NASA CR-176994

NASA-CR-176994
19860019487

A Reproduced Copy

OF

N86-28959

Reproduced for NASA

by the

NASA Scientific and Technical Information Facility

NF01698

FFNo 672 Aug 65

(NASA-CR-176994) FLUTTER PREDICTION FOR A WING WITH ACTIVE AILERON CONTROL Final Report (California Polytechnic State Univ.) 112 p CSCL 01C N86-28959
Unclas
G3/08 43346

FLUTTER PREDICTION FOR A WING WITH ACTIVE AILERON CONTROL

FINAL REPORT

PRINCIPAL INVESTIGATORS

Kevin Penning

Doral R. Sandlin

June 1983

California Polytechnic State University

Aeronautical Engineering Department

San Luis Obispo, California

Grant No. NCC4-1

N86-28959#

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

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_i	ii element of generalized damping matrix
B	generalized damping matrix
d_i	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
k	reduced frequency
k_a	aileron mode stiffness (from control law)
k_i	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_i	ii element of generalized mass matrix

m_{ij}	ij element of generalized mass matrix
M	generalized mass matrix
P1,P2, P3,R0	matrices representing aerodynamic influence coeficients
P1',P2', P3',R0'	nondimensional matrices representing aerodynamic 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 vibrational 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

ORIGINAL PAGE IS
OF POOR QUALITY

"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 aeroelastic 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 6.8 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 solved 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 eachother 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 it's derivatives. The equations of motion are represented by

$$(m_i) * u_i'' + (b_i) * u_i' + (k_i) * u_i = 0$$

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 notion:

$$\{U\} = \begin{Bmatrix} U_1 \\ U_2 \\ U_3 \\ \cdot \\ \cdot \\ U_n \end{Bmatrix} \qquad [M] = \begin{bmatrix} m_1 & 0 & 0 & . & . & 0 \\ 0 & m_2 & 0 & . & . & 0 \\ 0 & 0 & m_3 & . & . & 0 \\ . & . & . & . & . & 0 \\ . & . & . & . & . & 0 \\ 0 & 0 & 0 & 0 & 0 & m_n \end{bmatrix}$$

$$[B] = \begin{bmatrix} b_1 & 0 & 0 & . & . & 0 \\ 0 & b_2 & 0 & . & . & 0 \\ 0 & 0 & b_3 & . & . & 0 \\ . & . & . & . & . & 0 \\ . & . & . & . & . & 0 \\ 0 & 0 & 0 & 0 & 0 & b_n \end{bmatrix} \qquad [K] = \begin{bmatrix} k_1 & 0 & 0 & . & . & 0 \\ 0 & k_2 & 0 & . & . & 0 \\ 0 & 0 & k_3 & . & . & 0 \\ . & . & . & . & . & 0 \\ . & . & . & . & . & 0 \\ 0 & 0 & 0 & 0 & 0 & k_n \end{bmatrix}$$

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 multyplying 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].

$$\begin{bmatrix} m_1 & 0 & . & . & 0 \\ 0 & m_2 & . & . & 0 \\ . & . & . & . & 0 \\ . & . & . & . & 0 \\ 0 & 0 & 0 & 0 & m_n \end{bmatrix}^{-1} = \begin{bmatrix} 1/m_1 & 0 & . & . & 0 \\ 0 & 1/m_2 & . & . & 0 \\ . & . & . & . & 0 \\ . & . & . & . & 0 \\ 0 & 0 & 0 & 0 & 1/m_n \end{bmatrix}$$

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

$$\begin{Bmatrix} U' \\ U'' \end{Bmatrix} = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}B \end{bmatrix} \begin{Bmatrix} U \\ U' \end{Bmatrix}$$

Because both $[M]^{-1}$ and [K] are diagonal matrices, their product will also be a diagonal matrix.

$$[M]^{-1}[K] = \begin{bmatrix} k_1/m_1 & 0 & 0 & . & . & 0 \\ 0 & k_2/m_2 & 0 & . & . & 0 \\ 0 & 0 & k_3/m_3 & . & . & 0 \\ . & . & . & . & . & 0 \\ . & . & . & . & . & 0 \\ 0 & 0 & 0 & 0 & 0 & k_n/m_n \end{bmatrix}$$

The same statement holds true for the product of $[M]^{-1}$ and [B].

$$[M]^{-1}[B] = \begin{bmatrix} b_1/m_1 & 0 & 0 & . & . & 0 \\ 0 & b_2/m_2 & 0 & . & . & 0 \\ 0 & 0 & b_3/m_3 & . & . & 0 \\ . & . & . & . & . & 0 \\ . & . & . & . & . & 0 \\ 0 & 0 & 0 & 0 & 0 & b_n/m_n \end{bmatrix}$$

Two relationships that are helpful in solving vibration problems are

$$k_i/m_i = w_i{**}2$$

$$b_i/m_i = 2{*}d_i{*}w_i$$

where "w_i" is the natural frequency and "d_i" is the damping factor. Substituting these relationships into the matrices

$$[M]^{-1}[K] = \begin{bmatrix} w_1{**}2 & 0 & 0 & . & . & 0 \\ 0 & w_2{**}2 & 0 & . & . & 0 \\ 0 & 0 & w_3{**}2 & . & . & 0 \\ . & . & . & . & . & 0 \\ . & . & . & . & . & 0 \\ 0 & 0 & 0 & 0 & 0 & w_n{**}2 \end{bmatrix}$$

$$[M]^{-1}[B] = \begin{bmatrix} 2{*}d_1{*}w_1 & 0 & 0 & . & . & 0 \\ 0 & 2{*}d_2{*}w_2 & 0 & . & . & 0 \\ 0 & 0 & 2{*}d_3{*}w_3 & . & . & 0 \\ . & . & . & . & . & 0 \\ . & . & . & . & . & 0 \\ 0 & 0 & 0 & 0 & 0 & 2{*}d_n{*}w_n \end{bmatrix}$$

The natural frequency for a structure can be obtained from any program that does vibrational analysis such as "NASTRAN" 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 pr ides 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

$$[M]\{U''\} + [B]\{U'\} + [K]\{U\} = \{L\}$$

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

$$L(s) = q * [Q(s)] * \{U\}$$

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

$$k = (b * w) / v$$

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

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

where [P1'], [P2'], [P3'] and [RO'] are constants determined by performing a least squares curve firt 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

$$[Q(s)] = (sI*b/v - [RO'])^{-1} * ([P1'](s*b/v)^{2} + [P2']s*b/v + [p3'])$$

Multiplying [Q(s)] by U and labeling the product $\{X\}$

$$\{X\} = [Q(s)]\{U\}$$

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

$$(sI + [RO']*v/b) \{X\} = ([P1'](s)^{2} * b/v + [P2']s + [P3'] * v/b) \{U\}$$

For the ease of writing and to agree with the nomenclature used by other authors the following notation will be used:

$$[RO] = v/b*[RO']$$

$$[P1] = b/v*[P1']$$

$$[P2] = [P2']$$

$$[P3] = v/b*[P3']$$

Note that [P1], [P2], [P3] and [RO] must be recalculated for each different mach number. Using the above notion and taking the previous equation out of Laplace form,

$$[I]\{X'\} - [P1]\{U''\} = -[RO]\{X\} + [P3]\{U\} + [P2]\{U'\}$$

Remembering that

$$[M]\{U''\} + [B]\{U'\} + [K]\{U\} = \{L\}$$

and

$$\{L\} = q * \{X\}$$

the vibration equations can be written as

$$\begin{bmatrix} I & 0 & 0 \\ 0 & M & 0 \\ 0 & -P1 & I \end{bmatrix} \begin{matrix} U' \\ U'' \\ X' \end{matrix} = \begin{bmatrix} 0 & I & 0 \\ -K & -B & qI \\ P3 & P2 & -RO \end{bmatrix} \begin{Bmatrix} U \\ U' \\ X \end{Bmatrix}$$

Taking the inverse of the left hand side

$$\begin{bmatrix} I & 0 & 0 \\ 0 & M & 0 \\ 0 & -P1 & I \end{bmatrix}^{-1} = \begin{bmatrix} I & 0 & 0 \\ 0 & M^{-1} & 0 \\ 0 & P1*M^{-1} & I \end{bmatrix}$$

and multiplying both sides by the inverse

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

the vibration equations can be written as

$$\begin{Bmatrix} U' \\ U'' \\ X' \end{Bmatrix} = \begin{bmatrix} 0 & I & 0 \\ -M^{-1}*K & -M^{-1}*B & M^{-1}*q \\ P3-P1*M^{-1}*K & P2-P1*M^{-1}*B & -R0+P1*M^{-1}*q \end{bmatrix} \begin{Bmatrix} U \\ U' \\ X \end{Bmatrix}$$

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

$$g(s) = k / (s^2 + c*s + k)$$

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

$$G(s) = G'(s) * g(s)$$

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

$$G'(s) = V(s)/an(s)$$

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

$$\{Y'\} = [A]\, Y + [G]\{an\}$$

$$\{V\} = [H]\{Y\}$$

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

$$g_n'(s) = (s^2 + a*s + b) \;/\; (s^2 + c*s + d)$$

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

$$g_1'(s) = 1 \;/\; (s + d)$$

and $g_2'(s)$ is of the form

$g_2'(s) = 1 \ / \ (s^2 + c*s + d)$

[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

$g_1'(s) = y_1 / \ an$

$g_2'(s) = y_1 / \ y_1$

the following equations can be written

$y_1' = d*y_1 + an$

$y_2' = y_3$

$y_3' = y_1 - d*y_2 - c*y_3$

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 theaileron 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

$$g(s) = k_a / (s + b_a * s + k_a)$$

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

$$g(s) = u_a / \{V\}$$

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

$$k_a * \{V\} = u_a'' + b_a * u_a' + k_a * u_a$$

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 than 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_i" is the deflection at the accelerometer location for the k'th mode shape. Ewpressing "an" in matrix notation

$$\{an\} = [D] \{u''\}$$

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

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

and the analysis would require the units:

$$G(s) = ua \text{ (modal deflection)} * c / an \text{ (in/sec}^2\text{)}$$

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

$$1 \text{ g} = (32.2 \text{ ft/sec}^2) * (12 \text{ in/ft}) = 368.4 \text{ in/sec}^2$$

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

$$1 \text{ degree} = 1/10 \text{ deflection mode}$$

Therefore, the conversion factor is determined to be

$$c = (1/368.4) * (1/10)$$

The equations related to the aileron mode and control law are

$$\{Y'\} = [A]\{Y\} + [G]\{an\}$$

$$\{V\} = [H]\{Y\}$$

$$\{an\} = [D]\{U''\}$$

$$u_a'' = -k_a * u_a - b_a * u_a' + k_a * c * \{V\}$$

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

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

$$u_a'' = -k_a * u_a - b_a * u_a' + k_a * c * [H]\{Y\}$$

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

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

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 "ρ" 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-zere over the aileron. Assuming the aileron is rectangular and has a constant mass distribution, the generalized mass terms due to the aileron mode are

$$m_{ia} = -m * r_{at} * (r_{il}/3 + (r_{il} - r_{it})/2)$$

Where "m_{ia}" is the generalized mass of i'th row and the column representing the aileron mode; "m" is the mass of the aileron; "r_{at}" 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

$$[M] = \begin{bmatrix} m_1 & 0 & 0 & . & . & 0 & m_{1a} \\ 0 & m_2 & 0 & . & . & 0 & m_{2a} \\ 0 & 0 & m_3 & . & . & 0 & m_{2a} \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ 0 & 0 & 0 & . & . & m_n & m_{na} \\ 0 & 0 & 0 & . & . & 0 & 1. \end{bmatrix}$$

Putting the problem into state vector differential notation

$$\begin{bmatrix} I & 0 & 0 & 0 \\ 0 & M & 0 & 0 \\ 0 & -P1 & I & 0 \\ 0 & -GD & 0 & I \end{bmatrix} \begin{Bmatrix} U' \\ U'' \\ X' \\ Y' \end{Bmatrix} = \begin{bmatrix} 0 & I & 0 & 0 \\ -K & -B & qI & k_a cH \\ P3 & P2 & R0 & 0 \\ 0 & 0 & 0 & A \end{bmatrix} \begin{Bmatrix} U \\ U' \\ X \\ Y \end{Bmatrix}$$

The inverse of the square matrix on the left is

$$\begin{bmatrix} I & 0 & 0 & 0 \\ 0 & M & 0 & 0 \\ 0 & -P1 & I & 0 \\ 0 & -GD & 0 & I \end{bmatrix}^{-1} = \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & M^{-1} & 0 & 0 \\ 0 & P1M^{-1} & I & 0 \\ 0 & GDM^{-1} & 0 & I \end{bmatrix}$$

Multipling both sides by the inverse and simplifying

$$\begin{Bmatrix} U' \\ U'' \\ X' \\ Y' \end{Bmatrix} = \begin{bmatrix} 0 & I & 0 & 0 \\ -M^{-1}K & -M^{-1}B & M^{-1}q & M^{-1}kacH \\ P3-P1M^{-1}K & P2-P1M^{-1}B & -RO+P1M^{-1}q & P1M^{-1}kacH \\ -GDM^{-1}K & -GDM^{-1}B & GDM^{-1}q & A+GDM^{-1}k_a cH \end{bmatrix} \begin{Bmatrix} U \\ U' \\ X \\ Y \end{Bmatrix}$$

Because of the aileron mode, the $[M^{-1}K]$, $[M^{-1}B]$ and $[M^{-1}q]$ are no longer diagonal. Assuming that the aileron mode is placed last

$$M^{-1}K = \begin{bmatrix} k_1/m_1 & 0 & 0 & . & . & k_a*m_{1a}/m_1 \\ 0 & k_2/m_2 & 0 & . & . & k_a*m_{2a}/m_2 \\ 0 & 0 & k_3/m_3 & . & . & k_a*m_{3a}/m_3 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & . & k_a \end{bmatrix}$$

$$M^{-1}B = \begin{bmatrix} b_1/m_1 & 0 & 0 & . & . & b_a*m_{1a}/m_1 \\ 0 & b_2/m_2 & 0 & . & . & b_a*m_{2a}/m_2 \\ 0 & 0 & b_3/m_3 & . & . & b_a*m_{3a}/m_3 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & . & b_a \end{bmatrix}$$

$$M^{-1} q = \begin{bmatrix} q/m_1 & 0 & 0 & \cdot & \cdot & 0 & 0 \\ 0 & q/m_2 & 0 & \cdot & \cdot & 0 & 0 \\ 0 & 0 & q/m_3 & \cdot & \cdot & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & \cdot & \cdot & q/m_n & 0 \\ 0 & 0 & 0 & \cdot & \cdot & 0 & 0 \end{bmatrix}$$

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 doublet lattice portion of "NASTRAN" and used to calculate the aerodynamic influence coefficients (AICs) at fifteen reduced frequencies (see figure 4). The pade approximate curve fitting routine "QUEFIT" used the AICs 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

$$g(s) = (1256.6)^2 / (s^2 = 502.7s + (1256.6)^2)$$

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 continous 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 offand 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 neededto 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 ablility 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.

