A Reproduced Copy

OF

N86-24959

LIBRARY COPY

NOV 4 1969

LANGLEY RESEARCH CENTER
LIBRARY, NASA
HAMPTON, VIRGINIA

Reproduced for NASA
by the
NASA Scientific and Technical Information Facility

FFNo 672 Aug 65
FLUTTER PREDICTION FOR A WING WITH ACTIVE AILERON CONTROL

FINAL REPORT

PRINCIPAL INVESTIGATORS
Kevin Penning
Doral E. Sandlin

June 1983

California Polytechnic State University
Aeronautical Engineering Department
San Luis Obispo, California

Grant No. NCC4-1
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>LIST OF TABLES</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIST OF FIGURES</td>
<td>iv</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>v</td>
</tr>
<tr>
<td>NOMENCLATURE</td>
<td>vi</td>
</tr>
<tr>
<td>CHAPTER</td>
<td></td>
</tr>
<tr>
<td>1 INTRODUCCION</td>
<td>1</td>
</tr>
<tr>
<td>2 VIBRATION MODEL OF WING IN STILL AIR</td>
<td>2</td>
</tr>
<tr>
<td>3 AERODYNAMIC INFLUENCE COEFFICIENTS</td>
<td>6</td>
</tr>
<tr>
<td>4 CONTROL LAW</td>
<td>12</td>
</tr>
<tr>
<td>5 INCLUDING CONTROL LAW IN VIBRATION EQUATIONS</td>
<td>13</td>
</tr>
<tr>
<td>6 ANALYSIS OF DAST ARW-I</td>
<td>14</td>
</tr>
<tr>
<td>7 FLIGHT TEST DATA</td>
<td>26</td>
</tr>
<tr>
<td>8 RESULTS</td>
<td></td>
</tr>
<tr>
<td>9 CONCLUSIONS</td>
<td>26</td>
</tr>
<tr>
<td>APPENDIX</td>
<td></td>
</tr>
<tr>
<td>A TABLES AND FIGURES</td>
<td>29</td>
</tr>
<tr>
<td>B INPUT DOCUMENTATION FOR PROGRAM &quot;SAFSS&quot;</td>
<td>38</td>
</tr>
<tr>
<td>C LISTING OF PROGRAM &quot;SAFSS&quot;</td>
<td>47</td>
</tr>
<tr>
<td>D SAMPLE INPUT FILE</td>
<td>58</td>
</tr>
<tr>
<td>E SAMPLE OUTPUT</td>
<td>93</td>
</tr>
<tr>
<td>BIBLIOGRAPHY</td>
<td>104</td>
</tr>
</tbody>
</table>

ii
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Natural Frequencies of Vibration Modes</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>Vibration Frequencies for FSS Off</td>
<td>30</td>
</tr>
<tr>
<td>3</td>
<td>Vibration Frequencies for FSS On</td>
<td>31</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>-------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>1</td>
<td>DAST ARW-1 Vehicle Planform</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>Flow Diagram of Program &quot;SAFSS&quot;</td>
<td>31</td>
</tr>
<tr>
<td>3</td>
<td>Flow Diagram of main subroutine of &quot;SAFSS&quot;</td>
<td>32</td>
</tr>
<tr>
<td>4</td>
<td>Block Diagram of Computer Programs Part 1</td>
<td>33</td>
</tr>
<tr>
<td>5</td>
<td>Block Diagram of Computer Programs Part 2</td>
<td>34</td>
</tr>
<tr>
<td>6</td>
<td>Control Law for DAST ARW-1 Flight Test</td>
<td>35</td>
</tr>
<tr>
<td>7</td>
<td>Root Locus versus Mach Number</td>
<td>36</td>
</tr>
</tbody>
</table>
Nomenclature

\( a \) subscript represents aileron mode

\( an \) accelerometer signal

\( A,G,H \) control law matrices

\( b \) semi-span of the wing

\( b_a \) aileron mode damping (from control law)

\( b_{ii} \) element of generalized damping matrix

\( B \) generalized damping matrix

\( d \) damping factor of \( i \)th mode

\( D \) modal accelerometer deflection matrix

\( g(s) \) portion of control law used for aileron mode

\( G(s) \) entire control law

\( G'(s) \) portion of control law not used for aileron mode

\( I \) identity matrix

\( \lambda \) reduced frequency

\( k_a \) aileron mode stiffness (from control law)

\( k_{ii} \) element of generalized stiffness matrix

\( K \) generalized stiffness matrix

\( L \) generalized force matrix

\( L(s) \) generalized force matrix in Laplace form

\( m \) mass of the aileron

\( m_{ii} \) element of generalized mass matrix
\( m_{ij} \) element of generalized mass matrix

\( M \) generalized mass matrix

\( P_1, P_2, \) matrices representing aerodynamic influence

\( P_3, R_0 \) coefficients

\( P_1', P_2', \) nondimensional matrices representing aerodynamic influence

\( P_3', R_0' \) influence coefficients

\( q \) dynamic pressure

\( Q(s) \) aerodynamic influence coefficient matrix in Laplace form

\( Q(s') \) nondimensional aerodynamic influence coefficient matrices in Laplace form

\( r_i(x,y) \) deflection at point \((x,y)\) for the \( i'\)th mode

\( r_{il} \) leading edge deflection of aileron for \( i'\)th mode

\( r_{it} \) trailing edge deflection of aileron for \( i'\)th mode

\( r_k \) modal deflection at accelerometer location

\( u_i \) \( i'\)th mode displacement vector

\( u_i' \) first derivative of \( i'\)th mode displacement vector

\( u_i'' \) second derivative of \( i'\)th mode displacement vector

\( U \) displacement vector matrix

\( U' \) first derivative of displacement vector matrix

\( U'' \) second derivative of displacement vector matrix

\( v \) free stream velocity

\( w \) frequency of oscillation

\( w_i \) natural frequency of \( i'\)th vibration mode

\( y_i \) \( i'\)th control law state vector

\( Y \) control law state vector matrix

\( X \) generalized force state vector matrix

\( P(x,y) \) density of aircraft at point \((x,y)\)
CHAPTER 1
INTRODUCTION

The ability to predict the aeroelastic response of aircraft wings is of increasing importance as attempts are made to reduce the weight of aircraft wings. One of the methods presently being explored to reduce weight is the use of a flutter suppression system (FSS) to reduce the required structural stiffness of the wing. The wing must be stiff enough to remain vibrationally stable (positive damping) throughout its flight envelope. If the wing is not vibrationally stable, it will flutter and possibly cause the loss of the aircraft.

This paper explains a method for predicting the vibration stability of an aircraft with an analog active aileron FSS. Active aileron refers to the use of an active control system connected to the aileron to damp vibrations. Wing vibrations are sensed by accelerometers and the information is used to deflect the aileron. Aerodynamic forces caused by the aileron deflection oppose wing vibrations and effectively add additional damping to the system.

An assumed mode vibrational analysis approach is used with additional terms added to include the unsteady aerodynamics of a vibrating wing and the control system feedback. The assumed modes used are the actual vibration modes of the aircraft plus an aileron mode. The unsteady aerodynamic affects, modeled by the third order pade approximation method suggested by Edwards in the paper
"Applications of Laplace Transform Methods to Airfoil Motion and Stability Calculations", are used as the forcing functions for the vibration.

A computer program called "SAFSS" was written to determine the vibrational stability of an aircraft wing using the method described above. The input information needed to use 'SAFSS' consists of: the natural frequencies of vibration, the generalized mass for each mode, the generalized force matrices for the mach number of interest, the control law matrices and the aileron deflections for each mode. "SAFSS" produces a root locus plot from a CalComp plotter and a listing of the frequencies and damping.

A comparison between predicted and flight test data for the DAST ARW-1 vehicle is made. DAST stands for drones for aerodynamic and structural testing. ARW-1 stands for aerelastic research wing number one. The DAST ARW-1 is a modified Firebee II target drone fuselage mated to the ARW-1 wing (see Figure 1). ARW-1 is a supercritical, sweptback, transport-type wing with an aspect ratio of 8.8 and a performance design point of mach 0.98 at 45,000 feet. The ARW-1 wing is designed to be susceptible to flutter and is equipped with an active aileron flutter suppression system (FSS).
CHAPTER 2

VIBRATION MODEL OF WING IN STILL AIR

Since flutter is a vibration problem, it would seem to be a reasonable idea to start with a basic vibrational approach and work up to the harder aspects one at a time. This section will review the basic equations of the "assumed modes" method (sometimes called Rayleigh-Ritz method) used to solve vibration problems.

An aircraft wing can be treated as a normal beam if no air is flowing over it. Because the aircraft is symmetrical, two types of vibration are possible: the symmetric case where both wings vibrate 180 degrees out of phase. The symmetric and asymmetric vibrations do not couple with each other so two separate analyses will be run. The equations used to determine the frequencies and damping of the system are identical so no distinction is made between the symmetric and asymmetric cases.

Using the orthogonal normal modes (vibration modes) of the wing and generalized terms, each vibration mode can be written as a function of one variable and its derivatives. The equations of motion are represented by

\[(m_i) \dddot{u}_i + (b_i) \ddot{u}_i + (k_i) u_i = 0\]

where \(m_i\) is the generalized mass, \(b_i\) is the generalized damping, \(k_i\) is the generalized stiffness with \(i\) represent the \(i\)th mode. Using the matrix notation:
the vibration equations can be written as one matrix equation

\[ [M][u'] + [B]u' = [K][u] = 0 \]

Using the fact that \((u'\) is equal to \(u''\), and reordering the above equation, the vibration equations can be written as:

\[
\begin{bmatrix}
I & 0 \\
0 & M
\end{bmatrix}
\begin{bmatrix}
u' \\
u''
\end{bmatrix}
= \begin{bmatrix}
0 & I \\
-K & -B
\end{bmatrix}
\begin{bmatrix}
u' \\
u''
\end{bmatrix}
\]

The left side of the above equations can be simplified by multiplying by the inverse of the left hand square matrix.

The inverse is

\[
\begin{bmatrix}
I & 0 \\
0 & M
\end{bmatrix}^{-1} = \begin{bmatrix}
I & 0 \\
0 & M^{-1}
\end{bmatrix}
\]
Since $[M]$ is a diagonal matrix, $[M]^{-1}$ is a diagonal matrix with terms equal to the inverse of the terms in $[M]$.

$\begin{bmatrix}
m & 0 & \cdots & 0 \\
1 & m & \cdots & 0 \\
0 & 2 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & m \\
\end{bmatrix}^{-1} = 
\begin{bmatrix}
1/m & 0 & \cdots & 0 \\
1 & 0 & \cdots & 0 \\
0 & 1/m & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 1/m \\
\end{bmatrix}$

Multiplying both sides by the inverse solves the equations for the derivatives of "U".

\[
\begin{cases}
U' \\
U''
\end{cases} =
\begin{bmatrix}
0 & -1 & \ldots & 0 \\
-1 & 0 & \ldots & 0 \\
-M & -M & \ldots & 0 \\
\end{bmatrix}
\begin{bmatrix}
U' \\
U''
\end{bmatrix}
\]

Because both $[M]$ and $[K]$ are diagonal matrices, their product will also be a diagonal matrix.

$[M][K]^{-1} = 
\begin{bmatrix}
k/m & 0 & \cdots & 0 \\
1 & k/m & \cdots & 0 \\
0 & 1 & k/m & \cdots \\
\vdots & \vdots & \vdots & \ddots \\
0 & 0 & \cdots & k/m \\
\end{bmatrix}
\begin{bmatrix}
k/m & 0 & \cdots & 0 \\
1 & k/m & \cdots & 0 \\
0 & 1 & k/m & \cdots \\
\vdots & \vdots & \vdots & \ddots \\
0 & 0 & \cdots & k/m \\
\end{bmatrix}^{-1}$

The same statement holds true for the product of $[M]$ and $[B]$. 

5
Two relationships that are helpful in solving vibration problems are

\[ k/m = w \times w \]
\[ b/m = 2d \times w \]

where "w" is the natural frequency and "d" is the damping factor. Substituting these relationships into the matrices

\[
[M]^{-1} [B] = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & b/m & 0 & 0 & 0 \\
0 & 2 & b/m & 0 & 0 \\
0 & 0 & 3 & b/m & 0 \\
0 & 0 & 0 & 0 & b/m
\end{bmatrix}
\]

\[
[M]^{-1} [K] = \begin{bmatrix}
w \times w & 0 & 0 & 0 & 0 \\
0 & w \times w & 0 & 0 & 0 \\
0 & 2 & w \times w & 0 & 0 \\
0 & 0 & 3 & w \times w & 0 \\
0 & 0 & 0 & 0 & w \times w
\end{bmatrix}
\]

\[
[M]^{-1} [B] = \begin{bmatrix}
2d \times w & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 \\
0 & 2d \times w & 0 & 0 & 0 \\
0 & 0 & 2d \times w & 0 & 0 \\
0 & 0 & 0 & 0 & 2d \times w
\end{bmatrix}
\]
The natural frequency for a structure can be obtained from any program that does vibrational analysis such as "NASiRAN" or from a ground vibration test (a test in which the structure is shaken). Damping factors are harder to come by theoretically so a value is usually assumed. The wing provides only a small amount of damping so a small value, such as 0.005, can be assumed.
CHAPTER 3
AERODYNAMIC INFLUENCE COEFFICIENTS

The oscillatory motion of an aircraft wing in flight will produce oscillatory forces on the wing. As the wing oscillates up and down, the deflection and its derivatives cause an effective angle of attack which changes the lifting forces on the wing. Likewise, as the wing oscillates torsionally, the changes in pitch and its derivatives will produce changes in the lift.

The oscillation of the lift on the wing acts as a forcing function to the wing vibration and must be included in the vibration equations. The lift can be included in the following manner

\[
[M] \ddot{\mathbf{U}} + [B] \dot{\mathbf{U}} + [K] \mathbf{U} = \mathbf{L}
\]

where \( \mathbf{L} \) represents the oscillatory aerodynamic loads (generalized forces). The aerodynamic loads can be defined in the Laplace form to be

\[
L(s) = q \cdot [Q(s)] \mathbf{U}
\]

where "q" is the dynamic pressure and \( [Q(s)] \) represents the aerodynamic influence coefficients.

The \( [Q(s)] \) matrix is determined by curve fitting the aerodynamic influence coefficients (AICs) calculated at several vibrational frequencies. AICs for several reduced frequencies can be calculated by using a doublet lattice routine or some other unsteady aerodynamics routine. The
AICs are a function of the reduced frequency and the Mach number. The reduced frequency ($k$) is defined as

$$k = \frac{(b \cdot w)}{v}$$

where "b" is the semi-span, "w" is the frequency of oscillation and "v" is the free stream velocity. A nondimensional form of $[Q(s)]$ can be assumed to be

$$[Q(s')] = (s')^{-1} - ([P1'](s')^2 + [P2']s' + [P3'])$$

where $[P1']$, $[P2']$, $[P3']$ and $[R0']$ are constants determined by performing a least squares curve fit on the AICs calculated at the different reduced frequencies. The "s'" is used to indicate a function of reduced frequency instead of oscillatory frequency.

