRIX is printed.
2. The three determinant components (DTPHAS, DTLG10, and DPIVOT) are listed.
3. The three numbers describing the last modified determinant value, current modified determinant value, and eigenvalue convergence are printed.
4. The three numbers stating the last eigenvalue, current eigenvalue, and next eigenvalue are printed.

The three determinant components (2.) correspond to the CURRENT eigenvalue and determinant. The heading LAST refers to the values on the previous iteration. The NEXT eigenvalue is based on the interpolation of CURRENT and LAST values. It is to be noted that the possibility of the value of IFAC changing from one eigenvalue to another has been compensated for in the interpolation scheme. The convergence value is based on the change of eigenvalue from the current eigenvalue to the next eigenvalue relative to the current eigenvalue. Also, it may be noted during successive iterations that the normalized quantity in the updated LAMBDA MATRIX remains equal to one.

As the program iterates to a solution eigenvalue, the current modified determinant value will become smaller and approaches zero in the limit (at an eigenvalue). Furthermore, the difference between the current eigenvalue and the next eigenvalue, as predicted by the program, will approach zero. The last iteration will show eigenvalue convergence being less than $(.1) ** \text{MER}$, indicating that the eigenvalue has converged to within the prescribed accuracy. (Note that MER is the input parameter in Location number 61.) If this is not so, then the program ran out of iterations (prescribed in the input by NIT). As an example, consider the following sample printout of the last step of an iteration where the following is outputted:

```

LAST DETERMINANT VALUE =-0.688730D-06-0.344391D-06
CURRENT DETERMINANT VALUE =-0.570891D-09 0.161642D-09
EIGEN VALUE CONVERGENCE = 0.104667E-08

```

```

LAST EIGEN VALUE =-0.250890D 02 0.177439D 03
CURRENT EIGEN VALUE =-0.250888D 02 0.177439D 03
NEXT EIGEN VALUE =-0.250888D 02 0.177439D 03

```

Here, it is obvious that the eigenvalue has converged to a value of $(-25.0888+177.439i)$, rad/sec, and the last eigenvalue convergence is less than $(.1)**MER$ where MER is taken as 7.

After the program has converged to an eigenvalue, or if it has reached its upper limit on the number of iterations, then the following title for the shape vector output group (discussed in next section) is printed out:

AEROELASTIC STABILITY
STATE VECTORS

Shape Vector Printed Output

The shape vectors for the blades and fuselage are computed and printed after the eigenvalue has converged to a solution eigenvalue or the maximum number of iterations allowed has occurred. The shape vectors are printed out after the last iteration whether or not convergence has been obtained because quite often, even though the eigenvalue convergence is not within the prescribed accuracy, the convergence may be very close to that desired and the shape vectors may be quite satisfactory if they satisfy the boundary or matching conditions at the hub. It should be noted that the degree to which the boundary conditions are satisfied and the degree of convergence obtained determine the validity of the results of a run. The main reason a converged eigenvalue for a specific case may not be obtained before the number of allowed iterations has occurred is that the starting eigenvalue may not be sufficiently close to an actual solution eigenvalue of the system of interest. Quite often fairly good guesses are needed to assure that the starting eigenvalues converge to the proper solution eigenvalues.

The printout of the swashplate (control system) displacements and the shape vectors (or mode shapes) of the blades and support structure will be presented assuming $\pm 1/\text{rev}$ interharmonic coupling ($NHC = 1$) to be involved in the program run. The swashplate ring deflections, if $NSP \neq 0$, are printed out following a general heading, SWASHPLATE DEFLECTIONS, and separated into sets corresponding to a particular shifted frequency by headings of the form, p PER REV., where p goes from a value of $+NHC$ to $-NHC$. It should be noted that a positive value of p corresponds to an upward

shift in the frequency (for example, if p is equal to +1 the output corresponds to behavior at 1/rev above that of the frequency obtained on solution). Under each of these headings are printed the corresponding p/rev frequency shifted Fourier spatial harmonics of the swashplate motion in the order of the harmonics going from +NMAX to -NMAX, where the particular spatial harmonic is specified (with the opposite sign) in the printout. Thus, for $\text{NHC} = 1$, the swashplate information is printed out in the following order:

1. +NMAX to -NMAX spatial harmonics of swashplate displacement for $p = 1$ corresponding to a +1/rev motion relative to the main frequency
2. +NMAX to -NMAX spatial harmonics of swashplate displacement for $p = 0$ corresponding to a motion at the main frequency
3. +NMAX to -NMAX spatial harmonics of swashplate displacement for $p = 1$ corresponding to a -1/rev motion relative to the main frequency

The next set of output is the printout of the blade shape vectors. The shape vector printout for $\text{NHC} = 1$ can be listed in the order of output as

1. heading, BLADE 1 +1 PER REV SHAPE VECTOR (LOCAL AXIS SYS.)
2. +1/rev mode shapes for blade number 1 in the local section coordinate systems where k is -1 but motion is 1/rev above the main frequency
3. heading if NPD is 1 (corresponding to input Location 15), DISK PLANE AXIS SYSTEM
4. +1/rev mode shapes for blade number 1, if NPD is 1, in the disk plane axis system (rotating hub coordinate system) where k is -1 but motion is 1/rev above main frequency

5. heading if NPD is 1 (corresponding to input Location 15), DISK PLANE AXIS SYSTEM - POLAR COORDINATES AMPLITUDE AND PHASE ANGLE
6. +1/rev disk plane mode shapes for blade number 1, if NPD is 1, in polar form, where motion is 1/rev above main frequency
7. steps 1 thru 6 are repeated for blade number 2 (only if $NB > 2$), blade number 3 (only if $NB \geq 3$), etc.
8. steps 1 thru 7 are repeated for the blade mode shapes at the main frequency ($p = 0$)
9. steps 1 thru 7 are repeated for the blade mode shapes 1/rev below the main frequency ($p = -1$)

The blade shape vectors are printed out for all the blade sections with four sets of three columns each for the local coordinate system values and disk plane values with the sets in the following order:

1. radial, inplane, and vertical blade deflections
2. torsional, vertical bending, and inplane bending slopes
3. radial, inplane shear, and vertical shear forces
4. torsional, vertical bending, and inplane bending moments

Each column has a descriptive heading specifying the variable listed in the column and the positive convention of the variable. The state variable values for a blade section are the values of the variables acting on the outboard end of the next inboard section. If NHC is equal to zero only 0/rev shape vectors are involved and printed.

The blade shape vector output is followed by the printout of the support structure mode shapes if the configuration includes a support structure (i.e., MFUS = 1 corresponding to input Location 33). This printout can be listed in the order of output as

1. heading, +1 PER REV SHAPE VECTORS ON THE FUSELA. STRUCTURE (LOCAL AX)
2. +1/rev mode shapes for the support structure in local section coordinate systems where k is -1 but motion is 1/rev above the main frequency
3. heading if NPD is 1 (corresponding to input Location 15), FUSELAGE REFERENCE AXIS SYSTEM
4. +1/rev mode shapes for the support structure relative to the fixed support structure coordinate system where motion is 1/rev above main frequency
5. heading if NPD is 1 (corresponding to input Location 15), FUSELAGE REFERENCE AXIS SYSTEM - POLAR COORDINATES
6. +1/rev mode shapes for the support structure relative to the fixed support structure coordinate system in polar form, if NPD is 1, where motion is 1/rev above main frequency
7. steps 1 thru 6 are repeated for the support structure mode shapes at the main frequency
8. steps 1 thru 6 are repeated for the support structure mode shape 1/rev below the main frequency

The support structure shape vectors are also printed out for all support structure sections with four sets of three columns each for the local coordinate system values and fixed support structure coordinate system values, with the sets in the following order:

1. axial, vertical, and lateral support structure deflections
2. torsional, lateral bending, and vertical bending slopes
3. axial, vertical shear, and lateral shear forces
4. torsional, lateral bending, and vertical bending moments

Each column has a descriptive heading specifying the variable listed in the column and the positive convention of the variable. It should be noted that in these headings the notation, INPLANE, has been used to denote the lateral direction. The positive convention for the twist in the reference axis system is for the top of the structure to the left. Basically, the variable values outputted for a support structure section are the values for the variables acting on the hubward end of the section in the positive axis directions. For both the support structure and blade mode shape output, the section numbers are also printed in an adjacent column to allow easy association of results with the section to which they correspond. If NHC is equal to zero only the 0/rev shape vectors are printed.

The printout of the blade shape vectors and the support structure shape vectors, if a support structure is modelled, is the last printed output associated with the results obtained for a given starting eigenvalue. From this shape vector output the convergence of the eigenvalue could be verified, if desired. Normally, when the program converges to an eigenvalue, all of the boundary conditions at the hub are satisfied. One of these boundary conditions can be easily checked from the printed output. The blade deflection denoted by UBZ at the blade root (hub section) of each blade in the local coordinate system (or disk plane, since the variable values at hub section should be independent of coordinate system) should be equal to the support structure deflection denoted by UBX at the last section (hub section) of the support structure in the local support structure axis system. The remaining blade root variables and support structure variables at the hub may also be checked for matching of boundary conditions, but the equations describing the hub boundary conditions involve combinations of $-1/\text{rev}$,

0/rev, and +1/ rev frequency-shifted shape vectors due to the transformations from the blade rotating system to hub fixed system and vice versa.

Shape Vector Punched Output

The computer program provides the blade and support structure state variable information in punched card form if IPU (corresponding to array Location 166) is equal to 1. The punched output includes all state variables (shifted and unshifted) determined for all blade sections of all blades and all support structure sections. The punched data is generated with the use of a (I5, 3(1X,2(1PE12.4))) format such that each card contains the section number and three state variable values in the same order as they are printed. For every row of printed state variable output a punched card containing the same information is generated. The order in which the state variable information is punched coincides with the order in which all of the blade and support structure state variables are printed. This program capability was added to allow the state variable information to be available after a run for subsequent use.

SAMPLE INPUT

Bearingless Rotor Case

2011	1.0				
2514	1.0	1.0	0.0	1.0	
3014	1.0	1.0	0.0	1.0	
3514	1.0	1.0	0.0	1.0	
4014	1.0	1.0	0.0	1.0	
4514	1.0	1.0	0.0	1.0	
5014	1.0	1.0	0.0	1.0	
5514	1.0	1.0	0.0	1.0	
6014	1.0	1.0	0.0	1.0	
6514	1.0	1.0	0.0	1.0	
7014	1.0	1.0	0.0	1.0	
7514	1.0	1.0	0.0	1.0	
8014	1.0	1.0	0.0	1.0	
8514	1.0	1.0	0.0	1.0	
9014	1.0	1.0	0.0	1.0	
9515	1.0	1.0	0.0	1.0	1.0
10023	1.0	0.0	1.0		
10521	1.0				
11071	1.0				
11514	1.0	1.0	0.0	1.0	
12014	1.0	0.0	0.0	1.0	
12514	1.0	0.0	0.0	1.0	
13014	1.0	0.0	0.0	1.0	
13514	1.0	0.0	0.0	1.0	
14014	1.0	0.0	0.0	1.0	
14514	1.0	0.0	0.0	1.0	
15014	1.0	0.0	0.0	1.0	
15514	1.0	0.0	0.0	1.0	
16014	1.0	0.0	0.0	1.0	
16514	1.0	1.0	0.0	1.0	
17021	1.0				
2085	.0070752	-.61591	193.37000	.60591	.024216
2585	.0083197	-.61591	183.70150	.60591	.048432
3085	.0080135	-.61591	174.03300	.60591	.048432
3585	.0080135	-.61591	164.36450	.60591	.048432
4085	.0078979	-.61591	154.69600	.60591	.048432
4585	.0077380	-.61591	145.02750	.60591	.048432
5085	.0077380	-.61591	135.35900	.60591	.048432
5585	.0077380	-.61591	125.69050	.60591	.048432
6085	.0077380	-.61591	116.02200	.60591	.048432
6585	.0077380	-.61591	106.35350	.60591	.048432
7085	.0134267	-.61591	96.68500	.60591	.053236
7585	.0077380	-.61591	87.01650	.60591	.048432
8085	.0089997	-.18618	77.34800	.60591	.045456
8585	.0116129	-.01591	67.67950	.60591	.031488
9085	.0129953	-.11217	58.01100	.60591	.019826
9585	.032923	0.0	51.47500	0.0	.047789
11585	.0089010	0.0	48.34250	0.0	.034662
12085	.0057872	0.0	43.50825	0.0	.030318
12585	.0020748	0.0	38.67400	0.0	.012693

13085	.0020594	0.0	33.83975	0.0	.010097
13585	.0020897	0.0	29.00550	0.0	.007622
14085	.0021199	0.0	24.17125	0.0	.007141
14585	.0022519	0.0	19.33700	0.0	.007547
15085	.0044613	0.0	14.50275	0.0	.013266
15585	.0154515	0.0	9.66850	0.0	.026028
16085	.0444968	0.0	4.83425	0.0	.382518
16585	.0260251	0.0	0.0	0.0	.340449
2141	.024216				
2641	.048432				
3141	.048432				
3641	.048432				
4141	.048432				
4641	.048432				
5141	.048432				
5641	.048432				
6141	.048432				
6641	.048432				
7141	.053236				
7641	.048432				
8141	.045456				
8641	.031485				
9141	.019826				
9641	.067785				
11641	.034662				
12141	.030318				
12641	.012693				
13141	.010097				
13641	.007622				
14141	.007141				
14641	.007547				
15141	.013266				
15641	.026028				
16141	.382518				
16641	.340449				
2764	9.6685	.1360000E+07	.2380000E+07	.5940000E+08	
3264	9.6685	.1360000E+07	.2380000E+07	.5940000E+08	
3764	9.6685	.1360000E+07	.2380000E+07	.5940000E+08	
4264	9.6685	.1360000E+07	.2380000E+07	.5940000E+08	
4764	9.6685	.1360000E+07	.2380000E+07	.5940000E+08	
5264	9.6685	.1360000E+07	.2380000E+07	.5940000E+08	
5764	9.6685	.1360000E+07	.2380000E+07	.5940000E+08	
6264	9.6685	.1360000E+07	.2380000E+07	.5940000E+08	
6764	9.6685	.1360000E+07	.2380000E+07	.5940000E+08	
7264	9.6685	.1360000E+07	.2380000E+07	.5940000E+08	
7764	9.6685	.1361000E+07	.2380000E+07	.5940000E+08	
8264	9.6685	.1538300E+07	.2550300E+07	.5887500E+08	
8764	9.6685	.2175500E+07	.3974100E+07	.5496800E+08	
9264	9.6685	.3396600E+07	.5684300E+07	.5047400E+08	
9764	6.5360	.4533300E+07	.1076800E+08	.6460600E+08	

10264	2.4750	.1182500E+08	.1574600E+09	.3953500E+09		
11764	0.6575	.3929400E+08	.1210700E+09	.6106000E+09		
12264	4.83425	.4210500E+06	.1588500E+08	.1371800E+09		
12764	4.83425	.5850400E+05	.5273700E+07	.4555200E+08		
13264	4.83425	.4905900E+05	.5869500E+07	.3737800E+08		
13764	4.83425	.4942800E+05	.7047600E+07	.3734500E+08		
14264	4.83425	.4998300E+05	.8321000E+07	.3740700E+08		
14764	4.83425	.5000000E+05	.9934100E+07	.3870500E+08		
15264	4.83425	.7208000E+05	.1474200E+08	.4924300E+08		
15764	4.83425	.3156800E+06	.2848200E+08	.8572300E+08		
16264	4.83425	.2375500E+07	.4689400E+08	.1788100E+09		
16764	4.83425	.4777400E+08	.3999100E+09	.1346800E+10		
9891	.60591					
12251	.1668200E+08					
12751	.7044400E+07					
13251	.6361000E+07					
13751	.6419300E+07					
14251	.6488600E+07					
14751	.6874800E+07					
15251	.9400200E+07					
15751	.1620300E+08					
16251	.2698000E+08					
21	44.5059					
151	1.0					
173	1.0	4.0	31.0			
242	1.0	1.0				
311	19.0					
461	1.0					
595	50.0	15.0	7.0	5.0	6.0	
641	6.0					
1351	.6035700E+03					
1471	-.785398					
1531	.390008					
1724	1.0	2.58226	.2000000E+07	.2000000E+07		
1763	0.0	-263.08173	0.0			
1794	39.00573	.2585000E+07	.1593000E+07	.6930000E+06		
1834	3.1415927	.0736455	0.0	.2681840E+07		
1871	7.43277					
1912	1.064508	-.0101443				
121	386.088					
12021	1.0					
100111.0						
115410.0						
125211.0						
130211.0						
135211.0						
140211.0						
145211.0						
150211.0						
155211.0						

160211.0			
9581.0308687			
10081.0046190			
11581.0066174			
1010148.3425			
1060148.3425			
1110148.3425			
09623.077161	0.0	.077161	
11623.025285	0.0	.025285	
10161-.01309			
11661-.01309			
166610.0			
102643.1325	.1385800E+08	.1481100E+09	.4269400E+09
117640.0	0.0	0.0	0.0
01791 38.41774			
01841 .088644			
1591 10.56			
1951 3.06			
2531 1.0			
3031 1.0			
3531 1.0			
4031 1.0			
4531 1.0			
5031 1.0			
5531 1.0			
6031 1.0			
6531 1.0			
7031 1.0			
7531 1.0			
8031 1.0			
8531 1.0			
2682 1.0	9.6685		
3182 1.0	9.6685		
3682 1.0	9.6685		
4182 1.0	9.6685		
4682 1.0	9.6685		
5182 1.0	9.6685		
5682 1.0	9.6685		
6182 1.0	9.6685		
6682 1.0	9.6685		
7182 1.0	9.6685		
7682 1.0	9.6685		
8182 1.0	9.6685		
8682 1.0	8.34075		
2722 3.26591	10.64		
3222 3.26591	10.64		
3722 3.26591	10.64		
4222 3.26591	10.64		
4722 3.26591	10.64		
5222 3.26591	10.64		

5722	3.26591	10.64	
6222	3.26591	10.64	
6722	3.26591	10.64	
7222	3.26591	10.64	
7722	3.26591	10.64	
8222	3.26591	10.64	
8722	3.26591	10.64	
32	.1340300E+05	.1146300E=06	
63	1.570796	0.0	.8
261	1.0		
341	16.0		
2741	513.97		
3241	513.97		
3741	513.97		
4241	513.97		
4741	513.97		
5241	513.97		
5741	513.97		
6241	513.97		
6741	513.97		
7241	513.97		
7741	513.97		
8241	513.97		
8741	513.97		
1672	.001641	.005009	
2111	=0.1853881E+01		
2611	=0.1713217E+01		
3111	=0.1571979E+01		
3611	=0.1429330E+01		
4111	=0.1284368E+01		
4611	=0.1136753E+01		
5111	=0.9864085E+00		
5611	=0.8338016E+00		
6111	=0.6797164E+00		
6611	=0.5250247E+00		
7111	=0.3709548E+00		
7611	=0.2189657E+00		
8111	=0.7083029E=01		
8611	0.7135791E=01		
9111	0.2053451E+00		
9611	=0.3157269E+00		
10111	=0.2750667E+00		
10611	=0.2750667E+00		
11111	=0.2750667E+00		
11611	=0.2750667E+00		
12111	=0.2211854E+00		
12611	=0.1673041E+00		
13111	=0.1237797E+00		
13611	=0.8025533E=01		
14111	=0.5228527E=01		

14611=0,2431520E=01
15111=0,1341677E=01
15611=0,2518338E=02
16111=0,1259169E=02
2153=0,1454767E=01-0,2782708E=01 0,9605795E=01
2653=0,1454767E=01-0,2782708E=01 0,1027852E+00
3153=0,1460704E=01-0,2841647E=01 0,1093680E+00
3653=0,1475289E=01-0,2985435E=01 0,1158658E+00
4153=0,1499213E=01-0,3192549E=01 0,1223182E+00
4653=0,1526638E=01-0,3422624E=01 0,1287245E+00
5153=0,1554871E=01-0,3638931E=01 0,1350722E+00
5653=0,1578262E=01-0,3811616E=01 0,1413519E+00
6153=0,1593547E=01-0,3933148E=01 0,1475481E+00
6653=0,1599818E=01-0,3991526E=01 0,1536441E+00
7153=0,1593390E=01-0,3986597E=01 0,1596156E+00
7653=0,1571872E=01-0,3908069E=01 0,1653896E+00
8153=0,1532025E=01-0,3751607E=01 0,1711283E+00
8653=0,1470527E=01-0,3519034E=01 0,1771091E+00
9153=0,1385723E=01-0,3221410E=01 0,1833672E+00
9653=0,1297938E=01-0,3163933E=01 0,2288158E+00
10153=0,1297938E=01-0,3163933E=01 0,2285851E+00
10653 0,0 0,0 0,0
11153 0,0 0,0 0,0
11653=0,1297938E=01-0,3163933E=01 0,2285851E+00
12153=0,1114528E=01-0,2767443E=01 0,2341551E+00
12653=0,1114528E=01-0,2767443E=01 0,2413099E+00
13153=0,9003095E=02-0,2246325E=01 0,2455308E+00
13653=0,9003095E=02-0,2246325E=01 0,2497092E+00
14153=0,5785745E=02-0,1481477E=01 0,2431120E+00
14653=0,5785745E=02-0,1481477E=01 0,2365021E+00
15153=0,2254416E=02-0,6113276E=02 0,2246698E+00
15653=0,2254416E=02-0,6113276E=02 0,2115865E+00
16153=0,5209367E=03-0,1703219E=02 0,2106684E+00
1761 157,0
611 7,0
371 1,0
381 1,0
395 3,0 1,0 .015 6,1 .1000000E+10
441 .1000000E+10
971 6,5
1173 0,0 2,094395 4,188790
01213 .2359423E+05 .2359423E+05 .2359423E+05
1331 7363,5766
1351 .3846154E=05
623 8,0 9,0 9,0
601 12,0
2161 =,03061
2661 =,03061
3161 =,031258
3661 =,032839

4161 =.035118
 4661 =.037649
 5161 =.040028
 5661 =.041928
 6161 =.043264
 6661 =.043907
 7161 =.043853
 7661 =.042989
 8161 =.041268
 8661 =.038709
 9161 =.035435
 9661 =.034803
 10161 =.034803
 11661 =.034803
 12161 =.030441
 12661 =.030441
 13161 =.024709
 13661 =.024709
 14161 =.016297
 14661 =.016297
 15161 =.00672
 15661 =.00672
 16161 =.00187
 351 0.0
 651 2.0
 481 1.0
 491 =14.6
 541 48.6

8
 1
 81
 1
 2

180.	11	39	-180.						
	0.	0.3	0.4	0.5	0.6	0.7	0.75	0.8	0.85
	0.9	1.							
-180.	.04	.04	.04	.04	.04	.04	.04	.04	.04
	.04	.04							
-174.	.65	.65	.65	.65	.65	.65	.65	.65	.65
	.65	.65							
-170.	.65	.65	.65	.65	.65	.65	.65	.65	.65
	.65	.65							
-166.	.62	.62	.62	.62	.62	.62	.62	.62	.62
	.62	.62							
-133.	.865	.865	.865	.865	.865	.865	.865	.865	.865
	.865	.865							
-113.	.635	.635	.635	.635	.635	.635	.635	.635	.635
	.635	.635							
-58.	-.89	-.89	-.89	-.89	-.89	-.89	-.89	-.89	-.89

	-.89	-.89							
-38.	-1.14	-1.14	-1.14	-1.14	-1.14	-1.14	-1.14	-1.14	-1.14
	-1.14	-1.14							
-20.	-.98	-1.03	-1.07	-1.13	-1.14	-.868	-.86	-.845	-.74
	-1.01	-1.34							
-15.	-1.165	-1.225	-1.285	-1.23	-1.13	-.88	-.892	-.827	-.685
	-1.01	-1.34							
-13.5	-1.22	-1.28	-1.33	-1.26	-1.12	-.88	-.885	-.814	-.677
	-1.01	-1.34							
-12.	-1.095	-1.15	-1.2	-1.14	-1.092	-.86	-.877	-.791	-.655
	-1.01	-1.25							
-9.5	-.89	-.93	-.97	-.93	-1.05	-.83	-.85	-.77	-.62
	-.947	-1.1							
-5.5	-.47	-.493	-.51	-.55	-.585	-.64	-.688	-.65	-.58
	-.41	-.61							
-4.	-.312	-.33	-.335	-.36	-.39	-.41	-.5	-.475	-.28
	-.22	-.43							
-2.	-.1	-.107	-.105	-.115	-.12	-.12	-.127	-.16	-.165
	-.085	-.185							
-1.	0.0	0.0	0.0	0.0	.01	.025	.015	.05	-.005
	-.015	-.06							
0.0	.11	.11	.12	.13	.14	.17	.19	.24	.06
	.11	.06							
2.	.32	.332	.35	.375	.41	.475	.485	.37	.208
	.39	.3							
3.	.425	.443	.467	.5	.547	.62	.5	.415	.26
	.52	.426							
4.	.53	.55	.58	.62	.68	.675	.57	.475	.28
	.63	.545							
6.	.742	.772	.815	.868	.91	.767	.67	.51	.32
	.92	.787							
8.	.95	.995	1.035	1.115	.995	.856	.732	.545	.355
	1.19	1.03							
10.	1.165	1.22	1.26	1.27	1.075	.942	.791	.58	.393
	1.19	1.095							
11.	1.268	1.328	1.38	1.278	1.092	.985	.825	.6	.413
	1.19	1.095							
12.	1.375	1.44	1.42	1.25	1.113	1.028	.852	.615	.432
	1.19	1.095							
13.	1.48	1.55	1.395	1.225	1.13	1.069	.885	.633	.45
	1.19	1.095							
13.5	1.53	1.6	1.27	1.207	1.141	1.09	.9	.641	.461
	1.19	1.095							
14.	1.585	1.095	1.14	1.19	1.15	1.112	.913	.65	.47
	1.19	1.095							
15.	.915	.962	1.	1.192	1.17	1.155	.946	.668	.49
	1.19	1.095							
20.	1.	1.05	1.09	1.2	1.26	1.367	1.1	.76	.59
	1.19	1.095							
30.	1.135	1.135	1.135	1.135	1.135	1.135	1.135	1.135	1.135

58.	1.135 .89 .89	1.135 .89 .89	.89	.89	.89	.89	.89	.89	.89
113.	-.635 -.635	-.635 -.635	-.635	-.635	-.635	-.635	-.635	-.635	-.635
133.	-.865 -.865	-.865 -.865	-.865	-.865	-.865	-.865	-.865	-.865	-.865
158.	-1. -1.	-1. -1.	-1.	-1.	-1.	-1.	-1.	-1.	-1.
166.	-.72 -.72	-.72 -.72	-.72	-.72	-.72	-.72	-.72	-.72	-.72
170.	-.82 -.82	-.82 -.82	-.82	-.82	-.82	-.82	-.82	-.82	-.82
180.	.04 .04	.04 .04	.04	.04	.04	.04	.04	.04	.04
2									
180.		-180.							
11	45								
	0.0 0.9	0.4 1.0	0.5	0.6	0.65	0.7	0.75	0.8	0.85
-180.	.015 .015	.015 .015	.015	.015	.015	.015	.015	.015	.015
-175.	.04 .04	.04 .04	.04	.04	.04	.04	.04	.04	.04
-170.	.11 .11	.11 .11	.11	.11	.11	.11	.11	.11	.11
-164.	.22 .22	.22 .22	.22	.22	.22	.22	.22	.22	.22
-155.	.51 .51	.51 .51	.51	.51	.51	.51	.51	.51	.51
-130.	1.08 1.08	1.08 1.08	1.08	1.08	1.08	1.08	1.08	1.08	1.08
-100.	1.51 1.51	1.51 1.51	1.51	1.51	1.51	1.51	1.51	1.51	1.51
-90.	1.56 1.56	1.56 1.56	1.56	1.56	1.56	1.56	1.56	1.56	1.56
-80.	1.51 1.51	1.51 1.51	1.51	1.51	1.51	1.51	1.51	1.51	1.51
-60.	1.24 1.24	1.24 1.24	1.24	1.24	1.24	1.24	1.24	1.24	1.24
-40.	.9 .9	.9 .9	.9	.9	.9	.9	.9	.9	.9
-25.	.51 .51	.51 .51	.51	.51	.51	.51	.51	.51	.51
-8.	.0205 .193	.0205 .2175	.067	.136	.1375	.1405	.144	.17	.1818
-6.	.0154 .138	.0154 .1615	.03	.075	.0765	.079	.087	.115	.127
-4.	.0112 .083	.0112 .1062	.0115	.0141	.0163	.0182	.0287	.06	.0715

-3.	.0107 .0625	.0107 .089	.01095	.0123	.01335	.01445	.0117	.0322	.0448
-2.	.0104 .0425	.0104 .0715	.0104	.0105	.0104	.0107	.011	.0137	.0284
-1.	.0105 .0512	.0105 .0812	.0103	.01035	.01035	.0105	.0104	.0177	.0373
0.	.0106 .06	.0106 .091	.0102	.0102	.0103	.0103	.0131	.0287	.0445
1.	.0105 .079	.0105 .1105	.0103	.01035	.01075	.0119	.0217	.0474	.0631
2.	.0104 .0975	.0104 .1293	.0104	.0105	.0112	.0186	.0384	.0659	.082
3.	.0106 .1165	.0106 .1485	.0107	.0118	.015	.0298	.0569	.0848	.101
4.	.0108 .135	.0108 .167	.011	.0131	.0236	.0484	.0755	.1035	.1195
5.	.0114 .153	.0114 .186	.01175	.0191	.0345	.067	.0944	.122	.1383
6.	.012 .172	.012 .205	.0125	.0293	.053	.086	.113	.141	.157
7.	.01265 .1915	.01265 .223	.0133	.0426	.072	.1047	.1319	.16	.176
8.	.0133 .21	.0133 .2425	.0153	.061	.0905	.1235	.1505	.1788	.195
9.	.0137 .229	.0137 .261	.0212	.0798	.1092	.1422	.1693	.198	.2135
10.	.0153 .248	.0153 .28	.03	.0983	.1275	.1607	.188	.217	.232
11.	.0175 .266	.0175 .299	.0483	.118	.1465	.179	.207	.236	.2505
12.	.0205 .285	.0205 .318	.067	.1357	.165	.1985	.225	.256	.27
13.	.0282 .304	.0282 .3365	.0858	.1545	.1835	.217	.244	.275	.289
15.	.0465 .34	.0465 .374	.1233	.192	.221	.254	.281	.3135	.3265
30.	.66 .66	.66 .66	.66	.66	.66	.66	.66	.66	.66
50.	1.07 1.07	1.07 1.07	1.07	1.07	1.07	1.07	1.07	1.07	1.07
80.	1.5 1.5	1.5 1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5
90.	1.56 1.56	1.56 1.56	1.56	1.56	1.56	1.56	1.56	1.56	1.56
100.	1.51 1.51	1.51 1.51	1.51	1.51	1.51	1.51	1.51	1.51	1.51
120.	1.23 1.23	1.23 1.23	1.23	1.23	1.23	1.23	1.23	1.23	1.23
140.	.89 .89	.89 .89	.89	.89	.89	.89	.89	.89	.89

155.	.5	.5	.5	.5	.5	.5	.5	.5	.5
164.	.22	.22	.22	.22	.22	.22	.22	.22	.22
170.	.11	.11	.11	.11	.11	.11	.11	.11	.11
175.	.04	.04	.04	.04	.04	.04	.04	.04	.04
180.	.015	.015	.015	.015	.015	.015	.015	.015	.015
2									
180.		-180.							
11	44								
	0.	0.4	0.5	0.6	0.65	0.7	0.75	0.8	0.85
	0.9	1.0							
-180.	-.04	-.04	-.04	-.04	-.04	-.04	-.04	-.04	-.04
	-.04	-.04							
-172.	.37	.37	.37	.37	.37	.37	.37	.37	.37
	.37	.37							
-168.	.35	.35	.35	.35	.35	.35	.35	.35	.35
	.35	.35							
-164.	.39	.39	.39	.39	.39	.39	.39	.39	.39
	.39	.39							
-156.	.42	.42	.42	.42	.42	.42	.42	.42	.42
	.42	.42							
-150.	.445	.445	.445	.445	.445	.445	.445	.445	.445
	.445	.445							
-130.	.575	.575	.575	.575	.575	.575	.575	.575	.575
	.575	.575							
-115.	.6	.6	.6	.6	.6	.6	.6	.6	.6
	.6	.6							
-90.	.55	.55	.55	.55	.55	.55	.55	.55	.55
	.55	.55							
-60.	.4	.4	.4	.4	.4	.4	.4	.4	.4
	.4	.4							
-40.	.26	.26	.26	.26	.26	.26	.26	.26	.26
	.26	.26							
-30.	.18	.18	.18	.18	.18	.18	.18	.18	.18
	.18	.18							
-16.	.105	.105	.105	.105	.105	.105	.105	.105	.105
	.105	.105							
-4.	-.0075	-.0075	-.0065	-.0105	-.015	-.018	-.0175	.0238	.009
	.02	.005							
-3.	-.007	-.007	-.0067	-.0091	-.0112	-.015	-.0189	.009	.0055
	-.006	-.002							
-2.	-.007	-.0065	-.007	-.0075	-.0075	-.012	-.01	-.013	.004
	-.031	-.0095							
-1.	-.0075	-.0075	-.0075	-.0084	-.0084	-.0105	-.009	-.012	-.0275
	-.03	-.035							
0.0	-.008	-.008	-.008	-.009	-.009	-.009	-.008	-.033	-.0215

1.	-.0435	-.057							
	-.0085	-.0085	-.0083	-.009	-.0088	-.0075	-.0125	-.0505	-.0235
	-.05	-.0615							
2.	-.009	-.009	-.0087	-.009	-.0085	-.007	-.025	-.0515	-.037
	-.0505	-.068							
3.	-.0093	-.0093	-.009	-.0076	-.006	-.011	-.0415	-.0525	-.0375
	-.051	-.072							
4.	-.01	-.0095	-.0095	-.006	-.0035	-.0225	-.0525	-.057	-.06
	-.0655	-.0745							
5.	-.01	-.009	-.009	-.0024	-.005	-.04	-.0605	-.076	-.084
	-.0825	-.0805							
6.	-.0115	-.0075	-.0085	0.	-.015	-.0505	-.067	-.083	-.0975
	-.0935	-.085							
7.	-.013	-.0045	-.0078	.0005	-.031	-.058	-.0755	-.0785	-.09
	-.099	-.091							
8.	-.015	-.0005	-.007	-.011	-.0425	-.077	-.082	-.089	-.097
	-.104	-.096							
9.	.006	.006	-.006	-.0223	-.0525	-.082	-.089	-.094	-.102
	-.1088	-.1009							
10.	.0125	.0125	-.003	-.0315	-.0624	-.0869	-.096	-.0989	-.1068
	-.1137	-.1058							
11.	.0135	.0135	.001	-.0415	-.0723	-.0918	-.101	-.1028	-.1118
	-.1186	-.1106							
12.	.0035	.0035	.0035	-.0514	-.0822	-.0966	-.1058	-.1087	-.1167
	-.1235	-.1155							
13.	-.0185	-.0185	-.005	-.0613	-.0922	-.1017	-.1107	-.1137	-.1215
	-.1285	-.1205							
14.	-.0435	-.0435	-.0565	-.0612	-.1023	-.1065	-.1157	-.1186	-.1266
	-.1335	-.1255							
16.	-.102	-.102	-.0885	-.0912	-.1122	-.1165	-.1252	-.1285	-.1365
	-.143	-.1354							
25.	-.175	-.175	-.175	-.175	-.175	-.175	-.175	-.175	-.175
	-.175	-.175							
40.	-.29	-.29	-.29	-.29	-.29	-.29	-.29	-.29	-.29
	-.29	-.29							
60.	-.43	-.43	-.43	-.43	-.43	-.43	-.43	-.43	-.43
	-.43	-.43							
90.	-.58	-.58	-.58	-.58	-.58	-.58	-.58	-.58	-.58
	-.58	-.58							
115.	-.63	-.63	-.63	-.63	-.63	-.63	-.63	-.63	-.63
	-.63	-.63							
140.	-.555	-.555	-.555	-.555	-.555	-.555	-.555	-.555	-.555
	-.555	-.555							
160.	-.43	-.43	-.43	-.43	-.43	-.43	-.43	-.43	-.43
	-.43	-.43							
168.	-.38	-.38	-.38	-.38	-.38	-.38	-.38	-.38	-.38
	-.38	-.38							
172.	-.39	-.39	-.39	-.39	-.39	-.39	-.39	-.39	-.39
	-.39	-.39							
176.	-.28	-.28	-.28	-.28	-.28	-.28	-.28	-.28	-.28

180. -.28 -.28
 -.04 -.04 -.04 -.04 -.04 -.04 -.04 -.04
 -.04 -.04
 9

SAMPLE OUTPUT

Bearingless Rotor Case

LUC	NUM	VAL1	PRINT OUT OF INPUT DATA VAL2	VAL3	VAL4	VAL5
201	1	1.0000				
251	4	1.0000	1.0000	.0	1.0000	
301	4	1.0000	1.0000	.0	1.0000	
351	4	1.0000	1.0000	.0	1.0000	
401	4	1.0000	1.0000	.0	1.0000	
451	4	1.0000	1.0300	.0	1.0000	
501	4	1.0000	1.0000	.0	1.0000	
551	4	1.0000	1.0000	.0	1.0000	
601	4	1.0000	1.0000	.0	1.0000	
651	4	1.0000	1.0000	.0	1.0000	
701	4	1.0000	1.0000	.0	1.0000	
751	4	1.0000	1.0000	.0	1.0000	
801	4	1.0000	1.0000	.0	1.0000	
851	4	1.0000	1.0000	.0	1.0000	
901	4	1.0000	1.0000	.0	1.0000	
951	5	1.0000	1.0000	.0	1.0000	
1002	3	1.0000	1.0000	.0	1.0000	
1052	1	1.0000	.0	1.0000		1.0000
1107	1	1.0000				
1151	4	1.0000	1.0000	.0	1.0000	
1201	4	1.0000	.0	.0	1.0000	
1251	4	1.0000	.0	.0	1.0000	
1301	4	1.0000	.0	.0	1.0000	
1351	4	1.0000	.0	.0	1.0000	
1401	4	1.0000	.0	.0	1.0000	
1451	4	1.0000	.0	.0	1.0000	
1501	4	1.0000	.0	.0	1.0000	
1551	4	1.0000	.0	.0	1.0000	
1601	4	1.0000	.0	.0	1.0000	
1651	4	1.0000	1.0000	.0	1.0000	
1702	1	1.0000				
200	5	.70752E-02	.01591	193.37	.0591	.24214E-01
250	5	.01197E-02	.01591	193.70	.0591	.68432E-01
300	5	.01135E-02	.01591	174.03	.0591	.68432E-01
350	5	.01135E-02	.01591	184.36	.0591	.68432E-01
400	5	.78979E-02	.01591	158.70	.0591	.68432E-01
450	5	.77380E-02	.01591	159.03	.0591	.68432E-01
500	5	.77380E-02	.01591	135.36	.0591	.68432E-01
550	5	.77380E-02	.01591	135.49	.0591	.68432E-01
600	5	.77380E-02	.01591	116.02	.0591	.68432E-01
650	5	.77380E-02	.01591	106.35	.0591	.68432E-01
700	5	.13427E-01	.01591	96.685	.0591	.53236E-01
750	5	.77380E-02	.01591	87.016	.0591	.68432E-01
800	5	.09997E-02	.18618	77.348	.0591	.68432E-01
850	5	.11613E-01	.15010E-01	67.680	.0591	.31485E-01
900	5	.12955E-01	.11217	58.011	.0591	.19026E-01
950	5	.24933E-01	.0	51.475	.0	.67795E-01
1150	5	.09010E-02	.0	48.342	.0	.34622E-01
1200	5	.57872E-02	.0	43.508	.0	.59318E-01
1250	5	.20748E-02	.0	38.674	.0	.12693E-01
1300	5	.60596E-02	.0	33.840	.0	1.0097E-01
1350	5	.20867E-02	.0	29.005	.0	.74220E-02
1400	5	.21192E-02	.0	24.171	.0	.71418E-02
1450	5	.22519E-02	.0	19.337	.0	.78072E-02
1500	5	.46415E-02	.0	14.503	.0	.13266E-01
1550	5	.15451E-01	.0	9.6685	.0	.26028E-01

BEST AVAILABLE COPY

Line	Code	Value	Code	Value	Code	Value	Code	Value
100	44097E+01	.0	4.0343	.0				
101	46025E+01	.0						
102	42210E+01							
103	48032E+01							
104	48032E+01							
105	48032E+01							
106	48032E+01							
107	48032E+01							
108	48032E+01							
109	48032E+01							
110	48032E+01							
111	48032E+01							
112	48032E+01							
113	48032E+01							
114	48032E+01							
115	48032E+01							
116	48032E+01							
117	48032E+01							
118	48032E+01							
119	48032E+01							
120	48032E+01							
121	48032E+01							
122	48032E+01							
123	48032E+01							
124	48032E+01							
125	48032E+01							
126	48032E+01							
127	48032E+01							
128	48032E+01							
129	48032E+01							
130	48032E+01							
131	48032E+01							
132	48032E+01							
133	48032E+01							
134	48032E+01							
135	48032E+01							
136	48032E+01							
137	48032E+01							
138	48032E+01							
139	48032E+01							
140	48032E+01							
141	48032E+01							
142	48032E+01							
143	48032E+01							
144	48032E+01							
145	48032E+01							
146	48032E+01							
147	48032E+01							
148	48032E+01							
149	48032E+01							
150	48032E+01							
151	48032E+01							
152	48032E+01							
153	48032E+01							
154	48032E+01							
155	48032E+01							
156	48032E+01							
157	48032E+01							
158	48032E+01							
159	48032E+01							
160	48032E+01							
161	48032E+01							
162	48032E+01							
163	48032E+01							
164	48032E+01							
165	48032E+01							
166	48032E+01							
167	48032E+01							
168	48032E+01							
169	48032E+01							
170	48032E+01							
171	48032E+01							
172	48032E+01							
173	48032E+01							
174	48032E+01							
175	48032E+01							
176	48032E+01							
177	48032E+01							
178	48032E+01							
179	48032E+01							
180	48032E+01							
181	48032E+01							
182	48032E+01							
183	48032E+01							
184	48032E+01							
185	48032E+01							
186	48032E+01							
187	48032E+01							
188	48032E+01							
189	48032E+01							
190	48032E+01							
191	48032E+01							
192	48032E+01							
193	48032E+01							
194	48032E+01							
195	48032E+01							
196	48032E+01							
197	48032E+01							
198	48032E+01							
199	48032E+01							
200	48032E+01							

1375	1	.64193E+07					
1425	1	.64866E+07					
1475	1	.68748E+07					
1525	1	.94002E+07					
1575	1	.16203E+08					
1625	1	.26980E+08					
2	1	44.366					
15	1	1.0000					
17	3	1.0000	31.000				
24	2	1.0000					
31	1	19.000					
46	1	1.0000					
59	5	50.000	7.0000	5.0000	6.0000		
64	1	6.0000					
135	1	.60357E+03					
147	1	.78540					
153	1	.39001					
172	4	1.0000	.2000E+07	.2000E+07			
176	3	.0	.0	.0			
179	4	39.006	.19030E+07	.49300E+06			
183	4	3.1414	.0	.26018E+07			
187	1	7.4328					
191	2	1.0045					
12	1	366.04					
1202	1	1.0000					
1001	1	1.0000					
1154	1	.0					
1252	1	1.0000					
1302	1	1.0000					
1352	1	1.0000					
1402	1	1.0000					
1452	1	1.0000					
1502	1	1.0000					
1552	1	1.0000					
1602	1	1.0000					
958	1	.38669E+01					
1088	1	.46190E+02					
1158	1	.64174E+02					
1010	1	68.342					
1040	1	68.342					
1110	1	68.342					
162	3	.77161E+01	.77161E+01				
1162	3	.25285E+01	.25285E+01				
1016	1	-.13090E+01					
1166	1	-.13090E+01					
1646	1	.0					
1026	4	3.1325	.13958E+08	.18911E+09	.42694E+09		
1176	4	.0	.0	.0	.0		
179	1	38.418					
184	1	.88644E+01					
159	1	10.560					
195	1	3.0680					
253	1	1.0000					
303	1	1.0000					
353	1	1.0000					
403	1	1.0000					
453	1	1.0000					
503	1	1.0000					
553	1	1.0000					
603	1	1.0000					

653	1	1.0000		
703	1	1.0000		
753	1	1.0000		
803	1	1.0000		
853	1	1.0000		
268	2	1.0000	9.6685	
318	2	1.0000	9.6685	
368	2	1.0000	9.6685	
418	2	1.0000	9.6685	
468	2	1.0000	9.6685	
518	2	1.0000	9.6685	
568	2	1.0000	9.6685	
618	2	1.0000	9.6685	
668	2	1.0000	9.6685	
718	2	1.0000	9.6685	
768	2	1.0000	9.6685	
818	2	1.0000	9.6685	
868	2	1.0000	8.3407	
272	2	3.2659	10.640	
322	2	3.2659	10.640	
372	2	3.2659	10.640	
422	2	3.2659	10.640	
472	2	3.2659	10.640	
522	2	3.2659	10.640	
572	2	3.2659	10.640	
622	2	3.2659	10.640	
672	2	3.2659	10.640	
722	2	3.2659	10.640	
772	2	3.2659	10.640	
822	2	3.2659	10.640	
872	2	3.2659	10.640	
3	2	13403.	.11463E=06	
6	3	1.5708	.0	.80000
26	1	1.0000		
34	1	16.000		
274	1	513.97		
324	1	513.97		
374	1	513.97		
424	1	513.97		
474	1	513.97		
524	1	513.97		
574	1	513.97		
624	1	513.97		
674	1	513.97		
724	1	513.97		
774	1	513.97		
824	1	513.97		
874	1	513.97		
167	2	.16410E=02	.50090E=02	
211	1	-1.8539		
261	1	-1.7132		
311	1	-1.5720		
361	1	-1.4293		
411	1	-1.2844		
461	1	-1.1368		
511	1	-.98641		
561	1	-.83380		
611	1	-.67972		
661	1	-.52502		
711	1	-.37095		

216	1	-.30610E-01
266	1	-.30610E-01
316	1	-.31258E-01
366	1	-.32839E-01
416	1	-.35118E-01
466	1	-.37649E-01
516	1	-.40028E-01
566	1	-.41928E-01
616	1	-.43264E-01
666	1	-.43907E-01
716	1	-.43853E-01
766	1	-.42989E-01
816	1	-.41268E-01
866	1	-.38709E-01
916	1	-.35435E-01
966	1	-.34803E-01
1016	1	-.34803E-01
1166	1	-.34803E-01
1216	1	-.30441E-01
1266	1	-.30441E-01
1316	1	-.24709E-01
1366	1	-.24709E-01
1416	1	-.16297E-01
1466	1	-.16297E-01
1516	1	-.67200E-02
1566	1	-.67200E-02
1616	1	-.18700E-02
35	1	.0
65	1	2.0000
48	1	1.0000
49	1	-14.600
54	1	48.600

HELICOPTER AERDELASTIC STABILITY ANALYSIS INPUT PARAMETERS

NB = 1 NBP = 4 NSSEC = 31 NFSEC = 0 NBC = 0 NMC = 0 NFFB = 0 NAB = 16
 NPLN = 0 NFEA = 0 NCT = 0 NCOM = 19 NES = 3 MAXN = 1 NSY = 100 FUA = 0
 MEGN = 1 IPCT = 50 NIT = 12 MER = 7 NORM = 0 IREM = 9 VEX = 0 NPS = 2
 IC = 0 IF = 0 NLEL = 0 MCTV = 1

FLAP HINGE NO CONT TORQUE EQ NO PITCH BEARING NO ARTICULATED SYS NO CONSTRAINT APP YES
 FUSELAGE NO BLADE PHASING YES AERODYNAMICS YES GIMBALLED SYM. NO VARIABLE OUTPUT YES
 DISK PLANE CALC YES FUSELAGE AERU. NO SHASHPLATE IN YES COLL SPRING IN YES VI DTSY USED NO
 SEARCH OPTION NO SHAPE PUNCHED NO LEAD-LAG HINGE NO

ROTOR SPEED = 44.505905 VEL OF SOUND = 13403.0000 AIR DENSITY = 0.11463E+06 CRAFT VEL. = 0.0
 ANGLE CHI = 1.570796 ANGLE ZETA = 0.0 THETA GRAV. = 0.0 ROT GRAVITY = 0.0
 COEF. CLK) = 0.800000 DELTA3 ZETA = 0.0 ALPHA1 = 0.0 COLL. ANGLE = 0.0
 GR RATE STEP = 0.0 FREQ. STEP = 0.0 GRAV. ACC. = 366.087891 SHARP FLEX. = 0.0
 STR. DAMPING = 0.0 P/A CYCLIC = 0.001641 LONG CYCLIC = 0.005009 DAMPING OUT = 0.0
 SPUR LENGTH = 2.582260 SPUR E1Y = 0.20000E+07 SPUR E1Z = 0.20000E+07 COE1 = 0.15700E+03
 COE2 = -0.26308E+03 COE3 = 0.0

PHIM(1) = 0.0 AMKC(1) = 0.38462E+05 CTAU(1) = 0.0 PHTC(1) = -0.7833080 THEC(1) = 0.3900080
 ALI2(1) = 0.5600004 ALSTM(1) = 3.0600004

EL = 38.41740 EAC = 0.25850E+07 E1Y1 = 0.15930E+07 E1Z1 = 0.69500E+06 ALP = 3.161593
 BEY = 0.088648 GAM = 0.0 G1Y1 = 0.26810E+07

PLEN = 7.43270 EACP = 0.0 E1Y2 = 0.0 E1Z2 = 0.0 ALPP = 1.064508
 BEY2 = -0.010144 GAMP = 0.0 G1Y2 = 0.0

SHASHPLATE , TORSIONAL STIFF. = 0.10000E+10, BENDING STIFF. = 0.10000E+10, MASS = 0.01500, RADIUS = 6.100
 ECHI(1) = 0.0 AK(1) = 0.23594E+05 AC(1) = 0.0 BJ(1) = 0.0
 ECHI(2) = 2.09439468 AK(2) = 0.23594E+05 AC(2) = 0.0 BJ(2) = 0.0
 ECHI(3) = 4.18879032 AK(3) = 0.23594E+05 AC(3) = 0.0 BJ(3) = 0.0
 CAPK = 7363.98 CAPC = 0.0

STARTING EIGENVALUE 1 = -0.1466000E+02 0.4860001E+02

AERO TABLES

FOR ISER EQUAL 1 ONLY CCS ARRAYS ARE INPUTTED

11	39	0.0	0.3000	0.4000	0.5000	0.6000	0.7000	0.7500	0.8000	0.8500
		0.9000	1.0000							
-180.0		0.0400	0.0400	0.0400	0.0400	0.0400	0.0400	0.0400	0.0400	0.0400
		0.0400	0.0400							
-174.0		0.6500	0.6500	0.6500	0.6500	0.6500	0.6500	0.6500	0.6500	0.6500
		0.6500	0.6500							
-170.0		0.6500	0.6500	0.6500	0.6500	0.6500	0.6500	0.6500	0.6500	0.6500
		0.6500	0.6500							
-166.0		0.6200	0.6200	0.6200	0.6200	0.6200	0.6200	0.6200	0.6200	0.6200
		0.6200	0.6200							
-133.0		0.8650	0.8650	0.8650	0.8650	0.8650	0.8650	0.8650	0.8650	0.8650
		0.8650	0.8650							
-113.0		0.6350	0.6350	0.6350	0.6350	0.6350	0.6350	0.6350	0.6350	0.6350
		0.6350	0.6350							
-58.0		0.8900	0.8900	0.8900	0.8900	0.8900	0.8900	0.8900	0.8900	0.8900
		0.8900	0.8900							
-38.0		1.1400	1.1400	1.1400	1.1400	1.1400	1.1400	1.1400	1.1400	1.1400
		1.1400	1.1400							
-20.0		0.9800	1.0300	1.0700	1.1300	1.1400	0.8680	0.8600	0.8450	0.7400
		1.0100	1.3400							
-15.0		1.1650	1.2250	1.2850	1.2300	1.1300	0.8800	0.8920	0.8270	0.6850
		1.0100	1.3400							
-13.5		1.2200	1.2800	1.3300	1.2600	1.1200	0.8800	0.8850	0.8140	0.6770
		1.0100	1.3400							
-12.0		1.0950	1.1500	1.2000	1.1400	1.0920	0.8600	0.8770	0.7910	0.6550
		1.0100	1.2500							
-9.5		0.8900	0.9300	0.9700	0.9300	1.0500	0.8300	0.8500	0.7700	0.6200
		0.9470	1.1000							
-5.5		0.4700	0.4930	0.5100	0.5500	0.5850	0.6400	0.6880	0.6500	0.5800
		0.4100	0.6100							
-4.0		0.3120	0.3300	0.3350	0.3600	0.3900	0.4100	0.5000	0.4750	0.2800
		0.2200	0.4300							
-2.0		0.1000	0.1070	0.1050	0.1150	0.1200	0.1200	0.1270	0.1600	0.1650
		0.0850	0.1850							
-1.0		0.0	0.0	0.0	0.0	0.0100	0.0250	0.0150	0.0500	0.0050
		0.0150	0.0600							
0.0		0.1100	0.1100	0.1200	0.1300	0.1400	0.1700	0.1900	0.2400	0.0600
		0.1100	0.0600							
2.0		0.3200	0.3320	0.3500	0.3750	0.4100	0.4750	0.4850	0.3700	0.2080
		0.3900	0.3000							
3.0		0.4250	0.4430	0.4670	0.5000	0.5470	0.6200	0.5000	0.4150	0.2600
		0.5200	0.4260							
4.0		0.5300	0.5500	0.5800	0.6200	0.6800	0.6750	0.5700	0.4750	0.2800
		0.6300	0.5450							
6.0		0.7420	0.7720	0.8150	0.8680	0.9100	0.7670	0.6700	0.5100	0.3200
		0.9200	0.7870							
8.0		0.9500	0.9950	1.0350	1.1150	0.9950	0.8560	0.7320	0.5450	0.3550
		1.1900	1.0300							
10.0		1.1650	1.2200	1.2600	1.2700	1.0750	0.9420	0.7910	0.5800	0.3930
		1.1900	1.0950							
11.0		1.2680	1.3280	1.3800	1.2780	1.0920	0.9850	0.8250	0.6000	0.4130
		1.1900	1.0950							
12.0		1.3750	1.4400	1.4200	1.2500	1.1130	1.0280	0.8520	0.6150	0.4320
		1.1900	1.0950							
13.0		1.4800	1.5500	1.3950	1.2250	1.1300	1.0690	0.8850	0.6330	0.4500

	1.1900	1.0950								
13.5	1.5300	1.6000	1.2700	1.2070	1.1410	1.0900	0.9000	0.6410	0.4610	
	1.1900	1.0950								
14.0	1.5850	1.0950	1.1400	1.1900	1.1500	1.1120	0.9130	0.6500	0.4700	
	1.1900	1.0950								
15.0	0.9150	0.9620	1.0000	1.1920	1.1700	1.1550	0.9460	0.6680	0.4900	
	1.1900	1.0950								
20.0	1.0000	1.0500	1.0900	1.2000	1.2600	1.3670	1.1000	0.7600	0.5900	
	1.1900	1.0950								
38.0	1.1350	1.1350	1.1350	1.1350	1.1350	1.1350	1.1350	1.1350	1.1350	
	1.1350	1.1350								
58.0	0.8900	0.8900	0.8900	0.8900	0.8900	0.8900	0.8900	0.8900	0.8900	
	0.8900	0.8900								
113.0	-0.6350	-0.6350	-0.6350	-0.6350	-0.6350	-0.6350	-0.6350	-0.6350	-0.6350	
	-0.6350	-0.6350								
133.0	-0.8650	-0.8650	-0.8650	-0.8650	-0.8650	-0.8650	-0.8650	-0.8650	-0.8650	
	-0.8650	-0.8650								
158.0	-1.0000	-1.0000	-1.0000	-1.0000	-1.0000	-1.0000	-1.0000	-1.0000	-1.0000	
	-1.0000	-1.0000								
166.0	-0.7200	-0.7200	-0.7200	-0.7200	-0.7200	-0.7200	-0.7200	-0.7200	-0.7200	
	-0.7200	-0.7200								
170.0	-0.8200	-0.8200	-0.8200	-0.8200	-0.8200	-0.8200	-0.8200	-0.8200	-0.8200	
	-0.8200	-0.8200								
180.0	0.0400	0.0400	0.0400	0.0400	0.0400	0.0400	0.0400	0.0400	0.0400	
	0.0400	0.0400								
11	45									
	0.0	0.4000	0.5000	0.6000	0.6500	0.7000	0.7500	0.8000	0.8500	
	0.9000	1.0000								
-180.0	0.0150	0.0150	0.0150	0.0150	0.0150	0.0150	0.0150	0.0150	0.0150	
	0.0150	0.0150								
-175.0	0.0400	0.0400	0.0400	0.0400	0.0400	0.0400	0.0400	0.0400	0.0400	
	0.0400	0.0400								
-170.0	0.1100	0.1100	0.1100	0.1100	0.1100	0.1100	0.1100	0.1100	0.1100	
	0.1100	0.1100								
-164.0	0.2200	0.2200	0.2200	0.2200	0.2200	0.2200	0.2200	0.2200	0.2200	
	0.2200	0.2200								
-155.0	0.5100	0.5100	0.5100	0.5100	0.5100	0.5100	0.5100	0.5100	0.5100	
	0.5100	0.5100								
-130.0	1.0800	1.0800	1.0800	1.0800	1.0800	1.0800	1.0800	1.0800	1.0800	
	1.0800	1.0800								
-100.0	1.5100	1.5100	1.5100	1.5100	1.5100	1.5100	1.5100	1.5100	1.5100	
	1.5100	1.5100								
-90.0	1.5600	1.5600	1.5600	1.5600	1.5600	1.5600	1.5600	1.5600	1.5600	
	1.5600	1.5600								
-80.0	1.5100	1.5100	1.5100	1.5100	1.5100	1.5100	1.5100	1.5100	1.5100	
	1.5100	1.5100								
-60.0	1.2400	1.2400	1.2400	1.2400	1.2400	1.2400	1.2400	1.2400	1.2400	
	1.2400	1.2400								
-40.0	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	
	0.9000	0.9000								
-25.0	0.5100	0.5100	0.5100	0.5100	0.5100	0.5100	0.5100	0.5100	0.5100	
	0.5100	0.5100								
-8.0	0.0205	0.0205	0.0670	0.1360	0.1375	0.1405	0.1440	0.1700	0.1818	
	0.1930	0.2175								
-6.0	0.0154	0.0154	0.0300	0.0750	0.0765	0.0790	0.0870	0.1150	0.1270	
	0.1380	0.1615								
-4.0	0.0112	0.0112	0.0115	0.0141	0.0163	0.0182	0.0207	0.0600	0.0715	
	0.0830	0.1062								
-3.0	0.0107	0.0107	0.0109	0.0123	0.0133	0.0144	0.0117	0.0322	0.0448	
	0.0625	0.0890								