APPENDIX A

TABLES AND FIGURES

TABLE 1

NATURAL FREQUENCIES OF VIBRATION MODES PREDICTED BY NASTRAN AND MEASURED DURING GROUND VIBRATION TEST

Mode	frequency (hertz)	
	NASTRAN	GVT
First wing bending	9.1	9.6
First fuselage bending	16.5	16.2
Wing bending-torsion	29.6	29.1

TABLE 2

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

Mach Number	Frequency (hertz)	
	SAFSS	Flight Test
.75	13.24	12.25
.80	15.54	13.91

TABLE 3

VIBRATIONAL FREQUENCY OF FIRST WING BENDING MODE
PREDICTED BY SAFSS AND OBTAINED FROM FLIGHT TEST
FOR FSS ON AT 15000 FT.

Mach Number	Frequency (hertz)	
	SAFSS	Flight Test
.80	21.58	19.93
.825	21.71	20.

Wing Span: 14.5 ft.

Airfoil: Supercritical

Aspect Ratio: 6.8

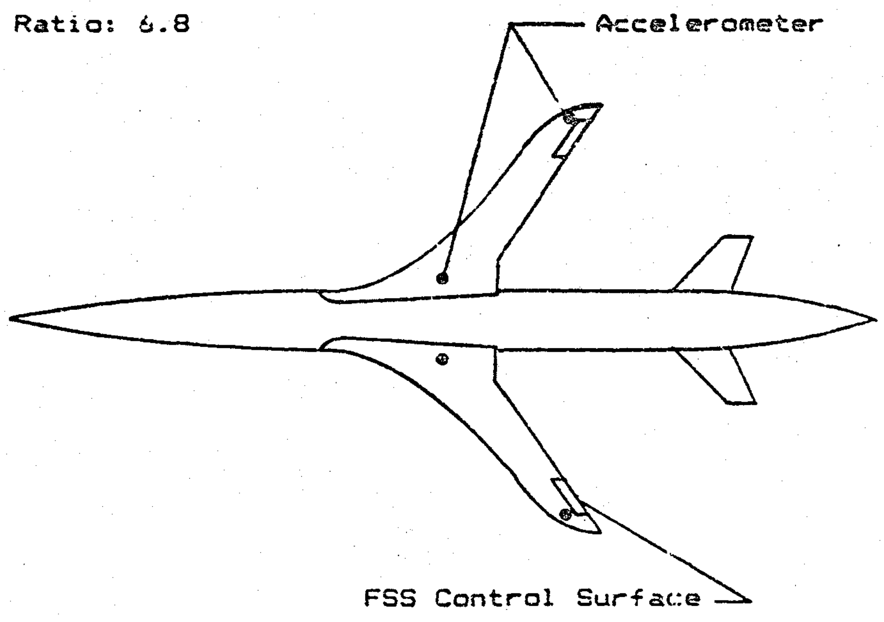

Figure 1. DAST ARW-1 planform

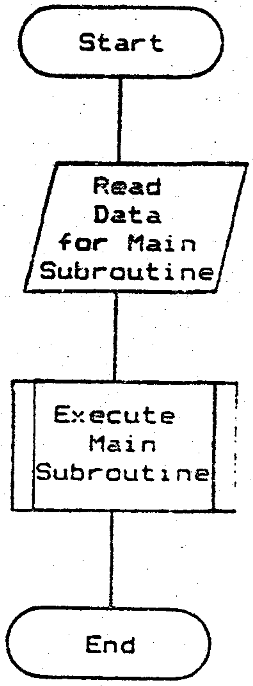

Figure 2. Flow Diagram of Program "SAFSS"

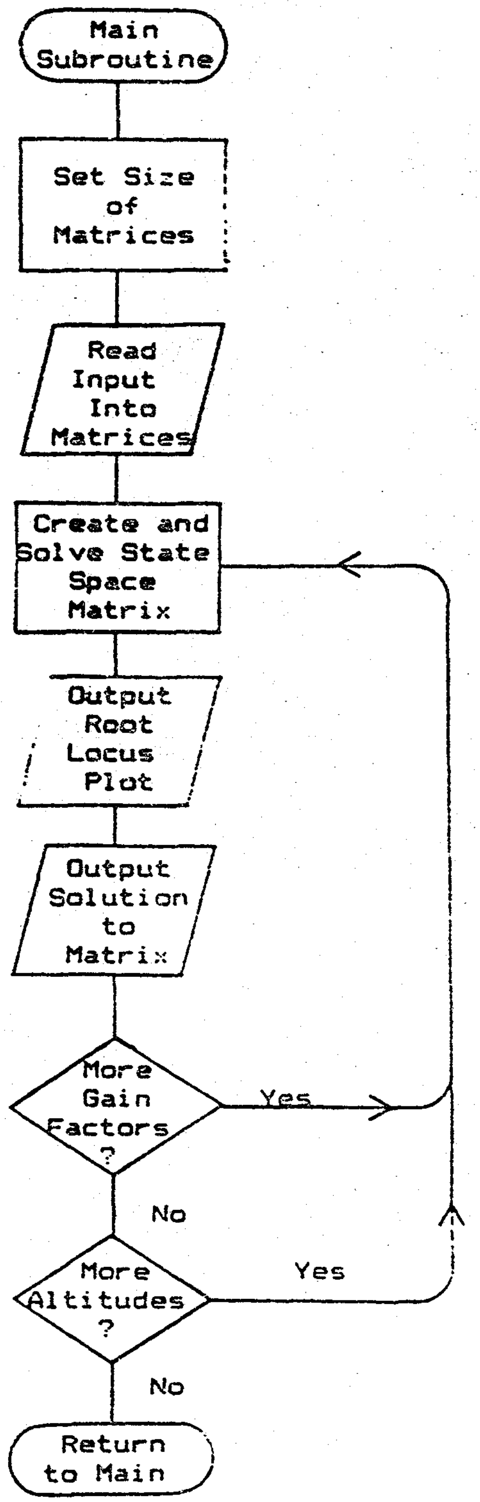

Figure 3. Flow Diagram of Main Subroutine of "SAFSS"

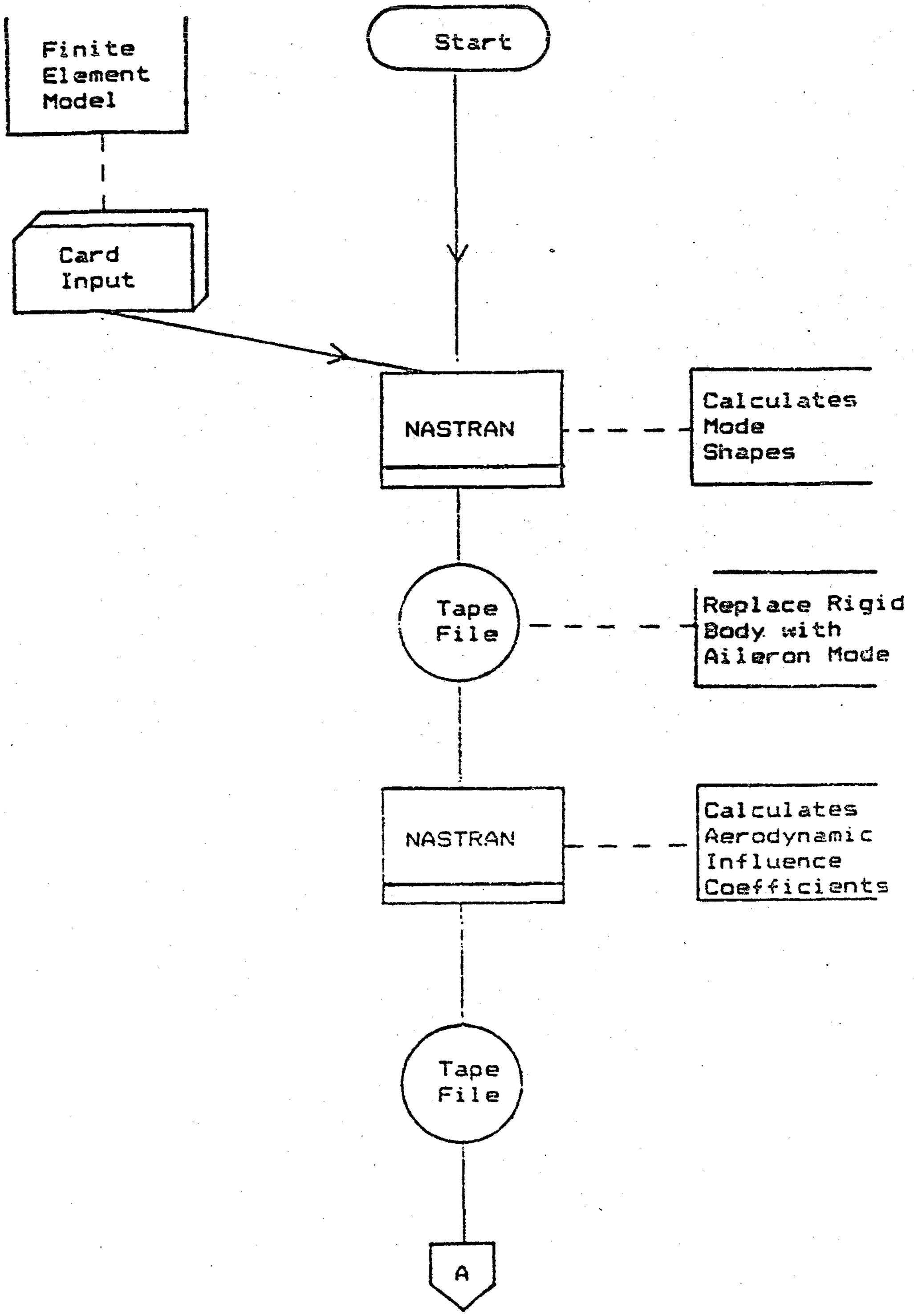

Figure 4. Block Diagram of Computer Programs Part 1

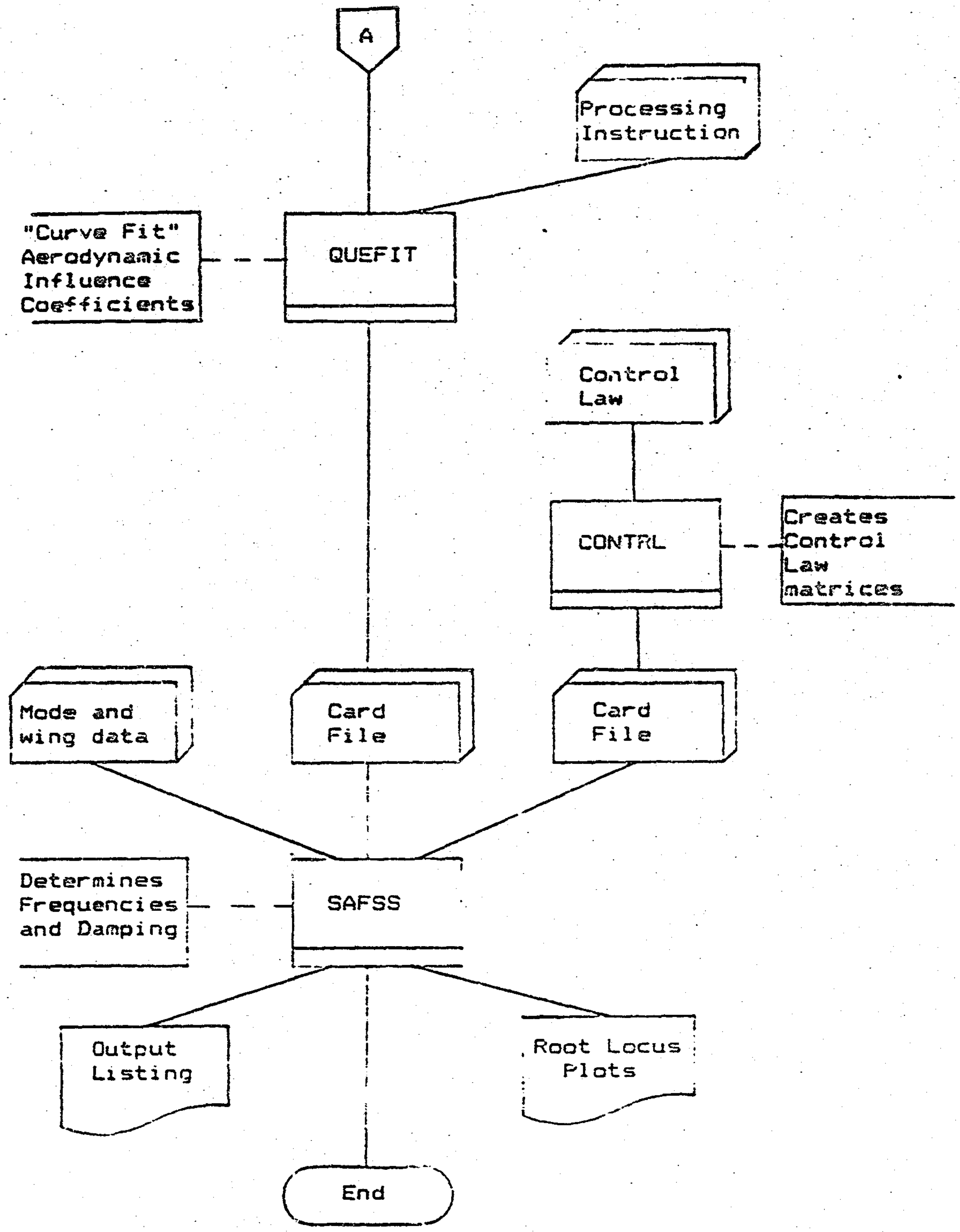

Figure 5. Block Diagram of Computer Programs Part 2

$$G(s) = \frac{(738)\,(2250)\; s\,\left(s^2 + 76.78s + (295.3)^2\right)\left(s^2 + 120s + (306)^2\right)\left(s^2 + 100s + (71)^2\right)}{(s+2)\,(s+295.3)^2\,(s+1500)^2\left(s^2 + 240s + (342)^2\right)\left(s^2 + 100s + (58)^2\right)\left(s^2 + 100s + (112)^2\right)}$$

$$\frac{\left(s^2 + 100s + (168)^2\right)(295.3)^2\,(1256.6)^2}{\left(s^2 + 76.78s + (295.3)^2\right)\left(s^2 + 589.4s + (439.8)^2\right)\left(s^2 + 502.7s + (1256.6)^2\right)}$$

Figure 6. Control Law for DAST ARW-1 Flight Test 3.