Rewriting this equation as a function of the oscillation frequency results in

$$[Q(s)] = (s^{\alpha}b/v = [R0']^{-1} - ([P1'](s^{\alpha}b/v) = [P2']s^{\alpha}b/v + [P3'])$$

Multiplying $[Q(s)]$ by $U$ and labeling the product $X_k$

$$X_k = [Q(s)]_iU$$

The above equations can be rewritten by substituting in for $[Q(s)]$ and rearranging

$$(sI + [R0']^{\alpha}v/b)X_k = ([P1'](s^2 \cdot b/v + [P2']s + [P3'] \cdot v/b))U$$

For the ease of writing and to agree with the nomenclature used by other authors the following notation will be used:
Note that \([P1]\), \([P2]\), \([P3]\) and \([R0]\) must be recalculated for each different mach number. Using the above notion and taking the previous equation out of Laplace form,

\[
\{1\}(X') - \{P1\}(U') = -\{R0\}(X) + \{P3\}(U) + \{P2\}(U')
\]

Remembering that

\[
[M] (U'') + [B] (U') + [K] (U) = (L)
\]

and

\[
(L) = q * (X)
\]

the vibration equations can be written as

\[
\begin{bmatrix}
1 & 0 & 0 \\
0 & M & 0 \\
0 & -P1 & I
\end{bmatrix}
\begin{bmatrix}
U'' \\
U' \\
X'
\end{bmatrix}
=
\begin{bmatrix}
0 & 1 & 0 \\
-1: & -B & q1 \\
P3 & P2 & -R0
\end{bmatrix}
\begin{bmatrix}
U' \\
U \\
X
\end{bmatrix}
\]

Taking the inverse of the left hand side

\[
\begin{bmatrix}
1 & 0 & 0 \\
0 & M & 0 \\
0 & -P1 & I
\end{bmatrix}^{-1}
\begin{bmatrix}
1 & 0 & 0 \\
0 & -1 & 0 \\
0 & P1*M & 1
\end{bmatrix}
\]

and multiplying both sides by the inverse
the vibration equations can be written as

\[
\begin{bmatrix}
    U' \\
    U'' \\
    X'
\end{bmatrix} =
\begin{bmatrix}
    I & 0 & 0 \\
    0 & M & -1 \\
    0 & P1*M & -I
\end{bmatrix}^{-1}
\begin{bmatrix}
    0 & I & 0 \\
    -K & -B & qI \\
    P3 & P2 & -R0
\end{bmatrix}
\begin{bmatrix}
    U' \\
    U'' \\
    X'
\end{bmatrix}
\]

The above equation can be used to solve for the frequency and damping of the open-loop aircraft wing vibration.
CHAPTER 4

CONTROL LAW

Although the procedure for putting a control law into matrix form is fairly standard, a few steps can be taken to simplify the solution of the vibration equations. In addition, part of the control law must be used elsewhere in the analysis (see chapter 5).

A control law is defined as the output of a system divided by the input of a system. The input signal for this analysis is an accelerometer signal and the output is the deflection of the aileron. The numerator and denominator of the control law are generally written as the product of several first and second order Laplace polynomials (see figure 6). Part of the control law, a second order polynomial from the denominator and a constant from the numerator

\[ g(s) = \frac{k}{(s + c^2s + k^2)} \]

where "k" and "c" are constants, must be used for the aileron vibration mode (see chapter 5). The entire control law is equal to the product of its parts so

\[ G(s) = G'(s) \cdot g(s) \]

where \( G(s) \) is the entire control law, \( G'(s) \) is the control law without the aileron term and \( g(s) \) is the aileron term.

The portion of the control law that needs to be put into matrix form is \( G'(s) \). Keeping in mind that the input to
G'(s) is the accelerometer signal and the output is a state space vector. The following equation can be written

\[ G'(s) = \frac{V(s)}{an(s)} \]

where "an" is the accelerometer signal and "V" is a state space vector.

Taking \( G'(s) \) out of the Laplace form and putting it into state vector differential equation form results in the following equation

\[
\begin{align*}
\{ \dot{Y} \} & = \{ A \} Y + \{ G \} \{ an \} \\
\{ Y \} & = \{ H \} \{ Y \}
\end{align*}
\]

where the matrices \( \{ A \} \), \( \{ G \} \) and \( \{ H \} \) have no unique solution but depend on the state vectors \( \{ Y \} \).

In order to keep the vibration equations as simple as possible, it is desirable to have \( \{ G \} \) and \( \{ H \} \) contain as many zeros as possible. If the denominator of \( G'(s) \) is at least 3 orders larger than the numerator, \( \{ G \} \) and \( \{ H \} \) can be constructed to each contain only one non-zero term.

\( G'(s) \) can be broken up into the product of its second order (or smaller) polynomials of the form

\[ g_n'(s) = \frac{\left( s + a_n s + b \right)}{\left( s + c_n s + d \right)} \]

where \( g_n'(s) \) is a portion of \( G'(s) \), "a", "b", "c" and "d" are constants.

Assuming that \( g_1'(s) \) is of the form

\[ g_1'(s) = \frac{1}{(s + d)} \]

and \( g_2'(s) \) is of the form
\[ g_2'(s) = \frac{1}{(s^2 + cs + d)} \]

\([G]\) and \([H]\) can be forced to have a minimum of terms. By assigning the state vectors \([y]\) to be the unlabeled inputs and outputs to each portion of the control law

\[
\begin{align*}
  g_1'(s) &= y_1/ a_n \\
  g_2'(s) &= y_1 / y_1
\end{align*}
\]

the following equations can be written

\[
\begin{align*}
  y_1' &= d^2 y_1 + a_n \\
  y_2' &= y_3 \\
  y_3' &= y_1 - d^2 y_2 - c^2 y_3
\end{align*}
\]

Note that the accelerometer signal, "an" is used only once, meaning that \([G]\) contains only one non-zero value. Assigning the state vectors \([y]\) in this manner also causes \([H]\) to contain only one non-zero value.
CHAPTER 5
INCLUDING CONTROL LAW IN VIBRATION EQUATIONS

The movement of the aileron will have two effects on the vibration equations. The first will be the addition of a mode shape (the aileron mode) and the calculation of AICs for the mode. The second will be the introduction of nondiagonal terms to \( [M] \) due to the addition of non-orthogonal aileron mode.

In order to include the aileron as a vibration mode, terms for the generalized mass, generalized stiffness, and damping term must be used. In the previous chapter the control law was divided up into two parts

\[
G(s) = G'(s) + g(s)
\]

where \( G'(s) \) was put into state vector differential form and \( g(s) \) was of the form

\[
G'(s) = \frac{k_a}{s} \left( s + \frac{b_a}{s} \right)
\]

As stated in the last chapter the output of \( G'(s) \) is \( [V] \). Since the output of \( G'(s) \) is the aileron deflection, the following equation must be true

\[
G'(s) = u_a
\]

where \( u \) is the aileron deflection. Substituting for \( g(s) \), to obtain

\[
k_a \left( \frac{b_a}{s} + u_a + u_a + k_a u_a \right)
\]
Solving the above equation for \( u_a''' \)

\[
u_a''' = -k_a u_a - b_a u_a' + k_a V
\]

The vibration mode for the aileron can then be defined as having a
generalized mass of 1.0, a generalized stiffness of \( k_a \), and a
generalized damping term of \( b_a \).

The input to the control law was described in the previous
chapter as the signal from the accelerometer. The accelerometer
signal is the acceleration at the accelerometer location. Since
the vibration model is based on superposition, the acceleration
at any point, due to the \( k \)'th vibration mode, is equal to the
acceleration state vector for the \( k \)'th vibration mode multiplied
by the deflection \( k \)'th mode shape at that point. Therefore,

\[
an = \sum r_k u_k'''
\]

where \( an \) is the accelerometer signal, \( u_k ''' \) is the acceleration
of the \( k \)'th vibration mode and \( r_k \) is the deflection at the
accelerometer location for the \( k \)'th mode shape. Expessing "an"
in matrix notation:

\[
\begin{bmatrix} an \end{bmatrix} = [D] \begin{bmatrix} u_k ''' \end{bmatrix}
\]

where \([D]\) is a row matrix containing the modal vibration deflections
at the accelerometer location.
A conversion factor may be needed between the units of the analysis and the units of the control law. If the control law is designed to convert a signal from g's (acceleration of gravity) to degrees deflection of the aileron, and the units being used in the analysis are inches and seconds, a conversion between the model and the real system must be made. Using the above example, the control law would have the units

\[ G(s) = AD \text{ (degrees)} / \text{an (gravities)}, \]

and the analysis would require the units:

\[ G(s) = uA \text{ (modal deflection)} \circledast c / \text{an (in/sec)}^2 \]

where \( G(s) \) is the control law, "AD" is the aileron deflection, "u" is the state space notation for the aileron mode and "c" is the conversion factor. The conversion from g's to in/sec\(^2\) is simply

\[ 1 \text{ g} = (32.2 \text{ ft/sec}^2) \times (12 \text{ in/ft}) = 368.4 \text{ in/sec}^2 \]

and assuming that a 10 degree deflection of the aileron is equivalent to the aileron mode

\[ 1 \text{ degree} = 1/10 \text{ deflection mode} \]

Therefore, the conversion factor is determined to be

\[ c = \left(\frac{1}{368.4}\right) \times \left(\frac{1}{10}\right) \]
The equations related to the aileron mode and control law are

$$\dot{Y}' = [A]Y' + [G]n$$
$$\dot{V}' = [H]Y'$$
$$n = [D]V'$$

$$u'' = -k* u - b* u' + k* c * [V]$$

Combining the above equations, the following state vector equations can be obtained:

$$\dot{Y}' = [A]Y' + [G][D]V'$$

The aileron mode is not orthogonal to the other vibration modes so non-diagonal terms will be introduced into [M] and/or [K]. In the case being examined here, there is no coupling between modes in [F] but there is coupling in [M]. The terms in [M] are defined as

$$m_{ij} = \iint r_i(x,y) \cdot r_j(x,y) \cdot \rho(x,y) \, dx \, dy$$

where "m_{ij}" is the element ij of the generalized mass matrix "r_i" and "r_j" are deflections at the point (x,y) due to the i'th and j'th mode shapes, and "\rho" is the density of the wing at point (x,y). Because the deflection of the aileron mode is zero everywhere except the aileron, the integral will only be non-zero over the aileron. Assuming the aileron is rectangular and has a constant mass distribution, the generalized mass terms due to the aileron mode are
\[ m_{ia} = -m_r \times \frac{(r_{il}/3 + (r_{it} - r_{il})/2)}{r_{il} \text{ at it}} \]

Where "\( m \)" is the generalized mass of \( i \)'th row and the column representing the aileron mode; "\( m \)" is the mass of the aileron; "\( r \)" is the deflection of the trailing edge of the aileron for the aileron mode shape; "\( r_{il} \)" is the deflection of leading edge of the aileron for the \( i \)'th mode shape; and "\( r_{it} \)" is the deflection of the trailing edge of the aileron for the \( i \)'th mode shape. Assuming that the aileron mode is placed last, \([M]\) would then be written as

\[
[M] = \begin{bmatrix}
m_{1a} & 0 & 0 & \cdots & 0 & m_{la} \\
0 & m_{2a} & 0 & \cdots & 0 & m_{2a} \\
0 & 0 & m_{3a} & \cdots & 0 & m_{3a} \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & m_{na} & m_{na} \\
0 & 0 & 0 & \cdots & 0 & 1.
\end{bmatrix}
\]

Putting the problem into state vector differential notation

\[
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & M & 0 & 0 \\
0 & -P1 & I & 0 \\
0 & -GD & 0 & I
\end{bmatrix} \left\{ \begin{array}{c}
U' \\
U'' \\
X' \\
Y'
\end{array} \right\} = \begin{bmatrix}
0 & 1 & 0 & 0 \\
-K & -B & qI & k_eH_a \\
P3 & P2 & R0 & 0 \\
0 & 0 & 0 & A
\end{bmatrix} \left\{ \begin{array}{c}
U' \\
U'' \\
X \\
Y
\end{array} \right\}
\]

The inverse of the square matrix on the left is
\[
\begin{bmatrix}
I & 0 & 0 & 0 \\
0 & M & 0 & 0 \\
0 & -P_1 & I & 0 \\
0 & -G_D & 0 & I
\end{bmatrix}^{-1} = \begin{bmatrix}
I & 0 & 0 & 0 \\
0 & M & 0 & 0 \\
0 & P_1M & I & 0 \\
0 & -G_D & 0 & I
\end{bmatrix}
\]

Multiplying both sides by the inverse and simplifying,

\[
\begin{align*}
\{U'\} &= \begin{bmatrix}
-1 & 0 & 0 & 0 \\
-M & K & -M & B \\
P_3 & -P_1M & K & P_2 - P_1M & B & -R_0 + P_1M & q & M & k_{acH} \\
G_D & -G_D & k_{acH} & G_D & q & P_1M & A + G_D & k_{acH} & a
\end{bmatrix} \begin{bmatrix}
U' \\
X' \\
Y'
\end{bmatrix} \\
\{U\} &= \begin{bmatrix}
-1 & 0 & 0 & 0 \\
-M & K & -M & B \\
P_3 & -P_1M & K & P_2 - P_1M & B & -R_0 + P_1M & q & M & k_{acH} \\
G_D & -G_D & k_{acH} & G_D & q & P_1M & A + G_D & k_{acH} & a
\end{bmatrix} \begin{bmatrix}
U' \\
X' \\
Y'
\end{bmatrix}
\end{align*}
\]

Because of the aileron mode, the \([M K]\), \([M B]\) and \([M q]\) are no longer diagonal. Assuming that the aileron mode is placed last,

\[
M_k = \begin{bmatrix}
k/m & 0 & 0 & \ldots & k \star m / m \\
\ldots & \ldots & \ldots & \ldots & \ldots \\
& 1 & 1 & \ldots & a \star 1 \star 1 \\
& 0 & k/m & 0 & \ldots & k \star m / m \\
& \ldots & \ldots & \ldots & \ldots & \ldots \\
& 0 & 0 & k/m & k \star m / m \\
& \ldots & \ldots & \ldots & \ldots & \ldots \\
0 & 0 & 0 & \ldots & k & a
\end{bmatrix}
\]

\[
M_B = \begin{bmatrix}
b/m & 0 & 0 & \ldots & b \star m / m \\
\ldots & \ldots & \ldots & \ldots & \ldots \\
1 & 1 & \ldots & a \star 1 \star 1 \\
0 & b/m & 0 & \ldots & b \star m / m \\
\ldots & \ldots & \ldots & \ldots & \ldots \\
& 0 & 0 & b/m & b \star m / m \\
& \ldots & \ldots & \ldots & \ldots & \ldots \\
0 & 0 & 0 & \ldots & b & a
\end{bmatrix}
\]
\[ M^{-1} q = \begin{bmatrix}
q/m & 0 & 0 & \cdots & 0 & 0 \\
0 & q/m & 0 & \cdots & 0 & 0 \\
0 & 0 & q/m & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & q/m & 0 \\
0 & 0 & 0 & \cdots & 0 & 0 \\
\end{bmatrix} \]

This is the form of the vibration equations that will be used to solve for the frequency and damping of an aircraft wing.
CHAPTER 6
ANALYSIS OF DAST ARW-1

The computer program "SAFSS" (Stability Analysis of Flutter Suppression Systems) was written to implement the analysis approach presented in chapters 2-5 (see figures 2 and 3). The input information to "SAFSS" consists of the generalized force matrices, the control law matrices, the generalized mass, natural frequency, and the aileron deflections for each mode. In order to obtain the input information, other computer programs were used.

The finite element analysis portion of the computer program "NASTRAN" and an input file provided by NASA Langley were used to calculate the first ten natural frequencies and mode shapes of the DAST ARW-1. The aileron mode shape was then substituted for one of the rigid body modes calculated by "NASTRAN". The mode shapes were then input to the double lattice portion of "NASTRAN" and used to calculate the aerodynamic influence coefficients (AMIL) at fifteen reduced frequencies (see figure 4). The pade approximate curve fitting routine "QUEFIT" used the AMIL at the reduced frequencies to calculate the generalized force matrices (see figure 5).

The control law for the third test flight of DAST ARW-1 was obtained from NASA Langley and is shown in figure 6. The portion of the control law used for the aileron mode was

\[ g(s) = \frac{2^2}{(1256.6) / (s = 502.7s + (1256.6))} \]
Therefore, the aileron mode had a natural frequency of 1256.6 hertz and a damping factor of 0.2. The remaining terms of the control law were put into matrices by the program "CONTRL".

The input file documentation needed to use the computer program "SAFSS" is presented in appendix B and a listing of the program is in appendix C. A sample input file for the DAST ARW-1 symmetric, .825 mach number, 15,000 feet altitude case is presented in appendix D. The output for the sample case is contained in appendix E.
CHAPTER 7

FLIGHT TEST DATA

Data from the third test flight of the DAST ARW-1 test vehicle was collected by the NASA Dryden Flight Test Research Center. The DAST was launched from a B-52 aircraft and remotely piloted to an altitude of 15,000 ft. Testing was performed at the following mach numbers: 0.70, 0.75, 0.775, 0.80 and 0.825.

