ANALYSIS OF STALL FLUTTER OF A HELICOPTER ROTOR BLADE

by Peter Crimi

Prepared by
AVCO SYSTEMS DIVISION
Wilmington, Mass. 01887
for Langley Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • NOVEMBER 1973
**Title and Subtitle**  
ANALYSIS OF STALL FLUTTER OF A HELICOPTER ROTOR BLADE

**Abstract**  
A study of rotor blade aeroelastic stability was carried out, using an analytic model of a two-dimensional airfoil undergoing dynamic stall and an elastomechanical representation including flapping, flapwise bending and torsional degrees of freedom. Results for a hovering rotor demonstrated that the models used are capable of reproducing both classical and stall flutter. The minimum rotor speed for the occurrence of stall flutter in hover was found to be determined from coupling between torsion and flapping. Instabilities analogous to both classical and stall flutter were found to occur in forward flight. However, the large stall-related torsional oscillations which commonly limit aircraft forward speed appear to be the response to rapid changes in aerodynamic moment which accompany stall and unstall, rather than the result of an aeroelastic instability. The severity of stall-related instabilities and response was found to depend to some extent on linear stability. Increasing linear stability lessens the susceptibility to stall flutter and reduces the magnitude of the torsional response to stall and unstall.

**Key Words (Suggested by Author(s))**  
Helicopter rotor, Aeroelasticity, Dynamic stall, Torsional stability

**Distribution Statement**  
Unclassified - Unlimited
ANALYSIS OF STALL FLUTTER
OF A HELICOPTER ROTOR BLADE

By Peter Crimi
Avco Systems Division

SUMMARY

A study of rotor blade aeroelastic stability was carried out, using an analytic model of a two-dimensional airfoil undergoing dynamic stall and an elastomechanical representation including flapping, flapwise bending and torsional degrees of freedom. Results for a hovering rotor demonstrated that the models used are capable of reproducing both classical and stall flutter. The minimum rotor speed for the occurrence of stall flutter in hover was found to be determined from coupling between torsion and flapping. Instabilities analogous to both classical and stall flutter were found to occur in forward flight. However, the large stall-related torsional oscillations which commonly limit aircraft forward speed appear to be the response to rapid changes in aerodynamic moment which accompany stall and unstall, rather than the result of an aeroelastic instability. The severity of stall-related instabilities and response was found to depend to some extent on linear stability. Increasing linear stability lessens the susceptibility to stall flutter and reduces the magnitude of the torsional response to stall and unstall.
INTRODUCTION

Aeroelastic stability of a helicopter rotor blade is a multifaceted problem because of the extreme variations of the aerodynamic environment within the flight envelope of the aircraft. In hovering flight, a blade can undergo classical binary flutter (Ref. 1) or stall flutter (Ref. 2). In forward flight, the linear instability experienced by systems with periodically varying parameters can occur (Ref. 3). While these types of instability are not normally encountered with blades of current design, due to the relatively low disc loading and weak coupling of translational and rotational degrees of freedom, they are certainly not precluded from new designs, particularly those intended to extend present performance capabilities. Of immediate concern, however, in both design and operation, is the occurrence of large-amplitude torsional oscillations and excessive control-linkage loads associated with blade stall on the retreating side of the rotor disc at high forward speed or gross weight, effectively limiting aircraft performance. This problem has prompted a number of recent studies of dynamic stall and the effects of stall on blade dynamics (Refs. 4-8).

While stall has been identified as a causal element of the problem, the nonlinearity of the stall process, coupled with the unsteady aerodynamic environment, has precluded an analysis to the depth required to gain a thorough understanding of the mechanisms involved. In particular, it has not been clear whether the blade undergoes a true aeroelastic instability, a simple forced response, or some hybrid phenomenon which takes on the character of one or the other extreme, depending on flight conditions and blade vibrational characteristics.

Stall flutter for axial flight is amenable to analysis by empirical methods similar to those developed for analyzing stall flutter in cascades (Ref. 9). The flutter mechanism for that case has been identified as deriving from the extraction of energy from the free stream by the periodic variation of the aerodynamic moment. Analogous methods applied to the forward-flight problem (Refs. 10 and 11) have been inconclusive, however, the primary difficulty possibly being in applying empirical methods without a clear definition of the underlying mechanism of the problem.