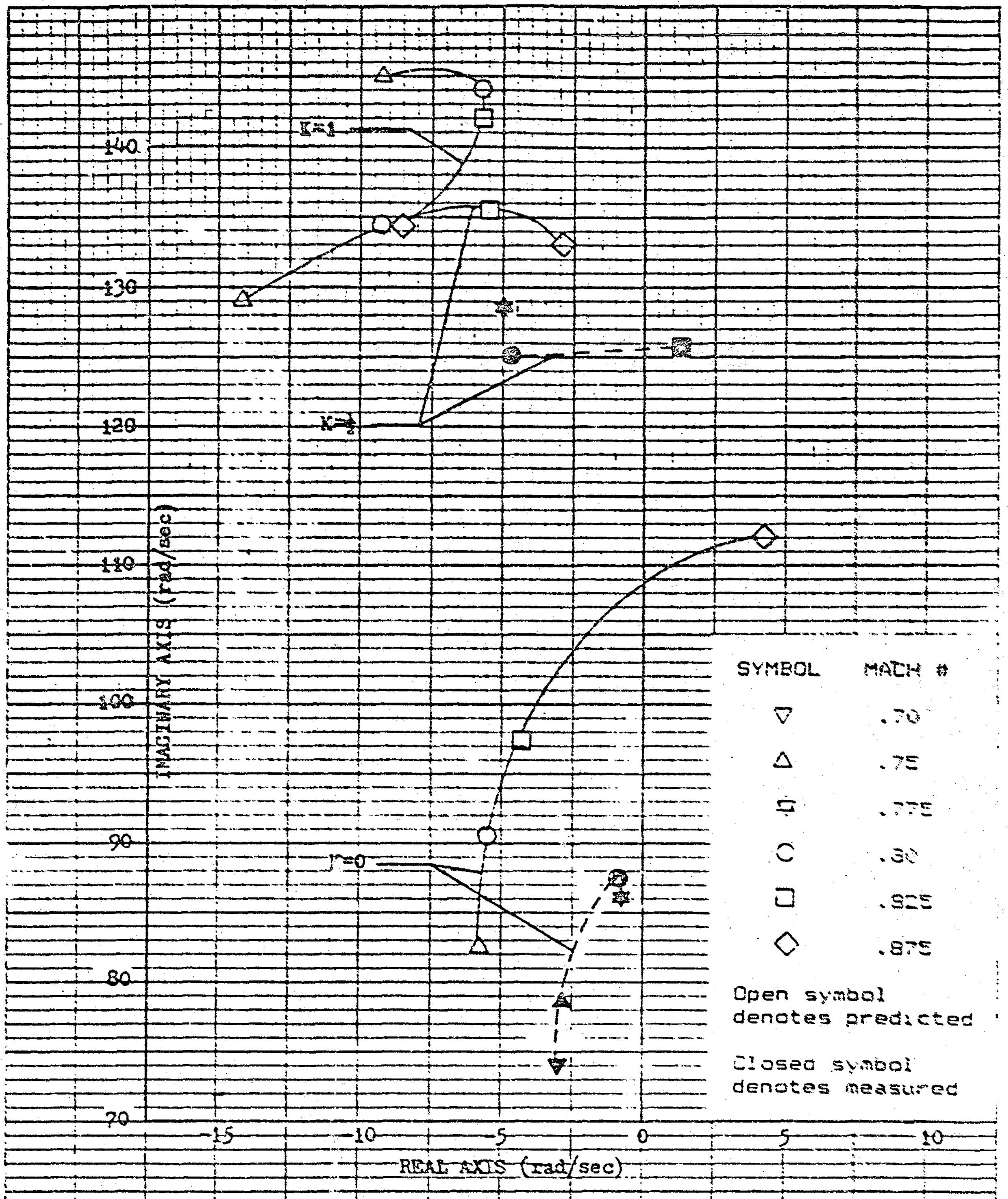

Figure 7. Root locus versus mach number for first wing bending mode at 15,000 ft. altitude.

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(6)   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:--

NCASE   1-5   I5

The number of different altitudes that calculations are to be performed at.

MODE   6-10   I5

The number of vibration modes being used for the calculations. Must include the aileron mode.

MALERN   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)

<u>NGAINS</u> 21-25 I5

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

<u>GAINUP</u> 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 .25.

<u>IEGVEC</u> 36-40 I5

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

<u>CLEN</u> 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.

<u>IPLT</u> 51-55 I5

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

<u>YMAX</u> 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 ofaltitudes 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 "Contrl" 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 I5

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:--

<u>HC(1.J)</u> 1-80 4E20.13

This set of cards contains the "H" matrix of the control law. It is a column matrix with Nx1 diminsions, 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:--

<u>MACH</u> 1-10 F10.6

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

CARD 9:--

<u>ISYM</u> 1-5 L5

This states whether the analysis is for a symmetric or assymetric 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:--

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

This set of cards contain the P1' matrix. This matrix is square of dimension MxM, 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:--

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

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

CARD SET 12:--

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

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

CARD SET 13:--

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

Same as card set 10 except that it is the RO' 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:--

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:---

<u>ZETA(I)</u>	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:---

<u>GMASS(I)</u>	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:---

<u>DEFLECT(I)</u>	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 (1g)/(32.2 ft/sec^2*12 in/ft). The output portion would be (1 degree)/(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.

PHI(I)   1-10   F10.6

The average deflection at leading edge of the aileron.

SIG(I)   21-22   F10.6

The average deflection at the trailing edge of the aileron.