-2.0	0.0104	0.0104	0.0104	0.0105	0.0104	0.0107	0.0110	0.0137	0.0284
	0.0425	0.0715							
-1.0	0.0105	0.0105	0.0103	0.0104	0.0104	0.0105	0.0104	0.0177	0.0373
	0.0512	0.0812							
0.0	0.0106	0.0106	0.0102	0.0102	0.0103	0.0103	0.0131	0.0287	0.0445
	0.0600	0.0910							
1.0	0.0105	0.0105	0.0103	0.0104	0.0107	0.0119	0.0217	0.0474	0.0631
	0.0790	0.1105							
2.0	0.0104	0.0104	0.0104	0.0105	0.0112	0.0186	0.0384	0.0659	0.0820
	0.0975	0.1293							
3.0	0.0106	0.0106	0.0107	0.0118	0.0150	0.0298	0.0569	0.0848	0.1010
	0.1165	0.1485							
4.0	0.0108	0.0108	0.0110	0.0131	0.0236	0.0484	0.0755	0.1035	0.1195
	0.1350	0.1670							
5.0	0.0114	0.0114	0.0118	0.0191	0.0345	0.0670	0.0944	0.1220	0.1383
	0.1530	0.1860							
6.0	0.0120	0.0120	0.0125	0.0293	0.0530	0.0860	0.1130	0.1410	0.1570
	0.1720	0.2050							
7.0	0.0127	0.0127	0.0133	0.0426	0.0720	0.1047	0.1319	0.1600	0.1760
	0.1915	0.2230							
8.0	0.0133	0.0133	0.0153	0.0610	0.0905	0.1235	0.1505	0.1788	0.1950
	0.2100	0.2425							
9.0	0.0137	0.0137	0.0212	0.0798	0.1092	0.1422	0.1693	0.1980	0.2135
	0.2290	0.2610							
10.0	0.0153	0.0153	0.0300	0.0983	0.1275	0.1607	0.1880	0.2170	0.2320
	0.2480	0.2800							
11.0	0.0175	0.0175	0.0483	0.1180	0.1465	0.1790	0.2070	0.2360	0.2505
	0.2660	0.2990							
12.0	0.0205	0.0205	0.0670	0.1357	0.1650	0.1985	0.2250	0.2560	0.2700
	0.2850	0.3180							
13.0	0.0282	0.0282	0.0858	0.1545	0.1835	0.2170	0.2440	0.2750	0.2890
	0.3040	0.3365							
15.0	0.0465	0.0465	0.1233	0.1920	0.2210	0.2540	0.2810	0.3135	0.3265
	0.3400	0.3740							
30.0	0.6600	0.6600	0.6600	0.6600	0.6600	0.6600	0.6600	0.6600	0.6600
	0.6600	0.6600							
50.0	1.0700	1.0700	1.0700	1.0700	1.0700	1.0700	1.0700	1.0700	1.0700
	1.0700	1.0700							
80.0	1.5000	1.5000	1.5000	1.5000	1.5000	1.5000	1.5000	1.5000	1.5000
	1.5000	1.5000							
90.0	1.5600	1.5600	1.5600	1.5600	1.5600	1.5600	1.5600	1.5600	1.5600
	1.5600	1.5600							
100.0	1.5100	1.5100	1.5100	1.5100	1.5100	1.5100	1.5100	1.5100	1.5100
	1.5100	1.5100							
120.0	1.2300	1.2300	1.2300	1.2300	1.2300	1.2300	1.2300	1.2300	1.2300
	1.2300	1.2300							
140.0	0.8900	0.8900	0.8900	0.8900	0.8900	0.8900	0.8900	0.8900	0.8900
	0.8900	0.8900							
155.0	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000
	0.5000	0.5000							
164.0	0.2200	0.2200	0.2200	0.2200	0.2200	0.2200	0.2200	0.2200	0.2200
	0.2200	0.2200							
170.0	0.1100	0.1100	0.1100	0.1100	0.1100	0.1100	0.1100	0.1100	0.1100
	0.1100	0.1100							
175.0	0.0400	0.0400	0.0400	0.0400	0.0400	0.0400	0.0400	0.0400	0.0400
	0.0400	0.0400							
180.0	0.0150	0.0150	0.0150	0.0150	0.0150	0.0150	0.0150	0.0150	0.0150
	0.0150	0.0150							
11	44								
	0.0	0.4000	0.5000	0.6000	0.6500	0.7000	0.7500	0.8000	0.8500

0.9000 1.0000
 -180.0 -0.0400 -0.0400 -0.0400 -0.0400 -0.0400 -0.0400 -0.0400 -0.0400 -0.0400
 -0.0400 -0.0400
 -172.0 0.3700 0.3700 0.3700 0.3700 0.3700 0.3700 0.3700 0.3700 0.3700
 0.3700 0.3700
 -168.0 0.3500 0.3500 0.3500 0.3500 0.3500 0.3500 0.3500 0.3500 0.3500
 0.3500 0.3500
 -164.0 0.3900 0.3900 0.3900 0.3900 0.3900 0.3900 0.3900 0.3900 0.3900
 0.3900 0.3900
 -156.0 0.4200 0.4200 0.4200 0.4200 0.4200 0.4200 0.4200 0.4200 0.4200
 0.4200 0.4200
 -150.0 0.4450 0.4450 0.4450 0.4450 0.4450 0.4450 0.4450 0.4450 0.4450
 0.4450 0.4450
 -130.0 0.5750 0.5750 0.5750 0.5750 0.5750 0.5750 0.5750 0.5750 0.5750
 0.5750 0.5750
 -115.0 0.6000 0.6000 0.6000 0.6000 0.6000 0.6000 0.6000 0.6000 0.6000
 0.6000 0.6000
 -90.0 0.5500 0.5500 0.5500 0.5500 0.5500 0.5500 0.5500 0.5500 0.5500
 0.5500 0.5500
 -60.0 0.4000 0.4000 0.4000 0.4000 0.4000 0.4000 0.4000 0.4000 0.4000
 0.4000 0.4000
 -40.0 0.2600 0.2600 0.2600 0.2600 0.2600 0.2600 0.2600 0.2600 0.2600
 0.2600 0.2600
 -30.0 0.1800 0.1800 0.1800 0.1800 0.1800 0.1800 0.1800 0.1800 0.1800
 0.1800 0.1800
 -16.0 0.1050 0.1050 0.1050 0.1050 0.1050 0.1050 0.1050 0.1050 0.1050
 0.1050 0.1050
 -4.0 -0.0075 -0.0075 -0.0065 -0.0105 -0.0150 -0.0180 -0.0175 0.0238 0.0090
 0.0200 0.0050
 -3.0 -0.0070 -0.0070 -0.0067 -0.0091 -0.0112 -0.0150 -0.0189 0.0090 0.0055
 -0.0060 -0.0020
 -2.0 -0.0070 -0.0065 -0.0070 -0.0075 -0.0075 -0.0120 -0.0100 -0.0130 0.0040
 -0.0310 -0.0095
 -1.0 -0.0075 -0.0075 -0.0075 -0.0084 -0.0084 -0.0105 -0.0090 -0.0120 -0.0275
 -0.0300 -0.0350
 0.0 -0.0080 -0.0080 -0.0080 -0.0090 -0.0090 -0.0090 -0.0080 -0.0330 -0.0215
 -0.0435 -0.0570
 1.0 -0.0085 -0.0085 -0.0083 -0.0090 -0.0088 -0.0075 -0.0125 -0.0505 -0.0235
 -0.0500 -0.0615
 2.0 -0.0090 -0.0090 -0.0087 -0.0090 -0.0085 -0.0070 -0.0250 -0.0515 -0.0370
 -0.0505 -0.0680
 3.0 -0.0093 -0.0093 -0.0090 -0.0076 -0.0060 -0.0110 -0.0415 -0.0525 -0.0375
 -0.0510 -0.0720
 4.0 -0.0100 -0.0095 -0.0095 -0.0060 -0.0035 -0.0225 -0.0525 -0.0570 -0.0600
 -0.0655 -0.0745
 5.0 -0.0100 -0.0090 -0.0090 -0.0024 -0.0050 -0.0400 -0.0605 -0.0760 -0.0840
 -0.0825 -0.0805
 6.0 -0.0115 -0.0075 -0.0085 0.0 -0.0150 -0.0505 -0.0670 -0.0830 -0.0975
 -0.0935 -0.0850
 7.0 -0.0130 -0.0045 -0.0078 0.0005 -0.0310 -0.0580 -0.0755 -0.0785 -0.0900
 -0.0990 -0.0910
 8.0 -0.0150 -0.0005 -0.0070 -0.0110 -0.0425 -0.0770 -0.0820 -0.0890 -0.0970
 -0.1040 -0.0960
 9.0 0.0060 0.0060 -0.0060 -0.0223 -0.0525 -0.0820 -0.0890 -0.0940 -0.1020
 -0.1088 -0.1009
 10.0 0.0125 0.0125 -0.0030 -0.0315 -0.0624 -0.0869 -0.0960 -0.0989 -0.1068
 -0.1137 -0.1058
 11.0 0.0135 0.0135 0.0010 -0.0415 -0.0723 -0.0918 -0.1010 -0.1028 -0.1118
 -0.1186 -0.1106
 12.0 0.0035 0.0035 0.0035 -0.0514 -0.0822 -0.0966 -0.1058 -0.1087 -0.1167

-0.1235-0.1155
 13.0-0.0185-0.0185-0.0050-0.0613-0.0922-0.1017-0.1107-0.1137-0.1215
 -0.1285-0.1205
 14.0-0.0435-0.0435-0.0565-0.0612-0.1023-0.1065-0.1157-0.1186-0.1266
 -0.1335-0.1255
 16.0-0.1020-0.1020-0.0885-0.0912-0.1122-0.1165-0.1252-0.1285-0.1365
 -0.1430-0.1354
 25.0-0.1750-0.1750-0.1750-0.1750-0.1750-0.1750-0.1750-0.1750-0.1750
 -0.1750-0.1750
 40.0-0.2900-0.2900-0.2900-0.2900-0.2900-0.2900-0.2900-0.2900-0.2900
 -0.2900-0.2900
 60.0-0.4300-0.4300-0.4300-0.4300-0.4300-0.4300-0.4300-0.4300-0.4300
 -0.4300-0.4300
 90.0-0.5800-0.5800-0.5800-0.5800-0.5800-0.5800-0.5800-0.5800-0.5800
 -0.5800-0.5800
 115.0-0.6300-0.6300-0.6300-0.6300-0.6300-0.6300-0.6300-0.6300-0.6300
 -0.6300-0.6300
 140.0-0.5550-0.5550-0.5550-0.5550-0.5550-0.5550-0.5550-0.5550-0.5550
 -0.5550-0.5550
 160.0-0.4300-0.4300-0.4300-0.4300-0.4300-0.4300-0.4300-0.4300-0.4300
 -0.4300-0.4300
 168.0-0.3800-0.3800-0.3800-0.3800-0.3800-0.3800-0.3800-0.3800-0.3800
 -0.3800-0.3800
 172.0-0.3900-0.3900-0.3900-0.3900-0.3900-0.3900-0.3900-0.3900-0.3900
 -0.3900-0.3900
 176.0-0.2800-0.2800-0.2800-0.2800-0.2800-0.2800-0.2800-0.2800-0.2800
 -0.2800-0.2800
 180.0-0.0400-0.0400-0.0400-0.0400-0.0400-0.0400-0.0400-0.0400-0.0400
 -0.0400-0.0400

SECTION	N	STEADY FORCES AND MOMENTS				M2	MY
		VY	VZ	T			
1	0.27087E+04	-0.33671E+01	-0.85743E+02	0.48235E+02	0.16685E+04	0.0	

AERO COEFFICIENTS

ALPMZ	AMACZ	CSUBL	CSUBD	CSUBM	CLALP	CDALP	CHALP
0.2603E+01	0.9112E+00	0.5004E+00	0.1178E-01	-0.7895E-02	0.7829E+01	0.1830E+00	0.9978E-01
0.2545E+01	0.9112E+00	0.4928E+00	0.1178E-01	-0.7906E-02	0.7874E+01	0.1804E+00	0.9988E-01
0.2532E+01	0.9112E+00	0.4797E+00	0.1150E-01	-0.8103E-02	0.7857E+01	0.1366E+00	0.9840E-01
0.2337E+01	0.9112E+00	0.4648E+00	0.1187E-01	-0.8330E-02	0.7836E+01	0.1276E+00	0.9739E-01
0.2222E+01	0.9112E+00	0.4479E+00	0.1187E-01	-0.8522E-02	0.7727E+01	0.1208E+00	0.9635E-01
0.2119E+01	0.9112E+00	0.4337E+00	0.1086E-01	-0.8692E-02	0.7608E+01	0.1168E+00	0.9548E-01
0.2046E+01	0.9112E+00	0.4237E+00	0.1074E-01	-0.8811E-02	0.7495E+01	0.1095E+00	0.9479E-01
0.2015E+01	0.9112E+00	0.4193E+00	0.1048E-01	-0.8940E-02	0.7390E+01	0.1075E+00	0.9431E-01
0.2077E+01	0.9112E+00	0.4213E+00	0.1071E-01	-0.8848E-02	0.7379E+01	0.1085E+00	0.9463E-01
0.2179E+01	0.9112E+00	0.4292E+00	0.1092E-01	-0.8752E-02	0.7333E+01	0.1125E+00	0.9513E-01
0.2303E+01	0.9112E+00	0.4420E+00	0.1130E-01	-0.8595E-02	0.7278E+01	0.1178E+00	0.9597E-01
0.2410E+01	0.9112E+00	0.4577E+00	0.1180E-01	-0.8358E-02	0.7225E+01	0.1248E+00	0.9698E-01
0.2512E+01	0.9112E+00	0.4738E+00	0.1132E-01	-0.8213E-02	0.7300E+01	0.1320E+00	0.9808E-01
0.2655E+01	0.9112E+00	0.4980E+00	0.1191E-01	-0.8033E-02	0.7500E+01	0.1364E+00	0.9993E-01
0.2817E+01	0.9112E+00	0.5284E+00	0.1191E-01	-0.7871E-02	0.7443E+01	0.1429E+00	0.9958E-01
2	0.5734E+02	-0.51153E+01	0.27127E+01	-0.55245E+02	0.34941E+04	0.83099E+03	0.9906E-01

AERO COEFFICIENTS

ALPMZ	AMACZ	CSUBL	CSUBD	CSUBM	CLALP	CDALP	CHALP
0.2782E+01	0.5792E+00	0.5004E+00	0.1133E-01	-0.8119E-02	0.7318E+01	0.8257E+01	0.6519E-01
0.2725E+01	0.5792E+00	0.5002E+00	0.1132E-01	-0.8179E-02	0.7322E+01	0.8257E+01	0.6602E-01
0.2652E+01	0.5792E+00	0.4877E+00	0.1172E-01	-0.8276E-02	0.7354E+01	0.8257E+01	0.6418E-01
0.2518E+01	0.5792E+00	0.4728E+00	0.1192E-01	-0.8395E-02	0.7382E+01	0.8257E+01	0.6342E-01
0.2401E+01	0.5792E+00	0.4567E+00	0.1022E-01	-0.8518E-02	0.7410E+01	0.8257E+01	0.6264E-01
0.2299E+01	0.5792E+00	0.4429E+00	0.1031E-01	-0.8625E-02	0.7434E+01	0.8256E+01	0.6195E-01
0.2266E+01	0.5792E+00	0.4331E+00	0.1073E-01	-0.8701E-02	0.7452E+01	0.8256E+01	0.6144E-01
0.2208E+01	0.5792E+00	0.4208E+00	0.1049E-01	-0.8734E-02	0.7460E+01	0.8256E+01	0.6125E-01
0.2266E+01	0.5792E+00	0.4307E+00	0.1071E-01	-0.8720E-02	0.7454E+01	0.8256E+01	0.6133E-01
0.2359E+01	0.5792E+00	0.4405E+00	0.1077E-01	-0.8659E-02	0.7482E+01	0.8256E+01	0.6173E-01
0.2489E+01	0.5792E+00	0.4643E+00	0.1100E-01	-0.8542E-02	0.7420E+01	0.8257E+01	0.6233E-01
0.2692E+01	0.5792E+00	0.4820E+00	0.1132E-01	-0.8321E-02	0.7393E+01	0.8257E+01	0.6311E-01
0.2853E+01	0.5792E+00	0.4958E+00	0.1132E-01	-0.8213E-02	0.7560E+01	0.8257E+01	0.6390E-01
0.2797E+01	0.5792E+00	0.5058E+00	0.1131E-01	-0.8137E-02	0.7522E+01	0.8257E+01	0.6507E-01
3	0.64957E+04	-0.58452E+01	0.78199E+02	-0.14793E+03	0.51395E+04	0.87249E+03	0.6528E-01

AERO COEFFICIENTS

ALPHZ	AMAZ	CSUBL	CSUBD	CSUMH	CLALP	CDALP	CHALP
0.2935E+01	0.5472E+00	0.5134E+00	0.1117E-01	-0.8772E+02	0.7244E+01	0.4421E-01	0.2815E-01
0.2677E+01	0.5472E+00	0.5041E+00	0.1112E-01	-0.8701E+02	0.7239E+01	0.4421E-01	0.2815E-01
0.2419E+01	0.5472E+00	0.4948E+00	0.1105E-01	-0.8648E+02	0.7233E+01	0.4421E-01	0.2815E-01
0.2161E+01	0.5472E+00	0.4791E+00	0.1096E-01	-0.8595E+02	0.7312E+01	0.4421E-01	0.2815E-01
0.1903E+01	0.5472E+00	0.4639E+00	0.1087E-01	-0.8542E+02	0.7343E+01	0.4421E-01	0.2815E-01
0.1645E+01	0.5472E+00	0.4490E+00	0.1074E-01	-0.8415E+02	0.7368E+01	0.4421E-01	0.2815E-01
0.1387E+01	0.5472E+00	0.4368E+00	0.1071E-01	-0.8668E+02	0.7394E+01	0.4421E-01	0.2815E-01
0.1129E+01	0.5472E+00	0.4306E+00	0.1073E-01	-0.8601E+02	0.7393E+01	0.4421E-01	0.2815E-01
0.0871E+01	0.5472E+00	0.4422E+00	0.1077E-01	-0.8631E+02	0.7374E+01	0.4421E-01	0.2815E-01
0.0613E+01	0.5472E+00	0.4503E+00	0.1084E-01	-0.8685E+02	0.7345E+01	0.4421E-01	0.2815E-01
0.0355E+01	0.5472E+00	0.4711E+00	0.1102E-01	-0.8728E+02	0.7324E+01	0.4421E-01	0.2815E-01
0.0097E+01	0.5472E+00	0.4884E+00	0.1102E-01	-0.8689E+02	0.7294E+01	0.4421E-01	0.2815E-01
0.0000E+01	0.5472E+00	0.5018E+00	0.1110E-01	-0.8418E+02	0.7267E+01	0.4421E-01	0.2815E-01
0.2917E+01	0.5472E+00	0.5113E+00	0.1115E-01	-0.8381E+02	0.7244E+01	0.4421E-01	0.2815E-01
0.2949E+01	0.5472E+00	0.5155E+00	0.1118E-01	-0.8365E+02	0.7240E+01	0.4421E-01	0.2815E-01
5	0.11103E+03	0.42743E+01	0.12877E+03	-0.22300E+03	0.66456E+04	0.27081E+03	

AERO COEFFICIENTS

ALPHZ	AMAZ	CSUBL	CSUBD	CSUMH	CLALP	CDALP	CHALP
0.1035E+01	0.5151E+00	0.5151E+00	0.1089E-01	-0.8748E+02	0.6960E+01	0.2987E-01	0.4628E-02
0.1000E+01	0.5151E+00	0.5071E+00	0.1087E-01	-0.8768E+02	0.6968E+01	0.2987E-01	0.4628E-02
0.0965E+01	0.5151E+00	0.4933E+00	0.1082E-01	-0.8784E+02	0.7014E+01	0.2987E-01	0.4628E-02
0.0930E+01	0.5151E+00	0.4809E+00	0.1077E-01	-0.8779E+02	0.7046E+01	0.2986E-01	0.4628E-02
0.0895E+01	0.5151E+00	0.4661E+00	0.1072E-01	-0.8774E+02	0.7078E+01	0.2986E-01	0.4628E-02
0.0860E+01	0.5151E+00	0.4531E+00	0.1067E-01	-0.8770E+02	0.7107E+01	0.2986E-01	0.4628E-02
0.0825E+01	0.5151E+00	0.4439E+00	0.1064E-01	-0.8770E+02	0.7127E+01	0.2986E-01	0.4628E-02
0.0790E+01	0.5151E+00	0.4398E+00	0.1063E-01	-0.8765E+02	0.7134E+01	0.2986E-01	0.4628E-02
0.0755E+01	0.5151E+00	0.4416E+00	0.1063E-01	-0.8766E+02	0.7132E+01	0.2986E-01	0.4628E-02
0.0720E+01	0.5151E+00	0.4490E+00	0.1066E-01	-0.8768E+02	0.7118E+01	0.2986E-01	0.4628E-02
0.0685E+01	0.5151E+00	0.4607E+00	0.1070E-01	-0.8772E+02	0.7090E+01	0.2986E-01	0.4628E-02
0.0650E+01	0.5151E+00	0.4751E+00	0.1075E-01	-0.8777E+02	0.7058E+01	0.2987E-01	0.4628E-02
0.0615E+01	0.5151E+00	0.4899E+00	0.1081E-01	-0.8782E+02	0.7024E+01	0.2987E-01	0.4628E-02
0.0580E+01	0.5151E+00	0.5030E+00	0.1085E-01	-0.8787E+02	0.6997E+01	0.2987E-01	0.4628E-02
0.0545E+01	0.5151E+00	0.5120E+00	0.1088E-01	-0.8795E+02	0.6960E+01	0.2987E-01	0.4628E-02
0.0510E+01	0.5151E+00	0.5150E+00	0.1090E-01	-0.8801E+02	0.6915E+01	0.2987E-01	0.4628E-02
5	0.13522E+03	0.66493E+01	0.15145E+03	0.27443E+03	0.81171E+04	0.67658E+03	

AERO COEFFICIENTS

ALPHZ	AMACZ	CSUBL	CSUBD	CSUBM	CLALP	CDALP	CHALP
0.3145E+01	0.4831E+00	0.5117E+00	0.1072E+01	-0.9116E+02	0.8842E+01	0.1898E+01	-0.1789E+01
0.3087E+01	0.4831E+00	0.5088E+00	0.1071E+01	-0.9090E+02	0.8822E+01	0.1883E+01	-0.2099E+01
0.2995E+01	0.4831E+00	0.4986E+00	0.1068E+01	-0.9049E+02	0.8802E+01	0.1872E+01	-0.2370E+01
0.2891E+01	0.4831E+00	0.4797E+00	0.1062E+01	-0.9015E+02	0.8841E+01	0.1862E+01	-0.2473E+01
0.2764E+01	0.4831E+00	0.4631E+00	0.1058E+01	-0.8980E+02	0.8877E+01	0.1852E+01	-0.2373E+01
0.262E+01	0.4831E+00	0.4524E+00	0.1059E+01	-0.8949E+02	0.8902E+01	0.1842E+01	-0.2292E+01
0.2509E+01	0.4831E+00	0.4437E+00	0.1057E+01	-0.8927E+02	0.8928E+01	0.1832E+01	-0.2232E+01
0.2371E+01	0.4831E+00	0.4344E+00	0.1056E+01	-0.8922E+02	0.8938E+01	0.1822E+01	-0.2192E+01
0.2229E+01	0.4831E+00	0.4246E+00	0.1056E+01	-0.8939E+02	0.8927E+01	0.1822E+01	-0.2192E+01
0.2095E+01	0.4831E+00	0.4144E+00	0.1046E+01	-0.8967E+02	0.8911E+01	0.1822E+01	-0.2337E+01
0.2035E+01	0.4831E+00	0.4042E+00	0.1044E+01	-0.9001E+02	0.8892E+01	0.1822E+01	-0.2337E+01
0.2052E+01	0.4831E+00	0.4055E+00	0.1047E+01	-0.9036E+02	0.8821E+01	0.1822E+01	-0.2337E+01
0.3055E+01	0.4831E+00	0.5010E+00	0.1070E+01	-0.9075E+02	0.8821E+01	0.1711E+01	-0.2274E+01
0.3128E+01	0.4831E+00	0.5096E+00	0.1072E+01	-0.9105E+02	0.8837E+01	0.1642E+01	-0.1881E+01
0.3159E+01	0.4831E+00	0.5134E+00	0.1073E+01	-0.9122E+02	0.8893E+01	0.2023E+01	-0.1708E+01
0	0.1574E+05	-0.89308E+01	0.1498E+03	0.3194E+03	0.49301E+08	-0.17565E+08	

AERO COEFFICIENTS

ALPHZ	AMACZ	CSUBL	CSUBD	CSUBM	CLALP	CDALP	CHALP
0.3190E+01	0.4511E+00	0.5069E+00	0.1070E+01	-0.9214E+02	0.8725E+01	0.1901E+01	-0.1898E+01
0.3132E+01	0.4511E+00	0.4982E+00	0.1068E+01	-0.9193E+02	0.8712E+01	0.1780E+01	-0.1309E+01
0.3054E+01	0.4511E+00	0.4884E+00	0.1063E+01	-0.9124E+02	0.8689E+01	0.1536E+01	-0.1833E+01
0.2928E+01	0.4511E+00	0.4789E+00	0.1060E+01	-0.9089E+02	0.8729E+01	0.1439E+01	-0.2002E+01
0.2806E+01	0.4511E+00	0.4607E+00	0.1056E+01	-0.9058E+02	0.8752E+01	0.1439E+01	-0.1966E+01
0.2706E+01	0.4511E+00	0.4433E+00	0.1056E+01	-0.9037E+02	0.8772E+01	0.1439E+01	-0.1935E+01
0.2633E+01	0.4511E+00	0.4399E+00	0.1055E+01	-0.9027E+02	0.8772E+01	0.1439E+01	-0.1935E+01
0.2602E+01	0.4511E+00	0.4356E+00	0.1055E+01	-0.9031E+02	0.8782E+01	0.1439E+01	-0.1935E+01
0.2616E+01	0.4511E+00	0.4373E+00	0.1055E+01	-0.9049E+02	0.8772E+01	0.1439E+01	-0.1935E+01
0.2674E+01	0.4511E+00	0.4433E+00	0.1057E+01	-0.9076E+02	0.8742E+01	0.1439E+01	-0.1935E+01
0.280E+01	0.4511E+00	0.4566E+00	0.1059E+01	-0.9111E+02	0.8742E+01	0.1439E+01	-0.1935E+01
0.297E+01	0.4511E+00	0.4692E+00	0.1062E+01	-0.9146E+02	0.8711E+01	0.1439E+01	-0.1935E+01
0.3099E+01	0.4511E+00	0.4832E+00	0.1062E+01	-0.9146E+02	0.8689E+01	0.1439E+01	-0.1935E+01
0.3172E+01	0.4511E+00	0.4955E+00	0.1068E+01	-0.9207E+02	0.8708E+01	0.1681E+01	-0.1559E+01
0.3204E+01	0.4511E+00	0.5039E+00	0.1069E+01	-0.9272E+02	0.8721E+01	0.1892E+01	-0.1103E+01
0.3304E+01	0.4511E+00	0.5078E+00	0.1070E+01	-0.9319E+02	0.8722E+01	0.1936E+01	-0.1088E+01
7	0.17817E+05	-0.75300E+01	0.12964E+03	-0.34888E+03	0.10453E+05	-0.27825E+08	

AERO COEFFICIENTS

ALPHZ	0.2181E+01	AMCZ	0.4192E+00	CSUBL	0.4940E+00	CSUDD	0.1066E-01	CSUBM	-0.9289E-02	CLALP	0.4597E+01	CDALP	0.1681E-01	CHALP	-0.6804E-02
0.2123E+01	0.4192E+00	0.4874E+00	0.1065E-01	0.4874E+00	0.4760E+00	0.1063E-01	-0.9274E-02	0.4597E+01	0.1549E-01	-0.6804E-02	0.1549E-01	0.1549E-01	-0.1337E-01	-0.1337E-01	
0.2017E+01	0.4192E+00	0.4635E+00	0.1060E-01	0.4635E+00	0.4406E+00	0.1058E-01	-0.9218E-02	0.4597E+01	0.1254E-01	-0.1337E-01	0.1254E-01	0.1254E-01	-0.1146E-01	-0.1146E-01	
0.2009E+01	0.4191E+00	0.4406E+00	0.1055E-01	0.4406E+00	0.4374E+00	0.1054E-01	-0.9152E-02	0.4627E+01	0.1124E-01	-0.1146E-01	0.1124E-01	0.1124E-01	-0.1012E-01	-0.1012E-01	
0.2007E+01	0.4191E+00	0.4288E+00	0.1054E-01	0.4288E+00	0.4257E+00	0.1053E-01	-0.9135E-02	0.4641E+01	0.1064E-01	-0.1012E-01	0.1064E-01	0.1064E-01	-0.0942E-01	-0.0942E-01	
0.2007E+01	0.4191E+00	0.4257E+00	0.1053E-01	0.4257E+00	0.4236E+00	0.1053E-01	-0.9125E-02	0.4644E+01	0.1004E-01	-0.0942E-01	0.1004E-01	0.1004E-01	-0.0832E-01	-0.0832E-01	
0.2007E+01	0.4191E+00	0.4236E+00	0.1053E-01	0.4236E+00	0.4215E+00	0.1053E-01	-0.9116E-02	0.4646E+01	0.0944E-01	-0.0832E-01	0.0944E-01	0.0944E-01	-0.0736E-01	-0.0736E-01	
0.2007E+01	0.4191E+00	0.4215E+00	0.1052E-01	0.4215E+00	0.4194E+00	0.1052E-01	-0.9204E-02	0.4582E+01	0.0884E-01	-0.0736E-01	0.0884E-01	0.0884E-01	-0.0676E-01	-0.0676E-01	
0.2007E+01	0.4192E+00	0.4194E+00	0.1044E-01	0.4194E+00	0.4173E+00	0.1044E-01	-0.9239E-02	0.4574E+01	0.0824E-01	-0.0676E-01	0.0824E-01	0.0824E-01	-0.0616E-01	-0.0616E-01	
0.2102E+01	0.4192E+00	0.4173E+00	0.1045E-01	0.4173E+00	0.4152E+00	0.1045E-01	-0.9285E-02	0.4592E+01	0.0764E-01	-0.0616E-01	0.0764E-01	0.0764E-01	-0.0556E-01	-0.0556E-01	
0.2102E+01	0.4192E+00	0.4152E+00	0.1046E-01	0.4152E+00	0.4131E+00	0.1046E-01	-0.9333E-02	0.4600E+01	0.0704E-01	-0.0556E-01	0.0704E-01	0.0704E-01	-0.0496E-01	-0.0496E-01	
0.219742E+03	-0.55117E+01	0.99070E+02	-0.36390E+03	0.11780E+05	-0.36390E+03	0.11780E+05	-0.36390E+03	0.11780E+05	-0.36390E+03	0.11780E+05	-0.36390E+03	0.11780E+05	-0.36390E+03	0.11780E+05	-0.36390E+03

AERO COEFFICIENTS

ALPHZ	0.2186E+01	AMCZ	0.3872E+00	CSUBL	0.4739E+00	CSUDD	0.1062E-01	CSUBM	-0.9223E-02	CLALP	0.4637E+01	CDALP	0.1389E-01	CHALP	-0.8114E-02
0.2095E+01	0.3872E+00	0.4694E+00	0.1061E-01	0.4694E+00	0.4580E+00	0.1059E-01	-0.9210E-02	0.4637E+01	0.1257E-01	-0.8114E-02	0.1257E-01	0.1257E-01	-0.1043E-01	-0.1043E-01	
0.2082E+01	0.3872E+00	0.4580E+00	0.1057E-01	0.4580E+00	0.4466E+00	0.1057E-01	-0.9253E-02	0.4641E+01	0.1146E-01	-0.1043E-01	0.1146E-01	0.1146E-01	-0.0931E-01	-0.0931E-01	
0.2075E+01	0.3872E+00	0.4466E+00	0.1055E-01	0.4466E+00	0.4352E+00	0.1055E-01	-0.9218E-02	0.4692E+01	0.1035E-01	-0.0931E-01	0.1035E-01	0.1035E-01	-0.0820E-01	-0.0820E-01	
0.2072E+01	0.3872E+00	0.4352E+00	0.1052E-01	0.4352E+00	0.4238E+00	0.1052E-01	-0.9187E-02	0.4717E+01	0.0924E-01	-0.0820E-01	0.0924E-01	0.0924E-01	-0.0813E-01	-0.0813E-01	
0.2072E+01	0.3872E+00	0.4238E+00	0.1051E-01	0.4238E+00	0.4124E+00	0.1051E-01	-0.9165E-02	0.4731E+01	0.0813E-01	-0.0813E-01	0.0813E-01	0.0813E-01	-0.0702E-01	-0.0702E-01	
0.2072E+01	0.3872E+00	0.4124E+00	0.1051E-01	0.4124E+00	0.4010E+00	0.1051E-01	-0.9153E-02	0.4745E+01	0.0702E-01	-0.0702E-01	0.0702E-01	0.0702E-01	-0.0591E-01	-0.0591E-01	
0.2072E+01	0.3872E+00	0.4010E+00	0.1052E-01	0.4010E+00	0.3896E+00	0.1052E-01	-0.9177E-02	0.4759E+01	0.0591E-01	-0.0591E-01	0.0591E-01	0.0591E-01	-0.0480E-01	-0.0480E-01	
0.2072E+01	0.3872E+00	0.3896E+00	0.1052E-01	0.3896E+00	0.3782E+00	0.1052E-01	-0.9205E-02	0.4773E+01	0.0480E-01	-0.0480E-01	0.0480E-01	0.0480E-01	-0.0369E-01	-0.0369E-01	
0.2072E+01	0.3872E+00	0.4403E+00	0.1050E-01	0.4403E+00	0.4289E+00	0.1050E-01	-0.9239E-02	0.4787E+01	0.0369E-01	-0.0369E-01	0.0369E-01	0.0369E-01	-0.0258E-01	-0.0258E-01	
0.2072E+01	0.3872E+00	0.4289E+00	0.1050E-01	0.4289E+00	0.4175E+00	0.1050E-01	-0.9204E-02	0.4801E+01	0.0258E-01	-0.0258E-01	0.0258E-01	0.0258E-01	-0.0147E-01	-0.0147E-01	
0.2009E+01	0.3872E+00	0.4175E+00	0.1048E-01	0.4175E+00	0.4061E+00	0.1048E-01	-0.9190E-02	0.4815E+01	0.0147E-01	-0.0147E-01	0.0147E-01	0.0147E-01	-0.0036E-01	-0.0036E-01	
0.2120E+01	0.3872E+00	0.4061E+00	0.1042E-01	0.4061E+00	0.3947E+00	0.1042E-01	-0.9226E-02	0.4829E+01	0.0036E-01	-0.0036E-01	0.0036E-01	0.0036E-01	-0.0025E-01	-0.0025E-01	
0.21319E+03	-0.10180E+02	0.64301E+02	-0.37718E+03	0.11820E+05	-0.37718E+03	0.11820E+05	-0.37718E+03	0.11820E+05	-0.37718E+03	0.11820E+05	-0.37718E+03	0.11820E+05	-0.37718E+03	0.11820E+05	-0.37718E+03

AERO COEFFICIENTS

ALPHZ	AMAZ	CSUBL	CSUBD	CUSUM	CLALP	COALP	CHALP
0.2048E+01	0.353E+00	0.423E+00	0.1058E-01	-0.920E-02	0.633E+01	0.1166E+01	-0.1488E+01
0.2092E+01	0.353E+00	0.423E+00	0.1058E-01	-0.920E-02	0.636E+01	0.1166E+01	-0.1492E+01
0.2797E+01	0.353E+00	0.423E+00	0.1058E-01	-0.920E-02	0.639E+01	0.1166E+01	-0.1518E+01
0.2688E+01	0.353E+00	0.423E+00	0.1058E-01	-0.920E-02	0.639E+01	0.1166E+01	-0.1508E+01
0.2567E+01	0.353E+00	0.423E+00	0.1058E-01	-0.917E-02	0.642E+01	0.1166E+01	-0.1572E+01
0.2466E+01	0.353E+00	0.395E+00	0.1049E-01	-0.913E-02	0.643E+01	0.1166E+01	-0.1602E+01
0.2391E+01	0.353E+00	0.367E+00	0.1042E-01	-0.911E-02	0.648E+01	0.1166E+01	-0.1622E+01
0.2376E+01	0.353E+00	0.340E+00	0.1042E-01	-0.911E-02	0.648E+01	0.1166E+01	-0.1622E+01
0.2376E+01	0.353E+00	0.313E+00	0.1042E-01	-0.911E-02	0.648E+01	0.1166E+01	-0.1622E+01
0.2528E+01	0.353E+00	0.286E+00	0.1042E-01	-0.915E-02	0.643E+01	0.1166E+01	-0.1610E+01
0.2638E+01	0.353E+00	0.259E+00	0.1038E-01	-0.919E-02	0.640E+01	0.1166E+01	-0.1592E+01
0.2753E+01	0.353E+00	0.232E+00	0.1038E-01	-0.922E-02	0.637E+01	0.1166E+01	-0.1592E+01
0.2857E+01	0.353E+00	0.205E+00	0.1037E-01	-0.923E-02	0.633E+01	0.1166E+01	-0.1592E+01
0.2930E+01	0.353E+00	0.178E+00	0.1037E-01	-0.923E-02	0.633E+01	0.1166E+01	-0.1592E+01
0.2962E+01	0.353E+00	0.151E+00	0.1036E-01	-0.920E-02	0.639E+01	0.1166E+01	-0.1588E+01
10 0.2318E+05	-0.12119E+02	0.3303E+02	-0.39011E+03	0.13788E+03	-0.40792E+04		

AERO COEFFICIENTS

ALPHZ	AMAZ	CSUBL	CSUBD	CUSUM	CLALP	COALP	CHALP
0.2681E+01	0.3233E+00	0.4128E+00	0.1058E-01	-0.920E-02	0.628E+01	0.1166E+01	-0.1703E+01
0.2622E+01	0.3233E+00	0.4063E+00	0.1052E-01	-0.918E-02	0.629E+01	0.1166E+01	-0.1702E+01
0.2531E+01	0.3233E+00	0.3999E+00	0.1051E-01	-0.915E-02	0.631E+01	0.1166E+01	-0.1705E+01
0.2417E+01	0.3233E+00	0.3931E+00	0.1048E-01	-0.912E-02	0.634E+01	0.1166E+01	-0.1709E+01
0.2310E+01	0.3233E+00	0.3862E+00	0.1045E-01	-0.909E-02	0.637E+01	0.1166E+01	-0.1712E+01
0.2212E+01	0.3233E+00	0.3794E+00	0.1042E-01	-0.905E-02	0.639E+01	0.1166E+01	-0.1712E+01
0.2092E+01	0.3233E+00	0.3726E+00	0.1042E-01	-0.902E-02	0.641E+01	0.1166E+01	-0.1712E+01
0.2107E+01	0.3233E+00	0.3658E+00	0.1042E-01	-0.903E-02	0.641E+01	0.1166E+01	-0.1712E+01
0.2216E+01	0.3233E+00	0.3590E+00	0.1043E-01	-0.905E-02	0.640E+01	0.1166E+01	-0.1712E+01
0.2259E+01	0.3233E+00	0.3522E+00	0.1043E-01	-0.907E-02	0.639E+01	0.1166E+01	-0.1712E+01
0.2371E+01	0.3233E+00	0.3454E+00	0.1047E-01	-0.911E-02	0.639E+01	0.1166E+01	-0.1712E+01
0.2480E+01	0.3233E+00	0.3386E+00	0.1050E-01	-0.914E-02	0.635E+01	0.1166E+01	-0.1712E+01
0.2591E+01	0.3233E+00	0.4026E+00	0.1052E-01	-0.917E-02	0.638E+01	0.1166E+01	-0.1707E+01
0.2664E+01	0.3233E+00	0.4108E+00	0.1053E-01	-0.919E-02	0.638E+01	0.1166E+01	-0.1705E+01
0.2692E+01	0.3233E+00	0.4144E+00	0.1053E+01	-0.919E-02	0.628E+01	0.1166E+01	-0.1703E+01
11 0.25715E+05	-0.16522E+02	-0.40552E+02	-0.37622E+03	0.15230E+03	-0.40313E+04		

AERO COEFFICIENTS

ALPMZ	AMCZ	CSUBL	CSUBD	CSUM	CLALP	CDALP	CHALP
0.20425E+01	0.2915E+00	0.3415E+00	0.1043E+01	-0.9081E-02	0.4290E+01	0.1146E-01	-0.1776E-01
0.2114E+01	0.2915E+00	0.3515E+00	0.1044E+01	-0.9081E-02	0.4303E+01	0.1146E-01	-0.1776E-01
0.2119E+01	0.2915E+00	0.3448E+00	0.1042E+01	-0.9036E-02	0.4328E+01	0.1146E-01	-0.1743E-01
0.2083E+01	0.2915E+00	0.3322E+00	0.1040E+01	-0.9002E-02	0.4349E+01	0.1146E-01	-0.1720E-01
0.1988E+01	0.2915E+00	0.3193E+00	0.1041E+01	-0.8944E-02	0.4359E+01	0.9337E-02	-0.1847E-01
0.1786E+01	0.2915E+00	0.3092E+00	0.1042E+01	-0.8893E-02	0.4359E+01	0.7772E-02	-0.1945E-01
0.1611E+01	0.2915E+00	0.2998E+00	0.1043E+01	-0.8854E-02	0.4359E+01	0.6537E-02	-0.2008E-01
0.1451E+01	0.2915E+00	0.2915E+00	0.1043E+01	-0.8841E-02	0.4359E+01	0.5977E-02	-0.2088E-01
0.1314E+01	0.2915E+00	0.2832E+00	0.1042E+01	-0.8848E-02	0.4359E+01	0.5220E-02	-0.2088E-01
0.1184E+01	0.2915E+00	0.2749E+00	0.1042E+01	-0.8877E-02	0.4359E+01	0.4220E-02	-0.2088E-01
0.1054E+01	0.2915E+00	0.2666E+00	0.1042E+01	-0.8923E-02	0.4359E+01	0.2888E-02	-0.2088E-01
0.1929E+01	0.2915E+00	0.2583E+00	0.1042E+01	-0.8960E-02	0.4359E+01	0.1074E-01	-0.1746E-01
0.2179E+01	0.2915E+00	0.2500E+00	0.1042E+01	-0.9023E-02	0.4333E+01	0.1146E-01	-0.1736E-01
0.2283E+01	0.2915E+00	0.2417E+00	0.1042E+01	-0.9054E-02	0.4310E+01	0.1146E-01	-0.1755E-01
0.2283E+01	0.2915E+00	0.2334E+00	0.1045E+01	-0.9075E-02	0.4298E+01	0.1146E-01	-0.1770E-01
12	0.27048E+05	-0.20141E+02	-0.47465E+02	-0.39947E+03	0.15900E+05	-0.48726E+04	-0.1777E-01

AERO COEFFICIENTS

ALPMZ	AMCZ	CSUBL	CSUBD	CSUM	CLALP	CDALP	CHALP
0.1672E+01	0.2597E+00	0.2942E+00	0.1043E+01	-0.8936E-02	0.4318E+01	0.5818E-02	-0.2099E-01
0.1614E+01	0.2597E+00	0.2879E+00	0.1044E+01	-0.8907E-02	0.4318E+01	0.4884E-02	-0.2161E-01
0.1521E+01	0.2597E+00	0.2776E+00	0.1045E+01	-0.8741E-02	0.4318E+01	0.3280E-02	-0.2347E-01
0.1408E+01	0.2597E+00	0.2651E+00	0.1046E+01	-0.8704E-02	0.4318E+01	0.1260E-02	-0.2598E-01
0.1291E+01	0.2597E+00	0.2522E+00	0.1047E+01	-0.8645E-02	0.4318E+01	-0.7332E-03	-0.2832E-01
0.1184E+01	0.2597E+00	0.2409E+00	0.1048E+01	-0.8594E-02	0.4318E+01	-0.2466E-02	-0.2849E-01
0.1088E+01	0.2597E+00	0.2294E+00	0.1049E+01	-0.8558E-02	0.4318E+01	-0.3744E-02	-0.2733E-01
0.1048E+01	0.2597E+00	0.2170E+00	0.1049E+01	-0.8540E-02	0.4318E+01	-0.4291E-02	-0.2700E-01
0.1158E+01	0.2597E+00	0.2378E+00	0.1048E+01	-0.8540E-02	0.4318E+01	-0.4048E-02	-0.2753E-01
0.1288E+01	0.2597E+00	0.2470E+00	0.1048E+01	-0.8624E-02	0.4318E+01	-0.3054E-02	-0.2866E-01
0.1362E+01	0.2597E+00	0.2601E+00	0.1046E+01	-0.8681E-02	0.4318E+01	0.1461E-02	-0.2850E-01
0.1479E+01	0.2597E+00	0.2730E+00	0.1045E+01	-0.8719E-02	0.4318E+01	0.4901E-03	-0.2850E-01
0.1581E+01	0.2597E+00	0.2830E+00	0.1045E+01	-0.8791E-02	0.4318E+01	0.2501E-02	-0.2814E-01
0.1594E+01	0.2597E+00	0.2931E+00	0.1043E+01	-0.8791E-02	0.4318E+01	0.4266E-02	-0.2199E-01
0.1686E+01	0.2597E+00	0.2986E+00	0.1043E+01	-0.8827E-02	0.4318E+01	0.5515E-02	-0.2113E-01
13	0.28425E+05	-0.17313E+02	-0.42038E+02	-0.45351E+03	0.15900E+05	-0.45846E+04	-0.2076E-01

AERO COEFFICIENTS

	ALPHZ	AMACZ	CSUBL	CSUBD	CSUBH	CLALP	CDALP	CHALP
1	0.2280E+00	0.2280E+00	0.2000E+00	0.1052E+01	-0.8114E+02	0.4277E+01	-0.5720E+02	-0.2065E+01
2	0.2280E+00	0.2280E+00	0.1945E+00	0.1052E+01	-0.8185E+02	0.4277E+01	-0.5720E+02	-0.2065E+01
3	0.2280E+00	0.2280E+00	0.1843E+00	0.1053E+01	-0.8339E+02	0.4277E+01	-0.5720E+02	-0.2065E+01
4	0.2280E+00	0.2280E+00	0.1749E+00	0.1054E+01	-0.8522E+02	0.4277E+01	-0.5720E+02	-0.2065E+01
5	0.2280E+00	0.2280E+00	0.1551E+00	0.1054E+01	-0.8828E+02	0.4277E+01	-0.5720E+02	-0.2065E+01
6	0.2280E+00	0.2280E+00	0.1478E+00	0.1057E+01	-0.9173E+02	0.4277E+01	-0.5720E+02	-0.2065E+01
7	0.2280E+00	0.2280E+00	0.1398E+00	0.1057E+01	-0.9368E+02	0.4277E+01	-0.5720E+02	-0.2065E+01
8	0.2280E+00	0.2280E+00	0.1364E+00	0.1058E+01	-0.9208E+02	0.4277E+01	-0.5720E+02	-0.2065E+01
9	0.2280E+00	0.2280E+00	0.1379E+00	0.1057E+01	-0.8127E+02	0.4277E+01	-0.5720E+02	-0.2065E+01
10	0.2280E+00	0.2280E+00	0.1442E+00	0.1057E+01	-0.8164E+02	0.4277E+01	-0.5720E+02	-0.2065E+01
11	0.2280E+00	0.2280E+00	0.1544E+00	0.1056E+01	-0.8203E+02	0.4277E+01	-0.5720E+02	-0.2065E+01
12	0.2280E+00	0.2280E+00	0.1648E+00	0.1055E+01	-0.8259E+02	0.4277E+01	-0.5720E+02	-0.2065E+01
13	0.2280E+00	0.2280E+00	0.1747E+00	0.1054E+01	-0.8318E+02	0.4277E+01	-0.5720E+02	-0.2065E+01
14	0.2280E+00	0.2280E+00	0.1909E+00	0.1053E+01	-0.8369E+02	0.4277E+01	-0.5720E+02	-0.2065E+01
15	0.2280E+00	0.2280E+00	0.1989E+00	0.1052E+01	-0.8369E+02	0.4277E+01	-0.5720E+02	-0.2065E+01
16	0.2280E+00	0.2280E+00	0.2023E+00	0.1052E+01	-0.8421E+02	0.4277E+01	-0.5720E+02	-0.2065E+01
17	0.2990E+05	-0.9967E+01	-0.2636E+02	-0.5178E+03	0.15799E+05	-0.48050E+04	-0.5720E+02	-0.2065E+01
18	0.3147E+05	-0.5023E+01	0.1368E+02	-0.5732E+03	0.15828E+05	-0.51820E+04	-0.5720E+02	-0.2065E+01
19	0.3461E+05	-0.3333E+02	-0.4346E+02	-0.5948E+03	-0.31653E+04	-0.52807E+04	-0.5720E+02	-0.2065E+01
20	0.3503E+05	-0.3403E+02	-0.1010E+03	-0.5948E+03	-0.32712E+04	-0.53661E+04	-0.5720E+02	-0.2065E+01
21	0.3503E+05	-0.4830E+03	0.11138E+04	-0.58507E+03	-0.44655E+04	-0.48739E+04	-0.5720E+02	-0.2065E+01
22	0.3503E+05	-0.4830E+03	0.11138E+04	0.15700E+03	-0.44655E+04	-0.48739E+04	-0.5720E+02	-0.2065E+01
23	0.3619E+05	-0.38570E+02	-0.12364E+03	0.15700E+03	-0.32051E+04	-0.55629E+04	-0.5720E+02	-0.2065E+01
24	0.3635E+05	-0.5380E+02	0.2554E+02	0.48487E+02	-0.35664E+04	-0.57061E+04	-0.5720E+02	-0.2065E+01
25	0.3635E+05	-0.5380E+02	0.2024E+02	0.42370E+02	-0.38428E+04	-0.59418E+04	-0.5720E+02	-0.2065E+01
26	0.3635E+05	-0.8270E+02	0.2334E+03	0.57233E+02	-0.42160E+04	-0.71885E+04	-0.5720E+02	-0.2065E+01
27	0.3635E+05	-0.1301E+03	0.55798E+03	0.53110E+02	-0.45318E+04	-0.82825E+04	-0.5720E+02	-0.2065E+01
28	0.3635E+05	-0.1301E+03	0.55798E+03	0.62229E+02	-0.53134E+04	-0.10771E+05	-0.5720E+02	-0.2065E+01
29	0.3635E+05	-0.1891E+03	0.72581E+03	0.7928E+02	-0.60329E+04	-0.13629E+05	-0.5720E+02	-0.2065E+01
30	0.3635E+05	-0.1891E+03	0.72581E+03	0.11877E+03	-0.71043E+04	-0.18031E+05	-0.5720E+02	-0.2065E+01
31	0.3635E+05	-0.2320E+03	0.91286E+03	0.10180E+03	-0.83120E+04	-0.23067E+05	-0.5720E+02	-0.2065E+01
32	0.3635E+05	-0.2320E+03	0.10867E+04	-0.19458E+02	-0.94358E+04	-0.27708E+05	-0.5720E+02	-0.2065E+01
33	0.3635E+05	-0.47330E+03	0.10725E+04	-0.44552E+01	-0.17110E+05	-0.30354E+05	-0.5720E+02	-0.2065E+01
34	0.3635E+05	-0.47330E+03	0.10725E+04	-0.44552E+01	-0.17110E+05	-0.30354E+05	-0.5720E+02	-0.2065E+01

SECTION INPUT DATA

ORDER OF SECTIONS: BLADE TIP(SECTION 1) TO HUB, THEN FROM FREE END OF FUSELAGE (SECTION 1) TO HUB

SECTION	M1	M2	M3	M4	M5	M6	M7	M	ERY	MRX	MRZ	IX	IY
1	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0708	-0.616	193.370	-1.854	0.024	0.0
2	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0832	-0.616	183.702	-1.713	0.048	0.0
3	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0801	-0.616	174.033	-1.472	0.048	0.0
4	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0801	-0.616	164.365	-1.429	0.048	0.0
5	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0790	-0.616	154.696	-1.284	0.048	0.0
6	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0774	-0.616	145.027	-1.137	0.048	0.0
7	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0774	-0.616	135.359	-0.984	0.048	0.0
8	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0774	-0.616	125.691	-0.834	0.048	0.0
9	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0774	-0.616	116.022	-0.680	0.048	0.0
10	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0774	-0.616	106.354	-0.525	0.048	0.0
11	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0774	-0.616	96.685	-0.371	0.048	0.0
12	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0774	-0.616	87.016	-0.219	0.048	0.0
13	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0900	-0.166	77.348	-0.071	0.045	0.0
14	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.1161	-0.016	67.680	0.071	0.031	0.0
15	1.0	1.0	0.0	0.0	1.0	0.0	0.0	0.1300	-0.112	58.011	0.205	0.020	0.0
16	1.0	1.0	0.0	0.0	1.0	0.0	0.0	0.1087	0.0	51.475	-0.316	0.077	0.0
17	1.0	1.0	0.0	0.0	1.0	0.0	0.0	0.0862	0.0	48.342	-0.275	0.0	0.0
18	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	48.342	-0.275	0.0	0.0
19	0.0	1.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	48.342	-0.275	0.0	0.0
20	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0662	0.0	48.342	-0.275	0.0	0.0
21	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0979	0.0	43.508	-0.221	0.025	0.0
22	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0207	0.0	38.674	-0.167	0.030	0.0
23	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0206	0.0	33.840	-0.124	0.013	0.0
24	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0209	0.0	29.005	-0.080	0.010	0.0
25	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0212	0.0	24.171	-0.052	0.008	0.0
26	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0225	0.0	19.337	-0.024	0.007	0.0
27	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0246	0.0	14.503	-0.013	0.008	0.0
28	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0345	0.0	9.668	-0.003	0.013	0.0
29	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0450	0.0	4.834	-0.001	0.024	0.0
30	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0260	0.0	0.0	0.0	0.383	0.0
31	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

SECTION	IZ	PHI	THETA	PBI	L(LEAD)	ROLL	YAW	ROLL
1	0.0242	-0.01455	-0.03061	0.09666	0.0	0.0	0.0	0.0
2	0.0484	-0.01455	-0.03061	0.10279	0.0668E+01	0.0	0.0	0.2268E+01
3	0.0484	-0.01461	-0.03124	0.10977	0.0668E+01	0.0	0.0	0.2268E+01
4	0.0484	-0.01475	-0.03282	0.11587	0.0668E+01	0.0	0.0	0.2268E+01
5	0.0484	-0.01499	-0.03512	0.12232	0.0668E+01	0.0	0.0	0.2268E+01
6	0.0484	-0.01527	-0.03765	0.12872	0.0668E+01	0.0	0.0	0.2268E+01
7	0.0484	-0.01555	-0.04093	0.13507	0.0668E+01	0.0	0.0	0.2268E+01
8	0.0484	-0.01578	-0.04493	0.14135	0.0668E+01	0.0	0.0	0.2268E+01
9	0.0484	-0.01594	-0.04932	0.14755	0.0668E+01	0.0	0.0	0.2268E+01
10	0.0484	-0.01608	-0.05401	0.15364	0.0668E+01	0.0	0.0	0.2268E+01
11	0.0512	-0.01593	-0.04935	0.15962	0.0668E+01	0.0	0.0	0.2268E+01
12	0.0488	-0.01572	-0.04299	0.16539	0.0668E+01	0.0	0.0	0.2268E+01
13	0.0455	-0.01532	-0.04127	0.17113	0.0668E+01	0.0	0.0	0.2268E+01
14	0.0315	-0.01471	-0.03871	0.17711	0.0668E+01	0.0	0.0	0.2268E+01
15	0.0198	-0.01386	-0.03543	0.18337	0.0341E+01	0.0	0.0	0.2268E+01
16	0.0772	-0.01298	-0.03480	0.22882	0.0	0.0	0.0	0.0
17	0.0	0.0	0.0	0.22859	0.0	0.0	0.0	0.0
18	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
20	0.0253	-0.01298	-0.03480	0.22859	0.0	0.0	0.0	0.0
21	0.0303	-0.01115	-0.03044	0.23416	0.0	0.0	0.0	0.0
22	0.0127	-0.01115	-0.03044	0.24131	0.0	0.0	0.0	0.0
23	0.0101	-0.00908	-0.02471	0.24523	0.0	0.0	0.0	0.0
24	0.0076	-0.00908	-0.02471	0.24971	0.0	0.0	0.0	0.0
25	0.0071	-0.00579	-0.01630	0.24311	0.0	0.0	0.0	0.0
26	0.0075	-0.00579	-0.01630	0.23650	0.0	0.0	0.0	0.0
27	0.0133	-0.00225	-0.00672	0.22477	0.0	0.0	0.0	0.0
28	0.0240	-0.00225	-0.00672	0.21159	0.0	0.0	0.0	0.0
29	0.3825	-0.00032	-0.00167	0.21047	0.0	0.0	0.0	0.0
30	0.3404	0.0	0.0	0.0	0.0	0.0	0.0	0.0
31	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

SECTION	CHORD	IND. VEL	XXXX	L(ELA)	(EJ3)	(EJ3)	(EJ3)	(EJ3)	(EJ3)
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.104E+02	0.514E+03	0.0	0.468E+01	0.136E+07	0.230E+07	0.594E+08	0.274E+04	0.0
3	0.104E+02	0.514E+03	0.0	0.468E+01	0.136E+07	0.230E+07	0.594E+08	0.274E+04	0.0
4	0.104E+02	0.514E+03	0.0	0.468E+01	0.136E+07	0.230E+07	0.594E+08	0.274E+04	0.0
5	0.104E+02	0.514E+03	0.0	0.468E+01	0.136E+07	0.230E+07	0.594E+08	0.274E+04	0.0
6	0.104E+02	0.514E+03	0.0	0.468E+01	0.136E+07	0.230E+07	0.594E+08	0.274E+04	0.0
7	0.104E+02	0.514E+03	0.0	0.468E+01	0.136E+07	0.230E+07	0.594E+08	0.274E+04	0.0
8	0.104E+02	0.514E+03	0.0	0.468E+01	0.136E+07	0.230E+07	0.594E+08	0.274E+04	0.0
9	0.104E+02	0.514E+03	0.0	0.468E+01	0.136E+07	0.230E+07	0.594E+08	0.274E+04	0.0
10	0.104E+02	0.514E+03	0.0	0.468E+01	0.136E+07	0.230E+07	0.594E+08	0.274E+04	0.0
11	0.104E+02	0.514E+03	0.0	0.468E+01	0.136E+07	0.230E+07	0.594E+08	0.274E+04	0.0
12	0.104E+02	0.514E+03	0.0	0.468E+01	0.136E+07	0.230E+07	0.594E+08	0.274E+04	0.0
13	0.104E+02	0.514E+03	0.0	0.468E+01	0.136E+07	0.230E+07	0.594E+08	0.274E+04	0.0
14	0.104E+02	0.514E+03	0.0	0.468E+01	0.136E+07	0.230E+07	0.594E+08	0.274E+04	0.0
15	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
16	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
17	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
18	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
20	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
22	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
23	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
24	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
26	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
27	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
28	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
29	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
30	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
31	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

SECTION	DEL X	DEL Y	DEL Z	NRDIZ	VZERIG	VZERIG	1/4K	1/4K	1/4K	TAU X	TAU Y	TAU Z	K
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
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
10	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
11	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
12	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
13	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
14	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
15	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
16	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
17	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
18	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
20	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
22	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
23	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
24	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
26	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
27	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
28	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
29	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
30	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
31	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

IS VALUES OF UPDATED LAMBDA MATRIX

1.00000+00	0.0	1.00000+00	0.0	1.00000+00	0.0	1.00000+00	0.0	1.00000+00	0.0	1.00000+00	0.0	1.00000+00	0.0
1.00000+00	0.0	1.00000+00	0.0	1.00000+00	0.0	1.00000+00	0.0	1.00000+00	0.0	1.00000+00	0.0	1.00000+00	0.0
1.00000+00	0.0	1.00000+00	0.0	1.00000+00	0.0	1.00000+00	0.0	1.00000+00	0.0	1.00000+00	0.0	1.00000+00	0.0

DIPHAS = 0.9999999999999999
 DILSIO = 0.1000000000000000
 DIPLOY = -0.2000000000000000
 INITIAL EIGENVALUE = -12.00 DETERMINANT = -3.00000+16 = -51270+16

IS VALUES OF UPDATED LAMBDA MATRIX

0.0 5.4210D+20 0.0 1.4293D-19 0.0 3.3081D-21 8.4495D-03 1.9195D-03 -7.3980D-04 -1.3260D-03
 -1.9498D-01 1.2130D+01 -1.3321D-03 7.0381D-04 1.0090D+00 5.0 6.7788D-03 1.5598D-03 8.8824D-03 1.1146D+02
 3.9328D-18 -1.3686D+11 2.1190D-10 1.3650D-12 1.8972D+00 7.1940D-01 -1.8972D+00 -7.1940D-01 4.5272D+00 2.4797D+00

DIPHAS # 0.99235884442583D+00 -0.1233850890212747D+00
 DTL610 # 0.1809158146730619D+02
 DPIVOT # -0.212084921995993D+02

15 VALUES OF UPDATED LAMBDA MATRIX

0.0 0.0 0.0 0.0 0.0 0.0 7.9935D-03 7.7846D+04 -7.0370D-04 -6.7664D-04
 -1.3386D-01 1.2619D+01 0.0 0.0 0.0 0.0 6.4481D-03 1.3315D-03 4.6690D-03 6.0333D-03
 3.1178D-13 2.4887D-13 -1.5926D-13 -2.9070D-13 1.6749D+00 2.7348D-01 -1.6749D+00 -2.7348D-01 5.7823D+00 9.5182D+01

DIPHAS # 0.97779235643485D+00 -0.200081606855495D+00
 DTL610 # 0.180893151068003D+02
 DPIVOT # 0.100840372303618D+03

LAST DETERMINANT VALUE = 0.492238D+16 = 0.228093D+14
 CURRENT DETERMINANT VALUE = 0.123628D+15 = 0.124468D+14
 EIGEN VALUE CONVERGENCE = 0.287671E-03

LAST EIGEN VALUE = 0.148730D+02 = 0.488430D+02
 CURRENT EIGEN VALUE = 0.148231D+02 = 0.488470D+02
 NEXT EIGEN VALUE = 0.148225D+02 = 0.488130D+02

15 VALUES OF UPDATED LAMBDA MATRIX

0.0 0.0 0.0 0.0 0.0 0.0 6.2033D-03 -7.3920D-04 6.4933D-04 2.5097D-04
 -7.0591D-02 1.2721D+01 -0.0826D-04 6.6593D-04 1.0000D+00 0.0 6.3307D-03 7.1770D-04 -5.0794D-04 -3.1230D-03
 -3.8416D-13 -3.6428D-14 1.3240D-14 1.0385D+00 -3.6219D-01 -1.0385D+00 3.6219D-01 3.5729D+00 0.12481D+00

DIPHAS # 0.98614097030808D+00 -0.194303338468591D+00
 DTL610 # 0.1809010827928212D+02
 DPIVOT # -0.149534686271479D-05

LAST DETERMINANT VALUE = 0.123628D+15 = 0.124468D+14
 CURRENT DETERMINANT VALUE = 0.123628D+13 = 0.214526D+13
 EIGEN VALUE CONVERGENCE = 0.590776E-03

LAST EIGEN VALUE = 0.148611D+02 = 0.488470D+02
 CURRENT EIGEN VALUE = 0.148225D+02 = 0.488130D+02
 NEXT EIGEN VALUE = 0.148222D+02 = 0.488112D+02

15 VALUES OF UPDATED LAMBDA MATRIX

0.0 0.0 0.0 0.0 0.0 0.0 6.2886D-03 -7.9165D-04 7.9382D-04 2.5174D-04

-7.033D-02 1.2053D-01 -0.2100D-04 0.7431D-04 1.0000D+00 0.0 0.2920D-03 7.4170D-04 4.3790D-06 -4.3100D-04
-3.0161D-13 1.2000D-13 0.0565D-15 2.6200D-16 1.0000D+00 -3.3000D-01 -1.0000D+00 3.5000D-01 3.0750D-06 -1.2300D-08

DIAGN S 0.90014133030799D+00
D1 S 0.10000000000000000000
DPIVOT S 0.1035013233767200D+00
-0.1902991708700012D+00
0.0950210000037130D+00

LAST DETERMINANT VALUE =0.140342D+13 0.210020D+13
CURRENT DETERMINANT VALUE = 0.140245D+10 0.000000D+00
EIGEN VALUE CONVERGENCE = 0.370000E-08

LAST EIGEN VALUE =0.140223D+02 0.402013D+02
CURRENT EIGEN VALUE =0.140222D+02 0.402012D+02
NEXT EIGEN VALUE =0.140222D+02 0.402012D+02

AEROELASTIC STABILITY
STATE VECTORS
S-WASPLATE DEFLECTIONS

0 PER REV.

X(=1) = 0.0 0.0
 Y(=0) = 0.0 0.0
 Z(=1) = 0.0 0.0

BLADE 1 0 PER REV SHAPE VECTOR (LOCAL AXTS SYS.)