A method was recently developed for analyzing dynamic stall of an airfoil undergoing arbitrary pitching and plunging motions which provides an ideal tool for analyzing the stall problem in forward flight. The method, which is described in detail in Ref. 7, employs models for each of
the basic flow elements contributing to the unsteady stall of a two-dimensional airfoil. Calculations of the loading during transient and sinusoidal pitching motions are in good qualitative agreement with measured loads. Dynamic overshoot, or lift in excess of the maximum static value, as well as unstable moment variation, are in clear evidence in the computed results.

This study was directed to analyzing the aeroelastic stability of a helicopter rotor, particularly as it relates to stall, using the method of Ref. 7 to compute aerodynamic loading. The representation of the elastomechanical system includes flapping and flapwise bending degrees of freedom as well as torsion. A listing of the computer program used to perform the calculations is given in Appendix A.
<table>
<thead>
<tr>
<th>SYMBOLS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>b</strong></td>
</tr>
<tr>
<td><strong>( \overline{C}_L )</strong></td>
</tr>
<tr>
<td><strong>C_1</strong></td>
</tr>
<tr>
<td><strong>C_{m,c/4}</strong></td>
</tr>
<tr>
<td><strong>c</strong></td>
</tr>
<tr>
<td><strong>f_\theta</strong></td>
</tr>
<tr>
<td><strong>f_\phi</strong></td>
</tr>
<tr>
<td><strong>h_\beta</strong></td>
</tr>
<tr>
<td><strong>h_\phi</strong></td>
</tr>
<tr>
<td><strong>h_i</strong></td>
</tr>
<tr>
<td><strong>I_o</strong></td>
</tr>
<tr>
<td><strong>I'_\theta</strong></td>
</tr>
<tr>
<td><strong>k_i</strong></td>
</tr>
<tr>
<td><strong>k_\theta</strong></td>
</tr>
<tr>
<td><strong>l</strong></td>
</tr>
<tr>
<td><strong>l_{s_i}</strong></td>
</tr>
<tr>
<td><strong>M_b</strong></td>
</tr>
<tr>
<td><strong>m</strong></td>
</tr>
<tr>
<td><strong>m_{c/4}</strong></td>
</tr>
</tbody>
</table>
\( m_1 \) masses of 2-D system, kg/m
\( R \) rotor radius, m
\( r_0 \) inner radius of blade lifting surface, m
\( r_R \) aerodynamic reference radius, m
\( U \) instantaneous free-stream speed at aerodynamic reference section, m/sec
\( U_0 \) reference speed, \( U_0 = \Omega r_R \), m/sec
\( x_m \) distance aft of elastic axis of blade section mass center, m
\( x \) distance aft of pitch axis of mass center of \( m_1 \), m
\( Z_\beta \) generalized coordinate of 2-D system, equivalent to \( h_\beta \), semichords
\( Z_\phi \) generalized coordinate of 2-D system, equivalent to \( h_\phi \), semichords
\( a \) angle of attack, deg
\( \delta \) flapping hinge offset, m
\( \Theta_o \) collective pitch angle, deg or rad
\( \Theta_1 \) blade tip torsional deflection, rad
\( \tilde{\Theta} \) angle of zero restraint of 2-D system torsion spring, rad
\( \mu \) advance ratio, ratio of forward speed to \( \Omega R \)
\( \rho \) free-stream density, kg/m^3
\( \tau \) dimensionless time, \( \tau = U_0 t/b \)
\( \psi \) blade azimuth angle measured from downwind direction, deg or rad
\( \Omega \) rotor rotational speed, rad/sec
\( \Omega^* \) dimensionless rotor speed, \( \Omega^* = \Omega R/(\omega_o b) \)
\( \omega_f \) flutter frequency, rad/sec
\( \omega_{\theta_0} \) frequency of first uncoupled, nonrotating torsion mode, rad/sec

\( \omega_{\phi_0} \) frequency of first uncoupled, nonrotating flapwise bending mode, rad/sec
Aerodynamic Loading

In the flutter analysis, only leading-edge stall was considered, so the following relates specifically only to that type, even though the basic method can treat trailing-edge stall as well. When the airfoil is not stalled, the flow elements represented are (see Figure la): (1) the laminar boundary layer from the stagnation point to separation near the leading-edge, (2) the small leading-edge separation bubble; and, (3) a potential flow, including a vortex wake generated by the variation with time of the circulation about the airfoil. When the airfoil is stalled, as indicated in Figure 1b, the flow elements are: (1) the laminar boundary layer, (2) a dead-air region extending from the separation point to the pressure recovery point; and, (3) a potential flow external to the airfoil and dead-air region, again including a vortex wake. The analytic representations of these elements are described briefly below. Details are given in Ref. 7.