The wing was vibrationally excited at each mach number to determine the vibrational stability. Excitation of the wing was produced by oscillating the aileron in a continuous frequency sweep from 10 to 40 hertz. Excitation sweeps were performed with the flutter suppression system (FSS) on and/or off (depending on the predicted stability at the test point) for both symmetric and asymmetric cases.

The FSS used for the DAST ARW-1 is active aileron control which operates in the following manner. Electrical signals from four accelerometers (two located in the fuselage and one in each wing) are sent to a compensator. The compensator separates the symmetric, asymmetric, and rigid body motions then signals the actuators to hydraulically move the ailerons. A time history of the accelerometer signals is used to calculate the frequencies of vibration and damping factors.

Because of an error in the implementation of the control law, the FSS was operated at one-half of the designed gain. This error caused test data for a gain factor
of one-half to be compared with predictions for a gain factor of one. As a result, the DAST ARW-1 with the FSS on entered a flutter region near mach 0.825 at 15,000 ft. and was lost.
CHAPTER 8

RESULTS

The natural frequencies for the symmetric case of DAST ARW-1 obtained from the program "NASTRAN" and from a ground vibrations test (GVT), are presented in Table 1. For the frequency range of interest (10-40 hertz), the predicted and actual frequencies agree to within 0.5 hertz. The first wing bending mode is of special importance because the aircraft fluttered in that mode. "NASTRAN" predicted the first bending mode to have a natural frequency of 9.1 hertz. The actual value was 9.6 hertz. The mode shapes and natural frequency were used to calculate the generalized force matrices therefore a 5% error could be considered large.

The data taken at mach 0.755 is believed to be unreliable because the wing showed uninitiated oscillations, possible due to atmospheric turbulence. The FSS (flutter suppression system) was left on for velocities of mach 0.755 and above. The FSS off data for mach 0.755 and above was calculated from FSS on information. Due to an error in the implementation of the control law, the FSS on condition was operating at one-half the desired gain (K=.5) for the test flight.

Table 2 shows the relationship between the predicted and experimental vibrational frequencies with the FSS off and Table 3 shows the relationship with the FSS on. The average error in frequency for the FSS off condition is 5% while the error for the FSS on condition is 8%.
The FSS increased the vibrational frequency from 13.91 hertz to 19.93 hertz. The analysis predicted the effect of the FSS quite well as an increase from 14.54 to 21.58 hertz.

A graph of root locus versus mach number for the 15,000 ft. altitude case for the first wing bending mode is shown in Figure 7. Three gain factors are plotted: full gain (K=1, the designed gain factor), half gain (K=.5, the actual FSS on gain) and no gain (K=0, the FSS off gain). The root locus plot is interpreted by noting that as the real term approaches the imaginary axis from the negative real side, the system becomes less stable. As the imaginary axis is crossed from the negative real to the positive real, the system goes from stable (no flutter) to unstable (flutter). The experimental results are shown to be consistently less damped and of a lower frequency than the predicted values. Trends seem to be predicted well, however more data is needed to draw any conclusions.

The predictions presented here are not the outcome of a single analysis, but rely on finite element, unsteady aerodynamics and vibrational analysis which tend to reduce the accuracy. The small difference which resulted can be caused by the inaccuracy (5%) of the finite element analysis alone.
CHAPTER 8
CONCLUSIONS

A method for analysis of an active aileron control flutter suppression system has been explained and compared with flight data. The method was shown to produce reasonable results but relies heavily on finite element and unsteady aerodynamic analysis. The ability of the finite element routine to match the mode shapes and natural frequencies of the ground vibration test will have a large affect on the accuracy of the flutter analysis.

Future work in this area will move toward digital instead of analog control systems. Digital systems will reduce space and weight requirements as well as make possible the use of dynamic control laws.
TABLE 1
NATURAL FREQUENCIES OF VIBRATION MODES PREDICTED BY NASTRAN
AND MEASURED DURING GROUND VIBRATION TEST

<table>
<thead>
<tr>
<th>Mode</th>
<th>frequency (hertz)</th>
<th>NASTRAN</th>
<th>GVT</th>
</tr>
</thead>
<tbody>
<tr>
<td>First wing bending</td>
<td>9.1</td>
<td>9.6</td>
<td></td>
</tr>
<tr>
<td>First fuselage bending</td>
<td>16.5</td>
<td>16.2</td>
<td></td>
</tr>
<tr>
<td>Wing bending-torsion</td>
<td>29.6</td>
<td>29.1</td>
<td></td>
</tr>
</tbody>
</table>

TABLE 2
VIBRATIONAL FREQUENCIES OF FIRST WING BENDING MODE
PREDICTED BY SAFSS AND OBTAINED FROM FLIGHT TESTS
FOR FSS OFF AT 15000 FT.

<table>
<thead>
<tr>
<th>Mach Number</th>
<th>Frequency (hertz)</th>
<th>SAFSS</th>
<th>Flight Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>.75</td>
<td>13.24</td>
<td>12.25</td>
<td></td>
</tr>
<tr>
<td>.80</td>
<td>15.54</td>
<td>13.91</td>
<td></td>
</tr>
</tbody>
</table>
TABLE 3
VIBRATIONAL FREQUENCY OF FIRST WING BENDING MODE
PREDICTED BY SAFSS AND OBTAINED FROM FLIGHT TEST
FOR FSS ON AT 15000 FT.

<table>
<thead>
<tr>
<th>Mach Number</th>
<th>Frequency (hertz)</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SAFSS</td>
<td>Flight Test</td>
<td></td>
<td></td>
</tr>
<tr>
<td>.80</td>
<td>21.58</td>
<td>19.93</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Wing Span: 14.5 ft.
Airfoil: Supercritical
Aspect Ratio: 6.8

Figure 1. DAST ARW-1 planform
Figure 2. Flow Diagram of Program "SAFSS"
Figure 3. Flow Diagram of Main Subroutine of "SAFSS"
Figure 4. Block Diagram of Computer Programs Part 1
"Curve Fit" Aerodynamic Influence Coefficients

 QUEFIT

 Processing Instruction

 Control Law

 CONTROL

 Creates Control Law matrices

 Card File

 Card File

 Mode and wing data

 SAFSS

 Determines Frequencies and Damping

 Output Listing

 Root Locus Plots

 End

 Figure 5. Block Diagram of Computer Programs Part 2
\[ C(s) = \frac{(738)(2250) s (s^2 + 76.78s + (295.3)^2) (s^2 + 120s + (306)^2) (s^2 + 100s + (71)^2)}{(s + 2) (s + 295.3)^2 (s + 1500)^2 (s^2 + 240s + (342)^2) (s^2 + 100s + (58)^2) (s^2 + 100s + (112)^2)} \]

\[ (s^2 + 100s + (168)^2) (295.3)^2 (1256.6)^2 \]

\[ (s^2 + 76.78s + (295.3)^2) (s^2 + 589.4s + (439.8)^2) (s^2 + 502.7s + (1256.6)^2) \]

Figure 6. Control Law for DAST AJ-1 Flight Test 3.
Figure 7. Root locus versus Mach number for first wing bending mode at 15,000 ft. altitude.
APPENDIX B

This appendix contains a description of the input file for the computer program "SAFSS" (Stability Analysis of Flutter Suppression System). This program is designed to use the output of an aerodynamic influence coefficient curve fitting routine has outlined by Edwards in "Applications of Laplace Transform Methods to Airfoil Motion and Stability Calculations".

The input information includes the aerodynamics, the control law, and information about each mode, such generalized mass, frequency of oscillation in still air and physical information about the wing. The input is grouped into four blocks. The first block contains information about the analysis options and the size of the problem. The second has all the information about the control law. The third block contains all the aerodynamic matrices and mach number. The last block contains the wing vibration and mode shape information.

The term "card" means that all the information is contained on one card. The term "card set" will refer to a group of cards that contain the stated information. Card sets usually contain matrices. Matrices will be read in by rows (first row, second row etc.). The last value in a row of a matrix will be the last value read from that card. As an example, a ten by ten matrix to be input with only eight values read from each card, will need two cards for each row. The first card for each row will contain eight values with the second card containing the
last two.

Variable Name               Column #   Format   Description

CARD BLOCK 1

CARD 1:--------------------------

TITLE(16)  1-46       6A8
   The title and/or description desired on the data output
   and the root locus plot. The time and date will be supplied
   by the computer.

CARD 2:--------------------------

NGASE  1-5       I5
   The number of different altitudes that calculations are
   to be performed at.

MODE  1-15       I5
   The number of vibration modes being used for the
   calculations. Must include the aileron mode.

MALEMN  11-15       I5
   The number of the vibration mode that contains the
   aileron mode. Usually the first or the last.

NGAINT  16-20       I5
   The type of arithmetic progression desired for the gain
   factor, (1=linear, 2=geometric).
   Example: linear (0.,.25,.5,.75,1.0,1.25,1.5)
geometric (0., .25, .5, 1.0, 2.0, 4.0, 8.0)

**NGAINS 21-25 15**

The number of gain factors to be analyzed including zero. In the example above, both linear and geometric have 7 gain factors.

**GAINUP 26-35 F10.6**

The desired value of the first gain factor after the pole (0.0). Both the linear and geometric series in the above example have a value of 0.25.

**IEGVEC 36-40 15**

If the eigenvectors are to be printed, use the number "1". If any other number is found the eigenvectors will not be printed.

**CLEN 41-50 F10.6**

The span of the wing. It is used in the doublet lattice routine as the reference length to calculate the reduced frequency.

**IPLT 51-55 15**

If a root locus plot is desired, set equal to "1". If any other number is used, no plot will be made.

**YMAX 56-65 F10.6**

The maximum frequency (rad/sec) that is of interest. This value is used in plotting only, and does not effect the calculations.
CARD SET 3:-------------------------------------------------------------

ALT(I)          1-80     8F10.6

This card(s) contains the altitudes to be used in the calculations. The maximum number of altitudes that can fit on one card is 8. If more than 8 altitudes are desired additional cards must be used.

CARD BLOCK 2 **************************************************************

This set of cards is obtained from the "Ctrl" program as output. The entire block of cards is produced and order ready to be inserted into this data card deck.

CARD 4:-------------------------------------------------------------

NCONTR           1-5       15

The size of the control matrix.

CARD SET 5:-------------------------------------------------------------

AC(I,J)          1-80     4E20.13

This set of cards contains the "A" matrix of the control law. It is a square matrix with dimensions NxN, wherein the order of the control matrix (does not include the terms used as the aileron mode).

CARD SET 6:-------------------------------------------------------------

GC(I,1)          1-80     4E20.13

This set of cards contains the "G" matrix of the control law. It is a row matrix with 1xN dimensions, where N
is the order of the control matrix.

CARD SET 7:---------------------------------------------------------------

HC(1,J) 1-80 4E20.13

This set of cards contains the "H" matrix of the control law. It is a column matrix with Nx1 dimensions, where N is the order of the control matrix.

CARD BLOCK 3:***************************************************************************

This block of cards contains the P1', P2', P3', and RO' matrices as described by Edwards (ref. 1, 2, 3 and 4). This block of cards is designed to be used as a unit. There is no need to separate the information in this block at any time.

CARD 8:---------------------------------------------------------------

MACH 1-10 F10.6

The mach number at which the calculations of the unsteady aerodynamics were performed are included here. The unsteady aerodynamic data is only valid for one Mach number.

CARD 9:---------------------------------------------------------------

ISYM 1-5 L5

This states whether the analysis is for a symmetric or asymmetric case, where "T" is for the symmetric and "F" is for the asymmetric case. These parameters have no effect on the problem, but does label the type of problem being solved.

CARD SET:---------------------------------------------------------------
This set of cards contain the Pl' matrix. This matrix is square of dimension \( M \times M \), where \( M \) is the number of mode shapes. One card is needed for every 8 elements in a row and there are \( M \) rows.

**CARD SET 11:**

Pl(K,L) 1-80 4E20.13

Same as card set 10 except that it is the P2' matrix.

**CARD SET 12:**

P2(K,L) 1-80 4E20.13

Same as card set 10 except that it is the P3' matrix.

**CARD SET 13:**

P3(K,L) 1-80 4E20.13

Same as card set 10 except that it is the R0' matrix.

**CARD BLOCK 4**

This card block contains the physical characteristics of the wing and should remain the same throughout the analysis.

**CARD SET 14:**

R0(K,L) 1-80 4E20.13

Same as card set 10 except that it is the R0' matrix.

**OMEGA(I)**

1-80 8F10.6

This card contains the vibrational frequency of each mode shape. It is important that the order of the mode shapes is the same as the order for the doublet lattice routine. The frequency
for the aileron mode is obtained from the 2’nd order term of the control law and was not input in the "Control" program.

CARD SET 15:-----------------------------------------------

ZETA(I)  
1-80  8F10.6

This card contains the damping factor for each vibration mode. The modal damping term for the aileron mode is obtained from the 2’nd order term of the control law that was not input into the "Control" program. The modal damping term for the other modes were hard to obtain therefore, a small value of approximately .005, was assumed.

CARD SET 16:-----------------------------------------------

GMMASS(I)  
1-80  8F10.6

This card contains the generalized mass terms for each mode. The value for the generalized mass term of the aileron has a value of 1.0000. If the user does not set the aileron term to 1.0000, the program will do so.

CARD SET 17:-----------------------------------------------

DEFLECT(I)  
1-80  8F10.6

The deflection of the wing at the accelerometer location for each mode shape. The deflection for the aileron mode should be zero.

CARD SET 18:-----------------------------------------------
XMD 1-10 F10.6

The mass of the aileron in consistent units. If the problem has been done in units of inches, seconds, and pounds (as is usually the case) the mass should be of such units as to be consistent with the rest of the problem. The consistent units for the above example are the mass of the aileron in slugs divided by 12 in/ft. The program does not do this type of conversion so that other system of units can be used.

CONVFT 31-40 F10.6

This card contains the conversion factor for the units of the input and output of the control law used in this analysis. If the control law input is in g's (gravities) and the analysis is in inches and seconds a conversion must be made. Likewise, the output of the control law output may be in degrees and the mode shape deflection may be one inch. For this example, the input portion of the conversion is \( \frac{1g}{(32.2 \text{ ft/sec}^2 \cdot 12 \text{ in/ft})} \). The output portion would be \( \frac{1 \text{ degree}}{(\text{the angle due to the one inch deflection})} \). The proper conversion is obtained by multiplying the two terms.

CARD SET 19:---------------------------------------------------------------

The information below is required for each mode shape. The aileron mode will have a hinge line deflection of zero. Each card contains the information for one mode.
PHIII   1-10   F10.6

The average deflection at leading edge of the aileron.

SIGII   21-22   F10.6

The average deflection at the trailing edge of the aileron.
#DECK SAFSS

PROGRAM MAIN (INPUT, OUTPUT, TAPE1=INPUT, TAPE6=OUTPUT)

C******************************************************************
C PROGRAM SAFSS (STABILITY ANALYSIS OF FLUTTER SUPPRESSION SYSTEM)
C******************************************************************
C THIS PROGRAM PUTS TOGETHER AND SOLVES THE MATRIX REPRESENTING THE
C WING MOTION AND THE CONTROL LAW.
C THE PROGRAM IS SETUP TO TAKE THE CONTROL LAW MATRICES, THE
C GENERALIZED FORCE MATRICES FOR A GIVEN MACH #,
C THE MODAL FREQUENCIES, THE GENERALIZED MASS AT EACH MODE, THE
C DEFLECTION AT THE ACCELEROMETER OF EACH MODE, THE AVERAGE DEFLECT-
C ION OF THE AILERON AT THE LEADING AND TRAILING EDGE FOR EACH MODE,
C AND THE MASS OF THE AILERON AND OUTPUT A ROOT LOCUS
C PLOT OF THE FREQUENCIES OF VIBRATION FOR THE ALTITUDES OF INTEREST.