APPENDIX C

LISTING OF PROGRAM "SAFSS"

```
*DECK SAFSS
      PROGRAM MAIN (INPUT,OUTPUT,TAPE1=INPUT,TAPE6=OUTPUT)
C**********************************************************************
C
C    PROGRAM SAFSS  (STABILITY ANALYSIS OF FLUTTER SUPPRESSION SYSTEM)
C
C**********************************************************************
C
C    THIS PROGRAM PUTS TOGETHER AND SOLVES THE MATRIX REPRESENTING THE
C    WING MOTION AND THE CONTOL LAW.
C
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 ACCELOROMETER 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
C**********************************************************************
C
C      DESCRIPTION OF INPUT DATA
C
C**********************************************************************
C
C      TITLE    = TITLE OF THE JOB
C      NCASE    = NUMBER OF ALTITUDES TO BE EVALUATED
C      MODE     = NUMBER OF VIBRATION MODE BEING USED
C      MALERN   = WHICH VIBRATION MODE IS THE AILERON DEFLECTION MODE
C      IEGVEC   = DO YOU WONT THE IEGEN VECTORS PRINTED (0=NO)
C      NGAINF   = NUMBER OF GAIN FACTORS TO BE SOLVED INCLUDING ZERO
C      NGAINT   = TYPE OF GAIN FACTOR PROGRESSION USED
C                 (1=LINEAR, 2=GEOMETRIC)
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
??
```

```
C      AC(I,J) = THE "A" MATRIX OF THE CONTROL LAW
C      GC(I,J) = THE "G" MATRIX OF THE CONTROL LAW
C      HC(I,1) = THE "H" MATRIX OF THE CONTROL LAW
C      P1(K,L) = THE S**2 TERM OF THE PADE APPROXIMANT OF THE LOAD
C      P2(K,L) = THE S TERM OF THE PADE APPROXIMANT OF THE LOAD
C      P3(K,L) = THE CONSTANT TERM OF THE PADE APPROXIMANT OF THE LOAD
C      RO(K,L) = THE EIGENVALUE TERM OF THE PADE APPROXIMANT TO THE LOAD
C      OMEGA(I)= FREQUENCIES OF THE CORRESPONDING VIBRATION MODE
C      ZETA(I) = DAMPING OF THE CORRESPONDING VIBRATION MODE
C      GMASS   = GENERALIZED MASS OF THE CORRESPONDING VIBRATION MODE
C      DEFLECT = DEFLECTION AT THE ACCELEROMETER FOR EACH MODE
C      XMD     = MASS OF THE AILERON
C      CONVFT  = CONVERSION FACTOR BETWEEN CONTROL LAW UNITS AND THE
C                UNITS USED WITH THIS PROGRAM.
C      PHI(I)  = AVERAGE DEFLECTION OF THE AILERON LEADING EDGE FOR
C                THE I'TH MODE
C      SIG(I)  = AVERAGE DEFLECTION OF THE AILERON TRAILING EDGE FOR
C                THE I'TH MODE
C
C*********************************************************************
      DIMENSION W1(11,11),W2(11,11),W3(11,11)
      DIMENSION P1(11,11),P2(11,11),P3(11,11),RO(11,11),VP3(11,11),
     1VRO(11,11),OMEGA2(11,11),QBM(11,11),B(11,11),VP1H2(11,11),
     2VP1(11,11),TITLE(8),OMEGA(11),ZETA(11),GMASS(11),A(48,48),WR(48),
     3WI(48),Z(11,11),Z3(48,48),INT(48),AC(15,15),AMTRX(33,33),
     4BMTRX(33,1),FMTRX(1,1),GMTRX(1,33),HMTRX(1,33),W4(33,33),
     5PHI(11),SIG(11),REACT(11),DEFLECT(11),ALT(20),
     .GC(15,1),HC(1,15),W5(1,11),W6(15,11),W7(11,15),W8(15,11),
     .W9(11,1),W10(1,15),W11(11,15),W12(15,15)
      READ 801,(TITLE(I),I=1,6)
      CALL DATE(TITLE(7))
      CALL TIME(TITLE(8))
      PRINT 901,TITLE
      READ 802, NCASE, MODE,MALERN,NGAINT,NGAINS,GAINUP,IEGVEC,CLEN,
     .      IPLT,YMAX
      READ 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,VRO,OMEGA2,QBM,B,VP1H2,VP1,OMEGA,ZETA,GMASS,
     .A,WR,WI,Z,Z3,IEGVEC,INT,AC,NCONTR,AMTRX,BMTRX,FMTRX,GMTRX,HMTRX,
     .W4,PHI,SIG,REACT,DEFLECT,ALT,GC,HC,W5,W6,W7,W8,W9,W10,W11,W12,
     .NGAINT,NGAINS,GAINUP,IPLT,YMAX)
  801 FORMAT(8A8)
  804 FORMAT(8F10.6)
  802 FORMAT(5I5,F10.6,I5,F10.6,I5,F10.6)
  901 FORMAT(1H1,10X,8A8)
      END
      SUBROUTINE PADJR(NCASE,MODF,MODE3,MSIZE,MALERN,CLEN,W1,W2,
     .W3,P1,P2,P3,RO,VP3,VRO,OMEGA2,QBM,B,VP1H2,VP1, OMEGA,ZETA,
     .GMASS,A,WR,WI,Z,Z3,IEGVEC,INT,AC,NCONTR,AMTRX,BMTRX,FMTRX,GMTRX,
     .HMTRX,W4,PHI,SIG,REACT,DEFLECT,ALT,GC,HC,W5,W6,W7,W8,W9,W10,
     .W11,W12,NGAINT,NGAINS,GAINUP,IPLT,YMAX)
C**********************************************************************
C
C     SUBROUTINE PADJR
C
C**********************************************************************
C
C     THIS PROGRAM RESIZES THE ARRAYS TO FIT THE PROBLEM THEN READS
C     THE INPUT INFORMATION AND THEN CREATS AND SOLVES THE STATE SPACE
C     MATRIX.
C
C**********************************************************************
      LOGICAL ISYM
      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),VRO(MODE,MODE),OMEGA2(MODE,MODE),QBM(MODE,MODE),
     .B(MODE,MODE),VP1H2(MODE,MODE),VP1(MODE,MODE),
```

```
     .OMEGA(MODE), ZETA(MODE),
     .GMASS(MODE), A(MSIZE,MSIZE),WR(MSIZE),WI(MSIZE),Z(MODE,MODE),
     .INT(MSIZE),Z3(MSIZE,MSIZE),ALT(NCASE),AC(NCONTR,NCONTR),
     .AMTRX(MODE3,MODE3),BMTRX(MODE3,1),FMTRX(1,1),GMTRX(1,MODE3),
     .HMTRX(1,MODE3),W4(MODE3,MODE3),GC(NCONTR,1),HC(1,NCONTR),
     .W5(1,MODE),W6(NCONTR,MODE),W7(MODE,NCONTR),W8(NCONTR,MODE),
     .W9(MODE,1),W10(1,NCONTR),W11(MODE,NCONTR),W12(NCONTR,NCONTR)
C*********************************************************************
C    READ MATRICES REPRESENTING CONTROL LAW  (FROM PROGRAM "CONTROL")
C*********************************************************************
      DO 11 I=1,NCONTR
   11 READ 801,(AC(I,J),J=1,NCONTR)
      READ 801,(GC(IR,1),IR=1,NCONTR)
      READ 801,(HC(1,IC),IC=1,NCONTR)
C*********************************************************************
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',RO'.
C*********************************************************************
      READ 802,MACH
      READ 803,ISYM
      DO 12 IR=1,MODE
      READ 801,(P1(IR,IC),IC=1,MODE)
   12 CONTINUE
      DO 13 IR=1,MODE
      READ 801,(P2(IR,IC),IC=1,MODE)
   13 CONTINUE
      DO 14 IR=1,MODE
      READ 801,(P3(IR,IC),IC=1,MODE)
   14 CONTINUE
      DO 15 IR=1,MODE
      READ 801,(RO(IR,IC),IC=1,MODE)
   15 CONTINUE
C*********************************************************************
C    READ NATURAL FREQUENCIES OF WING VIBRATION, MODAL DAMPING TERMS,
C    GENERALIZED MASS TERMS, AND THE DEFLECTION OF WING AT THE
```

```
C   ACCELEROMETER LOCATION.
C***********************************************************************
      READ 802,(OMEGA(I),I=1,MODE)
      READ 802,(ZETA(I),I=1,MODE)
      READ 802,(GMASS(I),I=1,MODE)
      READ 802,(DEFLECT(I),I=1,MODE)
C***********************************************************************
C   READ THE MASS OF THE AILERON, THE CONVERSION FACTOR BETWEEN
C   THE UNITS OF THE CONTROL LAW AND THE UNITS OF THIS ANALYSIS.
C***********************************************************************
      READ 802,XMD,CONVFT
C***********************************************************************
C   READ THE MODAL AVERAGE DEFLECTION AT THE LEADING EDGE OF
C   THE AILERON AND THE AVERAGE MODAL DEFLECTION AT THE TRAILING EDGE
C   OF THE AILERON FOR EACH MODE.
C***********************************************************************
      DO 16 I=1,MODE
   16 READ 802,PHI(I),SIG(I)
C***********************************************************************
C END OF INPUT REGION
C***********************************************************************
C   PRINT INFORMATION FOR LISTING
C***********************************************************************
      PRINT 904,MACH,MODE,NCONTR,CLEN
      PRINT 906,(ALT(I),I=1,NCASE)
      PRINT 901,(OMEGA(I),I=1,MODE)
      PRINT 902,(ZETA(I),I=1,MODE)
      PRINT 903,(GMASS(I),I=1,MODE)
      PRINT 908
      DO 17 IR=1,MODE
      PRINT 907,(P1(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=SIG(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.
C**************************     INITALIZE PLOTTER    *******************
      IF (IPLT.EQ.1) CALL INITPLT
C*********************************************************************
C   START OF LOOP FOR EACH ALTITUDE CALCULATIONS
C*********************************************************************
      DO 500 ICASE=1,NCASE
C******************      PLOT AXIS      ******************************
      IF (IPLT.EQ.1) CALL AXISPLT(ALT(NCASE),MACH,YMAX,YMIN,
     .XMAX,XMIN,SCALE,ISYM,TITLE)
C*********************************************************************
C   CALCULATE DYNAMIC PRESSURE AND AIR SPEED
C*********************************************************************
      CALL QBARC (MACH,ALT(ICASE),U,QBAR,INFLG)
      PRINT 921
      PRINT 922, MACH,ALT(ICASE),QBAR,U
C*********************************************************************
```

```
C   CALCULATE THE P3 AND RO MATRICES FORM THE P3' AND RO' MATRICES BY
C   MULTIPLYING BY THE VELOCITY OVER THE SEMI-SPAN. CALCULATE THE P1
C   MATRIX BY DIVIDING THE P1' MATRIX BY THE VELOCITY OVER THE
C   SEMI-SPAN.
C*********************************************************************
      UB=U/(CLEN/2.)
      CALL ARITH (UB,P3,0.,P3,VP3,MODE,MODE)
      CALL ARITH (UB,RO,0.,RO,VRO,MODE,MODE)
      UB=1./UB
      CALL ARITH (UB,P1,0.,P1,VP1,MODE,MODE)
C*********************************************************************
C   CONSTRUCT THE K/M, B/M AND Q/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.
      QBM(IR,IC)=0.
   30 CONTINUE
      OMEGA2(IR,IR)=OMEGA(IR)**2
      B(IR,IR)=2.*OMEGA(IR)*ZETA(IR)
      IF (IR.NE.MALERN) QBM(IR,IR)=QBAR/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*Q/M
C*********************************************************************
      CALL MULT(VP1,OMEGA2,W1,MODE)
      CALL ARITH(1.,VP3,-1.,W1,W1,MODE,MODE)
      CALL MULT(VP1,B,W2,MODE)
      CALL ARITH(1.,P2,-1.,W2,W2,MODE,MODE)
```

```
      CALL MULT(VP1,QBM,W3,MODE)
      CALL ARITH(-1.,VRO,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
   32 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)=QBM(IR,IC)
      A(IR,IC)=W3(IR,IC)
      A(IR,2*IC-1+MODE)=W1(IR,IC)
      A(IR,2*IC+MODE)=W2(IR,IC)
   33 CONTINUE
C*******************************************************************
C   ADD ADDITIONAL TERMS TO THE LARGE MATRIX THAT ARE CAUSED BY THE
C   ADDITION OF THE FLUTTER SUPPRESSION SYSTEM.
C*******************************************************************
      CALL FSS(A,OMEGA2,B,QBM,MODE,MSIZE,NCONTR,MALERN,
     .DEFLECT,REACT,GMASS,VP1,AC,GC,HC,W5,W6,W7,W8,W9,
     .W10,W11,W12,XK,CONVFT)
C*******************************************************************
C  END OF MATRIX CONSTUCTION, NOW GET THE EIGENVALUES
C*******************************************************************
      CALL ELMHES (MSIZE,MSIZE,1,MSIZE,A,INT)
      CALL ELTRAN (MSIZE,MSIZE,1,MSIZE,A,INT,Z3)
      CALL HQR2(MSIZE,MSIZE,1,MSIZE,A,WR,WI,Z3,IERR,0)
C*******  IF ZEROS WERE CALCULATED GO TO ZERO PLOTTER 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 (IEGVEC.EQ.0) GO TO 400
      PRINT 941
      L=1
  111 CONTINUE
      IF (MSIZE.LE.10*L) GO TO 110
```

```
      L=L+1
      GO TO 111
  110 CONTINUE
      LL=MSIZE-10*(L-1)
      LLL=1
  113 CONTINUE
      I=9
      IF (LLL.EQ.L)I=LL
      PRINT 968
      II=(LLL-1)*10+1
      III=II+I
      PRINT 942,((WR(J),WI(J)),J=II,III)
      PRINT 968
      DO 51 J=1,MSIZE
      PRINT 942,(Z3(J,K),K=II,III)
   51 CONTINUE
      LLL=LLL+1
      IF (LLL.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(L5)
  904 FORMAT(/,15X,"MACH -",F6.3,10X,I2," VIBRATION MODES"
     .,//10X,I2,"TH ORDER CONTROL LAW",10X,"CHARACTERISTIC LENGTH-",
     .F7.3)
  906 FORMAT(/10X,"ALTITUDES TO BE EVALUATED------",2(/10X(1X,G12.6)))
  901 FORMAT(/10X,"MODAL FREQUENCIES--------------",/10(1X,G12.6))
```

```
902 FORMAT(/10X,"MODAL DAMPING-------------------",/10(1X,G12.6))
903 FORMAT(/10X,"GENERALIZED MASSES--------------",/10(1X,G12.6))
908 FORMAT(/10X,"P1 MATRIX-----------------------"/)
907 FORMAT(10(1X,G12.6))
909 FORMAT(/10X,"P2 MATRIX-----------------------"/)
910 FORMAT(/10X,"P3 MATRIX-----------------------"/)
911 FORMAT(/10X,"RO MATRIX-----------------------"/)
912 FORMAT(10X," - END OF INPUT DATA _")
921 FORMAT(1H1,5X,10(" NEW CASE ")//)
922 FORMAT(10X,"MACH =",G16.6/10X,"ALT  =",G16.6/10X,
   ."QBAR =",G16.6/10X,"VTRUE=",G16.6)
931 FORMAT(//20X,"THE GAIN FACTOR =",F6.3,/)
932 FORMAT(/10X,"EIGENVALUES COMPUTED, ERROR CODE=",I4,//12X,
   ."REAL PART",8X,"IMAGINARY PART",2(9X,"FREQUENCY",3X),
   .4X,"DAMPING FACTOR",/5X,3(8X,"RAD/SEC",5X),8X,
   ."CYCLE/SEC",/)
933 FORMAT(5X,4(F15.2,5X),F15.4)
941 FORMAT(///10X,"THE EIGENVECTORS ARE",/)
968 FORMAT(//)
942 FORMAT(10E13.5)
951 FORMAT(///10X,"THE ZEROS ARE LOCATED AT:",/12X,"REAL PART",
   .8X,"IMAGINARY PART",/5X,2(8X,"RAD/SEC",5X),/)
952 FORMAT(2(5X,F15.2))
    RETURN
    END
    SUBROUTINE CROSS (A,B,C,I,J,K)
C***********************************************************************
C
C      SUBROUTINE CROSS
C
C***********************************************************************
C
C      THIS PROGRAM MULTIPLES 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
C     SUBROUTINE INITPLT   (INITIALIZE PLOTTER)
C
C*********************************************************************
C
C     THE ENTIRE 5 LINES OF THE PROGRAM IS SUPPOSED TO INITALIZE 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(.7874)
      RETURN
      END
      SUBROUTINE AXISPLT(ALT,MACH,YMAX,YMIN,XMAX,XMIN,SCALE,ISYM,TITLE)
C*********************************************************************
C
C     SUBROUTINE AXISPLT     (AXIS PLOT)
C
C*********************************************************************
```

```
C
C .   THIS PROGRAM DRAWS THE AXISES AND THE INFORMATION NECESSARY TO
C     DESCRIBE WHAT THE PLOT IS OF (MACH NUMBER, ALTITUDE, TITLE...).
C     OH, BY THE WAY THIS IS FOR A ROOT LOCUS PLOT.
C     THESE PROGRAM IS WRITEN WITH CALCOMP PLOTTER INSTRUCTIONS SO
C     TAKE IT OR WRITE YOUR OWN. IF DO NEED TO CHANGE IT THE OTHER
C     SUBROUTINES THAT ALSO USE PLOTTER INSTRUCTIONS ARE INITPLT,
C     POINT, ZEROPLT.
C
C*****************************************************************************
      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.,24HIMAGINARY AXIS (RAD/SEC),-24,11.,90.,0.,SCALE)
      CALL AXIS(-6.,0.,19HREAL AXIS (RAD/SEC),-19,8.,0.,XMIN,SCALE)
```

```
      CALL SYMBOL(-6.,-1.25,.10,12HGAIN FACTORS,0.,11)
      CALL SYMBOL(-6.,-1.5,.10,7HSYMBOLS,0.,7)
      CALL SYMBOL(-6.,12.,.14,11HALTITUDE = ,0.,11)
      CALL NUMBER(999.,999.,.14,ALT,0.,0)
      CALL SYMBOL(999.,999.,.14,12H     MACH = ,0.,12)
      CALL NUMBER(999.,999.,.14,MACH,0.,3)
      IF (ISYM) CALL SYMBOL(999.,999.,.14,14H     SYMMETRIC,0.,14)
      IF (.NOT.ISYM) CALL SYMBOL(999.,999.,.14,15H     ASYMMETRIC,0.,15)
      CALL SYMBOL(-6.,11.6,.10,TITLE,0.,80)
      RETURN
      END
      SUBROUTINE POINT(MSIZE,WR,WI,SCALE,KK,XK,YMAX,YMIN,XMAX,XMIN)
C*****************************************************************************
C
C     PROGRAM POINT   (DRAWS POINTS ON THE PLOT)
C
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 AWAYS 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*************************************************************************
C
C     THIS PROGRAMS 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
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,QBM,MODE,MSIZE,NCONTR,MALERN,
     .DEFLECT,REACT,GMASS,VP1,AC,GC,HC,W5,W6,W7,W8,W9,
     .W10,W11,W12,XK,CONVFT)
C***********************************************************************
C
C     PROGRAM FSS    (FLUTTER SURPRESSION SYSTEM)
C
C***********************************************************************
C
C     THIS PROGRAM ADDS THE ADDITIONAL TERMS TO THE MATRIX THAT A FLUTTER
C     SURPRESSION SYSTEM CAUSES SUCH AS THE CONTROL LAW AND THE REACTION
C     OF THE MODES TO THE MOTION OF THE AILERON RESPONDING TO THE CONTROL
C     LAW.
C
C***********************************************************************
      DIMENSION A(MSIZE,MSIZE),OMEGA2(MODE,MODE),B(MODE,MODE),
     .QBM(MODE,MODE),DEFLECT(MODE),REACT(MODE),GMASS(MODE),
     .VP1(MODE,MODE),AC(NCONTR,NCONTR),GC(NCONTR,1),HC(1,NCONTR),
     .W5(1,MODE),W6(NCONTR,MODE),W7(MODE,NCONTR),W8(NCONTR,MODE),
     .W9(MODE,1),W10(1,NCONTR),W11(MODE,NCONTR),W12(NCONTR,NCONTR)
      MODE3=MODE*3
      C=XK*OMEGA2(MALERN,MALERN)*CONVFT
      DO 11 L=1,MODE
      W5(1,L)=DEFLECT(L)
      W9(L,1)=REACT(L)/GMASS(L)
   11 CONTINUE
      DO 12 K=1,NCONTR
      W10(1,K)=C*HC(1,K)
   12 CONTINUE
C*********    CALCULATE THE G*D*Q/M MATRIX    *********************
      CALL CROSS(GC,W5,W6,NCONTR,1,MODE)
      CALL CROSS(W6,QBM,W8,NCONTR,MODE,MODE)
```

```
      DO 21 IR=1,NCONTR
      DO 21 IC=1,MODE
      A(IR+MODE3,IC)=W8(IR,IC)
   21 CONTINUE
C**********    CALCULATE THE -G*D*K/M MATRIX     ******************
      CALL CROSS(W6,OMEGA2,W8,NCONTR,MODE,MODE)
      DO 22 IR=1,NCONTR
      DO 22 IC=1,MODE
      A(IR+MODE3,2*IC-1+MODE)=-W8(IR,IC)
   22 CONTINUE
C**********     CALCULATE THE -G*D*B/M MATRIX     ****************
      CALL CROSS(W6,B,W8,NCONTR,MODE,MODE)
      DO 23 IR=1,NCONTR
      DO 23 IC=1,MODE
      A(IR+MODE3,2*IC+MODE)=-W8(IR,IC)
   23 CONTINUE
C**************    CLACULATE THE C*H/M MATRIX     ********************
      CALL CROSS(W9,W10,W11,MODE,1,NCONTR)
      DO 24 IR=1,MODE
      DO 24 IC=1,NCONTR
      A(2*IR+MODE,IC+MODE3)=W11(IR,IC)
   24 CONTINUE
C**************     CALCULATE THE A + G*D*C*H/M MATRIX    ************
      CALL CROSS(W6,W11,W12,NCONTR,MODE,NCONTR)
      DO 25 IR=1,NCONTR
      DO 25 IC=1,NCONTR
      A(IR+MODE3,IC+MODE3)=AC(IR,IC)+W12(IR,IC)
   25 CONTINUE