SECTION	UX, RADIAL DEFL. (+, OUTBOARD)	UY, INPLANE DEFL. (+, ROTATION DIR.)	UZ, VERT. DEFL. (+, UPWARD)
TIP			
1	6.2873E-03	-7.9046E-04	-1.0000E+00
2	6.2873E-03	-7.9046E-04	-9.1893E-01
3	5.4752E-03	-7.6374E-04	-8.7742E-01
4	4.2602E-03	-7.6814E-04	-8.1583E-01
5	2.8102E-03	-7.4842E-04	-7.5316E-01
6	4.9021E-04	-7.2417E-04	-6.9089E-01
7	-1.1667E-03	-7.1622E-04	-6.2805E-01
8	-2.3721E-03	-7.1903E-04	-5.6724E-01
9	-3.1376E-03	-7.2625E-04	-5.0620E-01
10	-3.4655E-03	-7.3171E-04	-4.4588E-01
11	-3.4403E-03	-7.2776E-04	-3.8631E-01
12	-3.1010E-03	-7.0872E-04	-3.2843E-01
13	-2.5265E-03	-6.7180E-04	-2.7203E-01
14	-1.8180E-03	-6.1855E-04	-2.1795E-01
15	-1.0904E-03	-5.5559E-04	-1.6696E-01
16	-4.189E-03	-1.6558E-04	-1.3378E-01
17	2.8547E-03	3.1137E-04	-1.1674E-01
18	2.8547E-03	3.1137E-04	-1.2042E-01
19	-1.418E-03	-1.6558E-04	-1.2042E-01
20	-8.0192E-04	-1.0278E-04	-1.1674E-01
21	-6.0192E-04	-1.0278E-04	-9.5656E-02
22	-4.4397E-04	-5.1891E-05	-7.3672E-02
23	-4.4397E-04	-5.1891E-05	-5.1815E-02
24	-1.2462E-04	-1.4142E-05	-3.6510E-02
25	-1.2462E-04	-1.4142E-05	-2.2332E-02
26	-7.6342E-06	-7.8622E-07	-1.1780E-02
27	-7.6342E-06	-7.8622E-07	-5.0581E-03
28	-2.4604E-07	-2.0799E-08	-1.4694E-03
29	1.4602E-10	1.5194E-10	-1.5118E-04
30	1.4602E-10	1.5194E-10	-1.4258E-04
31	1.4602E-10	1.5194E-10	-1.4258E-04
MID			
31	3.1941E-09	2.9425E-09	-1.3161E-08
30	3.1941E-09	2.9425E-09	-1.3161E-08
29	3.1941E-09	2.9425E-09	-1.3161E-08
28	3.1941E-09	2.9425E-09	-1.3161E-08
27	3.1941E-09	2.9425E-09	-1.3161E-08
26	3.1941E-09	2.9425E-09	-1.3161E-08
25	3.1941E-09	2.9425E-09	-1.3161E-08
24	3.1941E-09	2.9425E-09	-1.3161E-08
23	3.1941E-09	2.9425E-09	-1.3161E-08
22	3.1941E-09	2.9425E-09	-1.3161E-08
21	3.1941E-09	2.9425E-09	-1.3161E-08
20	3.1941E-09	2.9425E-09	-1.3161E-08
19	3.1941E-09	2.9425E-09	-1.3161E-08
18	3.1941E-09	2.9425E-09	-1.3161E-08
17	3.1941E-09	2.9425E-09	-1.3161E-08
16	3.1941E-09	2.9425E-09	-1.3161E-08
15	3.1941E-09	2.9425E-09	-1.3161E-08
14	3.1941E-09	2.9425E-09	-1.3161E-08
13	3.1941E-09	2.9425E-09	-1.3161E-08
12	3.1941E-09	2.9425E-09	-1.3161E-08
11	3.1941E-09	2.9425E-09	-1.3161E-08
10	3.1941E-09	2.9425E-09	-1.3161E-08
9	3.1941E-09	2.9425E-09	-1.3161E-08
8	3.1941E-09	2.9425E-09	-1.3161E-08
7	3.1941E-09	2.9425E-09	-1.3161E-08
6	3.1941E-09	2.9425E-09	-1.3161E-08
5	3.1941E-09	2.9425E-09	-1.3161E-08
4	3.1941E-09	2.9425E-09	-1.3161E-08
3	3.1941E-09	2.9425E-09	-1.3161E-08
2	3.1941E-09	2.9425E-09	-1.3161E-08
1	3.1941E-09	2.9425E-09	-1.3161E-08

PHI X, TWIST
(+, NUSE UP)

PHI Y, FLAPWISE SLOPE
(+, FLAP DOWN)

PHI Z, CHORDWISE SLOPE
(+, BLADE LEAD)

SECTION	N, AXIAL FORCE (+ , RADIAL OUTBOARD)	V, CHORDWISE SHEAR (+ , TOWARD L.E.)	V Z, FLAPWIRE SHEAR (+ , UPWARD)
TIP			
1	7.9445E+04	6.2517E+03	6.2161E+04
2	6.0021E+04	6.2790E+03	6.4304E+04
3	7.9172E+04	6.3354E+03	7.0300E+04
4	7.7211E+04	6.3798E+03	7.4126E+04
5	7.4428E+04	6.3942E+03	7.7622E+04
6	7.1015E+04	6.3832E+03	8.0874E+04
7	6.7075E+04	6.3517E+03	8.3553E+04
8	6.2724E+04	6.3031E+03	8.5632E+04
9	5.8034E+04	6.2401E+03	8.7027E+04
10	5.2975E+04	6.1601E+03	8.7687E+04
11	4.7453E+04	6.0493E+03	8.7365E+04
12	4.1934E+04	5.9047E+03	8.6072E+04
13	3.5845E+04	5.7061E+03	8.3620E+04
14	2.9717E+04	5.4235E+03	8.0120E+04
15	2.4590E+04	5.0515E+03	7.5812E+04
16	2.2621E+04	4.6068E+03	7.2987E+04
17	2.2497E+04	4.1797E+03	6.9153E+04
18	2.6046E+04	4.6719E+03	6.4568E+04
19	2.8046E+04	4.8719E+03	6.1122E+04
20	2.9497E+04	4.7979E+03	5.7859E+04
21	2.5042E+04	4.7040E+03	5.4722E+04
22	3.1590E+04	4.3243E+03	5.1452E+04
23	3.0160E+04	3.8428E+03	4.8032E+04
24	3.5254E+04	3.2672E+03	4.4485E+04
25	2.3423E+04	2.5740E+03	4.0715E+04
26	2.3716E+04	1.7550E+03	3.6819E+04
27	2.6397E+04	9.9453E+02	3.2822E+04
28	5.1076E+07	4.6932E+04	2.8249E+04
29	1.1076E+07	6.0595E+05	2.2149E+05
30	1.3022E+10	5.2043E+10	1.5991E+05
31	1.3062E+10	5.2043E+10	1.1656E+10
MJB			
1	1.3062E+10	4.8395E+10	1.1656E+10
2	1.2122E+10	4.8395E+10	1.0803E+10
3	1.1853E+10	4.8395E+10	1.0003E+10
4	1.1656E+10	4.8395E+10	9.2500E+09
5	1.1459E+10	4.8395E+10	8.5419E+09
6	1.1262E+10	4.8395E+10	7.8338E+09
7	1.1065E+10	4.8395E+10	7.1257E+09
8	1.0868E+10	4.8395E+10	6.4176E+09
9	1.0671E+10	4.8395E+10	5.7095E+09
10	1.0474E+10	4.8395E+10	5.0014E+09
11	1.0277E+10	4.8395E+10	4.2933E+09
12	1.0080E+10	4.8395E+10	3.5852E+09
13	9.883E+09	4.8395E+10	2.8771E+09
14	9.678E+09	4.8395E+10	2.1690E+09
15	9.473E+09	4.8395E+10	1.4609E+09
16	9.268E+09	4.8395E+10	7.5288E+08
17	9.063E+09	4.8395E+10	4.3967E+08
18	8.858E+09	4.8395E+10	1.2646E+08
19	8.653E+09	4.8395E+10	1.5525E+07
20	8.448E+09	4.8395E+10	1.8404E+06
21	8.243E+09	4.8395E+10	2.1283E+05
22	8.038E+09	4.8395E+10	2.4162E+04
23	7.833E+09	4.8395E+10	2.7041E+03

SECTION	Y, TORQUE (+,-) MUSE UP)	M Y, FLAPWISE MOMENT (+,-) TENSION UPPER FIBERS)	M Z, CHORDWISE MOMENT (+,-) TENSION T.E.)
24	-2.6045E+01	-4.9887E+01	-1.9339E+02
25	-2.7191E+01	-4.6411E+01	-1.9807E+02
26	-2.7264E+01	-4.7377E+01	-1.9844E+02
27	-2.5300E+01	-4.5014E+01	-1.9530E+02
28	-2.5311E+01	-4.2462E+01	-1.9592E+02
29	-2.3340E+01	-4.2272E+01	-1.9609E+02
30	-2.3965E+01	-3.1877E+01	-2.0044E+02
31	-2.3965E+01	-3.1877E+01	-2.0044E+02
MUB			
1	9.6209E+00	-1.3555E+00	-2.2299E+00
2	2.3643E+01	-1.5826E+01	-6.7832E+00
3	3.5830E+01	-1.5870E+01	-1.7724E+01
4	6.6033E+01	-9.3374E+00	-3.4249E+01
5	5.6022E+01	-1.7920E+00	-5.5428E+01
6	6.4266E+01	4.5513E+00	-8.0533E+01
7	7.1474E+01	9.5374E+00	-1.0930E+02
8	7.7865E+01	1.3531E+01	-1.4166E+02
9	8.3393E+01	1.7461E+01	-1.7805E+02
10	8.8334E+01	2.3001E+01	-2.1930E+02
11	9.3526E+01	3.4031E+01	-2.6733E+02
12	9.9324E+01	4.2238E+01	-3.2099E+02
13	1.0051E+02	7.7515E+01	-3.8512E+02
14	1.0084E+02	9.9324E+01	-4.6437E+02
15	1.0153E+02	1.5584E+02	-5.6544E+02
16	-6.1249E+00	2.9622E+02	-6.4780E+02
17	-4.3918E+00	4.1044E+02	-8.4780E+02
18	-2.1201E+01	6.0884E+02	-1.0892E+03
19	-1.1444E+00	8.0984E+02	-1.3831E+03
20	-2.6050E+01	1.0160E+03	-1.7474E+03
21	-2.4441E+01	1.1602E+03	-2.1598E+03
22	-2.3502E+01	1.3197E+03	-2.6457E+03
23	-2.3502E+01	1.4184E+03	-3.1950E+03
24	-1.9097E+01	1.4184E+03	-3.8265E+03
25	-2.1568E+01	1.9733E+03	-4.5222E+03
26	-8.4111E+00	2.6013E+03	-5.3346E+03
27	-9.6059E+00	3.5204E+03	-6.2807E+03
28	4.1586E+01	4.4211E+03	-7.3953E+03
29	4.3262E+00	5.6619E+03	-8.5757E+03
30	4.3262E+00	5.6619E+03	-8.5757E+03
31	4.3262E+00	5.6619E+03	-8.5757E+03
MUB			

DISK PLANE AXIS SYSTEM

SECTION	UX, RADIAL DEFL. (+, OUTBOARD)	UY, INPLANE DEFL. (+, ROTATION DIR.)	UZ, VERT. DEFL. (+, UPWARD)
TIP	1 7218E-02	1 279E-01	-1.0024E+00
	2 5558E-02	1.188E-01	-2.4156E-01
	3 3483E-02	1.1011E-01	-2.8025E-01
	4 1812E-02	1.0172E-01	-3.1840E-01
	5 6227E-02	9.3665E-02	-3.5622E-01
	6 2682E-02	8.5854E-02	-3.9399E-01
	7 4394E-02	7.8207E-02	-4.3194E-01
	8 1801E-02	7.0668E-02	-4.7026E-01
	9 1466E-02	6.3212E-02	-5.0909E-01
	10 1.478E-02	5.5831E-02	-5.4859E-01
	11 1.389E-02	4.8510E-02	-5.8896E-01
	12 1.322E-02	4.1275E-02	-6.3028E-01
	13 9.605E-03	3.4168E-02	-6.7391E-01
	14 6.830E-03	2.7289E-02	-7.1949E-01
	15 4.945E-03	2.0761E-02	-7.6819E-01
	16 3.3954E-03	1.4592E-02	-8.1970E-01
	17 2.547E-03	1.0637E-02	-8.7370E-01
	18 2.057E-03	7.637E-03	-9.2982E-01
	19 2.0547E-03	5.1137E-03	-9.8847E-01
	20 2.0947E-03	3.1137E-03	-1.0482E-01
	21 1.319E-03	1.671E-03	-1.2042E-01
	22 1.4429E-03	1.1654E-03	-1.2042E-01
	23 9.370E-04	6.647E-04	-9.7074E-02
	24 4.9104E-04	3.3151E-04	-7.4834E-02
	25 2.5274E-04	2.1532E-04	-5.4800E-02
	26 2.824E-05	1.2060E-04	-3.7111E-02
	27 2.787E-05	4.7401E-05	-2.2661E-02
	28 2.532E-06	1.2267E-05	-1.1932E-02
	29 1.6627E-10	1.1021E-05	-5.1107E-03
	30 1.6626E-10	2.9422E-06	-1.4851E-03
	31 1.6626E-10	2.9422E-06	-1.5613E-04
HUB			-1.4258E-08
			-1.4258E-08
			-1.3161E-08

SECTION	PHIBX, TWIST (+, L.A.E.)	PHIBY, VERT. BENDING SLOPE (+, TIP DOWNWARD)	PHIBZ, INPLANE BEND. SLOPE (+, TIP DIR. OF ROT.)
TIP	1 8.8598E-04	2.3158E-04	5.1802E-04
	2 9.207E-04	1.9339E-04	0.2040E-06
	3 8.855E-04	1.6232E-04	9.3112E-04
	4 8.631E-04	1.5054E-04	1.7456E-05
	5 8.000E-04	1.3314E-04	6.0382E-04
	6 6.0714E-04	1.1394E-04	1.5241E-05
	7 6.685E-04	9.2287E-05	4.4163E-05
	8 7.2535E-04	6.7169E-05	8.0345E-04
	9 6.778E-04	3.8855E-05	5.4237E-05
	10 6.2534E-04	8.6481E-06	7.4924E-04
	11 5.6705E-04	-2.0418E-05	0.1604E-05
	12 5.0736E-04	-6.1082E-05	0.9780E-05
	13 4.023E-04	-9.8652E-05	1.1944E-04
	14 3.7078E-04	-9.4931E-05	1.4106E-04
			1.6035E-04
			1.8033E-04
			6.0394E-04

SECTION	NB, RADIAL FORCE (+, OUTBOARD)	MBY, INPLANE SHEAR (+, DIR. OF ROT.)	VBZ, VERT. SHEAR (+, UPWARD)
15	3.0918E-04	5.0941E-03	2.1204E-04
16	2.6394E-04	4.6920E-03	2.2400E-04
17	2.8048E-04	4.8719E-03	2.2658E-04
18	2.6048E-04	4.6719E-03	2.2658E-04
19	2.6048E-04	4.6719E-03	2.2658E-04
20	2.6048E-04	4.6719E-03	2.2658E-04
21	2.6048E-04	4.6719E-03	2.2658E-04
22	3.5795E-04	4.7798E-03	2.2658E-04
23	3.1702E-04	4.1078E-03	2.2658E-04
24	3.7702E-04	4.8210E-03	2.2658E-04
25	2.4699E-04	3.2255E-03	2.0978E-04
26	2.4573E-04	2.9134E-03	1.8697E-04
27	3.0221E-05	1.7788E-03	1.5350E-04
28	1.2561E-05	1.0051E-03	1.0779E-04
29	5.4886E-07	4.7192E-04	6.5835E-05
30	1.1302E-10	6.2600E-05	3.7834E-05
31	1.1302E-10	5.2043E-10	-2.9680E-06
MUB		5.2043E-10	-1.1656E-10

SECTION

MBZ, INPLANE MOMENT
(+, TENSION T.E.)

SECTION	NB, RADIAL FORCE (+, OUTBOARD)	MBY, INPLANE SHEAR (+, DIR. OF ROT.)	VBZ, VERT. SHEAR (+, UPWARD)
1	-3.0360E+00	-4.5313E+01	-1.5880E+01
2	-5.0770E+00	-9.7145E+01	-3.6800E+01
3	-6.0527E+00	-1.4066E+02	-5.5620E+01
4	-6.5933E+00	-1.7566E+02	-7.2948E+01
5	-1.0017E+01	-2.0248E+02	-8.8635E+01
6	-1.1402E+01	-2.2086E+02	-1.0246E+02
7	-1.2716E+01	-2.3154E+02	-1.1541E+02
8	-1.3973E+01	-2.3500E+02	-1.2662E+02
9	-1.5145E+01	-2.3106E+02	-1.3694E+02
10	-1.6292E+01	-2.2337E+02	-1.4579E+02
11	-1.6350E+01	-2.0429E+02	-1.5292E+02
12	-1.9318E+01	-1.8781E+02	-1.5823E+02
13	-2.0334E+01	-1.8059E+02	-1.6452E+02
14	-2.1402E+01	-1.6881E+02	-1.7046E+02
15	-2.2541E+01	-1.5121E+02	-1.7637E+02
16	-2.4642E+01	-9.6358E+01	-1.8122E+02
17	-2.6823E+01	-9.1598E+01	-1.8009E+02
18	-2.9233E+01	-9.1598E+01	-1.8122E+02
19	-2.3030E+01	-8.0823E+01	-1.8122E+02
20	-2.3682E+01	-6.1691E+01	-1.8678E+02
21	-2.3728E+01	-3.6853E+01	-1.9049E+02
22	-2.3808E+01	-3.5333E+01	-1.9049E+02
23	-2.3858E+01	-3.4133E+01	-1.9091E+02
24	-2.3858E+01	-3.3255E+01	-2.0021E+02
25	-2.3963E+01	-3.2657E+01	-2.0038E+02
26	-2.3963E+01	-3.2294E+01	-2.0048E+02
27	-2.3968E+01	-3.1941E+01	-2.0051E+02
28	-2.3968E+01	-3.1670E+01	-2.0058E+02
29	-2.3968E+01	-3.1454E+01	-2.0063E+02
30	-2.3968E+01	-3.1277E+01	-2.0064E+02
31	-2.3968E+01	-3.1177E+01	-2.0064E+02
MUB		-3.1177E+01	-2.0064E+02

SECTION

MBY, VERTICAL MOMENT
(+, TENSION UPPER FIBERS)

SECTION

TB, TORQUE
(+, NOSE UP)

TIP

1 1.0700E+00 6.3529E+00
2 3.666E+01 5.417E+00
3 6.211E+01 3.4948E+00
4 4.767E+01 1.9925E+00
5 5.8052E+01 4.0148E+01
6 7.424E+01 -8.2711E+01
7 6.066E+01 -1.5355E+00
8 4.092E+01 -1.9274E+00
9 1.1572E+01 -2.2047E+00
10 9.8534E+01 -2.5133E+00
11 1.0798E+02 -8.3011E+01
12 1.1401E+02 0.3726E+00
13 1.1770E+02 3.4920E+00
14 1.209E+02 5.576E+00
15 1.2470E+02 -7.3228E+00
16 1.287E+01 -1.4975E+01
17 1.3201E+01 -1.5032E+01
18 2.1201E+01 -1.5032E+01
19 -1.1414E+00 -7.0689E+00
20 -1.1283E+00 -7.0333E+00
21 1.196E+01 -7.8897E+00
22 1.1581E+00 -8.968E+00
23 1.7453E+00 -9.7089E+00
24 3.243E+00 -1.0820E+01
25 2.557E+00 -1.1363E+01
26 4.2162E+00 -1.2320E+01
27 3.6626E+00 -1.2571E+01
28 4.2294E+00 -1.3127E+01
29 4.4078E+00 -1.380E+01
30 4.3242E+00 -1.372E+01
31 4.3242E+00 -1.372E+01

HUB

-1.2762E+00 -6.3379E+01
-1.5393E+01 0.0297E+01
-1.4371E+01 9.1855E+01
-6.0196E+00 7.1978E+01
4.1148E+00 4.1373E+01
1.3827E+01 1.4114E+01
2.8909E+01 -7.4594E+00
3.2050E+01 -3.3783E+01
4.1993E+01 -3.6800E+01
5.4723E+01 -9.824E+01
7.4398E+01 -6.821E+01
9.2727E+01 -6.788E+01
1.1018E+02 -6.1848E+01
2.3550E+02 -1.058E+02
3.7328E+02 -1.3849E+02
5.6668E+02 -1.8180E+02
6.0966E+02 -1.4274E+02
6.0966E+02 -1.4274E+02
3.0168E+02 -1.4916E+02
3.0170E+02 -1.4917E+02
5.0358E+02 -1.2687E+02
4.5952E+02 -1.0371E+02
6.0033E+02 -7.9534E+01
1.213E+03 -5.1787E+01
1.8519E+03 -1.6275E+01
2.2248E+03 2.9274E+01
2.9429E+03 8.5862E+01
3.7776E+03 1.5177E+02
4.6978E+03 2.2468E+02
5.6619E+03 3.0171E+02
5.6619E+03 3.0171E+02

-2.0541E+00 -1.8956E+01
-7.4438E+00 2.2542E+00
-1.821E+01 7.5337E+00
-3.354E+01 1.1936E+01
-5.3230E+01 2.3372E+01
-7.6827E+01 3.9782E+01
-1.0407E+02 6.8888E+01
-1.3495E+02 8.9701E+01
-1.6977E+02 1.1166E+02
-2.0912E+02 1.4182E+02
-2.5428E+02 1.7289E+02
-3.0518E+02 2.0802E+02
-3.6184E+02 2.4832E+02
-4.2541E+02 2.9732E+02
-4.9795E+02 3.4896E+02
-5.7771E+02 4.0833E+02
-5.6531E+02 4.0833E+02
-5.7474E+02 4.0833E+02
-5.7481E+02 4.0833E+02
-6.1641E+02 4.3482E+02
-6.3778E+02 4.7688E+02
-6.9751E+02 5.2888E+02
-7.3453E+02 5.9032E+02
-7.6694E+02 6.7873E+02
-7.9237E+02 7.7907E+02
-8.0999E+02 8.9799E+02
-8.2131E+02 1.0385E+03
-8.2627E+02 1.1781E+03
-8.2754E+02 1.3175E+03
-8.2754E+02 1.3175E+03

DISK PLANE AXIS SYSTEM - POLAR COORDINATES
AMPLITUDE AND PHASE ANGLE

SECTION	TIP	UBX, RADIAL DEPL. (+, OUTBOARD)	UBY, INPLANE DEPL. (+, ROTATION DIR.)	USZ, VERT. DEPL. (+, UPWARD)	PHIBX, TWIST (+, LIE. UPWARD)	PHIBY, VERT. BENDING SLOPE (+, TIP DOWNWARD)	PHIBZ, INPLANE BEND. SLOPE (+, TIP DIR. OP ROT.)
1	1	3.7225E-02	1.2000E-01	1.4471E+00	9.1574E-04	6.3040E-03	1.5453E+00
2	2	3.5360E-02	1.1906E-01	1.4369E+00	9.1379E-04	6.3207E-03	1.5453E+00
3	3	3.3443E-02	1.1535E-01	1.4244E+00	9.0935E-04	6.3457E-03	1.5453E+00
4	4	3.1412E-02	1.1032E-01	1.4119E+00	9.0432E-04	6.3707E-03	1.5453E+00
5	5	2.9277E-02	1.0502E-01	1.4003E+00	8.9929E-04	6.3957E-03	1.5453E+00
6	6	2.7082E-02	9.9522E-02	1.3898E+00	8.9426E-04	6.4207E-03	1.5453E+00
7	7	2.4894E-02	9.3712E-02	1.3802E+00	8.8923E-04	6.4457E-03	1.5453E+00
8	8	2.2650E-02	8.7632E-02	1.3715E+00	8.8420E-04	6.4707E-03	1.5453E+00
9	9	2.0398E-02	8.1335E-02	1.3637E+00	8.7917E-04	6.4957E-03	1.5453E+00
10	10	1.8182E-02	7.4857E-02	1.3568E+00	8.7414E-04	6.5207E-03	1.5453E+00
11	11	1.5948E-02	6.8244E-02	1.3508E+00	8.6911E-04	6.5457E-03	1.5453E+00
12	12	1.3742E-02	6.1452E-02	1.3457E+00	8.6408E-04	6.5707E-03	1.5453E+00
13	13	1.1511E-02	5.4542E-02	1.3415E+00	8.5905E-04	6.5957E-03	1.5453E+00
14	14	9.3022E-03	4.7462E-02	1.3382E+00	8.5402E-04	6.6207E-03	1.5453E+00
15	15	7.1692E-03	4.0262E-02	1.3358E+00	8.4900E-04	6.6457E-03	1.5453E+00
16	16	5.1400E-03	3.2982E-02	1.3343E+00	8.4400E-04	6.6707E-03	1.5453E+00
17	17	3.2716E-03	2.5662E-02	1.3337E+00	8.3900E-04	6.6957E-03	1.5453E+00
18	18	1.6162E-03	1.8342E-02	1.3340E+00	8.3400E-04	6.7207E-03	1.5453E+00
19	19	2.8716E-03	1.0865E-01	1.3352E+00	8.2900E-04	6.7457E-03	1.5453E+00
20	20	2.8716E-03	1.0865E-01	1.3372E+00	8.2400E-04	6.7707E-03	1.5453E+00
21	21	2.1522E-03	1.1192E-01	1.3402E+00	8.1900E-04	6.7957E-03	1.5453E+00
22	22	1.4524E-03	1.1417E-01	1.3442E+00	8.1400E-04	6.8207E-03	1.5453E+00
23	23	9.4195E-04	1.1514E-01	1.3492E+00	8.0900E-04	6.8457E-03	1.5453E+00
24	24	6.0352E-04	1.1514E-01	1.3552E+00	8.0400E-04	6.8707E-03	1.5453E+00
25	25	2.9466E-04	1.1468E-01	1.3622E+00	8.0000E-04	6.8957E-03	1.5453E+00
26	26	7.4931E-05	1.1187E-01	1.3702E+00	7.9600E-04	6.9207E-03	1.5453E+00
27	27	2.7444E-05	1.0994E-01	1.3792E+00	7.9200E-04	6.9457E-03	1.5453E+00
28	28	2.5452E-06	1.0887E-01	1.3892E+00	7.8800E-04	6.9707E-03	1.5453E+00
29	29	2.2524E-10	7.6042E-01	1.4002E+00	7.8400E-04	6.9957E-03	1.5453E+00
30	30	2.2523E-10	7.6042E-01	1.4122E+00	7.8000E-04	7.0207E-03	1.5453E+00
31	31	2.2523E-10	7.6042E-01	1.4252E+00	7.7600E-04	7.0457E-03	1.5453E+00
MUB							
SECTION	TIP						
1	1	9.1574E-04	2.5564E-01	9.4829E-04	6.3040E-03	1.5453E+00	1.5453E+00
2	2	9.1379E-04	2.1349E-01	9.3116E-04	6.3207E-03	1.5453E+00	1.5453E+00
3	3	9.0935E-04	1.6793E-01	9.0935E-04	6.3457E-03	1.5453E+00	1.5453E+00
4	4	8.9929E-04	1.2202E-01	8.9592E-04	6.3707E-03	1.5453E+00	1.5453E+00
5	5	8.9426E-04	8.7202E-01	8.8253E-04	6.3957E-03	1.5453E+00	1.5453E+00
6	6	8.8923E-04	1.5719E-01	8.6929E-04	6.4207E-03	1.5453E+00	1.5453E+00
7	7	8.8420E-04	1.4024E-01	8.5602E-04	6.4457E-03	1.5453E+00	1.5453E+00
8	8	8.7917E-04	1.1949E-01	8.4277E-04	6.4707E-03	1.5453E+00	1.5453E+00
9	9	8.7414E-04	9.2336E-02	8.2952E-04	6.4957E-03	1.5453E+00	1.5453E+00
10	10	8.6911E-04	7.3732E-02	8.1627E-04	6.5207E-03	1.5453E+00	1.5453E+00
11	11	8.6408E-04	5.4878E-02	8.0302E-04	6.5457E-03	1.5453E+00	1.5453E+00
12	12	8.5905E-04	3.5672E-02	7.8977E-04	6.5707E-03	1.5453E+00	1.5453E+00
13	13	8.5402E-04	1.6981E-01	7.7652E-04	6.5957E-03	1.5453E+00	1.5453E+00

SECTION	MB, RADIAL FORCE (+, OUTBOARD)	MBY, INPLANE SHEAR (+, DIR. OF ROT.)	VBZ., VERT. SHEAR (+, UPWARD)
18	3.0274E+00	5.4749E+03	7.1742E+04
19	3.2175E+00	5.1007E+02	4.5347E+00
20	2.9635E+00	4.8002E+03	1.2534E+00
21	2.0237E+00	4.8789E+02	1.2840E+00
22	2.0237E+00	4.8789E+02	1.2100E+00
23	2.0237E+00	4.8789E+02	1.2100E+00
24	2.0237E+00	4.8789E+02	1.2100E+00
25	2.0237E+00	4.8789E+02	1.2100E+00
26	2.0237E+00	4.8789E+02	1.2100E+00
27	2.0237E+00	4.8789E+02	1.2100E+00
28	2.0237E+00	4.8789E+02	1.2100E+00
29	2.0237E+00	4.8789E+02	1.2100E+00
30	2.0237E+00	4.8789E+02	1.2100E+00
31	2.0237E+00	4.8789E+02	1.2100E+00

MUB

SECTION MB, RADIAL FORCE (+, OUTBOARD)

MBY, INPLANE SHEAR (+, DIR. OF ROT.)

VBZ., VERT. SHEAR (+, UPWARD)

SECTION	MB, RADIAL FORCE (+, OUTBOARD)	MBY, INPLANE SHEAR (+, DIR. OF ROT.)	VBZ., VERT. SHEAR (+, UPWARD)
1	3.0480E+00	2.0639E+00	1.0751E+01
2	3.1643E+00	5.0511E+00	3.6032E+01
3	7.0310E+00	1.7434E+00	3.0509E+00
4	0.7673E+00	1.7233E+00	7.1333E+01
5	1.0350E+01	1.7364E+00	3.0530E+00
6	1.1798E+01	1.2047E+00	0.9107E+01
7	1.2147E+01	1.7203E+00	3.0300E+00
8	1.4615E+01	1.7203E+00	1.0326E+02
9	1.4615E+01	1.7203E+00	3.0300E+00
10	1.5032E+01	1.7203E+00	1.1599E+02
11	1.6648E+01	1.7203E+00	1.2734E+02
12	1.6648E+01	1.7203E+00	3.0511E+00
13	1.9632E+01	1.6995E+00	1.3734E+02
14	2.0411E+01	1.6995E+00	1.4611E+02
15	2.1742E+01	1.6995E+00	1.5030E+02
16	2.2736E+01	1.6995E+00	1.6002E+02
17	2.4077E+01	1.6995E+00	1.7000E+02
18	2.5000E+01	1.6995E+00	1.7688E+02
19	2.5000E+01	1.6995E+00	1.8130E+02
20	2.5000E+01	1.6995E+00	1.8130E+02
21	2.5000E+01	1.6995E+00	1.9170E+02
22	2.5000E+01	1.6995E+00	1.9170E+02
23	2.5000E+01	1.6995E+00	1.9170E+02
24	2.5000E+01	1.6995E+00	1.9170E+02
25	2.5000E+01	1.6995E+00	1.9170E+02
26	2.5000E+01	1.6995E+00	1.9170E+02
27	2.5000E+01	1.6995E+00	1.9170E+02
28	2.5000E+01	1.6995E+00	1.9170E+02
29	2.5000E+01	1.6995E+00	1.9170E+02
30	2.5000E+01	1.6995E+00	1.9170E+02
31	2.5000E+01	1.6995E+00	1.9170E+02

MUB

SECTION TB, TORQUE (+, NOSE UP)

MBY, VERTICAL MOMENT (+, TENSION UPPER FIBERS)

MBZ, INPLANE MOMENT (+, TENSION Y.E.)

TIP	1	1.571E+01	5.412E+01	1.479E+00	-2.413E+00	2.362E+00	-1.040E+00
	2	2.427E+01	2.025E+01	6.172E+01	1.760E+00	7.870E+00	2.854E+00
	3	3.639E+01	1.016E+01	4.494E+01	1.722E+00	1.894E+01	2.656E+00
	4	4.771E+01	4.176E+02	7.223E+01	1.654E+00	3.560E+01	2.790E+00
	5	5.805E+01	6.915E+03	4.157E+01	1.471E+00	5.813E+01	2.729E+00
	6	6.742E+01	-1.222E+02	1.975E+01	7.957E+01	4.952E+01	2.663E+00
	7	7.601E+01	-2.019E+02	2.416E+01	-3.137E+01	1.203E+02	2.614E+00
	8	8.411E+01	-2.291E+02	3.991E+01	-6.380E+01	1.593E+02	2.581E+00
	9	9.159E+01	-2.407E+02	5.587E+01	-7.203E+01	2.332E+02	2.559E+00
	10	9.866E+01	-2.557E+02	7.400E+01	-7.386E+01	2.722E+02	2.543E+00
	11	1.079E+02	-7.761E+03	1.012E+02	-7.450E+01	3.072E+02	2.545E+00
	12	1.140E+02	-1.203E+02	1.146E+02	-6.293E+01	3.685E+02	2.545E+00
	13	1.175E+02	-2.940E+02	1.622E+02	-5.274E+01	4.366E+02	2.547E+00
	14	1.209E+02	-4.661E+02	2.567E+02	-4.281E+01	5.134E+02	2.547E+00
	15	1.242E+02	-5.665E+02	4.159E+02	-3.310E+01	6.032E+02	2.541E+00
	16	1.273E+02	-6.508E+01	5.671E+02	-2.709E+01	6.974E+02	2.541E+00
	17	1.306E+02	-6.412E+01	6.241E+02	-2.299E+01	7.974E+02	2.539E+00
	18	1.340E+02	-6.412E+01	6.241E+02	-2.299E+01	9.074E+02	2.539E+00
	19	1.374E+02	-1.720E+00	4.057E+02	-3.725E+01	1.027E+03	2.536E+00
	20	1.408E+02	-1.720E+00	4.057E+02	-3.725E+01	1.152E+03	2.536E+00
	21	1.442E+02	-1.568E+00	5.193E+02	-2.466E+01	1.282E+03	2.527E+00
	22	1.476E+02	-1.417E+00	6.676E+02	-1.559E+01	1.425E+03	2.514E+00
	23	1.509E+02	-1.269E+00	8.393E+02	-6.953E+02	1.580E+03	2.492E+00
	24	1.542E+02	-1.127E+00	1.021E+03	-4.263E+02	1.741E+03	2.459E+00
	25	1.574E+02	-1.003E+00	1.215E+03	-2.250E+03	1.907E+03	2.417E+00
	26	1.605E+02	-1.241E+00	1.415E+03	-9.654E+03	1.024E+03	2.364E+00
	27	1.635E+02	-1.287E+00	1.620E+03	1.315E+02	1.112E+03	2.304E+00
	28	1.664E+02	-1.245E+00	1.831E+03	2.917E+02	1.209E+03	2.249E+00
	29	1.692E+02	-1.250E+00	4.703E+03	4.015E+02	1.317E+03	2.249E+00
	30	1.719E+02	-1.255E+00	5.669E+03	4.783E+02	1.434E+03	2.184E+00
	31	1.745E+02	-1.255E+00	5.669E+03	5.323E+02	1.555E+03	2.131E+00

SUMMARY OF PREDICTED MODES					
MODE	DAMPING	FREQ IN	FREQ	CONVERGENCE	PREDOMINANT
		RAD PER SEC	PER REV	CRITERIA SATISFIED	MODE TYPE
1	-0.146222D+02	0.492812D+02	1.1073	YES	VERTICAL

ADDITIONAL SAMPLE CASES

Sample Input (Flexstrap Case)

21125.66				
3113400.0				
41.000000115				
81.8				
1511.0				
1711.0				
1814.0				
19125.0				
2011.0				
2411.0				
2511.0				
2611.0				
31116.0				
3311.0				
3418.0				
3750.0	0.0	3.0	1.0	.0285
4233.44	9.00000000E+14	9.00000000E+14		
451.0000086				
4611.0				
4812.0				
492=28.0	2.4			
542175.00	204.40			
59150.0				
60110.0				
6117.0				
62111.0				
63112.0				
64112.0				
6512.0				
11740.0	2.0944	4.1888		
12149.00000000E+16	9.00000000E+16	9.00000000E+16	9.00000000E+16	
13316700.0				
1351.00000134				
1471.35457				
1531.81211				
159111.22				
17926.128824	100000000.			
181220000000.0	20000000.0			
18322.789096	-.025075			
20111.0				
2085.000244	-.9324	60.75	-4.4074	.000628
2135	.000628	-.15212	-.01619	.0853
25141.0	1.0		1.0	
2585.001538	-.9556	58.925	-4.1277	.003953
2635	.003953	-.15212	-.01619	.0901
27641.846	281000.0	341000.0	2020000.0	
30141.0	1.0	1.0	1.0	
3085.001001	-.9889	56.325	-3.7291	.002573
3135	.002573	-.15212	-.01619	.0969
32642.63	301000.0	460000.0	1820000.0	

35141.0	1.0	1.0	1.0	
3585.000901	-.8727	54.275	-3.4148	.002592
3635	.002592	-.15212	-.01619	.1023
37642.074	323000.0	588000.0	16100000.0	
40141.0	1.0	1.0	1.0	
4085.001103	-.7444	51.775	-3.0315	.003967
4135	.003967	-.15212	-.01619	.1088
42642.529	343000.0	711000.0	15600000.0	
45141.0	1.0	1.0	1.0	
4585.001551	-.8101	48.15	-2.4758	.006338
4635	.006338	-.15212	-.01619	.1183
47643.667	373000.0	816000.0	16600000.0	
50141.0	1.0	1.0	1.0	
5085.001472	-.8627	44.0	-1.8396	.006488
5135	.006488	-.15212	-.01619	.1292
52644.198	411000.0	894000.0	17800000.0	
55141.0	1.0	1.0	1.0	
5585.001499	-.9245	40.0	-1.2264	.007059
5635	.007059	-.15212	-.01619	.1396
57644.047	451000.0	975000.0	19200000.0	
60141.0	1.0	1.0	1.0	
6085.001527	-.9847	36.0	-0.6132	.007629
6135	.007629	-.15212	-.01619	.1501
62644.047	490000.0	1060000.0	20500000.0	
65141.0	1.0	1.0	1.0	
6585.001554	-1.0558	32.0	-0.000	.0082
6635	.0082	-.15212	-.01619	.1606
67644.047	529000.0	1140000.0	21800000.0	
70141.0	1.0	1.0	1.0	
7085.001472	-1.1153	28.137	.5922	.008149
7135	.008149	-.15212	-.01619	.1707
72643.908	568000.0	1210000.0	23100000.0	
75141.0	1.0	1.0	1.0	
7585.001954	-1.2778	24.775	1.1076	.008158
7635	.008158	-.15212	-.01619	.1791
77643.401	603000.0	1290000.0	24200000.0	
80141.0	1.0	1.0	1.0	
8085.002596	-1.4803	22.275	1.4909	.010329
8135	.010329	-.15212	-.01619	.1860
82642.529	796000.0	1690000.0	38000000.0	
85141.0	1.0	1.0	1.0	
8585.002912	-1.4089	20.775	1.7208	.007789
8635	.007789	-.15212	-.01619	.1900
87641.518	1120000.0	2370000.0	64300000.0	
90141.0	1.0	1.0	1.0	
9085.002694	-1.19	19.9	1.855	.006852
9135	.006852	-.15212	-.01619	.1923
9264.885	4340000.0	5270000.0	97900000.0	
95211.0				
95711.0				

100141.0	1.0	1.0	1.0	
10085.007778	-1.15	17.275	2.2574	.034386
10135	.034386	-.15212	-.01619	.1991
102642.656	2240000.0	806000.0	19900000.0	
105121.0	1.0			
105421.0	1.0			
10585.001255	0.0000	14.0	2.9289	.004421
10635	.004421		-.01619	0.000
107642.929	101000.0	500000.0	14000000.0	
108912.28374				
110141.0	1.0		1.0	
11085.002	0.0000	11.381		.00066
11143.00066		-.01619		
112642.619	60000.0	45000.0	15400000.0	
115111.0				
115411.0				
11585.0012	.000	9.0		.000676
11643.000676		-.01619		
117642.381	60000.0	45000.0	18200000.0	
120111.0				
120411.0				
12085.00053		7.0		.000692
12143.000692		-.01619		
122642.0	60000.0	45000.0	20700000.0	
125111.0				
125411.0				
12581.0005				
126015.0				
12621.000896				
12641.000896				
12661-.01619				
127642.0	60000.0	140000.0	21800000.0	
130141.0	1.0		1.0	
13081.000085				
131012.671				
13121.000086				
13141.000086				
132642.329	60000.0	66500.0	22200000.0	
133111.0				
135411.0				
13581.001				
136011.342				
13623.001		.001		
137641.392	10000000.0	66500.0	29200000.0	
140411.0				
142641.342	9.00000000E+16	9.00000000E+16	9.00000000E+16	
31912.9916				
36912.2974				
41913.0926				
46912.9159				

51913.1052		
56914.0393		
61914.0393		
66913.9696		
71913.6475		
76912.96		
81912.02		
86911.1992		
91911.7672		
101913.6228		
32232.9692	8.05	603.7
37233.0340	8.2	603.7
42233.1122	8.4	603.7
47233.2247	8.65	603.7
52233.3543	9.0	603.7
57233.4821	9.25	603.7
62233.6060	9.6	603.7
67233.7319	9.86	603.7
72233.8539	10.18	603.7
77233.9587	10.4	603.7
82234.0388	10.6	603.7
87234.0846	10.72	603.7
92234.1113	10.85	603.7
102234.1952	11.0	603.7
31811.0		
36811.0		
41811.0		
46811.0		
51811.0		
56811.0		
61811.0		
66811.0		
71811.0		
76811.0		
81811.0		
86811.0		
91811.0		
101811.0		
145112.0		
2162=,02013	.1144	
2662=,02013	.1192	
3162=,02013	.126	
3662=,02013	.1314	
4162=,02013	.1379	
4662=,02013	.1474	
5162=,02013	.1583	
5662=,02013	.1687	
6162=,02013	.1792	
6662=,02013	.1897	
7162=,02013	.1998	

7662=.02013 .2082
 8162=.02013 .2151
 8662=.02013 .2191
 9162=.02013 .2214
 10162=.02013 .2282
 10661=.02013
 11161=.02013
 11661=.02013
 12161=.02013
 12661=.02013

8
 1
 1
 15
 19.0 340. =.42045519 0.08899219 =1.5350914E-03
 7.4254402E-06=8.2868165E-0999.79237 =1.202794 0.0036020154
 1.0 1.0 0.0 1.0 1.0
 10
 0.0 0.3 0.4 0.505 0.555 0.618 0.68
 0.75 0.82 1.0
 6
 0.0 12.7 20.0 340. 350. 360.
 .19 1.5 1.1 -1.0 -.85 .19
 12
 0.0 8. 11. 13. 14. 15. 20.
 340. 350. 354. 356. 360.
 .205 1.09 1.39 1.58 1.63 1.48 1.18
 -1.0 -.7 -.45 -.24 .205
 12
 0. 8. 10. 11. 11.8 13. 20.
 340. 350. 354. 356. 360.
 .22 1.15 1.36 1.45 1.5 1.43 1.07
 -1.0 -.73 -.444 -.25 .22
 12
 0. 9. 11. 12. 12.5 12.7 20.
 340. 350. 354. 356. 360.
 .25 1.365 1.565 1.64 1.65 1.51 1.2
 -1.0 -.71 -.45 -.25 .25
 13
 0. 8. 10. 11. 12. 13. 20.
 340. 350. 352. 354. 356. 360.
 .26 1.3 1.48 1.495 1.48 1.38 1.25
 -1.0 -.69 -.61 -.46 -.255 .26
 15
 0. 3. 6. 7. 8. 9. 10.
 360. 20. 340. 350. 352. 354. 356.
 .285 .7 1.1 1.19 1.24 1.27 1.33
 1.34 1.3 -1.0 -.67 -.62 -.47 -.27

.285
12

0.	3.	3.7	6.	10.	14.	20.
340.	350.	354.	355.	360.		
.305	.75	.8	.96	1.18	1.33	1.48
-1.	-.67	-.53	-.42	.305		

15

0.	2.	6.	10.	15.	20.	340.
350.	352.	353.	354.	355.	356.	359.
360.						
.36	.56	.86	1.11	1.35	1.5	-1.
-.7	-.67	-.67	-.61	-.49	-.35	.23
.36						

13

0.	1.	2.	6.	8.	15.	20.
340.	350.	354.	355.4	356.8	360.	
.16	.3	.4	.79	.97	1.4	1.56
-1.	-.68	-.44	-.19	-.33	.16	

4

0.	20.	340.	360.
0.	2.5	-2.5	0.

10

13.0	354.0	-1.0783242	0.10546602	-0.0022458297
2.5438880E-05	-1.0989072E-075	.5012552	-0.060345398	0.00016618434

7

0.0	0.3	0.4	0.6	0.7	0.8	1.0
-----	-----	-----	-----	-----	-----	-----

15

0.0	1.	2.	3.	4.	6.	8.
10.	12.	14.	354.	356.	358.	359.
360.						
.008	.0085	.009	.009	.0085	.009	.0115
.013	.017	.03	.011	.01	.0086	.0085
.008						

15

0.0	1.	2.	3.	4.	6.	8.
10.	12.	14.	354.	356.	358.	359.
360.						
.008	.0085	.009	.009	.0085	.0088	.0106
.013	.017	.03	.011	.008	.0086	.0085
.008						

15

0.0	1.	2.	3.	4.	6.	8.
10.	12.	14.	354.	356.	358.	359.
360.						
.0075	.0075	.0075	.0076	.008	.009	.01145
.0145	.057	.08	.016	.0082	.0075	.0075
.0075						

15

0.0	1.	2.	3.	4.	6.	8.
10.	12.	14.	354.	356.	358.	359.

360.						
.008	.008	.008	.0088	.0087	.021	.08
.08	.14	.18	.058	.0139	.0085	.0085
.008						
15						
0.0	1.	2.	3.	4.	6.	8.
10.	12.	14.	354.	356.	358.	359.
360.						
.0105	.0155	.03	.06	.07	.1	.13
.14	.26	.27	.08	.02	.009	.009
.0105						
15						
0.0	1.	2.	3.	4.	6.	8.
10.	12.	14.	354.	356.	358.	359.
360.						
.07	.075	.08	.09	.1	.15	.2
.3	.3	.3	.09	.06	.05	.05
.07						
15						
0.0	1.	2.	3.	4.	6.	8.
10.	12.	14.	354.	356.	358.	359.
360.						
.21	.21	.24	.25	.26	.3	.34
.35	.35	.35	.25	.2	.18	.18
.21						
22						
13.0	352.0					
0.0014700305	-2.1980119E=05	0.16285396				
-9.4724852E=07	3.7413282E=09	-9.8201326				
-2.3662806E=06	46.119316	-0.4239566				
3.2327550E=08	0.045					
				0.22140169		0.037698898
				-0.014395669		1.2695044E=04
				0.051520593		4.2809828E=04
				0.0011007744		-6.7548054E=06
6						
0.0	0.3	0.45	0.67	0.75	1.	
14						
0.	2.	4.	6.	8.	10.	11.
12.	14.	352.	354.	356.	358.	360.
-.03	-.035	-.039	-.039	-.049	-.545	-.057
-.058	-.06	-.005	-.015	-.0215	-.026	-.03
14						
0.	2.	4.	6.	8.	10.	11.
12.	14.	352.	354.	356.	358.	360.
-.03	-.035	-.039	-.039	-.049	-.0545	-.057
-.058	-.06	-.005	-.015	-.0215	-.026	-.03
14						
0.	2.	4.	6.	8.	10.	11.
12.	14.	352.	354.	356.	358.	360.
-.03	-.035	-.039	-.045	-.049	-.049	-.05
-.05	-.06	-.005	-.022	-.0215	-.026	-.03
14						
0.	2.	4.	6.	8.	10.	11.

12. =.037 =.145 14	14. =.0435 =.15	352. =.05 =.006	354. =.0915 =.024	356. =.108 =.032	358. =.09 =.032	360. =.145 =.037
0. 12. =.12 =.17 14	2. 14. =.122 =.2	4. 352. =.12 =.008	6. 354. =.14 =.006	8. 356. =.15 =.047	10. 358. =.16 =.063	11. 360. =.17 =.12
0. 12. =.09 =.5 9	2. 14. =.169 =.5	4. 352. =.233 =.094	6. 354. =.29 =.03	8. 356. =.4 0.	10. 358. =.5 =.037	11. 360. =.5 =.09

Sample Results (Flexstrap Case)

```

DTMAB =
DTLS10 =
DPIVOT =
0.2098710571489194D+02
-0.316357858000868D+05
0.330100008191044D+07

LAST DETERMINANT VALUE = 0.984584D+03
CURRENT DETERMINANT VALUE = 0.263488D+04
EIGEN VALUE CONVERGENCE = 0.124553D+04

LAST EIGEN VALUE = 0.238873D+02
CURRENT EIGEN VALUE = 0.239884D+02
NEXT EIGEN VALUE = 0.170000D+03

15 VALUES OF UPDATED LAMBDA MATRIX
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
2.0405D+02 1.3232D+02 5.7745D+02 3.6873D+03 -3.3277D+01 -2.7328D+01 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
1.6268D+02 -7.2727D+04 2.6488D+02 1.6393D+02 -6.3564D+02 -1.7489D+02 7.4766D+03 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

DTMAB =
DTLS10 =
DPIVOT =
0.899450547365066D+00
0.209871061485692D+02
-0.798919899741622D+08
-0.434992684796340D+00
-0.190247882812887D+07

LAST DETERMINANT VALUE = 0.263488D+04
CURRENT DETERMINANT VALUE = 0.127170D+04
EIGEN VALUE CONVERGENCE = 0.268324E+04

LAST EIGEN VALUE = 0.239884D+02
CURRENT EIGEN VALUE = 0.239884D+02
NEXT EIGEN VALUE = 0.170000D+03

15 VALUES OF UPDATED LAMBDA MATRIX
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
2.0403D+02 1.3228D+02 5.7757D+02 3.6859D+03 -3.3278D+01 -2.7312D+01 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
1.6268D+02 -7.2688D+04 2.6488D+02 1.6393D+02 -6.3559D+02 -1.7489D+02 7.4766D+03 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

DTMAB =
DTLS10 =
DPIVOT =
0.899450733975316D+00
0.209871060661193D+02
0.31635197133612D+12
-0.43499183368281D+00
0.427687513677767D+11

LAST DETERMINANT VALUE = 0.127170D+04
CURRENT DETERMINANT VALUE = 0.198476D+10
EIGEN VALUE CONVERGENCE = 0.528339E+10

LAST EIGEN VALUE = 0.239884D+02
CURRENT EIGEN VALUE = 0.239884D+02
NEXT EIGEN VALUE = 0.170000D+03

```


AEROELASTIC STABILITY
STATE VECTORS

BLADE 1 0 PER REV SHAPE VECTOR(LOCAL AXIS SYS.)