Potential Flow.—Given the airfoil section characteristics and motions, together with the distribution of pressure in the dead-air region if the airfoil is stalled, the flow and pressure over the airfoil must be determined to compute the integrated load and analyze the boundary layer. The problem was formulated by imposing linearized boundary conditions of flow tangency and pressure, using a perturbation velocity potential derived from source and vortex distributions. The resulting coupled set of singular integral equations is solved by casting the singularity distributions in series form and solving for the unknown coefficients by imposing boundary conditions at prescribed points.

Boundary Layer.—Because the relative importance of the individual elements of the boundary layer flow as they affect dynamic stall could not be established in advance, the representation in Ref. 7 was made as general as possible. The method of finite differences for unsteady flows with variable step size in both streamwise and normal directions, was employed, with the error in each finite-difference approximation the order of the square of the step size. It was determined from preliminary calculations performed for this study that, at least for leading-edge stall, results are virtually unaffected by assuming quasi-steady flow in the boundary layer. That assumption was therefore employed for all flutter computations, to take advantage
Figure 1 FLOW ELEMENTS

(a) ATTACHED FLOW

(1) LAMINAR BOUNDARY LAYER
(2) LEADING-EDGE BUBBLE
(3) AIR FOIL AND VORTEX WAKE

(b) LEADING-EDGE STALL

(1) LAMINAR BOUNDARY LAYER
(2) DEAD-AIR REGION
(3) AIRFOIL AND VORTEX WAKE
of the resulting substantial savings in computer storage requirements and computing time.

Dead-Air Region.—The function of the model of the dead-air region is to define the streamwise distribution of pressure in that region, given the locations of the separation and recovery points and the pressure at the recovery point. The dead-air region is assumed to consist of a laminar constant-pressure free shear layer from separation to transition, a turbulent constant-pressure mixing region, and a turbulent pressure-recovery region. The laminar shear layer is analyzed by the method of Ref. 12, assuming quasi-steady flow. The turbulent mixing and pressure-recovery regions are analyzed using the steady-flow momentum integral and first moment equations. Profile parameters in these regions are assumed to be universal functions of a dimensionless streamwise coordinate, with those functions derived from an exact viscous-inviscid interaction calculation. Matching of approximate solutions for the mixing and pressure-recovery regions at their interface completes the analysis.

Leading-Edge Bubble.—The leading-edge bubble on an unstalled airfoil is analyzed using the same basic relations employed for the dead-air region. Given the boundary-layer parameters at separation, the length of the bubble and the amount of pressure rise possible, for that length, in the pressure recovery region, are computed. That pressure rise is compared with the rise in pressure in the potential flow over the length of the bubble. If the latter is greater than the former, the bubble is assumed to have burst, and the stall process is initiated.

Loading Calculation Procedure.—Calculations proceed by forward integration in time, using the blade motions derived by integrating the equations of motion of the elastomechanical system. If, at a given instant, the airfoil is not stalled, the potential flow is computed, and the boundary layer and leading-edge bubble are analyzed to check for bubble bursting. If the airfoil is stalled, the pressure distribution in the dead-air region is computed, the potential flow evaluated, and the boundary layer is analyzed to locate the separation point. The last two steps are repeated iteratively until assumed and computed separation points agree. Rate of growth of the dead-air region is determined from an estimate of the rate of fluid entrainment derived from the potential-flow solution. Unstall is determined by first postulating its occurrence and analyzing the leading-edge bubble which would then form to ascertain whether that event did in fact occur.
During unstall, the dead-air region is washed off the airfoil at the free-stream speed.