C***************     CALCULATE THE PI*C*H/M MATRIX     ***************
      CALL CROSS(VP1,W11,W7,MODE,MODE,NCONTR)
      DO 26 IR=1,MODE
      DO 26 IC=1,NCONTR
      A(IR,IC+MODE3)=W7(IR,IC)
   26 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE ARITH (SA,A,SB,B,C,NR,NC)
C*****************************************************************************
C
C     PROGRAM ARITH     (ARTIHMETIC)
C
C*****************************************************************************
C
C     THIS PROGRAM WILL MULTIPLE A MATRIX BY A CONSTANT AND ADD IT TO SECOND
C     MATRIX OF THE SAME SIZE AFTER THE SECOND MATRIX HAS BEEN MULTIPLED 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*****************************************************************************
      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 QBARC (MACH,ALT,U,QBAR,INFLG)
C*****************************************************************************
C
C     PROGRAM QBARC
C
C*****************************************************************************
C
C     THIS PROGRAM TAKES THE MACH NUMBER AND ALTITUDE AND ESTIMATES THE
```

```
C     AIRSPEED IN FT/SEC AND THE DYNAMIC PRESSURE.
C
C*************************************************************************
      REAL MACH
      DIMENSION ALTT(10), A(10), RHO(10)
      DIMENSION BKMACH(10),CLCORR(10)
      DATA ALTT/0.0,5000.,10000.,15000.,20000.,25000.,30000.,
     .35000.,40000.,50000./
      DATA  A/1116.45,1097.09,1077.40,1057.35,1036.92,1016.10
     .994.85,973.14,968.08,968.08/
      DATA  RHO/.0023769,.0020482,.0017556, .0014962,.0012673,.0010663,
     ..00089069,.00073281,.00058728,.00036392/
      DATA BKMACH/.70,.725,.75,.775,.80,.825,.85,.875,.90,.925/
      DATA CLCORR/1.025,1.026,1.027,1.03,1.05,1.07,1.08,1.1,1.125,1.155/
      IF (ALT.GT.0.0) GO TO 102
      AI=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*)
      AI=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))
      AI=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)
      QBAR=QBAR*CORFAC
      IF (INFLG.EQ.0) RETURN
      U=U*12
      QBAR=QBAR/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
   11 XX=XX+A(J,L)*B(L,K)
   10 C(J,K)=XX
      RETURN
      END
      SUBROUTINE ELMHES(NM,N,LOW,IGH,A,INT)                             423.
C                                                                       424.
      INTEGER I,J,M,N,LA,NM,IGH,KP1,LOW,MM1,MP1                         425.
      REAL    A(NM,N)
      REAL    X,Y
      INTEGER INT(IGH)                                                  429.
C                                                                       430.
C     THIS SUBROUTINE IS A TRANSLATION OF THE ALGOL PROCEDURE ELMHES,   431.
C     NUM. MATH. 12, 349-368(1968) BY MARTIN AND WILKINSON.             432.
C     APPLIED MATHEMATICS DIVISION, ARGONNE NATIONAL LABORATORY         465.
C     HANDBOOK FOR AUTO. COMP., VOL.II-LINEAR ALGEBRA, 339-358(1971).   433.
C                                                                       434.
C     GIVEN A REAL GENERAL MATRIX, THIS SUBROUTINE                      435.
C     REDUCES A SUBMATRIX SITUATED IN ROWS AND COLUMNS                  436.
C     LOW THROUGH IGH TO UPPER HESSENBERG FORM BY                       437.
```

```
C     STABILIZED ELEMENTARY SIMILARITY TRANSFORMATIONS.
C
C     ON INPUT:
C
C        NM MUST BE SET TO THE ROW DIMENSION OF TWO-DIMENSIONAL
C          ARRAY PARAMETERS AS DECLARED IN THE CALLING PROGRAM
C          DIMENSION STATEMENT;
C
C        N IS THE ORDER OF THE MATRIX;
C
C        LOW AND IGH ARE INTEGERS DETERMINED BY THE BALANCING
C          SUBROUTINE  BALANC.  IF  BALANC  HAS NOT BEEN USED,
C          SET LOW=1, IGH=N;
C
C        A CONTAINS THE INPUT MATRIX.
C
C     ON OUTPUT:
C
C        A CONTAINS THE HESSENBERG MATRIX.  THE MULTIPLIERS
C          WHICH WERE USED IN THE REDUCTION ARE STORED IN THE
C          REMAINING TRIANGLE UNDER THE HESSENBERG MATRIX;
C
C        INT CONTAINS INFORMATION ON THE ROWS AND COLUMNS
C          INTERCHANGED IN THE REDUCTION.
C          ONLY ELEMENTS LOW THROUGH IGH ARE USED.
C
C     QUESTIONS AND COMMENTS SHOULD BE DIRECTED TO B. S. GARBOW,
C
C     ------------------------------------------------------------------
C
      LA = IGH - 1
      KP1 = LOW + 1
      IF (LA .LT. KP1) GO TO 200
C
      DO 180 M = KP1, LA
         MM1 = M - 1
```

```
      X =  0.0E0
      I = M
C
      DO 100 J = M, IGH
         IF ( ABS(A(J,MM1)) .LE.  ABS(X)) GO TO 100
         X = A(J,MM1)
         I = J
  100 CONTINUE
C
      INT(M) = I
      IF (I .EQ. M) GO TO 130
C
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, IGH
         Y = A(J,I)
         A(J,I) = A(J,M)
         A(J,M) = Y
  120 CONTINUE
C
C     END INTERCHANGE
  130 IF (X .EQ. 0.0E0) GO TO 180
      MP1 = M + 1
C
      DO 160 I = MP1, IGH
         Y = A(I,MM1)
         IF (Y .EQ. 0.0E0) GO TO 160
         Y = Y / X
         A(I,MM1) = Y
C
         DO 140 J = M, N
```

```
  140       A(I,J) = A(I,J) - Y * A(M,J)
C .
            DO 150 J = 1, IGH
  150       A(J,M) = A(J,M) + Y * A(J,I)
C
  160    CONTINUE
C
  180 CONTINUE
C
  200 RETURN
C
      END
      SUBROUTINE ELTRAN(NM,N,LOW,IGH,A,INT,Z)
C
      INTEGER I,J,N,KL,MM,MP,NM,IGH,LOW,MP1
      REAL   A(NM,IGH),Z(NM,N)
      INTEGER INT(IGH)
C
C     THIS SUBROUTINE IS A TRANSLATION OF THE ALGOL PROCEDURE ELMTRANS,
C     NUM. MATH. 16, 181-204(1970) BY PETERS AND WILKINSON.
C     HANDBOOK FOR AUTO. COMP., VOL.II-LINEAR ALGEBRA, 372-395(1971).
C
C     THIS SUBROUTINE ACCUMULATES THE STABILIZED ELEMENTARY
C     SIMILARITY TRANSFORMATIONS USED IN THE REDUCTION OF A
C     REAL GENERAL MATRIX TO UPPER HESSENBERG FORM BY  ELMHES.
C
C     ON INPUT:
C
C        NM MUST BE SET TO THE ROW DIMENSION OF TWO-DIMENSIONAL
C          ARRAY PARAMETERS AS DECLARED IN THE CALLING PROGRAM
C          DIMENSION STATEMENT;
C
C        N IS THE ORDER OF THE MATRIX;
C
C        LOW AND IGH ARE INTEGERS DETERMINED BY THE BALANCING
C          SUBROUTINE  BALANC.  IF  BALANC  HAS NOT BEEN USED,
```

```
C          SET LOW=1, IGH=N;
C
C        A CONTAINS THE MULTIPLIERS WHICH WERE USED IN THE
C          REDUCTION BY  ELMHES  IN ITS LOWER TRIANGLE
C          BELOW THE SUBDIAGONAL;
C
C        INT CONTAINS INFORMATION ON THE ROWS AND COLUMNS
C          INTERCHANGED IN THE REDUCTION BY  ELMHES.
C          ONLY ELEMENTS LOW THROUGH IGH ARE USED.
C
C     ON OUTPUT:
C
C        Z CONTAINS THE TRANSFORMATION MATRIX PRODUCED IN THE
C          REDUCTION BY  ELMHES.
C
C     QUESTIONS AND COMMENTS SHOULD BE DIRECTED TO B. S. GARBOW,
C     APPLIED MATHEMATICS DIVISION, ARGONNE NATIONAL LABORATORY
C
C     ------------------------------------------------------------------
C
C
C      INITIALIZE Z TO IDENTITY MATRIX
      DO 80 I = 1, N
C
         DO 60 J = 1, N
   60    Z(I,J) = 0.0E0
C
         Z(I,I) = 1.0E0
   80 CONTINUE
C
      KL = IGH - LOW - 1
      IF (KL .LT. 1) GO TO 200
C      FOR MP=IGH-1 STEP -1 UNTIL LOW+1 DO --
C
      DO 140 MM = 1, KL
         MP = IGH - MM
```

```
          MP1 = MP + 1
C
          DO 100 I = MP1, IGH
  100     Z(I,MP) = A(I,MP-1)
C
          I = INT(MP)
          IF (I .EQ. MP) GO TO 140
C
          DO 130 J = MP, IGH
             Z(MP,J) = Z(I,J)
             Z(I,J) = 0.0E0
  130     CONTINUE
C
          Z(I,MP) = 1.0E0
  140 CONTINUE
C
  200 RETURN
C
      END
      SUBROUTINE HQR2(NM,N,LOW,IGH,H,WR,WI,Z,IERR,INUM)
C
      INTEGER I,J,K,L,M,N,EN,II,JJ,LL,MM,NA,NM,NN,
     X        IGH,ITS,LOW,MP2,ENM2,IERR
      REAL   H(NM,N),WR(N),WI(N),Z(NM,N)
      REAL   P,Q,R,S,T,W,X,Y,RA,SA,VI,VR,ZZ,NORM,MACHEP
      INTEGER MINO
      LOGICAL NOTLAS
      COMPLEX    Z3
      COMPLEX     CMPLX
      REAL   T3(2)
      EQUIVALENCE (Z3,T3(1))
C
C     THIS SUBROUTINE IS A TRANSLATION OF THE ALGOL PROCEDURE HQR2,
C     NUM. MATH. 16, 181-204(1970) BY PETERS AND WILKINSON.
C     HANDBOOK FOR AUTO. COMP., VOL.II-LINEAR ALGEBRA, 372-395(1971).
C
```

```
C     THIS SUBROUTINE FINDS THE EIGENVALUES AND EIGENVECTORS
C     OF A REAL UPPER HESSENBERG MATRIX BY THE QR METHOD.  THE
C     EIGENVECTORS OF A REAL GENERAL MATRIX CAN ALSO BE FOUND
C     IF  ELMHES  AND  ELTRAN  OR  ORTHES  AND  ORTRAN  HAVE
C     BEEN USED TO REDUCE THIS GENERAL MATRIX TO HESSENBERG FORM
C     AND TO ACCUMULATE THE SIMILARITY TRANSFORMATIONS.
C
C     ON INPUT:
C
C        NM MUST BE SET TO THE ROW DIMENSION OF TWO-DIMENSIONAL
C          ARRAY PARAMETERS AS DECLARED IN THE CALLING PROGRAM
C          DIMENSION STATEMENT;
C
C        N IS THE ORDER OF THE MATRIX;
C
C        LOW AND IGH ARE INTEGERS DETERMINED BY THE BALANCING
C          SUBROUTINE  BALANC.  IF  BALANC  HAS NOT BEEN USED,
C          SET LOW=1, IGH=N;
C
C        H CONTAINS THE UPPER HESSENBERG MATRIX;
C
C        Z CONTAINS THE TRANSFORMATION MATRIX PRODUCED BY  ELTRAN
C          AFTER THE REDUCTION BY  ELMHES, OR BY  ORTRAN  AFTER THE
C          REDUCTION BY  ORTHES, IF PERFORMED.  IF THE EIGENVECTORS
C          OF THE HESSENBERG MATRIX ARE DESIRED, Z MUST CONTAIN THE
C          IDENTITY MATRIX.
C
C     ON OUTPUT:
C
C        H HAS BEEN DESTROYED;
C
C        WR AND WI CONTAIN THE REAL AND IMAGINARY PARTS,
C          RESPECTIVELY, OF THE EIGENVALUES.  THE EIGENVALUES
C          ARE UNORDERED EXCEPT THAT COMPLEX CONJUGATE PAIRS
C          OF VALUES APPEAR CONSECUTIVELY WITH THE EIGENVALUE
C          HAVING THE POSITIVE IMAGINARY PART FIRST.  IF AN
```

```
C           ERROR EXIT IS MADE, THE EIGENVALUES SHOULD BE CORRECT
C           FOR INDICES IERR+1,...,N;
C
C         Z CONTAINS THE REAL AND IMAGINARY PARTS OF THE EIGENVECTORS.
C           IF THE I-TH EIGENVALUE IS REAL, THE I-TH COLUMN OF Z
C           CONTAINS ITS EIGENVECTOR.  IF THE I-TH EIGENVALUE IS COMPLEX
C           WITH POSITIVE IMAGINARY PART, THE I-TH AND (I+1)-TH
C           COLUMNS OF Z CONTAIN THE REAL AND IMAGINARY PARTS OF ITS
C           EIGENVECTOR.  THE EIGENVECTORS ARE UNNORMALIZED.  IF AN
C           ERROR EXIT IS MADE, NONE OF THE EIGENVECTORS HAS BEEN FOUND;
C
C         IERR IS SET TO
C           ZERO       FOR NORMAL RETURN,
C           J          IF THE J-TH EIGENVALUE HAS NOT BEEN
C                      DETERMINED AFTER 30 ITERATIONS.
C
C      ARITHMETIC IS REAL EXCEPT FOR THE REPLACEMENT OF THE ALGOL
C      PROCEDURE CDIV BY COMPLEX DIVISION.
C
C      QUESTIONS AND COMMENTS SHOULD BE DIRECTED TO B. S. GARBOW,
C      APPLIED MATHEMATICS DIVISION, ARGONNE NATIONAL LABORATORY
C
C      ------------------------------------------------------------------
C
C       MACHEP IS A MACHINE DEPENDENT PARAMETER SPECIFYING
C                THE RELATIVE PRECISION OF FLOATING POINT ARITHMETIC.
C                MACHEP = 16.0E0**(-13) FOR LONG FORM ARITHMETIC
C                ON S360
       DATA MACHEP/1.E-9/
C
      DO 5 K=1,IGH
      WR(K)=0.
      WI(K)=0.
    5 CONTINUE
      IERR = 0
C
```

```
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
      EN = IGH
      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
```

```
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
C       LOOK FOR TWO CONSECUTIVE SMALL
C                 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 = (R * 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 = Q / 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)
     X     * ( ABS(H(M-1,M-1)) +  ABS(ZZ) +  ABS(H(M+1,M+1)))) GO TO 150
  140 CONTINUE
  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
C       DOUBLE OR STEP INVOLVING ROWS L TO EN AND,
```

```
C                    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+1,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 = Q / X
         R = R / X
  170    S =  SIGN( SQRT(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)
  190    P = P + S
         X = P / S
         Y = Q / S
         ZZ = R / S
         Q = Q / P
         R = R / P
C
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 * ZZ
  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
C      COLUMN MODIFICATION
          DO 230 I = 1, J
             P = X * H(I,K) + Y * H(I,K+1)
             IF (.NOT. NOTLAS) GO TO 220
             P = P + ZZ * H(I,K+2)
             H(I,K+2) = H(I,K+2) - P * R
  220        H(I,K+1) = H(I,K+1) - P * Q
             H(I,K) = H(I,K) - P
  230     CONTINUE
C
C      ACCUMULATE TRANSFORMATIONS
          DO 250 I = LOW, IGH
             P = X * Z(I,K) + Y * Z(I,K+1)
             IF (.NOT. NOTLAS) GO TO 240
             P = P + ZZ * Z(I,K+2)
             Z(I,K+2) = Z(I,K+2) - P * R
  240        Z(I,K+1) = Z(I,K+1) - P * Q
             Z(I,K) = Z(I,K) - P
  250     CONTINUE
C
  260 CONTINUE
C
      GO TO 70
C
C      ONE ROOT FOUND
  270 H(EN,EN) = X + T
      WR(EN) = H(EN,EN)
      WI(EN) = 0.0E0
      EN = NA
      GO TO 60
C
C      TWO ROOTS FOUND
  280 P = (Y - X) / 2.0E0
      Q = P * P + W
      ZZ = SQRT( ABS(Q))
```

```
      H(EN,EN) - X + T
      X = H(EN,EN)
      H(NA,NA) = Y + T
      IF (Q .LT. 0.0E0) GO TO 320
C
C      REAL PAIR'
      ZZ = P +  SIGN(ZZ,P)
      WR(NA) - X + ZZ
      WR(EN) = WR(NA)
      IF (ZZ .NE. 0.0E0) WR(EN) = X   W / ZZ
      WI(NA) = 0.0E0
      WI(EN) = 0.0E0
      X = H(EN,NA)
      R =  SQRT(X*X+ZZ*ZZ)
      P = X / R
      Q = ZZ / R
C
C      ROW MODIFICATION
      DO 290 J - NA, N
         ZZ = H(NA,J)
         H(NA,J) = Q * ZZ + P * H(EN,J)
         H(EN,J) - Q * H(EN,J) - P * ZZ
  290 CONTINUE
C
C      COLUMN MODIFICATION
      DO 300 I = 1, EN
         ZZ = H(I,NA)
         H(I,NA) = Q * ZZ + P * H(I,EN)
         H(I,EN) - Q * H(I,EN) - P * ZZ
  300 CONTINUE
C
C      ACCUMULATE TRANSFORMATIONS
      DO 310 I = LOW, IGH
         ZZ = Z(I,NA)
         Z(I,NA) - Q * ZZ + P * Z(I,EN)
         Z(I,EN) - Q * Z(I,EN) - P * ZZ
```

```
  310 CONTINUE
C .
      GO TO 330
C
C      COMPLEX PAIR
  320 WR(NA) = X + P
      WR(EN) = X + P
      WI(NA) = ZZ
      WI(EN) = -ZZ
  330 EN = ENM2
      GO TO 60
C
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 350 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
C      FOR EN=N STEP -1 UNTIL 1 DO --
      DO 800 NN = 1, N
         EN = N + 1 - NN
         P = WR(EN)
         Q = WI(EN)
         NA = EN - 1
         IF (Q) 710, 600, 800
C
```

```
  600    M = EN
         H(EN,EN) = 1.0E0
         IF (NA .EQ. 0) GO TO 800
C
C       FOR I=EN-1 STEP -1 UNTIL 1 DO --
         DO 700 II = 1, NA
            I = EN - II
            W = H(I,I) - P
            R = H(I,EN)
            IF (M .GT. NA) GO TO 620
C
            DO 610 J = M, NA
  610       R = R + H(I,J) * H(J,EN)
C
  620       IF (WI(I) .GE. 0.0E0) GO TO 630
            ZZ = W
            S = R
            GO TO 700
  630       M = I
            IF (WI(I) .NE. 0.0E0) GO TO 640
            T = W
            IF (W .EQ. 0.0E0) T = MACHEP * NORM
            H(I,EN) = -R / T
            GO TO 700
C
C       SOLVE REAL EQUATIONS
  640       X = H(I,I+1)
            Y = H(I+1,I)
            Q = (WR(I) - P) * (WR(I) - P) + WI(I) * WI(I)
            T = (X * S - ZZ * R) / Q
            H(I,EN) = T
            IF ( ABS(X) .LE. ABS(ZZ)) GO TO 650
            H(I+1,EN) = (-R - W * T) / X
            GO TO 700
  650       H(I+1,EN) = (-S - Y * T) / ZZ
  700    CONTINUE
```

```
C
C     END REAL VECTOR
        GO TO 800
C
C     COMPLEX VECTOR
  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) = Q / H(EN,NA)
        H(NA,EN) = -(H(EN,EN) - P) / H(EN,NA)
        GO TO 730
  720   Z3 =  CMPLX(0.0E0,-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
        DO 790 II = 1, ENM2
           I = NA - II
           W = H(I,I) - P
           RA = 0.0E0
           SA = H(I,EN)
C
           DO 760 J = M, NA
              RA = RA + H(I,J) * H(J,NA)
              SA = SA + H(I,J) * H(J,EN)
  760      CONTINUE
C
           IF (WI(I) .GE. 0.0E0) GO TO 770
           ZZ = W
           R = RA
           S = SA
```

```
            GO TO 790
  770       M = I
            IF (WI(I) .NE. 0.0E0) GO TO 780
            Z3 =  CMPLX(-RA,-SA) /  CMPLX(W,Q)
            H(I,NA) = T3(1)
            H(I,EN) = T3(2)
            GO TO 790
C
C     SOLVE COMPLEX EQUATIONS
  780       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       * ( ABS(W) +  ABS(Q) +  ABS(X) +  ABS(Y) +  ABS(ZZ))
            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) +  ABS(Q)) GO TO 785
            H(I+1,NA) = (-RA - W * H(I,NA) + Q * H(I,EN)) / X
            H(I+1,EN) = (-SA - W * H(I,EN) - Q * H(I,NA)) / X
            GO TO 790
  785       Z3 =  CMPLX(-R-Y*H(I,NA),-S-Y*H(I,EN)) /  CMPLX(ZZ,Q)
            H(I+1,NA) = T3(1)
            H(I+1,EN) = T3(2)
  790    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
C
         DO 820 J = I, N
```

```
  820     Z(I,J) = H(I,J)
C
  840 CONTINUE
C
C      MULTIPLY BY TRANSFORMATION MATRIX TO GIVE
C               'VECTORS OF ORIGINAL FULL MATRIX.
C               FOR J=N STEP -1 UNTIL LOW DO -- ::::::::::
      DO 880 JJ = LOW, N
         J = N + LOW - JJ
         M = MIN0(J,IGH)
C
         DO 880 I = LOW, IGH
            ZZ = 0.0E0
C
            DO 860 K = LOW, M
  860       ZZ = ZZ + Z(I,K) * H(K,J)
C
            Z(I,J) = ZZ
  880 CONTINUE
C
      GO TO 1001
C
C      SET ERROR -- NO CONVERGENCE TO AN
C                EIGENVALUE AFTER 30 ITERATIONS
 1000 IERR = EN
 1001 RETURN
C
C      LAST CARD OF HQR2
      END
END OF FILE
```