SECTION	UX, RADIAL DEFL. (+, OUTWARD)	UY, INPLANE DEFL. (+, ROTATION DIR.)	UZ, VERT. DEFL (+, UPWARD)
TIP			
1	2.0403E-02	1.3228E-02	-1.0000E+00
2	2.0403E-02	1.3228E-02	-9.6838E-01
3	2.0403E-02	1.3228E-02	-9.2310E-01
4	2.0403E-02	1.3228E-02	-8.8723E-01
5	2.0403E-02	1.3228E-02	-8.5337E-01
6	2.0403E-02	1.3228E-02	-8.2069E-01
7	2.0403E-02	1.3228E-02	-7.8928E-01
8	2.0403E-02	1.3228E-02	-7.5915E-01
9	2.0403E-02	1.3228E-02	-7.3028E-01
10	2.0403E-02	1.3228E-02	-7.0268E-01
11	2.0403E-02	1.3228E-02	-6.7634E-01
12	2.0403E-02	1.3228E-02	-6.5125E-01
13	2.0403E-02	1.3228E-02	-6.2741E-01
14	2.0403E-02	1.3228E-02	-6.0482E-01
15	2.0403E-02	1.3228E-02	-5.8348E-01
16	1.7888E-02	7.9586E-03	-5.6339E-01
17	2.0403E-02	1.3228E-02	-5.4452E-01
18	-9.9255E-04	-1.0322E-04	-5.2688E-01
19	-9.9255E-04	-1.0322E-04	-5.1048E-01
20	-9.9255E-04	-1.0322E-04	-4.9528E-01
21	-9.9255E-04	-1.0322E-04	-4.8128E-01
22	-9.9255E-04	-1.0322E-04	-4.6848E-01
23	6.7585E-15	2.1677E-14	-4.5688E-01
24	6.7585E-15	2.1677E-14	-4.4648E-01
25	6.7585E-15	2.1677E-14	-4.3728E-01
MUB			
1	4.7757E-02	3.4958E-03	-6.6637E-03
2	4.7757E-02	3.4958E-03	-6.7439E-03
3	4.7865E-02	3.5237E-03	-6.8501E-03
4	4.7311E-02	3.4989E-03	-6.9856E-03
5	4.7022E-02	3.5101E-03	-7.1516E-03
6	4.6878E-02	3.6685E-03	-7.3486E-03
7	4.6742E-02	3.8678E-03	-7.5788E-03
8	4.6650E-02	4.1097E-03	-7.8348E-03
9	4.6608E-02	4.3927E-03	-8.1188E-03
10	4.6608E-02	4.7327E-03	-8.4348E-03
11	4.6608E-02	5.1311E-03	-8.7848E-03
12	4.6608E-02	5.5911E-03	-9.1728E-03
13	4.6633E-02	6.1181E-03	-9.6048E-03
14	4.7099E-02	6.7181E-03	-1.0098E-02
15	4.7999E-02	7.4041E-03	-1.0678E-02
16	5.0499E-02	8.1841E-03	-1.1298E-02
17	4.7132E-02	6.0932E-03	-1.2048E-02
18	4.5310E-02	6.6232E-03	-1.2928E-02

SECTION	PHI X, TWIST (+, NOSE UP)	PHI Y, FLAPWISE SLOPE (+, FLAP DOWN)	PHI Z, CHORDWISE SLOPE (+, BLADE LEAD)
TIP			
1	1.6288E-02	-7.2688E-04	-6.6637E-03
2	1.6276E-02	-7.6238E-04	-6.7439E-03
3	1.6445E-02	-9.0278E-04	-6.8501E-03
4	1.6524E-02	-1.0780E-03	-6.9856E-03
5	1.6914E-02	-1.2867E-03	-7.1516E-03
6	1.6793E-02	-1.5266E-03	-7.3486E-03
7	1.7162E-02	-1.8090E-03	-7.5788E-03
8	1.7474E-02	-2.1369E-03	-7.8348E-03
9	1.7807E-02	-2.5169E-03	-8.1188E-03
10	2.0550E-02	-3.1506E-03	-8.4348E-03
11	2.2474E-02	-3.1506E-04	-8.7848E-03
12	2.3697E-02	9.6294E-04	-9.1728E-03
13	2.4380E-02	1.2961E-03	-9.6048E-03
14	2.4524E-02	1.3877E-03	-1.0098E-02
15	2.8500E-02	1.3946E-03	-1.0678E-02
16	1.6500E-02	3.5311E-03	-1.1298E-02
17	2.8549E-02	2.5311E-03	-1.2048E-02
18	2.1016E-02	3.1111E-03	-1.2928E-02

BEST AVAILABLE COPY

19	3.4121E+02	4.2870E+03	1.7445E+02	2.4651E+03	-1.1710E+03	-2.0798E+03
20	2.7821E+02	4.6536E+03	1.6847E+02	2.3654E+03	-2.4733E+03	-1.7971E+03
21	1.9348E+02	5.0601E+03	1.5679E+02	2.2602E+03	-2.6378E+03	-1.2706E+03
22	1.1194E+02	4.3847E+03	1.4661E+02	2.1126E+03	-5.7482E+03	-4.1510E+03
23	4.1715E+01	1.6259E+05	4.4950E+01	1.2644E+05	-1.1461E+04	-2.3735E+04
24	-1.2806E+01	2.7721E+05	1.4211E+03	-9.4437E+03	-1.4113E+04	-4.7015E+04
25	7.0958E+01	4.3341E+05	-6.0366E+04	3.1126E+05	2.9291E+05	1.0828E+06

SECTION
 (→, AXIAL FORCE
 (→, TORSION)
 V, Y, CHORDWISE SHEAR
 (→, TOWARD L.F.)
 W, Z, FLAPWISE SHEAR
 (→, UPWARD)

TIP	1	3.3570E+00	-1.6915E+00	-2.4370E+00	3.9437E+00	-7.2056E+00	-1.9073E+00
	2	3.5846E+01	-1.1847E+01	-1.7631E+01	-2.7888E+01	-2.1266E+01	-1.3581E+01
	3	4.0929E+01	-1.4106E+01	-2.0473E+01	-4.1082E+01	-2.6544E+01	-4.5963E+00
	4	4.7702E+01	-2.2331E+01	-3.4384E+01	-5.2558E+01	-4.4549E+01	1.0091E+00
	5	6.0078E+01	-2.9369E+01	-5.3601E+01	-6.5559E+01	-7.2552E+01	4.6373E+00
	6	7.6542E+01	-3.6630E+01	-8.0033E+01	-8.2802E+01	-9.2456E+01	1.1043E+01
	7	9.7824E+01	-4.4049E+01	-1.1515E+02	-1.0949E+02	-1.2244E+02	1.5635E+01
	8	1.1154E+02	-5.2380E+01	-1.5944E+02	-1.2039E+02	-1.3590E+02	2.1640E+01
	9	1.1944E+02	-5.5413E+01	-1.9444E+02	-1.3704E+02	-1.4919E+02	2.9482E+01
	10	1.2614E+02	-5.8226E+01	-2.3066E+02	-1.4543E+02	-1.6042E+02	3.1672E+01
	11	1.3266E+02	-6.0526E+01	-2.6112E+02	-1.5489E+02	-1.7682E+02	3.10220E+01
	12	1.3909E+02	-6.2793E+01	-2.8777E+02	-1.6549E+02	-1.9709E+02	2.5161E+01
	13	1.4527E+02	-6.4955E+01	-3.1034E+02	-1.7304E+02	-2.2009E+02	1.6428E+01
	14	1.5230E+02	-6.6706E+01	-3.2954E+02	-1.7304E+02	-2.3790E+02	1.3590E+01
	15	1.6515E+02	-6.8245E+01	-3.5254E+02	-1.5647E+02	-2.1324E+02	-1.6993E+01
	16	2.0460E+02	-6.1725E+01	-1.1036E+02	-1.8574E+02	-2.3771E+02	1.0492E+01
	17	1.9911E+02	-6.1909E+01	-1.804E+01	-1.7314E+02	-2.6411E+02	-3.4133E+01
	18	1.9911E+02	-6.1909E+01	-1.9344E+01	-1.7437E+02	-2.7170E+02	-3.7545E+01
	19	1.9911E+02	-6.1909E+01	-2.1050E+01	-1.7470E+02	-2.7891E+02	-3.6943E+01
	20	1.9911E+02	-6.1909E+01	-2.2074E+01	-1.7488E+02	-2.7585E+02	-3.6943E+01
	21	1.9911E+02	-6.1909E+01	-2.2066E+01	-1.7492E+02	-2.7226E+02	-3.6943E+01
	22	1.9911E+02	-6.1909E+01	-2.2067E+01	-1.7493E+02	-2.7226E+02	-3.6943E+01
	23	2.0508E+02	-6.6988E+01	-2.2067E+01	-1.7493E+02	-2.7226E+02	-3.6943E+01
	24	2.0508E+02	-6.6988E+01	-2.2067E+01	-1.7493E+02	-2.7226E+02	-3.6943E+01
	25	2.0508E+02	-6.6988E+01	-2.2067E+01	-1.7493E+02	-2.7226E+02	-3.6943E+01

SECTION
 (→, TORQUE
 (→, NUSE UP)
 M, Y, FLAPWISE MOMENT
 (→, TENSION UPPER FIBERS)
 M, Z, CHORDWISE MOMENT
 (→, TENSION T.E.)

TIP	1	7.7102E+00	2.1133E+00	-1.0444E+01	-6.3361E+01	2.9298E+00	-1.6617E+00
	2	5.6658E+01	1.5465E+01	-6.9878E+01	-2.2331E+00	1.9244E+01	-1.4527E+01
	3	7.0379E+01	3.2167E+00	-6.7868E+01	3.3584E+01	1.5491E+01	-6.0611E+01
	4	6.4302E+01	-5.7684E+00	-6.9739E+01	4.5304E+01	6.1674E+00	-1.2735E+02
	5	6.8596E+01	-1.9274E+01	-6.7613E+01	4.9069E+01	-1.3355E+01	-2.1375E+02
	6	1.3607E+02	-2.7722E+01	-6.5847E+01	3.2090E+01	-5.2183E+01	-3.6671E+02
	7	1.8314E+02	-3.7244E+01	-1.3540E+02	6.1949E+00	-1.1392E+02	-5.8423E+02
	8	2.3580E+02	-5.0571E+01	-1.8935E+02	-2.4494E+01	-1.8486E+02	-4.3088E+02
	9	2.9784E+02	-6.1881E+01	-2.7369E+02	-7.4905E+01	-2.6594E+02	-1.1081E+03
	10	3.7418E+02	-6.9922E+01	-3.9244E+02	-1.5941E+02	-3.5558E+02	-1.4164E+03
	11	4.5812E+02	-7.3956E+01	-5.6278E+02	-2.6614E+02	-4.4909E+02	-1.7415E+03
	12	5.5542E+02	-6.6672E+01	-7.1202E+02	-3.9299E+02	-5.3141E+02	-2.0470E+03
	13	6.6131E+02	-5.2170E+01	-9.0707E+02	-5.0999E+02	-5.9083E+02	-2.2911E+03
	14	7.4835E+02	-3.4578E+01	-1.1727E+03	-5.8547E+02	-6.2163E+02	-2.5441E+03
	15	8.1190E+02	-2.3715E+01	-1.2900E+03	-6.2823E+02	-6.4144E+02	-2.5847E+03
	16	7.4410E+02	3.9210E+01	-6.6003E+02	-4.2568E+00	-1.0825E+03	-2.6586E+03

17	1.0511E+03	2.6926E+01	-1.5667E+03	-7.5719E+02	-8.6537E+02	-2.9793E+03
18	3.437E+02	3.6396E+01	1.0605E+02	2.1292E+01	-1.4503E+03	-3.1461E+03
19	3.2102E+02	3.6243E+01	2.3560E+01	1.4863E+00	-1.6432E+03	-3.1403E+03
20	3.0499E+02	3.4679E+01	1.6844E+01	3.3388E+02	-1.8231E+03	-3.8492E+03
21	2.9224E+02	3.3002E+01	4.2479E+01	3.7803E+00	-1.9787E+03	-4.1582E+03
22	2.8031E+02	3.1544E+01	1.0446E+02	1.2043E+01	-2.1432E+03	-4.4718E+03
23	3.1202E+02	1.1918E+02	2.4900E+02	4.1747E+01	-2.3410E+03	-4.8992E+03
24	3.0390E+02	1.1918E+02	5.5102E+02	8.3614E+01	-2.4679E+03	-5.0992E+03
25	3.0390E+02	1.1918E+02	9.2333E+02	1.3921E+02	-2.5910E+03	-5.3346E+03

MUB

Sample Input (Articulated Case)

0002140.0				
001511.0				
001721.0	4.0			
001917.0				
002111.0				
002221.0	1.0			
002816.0				
002924.0	3.0			
003711.0				
003923.0	1.0			
004140.1	0.9	.10	E+11.10	E+11
004611.0				
004811.0				
0054141.7228				
0059350.0	10.0	7.0		
006236.0	7.0	7.0		
00771.1	E=06			
00971=.5				
011730.0	2.0943951	4.1887902		
01213.9	E+17.9	E+17.9	E+17	
01331.6	E+05			
016921.0	5.0			
020111.0				
020830.25	0.005	10.0		
021210.5				
021410.5				
02153=.015	-.02	0.05		
025411.0				
02653=.015	-.02	0.05		
027649.1	.1	E+09.1	E+09.1	E+09
030211.0				
03152=.015	-.02			
035111.0				
03652=.015	-.02			
040211.0				
04161=.02				
045211.0				
50411.0				
05264.9	.1	E+12.1	E+12.1	E+12
8				
9				

Sample Results (Articulated Case)

CURRENT DETERMINANT VALUE = 0.9422859001E+03
 EIGEN VALUE CONVERGENCE = 0.504337E-08

LAST EIGEN VALUE = 0.0
 CURRENT EIGEN VALUE = 0.000370E+03
 NEXT EIGEN VALUE = 0.000000E+03

11 VALUES OF UPDATED LAMBDA MATRIX

-1.7700E-15 6.3320E-17 2.2541E-07 1.3710E-08 2.4001E-09 1.1045E-06 -4.0550E-02 4.0500E-02 1.0500E-02 -2.2440E-03 4.5010E-03
 1.0500E-08 0.0 1.0000E-01 4.5000E-01 -2.0700E-08 1.0700E-03 -4.5100E-04 1.0000E-01 1.0000E-01 0.0000E+00 0.0000E+00
 1.1000E-04 4.0000E-03 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00

DTM48 = 0.00000712714400E+00
 DTLS10 = 0.25076720120000E+02
 DP1VDT = 0.1207967201114100E-07

LAST DETERMINANT VALUE = 0.9422859001E+03
 CURRENT DETERMINANT VALUE = 0.9422859001E+03
 EIGEN VALUE CONVERGENCE = 0.504337E-08

LAST EIGEN VALUE = 0.0
 CURRENT EIGEN VALUE = 0.000370E+03
 NEXT EIGEN VALUE = 0.000000E+03

BEST AVAILABLE COPY

AEROELASTIC STABILITY
STATE VECTORS

SHASHPLEATE DEFLECTIONS

0 PER REV.

W(0) = 0.17980 = 14 0.853230 = 16

BLADE 1 0 PER REV SHAPE VECTOR (LOCAL AXIS SYS.)

SECTION	UX, RADIAL DEFL. (+, OUTBOARD)	UY, INPLANE DEFL. (+, ROTATION DIR.)	UZ, VERT. DEFL. (+, UPWARD)
TIP	-2.5499E-10 -2.8590E-11	-4.8654E-02 4.0969E-02	-1.0000E+00 0.0
2	-2.5499E-10 -2.8590E-11	-6.3348E-10 -2.7324E-11	-7.1726E-08 -8.4786E-09
3	-2.5500E-10 -2.8590E-11	2.9547E-09 3.9644E-10	-7.1726E-08 -8.4604E-09
4	-2.5500E-10 -2.8590E-11	2.9547E-09 3.9644E-10	-7.1726E-08 -8.4604E-09
5	-2.1082E-10 -2.2608E-11	2.9348E-09 3.9431E-10	-7.1726E-08 -8.4604E-09
6	1.2237E-09 1.4672E-10	2.9348E-09 3.9431E-10	-7.1726E-08 -8.4604E-09
HUB	1.2237E-09 1.4672E-10	3.2256E-09 3.8674E-10	-6.4807E-08 -7.7788E-09

SECTION	PHI X, TWIST (+, NOSE UP)	PHI Y, FLAPWISE SLOPE (+, FLAP DOWN)	PHI Z, CHORDWISE SLOPE (+, BLADE LEAD)
TIP	2.4041E-05 -1.1046E-06	1.0989E-01 1.8204E-09	-5.3467E-03 4.3021E-03
2	2.4022E-05 -1.1519E-06	1.0989E-01 -6.4361E-09	-5.3467E-03 4.3019E-03
3	2.4022E-05 -1.1519E-06	1.1002E-01 -2.2501E-08	1.5239E-08 4.4963E-03
4	-1.6501E-03 3.3546E-06	1.1002E-01 -2.2501E-08	1.5239E-08 4.4963E-03
5	-9.7182E-11 -1.0244E-11	1.1003E-01 -2.2368E-08	-3.1719E-10 3.2273E-11
6	-9.0819E-11 -1.0889E-11	8.2712E-09 7.7432E-10	-3.1907E-10 3.2062E-11
HUB	-9.0819E-11 -1.0889E-11	6.4878E-09 7.7788E-10	-3.2463E-10 -3.8023E-11

SECTION	N, AXIAL FORCE (+, RADIAL OUTBOARD)	V Y, CHORDWISE SHEAR (+, TOWARD L.E.)	V Z, FLAPWISE SHEAR (+, UPWARD)
TIP	-2.6290E+01 1.1266E+00	-2.0878E+01 1.7567E+01	-4.3940E+02 -1.9955E+02
2	-2.6290E+01 1.1266E+00	-2.0878E+01 1.7567E+01	-4.3940E+02 -1.9955E+02
3	-2.6290E+01 1.1266E+00	1.1106E+00 1.7544E+01	-4.3940E+02 -1.9955E+02
4	-2.6290E+01 1.1266E+00	1.1106E+00 1.7544E+01	-4.3940E+02 -1.9955E+02
5	-2.6272E+01 1.3896E+00	1.3727E+00 1.7527E+01	-4.3940E+02 -1.9955E+02
6	-1.7469E+01 1.3723E+00	1.3727E+00 1.7527E+01	-4.3940E+02 -1.9955E+02
HUB	-1.7469E+01 1.3723E+00	1.3727E+00 1.7527E+01	-4.3940E+02 -1.9955E+02

SECTION	T, TORQUE (+, NOSE UP)	M Y, FLAPWISE MOMENT (+, TENSION UPPER FIBERS)	M Z, CHORDWISE MOMENT (+, TENSION T.E.)
TIP	-1.9020E+00 5.2003E-01	2.1611E-01 -1.8160E-01	-4.5797E+00 3.9633E+00
2	6.0055E+01 -2.8798E+00	2.3613E-04 2.6314E-05	-2.2348E+07 1.0718E+08

3	-3.420E-06	-4.1010E-07	2.358E-04	2.827E-05	1.1802E-05	1.4151E-04
4	-3.420E-06	-4.1010E-07	2.358E-04	2.827E-05	1.1802E-05	1.4151E-04
5	1.1866E-07	1.4011E-06	2.358E-04	2.827E-05	1.1731E-05	1.4066E-04
6	-1.178E-07	-1.4121E-06	2.358E-04	2.827E-05	1.1731E-05	1.4066E-04
7	-1.039E-07	-1.2727E-06	3.9630E-02	-7.9233E-01	1.235E+00	1.977E+01

MUB

Sample Input (Gimballed Case)

0002140.0				
001511.0				
001721.0	4.0			
001916.0				
002221.0	1.0			
002924.0	3.0			
003211.0				
003711.0				
003923.0	1.0			
004140.1	0.9	.10	E+11.10	E+11
004611.0				
004811.0				
00541255.2501				
0059350.0	10.0	7.0		
006236.0	7.0	7.0		
00771.1	E=06			
00971-.5				
011730.0	2.0943951	4.1887902		
01213.9	E+17.9	E+17.9	E+17	
01331.6	E+05			
020111.0				
020830.25	0.005	10.0		
021210.5				
021410.5				
02153-.015	-.02	0.05		
025411.0				
02653-.015	-.02	0.05		
027649.1	.1	E+07.4	E+07.4	E+07
030211.0				
03152-.015	-.02			
035111.0				
03652-.015	-.02			
040211.0				
045411.0				
04764.9	.1	E+12.1	E+12.1	E+12
8				
9				

Sample Results (Gimballed Case)

```

LAST EIGEN VALUE = 0.0          0.2565260E+03
CURRENT EIGEN VALUE = 0.513050E-03  0.2565260E+03
NEXT EIGEN VALUE = 0.000000E+00  0.2565260E+03

          9 VALUES OF UPDATED LAMBDA MATRIX
-5.40910E-04 -5.13280E-05  1.00810E-07  1.67990E-09  6.28910E-08  1.65300E-04 -1.51870E-02  7.81020E-02  1.31010E-02
 1.00000E+00  0.0          1.00020E-01 -6.85010E-08  3.67180E+00  3.44850E-01  1.10130E-03  9.37630E-05
DIPHAS 0          -0.0000003205622830+00          0.0000002710981592280E-02
DILB10 0          0.4090959121034890E+01          0.000000000000000000
DPIVOT 0          0.22911427740032970E-08          -0.003233601673212020E-07

          9 VALUES OF UPDATED LAMBDA MATRIX
LAST DETERMINANT VALUE = 0.128090E+01=0.1474000E+01
CURRENT DETERMINANT VALUE = 0.1481480E-03  0.8851330E-02
EIGEN VALUE CONVERGENCE = 0.216080E-04
LAST EIGEN VALUE = 0.513050E-03  0.2565260E+03
CURRENT EIGEN VALUE = 0.4363940E-03  0.2565260E+03
NEXT EIGEN VALUE = 0.4373100E-03  0.2565260E+03

          9 VALUES OF UPDATED LAMBDA MATRIX
-5.40970E-04 -5.14320E-05 -1.37290E-08 -6.94280E-10  6.26690E-08  1.65550E-04 -1.51500E-02  7.81570E-02  1.31100E-02
 1.00000E+00  0.0          1.00610E-01 -1.40070E-07  3.67100E+00  3.44750E-01  1.10110E-03  1.33080E-04
DIPHAS 0          -0.00000032128612970+00          0.00001002303256300E-02
DILB10 0          0.4090961815866810E+01          0.000000000000000000
DPIVOT 0          0.21372100026413480E-11          -0.93395373556000510E-10

          9 VALUES OF UPDATED LAMBDA MATRIX
LAST DETERMINANT VALUE = 0.1481480E-03  0.8851330E-02
CURRENT DETERMINANT VALUE = 0.2503880E-06  6.9255520E-05
EIGEN VALUE CONVERGENCE = 0.226080E-09
LAST EIGEN VALUE = 0.4363940E-03  0.2565260E+03
CURRENT EIGEN VALUE = 0.4373100E-03  0.2565260E+03
NEXT EIGEN VALUE = 0.4373110E-03  0.2565260E+03

```

AEROELASTIC STABILITY
STATE VECTORS

SHASHPATE DEFLECTIONS

0 PER REV.

W(0)=0.54979D-03=0.51631D-04

BLADE 1 0 PER REV SHAPE VECTOR(LOCAL AXIS SYS.)

SECTION	UX, RADIAL DEFL. (+, OUTBOARD)	UY, INPLANE DEFL. (+, ROTATION DIR.)	UZ, VERT. DEFL. (+, UPWARD)
TIP	-1.2926E-08	-1.5198E-02	-1.0000E+00
2	-1.2926E-08	-6.3507E-09	-6.3036E-07
3	-1.2926E-08	2.2166E-08	-6.3004E-07
4	-1.2926E-08	2.2166E-08	-6.3004E-07
5	0.5041E-12	2.2166E-08	-6.3017E-07
6	0.5041E-12	2.2682E-11	-2.5571E+10
MUB	1.7137E-13	4.5279E-13	-6.0074E-12

SECTION	PHI X, TWIST (+, NOSE UP)	PHI Y, FLAPWISE SLOPE (+, FLAP DOWN)	PHI Z, CHORDWISE SLOPE (+, BLADE LEAD)
TIP	0.2469E-04	1.6461E-01	-2.5415E-03
2	1.1011E-03	1.3780E-04	-2.0450E-08
3	1.1011E-03	1.3773E-04	-2.0450E-08
4	-9.5044E-10	1.3773E-04	4.8830E-08
5	1.0740E-08	1.3772E-04	4.8821E-08
6	0.3289E-13	-1.2897E-14	-2.3371E-12
MUB	-6.0711E-11	6.6612E-13	-6.6635E-14

SECTION	N, AXIAL FORCE (+, RADIAL OUTBOARD)	V Y, CHORDWISE SHEAR (+, TOWARD L.E.)	V Z, FLAPWISE SHEAR (+, UPWARD)
TIP	-3.9228E+02	-2.3500E+02	-1.6851E+04
2	-3.9228E+02	5.0499E+02	-1.6451E+04
3	-3.9228E+02	5.0499E+02	-1.6442E+04
4	-3.9228E+02	5.0748E+02	-1.6442E+04
5	-5.4573E+01	5.0748E+02	-1.6467E+04
6	-5.4573E+01	5.0748E+02	-1.6467E+04
MUB	1.7761E+02	1.2148E+03	6.6212E+01

SECTION	T, TORQUE (+, NOSE UP)	M Y, FLAPWISE MOMENT (+, TENSION UPPER FIBERS)	M Z, CHORDWISE MOMENT (+, TENSION T.E.)
TIP	-5.5507E+01	1.1490E+01	-8.1776E+01
2	1.6711E+00	1.4571E+05	-2.1170E+03
3	-2.3199E-08	1.4561E+05	2.1170E+03
4	-2.3199E-08	1.4561E+05	2.1170E+03
5	2.0821E+03	1.4562E+05	5.1160E+03
6	2.0821E+03	1.6042E+05	5.6667E+03
MUB	-2.3192E+02	-6.1294E+02	1.2247E+04

Sample Input (Teetering Case)

0002140.0				
001511.0				
001721.0	2.0			
001916.0				
002221.0	1.0			
002924.0	3.0			
003212.0				
003711.0				
003923.0	1.0			
004140.1	0.9	.10	E+11.10	E+11
004611.0				
004811.0				
00541255.2501				
0059350.0	10.0	7.0		
006236.0	7.0	7.0		
00771.1	E=06			
00971-.5				
011730.0	2.0943951	4.1887902		
01213.9	E+17.9	E+17.9	E+17	
01331.6	E+05			
020111.0				
020830.25	0.005	10.0		
021210.5				
021410.5				
02153-.015	-.02	0.05		
025411.0				
02653-.015	-.02	0.05		
027649.1	.1	E+07.4	E+07.4	E+07
030211.0				
03152-.015	-.02			
035111.0				
03652-.015	-.02			
040211.0				
045411.0				
C4764.9	.1	E+12.1	E+12.1	E+12
8				
9				

Sample Results (Teetering Case)

```

          LAST EIGEN VALUE = 0.0      0.256579D+03
          CURRENT EIGEN VALUE = 0.256579D+03  0.256580D+03
          NEXT EIGEN VALUE = 0.484910D+03  0.256579D+03

          9 VALUES OF UPDATED LAMBDA MATRIX

-5.9637D+04 -5.5859D+05  1.2261D-07 -2.5100D-11  7.5840D+04  1.7774D+04 -1.5187D+02  7.6975D+02  1.3096D+02
 1.0000D+00  0.0      1.6462D-01 -1.0733D-07  7.9630D+00  7.4582D-01  1.1958D+03  1.1200D-04 -2.5466D+03  1.1200D+02

DPMAS =      -0.999917465312723D+00      -0.40628466710946821D+02
DTLS10 =      0.459087466633361D+01      -0.459726102716381D+07
DPIVOT =      0.593660663468584D+09

          LAST DETERMINANT VALUE = 0.504330D+02  0.343838D+00
          CURRENT DETERMINANT VALUE = 0.358234D+08  0.288836D+02
          EIGEN VALUE CONVERGENCE = 0.281933E+06

          LAST EIGEN VALUE = 0.256579D+03  0.256580D+03
          CURRENT EIGEN VALUE = 0.484910D+03  0.256579D+03
          NEXT EIGEN VALUE = 0.485860D+03  0.256579D+03

          9 VALUES OF UPDATED LAMBDA MATRIX

-5.9622D+04 -5.5982D+05  1.2261D-07 -2.5100D-11  7.5831D+04  1.7808D+04 -1.5157D+02  7.6148D+02  1.3104D+02
 1.0000D+00  0.0      1.6461D-01 -1.0733D-07  7.9620D+00  7.4788D-01  1.1958D+03  1.1226D-04 -2.5461D+03  1.1226D+02

DPMAS =      -0.999917461078791D+00      -0.40628466180402143D+02
DTLS10 =      0.459090254261147D+01      -0.460311079602833D+10
DPIVOT =      0.1039257484014397D+11

          LAST DETERMINANT VALUE = 0.555616D+04  0.288834D+02
          CURRENT DETERMINANT VALUE = 0.564714D+07  0.223170D+05
          EIGEN VALUE CONVERGENCE = 0.237053E+09

          LAST EIGEN VALUE = 0.484910D+03  0.256579D+03
          CURRENT EIGEN VALUE = 0.485860D+03  0.256579D+03
          NEXT EIGEN VALUE = 0.485862D+03  0.256579D+03
    
```

AEROELASTIC STABILITY
STATE VECTOR

SHASHPATE DEFLECTIONS

0 PER REV.

M(0)=0.59822D-03=0.55981D-04

BLADE 1 0 PER REV SHAPE VECTOR(LOCAL AXIS SYS.)

SECTION	UX, RADIAL DEFL. (+, OUTBOARD)	UY, INPLANE DEFL. (+, ROTATION DIR.)	UZ, VERT. DEFL. (+, UPWARD)
TIP	1 -1.2926E+00 -6.7359E-10	-1.5157E+02 7.617E+02	-1.0000E+00 0.0
	2 -1.2926E+00 -6.7359E-10	-9.3504E+09 4.617E+08	-6.3016E+07 4.4930E-12
	3 -1.2926E+00 -6.7359E-10	2.2156E+08 4.6117E+08	-6.2984E+07 2.4122E+09
	4 -1.2926E+00 -6.7359E-10	2.2156E+08 4.6117E+08	-6.2984E+07 2.4122E+09
	5 4.4961E-12 9.5492E-14	2.2156E+08 4.6122E+08	-6.2997E+07 2.3986E+09
HUB	6 4.4961E-12 9.5492E-14	1.1880E-11 2.2232E-13	-2.3869E+10 -5.0699E-12

SECTION	PHI X, TWIST (+, NOSE UP)	PHI Y, FLAPWISE SLOPE (+, FLAP DOWN)	PHI Z, CHORDWISE SLOPE (+, BLADE LEAD)
TIP	1 7.5931E-04 1.768E-04	1.6641E+01 -1.4798E+07	-2.5614E+03 1.3108E+02
	2 1.1956E+03 1.1226E+04	1.3780E+06 -1.5649E-11	-2.0450E+08 1.0545E+07
	3 1.1956E+03 1.1226E+04	1.3773E+06 -5.2859E+09	4.6452E-08 1.0532E-07
	4 -1.5205E-10 1.0192E-10	1.3773E+06 -5.2859E+09	4.6452E-08 1.0532E-07
	5 1.6734E+08 -2.0835E+09	1.3772E+04 -5.2859E+09	4.6423E+08 1.0532E+07
HUB	6 -3.3082E-13 -7.1876E-15	2.4466E-11 5.1962E-13	-1.2241E+12 -2.5998E+14

SECTION	M, AXIAL FORCE (+, RADIAL OUTBOARD)	V Y, CHORDWISE SHEAR (+, TOWARD L.E.)	V Z, FLAPWISE SHEAR (+, UPWARD)
TIP	1 -3.9223E+02 1.7761E+02	-2.3589E+02 1.2147E+03	-1.6531E+04 -4.2952E+02
	2 -3.9223E+02 1.7761E+02	-2.3589E+02 1.2147E+03	-1.6431E+04 -4.2952E+02
	3 -3.9223E+02 1.7761E+02	5.6661E+02 1.2131E+03	-1.6662E+04 6.0669E+01
	4 -3.9223E+02 1.7761E+02	5.6661E+02 1.2131E+03	-1.6662E+04 6.0669E+01
	5 -3.6525E+01 1.9454E+02	5.6749E+02 1.2104E+03	-1.6447E+04 6.4205E+01
HUB	6 -3.6525E+01 1.9454E+02	5.6749E+02 1.2104E+03	-1.6447E+04 6.4205E+01

SECTION	T, TORQUE (+, NOSE UP)	M Y, FLAPWISE MOMENT (+, TENSION UPPER FIBERS)	M Z, CHORDWISE MOMENT (+, TENSION T.E.)
TIP	1 -5.1216E+01 7.2325E+00	1.1704E+01 -5.2524E-01	-8.1772E+01 4.3099E+02
	2 7.9621E+00 7.4733E-01	1.4571E+03 -1.3448E-01	-2.1678E+03 1.1172E+06
	3 -1.2568E+08 -2.6692E-10	1.4563E+03 -5.3650E+02	5.1171E+03 1.1198E+04
	4 -1.2568E+08 -2.6692E-10	1.4563E+03 -5.3650E+02	5.1171E+03 1.1198E+04
	5 2.0821E+03 -2.3150E+02	1.6042E+05 -5.5509E+02	5.1161E+03 1.1156E+04
HUB	6 2.0821E+03 -2.3150E+02	1.6042E+05 -6.1287E+02	5.6489E+03 1.2245E+06

Sample Input (Rigid Case)

0002140.0				
001511.0				
001711.0				
001914.0				
002511.0				
004611.0				
004811.0				
00541255.2501				
0059350.0	10.0	7.0		
006235.0	6.0	6.0		
020111.0				
020830.25	0.005	10.0		
021210.5				
021410.5				
02153-.015	-.02	0.05		
025411.0				
02653-.015	-.02	0.05		
027649.1	.1	E+07.4	E+07.4	E+07
030211.0				
035411.0				
03764.9	.1	E+12.1	E+12.1	E+12
8				
9				

Sample Results (Rigid Case)

```

NEXT EIGEN VALUE =0.337005D+03 0.256578D+03

      * VALUES OF UPDATED LAMBDA MATRIX
1.0428D+07 -3.2759D+10 -9.3102D+04 1.9491D+05 -1.5192D+02 7.8204D+02 -2.5674D+03 1.3117D+02 1.0000D+00 0.0
1.6461D+01 -1.3739D+07

DTPH48 = 0.69992550818830D+00 0.3659834289641547D+02
DTL610 = -0.2310402174499107D+00 -0.260054590626962D+10
DPIVOT = 0.343264332896392D+07 0.3228257425752088D+09

      LAST DETERMINANT VALUE =0.382849D+05=0.261047D+07
      CURRENT DETERMINANT VALUE = 0.201637D+07 0.267469D+09
      EIGEN VALUE CONVERGENCE = 0.166127E+06

      LAST EIGEN VALUE =0.392152D+03 0.256586D+03
      CURRENT EIGEN VALUE =0.337005D+03 0.256578D+03
      NEXT EIGEN VALUE =0.337567D+03 0.256578D+03

      * VALUES OF UPDATED LAMBDA MATRIX
-1.3243D+08 -6.8044D+10 -9.3100D+04 1.9482D+05 -1.5187D+02 7.8266D+02 -2.5429D+03 1.3128D+02 1.0000D+00 0.0
1.6461D+01 -1.5014D+07

DTPH48 = 0.69992550818830D+00 0.3659834289641547D+02
DTL610 = -0.2310393753560832D+00 -0.260054590626962D+10
DPIVOT = 0.260054590626962D+10 0.4502671496829473D+12

      LAST DETERMINANT VALUE = 0.201637D+07 0.267469D+09
      CURRENT DETERMINANT VALUE = 0.164503D+10 0.326002D+12
      EIGEN VALUE CONVERGENCE = 0.135659E+09

      LAST EIGEN VALUE =0.337005D+03 0.256578D+03
      CURRENT EIGEN VALUE =0.337567D+03 0.256578D+03
      NEXT EIGEN VALUE =0.337568D+03 0.256578D+03
    
```

AEROELASTIC STABILITY
STATE VECTORS

BLADE 1 0 PER REV SHAPE VECTOR(LOCAL AXIS SYS.)

SECTION	UX, RADIAL DEPL. (+, OUTBOARD)	UY, INPLANE DEPL. (+, ROTATION DIR.)	UZ, VERT. DEPL. (+, UPWARD)
TIP	-1.2924E-08	-1.5167E-02	-1.0000E+00
1	-1.2924E-08	-9.3592E-09	-2.2903E+07
2	5.2766E-20	2.2135E-08	-9.2742E+07
3	5.2766E-20	4.8479E-18	-7.4859E-17
MUB	-5.2766E-20	-2.0575E-17	-6.5602E-19
SECTION	PHI X, TWIST (+, NOSE UP)	PHI Y, FLAPWISE SLOPE (+, FLAP DOWN)	PHI Z, CHORDWISE SLOPE (+, BLADE LEAD)
TIP	-9.3109E-06	1.6461E-01	-2.5429E+03
1	-1.3488E-09	1.3785E-06	-2.0457E+08
2	1.0322E-08	-1.5673E-11	5.8403E+08
3	5.0488E-20	2.8004E-11	-5.5054E+21
MUB	-3.9567E-20	4.5027E-13	5.9719E-19
SECTION	M, AXIAL FORCE (+, RADIAL OUTBOARD)	V Y, CHORDWISE SHEAR (+, TOWARD L.C.)	V Z, FLAPWISE SHEAR (+, UPWARD)
TIP	-3.9286E+02	-2.3605E+02	-1.6451E+08
1	-3.9286E+02	1.7757E+02	-4.3517E+02
2	-9.5133E+01	1.7757E+02	-1.6451E+08
3	-9.5133E+01	1.9452E+02	-1.6447E+04
MUB	-9.5133E+01	1.9452E+02	6.4302E+01
SECTION	Y, TORQUE (+, NOSE UP)	M Y, FLAPWISE MOMENT (+, TENSION UPPER FIBERS)	M Z, CHORDWISE MOMENT (+, TENSION T.E.)
TIP	-1.9567E+02	8.4693E-02	-8.1769E+01
1	-4.6288E+01	1.4571E+05	-2.1692E+03
2	2.0358E+03	1.4562E+05	5.1138E+03
3	2.0358E+03	-5.3621E+02	1.1172E+04
MUB	-2.0358E+03	-6.1376E+02	1.2268E+04

PROGRAM LISTING

<u>Program Components</u>	<u>Page</u>
Main Program	290
ACOEFF	327
AERO	322
BAERO	316
BARRAY	333
BEND	367
BLARO	372
BMASS	363
CDOT	410
DCMAT	391
ELAST	369
EPSOLN	363
EXCHI	404
EXPON	393
FAERO	320
FMASS	365
FUARO	371
LOADIN	302
MLCC2	360
MLRC2	359
POLAR	409
RCDOT	411
RIGID	368
ROWSUM	412
SARRAY	352
SECPAR	303
SETUP	295
SOLVE	388
STIFF	366
SUBMA	374
SUBMB	376
SUBMF	387
SUBMG	383
SWA	394
SWAPS	390
SWB	396
TABLU	325
TKNS	361
ULN	403
XNLQ	401
ZLN	399
ZTEGI	405
Overlay Structure	413

```

C      MAINLINE PROGRAM FOR HELICOPTER AEROELASTIC STABILITY
C      ANALYSIS
      REAL*8 DFAC,FACL,FACC,DIFF,CX,PFC,PLC
      REAL*8 ARRAY(3)
      DATA BYES/'YES '/,BNO/' NO '/
      DATA ARRAY/'INPLANE','VERTICAL','TORSION '/
      COMPLEX EGNS(5),START,EC(25),EC2(25)
      COMPLEX QXJ,QYJ,QZJ
      COMPLEX*16 CMS,CM1,CS1,CS2
      COMPLEX*16 EPS(63),ALM(63)
      COMPLEX*16 DUMMC
      COMPLEX*16 AMA(2550)
      COMPLEX*16 EGNN,EGNL,REML,EGNC,REMC,DETSV,FACT
      COMPLEX*16 EGNSL(5),ENGCH
      DIMENSION CONV(5),PENOM(5)
      DIMENSION NBIGMD(5)
      DIMENSION Y(100),SD(4000),CX(75)
      DIMENSION DETR(25),DETR2(25),APA(240)
      COMMON/RNAM/CS(4,6),SN(4,6)
      COMMON/RNAM1/CSA(4,24),SNA(4,24)
      COMMON/STAB/CPST(24,6),SPST(24,6)
      COMMON/ROTF/OM1,OM2,OMT
      COMMON/CALM/ALM
      COMMON/EPST/EPST
      COMMON/AMAT/AMA
      COMMON/FUSA/FAA
      COMMON/SUB/Y
      COMMON/SAIN/SD
      COMMON/FREF/CMS,CM1,CS1,CS2
      COMMON/INTER/NSY,NBSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
1MFLEX,MCON,MAER,MFUS,NBC,NFLAP,NPEA,NCT,NCON,NFFB,
2NAS,MHC,NVI,NSP,MAXN,NES,MSC,NEGN,IPCT,NIT,MER,NORM,
3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NCSS,
4NFP1,MXSMI,MXT2P1,MXKQ,MXCPL,MXCSB,MXCPH,MXCPK,MXSHB,
5NEBC,NEBSC,MFASB,MXPAB,NFUS,NRBD,NRIFC,MXQ,NEIFC,
6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MODE
      COMMON/PLOAD/STOR,OM,SOVEL,AIRD,FVEL,CHI,ZETA,THC,AFO,APD,COLL,AG,
1GT,GP,APD,ASK,BN,BPN,BNS,FNS,FLAPH,FEAM,CTM,CONM,APLEX,AERM,
2AFFB,FLAN,FEN,CTN,CON,BCT,FUSH,ASN,CNH,VIN,SPC,SPH,ESN,SCC,WMA,
4BRA,SHTS,SHBS,TPX,PRS,RT,EGNM,SR(5),SI(5),PCTI,ANIT,ERM,ANOR,AREM,
5ANEX,PNS,SCHN,AIG,AIF,STG,STP,PHIM(6),AKC(6),ATAU(6),TAK(4),TAC(4)
6,SMA(6),DBS(6),PAK(4),PAC(4),ECHI(4),AKE(4),ACE(4),BJE(4),CAKE,
7CACE,STKC(6),BTAU(6),PHIC(6),THEC(6),BL12(6),SDCO,PUNI,
8CYFA,CYLA,CELM,CELN,DACO,CTYM,AEXT,EYEX,EZEX,COE1,COE2,COE3,
9PHOTT(8),PHWTT(8),CLESP(6),P(2000),V(25,24)
      COMMON/DAMCO/OMDA
      COMMON/ISUMA/NBGMD
      COMMON/NLEAD/MLEL,NLEL,MCTY
      COMMON/QVTEM/QXJ,QYJ,QZJ
      COMMON/CPARS/ALSTH(6)

```

```

COMMON/TAER/THCK,THC1,THC2
COMMON/SPAR/AKCI(6),TAU(6),SMLA(6),DMS(6),AK(4),AC(4),BJ(4),
1CAPK,CAPC
COMMON/CPAR/AMKC(6),CTAU(6),AL(6,6),AL12(6)
COMMON/SWASH/SWGJ,SWEI,SWM,SWR
COMMON/SPAR1/AKT(4),ACT(4),AKP(4),ACP(4)
COMMON/CFLEX/CX
COMMON/SHAFT/TORFLX
COMMON/COUP/PFC,PLC
COMMON/DTERM/DFAC
COMMON/REMA/DETSV
COMMON/IPCH/IPU
COMMON/FGRAV/FGRA
ICNT=0
CM1=DCMPLX(0.00,1.00)
1931 CONTINUE
CALL SETUP(EGNS,ICNT)
IF(NSCH.LT.1) GO TO 50
IB=0
IC=0
ID=0
IE=0
MODE=0
START=EGNS(1)
CRS=REAL(START)
CIS=AIMAG(START)
DO 20 I=1,IF
CI=CIS+(I-1)*STF
DO 10 J=1,IG
CR=CRS+(J-1)*STG
CMS=CR+CI*CM1
DO 6 K=1,MXG
6 ALM(K)=DCMPLX(1.00,0.00)
CALL BARRAY
CALL SARRAY
CALL EPSOLN
CALL SOLVE
PRINT 943, CMS,DETSV
RESR1=DETSV
DUMMC=DETSV*DCMPLX(0.00,-1.00)
RESI1=DUMMC
IF(J.GT.1) GO TO 3
2 RESR2=RESR1
RESI2=RESI1
GO TO 10
3 IF((RESI1+RESI2).GT..0) GO TO 2
IF(RESI2.EQ.0.0) GO TO 2
R1=ABS(RESI1)
R2=ABS(RESI2)
CR=CR+STG*R1/(R1+R2)

```

```

RESR=RESR2+(RESR1-RESR2)*R2/(R1+R2)
IF(I,EQ,IR) GO TO 5
IB=I
IC=IC+1
EC(IC)=CR+CI*CM1
DETR(IC)=RESR
GO TO 2
5 ID=ID+1
EC2(ID)=CR+CI*CM1
DETR2(ID)=RESR
GO TO 20
10 CONTINUE
20 CONTINUE
IF(IC,LT,2) GO TO 28
DO 23 K=2,IC
CROSS=DETR(K-1)*DETR(K)
IF(CROSS,GT,0.) GO TO 23
IE=IE+1
R1=ABS(DETR(K-1))
R2=ABS(DETR(K))
EGNS(IE)=EC(K-1)+(EC(K)-EC(K-1))*R1/(R1+R2)
23 CONTINUE
IF(ID,LT,2) GO TO 27
DO 26 K=2,ID
CROSS=DETR2(K-1)*DETR2(K)
IF(CROSS,GT,0.) GO TO 26
IE=IE+1
R1=ABS(DETR2(K-1))
R2=ABS(DETR2(K))
EGNS(IE)=EC2(K-1)+(EC2(K)-EC2(K-1))*R1/(R1+R2)
26 CONTINUE
27 NOGNS=IE
GO TO 29
28 NOGNS=0
29 CONTINUE
IF(NOGNS,EQ,0) GO TO 40
GO TO 30
40 WRITE(6,944)
GO TO 1931
50 NOGNS=NEGN
30 DO 99 L=1,NOGNS
CONV(L)=2.0
DO 115 K=1,MAXQ
EPS(K)=DCMPLX(0.00,0.00)
115 ALM(K)=DCMPLX(1.00,0.00)
MODE=0
ITI=0
74 DO 105 K=1,MAXQ
ALM(K)=ALM(K)+EPS(K)
CMMM=CDAHS(ALM(K))

```

```

        IF(CMMM .GT. 0.1E-25) GO TO 105
        ALM(K)=DCMPLX(0.00,0.00)
105  CONTINUE
        WRITE(6,929) MXQ
        WRITE(6,930) (ALM(K),K=1,MXQ)
        IF(ITI.GT.1) GO TO 77
        IF(ITI.EQ.1) GO TO 75
        CMS=EGNS(L)
        GO TO 80
75  CMS=(1+IPCT*.0001)*EGNS(L)
        GO TO 80
77  CMS=EGNN
80  CONTINUE
        IF(MODE.EQ.0) GO TO 8850
        WRITE(6,941)
8850 CALL BARRAY
        NBIGMD(L)=NBGMD
        CALL SARRAY
        IF(MODE.EQ.1) GO TO 99
        CALL EPSULN
        CALL SOLVE
        IF(ITI.EQ.0) PRINT 943, CMS,DETSV
        IF(NIT.NE.0) GO TO 8851
        MODE=1
        WRITE(6,941)
        CALL BARRAY
        NBIGMD(L)=NBGMD
        CALL SARRAY
        GO TO 99
8851 IF(ITI.GT.1) GO TO 93
        IF(ITI.EQ.1) GO TO 92
        EGNL=CMS
        REML=DETSV
        FACL=DFAC
        GO TO 98
92  EGNC=CMS
        FACT=DCMPLX(1.00,0.00)
        REMC=DETSV
        FAC=DFAC
        IF(FACC.EQ.FACL) GO TO 65
        DIFF=FACL-FACC
        IF(DIFF.LE.0.000)DIFF=-DIFF
        IF(DIFF.GT.70.00) STOP
        FACT=DCMPLX(10.00**((FACL-FACC),0.00)
65  EGNN=EGNC-REMC*(EGNC-EGNL)/(REMC-REML*FACT)
        GO TO 98
93  EGNL=EGNC
        EGNC=CMS
        REML=REMC
        FACL=FACC

```

```

FACT=DCMPLX(1.000,0.00)
REMC=DETSV
FACC=DFAC
IF(FACC.EQ.FACL) GO TO 66
DIFF=FACL-FACC
IF(DIFF.LE.0.00)DIFF=-DIFF
IF(DIFF.GT.70.00) STOP
FACT=DCMPLX(10.00**(FACL-FACC),0.00)
06 EGNN=EGNC-REMC*(EGNC-EGNL)/(REMC-REML*FACT)
ER=CDABS((EGNN-EGNC)/EGNC)
WRITE(6,934) REML,REMC,ER
WRITE(6,935) EGNL,EGNC,EGNN
IF((FR=.1**MER).GT.0.) GO TO 95
CONV(L)=0.0
94 MODE=1
EGNSL(L)=EGNC
ENGCH=EGNC*DCMPLX(0.00,-1.00)
RENGCH=ENGCH
PENOM(L)=RENGCH/OM
GO TO 80
95 IF(ITI.EQ.NIT) GO TO 94
98 ITI=ITI+1
GO TO 74
99 CONTINUE
WRITE(6,946)
DO 950 L=1,NDGNS
CON=BYES
IF(CONV(L).GT.1.0) CON=BN0
MTYPE=NBIGMD(L)
950 WRITE(6,948) L,EGNSL(L),PENOM(L),CON,ARRAY(MTYPE)
920 FORMAT(3(/),41X,I4,' VALUES OF UPDATED LAMBDA MATRIX')
930 FORMAT(10(1PE12.4))
934 FORMAT(2(/),41X,'LAST DETERMINANT VALUE =',2E13.6/
1 38X,'CURRENT DETERMINANT VALUE =',2E13.6/40X,'EIGEN VALUE CONVERG
2ENCE =',E13.6)
935 FORMAT(2(/),47X,'LAST EIGEN VALUE =',2E13.6/
1 44X,'CURRENT EIGEN VALUE =',2E13.6/
2 47X,'NEXT EIGEN VALUE =',2E13.6)
941 FORMAT(1H1,53X,'AEROELASTIC STABILITY'/58X,'STATE VECTORS')
943 FORMAT('0 INITIAL EIGENVALUE =',2G12.4,' DETERMINANT =',
1 2G12.4)
944 FORMAT('/ 10 NO EIGENVALUE FOUND IN THIS AREA')
946 FORMAT(1H1,20X,'S U M M A R Y   O F   P R E D I C T E D   M O D
1 E S'//T7,'MODE',T15,'DAMPING',T32,'FREQ IN',T49,'FREQ',T66,'CONVE
2RGENCE',T88,'PREDOMINANT'//T32,'RAD PER SEC',T49,'PER REV',T66,'CR
3ITERIA SATISFIED',T88,'MODE TYPE')
948 FORMAT(T7,I4,T15,2(E13.6,4X),F8.4,T72,A4,T90,A6)
ICNT=1
GO TO 1931
END

```

```

SUBROUTINE SETUP(EGNS,ICNT)
REAL DAT(2800)
REAL*8 PLC,PFC
COMPLEX EGNS(5)
COMPLEX*16 AMA(2550)
COMMON/AMAT/AMA
COMMON/ROTF/OM1,OM2,OMT
COMMON/FWVEL/FVLT,FVLP,FVLA,SGHI,CCHI
COMMON/SWASH/SWGJ,SWEI,SWM,SWR
COMMON/COUP/PFC,PLC
COMMON/RNAM/CS(4,6),SN(4,6)
COMMON/RNAM1/CSA(4,24),SNA(4,24)
COMMON/STAB/CP9I(24,6),SP9I(24,6)
COMMON/SPAR/AKCI(6),TAU(6),SMLA(6),DMS(6),AK(4),AC(4),BJ(4),
1CAPK,CAPC
COMMON/SPAR1/AKT(4),ACT(4),AKP(4),ACP(4)
COMMON/CPAR/AMKC(6),CTAU(6),AL(6,6),AL12(6)
COMMON/INTER/NSY,NBSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
1MFLEX,MCON,MAER,MFUS,NBC,NFLAP,NFEA,NCT,NCON,NFFB,
2NAS,NHC,NVI,NSP,MAXN,NES,MSC,NEGN,IPCT,NIT,NER,NORM,
3IREM,NEX,NPS,N9CH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NC9B,
4NFP1,MXSMI,MXT2P1,MXKG,MXCPL,MXC9B,MXCPM,MXCPK,MXSMB,
5NEBC,NE9BC,MFASB,MXFAB,NFUS,NRBD,NRIFC,MXG,NEIFC,
6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MODE
COMMON/PLOAD/STOR,OM,SDVEL,AIRD,PVEL,CHI,ZETA,TNC,APD,APD,COLL,AG,
1GT,GP,APD,ASK,BN,BPN,BNS,FNS,FLAPM,FEAM,CTM,CONM,AFLEX,AERM,
2AFFB,FLAN,FEN,CTN,CON,8CT,FUSH,ASN,CNH,VIN,SPC,SPH,ESN,SCC,WMA,
48RA,8WTS,8WB8,TFX,PRS,RT,EGNM,8R(5),8I(5),PCTI,ANIT,ERM,ANOR,AREM,
5ANEX,PNS,8CHN,AIG,AIF,8TG,8TF,PHIM(6),AKC(6),ATAU(6),YAK(4),YAC(4)
6,8MA(6),DBS(6),PAK(4),PAC(4),ECHI(4),AKE(4),ACE(4),8JE(4),CAKE,
7CACE,8TKC(6),8TAU(6),PHIC(6),THEC(6),BL12(6),8DCO,PUNI,
8CYFA,CYLA,CELM,CELN,DACO,CTYM,AEXT,EYEX,EZEX,COE1,COE2,COE3,
9PHOTT(8),PHWTT(8),CLESP(6),P(2000),V(25,24)
COMMON/DAMCU/OMDA
COMMON/NLEAD/MLEL,NLEL,MCTY
COMMON/CPARS/ALSTH(6)
COMMON/AMAS1/GCTCP
COMMON/SHAFT/TORFLX
COMMON/IPCH/IPU
COMMON/PGRAV/FGRA
DATA ANO/INO ' ',AYES/YES ' '
EQUIVALENCE(STOR,DAT(1))
NSY=100
NDAT=2800
IF(ICNT.GT.0)GO TO 10
DO 5 I=1,NDAT
5 DAT(I)=0.0
10 CALL LOADIN(DAT)
NBSEC=BNS+.1
NFSEC=FNS+.1

```

```

NB=BN+.1
NBP=HPN+.1
MFLAP=FLAPM+.1
MFEA=FEAM+.1
MCT=CTM+.1
MFLEX=AFLEX+.1
MCUN=CONM+.1
MLEL=CELM+.1
MAER=AERM+.1
MFUS=FUSM+.1
NBC=HCT+.1
NFLAP=FLAN+.1
NFEA=FEN+.1
NCT=CTN+.1
NCON=CON+.1
NLEL=CELN+.1
NFFB=AFFB+.1
NAS=ASN+.1
NHC=CNH+.1
NVI=VIN+.1
NSP=SPC+.1
MAXN=SPH+.1
NES=ESN+.1
MSC=SCC+.1
NEGN=EGNM+.1
IPCT=PCTI+.1
NIT=ANIT+.1
MER=ERM+.1
MCTY=CTYM+.1
NORM=ANOR+.1
IREM=AREM+.1
NEX=ANEX+.1
NPS=PNS+.1
IF(PNS.LT..0)NPS=PNS-.1
NSCH=SCHN+.1
IG=AIG+.1
IF=AIF+.1
NPRS=PRS+.1
NPD=APD+.1
NSK=ASK+.1
IPU=PUNI+.1
PFC=AFO
PLC=APO

```

C
C
C
C

```

DU 20 I=1,NEGN
A=SR(I)

```



```

      R=SI(I)
20  EGNS(I)=CMPLX(A,R)
      SWGJ=SWTS
      SWEI=SWBS
      SWM=SWMA
      SWR=SWRA
      TURFLX=TFX
      I=6
      DO 30 MS=1,NB
      DO 25 L=1,I
      ARG=L*PHIM(MS)
      CS(MS,L)=COS(ARG)
25  SN(MS,L)=SIN(ARG)
      AKCI(MS)=AKC(MS)
      TAU(MS)=ATAU(MS)
      CTAU(MS)=BTAU(MS)
      SMLA(MS)=SMA(MS)
      AMKC(MS)=STKC(MS)
      AL12(MS)=HL12(MS)
      ALSTH(MS)=CLESP(MS)
      PHI=PHIC(MS)
      THE=THEC(MS)
      CPHI=COS(PHI)
      SPHI=SIN(PHI)
      CTHE=COS(THE)
      STHE=SIN(THE)
      AL(1,MS)=CPHI*CTHE
      AL(4,MS)=-SPHI
      AL(3,MS)=-STHE
      AL(2,MS)=SPHI*CTHE
      AL(5,MS)=CPHI
      AL(6,MS)=CTHE
30  DMS(MS)=DBS(MS)
      IF(NSP.EQ.0)GO TO 45
      I=24
      DO 40 MS=1,NES
      DO 35 L=1,I
      ARG=L*ECHI(MS)
      CSA(MS,L)=COS(ARG)
35  SNA(MS,L)=SIN(ARG)
      AK(MS)=AKE(MS)
      AC(MS)=ACE(MS)
      AKT(MS)=TAK(MS)
      ACT(MS)=TAC(MS)
      AKP(MS)=PAK(MS)
      ACP(MS)=PAC(MS)
40  BJ(MS)=BJE(MS)
      CAPK=CAKE
      CAPC=CACE
      IF(MAXN.GT.1) GO TO 45

```

```

IF(SWEI.NE.0.0) GO TO 45
SWEI=1.0
SWGJ=SWEI
IF(SWR.EQ.0.0) SWR=1.0
45 NCOLS=6
NFP1=NHC+1
MXSMI=2*NHC+1
MXT2P1=(2*MAXN+1)*NSP
MXKQ=12*MXSMI
NCSB=1
MXCPL=MXSMI*MXSMI
MXCSB=12*NCSB
MXCPM=12*NCOLS
MXCPK=MXCPL*MXCPM
MXSMB=MXCSB*MXCPL
NEBC=MXCPM*MXSMI
NESBC=MXCSB*MXSMI
MFASB=NCSB*MXCPL
MXFAB=NCOLS*MXCPL
NFUS=6*MFUS
NRBD=NCOLS+MCT+MFEA+MFLAP+MLEL
IF(MFLEX.EQ.0) GO TO 49
NRBD=NCOLS+3*MCUN
IF(MCTY.EQ.1) NRBD=NCOLS+6*MCUN
49 NRIFC=NFUS+MXT2P1+NB*NRBD
MXQ=NRIFC*MXSMI
NEIFC=NRIFC*MXT2P1
NEISC=NRIFC*NFUS
NEITC=NRIFC*NRBD
MXTKN=NRIFC*NRIFC
AMSC=AYES
ANVI=AYES
AMFL=AYES
ACTT=AYES
AFEA=AYES
BFLEX=AND
ACON=AYES
AFUS=AYES
ASCH=AYES
ANPD=AYES
ANSK=AYES
ANSP=AYES
AAER=AYES
APRL=AYES
APRS=AYES
ANBP=AYES
AIPU=AYES
AMLE=AYES
IF(MLEL.EQ.0) AMLE=AND
IF(MFLAP.EQ.0) AMFL=AND