C DESCRIPTION OF INPUT DATA

C TITLE = TITLE OF THE JOB
C NCASE = NUMBER OF ALTITUDES TO BE EVALUATED
C MODE = NUMBER OF VIBRATION MODE BEING USED
C NALERN = WHICH VIBRATION MODE IS THE AILERON DEFLECTION MODE
C IEGVEC = DO YOU WANT THE IEGEN VECTORS PRINTED (O=NO)
C NGAINF = NUMBER OF GAIN FACTORS TO BE SOLVED INCLUDING ZERO
C NGAINT = TYPE OF GAIN FACTOR PROGRESSION USED
C GAINUP = FIRST GAIN FACTOR TO BE USED AFTER ZERO
C ALT(I) = ALTITUDE OF THE I'TH ROOT LOCUS PLOT
C NCONTR = SIZE OF THE CONTROL MATRIX
AC(I,J) = THE "A" MATRIX OF THE CONTROL LAW
GC(I,J) = THE "G" MATRIX OF THE CONTROL LAW
HC(I,J) = THE "H" MATRIX OF THE CONTROL LAW
P1(K,L) = THE S**2 TERM OF THE PADE APPROXIMANT OF THE LOAD
P2(K,L) = THE S TERM OF THE PADE APPROXIMANT OF THE LOAD
P3(K,L) = THE CONSTANT TERM OF THE PADE APPROXIMANT OF THE LOAD
RQ(K,L) = THE E. GENVALUE TERM OF THE PADE APPROXIMANT TO THE LOAD
OMEGA(I) = FREQUENCIES OF THE CORRESPONDING VIBRATION MODE
ZETA(I) = DAMPING OF THE CORRESPONDING VIBRATION MODE
SMASS = GENERALIZED MASS OF THE CORRESPONDING VIBRATION MODE
DEFLECT = DEFLECTION AT THE ACCELEROMETER FOR EACH MODE
XMD = MASS OF THEAILERON
CONVFT = CONVERSION FACTOR BETWEEN CONTROL LAW UNITS AND THE
UNITS USED WITH THIS PROGRAM
PHI(I) = AVERAGE DEFLECTION OF THEAILERON LEADING EDGE FOR
THE I'TH MODE
SIG(I) = AVERAGE DEFLECTION OF THEAILERON TRAILING EDGE FOR
THE I'TH MODE

******************************************************************************
DIMENSION Wl(11,11),W2(11,11),W3(11,11)
DIMENSION P1(11,11),P2(11,11),P3(11,11),RQ(11,11),VP3(11,11),
1VRO(11,11),OMEGA2(11,11),O8M(11,11),B(11,11),VP1H2(11,11),
2VP1(11,11),TITLE(8),OMEGA(11),ZETA(11),GMASS(11),A(48,48),WR(48),
3W(48),2W(11,11),23(48,48),INT(48),AC(15,15),AMTRX(33,33),
4AMTRX(33,1),FMTRX(1,1),GMRX(1,1),HMTRX(1,1),W4(33,33),
5PHI(11),SIG(11),REACT(11),DEFLECT(11),ALT(20),
6GC(15,1),HC(15,1),HS(11,11),WB(15,11),W7(11,15),WB(15,11),
7WB(11,1),W10(1,15),W1(11,15),W12(15,15)
8READ 801,(TITLE(1),I=1,6)
9CALL DATE(TITLE(7))
10CALL TIME(TITLE(8))
11FINIT 801,TITLE
12READ 802,NCASE,MODE,MALERN,NGAINT,NGAINS,GAINUP,IEGVEC,CLEN,
13* IPLT,YMAX
14READ 804,(ALT(I),I=1,NCASE)
READ 802, NCONTR
MSIZE*MODE*3+NCONTR
MODE3*MODE*3
CALL PADJR (NCASe,MODE,MODE3,MSIZE,MALERN,CLEN,W1,W2,W3,
P1,P2,P3,RO,VP3,VRD,OMEGA2,OMG.B,VP1H2,VP1,OMEGA,ZETA,GMMASS,
,.A,W1,2,3,IEQVEC,INT.AC,NCONTR,AMTRX,BMTRX,FMTRX,GMTRX,AMTRX,
,.H4,PHI,SIG,REACT,DEFLECT,ALT,GC,HC,W5,W6,W7,W8,W9,W10,W11,W12,
,.NGAIN,NGAINS,GAINUP,IPLT,YMAX)
801 FORMAT(BAB)
804 FORMAT(BF10.6)
802 FORMAT(S15,F10.6,15,F10.6,15,F10.6)
801 FORMAT(1H1,10X,BAB)
END
SUBROUTINE PADJR(NCASE,MODE,MODE3,MSIZE,MALERN,CLEN,W1,W2,
,.W3,P1,P2,P3,RO,VP3,VRD,OMEGA2,OMG.B,VP1H2,VP1,OMEGA,ZETA,
,.GMMASS,A,W1,2,3,IEQVEC,INT.AC,NCONTR,AMTRX,BMTRX,FMTRX,GMTRX,AMTRX,
,.H4,PHI,SIG,REACT,DEFLECT,ALT,GC,HC,W5,W6,W7,W8,W9,W10,
,.W11,W12,NGAIN,NGAINS,GAINUP,IPLT,YMAX)
C******************************************************************************
C
C SUBROUTINE PADJR
C******************************************************************************
C
C THIS PROGRAM RESIZES THE ARRAYS TO FIT THE PROBLEM; THEN READS
C THE INPUT INFORMATION AND THEN CREATES AND SOLVES THE STATE SPACE
C MATRIX.
C******************************************************************************
C
LOGICAL I5YM
REAL MACH
DIMENSION PHI(MODE),SIG(MODE),REACT(MODE),DEFLECT(MODE),
,.W1(MODE,MODE),W2(MODE,MODE),W3(MODE,MODE),
,.P1(MODE,MODE),P2(MODE,MODE),P3(MODE,MODE),RO(MODE,MODE),
,.VP3(MODE,MODE),VRD(MODE,MODE),OMEGA2(MODE,MODE),OMG.B(MODE,MODE),
,.B(MODE,MODE),VP1H2(MODE,MODE),VP1(MODE,MODE),
OMEGA(MODE), ZETA(MODE),
GMSS(MODE), A(MSIZE,MSIZE), W0(MSIZE), W1(MSIZE), Z(MODE,MODE),
INT(MSIZE), 23(MSIZE,MSIZE), ALT(NCASE), AC(NCONTR,NCONTR),
AHTRX(MODE3,MODE3), BMTRX(MODE3), FMTRX(1,1), GMTRX(1,MODE3),
HMTRX(1,MODE3), W4(MODE3,MODE3), GC(NCONTR), HC(1,NCONTR),
HS1(MODE), HG(NCONTR,MODE), H7(MODE,NCONTR), H9(NCONTR,MODE),
H0(MODE), 101, W11(MODE,NCONTR), W12(MODE,NCONTR)

C READ MATRICES REPRESENTING CONTROL LAW (FROM PROGRAM "CONTROL")
C***********************************************************************
DO 11 I=1,NCONTR
  11 READ B01,(AC(I,J),J=1,NCONTR)
     READ B01,(GC(IR,1),IR=1,NCONTR)
     READ B01,(HC(1,IC),IC=1,NCONTR)

C READ MACH NUMBER. READ (T/F) WHETHER SYMMETRIC OR ASYMMETRIC
C READ PADE APPROXIMATE MATRICES FOR UNSTEADY AERODYNAMICS. MATRICES
C ARE P1',P2',P3',K0'.
C***********************************************************************
READ B02,MACH
READ B03,ISYM
DO 12 IR=1,MODE
  12 CONTINUE
     READ B01,(P1(IR,IC),IC=1,MODE)
DO 13 IR=1,MODE
  13 CONTINUE
     READ B01,(P2(IR,IC),IC=1,MODE)
DO 14 IR=1,MODE
  14 CONTINUE
     READ B01,(P3(IR,IC),IC=1,MODE)
DO 15 IR=1,MODE
  15 CONTINUE
     READ B01,(R0(IR,IC),IC=1,MODE)

C READ NATURAL FREQUENCIES OF WING VIBRATION, MODAL DAMPING TERMS,
C GENERALIZED MASS TERMS, AND THE DEFLECTION OF WING AT THE
ACCELEROMETER LOCATION.

READ 802, (OMEGA(I), I=1, MODE)
READ 802, (ZETA(I), I=1, MODE)
READ 802, (GMASCl, I=1, MODE)
READ 802, (DEFLECT(I), I=1, MODE)

READ THE MASS OF THE AILERON, THE CONVERSION FACTOR BETWEEN
THE UNITS OF THE CONTROL LAW AND THE UNITS OF THIS ANALYSIS.

READ 802, XMD, CONVFT

READ THE MODAL AVERAGE DEFLECTION AT THE LEADING EDGE OF
THE AILERON AND THE AVERAGE MODAL DEFLECTION AT THE TRAILING EDGE
OF THE AILERON FOR EACH MODE.

DO 16 I=1, MODE
16 READ 802, PHI(I), SIG(I)

END OF INPUT REGION

PRINT INFORMATION FOR LISTING

PRINT 904, MACH, MODE, NCONTR, CLN
PRINT 906, (ALI(I), I=1, NCASE)
PRINT 901, (OMEGA(I), I=1, MODE)
PRINT 902, (ZETA(I), I=1, MODE)
PRINT 903, (GMAScl, I=1, MODE)
PRINT 908
DO 17 IR=1, MODE
PRINT 907, (PI(IR, IC), IC=1, MODE)
17 CONTINUE
PRINT 909
DO 18 IR=1, MODE
PRINT 907, (P2(IR, IC), IC=1, MODE)
18 CONTINUE
PRINT 910
DO 19 IR=1,MODE
PRINT 907,(P3(IR,IC),IC=1,MODE)
19 CONTINUE
PRINT 911
DO 20 IR=1,MODE
PRINT 907,(RO(IR,IC),IC=1,MODE)
20 CONTINUE
PRINT 912
C*****************************************************************************
C START PUTTING TOGETHER THE MATRIX FOR SOLUTION
C*****************************************************************************
C CALCULATE EFFECT OF THE AILERON MODE ON OTHER MODES
C (NONDIAGONAL TERMS OF THE MASS MATRIX)
C*****************************************************************************
SI=SIGN(MALERN)-PHI(MALERN)
DO 21 I=1,MODE
21 REACT(I)=-XMD*SI*(PHI(I)/2+(SIG(I)-PHI(I))/3)
REACT(MALERN)=1.
GMASS(MALERN)=1.
*****************************************************************************
IF (IPLT.EQ.1) CALL INITPLT
*****************************************************************************
C START OF LOOP FOR EACH ALTITUDE CALCULATIONS
*****************************************************************************
DO 500 ICASE=1,NCASE
*****************************************************************************
C*****************************************************************************
IF (IPLT.EQ.1) CALL AXISPLT(ALT(NCASE),MACH,YMAX,YMIN,
.XMAX,XMIN,SCALE,ISYM,TITLE)
*****************************************************************************
C*****************************************************************************
CALL GBARC (MACH,ALT(ICASE),U,OBAR,INFLG)
PRINT 921
PRINT 922, MACH,ALT(ICASE),OBAR,U
*****************************************************************************
C CALCULATE THE P3 AND RO MATRICES FROM THE P3' AND RO' MATRICES BY
C MULTIPLYING BY THE VELOCITY OVER THE SEMI-SPAN. CALCULATE THE PI
C MATRIX BY DIVIDING THE PI' MATRIX BY THE VELOCITY OVER THE
C SEMI-SPAN.
C*************************************************************************************************************************
UB=U/(CLLEN/2.)
CALL ARITH (UB,P3,0.,P3,VP3,MODE,MODE)
CALL ARITH (UB,RO,0.,RO,VP0,MODE,MODE)
UB=1./UB
CALL ARITH (UB,PI,0.,PI,VP1,MODE,MODE)
C*************************************************************************************************************************
C CONSTRUCT THE K/M, B/M AND G/M MATRICES.
C*************************************************************************************************************************
DUM1=OMEGA(MALERN)**2
DUM2=2.*OMEGA(MALERN)*ZETA(MALERN)
DO 31 IR=1.,MODE
  DO 30 IC=1.,MODE
  OMEGA2(IR,IC)=0.
  B(IR,IC)=0.
  GBM(IR,IC)=0.
30 CONTINUE
OMEGA2(IR,IR)=OMEGA(IR)**2
B(IR,IR)=2.*OMEGA(IR)*ZETA(IR)
IF (IR.NE.MALERN) GBM(IR,IR)=DUM1*REACT(IR)/GMASS(IR)
IF (IR.NE.MALERN) OMEGA2(IR,MALERN)=DUM1*REACT(IR)/GMASS(IR)
IF (IR.NE.MALERN) B(IR,MALERN)=DUM2*REACT(IR)/GMASS(IR)
31 CONTINUE
C*************************************************************************************************************************
C CONSTRUCT: -P3 + P1*K/M
C -P2 + P1*B/M
C RO - P1*G/M
C*************************************************************************************************************************
CALL MUL(VP1,OMEGA2,WH,MODE)
CALL ARITH(1.,VP3,-1.,WH,WH,MODE,MODE)
CALL MUL(1.,VP1,B,WH,MODE)
CALL ARITH(1.,P2,-1.,WH,WH,MODE,MODE)
CALL MULT(VPL,GBM,W3,MODE)
CALL ARITH(-1.,VPL,1.,W3,W3,MODE,MODE)
XK=0.
IDUM1=NGAINS+1
DO 400 KK=1,IDUM1

C***********************************************************************
C PLACE SMALL MATRICES INTO ONE LARGE MATRIX
C***********************************************************************
DO 32 IR=1.,MSIZE
   DO 32 IC=1.,MSIZE
      A(IR,IC)=0.
      DO 33 IR=1.,MODE
         A(2*IR-1+MODE,2*IR+MODE)=1.
      DO 33 IC=1.,MODE
         A(2*IR+MODE,2*IC-1+MODE)=-OMEGA2(IR,IC)
         A(2*IR+MODE,2*IC+MODE)=B(IR,IC)
         A(2*IR+MODE,IC)=GBM(IR,IC)
         A(IR,IC)=W3(IR,IC)
         A(IR,2*IC-1+MODE)=H1(IR,IC)
         A(IR,2*IC+MODE)=W2(IR,IC)
      CONTINUE

C*****************************************************************************
C ADD ADDITIONAL TERMS TO THE LARGE MATRIX THAT ARE CAUSED BY THE
C ADDITION OF THE LUTTER SUPPRESSION SYSTEM.
C*****************************************************************************
CALL FSS(A,OMEGA2,B,GBM,MODE,MSIZE,NCONTR,MALERN,
       DEFLECT,REACT,GMAGS,VPL,AC,GC,HC,W5,W6,W7,W8,W9,
       WH,W11,W12,X,K,CONVF1)