Elastomechanical Representation

The equations of motion for a rotor blade with flapping, flapwise bending and torsional degrees of freedom can be written in the form (Ref. 3)

\[
\frac{d^2 h_\beta}{d \tau^2} + \frac{R}{b \beta} \frac{M_\beta}{M_\beta} \frac{d^2 \theta_1}{d \tau^2} + \frac{\bar{\omega}_\beta}{b} \frac{2}{h_\beta} - \frac{R}{b \beta} \frac{\Omega}{2} \frac{T_\beta}{M_\beta} \theta_1
\]

\[
= \frac{Rb}{U_0} \frac{F_\beta}{M_\beta}
\]

\[
\frac{d^2 h_\phi}{d \tau^2} + \frac{b \beta}{M_\phi \phi} \frac{M_\phi \theta}{M_\phi \phi} \frac{d^2 \theta_1}{d \tau^2} + \frac{\bar{\omega}_\phi}{b} \frac{2}{h_\phi} - \frac{\Omega}{2} \frac{T_\phi}{M_\phi \phi} \theta_1
\]

\[
= \frac{b}{U_0} \frac{F_\phi}{M_\phi \phi}
\]

\[
\frac{d^2 \theta_1}{d \tau^2} + \frac{b \beta}{M_\theta \theta} \frac{M_\beta \theta}{M_\theta \theta} \frac{d^2 h_\beta}{d \tau^2} + \frac{b \beta}{M_\theta \theta} \frac{d^2 h_\phi}{d \tau^2} + \frac{b \bar{\omega}_\theta}{b} \frac{2}{h_\theta} \theta_1
\]

\[
- \frac{b}{R} \frac{\Omega}{2} \frac{T_\beta}{M_\theta \theta} \frac{h_\beta}{h_\beta} - \frac{\Omega}{2} \frac{b}{M_\theta \theta} \frac{T_\phi}{h_\phi}
\]

\[
= \frac{b^2}{U_0} \frac{F_\theta}{M_\theta \theta}
\]
are tip displacements due to flapping and bending, respectively, in semichords, $\Theta_1$ is torsional displacement at the blade tip and the frequencies* are the following functions of rotational speed:

$$\overline{\omega}_\beta^2 = -\overline{\Omega}^2 \frac{T_{\beta\beta}}{M_{\beta\beta}}, \quad \overline{\omega}_\phi^2 = \overline{\omega}_\phi^2 - \overline{\Omega}^2 \frac{T_{\phi\phi}}{M_{\phi\phi}},$$

$$\overline{\omega}_\Theta^2 = \overline{\omega}_{\Theta_0}^2 - \overline{\Omega}^2 \frac{T_{\Theta\Theta}}{M_{\Theta\Theta}}.$$

The inertial and centrifugal-force coefficients are given by

$$M_{\beta\beta} = \int_{\delta}^{\infty} (r + \delta)^2 m dr, \quad M_{\phi\phi} = \int_{\delta}^{\infty} m f_\phi^2 dr,$$

$$M_{\Theta\Theta} = \int_{\delta}^{\infty} I_\Theta f_\Theta^2 dr,$$

$$M_{\beta\Theta} = -\int_{\delta}^{\infty} m x_m (r - \delta) f_\Theta dr,$$

$$M_{\phi\Theta} = -\int_{\delta}^{\infty} m x_m f_\phi f_\Theta dr,$$

$$T_{\beta\beta} = -\int_{\delta}^{\infty} r (r - \delta) m dr.$$

*Barred quantities are dimensionless frequencies, $U_0/b$ being reference frequency; e.g., $\overline{\Omega} = \Omega b/U_0.$
The complexity of the aerodynamic representation precludes evaluation of the generalized forces \( F_\beta \), \( F_\phi \) and \( F_\theta \) by the usual strip approximation. It was felt essential, however, to retain both translational degrees of freedom in the investigation of the forward-flight problem, so a simple two-dimensional model of the dynamics could not be used. Therefore, a two-dimensional airfoil suspended in such a way as to have three degrees of freedom was analyzed. Inertial and stiffness parameters were assigned to make the coupled natural frequencies of the two-dimensional system match those of the rotor blade.