APPENDIX D

SAMPLE INPUT FILE FOR "SAFSS"

```
DAST ARW-1 OLD DATA SYMETRIC ROOT LOCUS
    1   11    1    2    5    .25       0    97.8       0     270.
15000.
   15
-.200000000000E+01  0.                   0.                   0.
 0.                 0.                   0.                   0.
 0.                 0.                   0.                   0.
 0.                 0.                   0.
 0.                 0.                   .100000000000E+01    0.
 0.                 0.                   0.                   0.
 0.                 0.                   0.                   0.
 0.                 0.                   0.
 .100000000000E+01  -.872020900000E+05   -.590600000000E+03   0.
 0.                 0.                   0.                   0.
 0.                 0.                   0.                   0.
 0.                 0.                   0.
 0.                 0.                   0.                   0.
 .100000000000E+01  0.                   0.                   0.
 0.                 0.                   0.                   0.
 0.                 0.                   0.                   0.
 0.                 0.                   0.
 0.                 0.                   .225000000000E+04    -.116964000000E+06
-.240000000000E+03  0.                   0.                   0.
 0.                 0.                   0.                   0.
 0.                 0.                   0.
 0.                 0.                   0.                   0.
 0.                 0.                   .100000000000E+01    0.
 0.                 0.                   0.                   0.
 0.                 0.                   0.
 0.                 0.                   .225000000000E+04    -.297619100000E+05
-.163220000000E+03  -.225000000000E+07   -.300000000000E+04   0.
 0.                 0.                   0.                   0.
 0.                 0.                   0.
 0.                 0.                   0.                   0.
 0.                 0.                   0.                   0.
 .100000000000E+01  0.                   0.                   0.
 0.                 0.                   0.
```

0.	0.	.225000000000E+07	-.297619100000E+08
-.163220000000E+06	-.215636400000E+10	-.288000000000E+07	-.336400000000E+04
-.100000000000E+03	0.	0.	0.
0.	0.	0.	
0.	0.	0.	0.
0.	0.	0.	0.
0.	0.	.100000000000E+01	0.
0.	0.	0.	
0.	0.	.166050000000E+07	-.219642895800E+08
-.120456360000E+06	-.159139663200E+10	-.212544000000E+07	.123762600000E+04
0.	-.125440000000E+05	-.100000000000E+03	0.
0.	0.	0.	
0.	0.	0.	0.
0.	0.	0.	0.
0.	0.	0.	0.
.100000000000E+01	0.	0.	
0.	0.	.166050000000E+07	-.219642895800E+08
-.120456360000E+06	-.159139663200E+10	-.212544000000E+07	.123762600000E+04
0.	.156800000000E+05	0.	-.872020900000E+05
-.767800000000E+02	0.	0.	
0.	0.	0.	0.
0.	0.	0.	0.
0.	0.	0.	0.
0.	0.	.100000000000E+01	
0.	0.	0.	0.
0.	0.	0.	0.
0.	0.	0.	.872020900000E+05
0.	-.193424040000E+06	-.589400000000E+03	
.100000000000E+01	0.	0.	0.
0.	0.	0.	0.
0.	0.	0.	0.
0.	0.	0.	
0.	0.	0.	0.
0.	0.	0.	0.
0.	0.	0.	0.
0.	.193400000000E+06	0.	

```
.825
  T
 .92039179144393E-01-.11654315425123E-01-.12552934386826E+00 .12991616531590E+00
-.16731513315278E+00-.54906550467919E-01 .11232242190525E+00-.33951509821190E-01
 .25661250399087E+00 .23728743975749E+00 .10118167346036E-02
 .58621245639060E+00-.25110777647203E+01 .12924344888369E+01-.12036663230693E+01
 .83569732004162E+00 .16822540652744E+01-.17036101894468E+01 .14169123542322E+01
-.92846729342097E+00 .16730890244924E+00 .70324919680479E+00
-.20102552364770E+01 .13479801415545E+01-.28893520097497E+01-.75631838856816E+00
 .59551572876444E-02-.10211708574411E+01 .11019380405796E+01-.24411171257506E+01
-.45899823654996E+01 .35874013177045E+00-.11552424058953E+01
 .32648265353059E+01 .30501324024654E+00-.39416094664588E+00-.53470166785280E+00
 .79438629650588E+00 .95398101539848E+00 .40709346637868E-01-.72162782782854E-01
 .15688021899916E+01-.16164745279093E+00 .53470189487151E+00
-.24663332815949E+01-.60636874363625E+00 .55945127470874E+00 .19191692819152E+00
-.52778083708534E+00-.57664803542914E+00-.52223372994092E-01 .27506987432007E+00
-.16956864889228E+01 .15347589086946E+00-.29197626878675E+00
-.55199340125432E+00-.50236211061420E+00 .71023294701155E+00 .47708432986360E+00
-.47950990176764E+00 .55401111151139E+00-.15709235810732E+00 .75165332848883E-01
-.65173936262130E+00-.92891980944570E-01 .82061746183354E+00
-.98498704396806E+00-.67096859655902E-01 .23079295937717E+00 .81651303150363E+00
-.90595146986539E+00 .12797302193610E+00-.25008025732607E+00 .51424269756672E+00
-.99314303749173E+00 .26285576309988E+00 .24680836427240E+00
-.43482670248692E+00 .54066781255981E+00-.44241813951027E+00-.38435599792669E+00
 .14228218653233E+00 .51193680855241E+00 .10016069831443E+01-.27097624361962E+01
-.31238103462393E+01-.38694783048555E+00-.26136145461874E+00
-.73850091831787E+00 .11505315155233E-01 .66918583743035E+00 .24156012804542E+00
-.34263468442782E+00 .76670511740588E+00 .88394703731092E+00-.17848176806006E+01
-.23875742043448E+01-.40149807721605E+00 .54819872835219E+00
 .25887043032005E+00-.29191967602173E+00-.38190900917604E+00-.88921129390110E+00
 .73458594624844E+00 .15144635475451E+00-.10467828617440E+01-.55934779315791E+00
-.68569905629664E+00 .37797152162693E+00-.60139778707298E+00
-.35434242490282E+00-.12336329472674E-01-.37383658755999E+00 .32397609793043E+00
-.54754457791682E+00-.17790402055607E+00 .63507741636499E+00-.10993986600609E+01
-.52008283947555E+00 .29293621564148E+00 .32228651585433E+00
-.28316254435903E+02-.36922253202744E+01-.36554036997203E+01 .20863227313002E+01
```

```
-.23410501323667E+01 .34748537215446E+01 .50723884273585E+01-.10943066566818E+01
 .15902336097438E+01-.48312822837527E+01-.16574736441863E+01
-.87174673039796E+03 .98147694211E04E+01 .19220849940156E+02-.48600103690884E+02
 .52555295375289E+02 .65457822999557E+01-.10691198275147E+02 .18073525856157E+01
 .47459335700241E+01-.10661864551282E+02 .58666755885648E+00
 .44105495880696E+02-.73482818870111E+01-.17079864340348E+02 .48485980288144E+02
-.41808640326354E+02-.43472723397828E+01-.29025708591183E+02 .21615742960457E+01
 .53562940758184E-01-.59176413859617E+02 .14674504561756E+02
-.27824757927811E+03 .18677088843947E+02 .31509598814606E+02-.11441377471712E+03
 .11168344608313E+03 .40186091283816E+01-.24260696659121E+02 .66174521656123E-01
-.35600731261411E+01 .36301360315489E+02-.17112351187734E+02
 .74055513182824E+02-.13795641143187E+02-.25020512271379E+02 .99168837671441E+02
-.96440927084780E+02 .78331550042085E+00 .29348305843109E+02-.57223026050397E+00
 .46467135474532E+01-.32873051418308E+02 .14543137733572E+02
 .66989540899283E+02-.56209442405572E+01-.62799871302191E+01 .13056769692640E+01
-.21468288422260E+01-.10389941389754E+02-.13445527337460E+02 .21918986568037E+01
-.10486885851200E+02 .27467613664136E+02-.81563492557343E+01
 .39322618284221E+03-.14753187903035E+02-.14044618255786E+02 .13588717047707E+02
-.12875903754839E+02-.10993241586604E+02-.48960419563177E+02 .90877047842612E+01
-.89326974537793E+01 .17880693041715E+02 .15960474637689E+01
-.42032063698222E+02 .52919142777412E+01-.38323305457688E+01 .17269287355304E+01
-.65967918611850E+00-.64518689683676E+00-.25177438572896E+00-.43105767851042E+01
-.68084936285713E+01-.86699934266578E+01-.83478525248761E+01
 .11978457557420E+03-.59330739659482E+00-.08894267187564E+01 .42471838162885E+01
-.52552762879968E+01-.10737170366134E+02-.14193657067264E+02-.39784104653509E+00
-.16833334417793E+02 .36576551592823E+02-.14628835018360E+02
-.40227748361018E+03 .85093469503675E+01 .63575844560278E+01 .70327528698101E+00
 .37662642256966E+01 .15616143623346E+02 .39024356332322E+02-.10222673905792E+02
 .21379649759334E+02-.99084556688146E+02 .11716314064118E+02
 .13043285359691E+03-.10300579229145E+02-.12941784763901E+02 .20334411014025E+02
-.18739217323978E+02-.16505033760058E+02-.25069637169307E+02 .58163660858281E+00
-.13153465477671E+02 .68077319107801E+01-.55739009226153E+01
-.54844296202973E+02-.41478917581740E+01-.64523501424803E+01 .27063509598366E+02
-.28085711242721E+02-.86497331844157E+01 .24113772677008E+02-.25637019788490E+01
 .22506703373698E+01-.34986618885990E+01 .23804693579243E+01
-.38137938773773E+04-.11719386319597E+02 .22427381960061E+02 .17488317952889E+03
```

```
-.19147263672297E+03 .19383002326155E+01 .11886169374785E+03 .15348703039825E+02
-.12949504932573E+03 .62635253288503E+03-.14220689552270E+03
 .14421018867905E+03 .50770051431440E+01-.70609770551464E+01 .62111327287042E+02
-.84817446688314E+02-.18716296087002E+02 .11376028987670E+01 .20825246022869E+02
-.13046152726081E+03 .56312342897197E+03-.10484618271959E+03
-.13604205427455E+04-.28585916285834E+01 .80837596312301E+00-.42626869591150E+02
 .61780569634494E+02 .17575922673093E+02 .18148660941254E+01-.13071672852439E+02
 .74067086466879E+02-.29924925298366E+03 .50561039434504E+02
 .51018275664745E+03 .10038158303539E+01-.42962842481220E+01 .63594787312663E+02
-.83326439249458E+02-.14755691865287E+02 .24232518206760E+02 .12668151825561E+02
-.83955944402628E+02 .36025673916625E+03-.67134500147037E+02
 .27305530225092E+03-.33584056013845E+01-.22166280291548E+01 .17190279426658E+02
-.17264375578391E+02-.28539474067585E+01 .46401785661669E+01 .26784476690938E+01
-.21669940007385E+01-.31918821899603E+01-.66269949599385E+01
 .15523866166556E+04-.90016805027577E+00-.26978881648732E+01 .57178179578883E+01
-.87968204414007E+01-.67053885766076E+01-.28980443003465E+02 .85881396921185E+01
-.86888760714756E+01 .23549282096882E+02 .10447108389754E+02
-.19224108495903E+03 .45529502187516E+01 .96978819476809E+00-.17534599022224E+02
 .15775115193230E+02-.56195759037244E+01-.11666867470484E+02 .14238948188137E+01
-.21559137926217E+02 .78157595830413E+02-.26226096064333E+02
 .51564536694907E+03 .32972016819310E+01 .28269118516515E+01-.31010115553371E+02
 .32601157367370E+02-.91440350460321E+01-.34085155163512E+02 .29613739551999E+01
-.36777709730391E+01-.17332582422323E+02-.94202833464343E+01
-.17476351527789E+04-.37470562927037E+01-.95148205444707E+01 .71700041922159E+02
-.79909476869569E+02 .51153370281333E+01 .83010156684608E+02-.49933562327181E+01
-.35420782874133E+02 .17537479305902E+03-.43491568465755E+02
 .59061879125863E+03-.14568247294605E+01-.36537630156258E+01 .30106182475570E+02
-.37684652034670E+02-.13944588975457E+02-.21580463150182E+02 .95540419184419E+01
-.48990052843586E+02 .16773426624502E+03-.50971753777993E+02
 .20564254408671E+02 .44742053770105E+01 .14449930869777E+01-.29495074068992E+02
-.31161537477131E+02-.76133196683818E+00-.43053777830211E+01-.35369565752020E+01
 .42442003207834E+01-.14122121370427E+01-.87888177026054E+00
 .87198088824936E+03 .37739309233401E     .10604098033011E+01 .19839806747591E+02
 .49395034794321E+02-.27526529099710E+    .33165052037777E+01-.59890643457258E+00
 .49520950010003E+01-.61647218474467E+01-.21952423424222E+01
-.73126414532083E+02-.10866902996630E+03-.20198454158909E+02 .96230154803595E+03
```

```
 .102942849461O5E+04 .575899543015O9E+01 .117929575412780E+03 .29895413667553E+02
-.424863945716?3E+02 .66416269532399E+02 .82701211632067E+01
 .26486797294277E+03 .64676357201076E+02 .59518805458072E+01-.50688873793887E+03
-.53656254944509E+03-.69240165874378E+01-.56376216487331E+02-.54606914992725E+01
 .15346551010292E+02-.39277375595955E+02-.21409583294777E+01
-.59609576792676E+02-.46494897280932E+02-.46037835078468E+01 .43434287364156E+03
 .46639042268142E+03 .51232514465962E+01 .47575650340450E+02 .41771012867311E+01
-.11585404968995E+02 .31595147517030E+02 .13448848569767E+01
-.67129607831250E+02-.70778308954557E+01 .10236099848492E+02 .16613197024908E+02
 .98967291561651E+01 .11137663560403E+01-.48404154352111E+01-.26334145720587E+02
 .21710684844463E+02 .68142493477079E+01-.42787097934602E+01
-.41885309241911E+03-.75958099523776E+02-.55631252158745E+01 .48114785503447E+03
 .49628512896416E+03 .41829070732115E+01 .57493222652319E+02 .11191794722789E+01
-.12890172252288E+02 .40260623129873E+02 .25520618876759E+01
 .38866224654437E+02-.11289043516269E+02-.35659549206377E-01 .11756688460449E+03
 .12659173477832E+03-.44981850387235E+00 .12341671247793E+02-.36384762320232E+01
 .31319549980396E+01 .87764338494750E+01-.72449912277981E+00
-.11769541202100E+03-.96577271032884E+01 .78834689989398E+01 .28319725348426E+02
 .22974292045775E+02-.10257649135712E+00-.16475653442992E+01-.22541410549995E+02
 .19743873892509E+02 .68796834717914E+01-.40908491399911E+01
 .39336064116722E+03 .26127180309986E+02 .25515554756149E+01-.88005276287106E+02
-.83740773232124E+02-.50974245023331E+00-.14754909073324E+02-.42543646946992E+01
 .92764811913795E+01-.56440897769032E+01-.30741845094606E+01
-.14034585674589E+03 .17210049205555E+02 .11571764768592E+02 .72029475279936E+02
 .65852085070519E+02 .92514934352052E+00 .13432337355963E+01-.36230751180892E+02
 .32749697114220E+02 .16098723043222E+02-.71333023760451E+01
1256.        57.07        102.692      186.623      214.599      254.988      308.078      412.958
460.8        567.444      655.536
.2           .005         .005         .005         .005         .005         .005         .005
.005         .005         .005
1.           .025213      .309078      .0558515     .0324708     .285298      .00947477    .0186?16
.0239448     .0193181     .095727
0.0          -.7508       -.5217       .1062        -.1607       .0729        -.0979       -.006
.1083        -.0457       -.0004
.0022969     .000208
0.           .0265
```