C*****************************************************************************
C END OF MATRIX CONSTRUCTION, NOW GET THE EIGENVALUES
C*****************************************************************************
CALL ELMHS(S,MSIZE,MSIZE,1,MSIZE,A,INT)
CALL ELTRAN(MSIZE,MSIZE,1,MSIZE,A,INT,2)
CALL HPR2(MSIZE,MSIZE,1,MSIZE,A,IR,W1,23,IERR,0)
C******* IF ZEROS WERE CALCULATED GO TO ZERO PLOTTING SECTION *******
IF (XK.EQ.1000000.) GO TO 42
PRINT 931,XK
PRINT 932,IERR
C******** CALCULATE FREQUENCIES AND DAMPING ***********************
DO 41 IE=1,MSIZE
WN2=WR(IE)*WR(IE)+WI(IE)*WI(IE)
WN=SQRT(WN2)
ZTA=-WR(IE)/WN
CYCLES=WN*.16034
PRINT 933, WR(IE), WI(IE), WN, CYCLES, ZTA
41 CONTINUE
C************************************************************
C PLOT RESULTS ON ROOT LOCUS PLOT.
C*************************************************************
IF (IPLT.EQ.1) CALL POINT(MSIZE,WR,WI,SCALE,KK,XK,YMAX,YMIN,
.XMAX,XMIN)
C******** CHANGE THE GAIN FACTOR *****************************
IF (NGAINT.EQ.1) XK=XK*GAINUP
IF (NGAINT.EQ.2) XK=GAINUP*2**(KK-1)
IF (NGAINS.EQ.KK) XK=1000000.
GO TO 43
C************************************************************
C PLOT AND LIST THE ZEROS OF THE MATRIX
C*************************************************************
42 PRINT 951
PRINT 952,(WR(I),WI(I),I=1,MSIZE)
IF (IPLT.EQ.1) CALL ZEROPLT(MSIZE,WR,WI,SCALE,YMAX,YMIN,
.XMAX,XMIN,NGAINS)
C*************************************************************
C UNPACK THE EIGENVECTOR MATRIX (Z) PER EISPACK GUIDE
C AND LIST.
C*************************************************************
43 IF (IEVEC.EQ.0) GO TO 400
PRINT 941
L=L+1
111 CONTINUE
IF (MSIZE.LE.16*L) GO TO 110
L=L+1
GO TO 110
110 CONTINUE
LL=MSIZE-10*(L-1)
L,L=1
113 CONTINUE '1=*9
IF (LLL.EQ.L) I=LL
PRINT 966
II=(LLL-1)*10+I
II[!!]+1
PRINT 942,((WR(J),WI(J)),J-I;111)
PRINT 966
DO 51 J=1,MSIZE
PRINT 942,((ZJ(J,K),K-II,111)
51 CONTINUE
LL-LML+1
IF (LLI,GT,L) GO TO 115
GO TO 113
115 CONTINUE
C*******************************************************************************
C   ENDS LOOP, GO BACK AND CALCULATE NEXT GAIN FACTOR.
C*******************************************************************************
400 CONTINUE
C*******************************************************************************
C   ENDS LOOP, GO BACK FOR NEXT ALTITUDE
C*******************************************************************************
500 CONTINUE
801 FORMAT(4E20.13)
802 FORMAT(8F10.6)
803 FORMAT(5)
804 FORMAT(15X,"MACH","F6.3","L","VIBRATION MODES"
..//10X,"I","H ORDER CONTROL LAW","10X","CHARACTERISTIC LENGTH",".
..F7.3)
805 FORMAT(10X,"ALTITUDES TO BE EVALUATED","10X,(1X,G12.6))
801 FORMAT(10X,"MODAL FREQUENCIES","10X,(1X,G12.6))
SUBROUTINE CROSS (A,B,C,I,J,K)
C
C THIS PROGRAM MULTIPLIES MATRIX "A" BY MATRIX "B". MATRIX "A" IS
C NECESSARILY AN I BY J MATRIX AND MATRIX "B" IS NECESSARILY AN
C J BY K MATRIX. THE RESULT OF THE MULTIPLICATION IS MATRIX "C" (I BY K)
C * NOTE: IF THE ARRAY "C" IS THE SAME ARRAY AS "A" OR "B" THE PROBLEM
C IS PROBABLE SCREWED UP
C
C*********************************************************
DIMENSION A(I,J),B(J,K),C(I,K)
DO 25 L=1,I
DO 25 N=1,K
C(L,N)=0.0
DO 25 M=1,J
C(L,N)=C(L,N)+A(L,M)*B(M,N)
25 CONTINUE
RETURN
END
SUBROUTINE INITPLT
C*********************************************************
C SUBROUTINE INITPLT (INITIALIZE PLOTTER)
C
C*********************************************************
C THE ENTIRE 5 LINES OF THE PROGRAM IS SUPPOSED TO INITIALIZE THE PLOTTER.
C THESE INSTRUCTIONS ARE FOR A CALCOMP PLOTTER. IF YOU ARE USING
C ANOTHER TYPE OF PLOTTER YOU MAY NEED TO CHANGE THESE INSTRUCTIONS OR
C GET RID OF THEM ALL TOGETHER. THE OTHER PROGRAMS THAT DEAL WITH
C PLOTTING INSTRUCTIONS ARE AXISPLT, POINTS, AND ZEROPLT.
C
C*********************************************************
CALL PLOTS(0,0.4)
CALL FACTOR(.7071)
RETURN
END
SUBROUTINE AXISPLT(ALT,MACH,YMAX,YMIN,XMAX,XMIN,SCALE,ISYM,TITLE)
C*********************************************************
C SUBROUTINE AXISPLT (AXIS PLOT)
C
C*********************************************************
THIS PROGRAM DRAWS THE AXISES AND THE INFORMATION NECESSARY TO
DESCRIBE WHAT THE PLOT IS OF (MACH NUMBER, ALTITUDE, TITLE...).
OH, BY THE WAY THIS IS FOR A ROOT LOCUS PLOT.
THESE PROGRAM IS WRITTEN WITH CALCOMP PLOTTER INSTRUCTIONS SO
TAKE IT OR WRITE YOUR OWN. IF DO NEED TO CHANGE IT THE OTHER
SUBROUTINES THAT ALSO USE PLOTTER INSTRUCTIONS ARE INITPLT,
POINT, ZEROPLT.

INTEGER TITLE(8)
LOGICAL ISYM
SCL=YMAX/11.
NEXP=0
IF (SCL.LT.1.) GO TO 30
10 IF (SCL.LE.10.) GO TO 50
SCL=SCL/10.
NEXP=NEXP+1
GO TO 10
30 IF (SCL.GT.1.) GO TO 50
SCL=SCL/10.
NEXP=NEXP-1
GO TO 30
50 IF (SCL.GT.5.) SCALE=10.*10.**NEXP
IF (SCL.LE.5..AND.SCL.GT.2.) SCALE=5.*10.**NEXP
IF (SCL.LE.2.) SCALE=2.*10.**NEXP
IF (SCL.GT.2..AND.SCL.LE.2.5..AND.NEXP.GE.1) SCALE=2.5*10.**NEXP
XMIN=-SCALE*6.
XMAX=SCALE*2.
YMIN=0.
CALL PLOT(0.,0.,3)
CALL PLOT(10.,795.,0.,-3)
IF (ICASE.EQ.1) CALL PLOT(6.,1.,-3)
CALL PLOT(0.,11.,2)
CALL AXIS(2.,0.,24) IMAGINARY AXIS (RAD/SEC),-24,11.,90.,0.,SCALE)
CALL AXIS(-6.,0.,19) REAL AXIS (RAD/SEC),-19,0.,0.,XMIN,SCALE)
CALL SYMBOL(-6.,-1.25,.10,12HGAIN FACTORS,0.,1)
CALL SYMBOL(-6.,-1.5,.10,7HSYMBOLS,0.,7)
CALL SYMBOL(-6.,12.,14,11HALTITUDE = 0.,11)
CALL NUMBER(999.,999.,14,AL1,0.,0.)
CALL SYMBOL(999.,999.,14,12H MACH = 0.,12)
CALL NUMBER(999.,999.,14,MACH,0.,3)
IF (.NOT.ISYM) CALL SYMBOL(999.,999.,14,14H SYMMETRIC,0.,14)
IF (.NOT.ISYM) CALL SYMBOL(999.,999.,14,15H ASYMMETRIC,0.,15)
RETURN
SUBROUTINE PQINT(MSIZE,WR,WI,SCALE,KK,XK,YMAX,YMIN,XHAX,XHIN)
C
C PROGRAM POINT (DRAWS POINTS ON THE PLOT)
C
C
C *** THIS PROGRAMS TAKES THE EIGENVALUES AND PLOTTES THEM IF THEM ARE
C WITHIN THE AXIS OF THE PLOT. EACH GAIN FACTOR HAS A DIFFERENT
C SYMBOL TO REPRESENT IT. "X" IS ALWAYS USED FOR THE POLES AND "Z"
C IS SAVED FOR THE ZEROS WHICH ARE PLOTTED IN ANOTHER PROGRAM.
C THE INSTRUCTIONS USED IN THIS PROGRAM ARE FOR A CALCOMP PLOTTER,
C SO THEM MAY NEED TO BE CHANGED. THE OTHER PROGRAMS THAT USE PLOTTER
C INSTRUCTIONS ARE INITPLT, AXISPLT, AND ZEROPLT.
C
C
DIMENSION WR(MSIZE),WI(MSIZE)
IF (XK.EQ.0.) ISYMB = 4
DO 10 I=1,MSIZE
IF (WR(I).LT.XMIN.OR.WR(I).GT.XMAX.OR.
WI(I).LT.YMIN.OR.WI(I).GT.YMAX) GO TO 10
XN=WR(I)/SCALE
YN=WI(I)/SCALE
CALL SYMBOL(XN,YN,14,ISYMB,0.,1)
10 CONTINUE
PX=-4.5+KK*.6
CALL NUMBER(PX,-1.25,.1,XK,0.,2)
CALL SYMBOL(999.,999.,25,0.,-1)
PX=PX+.3
CALL SYMBOL(PX,-1.45,ISYMB,0.,-1)
IF (ISYMB.EQ.4) ISYMB=-1
ISYMB=ISYMB+1
IF (ISYMB.EQ.4) ISYMB=5
IF (ISYMB.EQ.8) ISYMB=9
RETURN
END

SUBROUTINE ZEROPLT(MSIZE,WR,WI,SCALE,YMAX,YMIN,XMAX,XMIN,KK)
C******************************************************************************
C
C PROGRAM ZEROPLT (PLOTTES ZEROS)
C******************************************************************************

C THIS PROGRAM TAKES THE EIGENVALUES WHICH REPRESENT THE ZEROS OF
C THE MATRIX AND PLOTTES THEM IF THEY ARE WITHIN THE BONDS OF THE
C AXISES. THE INSTRUCTIONS IN THIS PROGRAM ARE FOR A CALCOMP PLOTTER
C AND MAY NEED TO BE CHANGED. THE OTHER ROUTINES THAT USE PLOTTER
C COMMANDS ARE INITPLT, AXISPLT, AND POINT. THE ZEROS ARE REPRESENTED
C BY A "Z" ON THE PLOT.
C******************************************************************************

DIMENSION WR(MSIZE),WI(MSIZE)
ISYMB=8
DO 10 I=1,MSIZE
  IF (WR(I).LT.XMIN.OR.WR(I).GT.XMAX.OR.
    . WI(I).LT.YMIN.OR.WI(I).GT.YMAX) GO TO 10
  XN=WR(I)/SCALE
  YN=WI(I)/SCALE
  CALL SYMBOL(XN,YN,.14,ISYMB,0.,-1)
10 CONTINUE
PX=-4.5+(KK+1.5)*.6
CALL SYMBOL(PX,-1.20,24,0..-1)
CALL SYMBOL(PX,-1.45,8,0..-1)
RETURN
END

SUBROUTINE FSS(A,OMEGA2,B,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MODE,OMEGA2,MO
DO 21 IR=1,NCONTR
DO 21 IC=1,MODE
A(IR+MODE*3,IC)+WB(IR,IC)
21 CONTINUE
CALL CROSS(W6,OMEGA2,WB,NCONTR,MODE,MODE)
DO 22 IR=1,NCONTR
DO 22 IC=1,MODE
A(IR+MODE*3,2*IC+1-MODE)=WB(IR,IC)
22 CONTINUE
CALL CROSS(W6,B,WB,NCONTR,MODE,MODE)
DO 23 IR=1,NCONTR
DO 23 IC=1,MODE
A(IR+MODE*3,2*IC+MODE)-WB(IR,IC)
23 CONTINUE
CALL CROSS(W9,W10,W11,MODE,1,NCONTR)
DO 24 IR=1,MODE
DO 24 IC=1,NCONTR
A(2*IR+MODE,IC+MODE)+W11(IR,IC)
24 CONTINUE
CALL CROSS(W9,W11,W12,NCONTR,MODE,NCONTR)
DO 25 IR=1,NCONTR
DO 25 IC=1,NCONTR
A(IR+MODE*3,IC+MODE)-AC(IR,IC)+W12(IR,IC)
25 CONTINUE
CALL CROSS(W91,W11,W7,MAKE,MODE,NCONTR)
DO 26 IR=1,MODE
DO 26 IC=1,NCONTR
A(IR,IC+MODE*3)+W7(IR,IC)
26 CONTINUE
RETURN
END
SUBROUTINE ARITH (SA,A,SB,B,C,NR,NC)

C******************************************************************************
C
C PROGRAM ARITH     (ARITHMETIC)
C******************************************************************************
C
C THIS PROGRAM WILL MULTIPLY A MATRIX BY A CONSTANT AND ADD IT TO SECOND
C MATRIX OF THE SAME SIZE AFTER THE SECOND MATRIX HAS BEEN MULTIPLIED BY
C A SECOND CONSTANT.
C
C SA = CONSTANT THAT WILL MULTIPLY THE "A" MATRIX
C A = NR BY NC MATRIX
C SB = CONSTANT THAT WILL MULTIPLY THE "B" MATRIX
C B = NR BY NC MATRIX
C C = NR BY NC MATRIX (SOLUTION OF (SA*A)+(SB*B))
C NR = NUMBER OF ROWS
C NC = NUMBER OF COLUMNS
C
C******************************************************************************
C
DIMENSION A(NR,NC), B(NR,NC), C(NR,NC)
DO 500 IR=1,NR
DO 501 IC=1,NC
C(IR,IC)=SA*A(IR,IC)+SB*B(IR,IC)
501 CONTINUE
500 CONTINUE
RETURN
END

SUBROUTINE GBARC (MACH,ALT,U,GBAR,INFLG)

C******************************************************************************
C
C PROGRAM GBARC
C******************************************************************************
C
C THIS PROGRAM TAKES THE MACH NUMBER AND ALTITUDE AND ESTIMATES THE
C AIRSPEED IN FT/SEC AND THE DYNAMIC PRESSURE.
C
REAL MACH
DIMENSION ALTT(10), A(10), RHO(10)
DIMENSION BMACH(10), CLCORR(10)
DATA ALTT/0.0,5000.,10000.,15000.,20000.,25000.,30000.,
 .35000.,40000.,50000./
DATA A/1116.45,1097.09,1077.40,1057.35,1036.92,1016.16,
 .996.95,973.14,968.06,968.06/
DATA RHO/.0023769,.0020482,.0017556,.0014962,.0012673,.0010663,
 .00089069,.00073281,.00058728,.00036392/
DATA BMACH/.70,.725,.75,.775,.8,.825,.85,.875,.9,.925/
DATA CLCORR/1.025,1.028,1.027,1.03,1.05,1.07,1.08,1.1,1.125,1.155/
IF (ALT.GT.0.0) GO TO 102
AJ=A(1)
RHOI=RHO(1)
GO TO 103
102 CONTINUE
DO 200 I=1,10
ISAV=I
IF (ALT.LE.ALTT(I)) GO TO 100
200 CONTINUE
PRINT 950
950 FORMAT (10X,*ALT IS OUTSIDE TABLES. WILL USE RHO=0.0, VT=968.07*)
AJ=968.07
RHOI=0.0
GO TO 103
100 CONTINUE
DZ=(ALT-ALTT(ISAV-1))/(ALTT(ISAV)-ALTT(ISAV-1))
RHOI=RHO(ISAV-1)+DZ*(RHO(ISAV)-RHO(ISAV-1))
AJ=A(ISAV-1)+DZ*(A(ISAV)-A(ISAV-1))
103 CONTINUE
U=MACH*AI
QBAR=0.5*RHOI*U*U
C LIFT-CURVE SLOPE CORRECTION FACTOR
IX=1
DO 76 JA=1,10
  IF(MACH.GE.BKMACH(JA)) IX=JA
76 CONTINUE
CORFAC=CLCORR(IX)
GBAR=GBAR*CORFAC
IF (INFLG.EQ.0) RETURN
U=U*12
GBAR=GBAR/144.
RETURN
END
SUBROUTINE MULT(A,B,C,I)
DIMENSION A(I,I),B(I,I),C(I,I)
DO 10 J=1,I
  DO 10 K=1,I
    XX=0.
    DO 11 L=1,I
      XX=XX+A(J,L)*B(L,K)
11  C(J,K)=XX
10  RETURN
END
SUBROUTINE ELMHES(NM,N,LOW,IGH,A,INT)
C
INTEGER I,J,M,N,MN,LA,NM,IGH,KPI,LOW,MM,MPI
REAL  A(NM,N)
REAL  X,Y
INTEGER INT(IGH)
C
THIS SUBROUTINE IS A TRANSLATION OF THE ALGOL PROCEDURE ELMHES,
C
NUM. MATH. 12, 349-368(1966) BY MARTIN AND WILKINSON.
C
APPLIED MATHEMATICS DIVISION, ARGONNE NATIONAL LABORATORY
C
C
GIVEN A REAL GENERAL MATRIX, THIS SUBROUTINE
C
REDUCES A SUBMATRIX SITUATED IN ROWS AND COLUMNS
C
LOW THROUGH IGH TO UPPER Hessenberg FORM BY
STABILIZED ELEMENTARY SIMILARITY TRANSFORMATIONS.