The system analyzed is shown schematically in Figure 2. The matching of the two-dimensional system with the blade dynamics proceeds as follows. Three generalized coordinates are first defined to correspond to those of the blade. Clearly, angular displacement \( \theta_1 \) should correspond to blade torsional displacement at the blade tip. The counterparts of flapping and bending, \( Z_\beta \) and \( Z_\phi \), respectively, are defined by

\[
Z_\beta = A_1 h_1 + Bh_2, \quad Z_\phi = A_2 h_1 - Bh_2
\]

where

\[
A_1 = \frac{\bar{\omega}_\beta^2 - \bar{\omega}_\phi^2}{\bar{\omega}_\phi^2 - \bar{\omega}_\beta^2}, \quad A_2 = \frac{\bar{\omega}_2^2 - \bar{\omega}_\phi^2}{\bar{\omega}_\phi^2 - \bar{\omega}_\beta^2}
\]

\[
B = \frac{(\bar{\omega}_2^2 - \bar{\omega}_\phi^2)(\bar{\omega}_2^2 - \bar{\omega}_\beta^2)}{(\bar{\omega}_\phi^2 - \bar{\omega}_\beta^2)\bar{\omega}_2^2}
\]

(1)
Figure 2 TWO-DIMENSIONAL ELASTOMECHANICAL SYSTEM
and \[ \bar{\omega}_1^2 = (k_1/m_1)(b/U_0)^2, \quad 1 = 1, 2. \]

With the above definitions, \( Z_{\beta} + Z_{\phi} = -h_1 \), to give the correct translational correspondence. It can further be shown that the uncoupled natural frequencies of the two-dimensional system match those of the blade, provided

\[
\left( \frac{k_\theta + k_1 l_{s_1}^2 + k_2 l_{s_2}^2}{I_0} \right) \left( \frac{b}{U_0} \right)^2 = \bar{\omega}_\theta^2
\]

while \( \bar{\omega}_1^2 \) and \( \bar{\omega}_2^2 \) satisfy

\[
\bar{\omega}_1^2 \bar{\omega}_2^2 = \bar{\omega}_\phi^2 \bar{\omega}_\beta^2,
\]

\[
\bar{\omega}_1^2 + (1 + m_2/m_1) \bar{\omega}_2^2 = \bar{\omega}_\phi^2 + \bar{\omega}_\beta^2 \quad (2)
\]

By comparing the generalized masses of the two systems, it follows that

\[
m_1 b^2/I_0 = -A_1 M_{\beta\beta} b^2/(M_{\theta\theta} R^2)
\]

\[
A_2/A_1 = M_{\beta\beta} /(M_{\phi\phi} R^2) \equiv \lambda_m
\]

The last relation, together with Eqs. (1) and (2), fixes \( m_2/m_1 \):

\[
m_2/m_1 = \frac{(1 + \lambda_m)(\bar{\omega}_\phi^4 + \lambda_m \bar{\omega}_\beta^4)}{(\lambda_m \bar{\omega}_\beta^2 + \bar{\omega}_\phi^2)^2} - 1
\]

Equating the corresponding coefficients of the characteristic equations of the two systems provides three additional relations, which can be solved for the coupling parameters \( \bar{x}, l_{s_1}, l_{s_2} \). That calculation is outlined in Appendix B.
To complete the matching, quasi-steady approximations to the damping terms of the flapping equations are equated with the result that

\[ \frac{m_l R}{-A_l} = 4 \frac{r_R}{R} \frac{M_{\beta\theta}}{R^2 \left[ 1 - \left(\frac{r_0}{R}\right)^4 \right]} \]

\[ \frac{U}{U_0} = 1 + \frac{4}{3} \left[ \frac{1 - \left(\frac{r_0}{R}\right)^2}{1 - \left(\frac{r_0}{R}\right)^4} \right] \mu \sin \psi \]

where \( \Omega r_R = U_0 \). The aerodynamic reference radius \( r_R \) was selected to be \( .75R \).

The angle of zero restraint in torsion was varied periodically to approximate the effects of cyclic pitch variation in forward flight, according to the formula

\[ \bar{\theta} = \theta_0 \left[ 1 - 2 \left(\frac{R}{r_R}\right) \mu \sin \psi \right] \]

This variation gives nominally constant lift.

The equations of motion were solved by integrating analytically, using linear extrapolations to approximate the variation of lift and aerodynamic moment over the interval of integration. This scheme was found to give satisfactory results, provided the time interval of integration is no longer than about one fifth of the period of the coupled mode having the highest natural frequency.
RESULTS OF COMPUTATIONS

Configurations Analyzed

Vibrational and aerodynamic characteristics of the blade analyzed were selected to correspond to those of the model rotor blade described in Ref. 2. That blade is un-twisted, of constant chord, with offset flapping hinge. Pertinent dimensionless parameters of the model blade are listed in Table 1.