```

```

IF(MCT, EQ, 0) ACTT=AND
IF(MFEA, EQ, 0) AFEA=AND
IF(MFLEX, EQ, 0) BFLEX=AYES
IF(MCON, EQ, 0) ACON=AND
IF(MFUS, EQ, 0) AFUS=AND
IF(NSCH, EQ, 0) ASCH=AND
IF(NPD, EQ, 0) ANPD=AND
IF(NSK, EQ, 0) ANSK=AND
IF(NSP, EQ, 0) ANSP=AND
IF(MAER, EQ, 0) AAER=AND
IF(NBC, EQ, 0) APRL=AND
IF(NPRS, EQ, 0) APRS=AND
IF(NBP, EQ, 0) ANBP=AND
IF(NVI, EQ, 0) ANVI=AND
IF(MSC, EQ, 0) AMSC=AND
IF(IPU, EQ, 0) AIPU=AND
IF(NPRS, EQ, 0) GO TO 70
WRITE(6, 900)
WRITE(6, 902) NB, NBP, NBSEC, NFSEC, NBC, NHC, NFFB, NAS,
1NFLAP, NFEA, NCT, NCON, NES, MAXN, NSY, NSK
WRITE(6, 903) NEGN, IPCT, NIT, MER, NORM, IREM, NEX, NPS,
1IG, IF, NLEL, MCTY
WRITE(6, 905) AMFL, ACTT, AFEA, BFLEX, ACON, AFUS, ANBP,
1AAER, APRL, APRS, ANPD, ANSK, ANSP, AMSC, ANVI
WRITE(6, 906) ASCH, AIPU, AMLE
WRITE(6, 907) OM, SDVEL, AIRD, FVEL, CHI, ZETA, GT, GP,
1THC, AFO, APO, COLL
WRITE(6, 908) STG, STF, AG, TORFLX, SDCO, CYFA, CYLA, DACO, AEXT, EYEX, EZEX,
1COE1, COE2, COE3
DO 55 I=1, NB
IF(MFLEX, NE, 0) GO TO 56
WRITE(6, 909) I, PHIM(I), I, AKCI(I), I, TAU(I), I, SMLA(I), I, DMS(I)
GO TO 55
56 WRITE(6, 920) I, PHIM(I), I, AMKC(I), I, CTAU(I), I, PHIC(I), I,
1THEC(I), I, AL12(I), I, ALSTH(I)
55 CONTINUE
IF(MFLEX, EQ, 0) GO TO 57
WRITE(6, 915) PHOTT(1), PHOTT(2), PHOTT(3), PHOTT(4), PHOTT(5), PHOTT(6),
1PHOTT(7), PHOTT(8)
IF(MCTY, EQ, 0) GO TO 57
WRITE(6, 916) PHWTT(1), PHWTT(2), PHWTT(3), PHWTT(4), PHWTT(5), PHWTT(6),
1PHWTT(7), PHWTT(8)
57 CONTINUE
IF(NSP, EQ, 0) GO TO 65
WRITE(6, 910) SWGJ, SWEI, SWM, SWR
DO 60 I=1, NES
60 WRITE(6, 911) I, ECHI(I), I, AK(I), I, AC(I), I, BJ(I)
WRITE(6, 912) CAPK, CAPC
65 WRITE(6, 913)
WRITE(6, 914) (I, EGNS(I), I=1, NEGN)

```

```

70 OM1=OM
   OM2=OM*OM
   FGRA=AG
   GCTCP=COS(GT)*COS(GP)*AG
   OMT=2.0*OM
   IF(DACO,GT,0.0) OMT=0.0
   OMDA=OM*SDCO
   NFF=2*M*SMI

```

C

```

   IF(MAER,EQ,0)GO TO 80
   CALL TABLU
   DO 75 K=1,NAS
   DO 75 L=1,NFF
   ARG=L*((K-1)*3.1415926*2.0/NAS)
   CPSI(K,L)=COS(ARG)
75  SPSI(K,L)=SIN(ARG)
   SCHI=SIN(CHI)
   SZET=SIN(ZETA)
   CZET=COS(ZETA)
   CCHI=COS(CHI)
   FVLT=FVEL*SCHI*CZET
   FVLP=FVEL*CCHI*CZET
   FVLA=FVEL*SZET
80  CALL SECPAR
   IF(MFUS,EQ,0) GO TO 85
   IF(MAER,EQ,0) GO TO 85
   IF(NSK,EQ,0) GO TO 85
   CALL FAERU
85  CONTINUE
900 FORMAT(1H1,T37,'HELICOPTER AEROELASTIC STABILITY ANALYSIS INPUT PA
   1RAMETERS'//)
902 FORMAT(9X,'NB      ■ ',I3,'   NBP      ■ ',I3,'   NRSEC  ■ ',I3,3X,
   1'NFSEC  ■ ',I3,'   NBC      ■ ',I3,'   NMC      ■ ',I3,'   NFFR    ■ ',
   2I3,'   NAS      ■ ',I3,/,9X,'NFLAP  ■ ',I3,'   NFEA   ■ ',I3,3X,
   3'NCT    ■ ',I3,'   NCON    ■ ',I3,'   NES     ■ ',I3,'   MAXN   ■ ',
   4I3,'   NSY     ■ ',I3,'   FUA     ■ ',I3)
903 FORMAT(9X,'NEGN  ■ ',I3,'   IPCT   ■ ',I3,'   NIT    ■ ',I3,3X,
   1'NER    ■ ',I3,'   NORM   ■ ',I3,'   IREM   ■ ',I3,'   NEX    ■ ',
   2I3,'   NPS    ■ ',I3,/,9X,'IG    ■ ',I3,'   IF     ■ ',I3,
   3' NLEL  ■ ',I3,'   MCTY  ■ ',I3)
905 FORMAT(/,11X,'FLAP HINGE      ',A4,'   CONT TORQUE EQ  ',A4,
   1' PITCH BEARING  ',A4,'   ARTICULATED SYS ',A4,
   2' CONSTRAINT APP ',A4,/,11X,'FUSELAGE      ',A4,
   3' BLADE PHASING  ',A4,'   AERODYNAMICS   ',A4,
   4' GIMBALED SYSM. ',A4,'   VARIABLE OUTPUT ',A4,/,8X,
   5' DISK PLANE CALC ',A4,'   FUSELAGE AERO. ',A4,
   6' SWASHPLATE IN  ',A4,'   COLL SPRING IN  ',A4,
   7' VI DIST USED   ',A4)
906 FORMAT(11X,'SEARCH OPTION  ',A4,'   SHAPE PUNCHED  ',A4,
   1' LEAD-LAG HINGE ',A4)

```

```

907 FORMAT(/,9X,'ROTOR SPEED = ',F12.6,' VEL OF SOUND = ',F12.6,
1' AIR DENSITY = ',E12.5,' CRAFT VEL. = ',F12.6,/,6X,
2' ANGLE CHI = ',F12.6,' ANGLE ZETA = ',F12.6,
3' THETA GRAV. = ',F12.6,' PSI GRAVITY = ',F12.6,/,6X,
4' COEF. C(K) = ',F12.6,' DELTA3 = ',F12.6,
5' ALPHA1 = ',F12.6,' COLL. ANGLE = ',F12.6)
908 FORMAT(9X,'GR RATE STEP = ',F12.6,' FREQ. STEP = ',F12.6,
1' GRAV. ACC. = ',F12.6,' SHAFT FLEX. = ',E12.5,/,9X,
2' STR. DAMPING = ',F12.6,' F/A CYCLIC = ',F12.6,
3' LONG CYCLIC = ',F12.6,' DAMPING OUT = ',F12.6,/,9X,
4' SPUR LENGTH = ',F12.6,' SPUR E1Y = ',E12.5,
5' SPUR E1Z = ',E12.5,' COE1 = ',E12.5,/,9X,
6' COE2 = ',E12.5,' COE3 = ',E12.5,/)
909 FORMAT(10X,'PHIM(',I1,') = ',F11.7,' AKCI(',I1,') = ',F11.7,
1' TAU(',I1,') = ',F11.7,' SMLA(',I1,') = ',F11.7,
2' DMS(',I1,') = ',F11.7)
920 FORMAT(8X,'PHIM(',I1,') = ',F10.7,' AMKC(',I1,') = ',E11.5,
1' CTAU(',I1,') = ',F10.7,' PHIC(',I1,') = ',F10.7,
2' THEC(',I1,') = ',F10.7,/,6X,' AL12(',I1,') = ',F10.7,
3' ALSTH(',I1,') = ',F10.7,/)
910 FORMAT(/,8X,'SWASHPLATE , TORSIONAL STIFF. = ',E12.5,
1' BENDING STIFF. = ',E12.5,' MASS = ',F10.5,
2' RADIUS = ',F6.3,/)
911 FORMAT(19X,'ECHI(',I1,') = ',F12.6,' AK(',I1,') = ',E12.5,
1' AC(',I1,') = ',F12.6,' BJ(',I1,') = ',F12.7)
912 FORMAT(43X,'CAPK = ',F12.2,' CAPC = ',F12.6)
913 FORMAT(/)
914 FORMAT(40X,'STARTING EIGENVALUE ',I1,') = ',2E14.7)
915 FORMAT(/,9X,'EL = ',F12.6,' EAC = ',E12.5,' E1YY = ',E12.5,
1' E1ZZ = ',E12.5,' ALP = ',F12.6,/,9X,'BET = ',F12.6,
2' GAM = ',F12.6,' GJTT = ',E12.5,/)
916 FORMAT(/,9X,'PLEN = ',F12.6,' EACP = ',E12.5,' E1YP = ',E12.5,
1' E1ZP = ',E12.5,' ALPP = ',F12.6,/,9X,'BETP = ',F12.6,
2' GAMP = ',F12.6,' GJTP = ',E12.5,/)
RETURN
END

```

```

SUBROUTINE LOADIN(X)
C
C   DIMENSION T(5), X(1)
C
      IPPP = 0
      1 READ(5,10) K, J, (T(L),L=1,5)
      10 FORMAT(I4, I1, SE14.7)
         IF(J,EQ,6) GO TO 6
         IF(J,EQ,8) GO TO 8
         IF(J,EQ,9) GO TO 9
         IF(IPPP,NE,0) GO TO 11
         WRITE(6,45)
      45 FORMAT(1H1)
         WRITE(6,25)
         IPPP = 1
      11 DO 12 L=1,J
          N = K + L - 1
      12 X(N) = T(L)
         WRITE(6,55) K, J, (X(M),M=K,N)
      55 FORMAT(11X, I4, 8X, I1, 5X, 5(G14.5,4X))
         GO TO 1
      6  WRITE(6,35)
      35 FORMAT(5X, 'ERROR = INVALID CARD')
         GO TO 9
      25 FORMAT(T41, 'PRINT OUT OF INPUT DATA   '/11X, 'LOC', 8X, 'NUM',
      1  9X, 'VAL1', 14X, 'VAL2', 14X, 'VAL3', 14X, 'VAL4', 14X, 'VAL5'/)
      8  RETURN
      9  CALL EXIT
         END

```

```

SUBROUTINE SECPAR
REAL SD(4000)
REAL Y(100)
REAL SA(3,3),SAT(3,3),BC(3),SBT(3,3)
REAL*8 CX(75)
COMMON/SAIN/SD
COMMON/SUB/Y
COMMON/CFLEX/CX
COMMON/ROTF/OM1,OM2,OMT
COMMON/INTER/NSY,NBSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
1MFLEX,MCON,MAER,MFUS,NBC,NFLAP,NFEA,NCT,NCON,NFFB,
2NAB,NHC,NVI,NSP,MAXN,NES,MSC,NEGN,IPCT,NIT,NER,NORM,
3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NCSB,
4NFP1,MXSMI,MXT2P1,MXKU,MXCPL,MXCSB,MXCPM,MXCPK,MXSMR,
5NEBC,NEBSC,MFASB,MXFAB,MFUS,NRBD,NRIFC,MXQ,NEIFC,
6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MUDE
COMMON/PLOAD/STOR,DM,SDVEL,AIRD,FVEL,CHI,ZETA,THC,AFO,APD,COLL,AG,
1GT,GP,APD,ASK,BN,BPN,BNS,FNS,FLAPH,FEAM,CTM,CONM,AFLEX,AERM,
2AFFB,FLAN,FEN,CTN,CON,BCT,FUSM,ASN,CNH,VIN,SPC,SPH,ESN,SCC,WMA,
4SRA,SWTS,SWBS,TFX,PRS,RT,EGNM,SR(5),SI(5),PCTI,ANIT,ERM,ANQR,AREM,
5ANEX,PNS,SCHN,AIG,AIF,STG,STF,PHIM(6),AKC(6),ATAU(6),TAK(4),TAC(4)
6,SMA(6),DBS(6),PAK(4),PAC(4),ECHI(4),AKE(4),ACE(4),HJE(4),CAKE,
7CAGE,STKC(6),BTAU(6),PHIC(6),THEC(6),BL12(6),SDCO,PUNI,
8CYFA,CYLA,CELM,CELN,DACO,CTYM,AEXT,EYEX,EZEX,COE1,COE2,COE3,
9PHUTT(8),PHWTT(8),CLESP(6),P(2000),V(25,24)
COMMON/NLEAD/MLEL,NLEL,MCTY

```

C

```

COMMON/FWVEL/FVLT,FVLP,FVLA,SCHI,CCHI
COMMON/AMAS1/GCTCP
COMMON/FAIRS/ARVZ,ARVY,ARMD
COMMON/IARST/ILL,JSECK
ILL=0
TEMPN=0.0
TEMVY=0.0
TEMVZ=0.0
TEMPT=0.0
TEMMZ=0.0
TEMMY=0.0
WRITE(6,954)
954 FORMAT(/,34X,'STEADY FORCES AND MOMENTS',/,4X,'SECTION',5X,'N',
113X,'VY',12X,'VZ',12X,'T',13X,'MZ',12X,'MY',/)
DO 30 J=1,NSY
30 Y(J)=0.0
IF(COE2,NE.0.0) GO TO 98
COE2=1000.
WRITE(6,917)
917 FORMAT(/,10X,'COE2 WAS 0, RESET TO 1000, IF USED',/)
98 CONTINUE
MC=0
NTOT=NBSEC+NFSEC

```

```

DO 60 J=1,NTOT
JSECK=J
IF(J.NE.(NBSEC+1)) GO TO 32
TEMPN=0.0
TEMVY=0.0
TEMVZ=0.0
TEMPT=0.0
TEMMZ=0.0
TEMMY=0.0
32 IB=(J-1)*50
M1=P(IB+1)+.1
M2=P(IB+2)+.1
M3=P(IB+3)+.1
M4=P(IB+4)+.1
M5=P(IB+5)+.1
M6=P(IB+6)+.1
M7=P(IB+7)+.1
Y(1)=M1
Y(2)=M2
Y(3)=M3
Y(4)=M4
Y(5)=M5
Y(6)=M6
Y(7)=M7
P15=P(IB+15)
P16=P(IB+16)
P17=P(IB+17)
IF(J.GT.NBSEC) GO TO 33
IF(MFLEX.EQ.1)P17=P17+COLL
IF (MFLEX.EQ.0.AND.J.LE.NFFA)P17=P17+COLL
33 Y(11)=SIN(P15)
Y(12)=COS(P15)
Y(13)=SIN(P17)
Y(14)=COS(P17)
Y(15)=SIN(P16)
Y(16)=COS(P16)
Y(18)=Y(12)*Y(15)
Y(19)=Y(12)*Y(16)
Y(20)=Y(16)*Y(13)
Y(21)=Y(16)*Y(14)
Y(22)=Y(11)*Y(15)
Y(23)=Y(11)*Y(16)
Y(24)=-Y(11)*Y(14)+Y(18)*Y(13)
Y(25)=Y(12)*Y(14)+Y(22)*Y(13)
Y(26)=Y(11)*Y(13)+Y(18)*Y(14)
Y(27)=-Y(12)*Y(13)+Y(22)*Y(14)
GO TO 15
34 IF(M1.EQ.0) GO TO 35
Y(8)=P(IB+8)
Y(9)=P(IB+9)

```



```

P10=P(I8+10)
P11=P(I8+11)
P12=P(I8+12)
Y(10)=-P(I8+13)
P14=P(I8+14)
IF (M1 .EQ. 2) GO TO 1
Y(17)=OM2*Y(16)*Y(16)
Y(28)=OM2*Y(15)*Y(21)
Y(29)=OM2*Y(15)*Y(20)
Y(30)=OM2*Y(20)*Y(21)
C2TS2P=Y(20)*Y(20)
C2TC2P=Y(21)*Y(21)
S2T=Y(15)*Y(15)
Y(31)=OM2*(C2TS2P+S2T)
Y(32)=OM2*(C2TC2P+S2T)
Y(33)=Y(8)*Y(9)
EP2M=Y(33)*Y(9)
CZMX=P14-P12
Y(34)=P12+EP2M
Y(35)=P14+EP2M
CXMYPE=Y(34)-Y(10)
CZMYPE=Y(35)-Y(10)
O2ME=Y(33)*OM2
TDTVY=O2ME*(P10*Y(24)+P11*Y(25))
TDTVZ=O2ME*(P10*Y(26)+P11*Y(27))
TDTN=O2ME*(P10*Y(19)+P11*Y(23))
Y(36)=OMT*Y(20)*(Y(34)+Y(35)-Y(10))/2.0
Y(37)=OMT*Y(21)*(CZMX-Y(10))/2.0
Y(38)=OMT*Y(15)*(CZMX+Y(10))/2.0
Y(39)=CXMYPE*Y(28)
Y(40)=CXMYPE*OM2*(S2T-C2TS2P)+TDTVY
Y(41)=CXMYPE*Y(30)-TDTVZ
Y(42)=CZMYPE*Y(28)
Y(43)=CZMYPE*OM2*(C2TC2P-C2TS2P)+TDTVY
Y(44)=CZMYPE*Y(29)+TDTN
Y(45)=-CZMX*Y(29)
Y(46)=-CZMX*Y(30)
Y(47)=-CZMX*OM2*(C2TC2P-S2T)
TEHX=Y(8)*OM2*P10
TEHY=Y(8)*OM2*P11
TEMPN=TEMPN-Y(33)*Y(29)-TEHX*Y(19)-TEHY*Y(23)-Y(8)*GCTCP*Y(15)
TEHVV=TEHVV-Y(33)*Y(32)-TEHX*Y(24)-TEHY*Y(25)+Y(8)*GCTCP*Y(20)
TEHVZ=TEHVZ+Y(33)*Y(30)-TEHX*Y(26)-TEHY*Y(27)+Y(8)*GCTCP*Y(21)
TEMPT=TEMPT+CZMYPE*Y(30)-TDTVZ+Y(33)*GCTCP*Y(21)
TEHMZ=TEHMZ+CXMYPE*Y(29)+TDTN+Y(33)*GCTCP*Y(15)
TEHMY=TEHMY+CZMX*Y(28)
GO TO 35
1 Y(8)=-Y(8)
P12=P12
Y(10)=-Y(10)

```

```

P14=P14
Y(33)=Y(A)*Y(9)
Y(34)=P12+Y(33)*Y(9)
Y(35)=P14+Y(33)*Y(9)
35 IF(MAER.EQ.0) GO TO 36
   IF(M3.NE.1) GO TO 36
   ILL=ILL+1
   CALL HAERU
   TEMPT=TEMPT+ARMO
   TEMVZ=TEMVZ+ARVZ
   TEMVY=TEMVY+ARVY
   GO TO 36
2 IF(M2.EQ.0) GO TO 6
  IF (J.EQ.1) GO TO 5
  IF (J.EQ.(NRSEC+1)) GO TO 5
3 SA(1,1)=Y(19)
  SA(1,2)=Y(23)
  SA(1,3)=-Y(15)
  SA(2,1)=Y(24)
  SA(2,2)=Y(25)
  SA(2,3)=Y(20)
  SA(3,1)=Y(26)
  SA(3,2)=Y(27)
  SA(3,3)=Y(21)
  I=(J-2)*NSY
  SAT(1,1)=SD(I+19)
  SAT(1,2)=SD(I+24)
  SAT(1,3)=SD(I+26)
  SAT(2,1)=SD(I+23)
  SAT(2,2)=SD(I+25)
  SAT(2,3)=SD(I+27)
  SAT(3,1)=-SD(I+15)
  SAT(3,2)=SD(I+20)
  SAT(3,3)=SD(I+21)
  DO 4 I=1,3
  DO 4 N=1,3
  IND=59+3*(I-1)+N
  Y(IND)=0.0
  DO 4 K=1,3
4 Y(IND)=Y(IND)+SA(I,K)*SAT(K,N)
  ASTEN=TEMPN
  ASTVY=TEMVY
  ASTVZ=TEMVZ
  ASTET=TEMPT
  ASTMZ=TEMMZ
  ASTMY=TEMMY
  TEMPN= Y(60)*ASTEN+Y(61)*ASTVY+Y(62)*ASTVZ
  TEMVY= Y(63)*ASTEN+Y(64)*ASTVY+Y(65)*ASTVZ
  TEMVZ= Y(66)*ASTEN+Y(67)*ASTVY+Y(68)*ASTVZ
  TEMPT= Y(60)*ASTET+Y(62)*ASTMZ+Y(61)*ASTMY

```

```

      TEMMZ= Y(66)*ASTET+Y(68)*ASTMZ+Y(67)*ASTMY
      TEMMY= Y(63)*ASTET+Y(65)*ASTMZ+Y(64)*ASTMY
      GO TO 6
5  M2=0
      Y(2)=M2
      WRITE (6,902)
6  IF(M4, EQ, 0) GO TO 18
      STGJ=P(18+27)
      STEIY=P(18+28)
      STEIZ=P(18+29)
      Y(54)=TEMPN
      Y(55)=TEMVY
      Y(56)=TEMVZ
      Y(57)=TEMPT
      Y(58)=TEMMZ
      Y(59)=TEMMY
      IF (M4, EQ, 2) GO TO 13
      SLEN = P(18+26)
      SLEN2=SLEN*SLEN
      CFORC=TEMPN
      IF (J, EQ, 1) GO TO 14

```

C
C
C
C
C
C
C

```

      P(18+30)=CFORC
      TDTT=0.
      IF(P(18+25), LT, 0.001) GO TO 39
      IF(CFORC, EQ, 0.) GO TO 39
      TDTT=(STEIY+STEIZ)*CFORC/(2.0*P(18+25))
39  CONTINUE
      GAMZ=-SLEN*SQRT(CFORC/STEIZ)
      GAMY=-SLEN*SQRT(CFORC/STEIY)
      NT=0
      GAM=GAMZ
      STEI=STEIZ
9   GAM2=GAM*GAM
      IND=70+NT*6
      IF (GAM, LT, .02) GO TO 10
      SGAM=SINH(GAM)
      CGAM=COSH(GAM)
      Y(IND)=SLEN*SGAM/GAM
      Y(IND+1)=SLEN2*(CGAM-1.0)/(GAM2*STEI)
      Y(IND+2)=SLEN*SLEN2*(SGAM=GAM)/(GAM*GAM2*STEI)
      Y(IND+3)=CGAM
      GO TO 11
10  GAM4=GAM2*GAM2

```

```

Y(IND)=SLEN*(1. + GAM2/6.+GAM4/120.)
Y(IND+1)=SLEN2*(.5+GAM2/24.+GAM4/720.)/STEI
Y(IND+2)=SLEN*SLEN2*(1./6.+GAM2/120.+GAM4/5040.)/STEI
Y(IND+3)=1.0+GAM2/2.+GAM4/24.
11 IF(NT.EQ.1) GO TO 12
NT=1
GAM=GAMY
STEI=STEIY
GO TO 9
12 Y(69)=SLEN/(STGJ+TDTT)
Y(74)=Y(70)/STEIZ
Y(75)=SLEN
Y(80)=Y(76)/STEIY
Y(81)=SLEN
TEMMZ=-SLEN*TEMVY+TEMMZ
TEMMY=SLEN*TEMVZ+TEMMY
GO TO 18
13 SLEN = P(IR+26)
SLEN2=SLEN*SLEN
14 Y(69)=SLEN/STGJ
Y(70)=SLEN
Y(71)=SLEN2*.5/STEIZ
Y(72)=SLEN2*SLEN/(6.*STEIZ)
Y(73)=1.0
Y(74)=SLEN/STEIZ
Y(75)=SLEN
Y(76)=SLEN
Y(77)=SLEN2*.5/STEIY
Y(78)=SLEN2*SLEN/(6.*STEIY)
Y(79)=1.0
Y(80)=SLEN/STEIY
Y(81)=SLEN
TEMMZ=-SLEN*TEMVY+TEMMZ
TEMMY+=SLEN*TEMVZ+TEMMY
GO TO 18
15 IF(M5.EQ.0) GO TO 2
Y(82)=-P(IR+38)
Y(83)=-P(IR+39)
Y(84)=-P(IR+40)

```

C
C
C

```

IF (M5 .EQ. 1) GO TO 16
Y(82)=-Y(82)
Y(83)=-Y(83)
Y(84)=-Y(84)
16 Y(85)=Y(84)*TEMVZ+Y(83)*TEMVY
Y(86)=-Y(82)*TEMVZ
Y(87)=-Y(82)*TEMVY
Y(88) = -Y(84)*TEMPN

```

```

Y(89) = Y(83)*TEMVY+Y(82)*TEMPN
Y(90)=-Y(84)*TEMVY
Y(91)=-Y(83)*TEMPN
Y(92)=-Y(83)*TEMVZ
Y(93)=Y(82)*TEMPN+Y(84)*TEMVZ
TEMPT=+TEMVY*Y(84)-TEMVZ*Y(83)+TEMPT
TEMMZ=-TEMVY*Y(82)+TEMPN*Y(83)+TEMMZ
TEMMY=+TEMVZ*Y(82)-TEMPN*Y(84)+TEMMY
GO TO 2
17 IF(M6,EQ,0) GO TO 34
Y(94)=P(I8+44)
Y(95)=P(I8+45)
Y(96)=P(I8+46)
Y(97)=P(I8+47)
Y(98)=P(I8+48)
Y(99)=P(I8+49)
Y(100)=P(I8+50)
IF(M6,EQ,1) GO TO 34
Y(94)=-Y(94)
Y(95)=-Y(95)
Y(96)=-Y(96)
Y(100)=-Y(100)
GO TO 34
18 IF(M7,EQ,0) GO TO 17
Y(4)=1.

```

C

```

MC=MC+1
IF(MC ,EQ, 1) GO TO 19
WRITE (6,903)
STOP
19 EL=PHOTT(1)
EAC=PHUTT(2)
EIIY=PHOTT(3)
EIZZ=PHOTT(4)
ALP=PHOTT(5)
BET=PHOTT(6)
GAM=PHOTT(7)
GJTT=PHOTT(8)
EL80=EL*EL
ELCU=EL80*EL
IF(MCTY,GT,0) GO TO 70
CK1=3,0*EIIY/ELCU
CK2=3,0*EIZZ/ELCU
CK3=EAC/EL
CK4=CK1*EL
CK5=CK2*EL
CK6=CK1*EL80
CK7=CK2*EL80
GO TO 71
70 ALPP=PHWTT(5)

```

```

HETP=PHWTT(6)
GAMP=PHWTT(7)
SAL=SIN(ALPP)
CAL=COS(ALPP)
SBE=SIN(HETP)
CRE=COS(HETP)
SGA=SIN(GAMP)
CGA=COS(GAMP)
SAT(1,2)=-SAL*CGA+CAL*SBE*SGA
SAT(1,3)=+SAL*SGA+CAL*SBE*CGA
SAT(2,2)=+CAL*CGA+SAL*SBE*SGA
SAT(2,3)=-CAL*SGA+SAL*SBE*CGA
SAT(3,2)=CRE*SGA
SAT(3,3)=CRE*CGA
SAT11=SAT(3,2)*SAT(1,3)-SAT(3,3)*SAT(1,2)
SAT22=SAT(3,2)*SAT(2,3)-SAT(3,3)*SAT(2,2)
SAT33=SAT(2,2)*SAT(1,3)-SAT(2,3)*SAT(1,2)
AXTN=AEXT/EL
AXPEL=AEXT+EL
PHLEN=PHWTT(1)
TUS1=1.5*(1.+2.*AXTN)/EL
AXTE2=AEXT+PHLEN*SAT22
TUS2=EL/3.+0.5*PHLEN*SAT22
TUS3=0.25*AXTE2
TUS4=0.5*EL+PHLEN*SAT22
TUP1=TUS1*PHLEN
TUP2=TUS1*AXTE2
TUP3=0.5+1.5*AXTN
TUP4=TUP3*AXTE2
TUP5=EL+PHLEN*SAT22
AXTE1=AXTN*AXTN*AXTN
AXTZZ=AXTE1*(EIZZ/EZEX)
AXTTY=AXTE1*(EIYY/EYEX)
AXTE1=1.+3.*AXTN*(1.+AXTN)
AXTE2=AXTZZ+AXTE1
AXTE1=AXTTY+AXTE1
CK1=3.*EIYY/(AXTE1*ELCU)
CK2=3.*EIZZ/(AXTE2*ELCU)
CK3=0.0
CK4=CK1*AXPEL
CK5=CK2*AXPEL
CK6=CK4*AXPEL
CK7=CK5*AXPEL
71 SAL=SIN(ALP)
CAL=COS(ALP)
SBE=SIN(HET)
CRE=COS(HET)
SGA=SIN(GAM)
CGA=COS(GAM)
CASB=CAL*SBE

```

```

SASB=SAL*SBE
SA(1,1)=CAL*CBE
SA(1,2)=-SAL*CGA+CASB*SGA
SA(1,3)=SAL*SGA+CASB*CGA
SA(2,1)=SAL*CBE
SA(2,2)=CAL*CGA+SASB*SGA
SA(2,3)=-CAL*SGA+SASB*CGA
SA(3,1)=-SRE
SA(3,2)=CBE*SGA
SA(3,3)=CBE*CGA
DU 20 K=1,75
20 CX(K)=0,00
DO 24 I=1,3
BC(1)=-SA(I,1)*CK3
BC(2)=-SA(I,2)*CK2
BC(3)=-SA(I,3)*CK1
IXX=0
IF (I .EQ. 2) IXX=5
IF (I .EQ. 3) IXX=9
DO 21 N=I,3
IXA=IXX+N
DO 21 K=1,3
21 CX(IXA)=CX(IXA)+BC(K)*SA(N,K)
BC(2)=-SA(I,3)*CK4
BC(3)=-SA(I,2)*CK5
IXX=IXX+3
DO 22 N=1,3
IXA=IXX+N
DO 22 K=2,3
22 CX(IXA)=CX(IXA)+BC(K)*SA(N,K)
BC(2)=-SA(I,2)*CK6
BC(3)=-SA(I,3)*CK7
IXX=IXX+3*(5-I)
DO 23 N=I,3
IXA=IXX+N
DO 23 K=2,3
23 CX(IXA)=CX(IXA)+BC(K)*SA(N,K)
24 CONTINUE
IF(MCTY.EQ.0) GO TO 17
CK1=-1,0
CK3=-TUP1*SAT11/AXTE2
CK4=- (AXTZZ+TUP2)/AXTE2
CK5=TUP1*SAT33/AXTE1
CK6=- (AXTY+TUP2)/AXTE1
DO 72 I=1,3
IXX=21+(I-1)*3
BC(1)=-SA(I,1)*CK1
BC(2)=-SA(I,1)*CK3-SA(I,2)*CK4
BC(3)=-SA(I,1)*CK5-SA(I,3)*CK6
DU 72 N=1,3

```

```

IXA=IXX+N
DO 72 K=1,3
72 CX(IXA)=CX(IXA)+BC(K)*SA(N,K)
CK1=PHLEN*SAT33
CK2=PHLEN*SAT11
CK3=(AXTYT-TUP3)*PHLEN*SAT33/AXTE1
CK4=(AXTYT*TUP5+TUP4)/AXTE1
CK5=(AXTZZ-TUP3)*PHLEN*SAT11/AXTE2
CK6=(AXTZZ*TUP5+TUP4)/AXTE2
DO 73 I=1,3
IXX=30+(I-1)*3
BC(1)=-SA(I,2)*CK1-SA(I,3)*CK2
BC(2)=-SA(I,1)*CK3-SA(I,3)*CK4
BC(3)=-SA(I,1)*CK5-SA(I,2)*CK6
DO 73 N=1,3
IXA=IXX+N
DO 73 K=1,3
73 CX(IXA)=CX(IXA)+BC(K)*SA(N,K)
ICCP=0
CK1=PHLEN*SAT11
CK2=PHLEN*SAT22
CK3=PHLEN*SAT33
CK4=TUS1/AXTE1
CK5=-TUS1/AXTE2
DO 74 I=1,3
DO 74 N=1,3
74 SAT(I,N)=0.0
SAT(1,1)=1.0
SAT(2,1)=-CK1*CK5
SAT(3,1)=-CK3*CK4
SAT(2,2)=-CK2*CK5+(AXTZZ+AEXT*TUS1)/AXTE2
SAT(3,3)=-CK2*CK4+(AXTYT+AEXT*TUS1)/AXTE1
85 IXX=39+ICCP*9
DO 75 I=1,3
DO 75 N=1,3
SBT(N,I)=0.0
DO 75 K=1,3
75 SBT(N,I)=SBT(N,I)+SAT(K,I)*SA(N,K)
DO 76 N=1,3
IXAT=IXX+(N-1)*3
DO 76 I=1,3
IXA=IXAT+I
DO 76 K=1,3
76 CX(IXA)=CX(IXA)+SA(I,K)*SBT(N,K)
ICCP=ICCP+1
IF(ICCP.GT.3) GO TO 90
DO 78 I=1,3
DO 78 N=1,3
78 SAT(I,N)=0.0
IF(ICCP.GT.1) GO TO 79

```



```

CK4=(AXTZZ-TUP3)/AXTE2
CK5=(AXTYT-TUP3)/AXTE1
SAT(2,1)=-CK3*CK5
SAT(3,1)=-CK1*CK4
SAT(1,2)=CK3
SAT(3,2)=-CK2*CK4+EL*(AXTZZ+AXTN*TUP3)/AXTE2
SAT(1,3)=CK1
SAT(2,3)=CK2*CK5-EL*(AXTYT+AXTN*TUP3)/AXTE1
GO TO 85
79 IF(ICCP.GT.2) GO TO 80
CK4=TUS1/AXTE1
CK5=TUS1/AXTE2
SAT(2,1)=-CK1*CK5
SAT(3,1)=-CK3*CK4
SAT(2,2)=-CK2*CK5+(1.+1.5*AXTN)/AXTE2
SAT(3,3)=CK2*CK4+(1.+1.5*AXTN)/AXTE1
GO TO 85
80 CK4=EL*((AXTYT*TUS4+TUS3)/AXTE1)/EIYY
CK5=EL*((AXTZZ*TUS4+TUS3)/AXTE2)/EIZZ
CK6=EL*(.25+AXTZZ)*CK1/AXTE2)/EIZZ
CK7=EL*(.25+AXTYT)*CK3/AXTE1)/EIYY
SAT(1,1)=-CK3*CK7-CK1*CK6+EL/EAC
SAT(2,1)=-CK1*CK5
SAT(3,1)=-CK3*CK4
SAT(1,2)=-CK2*CK6+ELSQ*(CK1*(AXTN/4.-AXTZZ/2.)/AXTE2)/EIZZ
SAT(2,2)=CK3*CK3*EL/GJTT-CK2*CK5+ELSQ*((AXTZZ*TUS2+AXTN*TUS3)/AXTE
12)/EIZZ
SAT(3,2)=CK3*CK1*EL/GJTT
SAT(1,3)=CK2*CK7-ELSQ*(CK3*(AXTN/4.-AXTYT/2.)/AXTE1)/EIYY
SAT(2,3)=CK1*CK3*EL/GJTT
SAT(3,3)=CK1*CK1*EL/GJTT+CK2*CK4+ELSQ*((AXTYT*TUS2+AXTN*TUS3)/AXTE
11)/EIYY
GO TO 85
90 CONTINUE
CX(64)=-CX(64)
CX(65)=-CX(65)
CX(66)=-CX(66)
GO TO 17
36 IF(J.GT.NBSEC) GO TO 93
IF(MFLEX.GT.0) GO TO 43
IF(MFLAP.EQ.0) GO TO 37
IF(J.EQ.NFLAP) TEMMY=0.0
37 IF(MLEL.EQ.0) GO TO 38
IF(J.EQ.NLEL) TEMMZ=0.0
38 IF(MFEA.EQ.0) GO TO 41
IF(J.EQ.NFEA) TEMPT=0.0
41 IF(MCT.EQ.0) GO TO 93
IF(J.NE.NCT) GO TO 93
IF(MFEA.EQ.0) GO TO 91
TEMPT=TEMPT*COE3

```

```

GO TO 93
91 TEMVZ=TEMVZ-(COE1-TEMPT)/COE2
   TEMPT=COE1
   GO TO 93
43 IF(MCON, EQ, 0) GO TO 93
   IF(J, NE, NCON) GO TO 93
   TEMVZ=TEMVZ-(COE1-TEMPT)/COE2
   TEMPT=COE1
93 CONTINUE
   Y(48)=TEMPN
   Y(49)=TEMVY
   Y(50)=TEMVZ
   Y(51)=TEMPT
   Y(52)=TEMMZ
   Y(53)=TEMMY
   WRITE(6, 953) J, TEMPN, TEMVY, TEMVZ, TEMPT, TEMMZ, TEMMY
953 FORMAT(5X, I5, 6(2X, E12.5))
25 DO 50 L=1, NSY
   M=(J-1)*NSY+L
50 SD(M)=Y(L)
   DO 55 I=1, NSY
55 Y(I)=0.0
60 CONTINUE
   DO 200 KK=1, 4
   K=0
   DO 201 IS=1, NTOT
   ISS=IS
   IF (IS.GT. NBSEC) ISS=IS-NBSEC
   GO TO (301, 302, 303, 305), KK
301 IF (IS .EQ. 1) WRITE (6, 501)
   WRITE (6, 502) ISS, (P(L+K), L=1, 13)
   GO TO 202
302 IF (IS .EQ. 1) WRITE (6, 503)
   WRITE (6, 504) ISS, (P(L+K), L=14, 17), (P(L+K), L=19, 22)
   GO TO 202
303 IF (IS .EQ. 1) WRITE (6, 505)
   WRITE(6, 506) ISS, (P(L+K), L=23, 30)
   GO TO 202

```

C
C
C

```

305 IF (IS .EQ. 1) WRITE (6, 508)
   WRITE (6, 509) ISS, (P(L+K), L=38, 50)
202 K=K+50
201 CONTINUE
200 CONTINUE
501 FORMAT(1H1, //, T58, 'SECTION INPUT DATA', //, T18, 'ORDER OF SECTIONS:
1 BLADE TIP (SECTION 1) TO HUB, THEN FROM FREE END OF FUSELAGE (SECTI
2 ON 1) TO HUB', ///, T16, 'SECTION M1 M2 M3 M4 M5 M6
3 M7', 7X, 'M', 8X, 'ERY', 7X, 'HRX', 6X, 'HRY', 7X, 'IX', 8X, 'IY')

```

```

502 FORMAT(17X,I2,5X,7(F3.1,3X),F10.5,2X,F6.3,2(2X,F7.3),3X,F8.3,3X,
1F8.3)
503 FORMAT(1H1,8(/),T16,'SECTION',6X,'IZ',9X,'PHI',8X,'THETA',7X,'POI'
1,7X,'L(AERO)',6X,'ROLL',9X,'YAW',11X,'DEL')
504 FORMAT(17X,I2,6X,F10.4,3X,3(F8.5,3X),4(E11.4,2X))
505 FORMAT(1H1,8(/),T10,'SECTION',3X,'CHORD',9X,'IND. VEL',6X,'XXX',1
10X,'L(ELA)',8X,'(GJ)T',9X,'(EJ)Y',9X,'(EJ)Z',9X,'(P)T')
506 FORMAT(10X,I2,4X,8(E11.4,3X))

```

C
C

```

508 FORMAT(1H1,8(/),T4,'SECTION',2X,'DEL X   DEL Y   DEL Z   N-RIG
1VY-RIG  VZ-RIG   1/KX       1/KY       1/KZ       TAU X   TAU Y
2  TAU Z       K1)
509 FORMAT(6X,I2,4X,6(F7.4,1X),3(E11.4,1X),3(F7.4,1X),E11.4)
902 FORMAT(20X,'BEND NOT ALLOWED AT BLADE TIP OR FUSELAGE FREE END')
903 FORMAT(10X,'ONLY 1 CONSTRAINT SECTION ALLOWED, JOB ABORTE')
RETURN
END

```

```

SUBROUTINE BAERO
REAL SD(4000)
REAL Y (100)
REAL*8 TC(34)
REAL SPSI(24,6),CPSI(24,6)
COMPLEX EXPF
COMPLEX*16 AMA(2550)
COMMON/SAIN/SD
COMMON/SUB/Y
COMMON/AMAT/AMA
COMMON/STAB/CPSI,SPSI
COMMON/TAER/THCK,THC1,THC2
COMMON/AER2/RCD2,RC2D2,PRC2D4,PRC3D6,CD4MD,AZER
COMMON/AER3/DFYDP,DFZDP,DMDP,DFYDVY,DFZDVY,DMOVY,DFYDVZ,
1 DFZDVZ,DMOVZ,CAPPY,CAPFZ,CAPN,VLLY,VLLZ,CAPP,ATAB
COMMON/ROTF/OM1,OM2,OMT
COMMON/FWVEL/FVLT,FVLP,FVLA,SCHI,CCHI
COMMON/INTER/NSY,NRSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
1MFLEX,MCON,MAER,MFUS,NBC,NFLAP,NFEA,NCT,NCON,NFFB,
2NAS,NHC,NVI,NBP,MAXN,NES,NSC,NEGN,IPCT,NIT,NER,NORM,
3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NCSS,
4NFP1,MXSMI,MXT2P1,MXKQ,MXCPL,MXCSB,MXCPM,MXCPK,MXSMB,
5NEBC,NEBSC,MFASB,MXFAB,MFUS,NRBD,NRIPC,MXQ,NEIFC,
6NEISC,NEITC,MXTKN,NFP,MINPN,MAXPN,IBF,MODE
COMMON/NLEAD/NLEL,NLEL,MCTY
COMMON/LOAD/STOR,OM,SDVEL,AIRD,FVEL,CHI,ZETA,THC,AFO,APD,COLL,AG,
1GT,GP,APD,ASK,BN,BPN,BNS,FNS,FLAPM,FEAM,CTM,CONM,AFLEX,AERM,
2AFFB,FLAN,FEN,CTN,CON,BCT,PUSH,ASN,CNH,VIN,SPC,SPH,ESN,SCC,WMA,
4SRA,SWTS,SWBS,TFX,PRS,RT,EGNM,SR(5),SI(5),PCTI,ANIT,ERM,ANOR,AREM,
5ANEX,PNS,SCHN,AIG,AIF,STG,STF,PHIM(6),AKC(6),ATAU(6),TAK(4),TAC(4)
6,SMA(6),DBS(6),PAK(4),PAC(4),ECHI(4),AKE(4),ACE(4),BJE(4),CAKE,
7CACE,STKC(6),BTAU(6),PHIC(6),THEC(6),BL12(6),SDCO,PUNI,
8CYFA,CYLA,CELM,CELN,DACO,CTYM,AEXT,EYEX,EZEX,COE1,COE2,COE3,
9PHOTT(8),PHWTT(8),CLESP(6),P(2000),V(25,24)
COMMON/IARST/ILL,JSECK
COMMON/FAIRS/ARVZ,ARVY,ARMO
WRITE(6,101)
101 FORMAT(1H1,/,T58,'AERO COEFFICIENTS',/,
1T8,'ALPHZ',9X,'AMACZ',9X,'CSUBL',9X,'CSUBD'
2,9X,'CSUBM',9X,'CLALP',9X,'CDALP',9X,'CMALP')
IF(ILL.GT.1) GO TO 5
DO 12 I=1,2550
12 AMA(I)=DCMPLX(0.D0,0.D0)
5 PINU=3.141592654
NAMA=34*MXSMI
THCK=THC
I=JSECK
ARVZ=0.0
ARVY=0.0
ARMO=0.0

```

```

C      MAMA=(ILL-1)*NAMA
C
C      IB=(I-1)*50
C
C      P19=P(IB+19)
C      P22=P(IB+22)
C      P23=P(IB+23)
C      CD4=P23/4.0
C      RCD2=AIRO*P23/2.0
C      RC2D2=RCD2*P23
C      PRC2D4=PIRO*RC2D2/2.0
C      CD4MD=CD4-P22
C      PRC3D6=PRC2D4*CD4
C      THC1=1.0+THCK
C      THC2=(1.0-THCK)/4.0
C      ATAB=P(IB+18)
C      TFAC=1.0
C
C
C
C
C
C
C      OM15=OM1*Y(15)
C
C      OMD15=P22*OM15
C      OMHY=OM1*P(IB+11)
C      OMHX=OM1*P(IB+10)
C      CAPP=-OM15
C      VELIK=P(IB+24)
C      APSI=P(IB+17)
C      IF(MFLEX.EQ.1) APSI=APSI+COLL
C      IF(MFLEX.EQ.0.AND.I.LE.NFEA) APSI=APSI+COLL
C      DO 24 K=1,NAS
C      IF(NVI.GT.0) VELIK=V(ILL,K)
C      SSIK=SPSI(K,1)
C      CSIK=CPSI(K,1)
C      TPSI=CYLA*CSIK-CYFA*SSIK
C      STPSC=SIN(TPSI)
C      CTPSC=COS(TPSI)
C      TPSI=TPSI+APSI
C      STPS=SIN(TPSI)
C      CTPS=COS(TPSI)
C      FCY1=-Y(11)*CTPS+Y(18)*STPS
C      FCY2= Y(12)*CTPS+Y(22)*STPS
C      FCY3= Y(11)*STPS+Y(18)*CTPS
C      FCY4=-Y(12)*STPS+Y(22)*CTPS

```

OM20=OM1*Y(16)*STPS
 OM21=OM1*Y(16)*CTPS
 OMD20=P22*OM20
 OMD21=P22*OM21
 BVLI=FVLT*SSIK-OMHY+FVLA*CSIK
 BVLJ=FVLT*CSIK+OMHX=FVLA*SSIK
 BVLK=FVLP+VELIK
 VLLY=BVLI*FCY1+BVLJ*FCY2+BVLK*Y(16)*STPS
 VLZM=BVLI*FCY3+BVLJ*FCY4+BVLK*Y(16)*CTPS
 VLLZ=VLZM+OMD15
 VLXM=BVLI*Y(19)+BVLJ*Y(23)+BVLK*Y(15)
 VLLX=VLXM+OMD21
 AZER=SDVEL
 CALL AERQ(1,TFAC)
 DYPT=DFYDP
 DZPT=DFZDP
 DYYT=DFYDVY
 DZYT=DFZDVY
 DYZT=DFYDVZ
 DZZT=DFZDVZ
 CFYT=CAPFY
 CFZT=CAPFZ
 DFYDP=DYPT*CTPSC-DZPT*STPSC
 DFZDP=DZPT*CTPSC+DYPT*STPSC
 DFYDVY=DYYT*CTPSC-DZYT*STPSC
 DFZDVY=DZYT*CTPSC+DYYT*STPSC
 DFYDVZ=DYZT*CTPSC-DZZT*STPSC
 DFZDVZ=DZZT*CTPSC+DYZT*STPSC
 CAPFY=CFYT*CTPSC-CFZT*STPSC
 CAPFZ=CFZT*CTPSC+CFYT*STPSC
 TC(1)=DMDVZ*P22-DMDP
 TC(2)=DMDVY
 TC(3)=DMDVZ
 TC(4)=DFYDVZ*P22+DFYDP
 TC(5)=DFYDVY
 TC(6)=DFYDVZ
 TC(7)=DFZDVZ*P22-DFZDP
 TC(8)=DFZDVY
 TC(9)=DFZDVZ
 TC(10)=CAPFY
 TC(11)=CAPFZ
 TC(12)=DMDVZ*OM20=DMDVY*OM21
 TC(13)=DMDVZ*VLLY=DMDVY*VLZM
 TC(14)=DMDVZ*OM15
 TC(15)=DMDVZ*OMD20+DMDVY*VLXM=DMDP*OM20
 TC(16)=DMDVY*OM15
 TC(17)=DMDVZ*VLLX+DMDP*OM21
 TC(18)=CAPM
 TC(19)=DFYDVZ*OM20+DFYDVY*OM21
 TC(20)=DFYDVZ*VLLY+DFYDVY*VLZM=CAPFZ

```

TC(21)=-DFYDVZ*OM15
TC(22)=-DFYDVZ*OMD20-DFYDVY*VLXM+DFYDP*OM20
TC(23)=-DFYDVY*UM15
TC(24)=-DFYDVZ*VLLX-DFYDP*OM21
TC(25)=-CAPM
TC(26)=-DFZDVZ*OM20-DFZDVY*OM21
TC(27)=-DFZDVZ*VLLY-DFZDVY*VLZM-CAPFY
TC(28)=-DFZDVZ*OM15
TC(29)=-DFZDVZ*OMD20+DFZDVY*VLXM-DFZDP*OM20
TC(30)=-DFZDVY*OM15
TC(31)=-DFZDVZ*VLLX+DFZDP*OM21
TC(32)=-CAPM
TC(33)=-CAPFY
TC(34)=-CAPFZ
ARVZ=ARVZ+CAPFZ*P19
ARVY=ARVY+CAPFY*P19
ARMU=ARMU+CAPM*P19
DU 20 NN=1, MXSMI
N1=NN-NFP1
N=IABS(N1)
A=CPSI(K,N)
B=SPSI(K,N)
IF (N1.GT.0) GO TO 21
IF (N1.LT.0) GO TO 22
EXPF=CMPLX(1.0,0.0)
GO TO 23
21 EXPF=CMPLX(A,-B)
GO TO 23
22 EXPF=CMPLX(A,B)
23 IAM=MAMA+(NN-1)*34
DO 20 J=1,34
20 AMA(IAM+J)=AMA(IAM+J)-P19*TC(J)*EXPF
24 CONTINUE
26 CONTINUE
MAMP=MAMA+1
J=MAMA+NAMA
DU 28 I=MAMP,J
28 AMA(I)=AMA(I)/NAS
C WRITE(6,703)(AMA(IN),IN=1,204)
C 703 FORMAT(8F10.6)
ARVZ=ARVZ/NAS
ARVY=ARVY/NAS
ARMU=ARMU/NAS
RETURN
END

```

```

SUBROUTINE FAERU
DIMENSION SD(4000),Y(100),AFA(240)
COMMON/FUSA/AFA
COMMON/SAIN/SD
COMMON/SUR/Y
COMMON/AER2/RCD2,RC2D2,PRC2D4,PRC3D6,CD4MD,AZER
COMMON/AER3/DFYDP,DFZDP,DMDP,DFYDVY,DFZOVY,DMDVY,DFYDVZ,
1 DFZDVZ,DMDVZ,CAPPY,CAPFZ,CAPH,VLLY,VLLZ,CAPP,ATAB
COMMON/INTER/NSY,NBSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
1MFLEX,MCON,MAER,MFUS,NBC,NFLAP,NFEA,NCT,NCON,NFFB,
2NAS,NHC,NVI,NSP,MAXN,NES,MSC,NEGN,IPCT,NIT,MER,NORM,
3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NC9B,
4NFP1,MXSMI,MXT2P1,MXKQ,MXCPL,MXC9B,MXCPM,MXCPK,MXSMR,
5NEBC,NEHC,MFASB,MXFAB,NFUS,NRBD,NRIFC,MXQ,NEIFC,
6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MODE
COMMON/NLEAD/MLEL,NLEL,MCTY
COMMON/LOAD/STOR,OM,SDVEL,AIRD,FVEL,CHI,ZETA,THC,AFD,APD,COLL,AG,
1GT,GP,APD,ASK,BN,BPN,BNS,FNS,FLAPM,FEAM,CTM,CONM,AFLX,AERM,
2AFFB,FLAN,FEN,CTN,CUN,BCT,FUSM,ABN,CNH,VIN,SPC,SPH,ESN,SCC,WMA,
4SRA,SWTS,SWBS,TFX,PRS,RT,EGNM,SR(5),SI(5),PCTI,ANIT,ERM,ANOR,AREM,
5ANEX,PNS,SCHN,AIG,AIF,STG,STF,PHIM(6),AKC(6),ATAU(6),TAK(4),TAC(4)
6,SMA(6),DBS(6),PAK(4),PAC(4),ECHI(4),AKE(4),ACE(4),RJE(4),CAKE,
7CACE,STKC(6),BTAU(6),PHIC(6),THEC(6),BL12(6),SDCO,PUNI,
8CYFA,CYLA,CELM,CELN,DACO,CTYM,AEXT,EYEX,EZEX,COE1,COE2,COE3,
9PHOTT(8),PHWT(8),CLESP(6),P(2000),V(25,24)
AZER=SDVEL
IL=0
DO 12 I=1,240
12 AFA(I)=0.
PINU=3.141592654
DO 26 I=1,NFSEC
L=(I-1)*NSY+NBSEC*NSY
Y(3)=SD(L+3)
M3=Y(3)+.1
IF(M3,NE,2) GO TO 26
IL=IL+1
IB=(I-1)*50+NBSEC*50
DO 16 J=1,NBY
16 Y(J)=SD(L+J)
ATAB=P(IB+18)
MTAB=ATAB+.1
TFAC=1.0
IF(MTAB,EQ,7)TFAC=0.0
P19=P(IB+19)
P20=P(IB+20)
P21=P(IB+21)
P22=P(IB+22)
P23=P(IB+23)
CD4=P23/4.0
RCD2=AIRD+P23/2.0

```



```

RC2D2=RC02*P23
PRC2D4=PIN0*RC2D2/2.0
CD4MD=CD4*TFAC=P22
PRC3D6=PRC2D4*CD4
Y79=-FVEL*COS(P20)*COS(P21)
Y80=FVEL*SIN(P20)
Y81=-FVEL*COS(P20)*SIN(P21)
CAPP=0.
VLLX=Y80*Y(23)-Y81*Y(15)+Y79*Y(19)
VLLY=Y80*Y(25)+Y81*Y(20)+Y79*Y(24)
VLLZ=Y80*Y(27)+Y79*Y(26)+YA1*Y(21)
CALL AERO(2,TFAC)
IBB=(IL-1)*16
AFA(IBB+1)=CAPFY
AFA(IBB+2)=CAPFZ
AFA(IBB+3)=DMUP=P22*DMDVZ
AFA(IBB+4)=VLLZ*DMDVY-VLLY*DMDVZ
AFA(IBB+5)=DMDVY
AFA(IBB+6)=VLLX
AFA(IBB+7)=DMDVZ
AFA(IBB+8)=DFYDP=P22*DFYDVZ
AFA(IBB+9)=VLLZ*DFYDVY-VLLY*DFYDVZ=CAPFZ
AFA(IBB+10)=DFYDVY
AFA(IBB+11)=DFYDVZ
AFA(IBB+12)=DFZDP=P22*DFZDVZ
AFA(IBB+13)=VLLZ*DFZDVY-VLLY*DFZDVZ+CAPFY
AFA(IBB+14)=DFZDVY
AFA(IBB+15)=DFZDVZ
AFA(IBB+16)=CAPM
DO 20 K=1,5
20 AFA(IBB+K)=AFA(IBB+K)*P19
DU 21 K=7,16
21 AFA(IBB+K)=AFA(IBB+K)*P19
26 CONTINUE
RETURN
END

```

```

SUBROUTINE AERO(IBW,TFAC)
COMMON/TAER/THC1,THC2
COMMON/AER3/DFYDP,DFZDP,DMDP,DFYDVY,DFZDVY,DMDVY,DFYDVZ,
1DFZDVZ,DMDVZ,CAPFY,CAPFZ,CAPM,VLLY,VLLZ,CAPP,ATAB
COMMON/AER2/RCD2,RC2D2,PRC2D4,PRC3D6,CD4MD,SDVEL
MTAB=ATAB+.1
PIND=3.141592654
VZERO=SQRT(VLLY*VLLY+VLLZ*VLLZ)
IF (VLLZ.LT.0.0) GO TO 20
IF (VLLZ.GT.0.0) GO TO 30
IF (VLLY.LT.0.0) GO TO 11
ALPHZ=0.0
GO TO 40
11 ALPHZ=PIND
GO TO 40
20 IF (VLLY.LT.0.0) GO TO 21
IF (VLLY.GT.0.0) GO TO 22
ALPHZ=.5*PIND
GO TO 40
21 ALPHZ=PIND-ATAN(ABS(VLLZ/VLLY))
GO TO 40
22 ALPHZ=ATAN(ABS(VLLZ/VLLY))
GO TO 40
30 IF (VLLY.LT.0.0) GO TO 31
IF (VLLY.GT.0.0) GO TO 32
ALPHZ=1.5*PIND
GO TO 40
31 ALPHZ=PIND+ATAN(ABS(VLLZ/VLLY))
GO TO 40
32 ALPHZ=2.0*PIND-ATAN(ABS(VLLZ/VLLY))
40 CONTINUE
AMACZ=VZERO/SDVEL
AMAC1=1.0/(1.0-AMACZ*AMACZ)
AMAC2=(2.0-AMACZ*AMACZ)*AMAC1
IF (IBW.EQ.1.AND.VZERO.EQ.0.0)WRITE(6,101)
101 FORMAT(10X,'IMPENDING DIVISION BY VZERO EQUAL ZERO')
IF (IBW.EQ.2) THC1=2.0
IF (IBW.EQ.2) THC2=0.0
TC2C1=PRC2D4*THC1
TC3C1=PRC3D6*THC1
TC3C2=PRC3D6*THC2*4.0
IF (IBW.EQ.1)GO TO 2
IF (VZERO.NE.0.0)GO TO 2
RCD2V=0.0
TC3CM=0.0
GO TO 3
2 RCD2V=RCD2/VZERO
TC3CM=TC3C2/VZERO
3 RCYD2V=RCD2V*VLLY
RCZD2V=RCD2V*VLLZ