91

ORIGINAL PAGE IS
OF POOR QUALITY

.7073	.7536
.2797	.3461
.2615	-.5631
.3228	.6369
.0072	-.0377
-.1578	-.4537
.0218	.0317
.0532	-.0550
.1019	.0562
.0068	-.0085

END OF FILE

C-2

APPENDIX E

SAMPLE OUTPUT FROM "SAFSS"

1 DAST ARW-1 OLD DATA SYMETRIC ROOT LOCUS 830220 19.17.2

MACH = .825 11 VIBRATION MODES

15TH ORDER CONTROL LAW CHARACTERISTIC LENGTH 97.800

ALTITUDES TO BE EVALUATED--
15000.0

MODAL FREQUENCIES-

1256.00 57.0700 102.692 186.6[illegible] [illegible] 254.980 300.[illegible]70 412.958 460.800 507.444
655.536

MODAL DAMPING--

.200000 .500000E-02 .500000E-02 .500000E-02 .500000E-02 .500000E-02 .500000E-02 .500000E-02 .500000E-02 .500000E-02
.500000E-02

GENERALIZED MASSES

1.00000 .252130E-01 .[illegible]678 .556515E-01 .[illegible]4700E-01 .[illegible]05296 .947477E-02 .166485E-01 .239448E-01 .193181E-01
.957270E-01

P1 MATRIX--

.920392E-01	.116543E-01	.[illegible]529	.129916	.167315	.549066E-01	.112322	-.399516E-01	-.256613	.237287	.101182E-02
.586212	-2.51106	1.[illegible]9243	1.20367	.8[illegible]397	1.68225	1.70361	1.41691	-.928467	.167309	.703249
-2.[illegible]01626	1.54796	.[illegible]8935	.756310	.595516E-02	1.0211[illegible]	1.10194	2.44112	-4.58998	.358740	1.15524
3.26403	.305615	.[illegible]4161	.534702	.[illegible]94386	.953981	.407093E-01	-.721628E-01	1.56880	-.161647	.534702
2.46633	-.[illegible]	[illegible]	.131017	.5[illegible]	.576648	-.522234E-01	.275070	-1.69589	.153476	.291976
.551993	-.502362	.710233	.477084	.479510	.554011	.157692	.751653E-01	-.651739	-.928920E-01	.820617
.984987	-.670969E-01	.2[illegible]0793	.816513	.905951	.127973	-.250898	.514243	-.993143	.262856	.246008
-.434827	.540660	.44[illegible]418	.384356	.142282	.511937	1.00161	-2.76976	-3.12381	-.386948	.261361
-.730501	.115053E-01	.669186	.241560	.342635	.766705	.883947	-1.78482	-2.38757	-.401498	.548193
.258870	-.291920	-.381909	.889211	.714586	.151446	1.04678	.559348	-.685699	.377972	-.601398
-.354342	-.123363E-01	.37[illegible]837	.323976	.547545	.177904	.635077	-1.09940	-.520083	.292936	.322287

P2 MATRIX--------

-28.3183	-3.69223	-3.65540	2.08832	-2.34105	3.47485	5.07237	-1.08431	1.58023	-4.90129
-1.65747									
-871.747	8.91477	10.2208	-48.8001	52.5553	6.54578	-10.6812	1.80735	4.74593	-10.6620
.586658									
44.1055	-7.34828	-17.0800	48.4860	-41.8086	-4.34727	-28.0257	2.16157	.533629E-01	-59.1764
14.6745									
-278.248	18.6780	31.5086	-114.414	111.683	4.01881	-24.2607	.681745E-01	-3.56007	38.3014
-17.1124									
74.0555	-13.7956	-25.0205	88.1688	-86.4409	.783316	29.3485	.572230	4.84871	-32.8731
14.5431									
68.9895	-5.62094	-6.27999	1.30568	-2.14683	-10.3899	-13.4455	2.19150	-10.4869	27.4678
0.15635									
393.226	-14.7532	-14.0446	13.5887	-12.8759	-10.9932	-48.9604	9.08770	-8.93270	17.8807
1.59605									
-42.0321	5.29191	-3.83233	1.72683	-.659679	-.645187	-.251774	-4.31058	-6.80849	-8.66999
8.34785									
119.785	-.593087	-8.99943	4.24718	-5.25578	-10.7372	-14.1937	.397841	-18.8333	36.5766
-14.6288									
-402.277	6.50935	6.35758	.703275	3.76626	15.6161	39.0244	-10.2227	21.3796	-99.0846
11.7163									
130.433	-10.3006	-12.9418	20.3344	18.7392	-16.5050	-25.0696	.581637	-13.1535	6.80773
-5.57390									

P3 MATRIX------------- ---------

-54.8443	-4.14789	-6.45235	27.0635	-28.0857	-.864973E-01	24.1138	-2.56370	2.25067	-3.49066
2.38041									
3813.79	-11.7194	-22.4274	174.883	-191.473	1.93830	118.862	15.3487	-129.495	626.353
-142.207									
144.210	5.07701	-7.06098	62.1113	-84.8174	-18.7163	1.13760	20.8252	-130.462	563.123
-104.846									
1300.42	-2.65859	.006376	42.6269	61.7806	17.5758	1.81487	-13.0717	74.0671	-299.249
50.5610									
510.183	1.00302	-4.29628	63.5948	-83.3264	-14.7557	24.2325	12.6682	-83.9559	360.257
-67.1345									
273.055	-3.35841	-2.21603	17.1903	-17.2644	-2.85395	4.64018	2.67845	-2.16669	-3.19188
-0.62699									
1552.39	-.900168	-2.69789	5.71782	-8.79682	-6.70539	-28.8804	8.58814	-8.68888	23.5493
10.4471									
182.242	4.55295	.969789	-17.5346	15.7751	-5.81958	-11.6669	1.42389	-21.5581	78.1576
-26.2261									
515.645	3.29720	2.82691	-31.0161	32.6012	-8.14404	-34.0852	2.96137	-3.87777	-17.3323
-9.42028									
-1747.64	-3.74706	-9.51482	71.7000	-79.9085	5.11534	83.0102	-4.99336	-35.4208	175.375
-43.4916									
590.619	-1.45682	-3.65376	30.1062	-37.6847	-13.6446	-21.5805	8.55404	-48.9901	167.734
-50.8718									

RO MATRIX----------------------

20.5643	4.47421	1.44409	-29.4951	-31.1615	-.761332	-4.30538	-3.53698	4.24420	-1.41221
-.878882									
871.981	37.7393	1.08041	19.8388	49.3950	-2.75265	3.31657	-.598906	4.95210	-6.16472
-2.19524									
-73.1204	-108.869	-20.1985	962.302	1029.43	5.75900	117.930	29.8954	-42.4864	66.4163
8.27012									
264.868	64.6764	5.85188	-506.689	-536.563	-8.92402	-56.3762	-5.46069	15.3466	-39.2774
2.14036									
-59.6096	-46.4849	-4.60378	434.343	466.390	5.12325	47.5757	4.17718	-11.5854	31.5951
1.34488									
67.1296	-7.07783	10.2361	16.6132	9.89673	1.11377	-4.84042	26.3341	21.7107	6.81425
-4.27871									
-418.853	-75.8581	-5.56313	481.148	496.285	4.18281	57.4832	1.11918	-12.8802	40.2606
2.55208									
30.8662	-11.2890	-.356595E-01	117.567	126.592	-.449819	12.3419	-3.63848	3.13195	8.77643
.724499									
117.695	-9.65773	7.88347	28.3187	22.9743	-.102576	-1.64757	-22.5414	19.7439	6.87968
4.09085									
393.361	26.1272	2.55156	-88.0053	-83.7408	-.509742	-14.7549	-4.25438	9.27648	-5.64409
3.07418									
140.346	-17.2100	11.5718	72.0295	65.8521	.925149	1.34323	-36.2308	32.7497	16.0987
7.13330									

- END OF INPUT DATA _

1 NEW CASE NEW CASE NEW CASE NEW CASE NEW CASE NEW CASE NEW CASE NEW CASE NEW CASE NEW CASE

MACH = .825000
ALT = 15000.0
QBAR = 4.22986
VTRUE= 10467.8

THE GAIN FACTOR = 0.000

EIGENVALUES COMPUTED, ERROR CODE= 0

REAL PART RAD/SEC	IMAGINARY PART RAD/SEC	FREQUENCY RAD/SEC	FREQUENCY CYCLE/SEC	DAMPING FACTOR
-10812.44	0.00	10812.44	1733.67	1.0000
263.78	3716.92	3726.26	597.47	-.0708
263.78	3716.92	3726.26	597.47	-.0708
-2635.31	0.00	2635.31	422.55	1.0000
-1500.09	0.00	1500.09	240.52	1.0000