ON INPUT:

 NM MUST BE SET TO THE ROW DIMENSION OF TWO-DIMENSIONAL
 ARRAY PARAMETERS AS DECLARED IN THE CALLING PROGRAM
 DIMENSION STATEMENT;

 N IS THE ORDER OF THE MATRIX;

 LOW AND IGH ARE INTEGERS DETERMINED BY THE BALANCING
 SUBROUTINE BALANC. IF BALANC HAS NOT BEEN USED,
 SET LOW=1, IGH=N;

 A CONTAINS THE INPUT MATRIX.

ON OUTPUT:

 A CONTAINS THE HESSENBERG MATRIX. THE MULTIPLIERS
 WHICH WERE USED IN THE REDUCTION ARE STORED IN THE
 REMAINING TRIANGLE UNDER THE HESSENBERG MATRIX;

 INT CONTAINS INFORMATION ON THE ROWS AND COLUMNS
 INTERCHANGED IN THE REDUCTION.
 ONLY ELEMENTS LOW THROUGH IGH ARE USED;

 QUESTIONS AND COMMENTS SHOULD BE DIRECTED TO B. S. GARROW;

 LA = IGH - 1
 KPI = LOW + 1
 IF (LA .LT. KPI) GO TO 200
 DO 180 M = KPI, LA
 MM1 = M - 1
X = 0.00
I = M
C
Do 100 J = M, 1GH
    IF (ABS(A(I,J,MM1)) .LE. ABS(X)) GO TO 100
    X = A(I,J,MM1)
    I = J
100 CONTINUE
C
INT(M) = 1
    IF (I .EQ. M) GO TO 130
C
INTERCHANGE ROWS AND COLUMNS OF A
Do 110 J = MM1, N
    Y = A(I,J)
    A(I,J) = A(M,J)
    A(M,J) = Y
110 CONTINUE
C
Do 120 J = 1, 1GH
    Y = A(J,I)
    A(J,I) = A(J,M)
    A(J,M) = Y
120 CONTINUE
C
END INTERCHANGE
130 IF (X .EQ. 0.000) GO TO 180
    MPI = M + 1
C
Do 160 I = MPI, 1GH
    Y = A(I,MM1)
    IF (Y .EQ. 0.000) GO TO 160
    Y = Y / X
    A(I,MM1) = Y
C
Do 140 J = M, N
SUBROUTINE ELTRAN(NM,N,LOW,IGH,A,INT,ZCNM,N)

INTEGER I,J,N,KL,MM,MP,LOW,IGH,INT(N),MIP
REAL A(NM,IGH),ZCNM,N

INTEGER INT(IGH)

THIS SUBROUTINE IS A TRANSLATION OF THE ALGOL PROCEDURE ELMTRANS,
NUM. MATH. 16, 181-204(1970) BY PETERS AND WILKINSON.

THIS SUBROUTINE ACCUMULATES THE STABILIZED ELEMENTARY
SIMILARITY TRANSFORMATIONS USED IN THE REDUCTION OF A
REAL GENERAL MATRIX TO UPPER HESSENBORG FORM BY ELMHES.

ON INPUT:
NM MUST BE SET TO THE ROW DIMENSION OF TWO-DIMENSIONAL
ARRAY PARAMETERS AS DECLARED IN THE CALLING PROGRAM
DIMENSION STATEMENT;
N IS THE ORDER OF THE MATRIX;
LOW AND IGH ARE INTEGERS DETERMINED BY THE BALANCING
SUBROUTINE BALANC. IF BALANC HAS NOT BEEN USED,
SET LOW=1, IGH=N;

A CONTAINS THE MULTIPLIERS WHICH WERE USED IN THE REDUCTION BY ELMHES IN ITS LOWER TRIANGLE BELOW THE SUBDIAGONAL;

INT CONTAINS INFORMATION ON THE ROWS AND COLUMNS INTERCHANGED IN THE REDUCTION BY ELMHES. ONLY ELEMENTS LOW THROUGH IGH ARE USED.

ON OUTPUT:

Z CONTAINS THE TRANSFORMATION MATRIX PRODUCED IN THE REDUCTION BY ELMHES.

QUESTIONS AND COMMENTS SHOULD BE DIRECTED TO B. S. GARBOW, APPLIED MATHEMATICS DIVISION, ARGONNE NATIONAL LABORATORY.

-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

INITIALIZE Z TO IDENTITY MATRIX
DO 80 I = 1, N
    DO 70 J = 1, N
        Z(I, J) = 0.0EO
    DO 70 CONTINUE

KL = IGH - LOW - 1

IF (KL .LT. 1) GO TO 200

FOR MP=IGH-1 STEP -1 UNTIL LOW+1 DO --

DO 140 MM = 1, KL
    MP = IGH - MM
MP1 = MP + 1

DO 100 I = MP1, IGH
       100 Z(I,MP) = A(I,MP-1)

I = INT(MP)
IF (I .EQ. MP) GO TO 140

DO 130 J = MP, IGH
       Z(MP,J) = Z(I,J)
       Z(I,J) = 0.0EO
       CONTINUE

Z(I,MP) = 1.0EO

140 CONTINUE

200 RETURN

END

SUBROUTINE HQR2(NM,N,LOW,IGH,H,WR,WI,Z,IERR,INUM)

INTEGER I,J,K,L,M,N,EN,II,JI,LL,MM,NM,NN,
X
IGH,ITS,LOW,MP2,ENM2,IERR
REAL H(NM,N),WR(N,N),WI(N,N),Z(NM,N)
REAL P,Q,R,S,T,W,X,Y,KA,SA,VI,VK,ZZ,NORM,MACHEP
INTEGER MINO
LOGICAL NOTLAS
COMPLEX Z3
COMPLEX CMPLX
REAL T3(2)
EQUIVALENCE (Z3,T3(1))

THIS SUBROUTINE IS A TRANSLATION OF THE ALGOL PROCEDURE HQR2,
NUM. MATH. 16, 181-204 (1970) BY PETERS AND WILKINSON.
THIS SUBROUTINE FINDS THE EIGENVALUES AND EIGENVECTORS
OF A REAL UPPER HESSENBEG MATRIX BY THE QR METHOD. THE
EIGENVECTORS OF A REAL GENERAL MATRIX CAN ALSO BE FOUND
IF ELMHES AND ELTRAN OR ORTHES AND ORTRAN HAVE
BEEN USED TO REDUCE THIS GENERAL MATRIX TO HESSENBEG FORM
AND TO ACCUMULATE THE SIMILARITY TRANSFORMATIONS.

ON INPUT:

NM MUST BE SET TO THE ROW DIMENSION OF TWO-DIMENSIONAL
ARRAY PARAMETERS AS DECLARED IN THE CALLING PROGRAM
DIMENSION STATEMENT:
N IS THE ORDER OF THE MATRIX:
LOW AND IGH ARE INTEGERS DETERMINED BY THE BALANCING
SUBROUTINE BALANC. IF BALANC HAS NOT BEEN USED,
SET LOW=1, IGH=N;
H CONTAINS THE UPPER HESSENBEG MATRIX:
Z CONTAINS THE TRANSFORMATION MATRIX PRODUCED BY ELTRAN
AFTER THE REDUCTION BY ELMHES, OR BY ORTRAN AFTER THE
REDUCTION BY ORTHES, IF PERFORMED. IF THE EIGENVECTORS
OF THE HESSENBEG MATRIX ARE DESIRED, Z MUST CONTAIN THE
IDENTITY MATRIX.

ON OUTPUT:

H HAS BEEN DESTROYED;
WR AND WI CONTAIN THE REAL AND IMAGINARY PARTS,
RESPECTIVELY, OF THE EIGENVALUES. THE EIGENVALUES
ARE UNORDERED EXCEPT THAT COMPLEX CONJUGATE PAIRS
OF VALUES APPEAR CONSECUTIVELY WITH THE EIGENVALUE
HAVING THE POSITIVE IMAGINARY PART FIRST. IF AN
ERROR EXIT IS MADE. THE EIGENVALUES SHOULD BE CORRECT
FOR INDICES IERR+1,...,N:  60.
Z CONTAINS THE REAL AND IMAGINARY PARTS OF THE EIGENVECTORS. 63.
IF THE I-TH EIGENVALUE IS REAL, THE I-TH COLUMN OF Z  64.
CONTAINS ITS EIGENVECTOR. IF THE I-TH EIGENVALUE IS COMPLEX  65.
WITH POSITIVE IMAGINARY PART, THE I-TH AND (I+1)-TH  66.
EIGENVECTOR. THE EIGENVECTORS ARE UNNORMALIZED. IF AN  68.
ERROR EXIT IS MADE, NONE OF THE EIGENVECTORS HAS BEEN FOUND:  69.
IERR IS SET TO  70.
ZERO FOR NORMAL RETURN.  72.
J IF THE J-TH EIGENVALUE HAS NOT BEEN  73.
DETERMINED AFTER 30 ITERATIONS.  74.
ARITHMETIC IS REAL EXCEPT FOR THE REPLACEMENT OF THE ALGOL  76.
PROCEDURE CDIV BY COMPLEX DIVISION.  77.
QUESTIONS AND COMMENTS SHOULD BE DIRECTED TO B. S. GARBOW,  79.
APPLIED MATHEMATICS DIVISION, ARGONNE NATIONAL LABORATORY  80.
MACHEP IS A MACHINE DEPENDENT PARAMETER SPECIFYING  84.
THE RELATIVE PRECISION OF FLOATING POINT ARITHMETIC.  85.
MACHEP = 16.0E0**(-13) FOR LONG FORM ARITHMETIC  86.
ON S360
DATA MACHEP/1.E-9/
DO 5 K=1,IGH
       WH(K)=0.
       W1(K)=0.
5 CONTINUE
IERR = 0
C STORE ROOTS ISOLATED BY BALANC
   DO 50 I = 1, N
      IF (I .GE. LOW .AND. I .LE. IGH) GO TO 50
      WR(I) = H(I,I)
      WI(I) = 0.0E0
50 CONTINUE
C
C EN = IGH
C T = 0.0E0
C
C SEARCH FOR NEXT EIGENVALUES
60 IF (EN .LT. LOW) GO TO 340
   ITS = 0
   NA = EN - 1
   ENM2 = NA - 1
C
C LOOK FOR SINGLE SMALL SUB-DIAGONAL ELEMENT
C FOR L=EN STEP -1 UNTIL LOW DO --
70 DO 80 LL = LOW, EN
       L = EN + LOW - LL
       IF (L .EQ. LOW) GO TO 100
       IF (ABS(H(L,L-1)) .LE. MACHEP = ( ABS(H(L-1,L-1))
           X = ABS(H(L,L)) ) GO TO 100
80 CONTINUE
C
C FORM SHIFT
100 X = H(EN,EN)
   IF (L .EQ. EN) GO TO 270
   Y = H(NA,NA)
   W = H(EN,NA) + H(NA,EN)
   IF (L .EQ. NA) GO TO 280
   IF (ITS .EQ. 30) GO TO 1000
   IF (ITS .NE. 10 .AND. ITS .NE. 20) GO TO 130
C
C FORM EXCEPTIONAL SHIFT
T = T + X
122.
C
DO 120 I = LOW, EN
120 H(I,I) = H(I,I) - X
C
S = ABS(H(EN,NA)) + ABS(H(NA,ENM2))
X = 0.75E0 * S
Y = X
W = -0.4375E0 * S * S
130 ITS = ITS + 1
C
LOOK FOR TWO CONSECUTIVE SMALL
SUB-DIAGONAL ELEMENTS.
C
FOR M=EN-2 STEP -1 UNTIL L DO --
DO 140 MM = L, ENM2
M = ENM2 + L - MM
ZZ = H(M,M)
R = X - ZZ
S = Y - ZZ
P = (K * S - W) / H(M+1,M) + H(M,M+1)
Q = H(M+1,M+1) - ZZ - R - S
R = H(M+2,M+1)
S = ABS(P) + ABS(Q) + ABS(R)
P = P / S
Q = G / S
R = R / S
IF (M.EQ. L) GO TO 150
IF (ABS(H(M,M-1)) * (ABS(Q) + ABS(R)) .LE. MACHEP + ABS(P)) 148.
X = (ABS(H(M-1,M-1)) + ABS(2Z) + ABS(H(M+1,M+1))) GO TO 150 149.
140 CONTINUE 150.
150 MP2 = M + 2
DO 160 I = MP2, EN
H(I,I-2) = 0.0E0
IF (I .EQ. MP2) GO TO 160
H(I,I-3) = 0.0E0
160 CONTINUE 158.
C DOUBLE OR STEP INVOLVING ROWS L TO EN AND,
COLUMNS M TO EN

DO 260 K = M, NA

NOTLAS = K .NE. NA

IF (K .EQ. M) GO TO 170

P = H(K,K-1)
Q = H(K,K+1)
R = 0.0E0

IF (NOTLAS) R = H(K+2,K-1)

X = ABS(P) + ABS(Q) + ABS(R)

IF (X .EQ. 0.0E0) GO TO 260

P = P / X
Q = C / X
R = R / X

170 S = SIGN(SORT(P*P+Q*Q+R*R),P)

IF (K .EQ. M) GO TO 180

H(K,K-1) = -S * X

GO TO 190

180 IF (L .NE. M) H(K,K-1) = -H(K,K-1)

180

P = P + S
X = P / S
Y = Q / S
Z = R / S
Q = G / P
R = R / P

C

ROW MODIFICATION

DO 210 J = K, N

P = H(K,J) + Q * H(K+1,J)

IF (.NOT. NOTLAS) GO TO 200

P = P + R * H(K+2,J)

H(K+2,J) = H(K+2,J) - P * Z.