```

```

RCVD2=RCD2*VZERO
IF(IRW, EQ, 1) GO TO 5
IF(MTAB, GT, 5) GO TO 60
5 ALPHZ=180.*ALPHZ/PIN()
IF(ALPHZ, GT, 360.) ALPHZ=360.
CALL ACDEFF(ALPHZ, AMACZ, CSUBL, CSURD, CSUBM, CLALP, CDALP,
1 CMALP, MTAB)
GO TO 70
60 IF(MTAB, EQ, 6) GO TO 65
CLALP=0.0
CSUBL=0.0
CSURD=1.1
CDALP=0.0
GO TO 66
65 CLALP=5.73
IF(ALPHZ, GT, PIN()) ALPHZ=ALPHZ-2.*PIN()
CSUBL=CLALP*ALPHZ
CSURD=.006+.13131*ALPHZ*ALPHZ
CDALP=.26262*ALPHZ
IF(ALPHZ, LT, 0.) ALPHZ=ALPHZ+2.*PIN()
66 CMALP=0.
CSUBM=0.
ALPHZ=180.*ALPHZ/PIN()
70 CDTER=RCVD2*CSURD
100 FORMAT(5X, 8(E12.4, 2X))
WRITE(6, 100) ALPHZ, AMACZ, CSUBL, CSURD, CSUBM,
1 CLALP, CDALP, CMALP
CLTER=RCVD2*CSUBL
CMTER=RC2D2*AMAC2*CSUBM
CMATE=RC2D2*CMALP
VZCL=VLLZ*CSUBL
VZCD=VLLZ*CSURD
VYCL=VLLY*CSUBL
VYCD=VLLY*CSURD
FCSTL=VYCL*AMAC1+VLLZ*CLALP
FCWKL=VZCL*AMAC1-VLLY*CLALP
FCSTD=VYCD*AMAC1+VLLZ*CDALP
FCWKD=VZCD*AMAC1-VLLY*CDALP
DFYDP=-TC2C1*VLLZ
DFZDP=TC2C1*VLLY
DMDP=CD4MD*DFZDP-TC3C2*VZERO-TC3C1*VLLY*TFAC
DFYDVY=-RCZD2V*FCSTL-RCYD2V*FCSTD-CDTER
DFZDVY=RCYD2V*FCSTL-RCZD2V*FCSTD+CLTER+TC2C1*CAPP
DMDVY=CMTER*VLLY+CMATE*VLLZ+CD4MD*DFZDVY
1 =TC3CM*VLLY*CAPP-TC3C1*CAPP
DFYDVZ=-RCZD2V*FCWKL-RCYD2V*FCWKD-CLTER
1 =TC2C1*CAPP
DFZDVZ=RCYD2V*FCWKL-RCZD2V*FCWKD-CDTER
DMDVZ=CMTER*VLLZ+CMATE*VLLY+CD4MD*DFZDVZ
1 =TC3CM*VLLZ*CAPP

```

```
CAPFY=RCVD2*(VZCL+VYCD)+DFYDP*CAPP
CAPFZ=RCVD2*(VYCL-VZCD)+DFZDP*CAPP
CAPM=RC2D2*VZERO*VZERO*CSURM+CD4MU*CAPFZ
1      =CAPP*(TC3C2*VZERO+TC3C1*VLLY)
RETURN
END
```

```

SUBROUTINE TABLU
COMMON/COEFFS/NTAB,NMACH(1,3),AMACH(11,1,3),NCLS(11,1,3),
1ALPHS(11,50,1,3),CLDM(11,50,1,3),NCC(1,3),CCS(25,1,3)
1 FORMAT(14I5)
2 FORMAT(7F10.0)
3 FORMAT(5E14.7)
11 FORMAT(7X,9F7.4)
12 FURMAT(1F7.3,9F7.4)
14 FORMAT(1X,F6.1,9F7.4)
READ(5,1)NTAB
WRITE(6,901)
901 FORMAT(1H1,50X,'AERO TABLES',//)
DO 10 MM=1,NTAB
READ(5,1)ISER
IF(ISER.GT.0)WRITE(6,900)
900 FORMAT(10X,'FOR ISER EQUAL 1 ONLY CCS ARRAYS ARE INPUTTED')
READ(5,1)M
DO 20 I=1,3
READ(5,1)NCC1
NCC(M,I)=NCC1
READ(5,3)(CCS(K,M,I),K=1,NCC1)
IF(ISER.EQ.81) GO TO 6
IF(ISER.EQ.0)GO TU 5
IF(CCS(1,M,1).GE.0.)STOP
GO TO 20
5 READ(5,1)NMACHS
READ(5,2)(AMACH(K,M,I),K=1,NMACHS)
NMACH(M,I)=NMACHS
DO 30 J=1,NMACHS
READ(5,1)NCL
NCLS(J,M,I)=NCL
READ(5,2)(ALPHS(J,K,M,I),K=1,NCL)
30 READ(5,2)(CLDM(J,K,M,I),K=1,NCL)
GO TO 20
6 READ(5,1)NMACHS,NALF
WRITE(6,1)NMACHS,NALF
NMACH(M,I)=NMACHS
NXL=NMACHS
IF(NMACHS.GE.10) NXL=9
READ(5,11)(AMACH(K,M,I),K=1,NXL)
WRITE(6,11)(AMACH(K,M,I),K=1,NXL)
IF(NMACHS.GE.10) READ(5,11)(AMACH(K,M,I),K=10,NMACHS)
IF(NMACHS.GE.10)WRITE(6,11)(AMACH(K,M,I),K=10,NMACHS)
DO 7 J=1,NALF
READ(5,12)ALP,(CLDM(K,J,M,I),K=1,NXL)
WRITE(6,14) ALP,(CLDM(K,J,M,I),K=1,NXL)
IF(NMACHS.GE.10)READ(5,11)(CLDM(K,J,M,I),K=10,NMACHS)
IF(NMACHS.GE.10)WRITE(6,11)(CLDM(K,J,M,I),K=10,NMACHS)
DO 8 K=1,NMACHS
8 ALPHS(K,J,M,I)=ALP

```

```
7 CONTINUE
  DO 9 K=1, NMACHS
9  NCLS(K,M,I)=NALF
20 CONTINUE
10 CONTINUE
  RETURN
  END
```

```

SUBROUTINE ACOEFF(ALPDD,AMACZ,CSUHL,CSUBD,CSUBM,CLALP,CDALP,
1CMALP,MTAB)
COMMON/COEFFS/NTAB,NMACH(1,3),AMACH(11,1,3),NCLS(11,1,3),
1ALPHS(11,50,1,3),CLDM(11,50,1,3),NCC(1,3),CCS(25,1,3)
ALPHZ=ALPDD
PIN0=3.141592654
IF(CCS(1,MTAB,1).LT.0.0) GO TO 410
DO 400 L=1,3
MM=1
I=MTAB
IF(CCS(2,I,L).GT.180.)GO TO 52
IF(ALPHZ.GT.180.)ALPHZ=ALPHZ-360.
52 CONTINUE
IF((ALPHZ.LE.CCS(1,I,L)).OR.(ALPHZ.GE.CCS(2,I,L))) GO TO 50
GO TO 280
50 KSKIP=1
IF(AMACZ.LT.AMACH(1,I,L)) GO TO 70
GO TO 80
60 WRITE(6,901) ALPHZ,L
MM=2
GO TO 370
70 WRITE(6,902) AMACZ,L
MM=2
GO TO 370
80 NMAC=NMACH(I,L)
IF(AMACZ.GT.AMACH(NMAC,I,L)) GO TO 260
90 IF(ALPHZ.LT.ALPHS(1,1,I,L)) GO TO 60
NCL=NCLS(NMAC,I,L)
IF(ALPHZ.GT.ALPHS(NMAC,NCL,I,L)) GO TO 60
IF(KSKIP.EQ.2) GO TO 160
DO 150 K=1,NMAC
IF(AMACZ.LT.AMACH(K,I,L)) GO TO 140
IF(K.NE.NMAC) GO TO 150
140 I2 = K
I1 = I2-1
SGY = AMACH(I1,I,L)
RM=(AMACZ-SGY)/(AMACH(I2,I,L)-SGY)
GO TO 160
150 CONTINUE
160 NCLX=NCLS(I2,I,L)
ALF=ALPHZ
NT=1
193 DO 190 LL=1,NCLX
IF(ALF.LT.ALPHS(I2,LL,I,L)) GO TO 180
IF(LL.NE.NCLX) GO TO 190
180 J2=LL
J1=LL-1
SGY=ALPHS(I2,J1,I,L)
R2=(ALF-SGY)/(ALPHS(I2,J2,I,L)-SGY)
SGY=CLDM(I2,J1,I,L)

```

BEST AVAILABLE COPY

```

IF (NT.EQ.2) GO TO 192
CL2=SGY+R2*(CLDM(I2,J2,I,L)-SGY)
PLF=CLDM(I2,I,I,L)
GO TO 191
190 CONTINUE
191 NT=2
ALF=ALF+1
IF (ALF.GT.360.0) ALF=ALF-360.0
IF (CCS(2,I,L).LE.180..AND.ALF.GT.180.)ALF=ALF-360.
GO TO 193
192 SL2=SGY+R2*(CLDM(I2,J2,I,L)-SGY)
SLF=CLDM(I2,I,I,L)
IF (KSKIP.EQ.2) GO TO 270
NCLY=NCLS(I1,I,L)
ALF=ALPHZ
NT=1
243 DO 240 LL=1,NCLY
IF (ALF.LT.ALPHS(I1,LL,I,L)) GO TO 230
IF (LL.NE.NCLY) GO TO 240
230 J2=LL
J1=LL-1
SGY=ALPHS(I1,J1,I,L)
R1=(ALF-SGY)/(ALPHS(I1,J2,I,L)-SGY)
SGY=CLDM(I1,J1,I,L)
IF (NT.EQ.2) GO TO 242
CL1=SGY+R1*(CLDM(I1,J2,I,L)-SGY)
QMG=CLDM(I1,I,I,L)
GO TO 241
240 CONTINUE
241 NT=2
ALF=ALF+1
IF (ALF.GT.360.0) ALF=ALF-360.0
IF (CCS(2,I,L).LE.180..AND.ALF.GT.180.)ALF=ALF-360.
GO TO 243
242 SL1=SGY+R1*(CLDM(I1,J2,I,L)-SGY)
SMG=CLDM(I1,I,I,L)
250 CLCX=CL1+RM*(CL2-CL1)
DMDY=QMG+RM*(PLF-QMG)
SLCX=SL1+RM*(SL2-SL1)
SMDY=SMG+RM*(SLF-SMG)
GO TO 390
260 T2=NMAC
KSKIP=2
GO TO 90
270 CLCX=CL2
DMDY=PLF
SLCX=SL2
SMDY=SLF
390 IF (L=2) 391,392,393
391 CSUML=CLCX

```



```

      CLALP=180./PI*NO*(SLCX=CLCX)
      GO TO 400
392 CSUBO=CLCX
      CUALP=180./PI*NO*(SLCX=CLCX)
      GO TO 400
393 CSUMM=CLCX
      CMALP=180./PI*NO*(SLCX=CLCX)
400 CONTINUE
901 FORMAT(1H0,5X,'COMPUTED ALPHA = ',G12.4,
1'OUTSIDE RANGE OF COEFFS = ',I2)
902 FORMAT(1H0,5X,'COMPUTED MACH NO. = ',G12.4,
1'OUTSIDE RANGE OF COEFFS = ',I2)
370 RETURN
240 ALF=ALPHZ
      NT=1
      IF(L=2)600,700,800
600 DEL=ALF
      IF(ALF.GT.180.0)DEL=360.0-ALF
      TERM1=CCS(3,I,1)+CCS(4,I,1)*DEL+CCS(5,I,1)*DEL**2+
1      CCS(6,I,1)*DEL**3+CCS(7,I,1)*DEL**4
      TERM2=CCS(8,I,1)+CCS(9,I,1)*DEL+CCS(10,I,1)*DEL**2
      IF(ALF.GT.90.0) GO TO 610
      CLCXT=CCS(11,I,1)*TERM1
      GO TO 660
610 IF(ALF.GT.160.0)GO TO 620
      CLCXT=CCS(12,I,1)*TERM1
      GO TO 660
620 IF (ALF.GT.180.0) GO TO 630
      CLCXT=CCS(12,I,1)*TERM2+CCS(13,I,1)*(DEL-160.0)/20.0
      GO TO 660
630 IF(ALF.GT.200.0)GO TO 640
      CLCXT=-CCS(14,I,1)*TERM2+CCS(13,I,1)*(DEL-160.0)/20.0
      GO TO 660
640 IF(ALF.GT.270.0)GO TO 650
      CLCXT=-CCS(14,I,1)*TERM1
      GO TO 660
650 CLCXT=-CCS(15,I,1)*TERM1
660 IF(NT.EQ.1)CLCX=CLCXT
      SLCX=CLCXT
      IF(NT.EQ.2)GO TO 390
      NT=2
      ALF=ALF+1
      IF(ALF.GT.360.0) ALF=ALF-360.0
      GO TO 600
700 DEL=ALF
      IF(ALF.GT.90.0) GO TO 701
      XNO=DEL+(15.0-CCS(1,I,2))*((90.0-DEL)/(90.0-CCS(1,I,2)))
      GO TO 711
701 IF(ALF.GT.166.0) GO TO 703
      XNO=180.0-ALF

```

```

GO TO 711
703 IF(ALF.GT.140.0) GO TO 705
GO TO 713
705 IF(ALF.GE.194.0) GO TO 707
DEL=360.0-ALF
GO TO 713
707 IF(ALF.GE.270.0) GO TO 709
XNU=ALF-180.0
GO TO 711
709 XNU=360.0-ALF-(345.0-CCS(2,I,2))*((ALF-270.0)/
1 (CCS(2,I,2)-270.0))
711 CLCXT=CCS(3,I,2)+CCS(4,I,2)*XNU+CCS(5,I,2)*XNU**2+
1 CCS(6,I,2)*XNU**3+CCS(7,I,2)*XNU**4
GO TO 799
713 CLCXT=CCS(8,I,2)+CCS(9,I,2)*DEL+CCS(10,I,2)*DEL**2
799 IF(NT.EQ.1)CLCX=CLCXT
SLCX=CLCXT
IF(NT.EQ.2) GO TO 390
NT=2
ALF=ALF+1
IF(ALF.GT.360.0) ALF=ALF-360.0
GO TO 700
800 DEL=ALF
IF(ALF.GT.30.0) GO TO 801
CLCXT=CCS(3,I,3)*(30.0-DEL)/14.0+CCS(4,I,3)+
1 CCS(5,I,3)*DEL+CCS(6,I,3)*DEL**2+CCS(7,I,3)*DEL**3
GO TO 815
801 IF(ALF.GT.150.0) GO TO 803
802 CLCXT=CCS(8,I,3)+CCS(9,I,3)*DEL+CCS(10,I,3)*DEL**2+
1 CCS(11,I,3)*DEL**3+CCS(12,I,3)*DEL**4
GO TO 815
803 IF(ALF.GT.168.0) GO TO 805
804 CLCXT=CCS(13,I,3)+CCS(14,I,3)*DEL+CCS(15,I,3)*DEL**2+
1 CCS(16,I,3)*DEL**3
GO TO 815
805 IF(ALF.GT.140.0) GO TO 807
806 CLCXT=CCS(17,I,3)+CCS(18,I,3)*DEL+CCS(19,I,3)*DEL**2+
1 CCS(20,I,3)*DEL**3+CCS(21,I,3)*DEL**4
GO TO 815
807 DEL=360.0-ALF
IF(ALF.GE.192.0) GO TO 809
GO TO 806
809 IF(ALF.GE.210.0) GO TO 811
GO TO 804
811 IF(ALF.GE.330.0) GO TO 813
GO TO 802
813 CLCXT=CCS(22,I,3)*(ALF-330.0)/(CCS(2,I,3)-330.0)+CCS(4,I,3)+
1 CCS(5,I,3)*DEL-CCS(6,I,3)*DEL**2-CCS(7,I,3)*DEL**3
815 IF((ALF.GT.180.0).AND.(ALF.LT.330.0))CLCXT=-CLCXT
899 IF(NT.EQ.1)CLCX=CLCXT

```

```

SLCX=CLCXT
IF(NT,EN,2) GO TO 390
NT=2
ALF=ALF+1
IF(ALF,GT,360.0) ALF=ALF-360.0
GO TO 800
410 I=NTAB
CSUBM=0.0
CMALP=0.0
NEG=1
ALF=ALPHZ
SQT=SQRT(1.-AMACZ*AMACZ)
C1=1.-AMACZ
C2=CCS(2,I,1)*C1*(1.+CCS(3,I,1))
IF(ALF,LE,180.) GO TO 412
NEG=-1
ALF=-ALF+360.
412 ALF=ALF*PI/180.
IF(ALF,GT,C2) GO TO 414
CLALP=CCS(7,I,1)/SQT
CSUBL=CLALP*ALF
CSUBD=CCS(1,I,2)+CCS(2,I,2)*ALF*ALF
CDALP=2.0*CCS(2,I,2)*ALF
GO TO 450
414 IF(ALF,GT,CCS(4,I,1)) GO TO 416
C2=(CCS(11,I,1)+CCS(12,I,1)*AMACZ)*SQT
CLALP=(CCS(9,I,1)*AMACZ+CCS(10,I,1))/C2
CSUBL=CLALP*ALF+C1*CCS(8,I,1)/C2
NT=1
GO TO 418
416 IF(ALF,GT,CCS(5,I,1)) GO TO 420
NT=0
418 S1=SIN(ALF)
S2=SIN(2.0*ALF)
S3=SIN(3.0*ALF)
S4=SIN(4.0*ALF)
C1=COS(ALF)
C2=COS(2.0*ALF)
C3=COS(3.0*ALF)
C4=COS(4.0*ALF)
CSUBD=(CCS(3,I,2)+CCS(4,I,2)*C1+CCS(5,I,2)*C2
+CCS(6,I,2)*C3+CCS(7,I,2)*C4)/SQT
CDALP=(-CCS(4,I,2)*S1-2.*CCS(5,I,2)*S2-3.*CCS(6,I,2)*S3
+4.*CCS(7,I,2)*S4)/SQT
IF(NT,GT,0) GO TO 450
CSUBL=(CCS(13,I,1)*S1+CCS(14,I,1)*S2+CCS(15,I,1)*S3
+CCS(16,I,1)*S4)/SQT
CLALP=(CCS(13,I,1)*C1+2.*CCS(14,I,1)*C2+3.*CCS(15,I,1)*C3
+4.*CCS(16,I,1)*C4)/SQT
CSUBM=(CCS(1,I,3)*S1+CCS(2,I,3)*S2+CCS(3,I,3)*S3

```

```

1+CCS(4,I,3)*S4)/SQT=.25*(CSUBL*C1+CSUBD*S1)
CMALP=(CCS(10,I,3)*C1+CCS(11,I,3)*C2+CCS(12,I,3)*C3+CCS(13,I,3)*C4
1+S1*(CCS(14,I,3)*S2+CCS(15,I,3)*S3+CCS(16,I,3)*S4))/SQT
GO TO 450
420 IF(ALF.GT.CCS(6,I,1)) GO TO 422
CLALP=CCS(18,I,1)/SQT
CSUBL=CCS(17,I,1)/SQT+CLALP*ALF
CSUBM=CCS(5,I,3)/SQT
NT=1
GO TO 418
422 CLALP=CCS(20,I,1)/SQT
CSUBL=CCS(19,I,1)/SQT+CLALP*ALF
CSUBM=CCS(6,I,3)*(CCS(7,I,3)-((ALF+CCS(8,I,3))/CCS(9,I,3)
1))/SQT
CMALP=CCS(6,I,3)/(CCS(9,I,3)*SQT)
NT=1
GO TO 418
450 IF(NEG.GT.0) GO TO 370
CSUBL=CSUBL
CSUBM=CSUBM
CDALP=CDALP
GO TO 370
END

```

```

SUBROUTINE BARRAY
INTEGER EM(6)
REAL*8 RREAL(144),CX,PLC,PFC
COMPLEX UBX(1,35,3),UBY(1,35,3),UBZ(1,35,3),PHIX(1,35,3),
1PHIY(1,35,3),PHIZ(1,35,3),AN(1,35,3),VY(1,35,3),VZ(1,35,3),
2TEA(1,35,3),AMY(1,35,3),AMZ(1,35,3),QXJ,QYJ,QZJ
COMPLEX*16 CMS,CM1,CS1,CS2
COMPLEX*16 R(144),T(72),B(648),SMLB(108),SMLC(108),SMLD(108)
COMPLEX*16 BBAVE(648),SMLBSV(108),SMLCSV(108),SMLDSV(108)
COMPLEX*16 SHAPE(12),BLI,SMLBL,SMLCL,SMLDL,WSW
COMPLEX*16 EPS(63),ALM(63)
COMPLEX*16 CTB(54),FAB(54),FLB(54)
COMPLEX*16 CTB1(9),CTB2(9),CTB3(9),FAB1(9),FAB3(9),FLB1(9),FLB2(9)
COMPLEX*16 D(144),C(144)
COMPLEX*16 SMLE(108),SMLESV(108),FSB(54),SMLEL
COMPLEX*16 FSB1(9),FSB2(9),FSB3(9),FAB4(9),FLB4(9),CTB4(9)
COMPLEX*16 SMLF(108),SMLG(108),SMLFSV(108),SMLGSV(108),SMLFL,SMLGL
DIMENSION Y(100),SD(4000),CX(75)
COMMON/HARM/NHARM
COMMON/CALM/ALM
COMMON/EP5A/EP5
COMMON/ARR/RREAL
COMMON/ARI/R
COMMON/SUB/Y
COMMON/SAIN/SD
COMMON/ROTF/OM1,OM2,OMT
COMMON/FREF/CMS,CM1,CS1,CS2
COMMON/CFLEX/CX
COMMON/COUP/PFC,PLC
COMMON/BTS/B,SMLB,SMLC,SMLD,CTB,FAB,FLB,CTB1,CTB2,CTB3,FAB1,FAB3,
1FLB1,FLB2,SMLE,FSB,FSB1,FSB2,FSB3,FAB4,FLB4,CTB4,SMLF,SMLG
COMMON/QVTEM/QXJ,QYJ,QZJ
COMMON/ISUMA/NBGMD
COMMON/NLEAD/MLEL,NLEL,MCTY
COMMON/INTER/NSY,NBSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
1MFLEX,MCON,MAER,MFUS,NBC,NFLAP,NFEA,NCT,NCON,NFFB,
2NAS,NHC,NVI,NSP,MAXN,NEB,MSC,NEGN,IPCT,NIT,NER,NORM,
3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NC9B,
4NFP1,MXSMI,MXT2P1,MXKQ,MXCPL,MXC9B,MXCPM,MXCPK,MXSMB,
5NEBC,NE9BC,MFASB,MXFAB,NFUS,NRBD,NRIFC,MXQ,NEIFC,
6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MODE
COMMON/IPCH/IPU
DATA EM/1,15,29,42,57,70/
DATA BLANKS/' ',BTIPS/'TIP ',BHUB/'HUB '/
IL=0
SPACE=BLANKS
IWRITE=0
IF(MODE,EQ,1) IWRITE=1
NCCT=1
NCFEA=1

```

```

NCFLP=1
NCLEL=1
NCACF=1
NCACG=1
ICT=NCT
IFEA=NFEA
IFLA=NFLAP
ILEL=NLEL
IF(MCT,EQ,0) NCT=NHSEC+2
IF(MFEA,EQ,0) NFEA=NHSEC+2
IF(MFLAP,EQ,0) NFLAP=NHSEC+2
IF(MLEL,EQ,0) NLEL=NHSEC+2
IH=0
IC=0
ID=0
IE=0
IFF=0
IGG=0
MCACF=0
MCACG=0
DO 15 L=1,MXSMH
SMLE(L)=DCMPLX(0.00,0.00)
SMLF(L)=DCMPLX(0.00,0.00)
SMLG(L)=DCMPLX(0.00,0.00)
SMLH(L)=DCMPLX(0.00,0.00)
SMLC(L)=DCMPLX(0.00,0.00)
15 SMLD(L)=DCMPLX(0.00,0.00)
DO 22 K=1,MXCPK
22 H(K)=DCMPLX(0.00,0.00)
DO 30 I=1,MXSMI
L=(I-1)*MXSMI+I
LM1=(L-1)*MXCPM
DO 31 M=1,6
K=LM1+EM(M)
31 H(K)=DCMPLX(1.000,0.00)
IF(MFLEX,NE,0) GO TO 25
IF(MLEL,EQ,0) GO TO 26
K=L*MXCSH=6
SMLE(K)=DCMPLX(1.00,0.00)
26 IF(MFEA,EQ,0) GO TO 23
K=L*MXCSH=9
KA=K+3
SMLC(KA)=DCMPLX(PLC,0.00)
SMLC(K)=DCMPLX(1.00,0.00)
23 IF(MFLAP,EQ,0) GO TO 24
K=L*MXCSH=2
KA=K+7
SMLD(KA)=DCMPLX(PFC,0.00)
SMLD(K)=DCMPLX(1.000,0.00)
24 IF(MCT,EQ,0) GO TO 30

```

```

K=L*MXCSH-A
SMLB(K)=DCMPLX(1.00,0.00)
GO TO 30
25 IF(MCON,EN,0)GO TO 30
K=L*MXCSB-10
SMLB(K)=DCMPLX(CX(1),0.00)
SMLC(K)=DCMPLX(CX(2),0.00)
SMLD(K)=DCMPLX(-CX(3),0.00)
SMLE(K)=DCMPLX(-CX(22),0.00)
SMLF(K)=DCMPLX(-CX(25),0.00)
SMLG(K)=DCMPLX(-CX(2A),0.00)
K=K+2
SMLB(K)=DCMPLX(CX(4),0.00)
SMLC(K)=DCMPLX(CX(9),0.00)
SMLD(K)=DCMPLX(-CX(13),0.00)
SMLE(K)=DCMPLX(-CX(31),0.00)
SMLF(K)=DCMPLX(-CX(34),0.00)
SMLG(K)=DCMPLX(-CX(37),0.00)
K=K+3
SMLB(K)=DCMPLX(CX(6),0.00)
SMLC(K)=DCMPLX(CX(11),0.00)
SMLD(K)=DCMPLX(-CX(15),0.00)
SMLE(K)=DCMPLX(-CX(33),0.00)
SMLF(K)=DCMPLX(-CX(36),0.00)
SMLG(K)=DCMPLX(-CX(39),0.00)
K=K+1
SMLB(K)=DCMPLX(-CX(2),0.00)
SMLC(K)=DCMPLX(-CX(7),0.00)
SMLD(K)=DCMPLX(CX(4),0.00)
SMLE(K)=DCMPLX(CX(23),0.00)
SMLF(K)=DCMPLX(CX(26),0.00)
SMLG(K)=DCMPLX(CX(29),0.00)
K=K+3
SMLB(K)=DCMPLX(CX(5),0.00)
SMLC(K)=DCMPLX(CX(10),0.00)
SMLD(K)=DCMPLX(-CX(14),0.00)
SMLE(K)=DCMPLX(-CX(32),0.00)
SMLF(K)=DCMPLX(-CX(35),0.00)
SMLG(K)=DCMPLX(-CX(38),0.00)
K=K+1
SMLB(K)=DCMPLX(CX(3),0.00)
SMLC(K)=DCMPLX(CX(8),0.00)
SMLD(K)=DCMPLX(-CX(12),0.00)
SMLE(K)=DCMPLX(-CX(24),0.00)
SMLF(K)=DCMPLX(-CX(27),0.00)
SMLG(K)=DCMPLX(-CX(30),0.00)
30 CONTINUE
IF(MUDE,GT,0) GO TO 42
DO 40 I=1,N*PAH
FSB(I)=DCMPLX(0.00,0.00)

```

BEST AVAILABLE COPY

```

      FAB(I)=DCMPLX(0.00,0.00)
      FLB(I)=DCMPLX(0.00,0.00)
40  CTH(I)=DCMPLX(0.00,0.00)
      DO 41 I=1,NFASH
      FSH1(I)=DCMPLX(0.00,0.00)
      FSH2(I)=DCMPLX(0.00,0.00)
      FSH3(I)=DCMPLX(0.00,0.00)
      FAH4(I)=DCMPLX(0.00,0.00)
      FLH4(I)=DCMPLX(0.00,0.00)
      CTH4(I)=DCMPLX(0.00,0.00)
      FAH1(I)=DCMPLX(0.00,0.00)
      FAH3(I)=DCMPLX(0.00,0.00)
      CTH2(I)=DCMPLX(0.00,0.00)
      CTH3(I)=DCMPLX(0.00,0.00)
      FLH1(I)=DCMPLX(0.00,0.00)
41  FLH2(I)=DCMPLX(0.00,0.00)
42  DO 501 IS=1,NRSEC
      ISM1=(IS-1)*NSY
      DO 50 L=1,NSY
      M=ISM1+L
50  Y(L)=SD(M)
      IF(Y(5).EQ.0.)GO TO 80
      CALL RIGID
      LGO=1
60  DO 79 J=1,MXSMI
      DO 79 J=1,MXSMI
      L=(J-1)*MXSMI+I
      LM1=(L-1)*MXCPM
      LHM1=(L-1)*MXCSH
      DO 66 M=1,MXCPM
      K=LM1+M
66  T(M)=B(K)
      CALL MLRC2(RREAL,T,NCOLS)
      DO 70 M=1,MXCPM
      K=LM1+M
70  B(K)=T(M)
      IF(NCACF.EQ.1)GO TO 83
      DO 81 M=1,MXCSH
      K=LBM1+M
81  T(M)=SMLF(K)
      CALL MLRC2(RREAL,T,NCSH)
      DO 82 M=1,MXCSH
      K=LBM1+M
82  SMLF(K)=T(M)
83  IF(NCACG.EQ.1)GO TO 88
      DO 84 M=1,MXCSH
      K=LBM1+M
84  T(M)=SMLG(K)
      CALL MLRC2(RREAL,T,NCSH)
      DO 85 M=1,MXCSH

```



```

      K=LBM1+M
65 SMLG(K)=T(M)
68 IF(NCLFL.EQ.1) GO TO 69
   DO 66 M=1, MXCSH
      K=LBM1+M
66 T(M)=SMLF(K)
      CALL MLRC2(RREAL,T,NCSH)
   DO 67 M=1, MXCSH
      K=LBM1+M
67 SMLG(K)=T(M)
69 IF(NCFFA.EQ.1) GO TO 7A

```

C

```

   DO 72 M=1, MXCSH
      K=LBM1+M
72 T(M)=SMLC(K)
      CALL MLRC2(RREAL,T,NCSH)
   DO 76 M=1, MXCSH
      K=LBM1+M
76 SMLC(K)=T(M)
7A IF(NCFIP.EQ.1) GO TO 73

```

C

```

   DO 722 M=1, MXCSH
      K=LBM1+M
722 T(M)=SMLD(K)
      CALL MLRC2(RREAL,T,NCSH)
   DO 762 M=1, MXCSH
      K=LBM1+M
762 SMLD(K)=T(M)
73 IF(NCCT.EQ.1) GO TO 79

```

C

```

   DO 723 M=1, MXCSH
      K=LBM1+M
723 T(M)=SMLH(K)
      CALL MLRC2(RREAL,T,NCSH)
   DO 763 M=1, MXCSH
      K=LBM1+M
763 SMLH(K)=T(M)
79 CONTINUE
   IF(LG0.EQ.2) GO TO 100
   IF(LG0.EQ.3) GO TO 120
80 IF(Y(2).EQ.0.) GO TO 100
   CALL BEAD
   LG0=2
   GO TO 60
100 IF(Y(4).EQ.0.) GO TO 120
   CALL ELAST
   LG0=3
   GO TO 60
120 DO 200 I=1, MXSMI
      KSML=I-NFP1

```

BEST AVAILABLE COPY

```

NST=KSML
CS1=CMS=CM1*NST*DM1
CS2=CS1*CS1
IF (Y(6).EQ.0.) GO TO 145
CALL STIFF
LGO=1
127 DO 404 J=1, MXSMI
    L=(J-1)*MXSMI+1
    LM1=(L-1)*MXCPM
    LHM1=(L-1)*MXCSH
    DO 136 M=1, MXCPM
        K=LM1+M
136 T(M)=B(K)
    CALL MLCC2(R,T,NCDLS)
    DO 137 M=1, MXCPM
        K=LM1+M
137 B(K)=T(M)
    IF (NCACF.EQ.1) GO TO 153
    DO 151 M=1, MXCSH
        K=LHM1+M
151 T(M)=SMLF(K)
    CALL MLCC2(R,T,NCSH)
    DO 152 M=1, MXCSH
        K=LHM1+M
152 SMLF(K)=T(M)
153 IF (NCACG.EQ.1) GO TO 156
    DO 154 M=1, MXCSH
        K=LHM1+M
154 T(M)=SMLG(K)
    CALL MLCC2(R,T,NCSH)
    DO 155 M=1, MXCSH
        K=LHM1+M
155 SMLG(K)=T(M)
156 IF (NCLEL.EQ.1) GO TO 135
    DO 131 M=1, MXCSH
        K=LHM1+M
131 T(M)=SMLE(K)
    CALL MLCC2(R,T,NCSH)
    DO 132 M=1, MXCSH
        K=LHM1+M
132 SMLE(K)=T(M)
135 IF (NCFPA.EQ.1) GO TO 142
C
    DO 139 M=1, MXCSH
        K=LHM1+M
139 T(M)=SMLE(K)
    CALL MLCC2(R,T,NCSH)
    DO 140 M=1, MXCSH
        K=LHM1+M
140 SMLC(K)=T(M)

```

```

C
142 IF(NCFLP.EQ.1) GO TO 405
C
DO 402 M=1, MXCSH
K=LBM1+M
402 T(M)=SMLD(K)
CALL MLCC2(R,T,NCSH)
DO 403 M=1, MXCSH
K=LBM1+M
403 SMLD(K)=T(M)
405 IF(NCCT.EQ.1) GO TO 404
C
DO 407 M=1, MXCSH
K=LBM1+M
407 T(M)=SMLH(K)
CALL MLCC2(R,T,NCSH)
DO 408 M=1, MXCSH
K=LBM1+M
408 SMLH(K)=T(M)
404 CONTINUE
IF(LGI.GT.1) GO TO 200
145 IF(Y(1).EQ.0.) GO TO 200
NHAR=0
CALL BMASS
LGI=2
GO TO 127
200 CONTINUE
IF(MAER.EQ.0) GO TO 305
IF(Y(3).EQ.0.) GO TO 305
IL=IL+1
IF(-CACF.EQ.1) GO TO 158
DO 157 K=1, MXSMH
SMLFSV(K)=SMLF(K)
157 SMLF(K)=DCMPLX(0.00,0.00)
158 IF(NCACG.EQ.1) GO TO 160
DO 159 K=1, MXSMH
SMLGSV(K)=SMLG(K)
159 SMLG(K)=DCMPLX(0.00,0.00)
160 IF(NCLFL.EQ.1) GO TO 2140
DO 218 K=1, MXSMH
SMLFSV(K)=SMLF(K)
218 SMLF(K)=DCMPLX(0.00,0.00)
2140 IF(NCFFA.EQ.1) GO TO 2141
DO 214 K=1, MXSMH
SMLCSV(K)=SMLC(K)
214 SMLC(K)=DCMPLX(0.00,0.00)
2141 IF(NCFLP.EQ.1) GO TO 2143
DO 212 K=1, MXSMH
SMLDSV(K)=SMLD(K)
212 SMLD(K)=DCMPLX(0.00,0.00)

```

BEST AVAILABLE COPY

```

2143 IF (NCCT.EQ.1) GO TO 2142
      DO 216 K=1, MXSMH
      SMLBSV(K)=SMLH(K)
216 SMLH(K)=DCMPLX(0.00,0.00)
2142 DO 215 K=1, MXCPK
      HSAVE(K)=H(K)
215 H(K)=DCMPLX(0.00,0.00)
      NTIMS=MXSMI+1
      DO 300 NN=1, NFP1
      NHARM=ND=1
      NTIMS=NTIMS-1
      NSHFT=NN-NFP1-1
      NHACK=1
217 CALL MLARD(C,D,IL)
      DO 250 JQ=1, NTIMS
      NSHFT=NSHFT+1
      NST=NSHFT
      IQ=JQ+NN-NHACK
      I=JQ+NHACK-1
      CS1=CMS=CM1*OM1*NSHFT
      DO 220 M=1, 144
220 R(M)=CS1*C(M)+D(M)
      DO 250 J=1, MXSMI
      L=(J-1)*MXSMI+I
      LQ=(J-1)*MXSMI+IQ
      LQM1=(LQ-1)*MXCPM
      LM1=(L-1)*MXCPM
      LBQ=(LQ-1)*MXCSH
      LB=(L-1)*MXCSH
      DO 240 M=1, MXCPM
      K=LQM1+M
240 T(M)=HSAVE(K)
      CALL MLCC2(R,T,NCULS)
      DO 245 M=1, MXCPM
      K=LM1+M
245 B(K)=T(M)+R(K)
      IF (NCACF.EQ.1) GO TO 163
      DO 161 M=1, MXCSH
      K=LBQ+M
161 T(M)=SMLFSV(K)
      CALL MLCC2(R,T,NCSE)
      DO 162 M=1, MXCSH
      K=LH+M
162 SMLF(K)=T(M)+SMLF(K)
163 IF (NCACG.EQ.1) GO TO 166
      DO 164 M=1, MXCSH
      K=LHQ+M
164 T(M)=SMLGSV(K)
      CALL MLCC2(R,T,NCSE)
      DO 165 M=1, MXCSH

```

```

      K=LN+M
165 SMLG(K)=T(M)+SMLG(K)
166 IF(NCLEL.EQ.1) GO TO 244
      DO 241 M=1, MXCSH
      K=LH0+M
241 T(M)=SMLESV(K)
      CALL MLCC2(R,T,NCSH)
      DO 242 M=1, MXCSH
      K=LN+M
242 SMLF(K)=T(M)+SMLF(K)
244 IF(NCFEA.EQ.1) GO TO 249

```

C
C

```

      DO 247 M=1, MXCSH
      K=LH0+M
247 T(M)=SMLCSV(K)
      CALL MLCC2(R,T,NCSH)
      DO 248 M=1, MXCSH
      K=LB+M
248 SMLC(K)=T(M)+SMLC(K)
249 IF(NCFLP.EQ.1) GO TO 251

```

C
C

```

      DO 262 M=1, MXCSH
      K=LH0+M
262 T(M)=SMLDSV(K)
      CALL MLCC2(R,T,NCSH)
      DO 263 M=1, MXCSH
      K=LB+M
263 SMLD(K)=T(M)+SMLD(K)
251 IF(NCCT.EQ.1) GO TO 250

```

C
C

```

      DO 272 M=1, MXCSH
      K=LH0+M
272 T(M)=SMLRSV(K)
      CALL MLCC2(R,T,NCSH)
      DO 273 M=1, MXCSH
      K=LB+M
273 SMLB(K)=T(M)+SMLB(K)
250 CONTINUE
      IF(NQ.LE.1) GO TO 300
      IF(NBACK.EQ.2) GO TO 300
      NHARM=NHARM
      NBACK=2
      NSHFT=NFP1
      GO TO 217
300 CONTINUE
305 CONTINUE
      IF(MODE.EQ.1) GO TO 350

```

```

IF(MFLEX.EQ.0)GO TO 481
IF(MCUN.EQ.0)GO TO 501
IF(IS.NE.NCON)GO TO 500
NCFEA=2
NCFLP=2
NCCT=2
IF(MCTV.GT.0) GO TO 170
DO 482 I=1,MXFAM
K1=(I-1)*12+1
K3=K1+2
K5=K3+2
K6=K5+1
K9=K6+3
K0=K9+1
CTR(I)=CX(1)*H(K1)+CX(2)*H(K5)-CX(3)*H(K9)
1+CX(4)*H(K3)+CX(5)*H(K0)+CX(6)*H(K6)
FAB(I)=CX(2)*H(K1)+CX(7)*H(K5)-CX(8)*H(K9)
1+CX(9)*H(K3)+CX(10)*H(K0)+CX(11)*H(K6)
FLH(I)=CX(3)*H(K1)+CX(8)*H(K5)-CX(12)*H(K9)
1+CX(13)*H(K3)+CX(14)*H(K0)+CX(15)*H(K6)
482 CONTINUE
GO TO 501
170 NCLEL=2
NCACF=2
NCACG=2
DO 172 I=1,MXFAB
K1=(I-1)*12+1
K3=K1+2
K5=K3+2
K6=K5+1
K9=K6+3
K0=K9+1
CTR(I)=CX(40)*H(K1)+CX(43)*H(K5)-CX(46)*H(K9)
1+CX(44)*H(K3)+CX(52)*H(K0)+CX(55)*H(K6)
FAB(I)=CX(41)*H(K1)+CX(44)*H(K5)-CX(47)*H(K9)
1+CX(50)*H(K3)+CX(53)*H(K0)+CX(56)*H(K6)
FLH(I)=CX(3)*H(K1)+CX(8)*H(K5)-CX(12)*H(K9)
1+CX(13)*H(K3)+CX(14)*H(K0)+CX(15)*H(K6)
FSH(I)=CX(42)*H(K1)+CX(45)*H(K5)-CX(48)*H(K9)
1+CX(51)*H(K3)+CX(54)*H(K0)+CX(57)*H(K6)
172 CONTINUE
GO TO 501
481 IF(IS.NE.NLEL) GO TO 483
NCLEL=2
DO 314 I=1,MXFAM
K=(I-1)*12+7
314 FSB(I)=B(K)
IF(IS.LE.NFEA) GO TO 316
DO 315 I=1,MFASH
K=(I-1)*12+7

```

```

315 FSH2(I)=SMLC(K)
316 IF(IS,LE,NCT) GO TO 318
    DO 317 I=1,MFASH
        K=(I-1)*12+7
317 FSH1(I)=SMLH(K)
318 IF(IS,LE,NFLAP) GO TO 483
    DO 319 I=1,MFASH
        K=(I-1)*12+7
319 FSH3(I)=SMLD(K)
C
483 IF(IS,NE,NFEA) GO TO 326
    NCFEA=2
    DO 320 I=1,MXFASH
        K=(I-1)*12+4
        KA=K+3
320 FASH(I)=H(K)+H(KA)*PLC
    IF(IS,LE,NLEL) GO TO 303
    DO 302 I=1,MFASH
        K=(I-1)*12+4
        KA=K+3
302 FASH4(I)=SMLE(K)+SMLE(KA)*PLC
C
303 IF(IS,LT,NCT) GO TO 312
    DO 324 I=1,MFASH
        K=(I-1)*12+4
        KA=K+3
324 FASH1(I)=SMLH(K)+SMLH(KA)*PLC
C
312 IF(IS,LE,NFLAP) GO TO 326
    DO 327 I=1,MFASH
        K=(I-1)*12+4
        KA=K+3
327 FASH3(I)=SMLD(K)+SMLD(KA)*PLC
C
C
326 IF(IS,NE,NCT) GO TO 330
    NCCT=2
    DO 329 I=1,MXFASH
        K=(I-1)*12+3
329 CTH(I)=h(K)
    DO 349 I=1,MFASH
        K=(I-1)*12+4
349 CTH1(I)=SMLH(K)
    IF(IS,LE,NLEL) GO TO 306
    DO 304 I=1,MFASH
        K=(I-1)*12+3
304 CTH4(I)=SMLE(K)
C
306 IF(IS,LE,NFEA) GO TO 331
    DO 333 I=1,MFASH

```

```

      K=(I-1)*12+3
333 CTH2(I)=SMLC(K)
C
331 IF(IS.LT.NFLAP) GO TO 330
    DO 337 I=1,MFASH
      K=(I-1)*12+3
337 CTH3(I)=SMLD(K)
C
C
330 IF(IS.NE.NFLAP) GO TO 500
    NCFLP=2
    DO 336 I=1,MXFAR
      K=(I-1)*12+11
      KA=K-7
336 FLB(I)=H(K)+H(KA)*PFC
    IF(IS.LE.NLEL) GO TO 308
    DO 307 I=1,MFASH
      K=(I-1)*12+11
      KA=KA-7
307 FLB4(I)=SMLE(K)+SMLE(KA)*PFC
C
308 IF(IS.LE.NFEA) GO TO 345
    DO 344 I=1,MFASH
      K=(I-1)*12+11
      KA=K-7
344 FLB2(I)=SMLC(K)+SMLC(KA)*PFC
C
345 IF(IS.LT.NCT) GO TO 500
    DO 347 I=1,MFASB
      K=(I-1)*12+11
      KA=K-7
347 FLB1(I)=SMLB(K)+SMLB(KA)*PFC
    GO TO 501
C
C
C
350 IF(MFLEX.EQ.0) GO TO 484
    IF(MCON.EQ.0) GO TO 487
    IF(IS.NE.NCON) GO TO 487
    NCFEA=2
    NCFLP=2
    NCCT=2
    MCT=1
    MFEA=1
    MFLAP=1
    MLEL=0
    IB=1
    IC=1
    ID=1
    IF(MCTY.EQ.0) GO TO 487

```



```

MLEL=1
MCACF=1
MCACG=1
IE=1
IFF=1
IGG=1
NLEFL=2
NCACF=2
NCACG=2
GO TO 487
484 IF (IS,NE,NCT) GO TO 485
NCT=2
IH=1
485 IF (IS,NE,NFEA) GO TO 486
NCFEA=2
IC=1
486 IF (IS,NE,NFLAP) GO TO 488
N(FLP)=2
IU=1
488 IF (IS,NE,NLEL) GO TO 487
NLEL=2
IE=1

```

C
C
C

```

487 CONTINUE
J2=MXCSH
J3=MXCPM
DO 502 MS=1,NH
J1=NFUS+(MS-1)*NRBD+MXT2P1
DO 502 I=1,MXSMI
DO 351 IR=1,12
351 SHAPE(IR)=DCMPLX(0.00,0.00)
DO 356 J=1,MXSMI
JM1=J-1
IM1=I-1
IJ1=JM1*NEHC+J3*IM1
IJ2=JM1*NESHC+J2*IM1
IK1=JM1*NRIFC+J1
DO 356 IR=1,12
K=IJ1+IR
L=IJ2+IR
HLL=DCMPLX(0.00,0.00)
DO 371 IT=1,6
INB=K+12*(IT-1)
INL=IK1+IT
BLL=BLL+H(INB)*ALM(INL)
371 HLL=HLL+H(INB)*EPS(INL)
INSC=L
IF(IGG,EQ,0) GO TO 174

```

```

INL=IK1+NRBD
SMLGL=SMLG(INSC)*ALM(INL)
SMLGL=SMLGL+SMLG(INSC)*EPS(INL)
174 IF(IFF,EQ,0) GO TO 176
INL=IK1+NRBD=MCACG
SMLFL=SMLF(INSC)*ALM(INL)
SMLFL=SMLFL+SMLF(INSC)*EPS(INL)
176 IF(IE,EQ,0) GO TO 380
INL=IK1+NRBD=MCACG=MCACF
SMLEL=SMLE(INSC)*ALM(INL)
SMLEL=SMLEL+SMLE(INSC)*EPS(INL)
380 IF(IB,EQ,0) GO TO 381
INL=IK1+NRBD=MFEA=MFLAP=MLEL=MCACG=MCACF
SMLBL=SMLB(INSC)*ALM(INL)
SMLBL=SMLBL+SMLB(INSC)*EPS(INL)
381 IF(IC,EQ,0) GO TO 382
INL=IK1+NRBD=MFLAP=MLEL=MCACG=MCACF
SMLCL=SMLC(INSC)*ALM(INL)
SMLCL=SMLCL+SMLC(INSC)*EPS(INL)
382 IF(ID,EQ,0) GO TO 383
INL=IK1+NRBD=MLEL=MCACG=MCACF
SMLDL=SMLD(INSC)*ALM(INL)
SMLDL=SMLDL+SMLD(INSC)*EPS(INL)
383 SHAPE(IR)=SHAPE(IR)+BLL=IB*SMLBL=IC*SMLCL=ID*SMLDL=IE*SMLEL
1=IFF*SMLFL=IGG*SMLGL
356 CONTINUE
UBX(MS,IS,I)=SHAPE(1)
UBY(MS,IS,I)=SHAPE(5)
UBZ(MS,IS,I)=-SHAPE(9)
PHIX(MS,IS,I)=SHAPE(3)
PHIY(MS,IS,I)=SHAPE(10)
PHIZ(MS,IS,I)=SHAPE(6)
AN(MS,IS,I)=SHAPE(2)
VY(MS,IS,I)=-SHAPE(8)
VZ(MS,IS,I)=SHAPE(12)
TEA(MS,IS,I)=SHAPE(4)
AMY(MS,IS,I)=SHAPE(11)
AMZ(MS,IS,I)=SHAPE(7)

```

C

```

502 CONTINUE
500 CONTINUE
501 CONTINUE
IF(IWRITE,EQ,0) GO TO 971
IF(NSP,EQ,0) GO TO 1051
WRITE(6,1052)
DO 1053 IXX=1, MXSMI
IXM2=IXX-NHC=1
IXM2=-IXM2
WRITE(6,1055) IXM2
DO 1053 IYY=1, MXT2P1

```

```

      INL=(IXX=1)*NRIFC+IYY
      IEP=IYY-MAXN=1
      WSW=EPS(INL)+ALM(INL)
      WRITE(6,1054)IEP,WSW
1053 CONTINUE
1051 CONTINUE
1052 FORMAT(/55X,'SWASHPLATE DEFLECTIONS'/)
1054 FORMAT(/55X,'W(',I2,')=' ,2E12.5)
1055 FORMAT(/55X,I2,' PER REV.')
```

```

      DO 7000 IX=1,MXSMI
      DO 7000 MS=1,NB
      IXM2=IX-NHC-1
      IXM2=IXM2
      WRITE(6,1001) MS,IXM2
      WRITE(6,966)
      IJ=1
968 DO 601 J=1,NBSEC
      IF(J.EQ.1) SPACE=BTIPS
      IF(J.EQ.NBSEC) SPACE=BMUR
      WRITE(6,901) SPACE,J,UBX(MS,J,IX),UBY(MS,J,IX),UBZ(MS,J,IX)
      IF(IJ.NE.2) GO TO 390
      IF(IPU.EQ.1)WRITE(7,1111)J,UBX(MS,J,IX),UBY(MS,J,IX),UBZ(MS,J,IX)
390 CONTINUE
      SPACE=BLANKS
      IF(IJ.EQ.3) GO TO 601
      QXJ=UBX(MS,J,IX)
      QYJ=UBY(MS,J,IX)
      QZJ=UBZ(MS,J,IX)
      IF(IJ.EQ.2) GO TO 603
      ISM1=(J-1)*NSY
      DO 602 K=15,30
602 Y(K)=SD(ISM1+K)
C
C
C
      UBX(MS,J,IX)=QXJ*Y(19)+QYJ*Y(24)+QZJ*Y(26)
      UBY(MS,J,IX)=QXJ*Y(23)+QYJ*Y(25)+QZJ*Y(27)
      UBZ(MS,J,IX)=QXJ*(-Y(15))+QYJ*Y(20)+QZJ*Y(21)
      GO TO 601
603 CALL POLAR
      UBX(MS,J,IX)=QXJ
      UBY(MS,J,IX)=QYJ
      UBZ(MS,J,IX)=QZJ
601 CONTINUE
      IF(IJ.GT.1) GO TO 604
      WRITE(6,950)
      GO TO 961
604 WRITE(6,963)
961 DO 951 J=1,NBSEC
      IF(J.EQ.1)SPACE=BTIPS
```

```

IF(J, EQ, NBSEC) SPACE=BMHUB
WRITE(6, 901) SPACE, J, PHIX(MS, J, IX), PHIY(MS, J, IX), PHIZ(MS, J, IX)
IF(IJ, NE, 2) GO TO 391
IF(IPU, EQ, 1) WRITE(7, 1111) J, PHIX(MS, J, IX), PHIY(MS, J, IX), PHIZ(MS, J, I
IX)
391 CONTINUE
SPACE=BLANKS
IF(IJ, EQ, 3) GO TO 951
QXJ=PHIX(MS, J, IX)
QYJ=PHIY(MS, J, IX)
QZJ=PHIZ(MS, J, IX)
IF(IJ, EQ, 2) GO TO 810
ISM1=(J-1)*NSY
DU 800 K=15, 30
800 Y(K)=SD(ISM1+K)
C
C
C
PHIX(MS, J, IX)=QXJ*Y(19)+QYJ*Y(24)+QZJ*Y(26)
PHIY(MS, J, IX)=QXJ*Y(23)+QYJ*Y(25)+QZJ*Y(27)
PHIZ(MS, J, IX)=QXJ*(-Y(15))+QYJ*Y(20)+QZJ*Y(21)
GO TO 951
810 CALL POLAR
PHIX(MS, J, IX)=QXJ
PHIY(MS, J, IX)=QYJ
PHIZ(MS, J, IX)=QZJ
951 CONTINUE
IF(IJ, GT, 1) GO TO 958
WRITE(6, 955)
GO TO 956
958 WRITE(6, 964)
956 DU 957 J=1, NBSEC
IF(J, EQ, 1) SPACE=BTIPS
IF(J, EQ, NBSEC) SPACE=BMHUB
WRITE(6, 901) SPACE, J, AN(MS, J, IX), VY(MS, J, IX), VZ(MS, J, IX)
IF(IJ, NE, 2) GO TO 392
IF(IPU, EQ, 1) WRITE(7, 1111) J, AN(MS, J, IX), VY(MS, J, IX), VZ(MS, J, IX)
392 CONTINUE
SPACE=BLANKS
IF(IJ, EQ, 3) GO TO 957
QXJ=AN(MS, J, IX)
QYJ=VY(MS, J, IX)
QZJ=VZ(MS, J, IX)
IF(IJ, EQ, 2) GO TO 820
ISM1=(J-1)*NSY
DU 801 K=15, 30
801 Y(K)=SD(ISM1+K)
C
C
C

```

```

AN(MS,J,IX) =QXJ*Y(19)+QYJ*Y(24)+QZJ*Y(26)
VY(MS,J,IX) =QXJ*Y(23)+QYJ*Y(25)+QZJ*Y(27)
VZ(MS,J,IX) =QXJ*(-Y(15))+QYJ*Y(20)+QZJ*Y(21)
GO TO 957
820 CALL POLAR
AN(MS,J,IX)=QXJ
VY(MS,J,IX)=QYJ
VZ(MS,J,IX)=QZJ
957 CONTINUE
IF(IJ,GT.1) GO TO 970
WRITE(6,959)
GO TO 969
970 WRITE(6,965)
969 DO 960 J=1,NBSEC
IF(J,EQ.1)SPACE=RTIPS
IF(J,EQ,NBSEC)SPACE=BHUB
WRITE(6,901)SPACE,J,TEA(MS,J,IX),AMY(MS,J,IX),AMZ(MS,J,IX)
IF(IJ,NE.2) GO TO 393
IF(IPU,EQ.1)WRITE(7,1111)J,TEA(MS,J,IX),AMY(MS,J,IX),AMZ(MS,J,IX)
393 CONTINUE
SPACE=BLANKS
IF(IJ,EQ.3) GO TO 960
QXJ=TEA(MS,J,IX)
QZJ=AMZ(MS,J,IX)
QYJ=AMY(MS,J,IX)
IF(IJ,EQ.2) GO TO 840
ISM1=(J-1)*NSY
DO 802 K=15,30
802 Y(K)=SD(ISM1+K)
TEA(MS,J,IX) =QXJ*Y(19)+QYJ*Y(24)+QZJ*Y(26)
AMY(MS,J,IX) =QXJ*Y(23)+QYJ*Y(25)+QZJ*Y(27)
AMZ(MS,J,IX) =QXJ*(-Y(15))+QYJ*Y(20)+QZJ*Y(21)
GO TO 960
840 CALL POLAR
TEA(MS,J,IX)=QXJ
AMY(MS,J,IX)=QYJ
AMZ(MS,J,IX)=QZJ
960 CONTINUE
IJ=IJ+1
9001 IF(NPD,EQ.0)GO TO 7000
IF(IJ,GT.3) GO TO 7000
IF(IJ,EQ.3) GO TO 850
WRITE(6,1002)
GO TO 830
850 WRITE(6,983)
830 WRITE(6,962)
GO TO 968

```

C
C
C

C
C
C
C
C
C
C
C
C
C
C
C

```
7000 CONTINUE
      I=NHC+1
      IF(NPD.GT.0) GO TO 516
      UBYM=CABS(UBY(1,1,I))
      UBZM=CABS(UBZ(1,1,I))
      PHIXM=CABS(PHIX(1,1,I))
      DO 511 IS=2,NBSEC
      UBYT=CABS(UBY(1,IS,I))
      UBZT=CABS(UBZ(1,IS,I))
      PHIXT=CABS(PHIX(1,IS,I))
      UBYM=AMAX1(UBYM,UBYT)
      UBZM=AMAX1(UBZM,UBZT)
511  PHIXM=AMAX1(PHIXM,PHIXT)
      GO TO 520
516  UBYM=UBY(1,1,I)
      UBZM=UBZ(1,1,I)
      PHIXM=PHIX(1,1,I)
      DO 512 IS=2,NBSEC
      UBYT=UBY(1,IS,I)
      UBZT=UBZ(1,IS,I)
      PHIXT=PHIX(1,IS,I)
      UBYM=AMAX1(UBYM,UBYT)
      UBZM=AMAX1(UBZM,UBZT)
512  PHIXM=AMAX1(PHIXM,PHIXT)
520  PHIXM=PHIXM*.5.73
      IF(UBYM.GT.UBZM) GO TO 524
      NBGMDS=2
      IF(PHIXM.GT.UBZM) NBGMDS=3
      GO TO 971
524  NBGMDS=1
      IF(PHIXM.GT.UBYM) NBGMDS=3
971  NCT=ICT
      NFEA=IFEA
      NFLAP=IFLA
      NLEL=ILEL
      RETURN
901  FORMAT(3X,A4,3X,I2,3(9X,2(1PE12,4)))
950  FORMAT(//,T8,'SECTION',T28,'PHI X, TWIST',T57,'PHI y, FLAPWISE SLO
1PE1,T89,'PHI Z, CHORDWISE SLOPE',//,T28,'(+, NOSE UP)',T60,'(+, FLA
2P DOWN)',T93,'(+, BLADE LEAD)',//)
```

```

955 FORMAT( //,T8,'SECTION',T27,'N, AXIAL FORCE',T56,'V Y, CHORDWI
18E SHEAR',T91,'V Z, FLAPWISE SHEAR',/,T24,'(+, RADIAL OUTBOARD)',T
259,'(+, TOWARD L.E.)',T95,'(+, UPWARD)',//)
959 FORMAT(//,T8,'SECTION',T30,'T, TORQUE',T57,'M Y, FLAPWISE MOMENT',
1T90,'M Z, CHORDWISE MOMENT',/,T28,'(+, NOSE UP)',T55,'(+, TENSION
2UPPER FIBERS)',T90,'(+, TENSION T.E.)',//)
1002 FORMAT(1H1,//,T49,'DISK PLANE AXIS SYSTEM',//)
962 FORMAT(T8,'SECTION',T25,'UBX, RADIAL DEFL.',
1 T58,'UBY, INPLANE DEFL.',T92,'UBZ, VERT. DEFL.',/
2,T28,'(+, OUTBOARD)',T58,'(+, ROTATION DIR.)',T95,'(+, UPWARD)',//
3)
964 FORMAT( //,T8,'SECTION',T26,'NB, RADIAL FORCE',T58,'VBY, INPLA
1NE SHEAR',T92,'VBZ, VERT. SHEAR',/,T28,'(+, OUTBOARD)',T59,'(+, DI
2R, UP ROT.)',T94,'(+, UPWARD)',//)
965 FORMAT(//,T8,'SECTION',T29,'TB, TORQUE',T57,'MBY, VERTICAL MOMENT'
1,T91,'MBZ, INPLANE MOMENT',/,T28,'(+, NOSE UP)',T55,'(+, TENSION U
2PPER FIBERS)',T92,'(+, TENSION T.E.)',//)
966 FORMAT(T8,'SECTION',T25,'UX, RADIAL DEFL.',T58,'UY, INPLANE DEFL.'
1,T92,'UZ, VERT. DEFL.',/,T28,'(+, OUTBOARD)',T58,
2'(+, ROTATION DIR.)',T95,'(+, UPWARD)',//)
963 FORMAT(//,T8,'SECTION',T28,'PHIBX, TWIST',T54,'PHIBY, VERT. BENDIN
1G SLOPE',T87,'PHIBZ, INPLANE BEND. SLOPE',/,T26,'(+, L.E. UPWARD)'
2,T59,'(+, TIP DOWNWARD)',T90,'(+, TIP DIR. OF ROT.)',//)
983 FORMAT(1H1,//,T39,'DISK PLANE AXIS SYSTEM = POLAR COORDINATES',/,
1T47,'AMPLITUDE AND PHASE ANGLE',//)
1001 FORMAT( //,T45,'BLADE',I2,I4,' PER REV SHAPE VECTOR(LOCAL AXIS
18YS.)',//)
1111 FORMAT(I5,3(1X,2(1PE12.4)))
END