-1499.91	0.00	1499.91	240.50	1.0000
1145.58	0.00	1145.58	183.68	-1.0000
-2.02	655.42	655.42	105.09	.0031
-2.02	-655.42	655.42	105.09	.0031
136.18	533.34	550.45	88.26	-.2474
136.18	-533.34	550.45	88.26	-.2474
-448.65	300.27	539.86	86.56	.8310
-448.65	-300.27	539.86	86.56	.8310
-18.68	496.15	496.50	79.61	.0376
-18.68	-496.15	496.50	79.61	.0376
-4.64	459.27	459.29	73.64	.0101
-4.64	-459.27	459.29	73.64	.0101
-294.70	326.46	439.80	70.52	.6701
-294.70	-326.46	439.80	70.52	.6701
-3.96	414.33	414.35	66.44	.0096
-3.96	-414.33	414.35	66.44	.0096
-120.00	320.26	342.00	54.84	.3509
-120.00	-320.26	342.00	54.84	.3509
-12.91	307.74	308.01	49.38	.0419
-12.91	-307.74	308.01	49.38	.0419
-295.30	.00	295.30	47.35	1.0000
-295.30	-.00	295.30	47.35	1.0000
-1.45	254.78	254.78	40.85	.0057
-1.45	-254.78	254.78	40.85	.0057
-2.52	202.62	202.64	32.49	.0125
-2.52	-202.62	202.64	32.49	.0125
-177.84	0.00	177.84	28.53	1.0000
-15.60	145.16	145.99	23.41	.1069
-15.60	145.16	145.99	23.41	.1069
-50.00	100.22	112.00	17.96	.4464
-50.00	-100.22	112.00	17.96	.4464
-2.18	102.28	102.30	16.40	.0213
-2.18	102.28	102.30	16.40	.0213
-4.33	97.31	97.41	15.62	.0444
-4.33	97.31	97.41	15.62	.0444
-71.20	0.00	71.20	11.42	1.0000
-50.00	29.39	58.00	9.30	.8621
-50.00	-29.39	58.00	9.30	.8621
-2.00	0.00	2.00	.32	1.0000
-38.39	292.79	295.30	47.35	.1300
-38.39	-292.79	295.30	47.35	.1300
-251.20	1230.62	1256.00	201.39	.2000
-251.20	1230.62	1256.00	201.39	.2000

THE GAIN FACTOR = .250

EIGENVALUES COMPUTED, ERROR CODE- 0

REAL PART RAD/SEC	IMAGINARY PART RAD/SEC	FREQUENCY RAD/SEC	FREQUENCY CYCLE/SEC	DAMPING FACTOR
0812.44	0.00	10812.44	1733.67	1.0000
263.78	3716.81	3726.26	587.47	-.0708
263.78	-3716.81	3726.26	587.47	-.0708
-2634.84	0.00	2634.84	422.47	1.0000
1145.74	0.00	1145.74	183.71	1.0000
-1632.04	0.00	1632.04	261.68	1.0000
-244.66	1234.72	1258.77	201.82	.1944
-244.66	-1234.72	1258.77	201.82	.1944
-1307.10	0.00	1307.10	209.58	1.0000
-590.76	237.34	636.65	102.08	.9279
-590.76	-237.34	636.65	102.08	.9279
-2.02	655.35	655.36	105.08	.0031
-2.02	-655.35	655.36	105.08	.0031
-408.10	356.84	542.11	86.92	.7528
-408.10	-356.84	542.11	86.92	.7528
136.05	533.51	550.59	88.28	-.2471
136.05	-533.51	550.59	88.28	-.2471
-18.65	496.42	496.77	79.65	.0375
-18.65	-496.42	496.77	79.65	.0375
-3.96	459.87	459.89	73.74	.0086
-3.96	-459.87	459.89	73.74	.0086
-3.94	414.33	414.35	66.44	.0095
-3.94	-414.33	414.35	66.44	.0095
-110.94	359.60	376.33	60.34	.2948
-110.94	-359.60	376.33	60.34	.2948
-12.80	307.52	307.79	49.35	.0416
-12.80	-307.52	307.79	49.35	.0416
-118.84	224.73	254.22	40.76	.4675
-118.84	-224.73	254.22	40.76	.4675
-1.45	254.78	254.78	40.85	.0057
-1.45	-254.78	254.78	40.85	.0057
-3.00	202.76	202.78	32.51	.0148
-3.00	-202.76	202.78	32.51	.0148
-141.44	0.00	141.44	22.68	1.0000
-19.50	146.45	147.75	23.68	.1320
-19.50	-146.45	147.75	23.68	.1320
-9.47	125.79	126.15	20.23	.0751
-9.47	-125.79	126.15	20.23	.0751
-2.01	101.40	101.42	16.26	.0198
-2.01	-101.40	101.42	16.26	.0198
-78.64	0.00	78.64	12,61	1.0000
-33.60	68.06	75.90	12.17	.4427
-33.60	-68.06	75.90	12.17	.4427

-39.61	34.05	52.23	8.37	.7584
-39.61	-34.05	52.23	8.37	.7584
-2.00	0.00	2.00	.32	1.0000
-38.39	292.79	295.30	47.35	.1300
-38.39	-292.79	295.30	47.35	.1300

THE GAIN FACTOR = .500

EIGENVALUES COMPUTED, ERROR CODE= 0

REAL PART RAD/SEC	IMAGINARY PART RAD/SEC	FREQUENCY RAD/SEC	FREQUENCY CYCLE/SEC	DAMPING FACTOR
-10812.44	0.00	10812.44	1733.67	1.0000
263.79	3716.91	3726.26	587.47	-.0708
263.79	-3716.91	3726.26	587.47	-.0708
-2634.35	0.00	2634.35	422.38	1.0000
1145.90	0.00	1145.90	183.73	-1.0000
-1678.04	0.00	1678.04	269.06	1.0000
-238.51	1238.83	1261.58	202.28	.1891
-238.51	-1238.83	1261.58	202.28	.1891
-1176.06	0.00	1176.06	188.57	1.0000
-688.79	245.67	731.29	117.25	.9419
-688.79	-245.67	731.29	117.25	.9419
-2.02	655.30	655.30	105.07	.0031
-2.02	-655.30	655.30	105.07	.0031
135.94	533.71	550.75	88.31	-.2468
135.94	-533.71	550.75	88.31	-.2468
-419.30	364.61	555.65	89.09	.7546
-419.30	-364.61	555.65	89.09	.7546
-18.54	496.69	497.03	79.69	.0373
-18.54	-496.69	497.03	79.69	.0373
-3.05	460.36	460.37	73.82	.0066
-3.05	-460.36	460.37	73.82	.0066
-3.91	414.34	414.36	66.44	.0094
-3.91	-414.34	414.36	66.44	.0094
-88.94	370.88	381.40	61.15	.2332
-88.94	-370.88	381.40	61.15	.2332
-12.65	307.24	307.51	49.31	.0411
-12.65	-307.24	307.51	49.31	.0411
-1.44	254.77	254.78	40.85	.0057
-1.44	-254.77	254.78	40.85	.0057
-87.78	227.64	243.98	39.12	.3598
-87.78	-227.64	243.98	39.12	.3598
-3.71	202.77	202.80	32.52	.0183
-3.71	-202.77	202.80	32.52	.0183

-138.11	0.00	138.11	22.15	1.0000
-23.01	151.38	153.12	24.55	.1503
-23.01	-151.38	153.12	24.55	.1503
-5.55	135.30	135.41	21.71	.0410
-5.55	-135.30	135.41	21.71	.0410
-1.94	101.42	101.44	16.26	.0192
-1.94	-101.42	101.44	16.26	.0192
-80.60	0.00	80.60	12.92	1.0000
-37.84	54.62	66.45	10.65	.5695
-37.84	-54.62	66.45	10.65	.5695
-29.72	38.35	48.52	7.78	.6126
-29.72	-38.35	48.52	7.78	.6126
-2.00	0.00	2.00	.32	1.0000
-38.39	292.79	295.30	47.35	.1300
-38.39	-292.79	295.30	47.35	.1300

THE GAIN FACTOR = 1.000

EIGENVALUES COMPUTED, ERROR CODE= 0

REAL PART RAD/SEC	IMAGINARY PART RAD/SEC	FREQUENCY RAD/SEC	FREQUENCY CYCLE/SEC	DAMPING FACTOR
-10812.44	0.00	10812.44	1733.67	1.0000
263.80	3716.90	3726.25	597.47	-.0708
263.80	-3716.90	3726.25	597.47	-.0708
-2633.39	0.00	2633.39	422.24	1.0000
-1737.35	0.00	1737.35	278.57	1.0000
-227.27	1247.02	1267.56	203.24	.1793
-227.27	1247.02	1267.56	203.24	.1793
1146.22	0.00	1146.22	183.78	-1.0000
-915.25	337.20	975.39	156.39	.8383
-915.25	-337.20	975.39	156.39	.8383
-803.99	0.00	803.99	128.91	1.0000
-2.03	655.20	655.20	105.06	.0031
-2.03	-655.20	655.20	105.06	.0031
-427.67	367.26	563.72	90.39	.7586
-427.67	-367.26	563.72	90.39	.7586
135.74	534.14	551.12	88.37	-.2463
135.74	-534.14	551.12	88.37	-.2463
-18.12	497.05	497.38	79.75	.0364
-18.12	497.05	497.38	79.75	.0364
-.75	460.61	460.61	73.85	.0016
-.75	-460.61	460.61	73.85	.0016
-3.84	414.33	414.34	66.44	.0093
-3.84	-414.33	414.34	66.44	.0093

-57.35	373.95	378.32	60.66	.1516
-57.35	-373.95	378.32	60.66	.1516
-12.09	306.48	306.72	48.10	.0384
-12.09	-306.48	306.72	48.10	.0384
-1.43	254.76	254.76	40.85	.0056
-1.43	-254.76	254.76	40.85	.0056
-54.07	237.23	243.31	39.01	.2222
-54.07	-237.23	243.31	39.01	.2222
-5.48	201.22	201.30	32.28	.0272
-5.48	-201.22	201.30	32.28	.0272
-136.03	0.00	136.03	21.81	1.0000
-28.75	161.53	164.07	26.31	.1752
-28.75	-161.53	164.07	26.31	.1752
-5.78	142.03	142.15	22.79	.0407
-5.78	-142.03	142.15	22.79	.0407
-1.91	101.43	101.45	16.27	.0188
-1.91	-101.43	101.45	16.27	.0188
-82.01	0.00	82.01	13.15	1.0000
-45.18	51.63	68.61	11.00	.6585
-45.18	-51.63	68.61	11.00	.6585
-16.93	33.13	37.21	5.97	.4549
-16.93	-33.13	37.21	5.97	.4549
-2.00	0.00	2.00	.32	1.0000
-38.39	292.79	295.30	47.35	.1300
-38.39	-292.79	295.30	47.35	.1300

THE GAIN FACTOR = 2.000

EIGENVALUES COMPUTED, ERROR CODE= 0

REAL PART RAD/SEC	IMAGINARY PART RAD/SEC	FREQUENCY RAD/SEC	FREQUENCY CYCLE/SEC	DAMPING FACTOR
-10812.44	0.00	10812.44	1733.67	1.0000
263.81	3716.89	3726.24	597.47	-.0708
263.81	-3716.89	3726.24	597.47	-.0708
-2631.41	0.00	2631.41	421.82	1.0000
-1812.64	0.00	1812.64	280.64	1.0000
1146.86	0.00	1146.86	183.88	-1.0000
-208.12	1262.85	1279.88	205.22	.1628
-208.12	-1262.85	1279.88	205.22	.1629
-1014.27	553.18	1155.32	185.24	.8779
-1014.27	-553.18	1155.32	185.24	.8779
-709.98	0.00	709.98	113.84	1.0000
-2.10	655.06	655.06	105.03	.0032
-2.10	655.06	655.06	105.03	.0032

-432.60	367.60	567.69	91.02	.7620
-432.60	-367.60	567.69	91.02	.7620
135.58	535.15	552.06	88.52	-.2456
135.58	-535.15	552.06	88.52	-.2456
-17.32	487.01	487.31	79.74	.0348
-17.32	-487.01	487.31	78.74	.0348
2.95	457.65	457.66	73.38	-.0064
2.95	-457.65	457.66	73.38	-.0064
-3.75	414.09	414.11	66.40	.0090
-3.75	-414.09	414.11	66.40	.0090
-9.24	361.74	361.86	58.02	.0255
-9.24	-361.74	361.86	58.02	.0255
-6.17	305.81	305.87	48.04	.0202
-6.17	-305.81	305.87	49.04	.0202
-31.78	275.19	277.02	44.42	.1147
-31.78	-275.19	277.02	44.42	.1147
-1.38	254.80	254.81	40.88	.0054
-1.38	-254.80	254.81	40.86	.0054
-2.68	198.69	198.71	31.86	.0133
-2.68	-198.69	198.71	31.86	.0135
-134.84	0.00	134.84	21.62	1.0000
-40.86	164.96	169.94	27.25	.2405
-40.86	-164.96	169.94	27.25	.2405
-7.89	144.33	144.54	23.18	.0546
-7.89	-144.33	144.54	23.18	.0546
-1.89	101.44	101.46	16.27	.0187
-1.89	-101.44	101.46	16.27	.0187
-82.89	0.00	82.89	13.29	1.0000
-47.78	51.01	69.89	11.21	.6836
-47.78	-51.01	69.89	11.21	.6836
-9.70	28.41	28.14	4.51	.3447
-9.70	-26.41	28.14	4.51	.3447
-1.99	0.00	1.99	.32	1.0000
-38.39	292.79	295.30	47.35	.1300
-38.39	-292.79	295.30	47.35	.1300

THE ZEROS ARE LOCATED AT:

REAL PART RAD/SEC	IMAGINARY PART RAD/SEC
-11034.83	0.00
-4783 10	7047.97
-4783.10	-7047.97
3346.81	7142.08
3346.81	-7142.08
-8401.13	0.00

7407.26	0.00
143.42	3539.80
143.42	-3539.80
-2950.16	0.00
791.25	173.16
791.25	-173.16
-668.29	0.00
-2.70	654.83
-2.70	-654.83
-437.72	366.90
-437.72	-366.90
141.57	534.13
141.57	-534.13
-17.01	495.71
-17.01	-495.71
-3.46	452.61
-3.46	-452.61
-4.08	414.20
-4.08	414.20
-13.26	309.79
-13.26	-309.79
-1.44	254.80
-1.44	-254.80
-60.00	300.06
-60.00	-300.06
-1.21	200.60
-1.21	-200.60
-133.52	0.00
-8.80	145.05
-9.80	-145.05
-1.88	101.45
-1.88	101.45
-83.92	0.00
-50.00	160.39
-50.00	-160.39
-50.00	50.41
-50.00	-50.41
.08	.14
.08	-.14
-.16	0.00
-38.39	292.79
-38.39	-292.79

END OF FILE

BIBLIOGRAPHY

1. Bisplinghoff, R. L.; Ashley, H.; and H fman, R. L.: [illegible] Addison-Wesley.; Reading, Mass., 1955.
2. Edwards, John W.: "Applications of Laplace Transform Methods to Airfoil Motion and Stability Calculations". AIAA Paper 79-077?, St. Louis, Missouri, 1979.
3. [illegible] Unsteady Aerodynamic Modeling for [illegible] AIAA Journal., vol. 15, no. 4, April 1977, pp. 593-595
4. Gilyard, Glenn B.; and Edwards, John W.: "Real Time Flutter Analysis of an Active Flutter-Suppression System on a Remotely Piloted Research Aircraft". NASA TM-84901, January 1983.
5. [illegible] R.; and Potzky, A. S.: "Comparison of Analysis and Flight Test for a Drone Aircraft with Active Flutter Suppression". AIAA Paper 81-0640, Atlanta, Georgia 981.
6. Wykes, John H.; Byar, Thomas R.; Macmiller, Cary J.; and Greek, David C.: "Analysis and Tests of the B-1 Aircraft Structural Mode Control System". NASA CR-144887, January 1980.

End of Document