200 H(K+1,J) = H(K+1,J) - P * Y

H(K,J) = H(K,J) - P * X

210 CONTINUE

C

J = MIN0(EN,K+3)

C

CONTINUE
COLUMN MODIFICATION

DO 230 I = 1, J
   P = X * H(I,K) + Y * H(I,K+1)
   IF (.NOT. NOTLAG) GO TO 220
   IF (.NOT. NOTLAG) GO TO 220
   P = P + ZZ * H(I,K+2)
   H(I,K+2) = H(I,K+2) - P
   H(I,K+1) = H(I,K+1) - P
   H(I,K) = H(I,K) - P
   CONTINUE

ACCUMULATE TRANSFORMATIONS

DO 250 I = LOW, IGH
   P = X * Z(I,K) + Y * Z(I,K+1)
   IF (.NOT. NOTLAG) GO TO 240
   P = P + ZZ * Z(I,K+2)
   Z(I,K+2) = Z(I,K+2) - P
   Z(I,K+1) = Z(I,K+1) - P
   Z(I,K) = Z(I,K) - P
   CONTINUE

CONTINUE

GO TO 70

ONE ROOT FOUND

270 H(EN,EN) = X + T
   H(EN) = H(EN,EN)
   EN = 0.0*0
   GO TO GO

TWO ROOTS FOUND

290 P = (Y - X) / 2.0*0
   Q = P * P + W
   Z7 = SORT(ABS(Q))
310 CONTINUE
C    GO TO 330
C    COMPLEX PAIR
C 320 WR(NA) = X' + P
    WR(EN) = X + P
    WI(NA) = ZI
    WI(EN) = -ZI
330 EN = ENM2
    GO TO 60
C    ALL ROOTS FOUND. BACKSUBSTITUTE TO FIND
C    VECTORS OF UPPER TRIANGULAR FORM
340 IF (INUM.EQ.1) RETURN
    NORM=0.0E0
    K = 1
C    DO 360 I = 1, N
C    DO 250 J = K, N
350 NORM=NORM+ABS(H(I,J))
C    K = I
360 CONTINUE
C    IF (NORM .EQ. 0.0E0) GO TO 1001
C    FOR EN=N STEP -1 UNTIL 1 DO --
300 NN = I, N
    EN = N + 1 - NN
    P = WR(EN)
    Q = WI(EN)
    NA = EN - I
    IF (Q) 710, 600, 600
    CONTINUE
M = EN
H(EN,EN) = 1.0E0
IF (NA .EQ. 0) GO TO 800

FOR I = EN-1 STEP -1 UNTIL 1 DO --
  DO 700 II = I, NA
    I = EN - II
    W = H(I,I) - P
    R = H(I,EN)
    IF (M .GT. NA) GO TO 620

  DO 610 J = M, NA
    R = R + H(I,J) * H(J,EN)
  C
  IF (W(I,I) .GE. 0.0E0) GO TO 630
    ZZ = W
    S = R
    GO TO 700
  C

  M = I
  IF (W(I,I) .NE. 0.0E0) GO TO 640
    T = W
    IF (W .EQ. 0.0E0) T - MACHPE + NORM
    H(I,EN) = -R / T
    GO TO 100

C
SOLVE REAL EQUATIONS

  X = H(I,I+1)
  Y = H(I+1,I)
  Q = (W(I) - P) * (W(I) - P) + W(I+1) * W(I)
  T = (X + S) * ZZ * R / Q
  H(I+1,EN) = T
  IF (ABS(X) .LE. ABS(ZZ)) GO TO 650
    H(I+1,EN) = (-R - W * T) / X
    GO TO 700

  H(I+1,EN) = (-S - Y * T) / ZZ
  GO TO 700

CONTINUE
C END REAL VECTOR
    GO TO 800
C
C COMPLEX VECTOR
C 710 M = NA
C
C LAST VECTOR COMPONENT CHOSEN IMAGINARY SO THAT
C EIGENVECTOR MATRIX IS TRIANGULAR
    IF (ABS(H(EN,NA)) .LE. ABS(H(NA,EN))) GO TO 720
    H(NA,NA) = 0 / H(EN,NA)
    H(NA,EN) = -(H(EN,EN) - P) / H(EN,NA)
    GO TO 730
C 720 ZZ = CMPLX(0,0.E0,-H(NA,EN)) / CMPLX(H(NA,NA)-P,Q)
    H(NA,NA) = T3(1)
    H(NA,EN) = T3(2)
    730 H(EN,NA) = 0.0E0
    H(EN,EN) = 1.0E0
    ENM2 = NA - 1
    IF (ENM2 .EQ. 0) GO TO 800
C
C DO 790 II = 1, ENM2
    I = NA - II
    W = H(I,II) - P
    RA = 0.0E0
    SA = H(I,EN)
C
C DO 760 J = M, NA
    RA = RA + H(I,J) + H(J,NA)
    SA = SA + H(I,J) + H(J,EN)
C 760 CONTINUE
C
C IF (W(I,I) .GE. 0.0E0) GO TO 770
    ZZ = W
    R = RA
    S = SA
GO TO 790

M = I

IF (WI(I) .NE. 0.0E0) GO TO 780
Z3 = CMPLX(-RA,-SA) / CMPLX(W,0)
H(I,NA) = T3(1)
H(I,EN) = T3(2)
GO TO 790

C
C SOLVE COMPLEX EQUATIONS

760  
X = H(I,I+1)
Y = H(I+1,I)
VR = (WR(I) - P) * (WR(I) - P) + WI(I) * WI(I) - Q * Q
VI = (WR(I) - P) * 2.0E0 * Q
IF (VR .EQ. 0.0E0 .AND. VI .EQ. 0.0E0) VR = MACHEP * NORM

X

Z3 = CMPLX(X*R-ZZ*RA+Q*SA,X*S-ZZ*SA-Q*RA) / CMPLX(VR,VI)
H(I,NA) = T3(1)
H(I,EN) = T3(2)
IF (ABS(X) .LE. ABS(ZZ)) GO TO 785
H(I+1,NA) = (-RA - W * H(I,NA)) / H(I+1,EN) / X
H(I+1,EN) = (-SA - W * H(I,EN)) / H(I,NA) / X
GO TO 790

785  
Z3 = CMPLX(-R-Y*H(I,NA),-S-Y*H(I,EN)) / CMPLX(ZZ,0)
H(I+1,NA) = T3(1)
H(I+1,EN) = T3(2)
CONTINUE

C
C END COMPLEX VECTOR

800 CONTINUE

C
C END BACK SUBSTITUTION.

C VECTORS OF ISOLATED ROOTS

DO 840 I = 1, N
IF (I .GE. LOW .AND. I .LE. IGH) GO TO 840

840 J = I, N
820    Z(I,J) = H(I,J)
C
840 CONTINUE
C
MULTIPLY BY TRANSFORMATION MATRIX TO GIVE
VECTORS OF ORIGINAL FULL MATRIX.
FOR J=N STEP -1 UNTIL LOW DO -- 

DO 880 JJ = LOW, N
   J = N + LOW - JJ
   M = MINO(J,IGH)
C
DO 880 I = LOW, IGH
   ZZ = 0.0EO
C
DO 860 K = LOW, M
   ZZ = ZZ + Z(I,K) * H(K,J)
C
Z(I,J) = ZZ
880 CONTINUE
C
GO TO 1001
C
SET ERROR -- NO CONVERGENCE IN AN
EIGENVALUE AFTER 30 ITERATIONS

1000 IERR = EN
1001 RETURN
C
C    LAST CARD OF HGRZ
END
END OF FILE
<table>
<thead>
<tr>
<th>DATA ARR-1 OLD DATA SYMMETRIC ROOT LOCUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>15000.</td>
</tr>
<tr>
<td>15</td>
</tr>
<tr>
<td>.2000000000000E+01</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>1000000000000E+01</td>
</tr>
<tr>
<td>5906000000000E+03</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>2400000000000E+03</td>
</tr>
<tr>
<td>2250000000000E+04</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>1000000000000E+01</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>1532200000000E+03</td>
</tr>
<tr>
<td>.3000000000000E+04</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>0.</td>
</tr>
<tr>
<td>Value 1</td>
</tr>
<tr>
<td>--------</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>MODAL FREQUENCIES</td>
</tr>
<tr>
<td>-----------------</td>
</tr>
<tr>
<td>MODAL DAMPING</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GENERALIZED MODES</th>
<th>1.000000</th>
<th>.25213E-01</th>
<th>.160780</th>
<th>.556518</th>
<th>.494800</th>
<th>.94747E-02</th>
<th>.16548E-01</th>
<th>.23944E-01</th>
<th>.19318E-01</th>
</tr>
</thead>
</table>

**MATLAB**

```
% Modal Analysis

% Define matrices

% Modal Frequencies

% Modal Damping

% Generalized Modes

% MATLAB Code
```
<p>| | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-1.02747</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.81477</td>
<td>9.81477</td>
<td>10.2268</td>
<td>-48.8601</td>
<td>52.5553</td>
<td>6.56578</td>
<td>-10.6812</td>
</tr>
<tr>
<td>44.1055</td>
<td>-7.34028</td>
<td>-17.0800</td>
<td>46.8860</td>
<td>-4.58982</td>
<td>-4.34727</td>
<td>-28.0257</td>
</tr>
<tr>
<td>14.6715</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>270.2146</td>
<td>18.63730</td>
<td>31.5006</td>
<td>-114.414</td>
<td>111.663</td>
<td>4.01091</td>
<td>-24.2607</td>
</tr>
<tr>
<td>74.6555</td>
<td>-13.7556</td>
<td>-25.0205</td>
<td>89.1889</td>
<td>-98.4409</td>
<td>7.82316</td>
<td>29.3405</td>
</tr>
<tr>
<td>14.3431</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.86803</td>
<td>-5.62064</td>
<td>-6.27999</td>
<td>1.30866</td>
<td>-7.14641</td>
<td>-10.2893</td>
<td>-13.4445</td>
</tr>
<tr>
<td>0.15035</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.56605</td>
<td>-42.9031</td>
<td>-6.83233</td>
<td>1.72663</td>
<td>-6.95976</td>
<td>-6.45167</td>
<td>-21.1774</td>
</tr>
<tr>
<td>8.34703</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14.6288</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>402.277</td>
<td>6.50635</td>
<td>6.33758</td>
<td>0.70327</td>
<td>3.76628</td>
<td>15.6161</td>
<td>39.0244</td>
</tr>
<tr>
<td>130.493</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.57300</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**P3 Matrix**

<p>| | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-54.8443</td>
<td>-4.14780</td>
<td>-5.45235</td>
<td>27.0635</td>
<td>-20.6957</td>
<td>-18.49736</td>
<td>-0.01</td>
</tr>
<tr>
<td>2.3241</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>142.207</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>144.210</td>
<td>5.07761</td>
<td>-7.06098</td>
<td>62.1113</td>
<td>-84.4174</td>
<td>-18.1783</td>
<td>1.13760</td>
</tr>
<tr>
<td>164.044</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>336.42</td>
<td>-2.65690</td>
<td>-16.0479</td>
<td>42.6269</td>
<td>81.7966</td>
<td>17.5759</td>
<td>1.84187</td>
</tr>
<tr>
<td>56.108</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>311.183</td>
<td>1.00302</td>
<td>-4.26628</td>
<td>63.5948</td>
<td>-82.3264</td>
<td>-14.7557</td>
<td>24.2325</td>
</tr>
<tr>
<td>67.1345</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>273.055</td>
<td>-3.35841</td>
<td>-2.72163</td>
<td>17.1903</td>
<td>-17.2644</td>
<td>-2.05395</td>
<td>4.64010</td>
</tr>
<tr>
<td>6.62809</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.4471</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>182.342</td>
<td>4.35205</td>
<td>8.05380</td>
<td>-17.5346</td>
<td>15.7751</td>
<td>-5.87598</td>
<td>-11.6669</td>
</tr>
<tr>
<td>28.281</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.42029</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>174.724</td>
<td>-2.74706</td>
<td>-9.51402</td>
<td>71.1700</td>
<td>-70.9855</td>
<td>5.11534</td>
<td>83.0102</td>
</tr>
<tr>
<td>-43.4916</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-59.8718</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The image contains a table with numerical data, which appears to be related to an engineering or scientific context, possibly dealing with matrix computations or eigenvalue problems. The table includes columns for real and imaginary parts of real matrices, frequencies, and damping factors. Here is the table converted into a readable format:

<table>
<thead>
<tr>
<th>RD MATRIX</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.3700</td>
<td>37.7303</td>
<td>1.00041</td>
<td>19.8088</td>
<td>59.3500</td>
<td>2.75265</td>
<td>3.31657</td>
<td>-5.59800</td>
</tr>
<tr>
<td>-0.21926</td>
<td>-100.065</td>
<td>-20.1985</td>
<td>962.302</td>
<td>1092.43</td>
<td>5.75900</td>
<td>117.830</td>
<td>29.8954</td>
</tr>
<tr>
<td>-0.07041</td>
<td>8.46764</td>
<td>5.05188</td>
<td>-506.689</td>
<td>-536.563</td>
<td>-6.62402</td>
<td>-56.3762</td>
<td>-5.46069</td>
</tr>
<tr>
<td>2.14018</td>
<td>-46.408</td>
<td>-6.60378</td>
<td>434.343</td>
<td>466.390</td>
<td>3.12325</td>
<td>47.3578</td>
<td>4.17718</td>
</tr>
<tr>
<td>-4.27817</td>
<td>-75.858</td>
<td>-5.56313</td>
<td>461.148</td>
<td>486.285</td>
<td>4.18201</td>
<td>57.4832</td>
<td>1.11937</td>
</tr>
<tr>
<td>30.0662</td>
<td>-11.2690</td>
<td>3.35605E-01</td>
<td>117.567</td>
<td>126.592</td>
<td>-.449819</td>
<td>12.3419</td>
<td>-.63884</td>
</tr>
<tr>
<td>-1.72449</td>
<td>-9.65773</td>
<td>7.03347</td>
<td>29.3197</td>
<td>22.9743</td>
<td>-.102576</td>
<td>-1.64757</td>
<td>22.3414</td>
</tr>
<tr>
<td>-117.635</td>
<td>3413.391</td>
<td>26.1272</td>
<td>2.35516</td>
<td>-88.0053</td>
<td>-60.7408</td>
<td>-509742</td>
<td>-14.7549</td>
</tr>
<tr>
<td>3.97418</td>
<td>-140.348</td>
<td>-17.2109</td>
<td>11.5710</td>
<td>72.0295</td>
<td>65.8521</td>
<td>.925140</td>
<td>1.34323</td>
</tr>
<tr>
<td>-7.13330</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- END OF INPUT DATA -

MACH = .025900
ALT = 15000.0
ODeR = 4.29200
VTRVL = 1.19671

THE GAIN FACTOR = 0.60

EIGENVALUES COMPUTED. ERROR CODE = 0

<p>| REAL PART | IMAGINARY PART | FREQUENCY | FREQUENCY | DAMPING FACTOR |</p>
<table>
<thead>
<tr>
<th>RAD/SEC</th>
<th>RAD/SEC</th>
<th>RAD/SEC</th>
<th>CYCLE/SEC</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-10812.44</td>
<td>0.00</td>
<td>10812.44</td>
<td>1732.67</td>
<td>1.0000</td>
</tr>
<tr>
<td>263.30</td>
<td>3716.32</td>
<td>3726.26</td>
<td>587.47</td>
<td>-0.0000</td>
</tr>
<tr>
<td>263.30</td>
<td>2710.32</td>
<td>2726.26</td>
<td>587.47</td>
<td>-0.0000</td>
</tr>
<tr>
<td>-2635.31</td>
<td>0.00</td>
<td>2635.31</td>
<td>422.55</td>
<td>1.0000</td>
</tr>
<tr>
<td>-1500.09</td>
<td>0.00</td>
<td>1500.00</td>
<td>240.52</td>
<td>1.0000</td>
</tr>
<tr>
<td>Value 1</td>
<td>Value 2</td>
<td>Value 3</td>
<td>Value 4</td>
<td>Value 5</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>Value 1</td>
<td>Value 2</td>
<td>Value 3</td>
<td>Value 4</td>
<td>Value 5</td>
</tr>
<tr>
<td>Value 1</td>
<td>Value 2</td>
<td>Value 3</td>
<td>Value 4</td>
<td>Value 5</td>
</tr>
<tr>
<td>Value 1</td>
<td>Value 2</td>
<td>Value 3</td>
<td>Value 4</td>
<td>Value 5</td>
</tr>
<tr>
<td>Value 1</td>
<td>Value 2</td>
<td>Value 3</td>
<td>Value 4</td>
<td>Value 5</td>
</tr>
<tr>
<td>Value 1</td>
<td>Value 2</td>
<td>Value 3</td>
<td>Value 4</td>
<td>Value 5</td>
</tr>
<tr>
<td>Value 1</td>
<td>Value 2</td>
<td>Value 3</td>
<td>Value 4</td>
<td>Value 5</td>
</tr>
<tr>
<td>Value 1</td>
<td>Value 2</td>
<td>Value 3</td>
<td>Value 4</td>
<td>Value 5</td>
</tr>
<tr>
<td>Value 1</td>
<td>Value 2</td>
<td>Value 3</td>
<td>Value 4</td>
<td>Value 5</td>
</tr>
<tr>
<td>Value 1</td>
<td>Value 2</td>
<td>Value 3</td>
<td>Value 4</td>
<td>Value 5</td>
</tr>
<tr>
<td>Value 1</td>
<td>Value 2</td>
<td>Value 3</td>
<td>Value 4</td>
<td>Value 5</td>
</tr>
<tr>
<td>Value 1</td>
<td>Value 2</td>
<td>Value 3</td>
<td>Value 4</td>
<td>Value 5</td>
</tr>
<tr>
<td>Value 1</td>
<td>Value 2</td>
<td>Value 3</td>
<td>Value 4</td>
<td>Value 5</td>
</tr>
</tbody>
</table>