```

```

SUBROUTINE SARRAY
INTEGER EM(6),EE(6)
REAL*8 RREAL(144)
COMPLEX UBX(25,3),UBY(25,3),UBZ(25,3),PHIX(25,3),PHIY(25,3),
1PHIZ(25,3),AN(25,3),VY(25,3),VZ(25,3),TEA(25,3),AMY(25,3),AMZ(25,3
2),QXJ,QYJ,QZJ
COMPLEX*16 CMS,CM1,CS1,CS2
COMPLEX*16 EPS(63),ALM(63),S(216),R(144),T(72),QBAR(12),SLL
DIMENSION Y(100),SD(4000)
COMMON/CALM/ALM
COMMON/EP5A/EP5
COMMON/SS1/S
COMMON/ARR/RREAL
COMMON/ARI/R
COMMON/ROTF/OM1,OM2,OMT
COMMON/FREF/CMS,CM1,CS1,CS2
COMMON/SUB/Y
COMMON/SA1N/SD
COMMON/INTER/NSY,NBSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
1MFLEX,MCON,MAER,MFUS,NBC,NFLAP,NFEA,NCT,NCON,NFFB,
2NAS,NHC,NVI,NSP,MAXN,NES,MSC,NEGN,IPCT,NIT,MER,NORM,
3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NC8B,
4NFP1,MXSMI,MXT2P1,MXKQ,MXCPL,MXC8B,MXCPM,MXCPK,MX8MB,
5NEBC,NE8BC,MFA8B,MXFAB,NFUS,NR8D,NRIFC,MXQ,NEIFC,
6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MODE
COMMON/NLEAD/MLEL,NLEL,MCTY
COMMON/QVTEM/QXJ,QYJ,QZJ
COMMON/IPCH/IPU
DATA EM/2,16,31,44,59,72/
DATA EE/1,15,29,42,57,70/
DATA BLANKS/' ',' ',BTIPS/'TIP '/,BHUB/'HUB '/
IF(MFUS.EQ.0) GO TO 971
MSCPK=MXCPM*MXSMI
IWRITE=0
IF(MODE.EQ.1) IWRITE=1
IL=0
SPACE=BLANKS
DO 22 L=1,MSCPK
22 S(L)=DCMPLX(0.00,0.00)
DO 30 L=1,MX8MI
LM1=(L-1)*MXCPM
DO 30 M=1,6
K=LM1+EM(M)
IF(NFFB.EQ.1) K=LM1+EE(M)
30 S(K)=UCMPLX(1.00,0.00)
DO 50 IS=1,NFSEC
ISM1=(NBSEC+IS-1)*NSY
DO 50 L=1,NSY
M=ISM1+L
50 Y(L)=SD(M)

```



```

        IF(Y(5),EQ,0.) GO TO 80
        CALL RIGID
        LGU=1
    60  DO 75 I=1, MXSMI
        LM1=(I-1)*MXCPM
        DO 65 M=1, MXCPM
        K=LM1+M
    65  T(M)=S(K)
        CALL MLRC2(RREAL,T,NCOLS)
        DO 691 M=1, MXCPM
        K=LM1+M
    691  S(K)=T(M)
    75  CONTINUE
        IF(LGU,EQ,2) GO TO 100
        IF(LGU,EQ,3) GO TO 120
    80  IF(Y(2),EQ,0.) GO TO 100
        CALL BEND
        LGU=2
        GO TO 60
    100 IF(Y(4),EQ,0.) GO TO 120
        CALL ELAST
        LGU = 3
        GO TO 60
    120 DO 200 I=1, MXSMI
        IM1=(I-1)*12
        LM1=IM1*NCOLS
        KSM1=I-NFP1
        NST=KSM1
        CS1=CHS=CM1*NST*IM1
        CS2=CS1*CS1
        IF(Y(6),EQ,0.) GO TO 148
        CALL STIFF
        LGU=1
    130 CONTINUE
        DO 136 M=1, MXCPM
        K=LM1+M
    136 T(M)=S(K)
        CALL MLCC2(R,T,NCOLS)
        DO 140 M=1, MXCPM
        K=LM1+M
    140 S(K)=T(M)
        IF(LGU,EQ,2) GO TO 175
        IF(LGU,EQ,3) GO TO 200
    148 IF(Y(1),EQ,0.) GO TO 175
        CALL FMASS
        LGU=2
        GO TO 130
    175 IF(Y(3),EQ,0.) GO TO 200
        IL=IL+1
        CALL FUARD(IL)

```

```

LGO=3
GO TO 130
200 CONTINUE
IF(MODE.EQ.0) GO TO 500
L1=MXCPM
I2=NRIFC
J1=MXI2P1
DO 499 I=1, MXSMI
DO 250 L=1, I2
IM1=I-1
K=L1*IM1+L
SLL=DCMPLX(0,00,0,00)
DO 252 IT=1,6
INB=K+I2*(IT-1)
INL=J1+I2*IM1+IT
SLL=SLL+S(INB)*ALM(INL)
252 SLL=SLL+S(INB)*EPS(INL)
250 QBAR(L)=SLL
UBX(IS,I)=QBAR(1)
UBY(IS,I)=QBAR(5)
UBZ(IS,I)=QBAR(9)
PHIX(IS,I)=QBAR(3)
PHIY(IS,I)=QBAR(10)
PHIZ(IS,I)=QBAR(6)
AN(IS,I)=QBAR(2)
VY(IS,I)=QBAR(8)
VZ(IS,I)=QBAR(12)
TEA(IS,I)=QBAR(4)
AMY(IS,I)=QBAR(11)
AMZ(IS,I)=QBAR(7)
C
499 CONTINUE
500 CONTINUE
C
IF(IWRITE.EQ.0) GO TO 971
DO 700 IX=1, MXSMI
IXM2=IX+NHG+1
WRITE(6,916)IXM2
WRITE(6,917)
IJ=1
96A DO 503 J=1, NFSEC
IF(J.EQ.1) SPACE=BTIPS
IF(J.EQ.NFSEC) SPACE=HMUR
WRITE(6,901) SPACE,J,UBX(J,IX),UBY(J,IX),UBZ(J,IX)
IF(IPU.EQ.1)WRITE(7,1111)J,UBX(J,IX),UBY(J,IX),UBZ(J,IX)
SPACE=BLANKS
IF(IJ.EQ.3) GO TO 503
QXJ=UBX(J,IX)
QYJ=UBY(J,IX)
QZJ=UBZ(J,IX)

```

```

IF(IJ,EQ.2) GO TO 702
ISM1=(NBSEC+J-1)*NSY
DO 504 K=15,30
504 Y(K)=SD(ISM1+K)
C
C
C
UBX(J,IX) =QXJ*Y(19)+QYJ*Y(24)+QZJ*Y(26)
UBY(J,IX) =QXJ*Y(23)+QYJ*Y(25)+QZJ*Y(27)
UBZ(J,IX) =QXJ*(-Y(15))+QYJ*Y(20)+QZJ*Y(21)
GO TO 503
702 CALL POLAR
UBX(J,IX)=QXJ
UBY(J,IX)=QYJ
UBZ(J,IX)=QZJ
503 CONTINUE
IF(IJ,GT.1) GO TO 703
WRITE(6,950)
GO TO 961
703 WRITE(6,963)
961 DO 951 J=1,NFSEC
IF(J,EQ.1)SPACE=RTIPS
IF(J,EQ,NFSEC)SPACE=RHUB
WRITE(6,901)SPACE,J,PHIX(J,IX),PHIY(J,IX),PHIZ(J,IX)
IF(IPU,EQ.1)WRITE(7,1111)J,PHIX(J,IX),PHIY(J,IX),PHIZ(J,IX)
SPACE=BLANKS
IF(IJ,EQ.3) GO TO 951
QXJ=PHIX(J,IX)
QYJ=PHIY(J,IX)
QZJ=PHIZ(J,IX)
IF(IJ,EQ.2) GO TO 704
ISM1=(NBSEC+J-1)*NSY
DO 800 K=15,30
800 Y(K)=SD(ISM1+K)
C
C
C
PHIX(J,IX)=QXJ*Y(19)+QYJ*Y(24)+QZJ*Y(26)
PHIY(J,IX)=QXJ*Y(23)+QYJ*Y(25)+QZJ*Y(27)
PHIZ(J,IX)=QXJ*(-Y(15))+QYJ*Y(20)+QZJ*Y(21)
GO TO 951
704 CALL POLAR
PHIX(J,IX)=QXJ
PHIY(J,IX)=QYJ
PHIZ(J,IX)=QZJ
951 CONTINUE
IF(IJ,GT.1) GO TO 958
WRITE(6,955)
GO TO 956
958 WRITE(6,964)

```

```

956 DO 957 J=1,NFSEC
      IF(J.EQ.1)SPACE=RTIPS
      IF(J.EQ.NFSEC)SPACE=RHUB
      WRITE(6,901)SPACE,J,AN(J,IX),VY(J,IX),VZ(J,IX)
      IF(IPU.EQ.1)WRITE(7,1111)J,AN(J,IX),VY(J,IX),VZ(J,IX)
      SPACE=HLANKS
      IF(IJ.EQ.3) GO TO 957
      QXJ=AN(J,IX)
      QYJ=VY(J,IX)
      QZJ=VZ(J,IX)
      IF(IJ.EQ.2) GO TO 705
      ISM1=(NBSEC+J-1)*NSY
      DO 801 K=15,30
801  Y(K)=SD(ISM1+K)

```

C
C
C

```

      AN(J,IX) =QXJ*Y(19)+QYJ*Y(24)+QZJ*Y(26)
      VY(J,IX) =QXJ*Y(23)+QYJ*Y(25)+QZJ*Y(27)
      VZ(J,IX) =QXJ*(-Y(15))+QYJ*Y(20)+QZJ*Y(21)
      GO TO 957
705 CALL POLAR
      AN(J,IX)=QXJ
      VY(J,IX)=QYJ
      VZ(J,IX)=QZJ
957 CONTINUE
      IF(IJ.GT.1) GO TO 970
      WRITE(6,959)
      GO TO 969
970 WRITE(6,965)
969 DO 960 J=1,NFSEC
      IF(J.EQ.1)SPACE=BTIPS
      IF(J.EQ.NFSEC)SPACE=RHUB
      WRITE(6,901)SPACE,J,TEA(J,IX),AMY(J,IX),AMZ(J,IX)
      IF(IPU.EQ.1)WRITE(7,1111)J,TEA(J,IX),AMY(J,IX),AMZ(J,IX)
      SPACE=BLANKS
      IF(IJ.EQ.3) GO TO 960
      QXJ=TEA(J,IX)
      QYJ=AMY(J,IX)
      QZJ=AMZ(J,IX)
      IF(IJ.EQ.2) GO TO 706
      ISM1=(NBSEC+J-1)*NSY
      DO 802 K=15,30
802  Y(K)=SD(ISM1+K)
      TEA(J,IX) =QXJ*Y(19)+QYJ*Y(24)+QZJ*Y(26)
      AMY(J,IX) =QXJ*Y(23)+QYJ*Y(25)+QZJ*Y(27)
      AMZ(J,IX) =QXJ*(-Y(15))+QYJ*Y(20)+QZJ*Y(21)
      GO TO 960
706 CALL POLAR
      TEA(J,IX)=QXJ

```

```

      AMY(J,IX)=QYJ
      AMZ(J,IX)=QZJ
960  CONTINUE
      IJ=IJ+1
9001 IF(NPD,EQ,0)GO TO 7000
      IF(IJ,GT,3) GO TO 7000
      IF(IJ,EQ,3) GO TO 707
      WRITE(6,919)
      GO TO 708
907  WRITE(6,920)
908  WRITE(6,962)
      GO TO 968

```

C
C
C
C
C
C
C
C
C
C
C
C

```

7000 CONTINUE
971  RETURN
901  FORMAT(3X,A4,3X,I2,3(9X,2(1PE12.4)))
950  FORMAT(//,T8,'SECTION',T28,'PHI X, TWIST',T57,'PHI Y, INPLANE SLO
      1PE',T89,'PHI Z, VERTICAL SLOPE',//,T28,'(+, TOP LEFT)',T60,'(+, VEC
      2T, UP)',T93,'(+, VECT. LEFT)',//)
955  FORMAT( //,T8,'SECTION',T27,'N, AXIAL FORCE',T56,'V Y, VERTICA
      1L SHEAR',T91,'V Z, INPLANE SHEAR',//,T24,'(+, RADIAL TO TAIL)',T
      259,'(+, UPWARDS)',T95,'(+, L.H.S.)',//)
959  FORMAT(//,T8,'SECTION',T30,'T, TORQUE',T57,'M Y, INPLANE MOMENT',
      1T90,'M Z, VERTICAL MOMENT',//,T28,'(+, TOP LEFT)',T55,'(+, TENSION
      2LHS FIBERS)',T90,'(+, TENSION LOWER FIBS.)',//)
962  FORMAT(//,T8,'SECTION',T25,'UBX, AXIAL DEFL.',
      1T58,'UBY, VERTICAL DEF.',T92,'UBZ, INPL. DEFL.',//,T28,
      2'(+, TO TAIL)',T58,'(+, UPWARD DIR.)',T95,'(+, L.H.S.)',//)
C
964  FORMAT( //,T8,'SECTION',T26,'NB, AXIAL FORCE',T58,'VBY, VERTI
      1C, SHEAR',T92,'VRZ, INPL. SHEAR',//,T28,'(+, TO TAIL)',T59,'(+, UP
      2WARD)',T94,'(+, TO LEFT)',//)
965  FORMAT(//,T8,'SECTION',T29,'TB, TORQUE',T57,'MBY, INPLANE MOMENT'
      1,T91,'MBZ, VERTIC. MOMENT',//,T28,'(+, TOP LEFT)',T55,'(+, TENSION L
      2,H,S,FIBERS)',T92,'(+, TENSION LOWER FIBS.)',//)
963  FORMAT(//,T8,'SECTION',T28,'PHIBX, TWIST',T54,'PHIBY, INPL. BENDIN
      1G SLOPE',T87,'PHIBZ, VERTIC. BEND. SLOPE',//,T26,'(+, NOSE UPWARD)'
      2,T59,'(+, VECT. UPWARD)',T90,'(+, VECTOR LEFT SIDE)',//)

```

```

916 FORMAT ( 1H1,35X,I4, ' PER REV SHAPE VECTORS ON THE FUSELA. STRUCTU
    1RE(LOCAL AX)'//)
917 FORMAT(/,T8,'SECTION',T25,'UX, AXIAL DEFL.',
    1          T58,'UY, VERTIC. DEFL.',T92,'UZ, INPL. DEFL.',/
    2,T28,'(+, TO TAIL )',T58,'(+, UPWARD DIR.)',T95,'(+, L.H.S.)',//
    3)
919 FORMAT(1H1,/,/,T45,'FUSELAGE REFERECE AXIS SYSTEM',/)
920 FORMAT(1H1,/,/,T35,'FUSELAGE REFERECE AXIS SYSTEM = POLAR COORDINA
    1TES',/)
900 FORMAT (10(1PE12.4))
1111 FORMAT(I5,3(1X,2(1PE12.4)))
    END

```

```

SUBROUTINE MLRC2(R,T,NCOLS)
REAL*8 R(144)
COMPLEX*16 T(84),THLD(84)
COMPLEX*16 RCDOT
NCR=12*NCOLS
DO 100 N=1,NCOLS
  NN1=(N-1)*12
  DO 100 M=1,12
    MM1=(M-1)*12
    K=NN1+M
    LM=MM1+1
    LN=NN1+1
100 THLD(K)=RCDOT(12,R(LM),T(LN))
  DO 200 I=1,NCR
200 T(I)=THLD(I)
RETURN
END

```

```

SUBROUTINE MLCC2 (R,T,NCOLS)
COMPLEX*16 R(144),T(84),THLD(84)
COMPLEX*16 CDOT
NCR=12*NCOLS
DO 100 N=1,NCOLS
  NN1=(N-1)*12
  DO 100 M=1,12
    MM1=(M-1)*12
    K=NN1+M
    LM=MM1+1
    LN=NN1+1
100 THLD(K)=CDOT(12,R(LM),T(LN))
  DO 200 I=1,NCR
200 T(I)=THLD(I)
  RETURN
END

```



```

SUBROUTINE TKNS(I,J)
COMPLEX*16 TKN(441)
COMMON/TKN1/TKN
COMMON/INTER/NSY,NHSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
1MFLEX,MCON,MAER,MFUS,NHC,NFLAP,NFEA,NCT,NCON,NFFB,
2NAS,NHC,NVI,NSP,MAXN,NES,MSC,NEGN,IPCT,NIT,NER,NORM,
3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NCSB,
4NFP1,MXSMI,MXT2P1,MXKQ,MXCPL,MXCSB,MXCPM,MXCPK,MXSMR,
5NEBC,NESRC,MFASB,MXFAB,NFUS,NRHO,NRIFC,MXQ,NEIFC,
6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MODE
COMMON/NLEAD/MLEI,NLEL,MCTY
DO 5 L=1,MXTKN
5 TKN(L)=DCMPLX(0.D0,0.D0)
IF(MFUS.EQ.0) GO TO 6
CALL SUBMA(I,J)
6 CALL SUBMB(I,J)
CALL SUBMG(I,J)
IF(NH.EQ.1) GO TO 10
CALL SUBMF(I,J)
10 IF(NHC.GT.0)CALL ZTEGI(I,J)
IF(NSP.EQ.0) GO TO 20
CALL SWA(I,J)
CALL SWB(I,J)
20 CONTINUE
RETURN
END

```

```

SUBROUTINE EPSOLN
COMPLEX*16 TKN(441)
COMPLEX*16 FTEMP(21)
COMPLEX*16 EPS(63)
COMPLEX*16 ALM(63)
COMMON/TKN1/TKN
COMMON/EP5A/EP5
COMMON/CALM/ALM
COMMON/INTER/NSY,NBSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
1MFLEX,MCON,MAER,MFUS,NBC,NFLAP,NFEA,NCT,NCON,NFFB,
2NAS,NHC,NVI,NSP,MAXN,NE9,MSC,NEGN,IPCT,NIT,MER,NORM,
3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NCSR,
4NFP1,MXSMI,MXT2P1,MXKU,MYCPL,MYCSB,MYCPM,MYCPK,MXSMB,
5NEBC,NE9HC,MFASB,MYFAH,NFUS,NRBD,NRIFC,MXQ,NEIFC,
6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MODE
COMMON/NLEAD/MLEL,NLEL,MCTY
IREM1=IREM
REWIND 1
WRITE (1) MXSMI,NRIFC,NHC,NORM,IREM1,NEX
DO 2 I=1,MXSMI
DO 3 K=1,NRIFC
3 FTEMP(K)=DCMPLX(0.00,0.00)
DO 4 J=1,MXSMI
CALL TKNS(I,J)
WRITE(1)(TKN(IRITE),IRITE=1,MXTKN)
DO 5 IQ=1,NRIFC
DO 5 IP=1,NRIFC
KP=(IP-1)*NRIFC+IQ
IJ=(J-1)*NRIFC+IP
5 FTEMP(IQ)=FTEMP(IQ)-ALM(IJ)*TKN(KP)
4 CONTINUE
DO 2 L=1,NRIFC
2 EPS(NRIFC*(I-1)+L)=FTEMP(L)
REWIND 1
RETURN
END

```

```

SUBROUTINE HMASS
REAL Y(100)
COMPLEX*16 CMS,CM1,CS1,CS2,TOMS
COMPLEX*16 CSTE
COMPLEX*16 A(144)
COMMON/ARI/A
COMMON/SUB/Y
COMMON/ROTF/OM1,OM2,OMT
COMMON/DAMCO/OMDA
COMMON/PREF/CMS,CM1,CS1,CS2
COMMON/AMAS1/GCTCP
CN1=Y(33)*GCTCP
CSTE=CS2
CS2=CS2+OMDA*CS1
TOMS=OMT*CS1
DO 5 I=1,144
5 A(I)=DCMPLX(0.D0,0.D0)
DO 11 I=1,144,13
11 A(I)=DCMPLX(1.D0,0.D0)
A(13)=Y(8)*(CS2-Y(17))
A(15)=Y(33)*(TOMS*Y(20)-Y(28))
A(17)=Y(8)*(TOMS*Y(21)+Y(29))
A(18)=Y(9)*A(13)
A(21)=Y(8)*(TOMS*Y(20)-Y(28))
A(37)=Y(33)*(TOMS*Y(20)+Y(28))
A(39)=Y(34)*CS2+Y(43)-Y(20)*CN1
A(41)=Y(33)*(TOMS*Y(15)-Y(30))
A(42)=Y(36)*CS1+Y(39)
A(45)=Y(33)*(CS2-Y(31))
A(46)=Y(37)*CS1+Y(45)
A(73)=A(18)
A(75)=Y(36)*CS1+Y(42)
A(77)=Y(9)*A(17)
A(78)=Y(35)*CS2+Y(40)-Y(20)*CN1
A(81)=A(15)
A(82)=Y(38)*CS1+Y(46)
A(85)=Y(8)*(TOMS*Y(21)-Y(29))
A(87)=Y(33)*(TOMS*Y(15)+Y(30))
A(89)=Y(8)*(CS2-Y(32))
A(90)=Y(9)*A(85)
A(93)=Y(8)*(TOMS*Y(15)+Y(30))
A(123)=Y(37)*CS1-Y(44)-Y(15)*CN1
A(126)=Y(38)*CS1+Y(41)+Y(21)*CN1
A(130)=Y(10)*CS2-Y(47)
A(133)=Y(8)*(TOMS*Y(20)+Y(28))
A(135)=A(45)
A(137)=Y(8)*(TOMS*Y(15)-Y(30))
A(138)=A(37)
A(141)=Y(8)*(CS2-Y(31))
CS2=CSTE
RETURN
END

```

```

SUBROUTINE FMASS
REAL Y(100)
COMPLEX*16 W(144)
COMPLEX*16 CMS,CM1,CS1,CS2,WI,WII
COMMON/FREF/CMS,CM1,CS1,CS2
COMMON/ARI/W
COMMON/SIJB/Y
COMMON/FGRAV/FGRA
WI=CS2*Y(8)
WII=Y(9)*WI
CN1=FGRA*Y(33)
DO 10 I=1,144
10 W(I)=DCMPLX(0.00,0.00)
W(13)=WI
W(18)=-WII
W(39)=CS2*Y(34)-CN1*Y(25)
W(45)=-WII
W(73)=-WII
W(78)=CS2*Y(35)-CN1*Y(25)
W(89)=-WI
W(123)=CN1*Y(23)
W(126)=CN1*Y(27)
W(130)=CS2*Y(10)
W(135)=+WII
W(141)=-WI
DO 20 I=1,144,13
20 W(I)=DCMPLX(1.000,0.00)
RETURN
END

```

```

SUBROUTINE STIFF
COMPLEX*16 SK(144)
REAL Y(100)
COMPLEX*16 CMS,CM1,CS1,CS2
COMMON/ARI/SK
COMMON/SUH/Y
COMMON/FREF/CMS,CM1,CS1,CS2
DO 5 I=1,144
5 SK(I)=DCMPLX(0.00,0.00)
IF(Y(100).EQ.0.0)GO TO 10
SK(39)=Y(100)
GO TO 20
10 SK(28)=Y(94)/(1.+Y(97)*CS1)
SK(67)=Y(96)/(1.+Y(99)*CS1)
SK(119)=Y(95)/(1.+Y(98)*CS1)
20 DO 30 I=1,144,13
30 SK(I)=DCMPLX(1.00,0.00)
RETURN
END

```

```

SUBROUTINE BEND
REAL Y(100)
REAL*8 B(144)
COMMON/SUB/Y
COMMON/ARR/B
DO 5 I=1,144
5 B(I)=0.D0
B(1)=Y(60)
B(5)=Y(61)
B(9)=-Y(62)
B(14)=Y(60)
B(20)=-Y(61)
B(24)=Y(62)
B(27)=Y(60)
B(30)=Y(62)
B(34)=Y(61)
B(40)=Y(60)
B(43)=Y(62)
B(47)=Y(61)
B(49)=Y(63)
B(53)=Y(64)
B(57)=-Y(65)
B(63)=Y(66)
B(66)=Y(68)
B(70)=Y(67)
B(76)=Y(66)
B(79)=Y(68)
B(83)=Y(67)
B(86)=-Y(63)
B(92)=Y(64)
B(96)=-Y(65)
B(97)=-Y(66)
B(101)=-Y(67)
B(105)=Y(68)
B(111)=Y(63)
B(114)=Y(65)
B(118)=Y(64)
B(124)=Y(63)
B(127)=Y(65)
B(131)=Y(64)
B(134)=Y(66)
B(140)=-Y(67)
B(144)=Y(68)
RETURN
END

```

```

SUBROUTINE RIGID
REAL Y(100)
REAL*8 R(144)
COMMON/SUH/Y
COMMON/ARR/R
DO 5 L=1,144
5 R(L)=0.00
DO 10 L=1,144,13
10 R(L)=1.00
R(6)=-Y(85)
R(10)=Y(84)
R(44)=-Y(84)
R(48)=-Y(83)
R(51)=-Y(84)
R(54)=Y(82)
R(74)=Y(83)
R(80)=Y(82)
R(99)=-Y(83)
R(106)=Y(82)
R(122)=-Y(84)
R(132)=Y(82)
R(39)=Y(85)
R(42)=Y(86)
R(46)=Y(87)
R(75)=Y(88)
R(78)=Y(89)
R(82)=Y(90)
R(123)=Y(91)
R(126)=Y(92)
R(130)=Y(93)
RETURN
END

```

```

SUBROUTINE ELAST
REAL Y(100)
REAL*8 CX(75)
REAL*8 E(144)
COMMON/SUB/Y
COMMON/CFLEX/CX
COMMON/ARR/E

```

C

```

DO 5 I=1,144
5 E(I)=0.00
DO 8 K=1,144,13
8 E(K)=1.00
IF(Y(7).GT..9)GO TO 10
E(28)=Y(69)
E(30)=+Y(69)*Y(59)
E(34)=-Y(69)*Y(58)
E(42)=-Y(56)*Y(70)+.5*Y(55)*Y(77)*Y(57)*(1.0+Y(73))
E(43)=-Y(56)*Y(71)+.5*Y(55)*Y(77)*Y(57)*Y(74)
E(44)=-Y(56)*Y(72)+.5*Y(55)*Y(77)*Y(57)*Y(71)
E(46)=-Y(76)*Y(55)
E(47)=-Y(77)*Y(55)
E(48)=-Y(78)*Y(55)
E(54)=Y(70)
E(55)=Y(71)
E(56)=Y(72)
E(66)=Y(73)
E(67)=Y(74)
E(68)=Y(71)
E(78)=+Y(70)*Y(54)
E(79)=+Y(71)*Y(54)+1.0
E(80)=+Y(72)*Y(54)+Y(75)
E(102)=-.5*Y(77)*Y(57)*(1.0+Y(73))
E(103)=-.5*Y(77)*Y(57)*Y(74)
E(104)=-.5*Y(77)*Y(57)*Y(71)
E(106)=Y(76)
E(107)=Y(77)
E(108)=Y(78)
E(114)=-.5*Y(80)*Y(57)*(1.0+Y(73))
E(115)=-.5*Y(80)*Y(57)*Y(74)
E(116)=-.5*Y(80)*Y(57)*Y(71)
E(118)=Y(79)
E(119)=Y(80)
E(120)=Y(77)
E(126)=-.5*Y(54)*Y(77)*Y(57)*(1.0+Y(73))
E(127)=-.5*Y(54)*Y(77)*Y(57)*Y(74)
E(128)=-.5*Y(54)*Y(77)*Y(57)*Y(71)
E(130)=+Y(76)*Y(54)
E(131)=+Y(77)*Y(54)+1.0
E(132)=+Y(78)*Y(54)+Y(81)
GO TO 15

```


10

CONTINUE

E(13)=CX(1)
E(15)=CX(4)
E(17)=CX(2)
E(18)=CX(6)
E(21)=CX(3)
E(22)=CX(5)
E(37)=CX(4)
E(39)=CX(16)
E(41)=CX(9)
E(42)=CX(18)
E(45)=CX(13)
E(46)=CX(17)
E(73)=CX(6)
E(75)=CX(18)
E(77)=CX(11)
E(78)=CX(21)
E(81)=CX(15)
E(82)=CX(20)
E(85)=CX(2)
E(87)=CX(9)
E(89)=CX(7)
E(90)=CX(11)
E(93)=CX(8)
E(94)=CX(10)
E(121)=CX(5)
E(123)=CX(17)
E(125)=CX(10)
E(126)=CX(20)
E(129)=CX(14)
E(130)=CX(19)
E(133)=CX(3)
E(135)=CX(13)
E(137)=CX(8)
E(138)=CX(15)
E(141)=CX(12)
E(142)=CX(14)

C
C
C
C
C
C
C
C
C
C

15 RETURN
END

```

SUBROUTINE FUARO(IL)
REAL Y(100),AFA(240)
COMPLEX*16 CMS,CM1,CS1,CS2
COMPLEX*16 C(144)
COMMON/SUB/Y
COMMON/FUSA/AFA
COMMON/ARI/C
COMMON/FREF/CMS,CM1,CS1,CS2
IPQ=16*(IL-1)
DO 5 I=1,144
5 C(I)=DCMPLX(0.00,0.00)
C(18)=AFA(IPQ+1)
C(22)=-AFA(IPQ+2)
C(39)=-AFA(IPQ+3)*CS1+AFA(IPQ+4)
C(41)=-AFA(IPQ+5)*CS1
C(42)=AFA(IPQ+5)*AFA(IPQ+6)
C(45)=AFA(IPQ+7)*CS1
C(46)=-AFA(IPQ+7)*AFA(IPQ+6)
C(82)=AFA(IPQ+16)
C(87)=AFA(IPQ+8)*CS1+AFA(IPQ+9)
C(89)=AFA(IPQ+10)*CS1
C(90)=-AFA(IPQ+10)*AFA(IPQ+6)
C(93)=-AFA(IPQ+11)*CS1
C(94)=+AFA(IPQ+11)*AFA(IPQ+6)
C(126)=-AFA(IPQ+16)
C(135)=-AFA(IPQ+12)*CS1-AFA(IPQ+13)
C(137)=-AFA(IPQ+14)*CS1
C(138)=AFA(IPQ+14)*AFA(IPQ+6)
C(141)=AFA(IPQ+15)*CS1
C(142)=-AFA(IPQ+15)*AFA(IPQ+6)
DO 8 I=1,144,13
8 C(I)=DCMPLX(1.00,0.00)
RETURN
END

```

```

SUBROUTINE BLARD(AC,AD,IL)
COMPLEX*16 AC(144),AD(144)
COMPLEX*16 AMA(2550)
COMMON/AMAT/AMA
COMMON/HARM/NHARM
COMMON/INTER/NSY,NHSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
1MFLEX,MCON,MAER,MFUS,NBC,NFLAP,NFEA,NCT,NCON,NFFB,
2NAS,NHC,NVI,NSP,MAXN,NES,MSC,NEGN,IPCT,NIT,MER,NORM,
3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NCSR,
4NFP1,MXSMI,MXT2P1,MXKU,MXCPL,MXCSB,MXCPM,MXCPK,MXSMR,
5NEBC,NEBSC,MFASB,MXFAH,NFUS,NRBD,NRIFC,MXQ,NEIFC,
6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MODE
COMMON/NLEAD/MLEL,NLEL,MCTY
DO 10 J=1,144
AC(J)=DCMPLX(0.00,0.00)
10 AD(J)=DCMPLX(0.00,0.00)
IPQ = 34 * (IL-1)*MXSMI+ (NHC+NHARM ) * 34
21 FORMAT (1H0,10E12.6)
AC(39) = AMA(IPQ+1)
AC(41) = AMA(IPQ+2)
AC(45) = AMA(IPQ+3)
AC(87) = AMA(IPQ+4)
AC(89) = AMA(IPQ+5)
AC(93) = AMA(IPQ+6)
AC(135) = AMA(IPQ+7)
AC(137) = AMA(IPQ+8)
AC(141) = AMA(IPQ+9)
IF(NHARM.NE.0) GO TO 9
C SET DIAGONALS = 1., CREATE MATRIX D.
R DO 101 I = 1,144,13
101 AD(I)=DCMPLX(1.00,0.00)
9 AD(18) = AMA(IPQ+10)
AD(22) = AMA(IPQ+11)
AD(37) = AMA(IPQ+12)
AD(39) = AMA(IPQ+13)
AD(41) = AMA(IPQ+14)
AD(42) = AMA(IPQ+15)
AD(45) = AMA(IPQ+16)
AD(46) = AMA(IPQ+17)
AD(82) = AMA(IPQ+18)
AD(85) = AMA(IPQ+19)
AD(87) = AMA(IPQ+20)
AD(89) = AMA(IPQ+21)
AD(90) = AMA(IPQ+22)
AD(93) = AMA(IPQ+23)
AD(94) = AMA(IPQ+24)
AD(126) = AMA(IPQ+25)
AD(133) = AMA(IPQ+26)
AD(135) = AMA(IPQ+27)
AD(137) = AMA(IPQ+28)

```

```
AD(138) ■ AMA(IPQ+29)
AD(141) ■ AMA(IPQ+30)
AD(142) ■ AMA(IPQ+31)
RETURN
END
```

```

SUBROUTINE SUBMA(I,J)
  INTEGER IROW1(6)
  COMPLEX EX,EXPON
  COMPLEX*16 CM1
  COMPLEX*16 S(216)
  COMPLEX*16 TKN(441)
  COMMON/SS1/S
  COMMON/TKN1/TKN
  COMMON/INTER/NSY,NHSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
  IMFLEX,MCON,MAER,MFUS,NBC,NFLAP,NFEA,NCT,NCON,NFFB,
  2NAS,NHC,NVI,NSP,MAXN,NES,MSC,NEGN,IPCT,NIT,NER,NORM,
  3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NCSR,
  4NFP1,MXSMI,MXT2P1,MXKU,MXCPL,MXCSB,MXCPM,MXCPK,MXSMR,
  5NEBC,NEBSC,MFASB,MXFAB,MFUS,NRBD,NRIFC,MXQ,NEIFC,
  6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IRF,MODE
  COMMON/NLEAD/MLEL,NLEL,MCTY
  DATA IROW1/2,4,7,8,11,12/
  CM1=DCMPLX(0.00,1.00)
  NMK=J-I
  MNBC=(1-NBC)*(2-NBC)/2
  NNBC=(1-NBC)*(1-NBC)
  IF(NMK.NE.0)GO TO 225
  IMNC=(I-1)*12*NCOLS
  DO 210 IQ=1,NCOLS
  DO 210 IP=1,6
  KT=NEIFC+(IQ-1)*NRIFC+MXT2P1+IP
  KB=IMNC+(IQ-1)*12+IROW1(IP)
210  TKN(KT)=S(KB)
  DO 220 MS=1,NH
  DO 220 IQ=1,NCOLS
  KB1=IMNC+(IQ-1)*12+J
  KT1=NEIFC+(IQ-1)*NRIFC+MXT2P1+MFUS+NRBD*(MS-1)+4
  KB2=KB1-2
  KT2=KT1+1
220  TKN(KT2)=S(KB2)
  GO TO 240
225  IF(NMK.NE.1)GO TO 226
  IF(I.EQ.MXSMI)GO TO 240
  INT=1
  GO TO 230
226  IF(NMK.NE.-1)GO TO 240
  IF(I.EQ.1)GO TO 240
  INT=1
230  JMNC=(J-1)*12*NCOLS
  DO 235 MS=1,NH
  EX=EXPON(-INT,MS)
  DO 235 IQ=1,NCOLS
  KB=JMNC+(IQ-1)*12
  KB1=KB+10
  KB2=KB+9

```

```

KB3=KB+6
KB4=KB+5
KT=NEIFC+(IQ-1)*NRIFC+NFUS+MXT2P1+NRBD*(MS-1)
KT1=KT+1
KT2=KT+2
KT3=KT+3
KT4=KT+6
TKN(KT1)=-.5*(S(KB2)+INT*CM1*S(KB4))*EX
TKN(KT2)=-.5*(-S(KB3)+INT*CM1*S(KB1))*EX*NNBC
TKN(KT3)=-.5*(S(KB4)-INT*CM1*S(KB2))*EX
235 TKN(KT4)=-.5*(S(KB1)+INT*CM1*S(KB3))*EX*MNBC
240 CONTINUE
RETURN
END

```

```

SUBROUTINE SUBMB(I,J)
INTEGER IROW3(2),IALP(6),IROWC(2)
INTEGER P,Q
REAL*8 CX
COMPLEX EXPON
COMPLEX*16 CM1
COMPLEX*16 B(648),SMLB(108),SMLC(108),SMLD(108)
COMPLEX*16 CTB(54),FAB(54),FLB(54)
COMPLEX*16 CTB1(9),CTB2(9),CTB3(9),FAB1(9),FAB3(9),FLB1(9),FLB2(9)
COMPLEX*16 TKN(441)
COMPLEX*16 SMLE(108),FSB(54)
COMPLEX*16 FSB1(9),FSB2(9),FSB3(9),FAB4(9),FLB4(9),CTB4(9)
COMPLEX*16 SMLF(108),SMLG(108)
DIMENSION CX(75)
COMMON/BTS/B,SMLB,SMLC,SMLD,CTB,FAB,FLB,CTB1,CTB2,CTB3,FAB1,
1FAB3,FLB1,FLB2,SMLE,FSB,FSB1,FSB2,FSB3,FAB4,FLB4,CTB4,SMLF,SMLG
COMMON/CFLEX/CX
COMMON/SHAFT/TORFLX
COMMON/TKN1/TKN
COMMON/INTER/NSY,NBSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
1MFLX,MCON,MAER,MFUS,NBC,NFLAP,NPEA,NCT,NCON,NFFB,
2NAS,NHC,NVI,NSP,MAXN,NES,MSC,NEGN,IPCT,NIT,NER,NORM,
3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRB,NPD,NSK,NCOLS,NCSB,
4NFPI,MXSMI,MXT2P1,MXKG,MXCPL,MXCSB,MXCPM,MXCPK,MXSMB,
5NEBC,LESB,MFASB,MXFAB,MFUS,NRBD,NRIFC,MXG,NEIFC,
6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MODE
COMMON/NLEAD/MLEL,NLEL,MCTY
DATA IROWC/1,0/
DATA IROW3/12,7/
DATA IALP/1,3,5,6,9,10/
MCACF=0
MCACG=0
IF(MFUS.EQ.0) GO TO 65
CM1=DCMPLX(0.00,1.00)
NMK=J-I
NOND=0
IF(NMK.EQ.0)NOND=1
MNBC=(1-NBC)*(2-NBC)/2
NNBC=(1-NBC)*(1-NBC)
DO 10 Q=1,NCOLS
DO 10 P=1,2
L=(J-1)*NEBC+(I-1)*MXCPM+(Q-1)*12+IROW3(P)
KT=NEIFC+MXT2P1+NEISC+(Q-1)*NRIFC+P
LC=(J-1)*NCOLS+MXSMI+(I-1)*NCOLS+Q
10 TKN(KT)=B(L)+FLB(LC)*MFLX*MCON*IROWC(P)
IF(MFLX.EQ.0) GO TO 11
IF(MCON.EQ.0) GO TO 29
MCT=1
MFEA=1
MFLAP=1

```

```

MLEL=0
IF(MCTY.EQ.0) GO TO 11
MCACF=1
MCACG=1
MLEL=1
11 DO 15 P=1,2
L=(J-1)*NESBC+(I-1)*MXCSR+IR0W3(P)
KT=NEIFC+MXT2P1+NEISC+NCOLS*NRIFC+P
IF(MCT.EQ.0) GO TO 12
TKN(KT)=SMLB(L)+CX(3)*MFLEX*IR0WC(P)*NONO
KT=KT+NRIFC
12 IF(MFEA.EQ.0) GO TO 13
TKN(KT)=SMLC(L)+CX(8)*MFLEX*IR0WC(P)*NONO
KT=KT+NRIFC
13 IF(MFLAP.EQ.0) GO TO 14
TKN(KT)=SMLD(L)+CX(12)*MFLEX*IR0WC(P)*NONO
KT=KT+NRIFC
14 IF(MLEL.EQ.0) GO TO 16
TKN(KT)=SMLE(L)+CX(24)*MFLEX*IR0WC(P)*NONO
KT=KT+NRIFC
16 IF(MCACF.EQ.0) GO TO 17
TKN(KT)=SMLF(L)+CX(27)*MFLEX*IR0WC(P)*NONO
KT=KT+NRIFC
17 IF(MCACG.EQ.0) GO TO 15
TKN(KT)=SMLG(L)+CX(30)*MFLEX*IR0WC(P)*NONO
15 CONTINUE
20 CONTINUE
IF(I.EQ.1) GO TO 30
30 DO 31 Q=1,NCOLS
LM=(J-1)*NEBC+(I-2)*MXCPH+(Q-1)*12
KT=NEIFC+MXT2P1+NEISC+(Q-1)*NRIFC+3
LM1=LM+4
LM2=LM+11
TKN(KT)=TKN(KT)+.5*(B(LM1)-CM1*MNBC*B(LM2))*NNBC
KT=KT+1
LM3=LM+8
LM4=LM+2
TKN(KT)=TKN(KT)+.5*(B(LM3)-CM1*B(LM4))
KT=KT+1
TKN(KT)=TKN(KT)+.5*(MNBC*B(LM2)+CM1*B(LM1))*NNBC
KT=KT+1
31 TKN(KT)=TKN(KT)+.5*(B(LM4)+CM1*B(LM3))
IF(MFLEX.EQ.1.AND.MCON.EQ.0) GO TO 35
LM=(J-1)*NESBC+(I-2)*MXCSR
KT=NEIFC+MXT2P1+NEISC+NCOLS*NRIFC+3
LM1=LM+4
LM2=LM+11
LM3=LM+8
LM4=LM+2
IF(MCT.EQ.0) GO TO 32

```



```

TKN(KT) = TKN(KT) - .5 * (SMLB(LM1) - CM1 * MNBC * SMLB(LM2)) * NNRC
KT = KT + 1
TKN(KT) = TKN(KT) + .5 * (SMLB(LM3) - CM1 * SMLB(LM4))
KT = KT + 1
TKN(KT) = TKN(KT) + .5 * (MNBC * SMLB(LM2) + CM1 * SMLB(LM1)) * NNRC
KT = KT + 1
TKN(KT) = TKN(KT) - .5 * (SMLB(LM4) + CM1 * SMLB(LM3))
KT = KT - 3 + NRIFC
32 IF (MFEA, EQ, 0) GO TO 33
TKN(KT) = TKN(KT) - .5 * (SMLC(LM1) - CM1 * MNBC * SMLC(LM2)) * NNRC
KT = KT + 1
TKN(KT) = TKN(KT) + .5 * (SMLC(LM3) - CM1 * SMLC(LM4))
KT = KT + 1
TKN(KT) = TKN(KT) + .5 * (MNBC * SMLC(LM2) + CM1 * SMLC(LM1)) * NNRC
KT = KT + 1
TKN(KT) = TKN(KT) - .5 * (SMLC(LM4) + CM1 * SMLC(LM3))
KT = KT - 3 + NRIFC
33 IF (MFLAP, EQ, 0) GO TO 34
TKN(KT) = TKN(KT) - .5 * (SMLD(LM1) - CM1 * MNBC * SMLD(LM2)) * NNRC
KT = KT + 1
TKN(KT) = TKN(KT) + .5 * (SMLD(LM3) - CM1 * SMLD(LM4))
KT = KT + 1
TKN(KT) = TKN(KT) + .5 * (MNBC * SMLD(LM2) + CM1 * SMLD(LM1)) * NNRC
KT = KT + 1
TKN(KT) = TKN(KT) - .5 * (SMLD(LM4) + CM1 * SMLD(LM3))
KT = KT - 3 + NRIFC
34 IF (MLEL, EQ, 0) GO TO 36
TKN(KT) = TKN(KT) - .5 * (SMLE(LM1) - CM1 * MNBC * SMLE(LM2)) * NNRC
KT = KT + 1
TKN(KT) = TKN(KT) + .5 * (SMLE(LM3) - CM1 * SMLE(LM4))
KT = KT + 1
TKN(KT) = TKN(KT) + .5 * (MNBC * SMLE(LM2) + CM1 * SMLE(LM1)) * NNRC
KT = KT + 1
TKN(KT) = TKN(KT) - .5 * (SMLE(LM4) + CM1 * SMLE(LM3))
KT = KT - 3 + NRIFC
36 IF (MCACF, EQ, 0) GO TO 37
TKN(KT) = TKN(KT) - .5 * (SMLF(LM1) - CM1 * MNBC * SMLF(LM2)) * NNRC
KT = KT + 1
TKN(KT) = TKN(KT) + .5 * (SMLF(LM3) - CM1 * SMLF(LM4))
KT = KT + 1
TKN(KT) = TKN(KT) + .5 * (MNBC * SMLF(LM2) + CM1 * SMLF(LM1)) * NNRC
KT = KT + 1
TKN(KT) = TKN(KT) - .5 * (SMLF(LM4) + CM1 * SMLF(LM3))
KT = KT - 3 + NRIFC
37 IF (MCACG, EQ, 0) GO TO 35
TKN(KT) = TKN(KT) - .5 * (SMLG(LM1) - CM1 * MNBC * SMLG(LM2)) * NNRC
KT = KT + 1
TKN(KT) = TKN(KT) + .5 * (SMLG(LM3) - CM1 * SMLG(LM4))
KT = KT + 1
TKN(KT) = TKN(KT) + .5 * (MNBC * SMLG(LM2) + CM1 * SMLG(LM1)) * NNRC

```

```

KT=KT+1
TKN(KT)=TKN(KT)+.5*(SMLG(LM4)+CM1*SMLG(LM3))
35 CONTINUE
50 IF(I.GE.MXSMI) GO TO 45
40 DO 41 Q=1,NCOLS
LP=(J-1)*NEHC+I*MXCPM+(Q-1)*12
KT=NEIFC+MXT2P1+NEISC+(Q-1)*NRIFC+3
LP1=LP+4
LP2=LP+11
LP3=LP+8
LP4=LP+2
TKN(KT)=TKN(KT)+.5*(B(LP1)+CM1*MNBC*B(LP2))*NNBC
KT=KT+1
TKN(KT)=TKN(KT)+.5*(B(LP3)+CM1*B(LP4))
KT=KT+1
TKN(KT)=TKN(KT)+.5*(MNBC*B(LP2)+CM1*B(LP1))*NNBC
KT=KT+1
41 TKN(KT)=TKN(KT)+.5*(B(LP4)+CM1*B(LP3))
IF(MFLEX.EQ.1.AND.MCON.EQ.0) GO TO 45
LP=(J-1)*NESHC+I*MXCSB
KT=NEIFC+MXT2P1+NEISC+NCOLS*NRIFC+3
LP1=LP+4
LP2=LP+11
LP3=LP+8
LP4=LP+2
IF(MCT.EQ.0) GO TO 42
TKN(KT)=TKN(KT)+.5*(SMLB(LP1)+CM1*MNBC*SMLB(LP2))*NNBC
KT=KT+1
TKN(KT)=TKN(KT)+.5*(SMLB(LP3)+CM1*SMLB(LP4))
KT=KT+1
TKN(KT)=TKN(KT)+.5*(MNBC*SMLB(LP2)+CM1*SMLB(LP1))*NNBC
KT=KT+1
TKN(KT)=TKN(KT)+.5*(SMLB(LP4)+CM1*SMLB(LP3))
KT=KT+3+NRIFC
42 IF(MFEA.EQ.0) GO TO 43
TKN(KT)=TKN(KT)+.5*(SMLC(LP1)+CM1*MNBC*SMLC(LP2))*NNBC
KT=KT+1
TKN(KT)=TKN(KT)+.5*(SMLC(LP3)+CM1*SMLC(LP4))
KT=KT+1
TKN(KT)=TKN(KT)+.5*(MNBC*SMLC(LP2)+CM1*SMLC(LP1))*NNBC
KT=KT+1
TKN(KT)=TKN(KT)+.5*(SMLC(LP4)+CM1*SMLC(LP3))
KT=KT+3+NRIFC
43 IF(MFLAP.EQ.0) GO TO 44
TKN(KT)=TKN(KT)+.5*(SMLD(LP1)+CM1*MNBC*SMLD(LP2))*NNBC
KT=KT+1
TKN(KT)=TKN(KT)+.5*(SMLD(LP3)+CM1*SMLD(LP4))
KT=KT+1
TKN(KT)=TKN(KT)+.5*(MNBC*SMLD(LP2)+CM1*SMLD(LP1))*NNBC
KT=KT+1

```

```

      TKN(KT)=TKN(KT)-.5*(SMLD(LP4)-CM1*SMLD(LP3))
      KT=KT+3+NRIFC
44  IF(MLEL,EQ,0) GO TO 38
      TKN(KT)=TKN(KT)+.5*(SMLE(LP1)+CM1*MNBC*SMLE(LP2))*NNBC
      KT=KT+1
      TKN(KT)=TKN(KT)+.5*(SMLE(LP3)+CM1*SMLE(LP4))
      KT=KT+1
      TKN(KT)=TKN(KT)+.5*(MNBC*SMLE(LP2)-CM1*SMLE(LP1))*NNBC
      KT=KT+1
      TKN(KT)=TKN(KT)-.5*(SMLE(LP4)-CM1*SMLE(LP3))
      KT=KT+3+NRIFC
38  IF(MCACF,EQ,0) GO TO 39
      TKN(KT)=TKN(KT)+.5*(SMLF(LP1)+CM1*MNBC*SMLF(LP2))*NNBC
      KT=KT+1
      TKN(KT)=TKN(KT)+.5*(SMLF(LP3)+CM1*SMLF(LP4))
      KT=KT+1
      TKN(KT)=TKN(KT)+.5*(MNBC*SMLF(LP2)-CM1*SMLF(LP1))*NNBC
      KT=KT+1
      TKN(KT)=TKN(KT)-.5*(SMLF(LP4)-CM1*SMLF(LP3))
      KT=KT+3+NRIFC
39  IF(MCACG,EQ,0) GO TO 45
      TKN(KT)=TKN(KT)+.5*(SMLG(LP1)+CM1*MNBC*SMLG(LP2))*NNBC
      KT=KT+1
      TKN(KT)=TKN(KT)+.5*(SMLG(LP3)+CM1*SMLG(LP4))
      KT=KT+1
      TKN(KT)=TKN(KT)+.5*(MNBC*SMLG(LP2)-CM1*SMLG(LP1))*NNBC
      KT=KT+1
      TKN(KT)=TKN(KT)-.5*(SMLG(LP4)-CM1*SMLG(LP3))
45  CONTINUE
      IF(NBP,EQ,0) GO TO 100
      NPMK=NPS+NFP1=I
      INMK=IABS(NPMK)
      RINMK=1.0*INMK
      RFA=RINMK/NBP
      NFA=INMK/NBP
      DIF=ABS(RFA-1.0*NFA)
      IF(DIF,LT,(.05)) GO TO 70
      ISEXP=0
      GO TO 71
70  ISEXP=NBP
71  IQMAX=NRBD
      DO 73 P=1,6
      DO 73 Q=1,IQMAX
      KT=NEIFC+MXT2P1+NEISC+(Q-1)*NRIFC+P
73  TKN(KT)=TKN(KT)*ISEXP
      GO TO 65
100 IF(NB,LT,2) GO TO 65
46  NMK=J=I
      DO 47 M=2,NB
      DO 47 Q=1,NRBD

```

```

DO 47 P=1,6
KT=NEIFC+MXT2P1+NEISC+(Q-1)*NRIFC+P
KT2=KT+(MS-1)*NEITC
TKN(KT2)=TKN(KT)*EXPON(NMK,MS)
IF(P,NE,2) GO TO 47
DO 57 MM=1,NR
IF(MS,GT,2) GO TO 56
IF(MM,EQ,1) GO TO 56
KT4=KT+B+(MM-1)*NRBD
TKN(KT4)=TKN(KT)*TORFLX
56 CONTINUE
KT3=KT2+B+(MM-1)*NRBD
TKN(KT3)=TKN(KT2)*TORFLX
57 CONTINUE
TKN(KT2)=DCMPLX(0,00,0,00)
47 CONTINUE
65 DO 110 KK=1,6
DO 110 W=1,NCOLS
L=(J-1)*NESBC+(I-1)*MXCPM+(Q-1)*12+IALP(KK)
KT=NEIFC+NEISC+(Q-1)*NRIFC+MXT2P1+NFUS+KK
IF(KK,EQ,4,AND,MFUS,NE,0) GO TO 111
TKN(KT)=H(L)
GO TO 110
111 KTT=KT+B
TKN(KT)=H(L)+TKN(KTT)*TORFLX
TKN(KTT)=DCMPLX(0,00,0,00)
110 CONTINUE
IF(MFLEX,EQ,0) GO TO 112
IF(MCUN,EQ,0) GO TO 129
MCT=1
MFEA=1
MFLAP=1
MLEL=0
IF(MCTY,EQ,0) GO TO 112
MCACF=1
MCACG=1
MLEL=1
112 DO 121 KK=1,6
L=(J-1)*NESBC+(I-1)*MXCSB+IALP(KK)
KT=NEIFC+NEISC+NCOLS*NRIFC+MXT2P1+NFUS+KK
IF(MCT,EQ,0) GO TO 122
TKN(KT)=SMLB(L)
IF(KK,NE,4) GO TO 113
IF(MFUS,EQ,0) GO TO 113
KTT=KT+B
TKN(KT)=SMLB(L)+TKN(KTT)*TORFLX
TKN(KTT)=DCMPLX(0,00,0,00)
113 KT=KT+NRIFC
122 IF(MFEA,EQ,0) GO TO 123
TKN(KT)=SMLC(L)

```

```

IF(KK,NE,4) GO TO 114
IF(MFUS,EQ,0) GO TO 114
KTT=KT-8
TKN(KT)=-SMLC(L)+TKN(KTT)*TORFLX
TKN(KTT)=DCMPLX(0,00,0,00)
114 KT=KT+NRIFC
123 IF(MFLAP,EQ,0) GO TO 124
TKN(KT)=-SMLD(L)
IF(KK,NE,4) GO TO 115
IF(MFUS,EQ,0) GO TO 115
KTT=KT-8
TKN(KT)=-SMLD(L)+TKN(KTT)*TORFLX
TKN(KTT)=DCMPLX(0,00,0,00)
115 KT=KT+NRIFC
124 IF(MLEL,EQ,0) GO TO 125
TKN(KT)=-SMLE(L)
IF(KK,NE,4) GO TO 116
IF(MFUS,EQ,0) GO TO 116
KTT=KT-8
TKN(KT)=-SMLE(L)+TKN(KTT)*TORFLX
TKN(KTT)=DCMPLX(0,00,0,00)
116 KT=KT+NRIFC
125 IF(MCACF,EQ,0) GO TO 126
TKN(KT)=-SMLF(L)
IF(KK,NE,4) GO TO 117
IF(MFUS,EQ,0) GO TO 117
KTT=KT-8
TKN(KT)=-SMLF(L)+TKN(KTT)*TORFLX
TKN(KTT)=DCMPLX(0,00,0,00)
117 KT=KT+NRIFC
126 IF(MCACG,EQ,0) GO TO 121
TKN(KT)=-SMLG(L)
IF(KK,NE,4) GO TO 121
IF(MFUS,EQ,0) GO TO 121
KTT=KT-8
TKN(KT)=-SMLG(L)+TKN(KTT)*TORFLX
TKN(KTT)=DCMPLX(0,00,0,00)
121 CONTINUE
129 CONTINUE
RETURN
END

```

```

SUBROUTINE SUBMG(I,J)
INTEGER P,Q
REAL*8 CX
COMPLEX*16 B(648),SMLB(108),SMLC(108),SMLD(108)
COMPLEX*16 CTB(54),FAB(54),FLB(54)
COMPLEX*16 CTB1(9),CTB2(9),CTB3(9),FAB1(9),FAB3(9),FLB1(9),FLB2(9)
COMPLEX*16 TKN(441)
COMPLEX*16 SMLE(108),FSB(54)
COMPLEX*16 SMLF(108),SMLG(108)
COMPLEX*16 FSB1(9),FSB2(9),FSB3(9),FAB4(9),FLB4(9),CTB4(9)
COMPLEX*16 YKM(3,6),CS1,CS2,CM1,CMS
DIMENSION CX(75)
DIMENSION AL536(6),AL436(6)
COMMON/CPARS/ALSTH(6)
COMMON/SPAR/AKCI(6),TAU(6),SMLA(6),DMS(6),AK(4),AC(4),BJ(4),
1CAPK,CAPC
COMMON/CPAR/AMKC(6),CTAU(6),AL(6,6),AL12(6)
COMMON/INTER/NSY,NBSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
1MFLEX,MCON,MAER,MFUS,NBC,NFLAP,NFEA,NCT,NCON,NFFB,
2NAS,NHC,NVI,NSP,MAXN,NES,MSC,NEGN,IPCT,NIT,NER,NORM,
3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NCSB,
4NFP1,MXSMI,MXT2P1,MXKQ,MXCPL,MXCSB,MXCPM,MXCPK,MXSMB,
5NEBC,NEBC,MFASB,MXFAB,NFUS,NKBD,NRIFC,MXQ,NEIFC,
6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MODE
COMMON/ROTF/OM1,OM2,OMT
COMMON/FREF/CMS,CM1,CS1,CS2
COMMON/TKN1/TKN
COMMON/CFLEX/CX
COMMON/BTS/B,SMLB,SMLC,SMLD,CTB,FAB,FLB,CTB1,CTB2,CTB3,FAB1,FAB3,
1FLB1,FLB2,SMLE,FSB,FSB1,FSB2,FSB3,FAB4,FLB4,CTB4,SMLF,SMLG
COMMON/NLEAD/MLEL,NLEL,MCTY
NMK=J-1
NUNU=1
KSMLE=I-NFP1
CS1=CMS=CM1*KSMLE*OM1
IF(NMK.NE.0) NONQ=0
IF(MFLEX.EQ.0) GO TO 100
IF(MCON.EQ.0) GO TO 900
M3=1
IF(MCTY.GT.0) GO TO 200
YKM(I,M3)=AMKC(M3)/((1.+CS1*CTAU(M3)*AMKC(M3))*AL(6,M3)*AL(6,M3))
DO 300 Q=1,NRBD
KT=NEIFC+NEISC+(Q-1)*NRIFC+MXT2P1+NFUS+NCOLS
KUZ=NEIFC+NEISC+(Q-1)*NRIFC+MXT2P1+NFUS+5
KUX=KUZ-4
KPY=KUZ+1
KUY=KUZ-2
KPX=KUZ-3
IF(Q.GT.6) GO TO 301
L=(J-1)*NCOLS+MXSMI+(I-1)*NCOLS+Q

```

```

TKN(KT+1)=AL(1,MS)*CTB(L)-AL(2,MS)*FAB(L)-AL(3,MS)*FLB(L)
TKN(KT+2)=AL(4,MS)*CTB(L)-AL(5,MS)*FAB(L)
TKN(KT+3)=YKM(I,MS)*FLB(L)+(NSP=1)*TKN(KUZ)
1=(TKN(KUX)-AL12(MS)*TKN(KPY))*AL(3,MS)/AL(6,MS)*AL(5,MS)
2=(TKN(KUY)+AL12(MS)*TKN(KPX))*AL(3,MS)/AL(6,MS)*AL(4,MS)
GO TO 300
301 IF(Q=8) 302,303,304
302 TKN(KT+1)=(AL(1,MS)*CX(1)+AL(2,MS)*CX(2)+AL(3,MS)*CX(3))*NONO
TKN(KT+2)=(AL(4,MS)*CX(1)+AL(5,MS)*CX(2))*NONO
TKN(KT+3)=YKM(I,MS)*CX(3)*NONO+(NSP=1)*TKN(KUZ)
1=(TKN(KUX)-AL12(MS)*TKN(KPY))*AL(3,MS)/AL(6,MS)*AL(5,MS)
2=(TKN(KUY)+AL12(MS)*TKN(KPX))*AL(3,MS)/AL(6,MS)*AL(4,MS)
3=AL(3,MS)/AL(6,MS)*AL(5,MS)*NONO
GO TO 300
303 TKN(KT+1)=(AL(1,MS)*CX(2)+AL(2,MS)*CX(7)+AL(3,MS)*CX(8))*NONO
TKN(KT+2)=(AL(4,MS)*CX(2)+AL(5,MS)*CX(7))*NONO
TKN(KT+3)=YKM(I,MS)*CX(8)*NONO+(NSP=1)*TKN(KUZ)
1=(TKN(KUX)-AL12(MS)*TKN(KPY))*AL(3,MS)/AL(6,MS)*AL(5,MS)
2=(TKN(KUY)+AL12(MS)*TKN(KPX))*AL(3,MS)/AL(6,MS)*AL(4,MS)
3=AL(3,MS)/AL(6,MS)*AL(4,MS)*NONO
GO TO 300
304 TKN(KT+1)=(AL(1,MS)*CX(3)+AL(2,MS)*CX(8)+AL(3,MS)*CX(12))*NONO
TKN(KT+2)=(AL(4,MS)*CX(3)+AL(5,MS)*CX(8))*NONO
TKN(KT+3)=(YKM(I,MS)*CX(12)+DCMPLX(1.00,0.00))*NONO+(NSP=1)*
TKN(KUZ)
1=(TKN(KUX)-AL12(MS)*TKN(KPY))*AL(3,MS)/AL(6,MS)*AL(5,MS)
2=(TKN(KUY)+AL12(MS)*TKN(KPY))*AL(3,MS)/AL(6,MS)*AL(4,MS)
300 CONTINUE
GO TO 900
200 YKM(I,MS)=AMKC(MS)/(1.+CS1*CTAU(MS)*AMKC(MS))
AL536(MS)=AL(5,MS)*AL(3,MS)/AL(6,MS)
AL436(MS)=AL(4,MS)*AL(3,MS)/AL(6,MS)
DO 250 Q=1,NR8D
KT=NEIFC+NEISC+(Q-1)*NRIFC+MXT2P1+NFUS+NCOL8
KUZ=NEIFC+NEISC+(Q-1)*NRIFC+MXT2P1+NFUS+5
KUX=KUZ-4
KPY=KUZ+1
KUY=KUZ-2
KPX=KUZ-3
TKN(KT+1)=TKN(KUX)-ALSTH(MS)*TKN(KPY)
TKN(KT+2)=TKN(KUY)+ALSTH(MS)*TKN(KPX)
TKN(KT+3)=TKN(KUZ)
IF(NONO.EQ.0) GO TO 201
IF(Q.LT.7) GO TO 201
IF(Q.GT.9) GO TO 201
IF(Q.EQ.7) TKN(KT+1)=TKN(KT+1)-DCMPLX(1.00,0.00)
IF(Q.EQ.8) TKN(KT+2)=TKN(KT+2)-DCMPLX(1.00,0.00)
IF(Q.EQ.9) TKN(KT+3)=TKN(KT+3)-DCMPLX(1.00,0.00)
201 IF(Q.GT.6) GO TO 205
L=(J=1)*NCOL8+MXSMI+(I=1)*NCOL8+Q

```

```

      TKN(KT+6)=AL536(MS)*CTH(L)-AL436(MS)*FAB(L)-FSB(L)-
1AL536(MS)*(TKN(KUX)-AL12(MS)*TKN(KPY))+
2AL436(MS)*(TKN(KUY)+AL12(MS)*TKN(KPX))+(NSP-1)*TKN(KUZ)
      GO TO 250
205 LL1=58+(Q-7)*3
      LL2=LL1+1
      LL3=LL1+2
      TKN(KT+6)=-AL536(MS)*(TKN(KUX)-AL12(MS)*TKN(KPY))+
1AL436(MS)*(TKN(KUY)+AL12(MS)*TKN(KPX))+(NSP-1)*TKN(KUZ)+
2(AL536(MS)*CX(LL1)-AL436(MS)*CX(LL2)-CX(LL3))*NONO
C  NOTE ELEMENTS 64,65,66 OF CX ARRAY CHANGED SIGN IN SECPAR
      IF(Q,LT,10) GO TO 250
      IF(NON0,EQ,0) GO TO 250
      IF(Q,GT,10) GO TO 210
      TKN(KT+4)=AL(1,MS)
      TKN(KT+5)=AL(4,MS)
      TKN(KT+6)=TKN(KT+6)+AL536(MS)*YKM(I,MS)
      GO TO 250
210 IF(Q,GT,11) GO TO 220
      TKN(KT+4)=AL(2,MS)
      TKN(KT+5)=AL(5,MS)
      TKN(KT+6)=TKN(KT+6)-AL436(MS)*YKM(I,MS)
      GO TO 250
220 TKN(KT+4)=AL(3,MS)
      TKN(KT+6)=TKN(KT+6)-YKM(I,MS)
250 CONTINUE
      GO TO 900
100 CONTINUE
      MS=1
      YKM(I,MS)=SMLA(MS)*(1+CS1*TAU(MS))
      DO 10 Q=1,NCOLS
      KT=NEIFC+NEISC+(Q-1)*NRIFC+MXT2P1+NFUS+7
      L=(J-1)*NCOLS+MXSMI+(I-1)*NCOLS+Q
      IF (MCT,EQ,0) GO TO 20
      TKN(KT)=SMLA(MS)*CTB(L)+YKM(I,MS)
      KT=KT+1
20 IF(MFEA,EQ,0) GO TO 30
      TKN(KT)=FAB(L)
      KT=KT+1
30 IF(MFLAP,EQ,0) GO TO 31
      TKN(KT)=FLB(L)
      KT=KT+1
31 IF(MLEL,EQ,0) GO TO 10
      TKN(KT)=FSB(L)
10 CONTINUE
      KT=NEIFC+NEISC+NCOLS*NRIFC+MXT2P1+NFUS+7
      L=(J-1)*MXSMI+(I-1)+1
      IF (MCT,EQ,0) GO TO 40
      TKN(KT)=AKCI(MS)*CTB1(L)
      KT1=KT+NRIFC

```



```

        IF(MFEA,EQ,0) GO TO 32
        TKN(KT1)=SMLA(MS)*YKM(I,MS)*CTB2(L)
        KT1=KT1+NRIFC
32  IF(MFLAP,EQ,0) GO TO 34
        TKN(KT1)=SMLA(MS)*CTB3(L)*YKM(I,MS)
        KT1=KT1+NRIFC
34  IF(MLEL,EQ,0) GO TO 36
        TKN(KT1)=SMLA(MS)*CTB4(L)*YKM(I,MS)
36  KT=KT+1
40  IF(MFEA,EQ,0) GO TO 50
        KT2=KT+NRIFC
        IF(MCT,EQ,0) GO TO 42
        TKN(KT)=FAB1(L)
        KT2=KT2+NRIFC
42  IF(MFLAP,EQ,0) GO TO 44
        TKN(KT2)=FAB3(L)
        KT2=KT2+NRIFC
44  IF(MLEL,EQ,0) GO TO 46
        TKN(KT2)=FAB4(L)
46  KT=KT+1
50  IF(MFLAP,EQ,0) GO TO 56
        KT1=KT
        IF(MCT,EQ,0) GO TO 52
        TKN(KT)=FLB1(L)
        KT1=KT1+NRIFC
52  IF(MFEA,EQ,0) GO TO 54
        TKN(KT1)=FLB2(L)
        KT1=KT1+NRIFC
54  KT1=KT1+NRIFC
        IF(MLEL,EQ,0) GO TO 55
        TKN(KT1)=FLB4(L)
55  KT=KT+1
56  IF(MLEL,EQ,0) GO TO 60
        IF(MCT,EQ,0) GO TO 57
        TKN(KT)=FSB1(L)
        KT=KT+NRIFC
57  IF(MFEA,EQ,0) GO TO 58
        TKN(KT)=FSB2(L)
        KT=KT+NRIFC
58  IF(MFLAP,EQ,0) GO TO 60
        TKN(KT)=FSB3(L)
60  CONTINUE
900 CONTINUE
    RETURN
    END

```

```

SUBROUTINE SURMF(I,J)
  INTEGER P,Q
  COMPLEX EXPON
  COMPLEX*16 TKN(441)
  COMMON/INTER/NSY,NHSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
  1MFLEX,MCON,MAER,MFUS,NBC,NFLAP,NFEA,NCT,NCON,NFFB,
  2NAS,NHC,NVI,NSP,MAXN,NES,MSC,NEGN,IPCT,NIT,NER,NORM,
  3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NCSB,
  4NFPI,MXSMI,MXT2P1,MXKQ,MXCPL,MXCSB,MXCPM,MXCPK,MXSMB,
  5NEBC,NEBRC,MFASH,MXFAB,NFUS,NRBD,NRIFC,MXQ,NEIFC,
  6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MODE
  COMMON/TKN1/TKN
  COMMON/NLEAD/MLEL,NLEL,MCTY
  IF(NB.EQ.1)GO TO 586
  NMK=J-I
  DO 547 MS=2,NB
  DO 547 Q=1,NRBD
  DO 547 P=1,NRBD
  KT=NEIFC+NEISC+(Q-1)*NRIFC+NFUS+MXT2P1+P
  KT2=KT+(MS-1)*(NEITC+NRBD)
547 TKN(KT2)=TKN(KT)*EXPON(NMK,MS)
586 RETURN
END

```

```

SUBROUTINE SOLVE
REAL*8 DTLG10,DFAC,FAC
COMPLEX*16 ZERO,TWO,SUM,SWAP,DTPHAS,DPIVOT,DETSV
COMPLEX*16 EPS(63),FORCE(63),DB(63,63)
COMMON/EPSA/EPS
COMMON/REMA/DETSV
COMMON/CDETRM/DPIVOT,DTPHAS,DTLG10,IDET
COMMON/DTERM/DFAC
FAC=20,00
TWO=DCMPLX(2,00,0,00)
ZERO=DCMPLX(0,00,0,00)
REWIND 1
READ(1)MXSMI,NRIFC,NHC,NORM,IHEM1,NEX
NORDER=MXSMI*NRIFC
DO 11 L=1,MXSMI
  IROWF=L*NRIFC
  IROWS=IROWF+NRIFC+1
  DO 11 K=1,MXSMI
    ICOLF=K*NRIFC
    ICOLS=ICOLF+NRIFC+1
11 READ(1)((DB(I,J),I=IROWS,IROWF),J=ICOLS,ICOLF)
REWIND 1
DO 12 I=1,NORDER
  IK=NORDER+1-I
12 FORCE(IK)=EPS(IK)
C 200 J=1,45
C 200 WRITE(6,300)(DB(I,J),I=1,45)
C 300 FORMAT(5X,10D12.4)
IDET=1
ISBLK=NHC*NRIFC
ICOL=ISBLK+NORM
IROW=ISBLK+IHEM1
NEXCOL=ISBLK+NEX
CALL SWAPS(DB,NORDER,ICOL,IROW)
SWAP=FORCE(IROW)
FORCE(IROW)=FORCE(NORDER)
FORCE(NORDER)=SWAP
N=NORDER-1
CALL ERRSET(208,500)
CALL DCMAT(DB,N,FORCE)
CALL ERKSET(208,10)
WRITE(6,102)DTPHAS,DTLG10,DPIVOT
102 FORMAT(/2X,'DTPHAS = ',2D40.16,/2X,'DTLG10 = ',
1D40.16,/2X,'DPIVOT = ',2D40.16)
IFAC=DTLG10/FAC
DFAC=IFAC*FAC
DTLG10=DTLG10-DFAC
DTPHAS=DTPHAS+DCMPLX(10,00**DTLG10,0,00)
DETSV=DTPHAS*DPIVOT
SUM=ZERO

```

```

DO 2 I=1,N
IF(I.EQ.NEXCOL) GO TO 2
SUM=SUM+DH(NORDER,I)*FORCE(I)
2 CONTINUE
SUM=FORCE(NORDER)=SUM
IF(CDABS(DB(NORDER,NEXCOL)).NE.0.00)GO TO 5
WRITE(6,6)NORDER,NEXCOL
6 FORMAT('0   DB(' ,IS,' ,',IS,' )   IS ZERO')
STOP
5 CONTINUE
SUM=SUM/DH(NORDER,NEXCOL)
FORCE(NEXCOL)=(SUM+FORCE(NEXCOL))/TWO
FORCE(NORDER)=FORCE(ICOL)
FORCE(ICOL)=ZERO
DO 3 I=1,NORDER
3 EPS(I)=FORCE(I)
RETURN
END

```

```

SUBROUTINE SWAPS(DB,NORDER,ICOL,IROW)
COMPLEX*16 DB(63,63),SWAP
DO 1 I=1,NORDER
  SWAP=DB(I,ICOL)
  DB(I,ICOL)=DB(I,NORDER)
1 DB(I,NORDER)=SWAP
DO 2 I=1,NORDER
  SWAP=DB(IROW,I)
  DB(IROW,I)=DB(NORDER,I)
2 DB(NORDER,I)=SWAP
RETURN
END

```

```

SUBROUTINE DCMAT(A,N,Y)
REAL*8 ADET,AMAG,USIGN
COMPLEX*16 A,Y,ODET,DPIVOT,TK,X
COMPLEX*16 DAIJ,AMX,DONE,DYI,TEMP,DAKJ,DYK,DAKK,DAIK
1,UNE,DPHAS
DIMENSION ICHG(63),A(63,63),Y(63),X(63)
COMMON /CDETRM/ DPIVOT,DPHAS,ADET,IDET
NDIM=63
ADET=0.00
DSIGN=1.00
DPHAS=DCMPLX(1.000,0.00)
NP1=N
IF(IDET.EQ.0) GO TO 650
NP1=N+1
DO 651 I=1,NP1
651 X(I)=A(NP1,I)
650 CONTINUE
DO 118 K=1,N
AMX = A(K,K)
IMX=K
DO 100 I=K,N
IF(CDABS(A(I,K)) .LE. CDABS(AMX)) GO TO 100
AMX = A(I,K)
IMX=I
100 CONTINUE
102 IF (IMX.EQ.K) GO TO 106
DO 104 J=1,NP1
TEMP=A(K,J)
A(K,J)=A(IMX,J)
104 A(IMX,J)=TEMP
ICHG(K)=IMX
TEMP=Y(K)
Y(K)= Y(IMX)
Y(IMX)= TEMP
DPHAS=-DPHAS
GO TO 108
106 ICHG(K)=K
108 CONTINUE
DAKK=A(K,K)
901 FORMAT(1X,I5,2D40.16/)
C WRITE(6,1000) DAKK
C1000 FORMAT(5X,'DAKK',5X,2D20.10)
AMAG=CDABS(DAKK)
IF(AMAG.NE.0.00) GO TO 6
WRITE(6,7)
7 FORMAT('0 MATRIX IN DCMAT IS SINGULAR')
STOP
6 CONTINUE
ADET=ADET+DLOG10(AMAG)
DPHAS=DPHAS*DAKK/AMAG

```

```

DUNE=DCMPLX(1.00,0.00)
DAKK=DUNE/DAKK
DO 110 J=1,NP1
110 A(K,J)=A(K,J)*DAKK
A(K,K)=DAKK
IF(IDET.EQ.0) GO TO 652
TK=X(K)
DO 653 J=K,NP1
653 X(J)=X(J)-TK*A(K,J)
652 CONTINUE
DYK=Y(K)
Y(K)=DYK*DAKK
DO 114 I=1,N
IF (I.EQ.K) GO TO 114
DAIK=A(I,K)
CALL ROWSUM(NP1,NDIM,A(I,1),A(K,1),DAIK)
C DO 112 J=1,NP1
C 112 A(I,J)=A(I,J)-DAIK*A(K,J)
A(I,K)=DAIK
DYI=Y(I)
DYK=Y(K)
Y(I)=DYI-DAIK*DYK
114 CONTINUE
DO 116 I=1,N
116 A(I,K)=A(I,K)*DAKK
A(K,K)=DAKK
118 CONTINUE
DO 122 K=1,N
L=N+1-K
KI=ICHG(L)
IF (L.EQ.KI) GO TO 122
DO 120 I=1,N
TEMP = A(I,L)
A(I,L) = A(I,KI)
120 A(I,KI) = TEMP
122 CONTINUE
IF(IDET.NE.0) DPIVOT=X(NP1)
124 RETURN
END

```

```

      COMPLEX FUNCTION EXPON(L,MS)
C  CREATE EXP(I*L*PHIM)
      DIMENSION CS(4,6),SN(4,6)
      COMMON/RNAM/CS,SN
      IL=IABS(L)
      IF(L) 16,15,17
15  EXPON=CMPLX(1.,0.)
      GO TO 18
16  A=CS(MS,IL)
      B=SN(MS,IL)
      EXPON=CMPLX(A,-B)
      GO TO 18
17  A=CS(MS,L)
      B=SN(MS,L)
      EXPON=CMPLX(A,B)
18  CONTINUE
      RETURN
      END

```



```

SUBROUTINE SWA(I,J)
INTEGER P,Q,QS
COMPLEX EXPON
COMPLEX*16 CMS,CM1,CS1,CS2
COMPLEX*16 TKN(441)
COMPLEX*16 ZLN
COMPLEX*16 XNLQ
COMPLEX*16 YKM(3,4)
COMMON/SPAR/AKCI(6),TAU(6),SMLA(6),DMS(6),AK(4),AC(4),BJ(4),
1CAPK,CAPC
COMMON/INTER/NSY,NBSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
1MFLEX,MCON,MAER,MFUS,NBC,NFLAP,NFEA,NCT,NCON,NFFB,
2NAS,NHC,NVI,NSP,MAXN,NES,MSC,NEGN,IPCT,NIT,NER,NORM,
3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NCSH,
4NFP1,MXSM1,MXT2P1,MXKQ,MXCPL,MXCSH,MXCPM,MXCPK,MXSMB,
5NEBC,NEBSC,MFASH,MXFAB,NFUS,NRBD,NRIFC,MXQ,NEIFC,
6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MODE
COMMON/NLEAD/MLEL,NLEL,MCTY
COMMON/ROTF/OM1,OM2,OMT
COMMON/FREF/CMS,CM1,CS1,CS2
COMMON/SWASH/SWGJ,SWEI,SWM,SWR
COMMON/TKN1/TKN
KSM1=I-NFP1
IF(I,NE,J) GO TO 30
DO 17 L=1,MXT2P1
LS=L-MAXN=1
LL=(L-1)*NRIFC+L
17 TKN(LL)=ZLN(LS,I)
DO 20 MS=1,NB
DO 20 L=1,MXT2P1
LS=L-MAXN=1
CFDL=1.0+DMS(MS)*(1.+(LS*LS=1)/(1.+LS*LS*SWGJ/SWEI))/SWR
LL=(L-1)*NRIFC+MXT2P1+(MS-1)*NRBD+NCOLS+NFUS
IF(MFLEX.EQ.0) GO TO 21
IF(MCON.EQ.0) GO TO 20
LL=LL+3
IF(MCTY.GT.0) LL=LL+3
TKN(LL)=EXPON(LS,MS)*CFDL
GO TO 20
21 LL=LL+MCT
IF(MCT.EQ.0) GO TO 20
CS1=CMS-CM1*KSM1*OM1
YKM(I,MS)=SMLA(MS)*(1+CS1*TAU(MS))
TKN(LL)=YKM(I,MS)*EXPON(LS,MS)*CFDL
20 CONTINUE
GO TO 50
30 IMJ=I-J
DO 18 L=1,MXT2P1
DO 18 Q=1,MXT2P1
IF(L.EQ.Q) GO TO 18

```

```
LMQ=L-Q  
IF(IMJ.NE.LMQ) GO TO 10  
LS=L-MAXN-1  
QS=Q-MAXN-1  
LL=(L-1)*NRIFC+Q  
TKN(LL)=XNLQ(I,LS,QS)  
10 CONTINUE  
50 RETURN  
END
```

```

SUBROUTINE SWB(I,J)
INTEGER P,Q,QS
REAL*8 CX(75)
COMPLEX EXPON
COMPLEX*16 ULN,S(216)
COMPLEX*16 TKN(441)
COMPLEX*16 B(648),SMLB(108),SMLC(108),SMLD(108)
COMPLEX*16 CTB(54),FAB(54),FLB(54)
COMPLEX*16 CTB1(9),CTB2(9),CTB3(9),FAB1(9),FAB3(9),FLB1(9),FLB2(9)
COMPLEX*16 SMLE(108),FSB(54)
COMPLEX*16 SMLF(108),SMLG(108)
COMPLEX*16 FSB1(9),FSB2(9),FSB3(9),FAB4(9),FLB4(9),CTB4(9)
COMMON/BTS/B,SMLB,SMLC,SMLD,CTB,FAB,FLB,CTB1,CTB2,CTB3,FAB1,
1FAB3,FLB1,FLB2,SMLE,FSB,FSB1,FSB2,FSB3,FAB4,FLB4,CTB4,SMLF,SMLG
COMMON/SWASH/SWGJ,SWEI,SWM,SWR
COMMON/INTER/NSY,NBSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
1MFLEX,MCUN,MAER,MFUS,NBC,NFLAP,NFEA,NCT,NCON,NFFB,
2NAS,NHC,NVI,NSP,MAXN,NES,MSC,NEGN,IPCT,NIT,NER,NORM,
3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NC9B,
4NFP1,MXSMI,MXT2P1,MXKQ,MXCPL,MXC9B,MXCPM,MXCPK,MX9MB,
5NEBC,NEBC,MFASB,MXFAB,NFUS,NRBD,NRIFC,MXQ,NEIFC,
6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MODE
COMMON/SPAR/AKCI(6),TAU(6),SMLA(6),DMS(6),AK(4),AC(4),BJ(4),
1CAPK,CAPC
COMMON/CFLEX/CX
COMMON/TKN1/TKN
COMMON/NLEAD/MLEL,NLEL,MCTY
COMMON/S81/S
NONO=1
IMJ=I-J
JMNC=(J-1)*12*NCOLS
IF(MFLEX.EQ.0.AND.IMJ.NE.0) GO TO 23
IF(MFLEX.EQ.0) GO TO 3
IF(MCTY.EQ.0) GO TO 3
IF(IMJ.NE.0) GO TO 23
3 IF(IMJ.NE.0) NONO=0
IF(NBP.EQ.0) GO TO 13
KSML=I-NFP1
NPMK=NPS-KSML
DO 14 Q=1,MXT2P1
QS=Q-MAXN-1
NMKQ=NPMK-QS
IF(NMKQ.EQ.0) GO TO 5
RNMKQ=1.0*NMKQ
RFA=RNMKQ/NBP
NFA=NMKQ/NBP
DIF=RFA-1.0*NFA
IF(DIF.GE.0.0) GO TO 2
DIF=-DIF
2 IF(DIF.GT..05) GO TO 14

```

```

5 DFQR=1.0+DMS(1)*(1.+(QS*QS-1.)/(1.+(QS*QS*SWGJ/SWEI)))/SWR
  IF(MFLEX.EQ.0) GO TO 7
  IF(MCON.EQ.0) GO TO 14
  IF(MCTY.GT.0) GO TO 6
  DU 8 IQ=1,NCOLS
  LL=NEIFC+NEISC+(IQ-1)*NRIFC+Q
  L=(J-1)*NCOLS*MXSMI+(I-1)*NCOLS+IQ
  A TKN(LL)=-NBP*DFQR*FLA(L)
  LL=NEIFC+NEISC+NCOLS*NRIFC+Q
  TKN(LL)=+NBP*DFQR*CX(3)*NONO
  TKN(LL+NRIFC)=+NBP*DFQR*CX(8)*NONO
  TKN(LL+NRIFC*2)=+NBP*DFQR*CX(12)*NONO
  GO TO 14
6 LL=NEIFC+NEISC+NRIFC*NCOLS+5*NRIFC+Q
  TKN(LL)=-NBP*DFQR
  GO TO 14
7 LL=NEIFC+NEISC+NRIFC*NCOLS+Q
  IF(MCT.EQ.0) GO TO 14
  TKN(LL)=-NBP*DFQR/SMLA(1)
14 CONTINUE
  GO TO 23
13 DU 21 MS=1,NB
  DU 21 P=1,MXT2P1
  QS=P*MAXN=1
  DFQR=1.0+DMS(MS)*(1+(QS*QS-1)/(1+(QS*QS*SWGJ/SWEI)))/SWR
  IF(MFLEX.EQ.0) GO TO 22
  IF(MCON.EQ.0) GO TO 21
  IF(MCTY.GT.0) GO TO 25
  DU 24 IQ=1,NCOLS
  LL=NEIFC+NEISC+(MS-1)*NEITC+(IQ-1)*NRIFC+P
  L=(J-1)*NCOLS*MXSMI+(I-1)*NCOLS+IQ
  24 TKN(LL)=-EXPON(-QS,MS)*DFQR*FLA(L)
  LL=NEIFC+NEISC+(MS-1)*NEITC+NCOLS*NRIFC+P
  TKN(LL)=+EXPON(-QS,MS)*DFQR*CX(3)*NONO
  LL=LL+NRIFC
  TKN(LL)=+EXPON(-QS,MS)*DFQR*CX(8)*NONO
  LL=LL+NRIFC
  TKN(LL)=-EXPON(-QS,MS)*DFQR*CX(12)*NONO
  GO TO 21
25 LL=NEIFC+NEISC+(MS-1)*NEITC+NRIFC*NCOLS+5*NRIFC+P
  TKN(LL)=-EXPON(-QS,MS)*DFQR
  GO TO 21
22 LL=NEIFC+NEISC+(MS-1)*NEITC+NRIFC*NCOLS+P
  IF(MCT.EQ.0) GO TO 21
  TKN(LL)=-EXPON(-QS,MS)*DFQR/SMLA(MS)
21 CONTINUE
23 CONTINUE
  IF(MFUS.EQ.0) GO TO 52
  JMI=-IMJ
  DU 50 IPP=1,MXT2P1

```

```
LS=IPP-MAXN=1
IF(JMI,NE,LS) GO TO 50
DO 51 IQQ=1,NCOLS
LLP=JMNC+(IQQ-1)*12+1
LL=NEIFC+(IQQ-1)*NRIFC+IPP
51 TKN(LL)=ULN(LS,KSML)*S(LLP)
50 CONTINUE
52 CONTINUE
RETURN
END
```

```

COMPLEX FUNCTION ZLN*16(LS,I)
REAL*8 C3,C4
COMPLEX EXCHI
COMPLEX*16 CLNJ,C1,C2,C5,C6,C7,C8,C9,C10
COMPLEX*16 CMS,CM1,CS1,CS2,VN,VN1,VN2
COMMON/SWASH/SWGJ,SWEI,SWM,SWR
COMMON/INTER/NSY,NBSEC,NFSEC,NR,NBP,MFLAP,MFEA,MCT,
1MFLEX,MCON,MAER,MFUS,NBC,NFLAP,NFEA,NCT,NCON,NFFB,
2NAS,NHC,NVI,NSP,MAXN,NES,MSC,NEGN,IPCT,NIT,NER,NORM,
3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NCSB,
4NFP1,MXSMI,MXT2P1,MXKQ,MXCPL,MXCSB,MXCPM,MXCPK,MXSM6,
5NEBC,NESEC,MFASB,MXFAB,NFUS,NRBD,NRIFC,MXQ,NEIFC,
6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MODE
COMMON/SPAR/AKCI(6),TAU(6),SMLA(6),DMS(6),AK(4),AC(4),BJ(4),
1CAPK,CAPC
COMMON/NLEAD/MLEL,NLEL,MCTY
COMMON/FREF/CMS,CM1,CS1,CS2
COMMON/ROTF/OM1,OM2,OMT
COMMON/SPAR1/AKT(4),ACT(4),AKP(4),ACP(4)
R=SWR
KSML=I-NFP1
CS1=CMS-CM1*KSML*UM1
CS2=CS1*CS1
C1=SWM*(CS2-OMT*LS*CM1*CS1-LS*LS*OM2)
C2=DCMPLX(0.00,0.00)
C9=DCMPLX(0.00,0.00)
CFL=1.+(LS*LS-1)/(1+LS*LS*SWGJ/SWEI)
CFLR=CFL/SWR
DO 10 JJ=1,NES
CLNJ=AK(JJ)+CS1*AC(JJ)-CM1*LS*OM1*AC(JJ)
C2=C2+CLNJ
10 C9=C9+CLNJ*(1-BJ(JJ))*CFLR
C10=C2
C2=C9
IF(MSC.EQ.0) GO TO 11
C2=C10
C5=CAPK+CAPC*(CS1-CM1*LS*OM1)
C6=DCMPLX(0.00,0.00)
C7=DCMPLX(0.00,0.00)
DO 8 JJ=1,NES
C8=AK(JJ)+CS1*AC(JJ)-CM1*LS*OM1*AC(JJ)
C6=C6+C8*EXCHI(LS,0,JJ)*(1-BJ(JJ))*CFLR)
8 C7=C7+C8*EXCHI(0,LS,JJ)*(1-BJ(JJ))*CFLR)
C2=C9-C7*C6/(C5+C2)
11 IF(MAXN.EQ.1) GO TO 12
KX=1-LS*LS
C3=2.00*3.141592654*LS*LS*KX*KX
C4=R*R*R*(1./SWGJ+LS*LS/SWEI)
C3=C3/C4
ZLN=C1+C2+DCMPLX(C3,0.00)

```

```

GO TO 13
12 ZLN=C1+C2
13 CONTINUE
   VN=DCMPLX(0.00,0.00)
   VN1=DCMPLX(0.00,0.00)
   VN2=DCMPLX(0.00,0.00)
   DO 5 JJ=1,NES
   VN=VN+AKT(JJ)+(CS1-CM1*LS*DM1)*ACT(JJ)
   VN1=VN1+(AK(JJ)+(CS1-CM1*LS*DM1)*AC(JJ))*BJ(JJ)*(R=BJ(JJ)*CFL)
5  VN2=VN2+AKP(JJ)+(CS1-CM1*LS*DM1)*ACP(JJ)
   ZLN=ZLN+(LS*LS*VN-VN1*CFL+CFL*CFL*VN2)/(R*R)
RETURN
END

```

```

COMPLEX FUNCTION XNLQ*16(I,LS,QS)
INTEGER QS
COMPLEX EXCHI
COMPLEX*16 CS1,CS2,CM1,CM3,XN,XN1,XN2,XN3
COMPLEX*16 CS,WN,WN1,WN2
COMMON/SWASH/SWGJ,SWEI,SWM,SWR
COMMON/INTER/NSY,NBSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
1MFLEX,MCON,MAER,MFUS,NBC,NFLAP,NFEA,NCT,NCON,NFFB,
2NAS,NHC,NVI,NSP,MAXN,NES,MSC,NEGN,IPCT,NIT,NER,NORM,
3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRS,NPD,NSK,NCOLS,NCSB,
4NFP1,MXSMI,MXT2P1,MXKQ,MXCPL,MXCSB,MXCPM,MXCPK,MXSMB,
5NEHC,NESHC,MFASH,MXFAB,NFUS,NRBD,NRIFC,MXQ,NEIFC,
6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MODE
COMMON/SPAR/AKCI(6),TAU(6),SMLA(6),DMS(6),AK(4),AC(4),BJ(4),
1CAPK,CAPC
COMMON/NLEAD/MLEL,NLEL,MCTV
COMMON/FREF/CM3,CM1,CS1,CS2
COMMON/ROT/OM1,OM2,OMT
COMMON/SPAR1/AKT(4),ACT(4),AKP(4),ACP(4)
KSML=I-NFP1
CS1=CM3-CM1*KSML*UM1
IF(QS.EQ.LS) GO TO 15
XNLQ=DCMPLX(0.D0,0.D0)
CFQ=1.+(QS*QS-1)/(1.+QS*QS*SWGJ/SWEI)
CFL=1.+(LS*LS-1)/(1+LS*LS*SWGJ/SWEI)
CFQR=CFQ/SWR
CFLR=CFL/SWR
DO 10 JJ=1,NES
10 XNLQ=XNLQ+(AK(JJ)+(CS1-CM1*QS*OM1)*AC(JJ))*EXCHI(LS,QS,JJ)*
1(1.-BJ(JJ)*CFQR)
IF(MSC.EQ.0) GO TO 16
CS=CAPK+CAPC*(CS1-CM1*QS*UM1)
XN1=DCMPLX(0.D0,0.D0)
XN2=DCMPLX(0.D0,0.D0)
XN3=DCMPLX(0.D0,0.D0)
DO 12 JJ=1,NES
XN=AK(JJ)+(CS1-CM1*QS*OM1)*AC(JJ)
XN1=XN1+XN
XN2=XN2+XN*EXCHI(LS,0,JJ)*(1.-BJ(JJ)*CFLR)
12 XN3=XN3+XN*EXCHI(0,QS,JJ)*(1.-BJ(JJ)*CFQR)
XNLQ=XNLQ+XN3*XN2/(CS+XN1)
GO TO 16
15 XNLQ=DCMPLX(0.D0,0.D0)
16 CONTINUE
R=SWR
WN=DCMPLX(0.D0,0.D0)
WN1=DCMPLX(0.D0,0.D0)
WN2=DCMPLX(0.D0,0.D0)
DO 5 JJ=1,NES
WN=WN+(AKT(JJ)+(CS1-CM1*QS*UM1)*ACT(JJ))*EXCHI(LS,QS,JJ)

```



```

WN1=WN1+(AK(JJ)+(CS1-CM1*QS*OM1)*AC(JJ))*BJ(JJ)*(R=BJ(JJ)*CFQ)*
1 EXCHI(LS, QS, JJ)
5 WN2=WN2+(AKP(JJ)+(CS1-CM1*QS*UM1)*ACP(JJ))*EXCHI(LS, QS, JJ)
XNLQ=XNLQ+(QS*LS*WN-WN1*CFL+CFQ*CFL*WN2)/(R*R)
RETURN
END

```

```

COMPLEX FUNCTION ULN*16(LS, KSML)
COMPLEX*16 UN, UNN, UNC, CS1, CS2, CMS, CM1
COMPLEX EXCHI
COMMON/SPAR/AKCI(6), TAU(6), SMLA(6), DMS(6), AK(4), AC(4), BJ(4),
1 CAPK, CAPC
COMMON/ROTF/OM1, OM2, OMT
COMMON/PREF/CMS, CM1, CS1, CS2
COMMON/SWASH/SWGJ, SWEI, SWR, SWR
COMMON/INTER/NSY, NRSEC, NFSEC, NB, NBP, MFLAP, MFEA, MCT,
1 MFLEX, MGIN, MAER, MFUS, NBC, NFLAP, NFEA, NCT, NCON, NFFB,
2 NAS, NHC, NVI, NSP, MAXN, NES, MSC, NEGN, IPCT, NIT, MER, NORM,
3 IHEN, NEX, NPS, NSCH, IG, IF, NPRL, NPRB, NPD, NSK, NCOLS, NCSH,
4 NFP1, MXSMI, MXT2P1, MXKQ, MXCPL, MXCSB, MXCPH, MXCPK, MXSMR,
5 NEBC, NESHG, MFASH, MXFAB, NFUS, NRBV, NRIFC, MXQ, NEIFC,
6 NEISC, NEITC, MXTKN, NFF, MINPN, MAXPN, IBF, MUDE
COMMON/NLEAD/NLEL, NLEL, MCTY
UN =DCMPLX(0.00, 0.00)
UNN=DCMPLX(0.00, 0.00)
CFL=1.+(LS*LS-1)/(1+LS*LS*SWGJ/SWEI)
CFLR=CFL/SWR
CS1=CMS-CM1*KSML*UM1
DO 5 JJ=1, NES
UN=UN+(AK(JJ)+(CS1-CM1*LS*OM1)*AC(JJ))*EXCHI(0, LS, JJ)*
1 (1.-BJ(JJ)*CFLR)
5 UNN=UNN+(AK(JJ)+(CS1-CM1*LS*OM1)*AC(JJ))
IF(MSC.EQ.0) GO TO 6
UNC=CAPK+(CS1-CM1*LS*OM1)*CAPC
ULN=UN*UNC/(UNC+UNN)
6 IF(MSC.EQ.0) ULN=UN
RETURN
END

```

```

COMPLEX FUNCTION EXCHI (L,Q,J)
INTEGER Q
COMMON/RNAM1/CSA(4,24),SNA(4,24)
LQ=L-Q
ILQ=IAHS(LQ)
IF(LQ)16,15,17
15 EXCHI=CMPLX(1.0,0.0)
GO TO 18
16 A=CSA(J,ILQ)
B=SNA(J,ILQ)
EXCHI=CMPLX(A,-B)
GO TO 18
17 A=CSA(J,ILQ)
B=SNA(J,ILQ)
EXCHI=CMPLX(A,B)
18 CONTINUE
RETURN
END

```

```

SUBROUTINE ZTEGI(I,J)
INTEGER P,Q
COMPLEX EXPON,EXPM1,EXPP1
COMPLEX*16 CM1
COMPLEX*16 B(648),SMLB(108),SMLC(108),SMLD(108)
COMPLEX*16 CTB(54),FAB(54),FLB(54)
COMPLEX*16 CTB1(9),CTB2(9),CTB3(9),FAB1(9),FAB3(9),FLB1(9),FLB2(9)
COMPLEX*16 SMLF(108),FSB(54)
COMPLEX*16 SMLP(108),SMLG(108)
COMPLEX*16 FSB1(9),FSB2(9),FSB3(9),FAB4(9),FLB4(9),CTB4(9)
COMPLEX*16 TKN(441)
COMMON/BTS/B,SMLB,SMLC,SMLD,CTB,FAB,FLB,CTB1,CTB2,CTB3,FAB1,FAB3,
1FLB1,FLB2,SMLF,FSB,FSB1,FSB2,FSB3,FAB4,FLB4,CTB4,SMLP,SMLG
COMMON/TKN1/TKN
COMMON/INTER/NSY,NBSEC,NFSEC,NB,NBP,MFLAP,MFEA,MCT,
1MFLEX,MCON,MAER,MFUS,NBC,NFLAP,NFEA,NCT,NCON,NFFB,
2NAS,NHC,NVI,NSP,MAXN,NES,MSC,NEGN,IPCT,NIT,NER,NORM,
3IREM,NEX,NPS,NSCH,IG,IF,NPRL,NPRB,NPD,NSK,NCOLS,NCSB,
4NFP1,MXSMI,MXT2P1,MXKQ,MXCPL,MXCSB,MXCPM,MXCPK,MXSMB,
5NEBC,NESBC,MFASB,MXFAB,NFUS,NRBD,NRIFC,MXQ,NEIFC,
6NEISC,NEITC,MXTKN,NFF,MINPN,MAXPN,IBF,MODE
COMMON/NLEAD/MLEL,NLEL,MCTY
CM1=DCMPLX(0.00,1.00)
NMK=J-1
KB=NEIFC+NEISC+MXT2P1+NFUS
KBB=KB+NCOLS+NRIFC
NMKP1=NMK+1
NMKM1=NMK-1
NPK=NPS-I+NFP1
NBM1=(NB-1)*NRBD
LSMA=(J-1)*NESBC+(I-1)*MXCSB
LLAR=(J-1)*NEBC+(I-1)*MXCPM
IF(MFLEX.EQ.0) GO TO 10
WRITE(6,900)
900 FORMAT(/,9X,'GIMBALLED OR TEETERING ROTOR, MFLX MUST EQUAL ZERO')
GO TO 90
10 IF(NBC.EQ.2)GO TO 13
IF(NBP.EQ.0)GO TO 11
IF(NBP.LE.2)GO TO 12
GO TO 16
11 IF(NB.GT.2)GO TO 16
12 WRITE(6,901)
901 FORMAT(/,9X,'GIMBALLED ROTOR MUST HAVE MORE THAN TWO BLADES')
GO TO 90
13 IF(NBP.EQ.0)GO TO 14
IF(NBP.NE.2)GO TO 15
GO TO 16
14 IF(NB.EQ.2)GO TO 16
15 WRITE(6,902)
902 FORMAT(/,9X,'TEETERING ROTOR MUST HAVE TWO BLADES')

```

```

GO TO 90
16 NE=2
   NA=1
   IF(NHC, EQ, 1) GO TO 17
   NE=1
   NA=0
17 N1=1
   N2=6
   DO 22 NN=1, NE
   IF(NN, EQ, 1) GO TO 18
   N1=1
   N2=2
18 DO 19 Q=1, NCOLS
   L3=LLAR+(Q-1)*12+3
   L10=L3+7
   KK=KB+(Q-1)*NRIFC+N2
19 TKN(KK)=NA*B(L3)+N1*CM1*B(L10)
   L3=LSMA+3
   L10=L3+7
   KK=KB+N2
   IF(MCT, EQ, 0) GO TO 20
   TKN(KK)=NA*SMLB(L3)-N1*CM1*SMLB(L10)
   KK=KK+NRIFC
20 IF(MFEA, EQ, 0) GO TO 21
   TKN(KK)=NA*SMLC(L3)-N1*CM1*SMLC(L10)
   KK=KK+NRIFC
21 IF(MFLAP, EQ, 0) GO TO 23
   TKN(KK)=NA*SMLD(L3)-N1*CM1*SMLD(L10)
   KK=KK+NRIFC
23 IF(MLEL, EQ, 0) GO TO 22
   TKN(KK)=NA*SMLE(L3)-N1*CM1*SMLE(L10)
22 CONTINUE
   IF(NBP, NE, 0) GO TO 40
   IF(NHC, EQ, 2) GO TO 28
   DO 24 Q=1, NRBD
   KT=KB+(Q-1)*NRIFC+2
   KK=KT+4
   K2=KT+NBM1
   K6=KK+NBM1
   TKN(K2)=TKN(KT)
24 TKN(K6)=TKN(KK)
   DO 25 MS=2, NB
   EXPM1=EXPON(NMKM1, MS)
   EXPP1=EXPON(NMKP1, MS)
   MSHIFT=(MS-1)*NEITC+(MS-2)*NRBD
   DO 25 Q=1, NRBD
   KT=KB+(Q-1)*NRIFC+2
   KK=KT+4
   K2=KT+MSHIFT
   K6=KK+MSHIFT

```

```

K22=K2+NRBD
K66=K6+NRBD
TKN(K2)=-TKN(KT)*EXPP1
TKN(K6)=-TKN(KK)*EXPM1
TKN(K22)=-TKN(K2)
25 TKN(K66)=-TKN(K6)
GO TO 100
28 DO 30 Q=1,NCOLS
L11=LLAR+(Q-1)*12+11
KK=KB+(Q-1)*NRIFC+NRBD+6
30 TKN(KK)=H(L11)
L11=LSMA+11
KK=KBB+NRBD+6
IF(MCT.EQ.0)GO TO 31
TKN(KK)=-SMLB(L11)
KK=KK+NRIFC
31 IF(MFEA.EQ.0)GO TO 32
TKN(KK)=-SMLC(L11)
KK=KK+NRIFC
32 IF(MFLAP.EQ.0) GO TO 34
TKN(KK)=-SMLD(L11)
KK=KK+NRIFC
34 IF(MLEL.EQ.0) GO TO 33
TKN(KK)=-SMLE(L11)
33 EXPM1=EXPON(NMKM1,2)
DO 35 Q=1,NRBD
KT=KB+(Q-1)*NRIFC+6
KK=KT+NEITC
35 TKN(KK)=-TKN(KT)*EXPM1
DO 38 Q=1,NRBD
KT=KB+(Q-1)*NRIFC+NRBD+6
KK=KT+NEITC
38 TKN(KK)=TKN(KT)*EXPM1
GO TO 100
40 N1=-1
N2=6
DO 50 NN=1,NE
IF(NN.EQ.1)GO TO 42
N1=1
N2=2
42 NPMK=NPK+N1
INMK=IABS(NPMK)
RINMK=1.0*INMK
RFA=RINMK/NBP
NFA=INMK/NBP
DIF=ABS(RFA-1.0*NFA)
IF(DIF.GT.(.05))GO TO 50
DO 44 Q=1,NCOLS
L4=LLAR+(Q-1)*12+4
KK=KB+(Q-1)*NRIFC+N2

```

```

L11=L4+7
44 TKN(KK)=NA*B(L4)+N1*CM1*B(L11)
L4=LSMA+4
KK=KBB+N2
L11=L4+7
IF(MCT.EQ.0)GO TO 45
TKN(KK)=NA*SMLB(L4)-N1*CM1*SMLB(L11)
KK=KK+NRIFC
45 IF(MFEA.EQ.0)GO TO 46
TKN(KK)=NA*SMLC(L4)-N1*CM1*SMLC(L11)
KK=KK+NRIFC
46 IF(MFLAP.EQ.0) GO TO 47
TKN(KK)=NA*SMLD(L4)-N1*CM1*SMLD(L11)
KK=KK+NRIFC
47 IF(MLEL.EQ.0)GO TO 50
TKN(KK)=NA*SMLE(L4)-N1*CM1*SMLE(L11)
50 CONTINUE
GO TO 100
90 STOP
100 RETURN
END

```

```

SUBROUTINE POLAR
COMPLEX QXJ,QYJ,QZJ,DTX,DTY,DTZ
COMMON/UVTEM/UXJ,UYJ,QZJ
DTX=QXJ*CMPLX(0.0,-1.0)
DTY=QYJ*CMPLX(0.0,-1.0)
DTZ=QZJ*CMPLX(0.0,-1.0)
DXR=UXJ
DZR=QZJ
DXI=DTX
DYI=DTY
DZI=DTZ
IF(DXR.NE.0.0) GO TO 2
IF(DXI.NE.0.0) GO TO 2
DXA=0.0
GO TO 3
2 DXA=ATAN2(DXI,DXR)
3 IF(DYR.NE.0.0) GO TO 4
IF(DYI.NE.0.0) GO TO 4
DYA=0.0
GO TO 5
4 DYA=ATAN2(DYI,DYR)
5 IF(DZR.NE.0.0) GO TO 6
IF(DZI.NE.0.0) GO TO 6
DZA=0.0
GO TO 7
6 DZA=ATAN2(DZI,DZR)
7 CONTINUE
DXR=SQRT(DXR*DXR+DXI*DXI)
DZR=SQRT(DZR*DZR+DZI*DZI)
QXJ=CMPLX(DXR,DXA)
QYJ=CMPLX(DYR,DYA)
QZJ=CMPLX(DZR,DZA)
RETURN
END

```

*FORTRAN CALLABLE COMPLEX FUNCTION TO OBTAIN DOT PRODUCTS.
 *ARGUMENT LIST IS (N,A,B), WHERE N IS THE DIMENSION OF THE VECTORS
 *A AND B. A IS PRESUMED SPARSE FOR MAXIMUM PROGRAM SPEED.
 *INTERMEDIATE RESULTS ARE CARRIED IN DOUBLE PRECISION AND THE
 *FUNCTION MAY BE DECLARED DOUBLE PRECISION COMPLEX, IF DESIRED.
 *

```

      SPACE 2
#INCR   EQU  0
#COMPR  EQU  1
#INDEX  EQU  2
#N      EQU  2
#A      EQU  3
#B      EQU  4
#MAXR   EQU  4

      SPACE
#REAL   EQU  0
#IMAG   EQU  2
#ZERO   EQU  4
#TEMP   EQU  6

      SPACE 2
A       DSECT
AREAL   DS    D
AIMAG   DS    D
B       DSECT
BREAL   DS    D
BIMAG   DS    D

      EJECT
CDOT    CSECT
      SAVE  (2,#MAXR),,*
      USING CDOT,15
      LM    #N,#B,0(1)
      USING A,#A
      USING B,#B
      L     #COMPR,0(#N)
      BCTR  #COMPR,0
      SLA   #COMPR,4
      LA    #INCR,16
      SR    #INDEX,#INDEX
      SDR   #REAL,#REAL
      SDR   #IMAG,#IMAG
      SDR   #ZERO,#ZERO

      SPACE 2
LOOP    CD    #ZERO,A(#INDEX)
        BNF  CONTINUE
        BXLE #INDEX,#INCR,LOOP
EXIT    RETURN (2,#MAXR)

      SPACE
CONTINUE LD    #TEMP,AREAL(#INDEX)
        MD    #TEMP,BREAL(#INDEX)
        ADR   #REAL,#TEMP

```



```

LD    #TEMP,AIMAG(#INDEX)
MD    #TEMP,BIMAG(#INDEX)
SDR   #RFAL,#TEMP
LD    #TEMP,AREAL(#INDEX)
MD    #TEMP,BIMAG(#INDEX)
ADR   #IMAG,#TEMP
LD    #TEMP,AIMAG(#INDEX)
MD    #TEMP,BREAL(#INDEX)
ADR   #IMAG,#TEMP
RXLF  #INDEX,#INCR,LOOP
B     EXIT
SPACE 2
LTORG
END ← (Start of RCDOT)

```

*FORTRAN CALLABLE COMPLEX FUNCTION TO OBTAIN DOT PRODUCTS.
*ARGUMENT LIST IS (N,A,B), WHERE N IS THE DIMENSION OF THE VECTORS
*A AND B. A IS PRESUMED SPARSE FOR MAXIMUM PROGRAM SPEED.
*A IS A REAL VECTOR, WHILE B IS COMPLEX.
*INTERMEDIATE RESULTS ARE CARRIED IN DOUBLE PRECISION AND THE
*FUNCTION MAY BE DECLARED DOUBLE PRECISION COMPLEX, IF DESIRED.
*

```

SPACE 2
#INCR EQU 0
#COMPR EQU 1
#INDEXA EQU 2
#INDEXB EQU 3
#N EQU 3
#A EQU 4
#B EQU 5
#MAXR EQU 5
SPACE 2
#REAL EQU 0
#IMAG EQU 2
#ZERO EQU 4
#TEMP EQU 6
SPACE 2
A DSFCT
B DSFCT
BREAL DS D
RIMAG DS D
EJECT
RCDOT CSECT
SAVE (2,#MAXR),,*
USING RCDOT,15
LM #N,#B,0(1)
USING A,#A
USING B,#B
L #COMPR,0(#N)
RCTR #COMPR,0
SLA #COMPR,3

```

```

      LA    #INCR,R
      SP    #INDEXA,#INDEXA
      SDR   #RFAL,#REAL
      SDR   #IMAG,#IMAG
      SDR   #ZERO,#ZERO
      SPACE 2
LOOP   CD    #ZERO,A(#INDEXA)
      BNF   CONTINUE
      BXLE  #INDEXA,#INCR,LOOP
EXIT   RETURN (2,#MAXR)
      SPACE 2
CONTINUE LA  #INDEXR,0(#INDEXA,#INDEXA)
      LD    #TEMP,A(#INDEXA)
      MD    #TEMP,BREAL(#INDEXR)
      ADR   #RFAL,#TEMP
      LD    #TEMP,A(#INDEXA)
      MD    #TEMP,RIMAG(#INDEXR)
      ADR   #IMAG,#TEMP
      BXLE  #INDEXA,#INCR,LOOP
      B     EXIT
      SPACE 2
      LTORG
      END

```

(Start of ROWSUM)

*FORTRAN CALLABLE SUBROUTINE TO PERFORM MATRIX ROW OPERATIONS.
 *ARGUMENT LIST IS (N,NDIM,A,R,X).
 *THE ROW OPERATION $A=A-X*B$ IS PERFORMED, WHERE A, B, AND X ARE
 *DOUBLE PRECISION COMPLEX.
 *NDIM IS THE COLUMN DIMENSION OF THE MATRICES, AND N IS THE NUMBER OF
 *ELEMENTS TO BE OPERATED ON IN THE ROWS. THE INDEXING SCHEME IS
 *THEREFORE $A(I)=A(I)-X*B(I)$, $I=1,1+(N-1)*NDIM,NDIM$
 *

```

      SPACE 2
#INCR   EQU 0
#COMPR  EQU 1
#INDEX  EQU 2
#N      EQU 1
#NDIM   EQU 2
#A      EQU 3
#B      EQU 4
#X      EQU 5
#MAXR   EQU 5
      SPACE 2
#ATEMP  EQU 0
#BTEMP  EQU 2
#XRFAL  EQU 4
#XIMAG  EQU 6
      SPACE 2
A       DSECT
AREAL   DS  D
AIMAG   DS  D

```

```

A          DSECT
BREAL     DS      D
RIMAG     DS      D
          EJECT
ROWSUM    CSECT
          USING ROWSUM,15
          SAVE  (14,#MAXR),,*
          LM    #N,#X,0(1)
          USING A,#A
          USING B,#B
          L     #NDIM,0(#NDIM)
          SLA  #NDIM,4
          L     #COMPR,0(#N)
          BCTR #COMPR,0
          MR   #COMPR-1,#NDIM
          LR   #INCR,#NDIM
          SR   #INDEX,#INDEX
          LD   #XREAL,0(#X)
          LD   #XIMAG,R(#X)
          SPACE 2
LOOP      LD   #ATEMP,RIMAG(#INDEX)
          MDR  #ATEMP,#XIMAG
          AD   #ATEMP,AREAL(#INDEX)
          LD   #BTEMP,BREAL(#INDEX)
          MDR  #BTEMP,#XREAL
          SDR  #ATEMP,#BTEMP
          STD  #ATEMP,AREAL(#INDEX)
          LD   #ATEMP,AIMAG(#INDEX)
          LD   #BTEMP,BREAL(#INDEX)
          MDR  #BTEMP,#XIMAG
          SDR  #ATEMP,#BTEMP
          LD   #BTEMP,RIMAG(#INDEX)
          MDR  #BTEMP,#XREAL
          SDR  #ATEMP,#BTEMP
          STD  #ATEMP,AIMAG(#INDEX)
          BXLE #INDEX,#INCR,LOOP
          SPACE
          RETURN (2,#MAXR),T
          LTORG
          END
OVERLAY ALPHA ← (Start of Overlay Structure)
INSERT ARI
INSERT ARR
INSERT COEFFS
INSERT BTS
INSERT SSI
INSERT TKN1
OVERLAY BETA
INSERT BAERO
INSERT FAERO

```

INSERT ACRD
INSERT ACOEFF
INSERT SETUP
INSERT SEC PAR
INSERT LOADIN
INSERT TABLU
OVERLAY BETA
INSERT ELAST
INSERT RIGID
INSERT STIFF
INSERT BEND
INSERT MLPC2
INSERT MLCC2
INSERT CDDT
INSERT RCDOT
OVERLAY GAMMA
INSERT BARRAY
INSERT RMASS
INSERT BLARD
OVERLAY GAMMA
INSERT SARRAY
INSERT FMASS
INSERT FUARD
OVERLAY PETA
INSERT EPSOLN
INSERT SURMA
INSERT SURMB
INSERT SURMF
INSERT SURMG
INSERT ZTEGI
INSERT EXPON
INSERT TKNS
INSERT SWA
INSERT SWB
INSERT ZLN
INSERT XNLD
INSERT ULN
INSERT FXCHI
OVERLAY ALPHA
INSERT SOLVE
INSERT DCMAT
INSERT SWAPS
INSERT ROWSUM
INSERT FPRSET
ENTPY MAIN



REFERENCES

1. Sutton, Lawrence R. and Gangwani, Santu T., THE DEVELOPMENT AND APPLICATION OF AN ANALYSIS FOR THE DETERMINATION OF COUPLED TAIL ROTOR/HELICOPTER AIR RESONANCE BEHAVIOR, USAAMRDL-TR-75-35, Eustis Directorate, US Army Air Mobility Research and Development Laboratory, Fort Eustis, Virginia, August, 1975, AD A014989.
2. Sutton, Lawrence R. and White, Richard P. Jr., THE DEVELOPMENT AND APPLICATION OF AN ANALYSIS FOR THE STABILITY BEHAVIOR OF BEARINGLESS MAIN ROTOR SYSTEMS, Letter Report SRL 14-77-3, RASA Division, Systems Research Laboratories, Inc., 1055 J. Clyde Morris Blvd., Newport News, Virginia.
3. Davis, John M., ROTORCRAFT FLIGHT SIMULATION WITH AEROELASTIC ROTOR AND IMPROVED AERODYNAMIC REPRESENTATION, Volume II - User's Manual, Bell Helicopter Company; USAAMRDL-TR-74-10B, Eustis Directorate, US Army Air Mobility Research and Development Laboratory, Fort Eustis, Virginia, June 1974, AD 782756.