**THE GAIN FACTOR**

**OF POOR QUALITY**

**ORIGIN OF FACT 1**
<table>
<thead>
<tr>
<th>REAL PART</th>
<th>IMAGINARY PART</th>
<th>FREQUENCY</th>
<th>FREQUENCY</th>
<th>DAMPING FACTOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAD/SEC</td>
<td>RAD/SEC</td>
<td>RAD/SEC</td>
<td>CYCLE/SEC</td>
<td></td>
</tr>
<tr>
<td>0.81244</td>
<td>0.0000</td>
<td>1.081244</td>
<td>1.73387</td>
<td>1.0000</td>
</tr>
<tr>
<td>253.70</td>
<td>3716.81</td>
<td>3726.26</td>
<td>597.47</td>
<td>0.9708</td>
</tr>
<tr>
<td>-2634.64</td>
<td>-3716.81</td>
<td>2634.64</td>
<td>427.47</td>
<td>1.0000</td>
</tr>
<tr>
<td>-1632.68</td>
<td>0.0000</td>
<td>1632.68</td>
<td>183.71</td>
<td>1.0000</td>
</tr>
<tr>
<td>-244.66</td>
<td>1234.72</td>
<td>1234.72</td>
<td>201.48</td>
<td>1.0000</td>
</tr>
<tr>
<td>-1307.10</td>
<td>0.0000</td>
<td>1307.10</td>
<td>209.38</td>
<td>1.0000</td>
</tr>
<tr>
<td>-530.76</td>
<td>237.34</td>
<td>237.34</td>
<td>162.08</td>
<td>0.9709</td>
</tr>
<tr>
<td>-590.76</td>
<td>-237.34</td>
<td>237.34</td>
<td>162.08</td>
<td>0.9709</td>
</tr>
<tr>
<td>-2.02</td>
<td>655.33</td>
<td>655.33</td>
<td>103.00</td>
<td>0.9931</td>
</tr>
<tr>
<td>-468.10</td>
<td>542.11</td>
<td>542.11</td>
<td>86.92</td>
<td>0.9709</td>
</tr>
<tr>
<td>-136.05</td>
<td>533.51</td>
<td>533.51</td>
<td>88.28</td>
<td>0.9709</td>
</tr>
<tr>
<td>-18.35</td>
<td>496.42</td>
<td>496.42</td>
<td>79.68</td>
<td>0.9709</td>
</tr>
<tr>
<td>-10.05</td>
<td>-496.42</td>
<td>496.42</td>
<td>79.68</td>
<td>0.9709</td>
</tr>
<tr>
<td>-3.36</td>
<td>450.87</td>
<td>450.87</td>
<td>73.74</td>
<td>0.9986</td>
</tr>
<tr>
<td>-3.86</td>
<td>-450.87</td>
<td>450.87</td>
<td>73.74</td>
<td>0.9986</td>
</tr>
<tr>
<td>-3.04</td>
<td>414.35</td>
<td>414.35</td>
<td>66.44</td>
<td>0.9985</td>
</tr>
<tr>
<td>-3.34</td>
<td>-414.33</td>
<td>414.33</td>
<td>66.44</td>
<td>0.9985</td>
</tr>
<tr>
<td>-110.04</td>
<td>376.33</td>
<td>376.33</td>
<td>60.34</td>
<td>0.9989</td>
</tr>
<tr>
<td>-110.04</td>
<td>358.60</td>
<td>358.60</td>
<td>60.34</td>
<td>0.9989</td>
</tr>
<tr>
<td>-12.00</td>
<td>307.52</td>
<td>307.52</td>
<td>45.35</td>
<td>0.916</td>
</tr>
<tr>
<td>-12.80</td>
<td>-201.75</td>
<td>201.75</td>
<td>45.35</td>
<td>0.916</td>
</tr>
<tr>
<td>-118.04</td>
<td>254.22</td>
<td>254.22</td>
<td>40.76</td>
<td>0.9475</td>
</tr>
<tr>
<td>-1.45</td>
<td>254.78</td>
<td>254.78</td>
<td>40.85</td>
<td>0.9475</td>
</tr>
<tr>
<td>-3.00</td>
<td>202.76</td>
<td>202.76</td>
<td>32.51</td>
<td>0.9475</td>
</tr>
<tr>
<td>-3.00</td>
<td>-202.76</td>
<td>202.76</td>
<td>32.51</td>
<td>0.9475</td>
</tr>
<tr>
<td>-141.44</td>
<td>0.0000</td>
<td>141.44</td>
<td>22.68</td>
<td>1.0000</td>
</tr>
<tr>
<td>-135.50</td>
<td>147.79</td>
<td>147.79</td>
<td>23.68</td>
<td>1.0000</td>
</tr>
<tr>
<td>-10.50</td>
<td>147.79</td>
<td>147.79</td>
<td>23.68</td>
<td>1.0000</td>
</tr>
<tr>
<td>-9.47</td>
<td>126.15</td>
<td>126.15</td>
<td>20.23</td>
<td>0.9701</td>
</tr>
<tr>
<td>-5.47</td>
<td>126.15</td>
<td>126.15</td>
<td>20.23</td>
<td>0.9701</td>
</tr>
<tr>
<td>-2.01</td>
<td>101.42</td>
<td>101.42</td>
<td>16.26</td>
<td>0.919</td>
</tr>
<tr>
<td>-78.64</td>
<td>78.64</td>
<td>78.64</td>
<td>12.61</td>
<td>1.0000</td>
</tr>
<tr>
<td>-33.60</td>
<td>75.00</td>
<td>75.00</td>
<td>12.17</td>
<td>0.8427</td>
</tr>
<tr>
<td>REAL PART</td>
<td>IMAGINARY PART</td>
<td>FREQUENCY</td>
<td>CYCLE/SEC</td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>----------------</td>
<td>-----------</td>
<td>-----------</td>
<td></td>
</tr>
<tr>
<td>-10812.34</td>
<td>0.00</td>
<td>1.733.67</td>
<td>-1.0000</td>
<td></td>
</tr>
<tr>
<td>1735.47</td>
<td>0.00</td>
<td>0.0000</td>
<td>1.0000</td>
<td></td>
</tr>
<tr>
<td>238.91</td>
<td>0.00</td>
<td>0.0000</td>
<td>1.0000</td>
<td></td>
</tr>
</tbody>
</table>

**Eigenvalues Computed, Error Code:** 1

**Gain Factor:** 0.5 (11)

**Eigenvalues**

-2.02 -655.30 105.07 0.0031
\[
\begin{array}{cccccc}
\text{REAL PART (RAD/SEC)} & \text{IMAGINARY PART (RAD/SEC)} & \text{FREQUENCY (RAD/SEC)} & \text{FREQUENCY (CYCLE/SEC)} & \text{DAMPING FACTOR}
\end{array}
\]

| -10812.44 | 0.00 | 10812.44 | 1733.67 | 1.0000 |
| 260.80 | 3718.80 | 3726.23 | 3726.23 | 0.0708 |
| 260.80 | -3718.80 | 3726.23 | 3726.23 | 1.0000 |
| -227.27 | 1247.02 | 1267.35 | 1267.35 | 0.0000 |
| 1146.22 | 3172.20 | 975.35 | 975.35 | 1.0000 |
| 915.25 | -3172.20 | 975.35 | 975.35 | 0.0000 |
| -2.03 | 655.20 | 655.20 | 655.20 | 0.0000 |
| -2.03 | -655.20 | 655.20 | 655.20 | 0.0000 |
| -427.67 | 367.26 | 563.72 | 563.72 | 0.0000 |
| 135.74 | 534.14 | 551.12 | 551.12 | 0.0000 |
| 135.74 | -534.14 | 551.12 | 551.12 | 0.0000 |
| -18.12 | 497.00 | 497.30 | 497.30 | 0.0000 |
| -0.75 | 460.61 | 460.61 | 460.61 | 0.0000 |
| -0.75 | -460.61 | 460.61 | 460.61 | 0.0000 |
| -3.04 | 414.33 | 414.34 | 414.34 | 0.0000 |
| -3.04 | -414.33 | 414.34 | 414.34 | 0.0000 |

**THE GAIN FACTOR = 1.000**

**EIGENVALUES COMPUTED, ERROR CODE = 0**
<table>
<thead>
<tr>
<th>REAL PART</th>
<th>IMAGINARY PART</th>
<th>FREQUENCY</th>
<th>FREQUENCY</th>
<th>DAMPING FACTOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAD/SEC</td>
<td>RAD/SEC</td>
<td>RAD/SEC</td>
<td>CYCLE/SEC</td>
<td></td>
</tr>
<tr>
<td>-10812.44</td>
<td>6.00</td>
<td>10912.44</td>
<td>1732.87</td>
<td>1.0000</td>
</tr>
<tr>
<td>263.61</td>
<td>3716.89</td>
<td>3726.24</td>
<td>597.47</td>
<td>-0.0708</td>
</tr>
<tr>
<td>263.61</td>
<td>-3716.89</td>
<td>3726.24</td>
<td>597.47</td>
<td>-0.0708</td>
</tr>
<tr>
<td>-203.41</td>
<td>6.00</td>
<td>2031.41</td>
<td>421.82</td>
<td>1.0000</td>
</tr>
<tr>
<td>-1015.64</td>
<td>1146.86</td>
<td>1146.86</td>
<td>1053.84</td>
<td>-0.0000</td>
</tr>
<tr>
<td>-200.12</td>
<td>1293.89</td>
<td>1293.89</td>
<td>205.22</td>
<td>-1.0000</td>
</tr>
<tr>
<td>-200.12</td>
<td>-1293.89</td>
<td>1293.89</td>
<td>205.22</td>
<td>-1.0000</td>
</tr>
<tr>
<td>-1014.27</td>
<td>350.18</td>
<td>1155.32</td>
<td>215.24</td>
<td>1.0000</td>
</tr>
<tr>
<td>-1014.27</td>
<td>-350.18</td>
<td>1155.32</td>
<td>215.24</td>
<td>1.0000</td>
</tr>
<tr>
<td>-708.88</td>
<td>55.96</td>
<td>708.88</td>
<td>105.03</td>
<td>1.0000</td>
</tr>
<tr>
<td>-2.10</td>
<td>0.00</td>
<td>655.00</td>
<td>105.03</td>
<td>0.0002</td>
</tr>
<tr>
<td>-2.10</td>
<td>0.00</td>
<td>655.00</td>
<td>105.03</td>
<td>0.0002</td>
</tr>
</tbody>
</table>

THE GAIN FACTOR = 2.000

EIGENVALUES COMPUTED, ERROR CODE = 0
<table>
<thead>
<tr>
<th>REAL PART</th>
<th>IMAGINARY PART</th>
</tr>
</thead>
<tbody>
<tr>
<td>-432.60</td>
<td>387.69</td>
</tr>
<tr>
<td>-387.60</td>
<td>567.69</td>
</tr>
<tr>
<td>-335.59</td>
<td>535.15</td>
</tr>
<tr>
<td>-335.28</td>
<td>497.01</td>
</tr>
<tr>
<td>-17.32</td>
<td>-487.01</td>
</tr>
<tr>
<td>2.95</td>
<td>-457.65</td>
</tr>
<tr>
<td>-3.75</td>
<td>-414.03</td>
</tr>
<tr>
<td>-9.24</td>
<td>-361.74</td>
</tr>
<tr>
<td>-9.24</td>
<td>-301.74</td>
</tr>
<tr>
<td>-6.17</td>
<td>-305.81</td>
</tr>
<tr>
<td>-31.78</td>
<td>275.19</td>
</tr>
<tr>
<td>-31.70</td>
<td>275.19</td>
</tr>
<tr>
<td>1.36</td>
<td>254.80</td>
</tr>
<tr>
<td>-1.28</td>
<td>-254.80</td>
</tr>
<tr>
<td>2.68</td>
<td>189.68</td>
</tr>
<tr>
<td>2.68</td>
<td>199.68</td>
</tr>
<tr>
<td>-134.94</td>
<td>0.00</td>
</tr>
<tr>
<td>-40.06</td>
<td>164.96</td>
</tr>
<tr>
<td>-40.06</td>
<td>164.96</td>
</tr>
<tr>
<td>-7.09</td>
<td>144.33</td>
</tr>
<tr>
<td>-7.69</td>
<td>-144.33</td>
</tr>
<tr>
<td>-1.09</td>
<td>-101.44</td>
</tr>
<tr>
<td>62.69</td>
<td>0.00</td>
</tr>
<tr>
<td>47.78</td>
<td>51.01</td>
</tr>
<tr>
<td>47.76</td>
<td>-51.01</td>
</tr>
<tr>
<td>-9.70</td>
<td>20.41</td>
</tr>
<tr>
<td>-9.70</td>
<td>-20.41</td>
</tr>
<tr>
<td>-1.90</td>
<td>1.39</td>
</tr>
<tr>
<td>-36.39</td>
<td>292.79</td>
</tr>
<tr>
<td>-38.39</td>
<td>292.79</td>
</tr>
</tbody>
</table>

The zeros are located at:

- REAL PART
- IMAGINARY PART

<table>
<thead>
<tr>
<th>RADS/SEC</th>
<th>RADS/SEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>-11034.63</td>
<td>0.00</td>
</tr>
<tr>
<td>-4702.10</td>
<td>7047.87</td>
</tr>
<tr>
<td>-4703.10</td>
<td>7047.87</td>
</tr>
<tr>
<td>3246.91</td>
<td>7142.08</td>
</tr>
<tr>
<td>3246.91</td>
<td>7142.08</td>
</tr>
<tr>
<td>-8401.13</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>----</td>
<td>-----</td>
</tr>
<tr>
<td>0</td>
<td>7407.26</td>
</tr>
<tr>
<td>1493.42</td>
<td>3338.80</td>
</tr>
<tr>
<td>-2599.16</td>
<td>0.00</td>
</tr>
<tr>
<td>781.25</td>
<td>173.16</td>
</tr>
<tr>
<td>781.25</td>
<td>173.16</td>
</tr>
<tr>
<td>-668.29</td>
<td>0.00</td>
</tr>
<tr>
<td>-2.70</td>
<td>854.03</td>
</tr>
<tr>
<td>-2.70</td>
<td>854.03</td>
</tr>
<tr>
<td>-437.72</td>
<td>366.90</td>
</tr>
<tr>
<td>-437.72</td>
<td>366.90</td>
</tr>
<tr>
<td>141.27</td>
<td>534.13</td>
</tr>
<tr>
<td>141.57</td>
<td>534.13</td>
</tr>
<tr>
<td>-4.46</td>
<td>452.61</td>
</tr>
<tr>
<td>-4.46</td>
<td>452.61</td>
</tr>
<tr>
<td>-4.08</td>
<td>414.20</td>
</tr>
<tr>
<td>-4.08</td>
<td>414.20</td>
</tr>
<tr>
<td>-13.26</td>
<td>302.79</td>
</tr>
<tr>
<td>-13.26</td>
<td>302.79</td>
</tr>
<tr>
<td>-1.44</td>
<td>234.89</td>
</tr>
<tr>
<td>-1.44</td>
<td>234.89</td>
</tr>
<tr>
<td>-60.00</td>
<td>309.08</td>
</tr>
<tr>
<td>-60.00</td>
<td>309.08</td>
</tr>
<tr>
<td>-1.21</td>
<td>200.50</td>
</tr>
<tr>
<td>-1.21</td>
<td>200.50</td>
</tr>
<tr>
<td>-133.52</td>
<td>0.00</td>
</tr>
<tr>
<td>-9.80</td>
<td>145.05</td>
</tr>
<tr>
<td>-9.80</td>
<td>145.05</td>
</tr>
<tr>
<td>-1.82</td>
<td>101.45</td>
</tr>
<tr>
<td>-1.82</td>
<td>101.45</td>
</tr>
<tr>
<td>-82.82</td>
<td>0.00</td>
</tr>
<tr>
<td>-50.00</td>
<td>160.39</td>
</tr>
<tr>
<td>-50.00</td>
<td>160.39</td>
</tr>
<tr>
<td>-50.00</td>
<td>160.39</td>
</tr>
<tr>
<td>-50.00</td>
<td>160.39</td>
</tr>
<tr>
<td>0.99</td>
<td>1.14</td>
</tr>
<tr>
<td>0.99</td>
<td>1.14</td>
</tr>
<tr>
<td>-1.16</td>
<td>0.00</td>
</tr>
<tr>
<td>-38.39</td>
<td>282.79</td>
</tr>
<tr>
<td>-38.39</td>
<td>282.79</td>
</tr>
</tbody>
</table>
BIBLIOGRAPHY


End of Document