TABLE 1

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>b/R</td>
<td>0.0435</td>
</tr>
<tr>
<td>$\delta / R$</td>
<td>0.0543</td>
</tr>
<tr>
<td>$r_o / R$</td>
<td>0.174</td>
</tr>
<tr>
<td>$\omega_0 / \omega_\phi$</td>
<td>3.69</td>
</tr>
<tr>
<td>$\rho R b^2 / M_b$</td>
<td>0.00431</td>
</tr>
<tr>
<td>$x_m / b$</td>
<td>0.216</td>
</tr>
<tr>
<td>$m R / M_b$</td>
<td>1.055</td>
</tr>
<tr>
<td>$I_\phi / M_b R$</td>
<td>$3.51 \times 10^{-4}$</td>
</tr>
</tbody>
</table>

Two elastomechanical configurations in addition to the nominal one were analyzed. One of these had $\omega_0 / \omega_\phi = 2.5$, with all other parameters as listed in Table 1. The third configuration had $x_m / b = 0.108$, with the remaining parameters as listed in Table 1.

The bending mode shape, which was computed by a finite-element method, was found not to vary appreciably over the range of rotational speeds of interest. The mode shape for $\omega_\phi / \Omega = 1.26$, which is plotted in Figure 3, was used for all computations. The torsional mode shape for the nonrotating blade, also shown in Figure 3, was used to evaluate torsional inertia parameters.
Figure 3  BENDING AND TORSION MODE SHAPES
The test blade had a NACA 23012 section. The variation of static lift and moment coefficients with angle of attack for this section were computed from a series of transient pitch calculations, and are shown in Figure 4, together with the measured section characteristics, from Ref. 13. The aerodynamic model is seen to give nearly the correct maximum lift, but at a slightly lower angle of attack, and, as indicated from the variation of $C_m c/4$, the computed center of pressure is somewhat further aft than that of the actual airfoil section below the stall angle.

### Stability in Hover

Initial calculations were performed for hovering flight, with the nominal configuration, to allow a direct comparison with the test results of Ref. 2. First, rotor speed was varied parametrically, with the collective pitch at a value well below the stall incidence. A classical bending-torsion instability was encountered at $\Omega^* = \Omega R/(\omega_0 b) = 5.3$ with $\omega_f/\omega_0 = .803$. The variation of bending, flapping, and torsional displacements with azimuth angle at flutter onset are shown in Figure 5. By way of comparison, tests (Ref. 2) yielded classical flutter at about $\Omega^* = 7.1$ with $\omega_f/\omega_0 = .72$.

It should be noted that since the system stability was analyzed by direct simulation, a precise point of linear instability was not computed. The values of $\Omega^*$ at onset of a linear instability, both for hover and forward flight, were obtained by successively increasing or decreasing rotor speed, in small steps, until the transient response changed from convergent to divergent, or visa versa. The maximum error in the value of flutter speed, for the results presented here, is estimated to be about three percent.

Susceptibility of the system to stall flutter was investigated next. It was found that a torsional limit cycle, at approximately the highest coupled natural frequency of the system, could be triggered for $\Omega^*$ as low as 3.4. Computed blade motions for stall flutter at $\Omega^*$ of 3.5 are shown in Figure 6.

For $\Omega^*$ below 3.4, a limit cycle could not be set up, regardless of the initial conditions or the collective pitch angle. Severe oscillations involving repeated stall and unstall could be made to occur by imposing a large initial bending deflection. However, the flapping response modulated the torsional response, and caused continuous stall and/or unstall of the blade over a significant portion of
Figure 4 AIRFOIL SECTION CHARACTERISTICS FOR NACA 23012
Figure 5 DISPLACEMENT TIME HISTORIES AT CLASSICAL FLUTTER ONSET
$\Omega^* = 5.3, \theta_0 = 11$ deg, $\mu = 0$
Figure 6  DISPLACEMENT TIME HISTORIES FOR STALL FLUTTER

$\Omega^* = 3.5, \theta_0 = 15.0 \text{ deg}, \mu = 0$
a revolution, due to the large plunging rate generated by the flapping motion. An example of this occurrence is shown in Figure 7. Thus, while stall flutter involves only the rotational degree of freedom, the results obtained indicate that the minimum speed for its occurrence is determined by coupling with a translational degree of freedom.

Results for the hovering case are summarized in Figure 8, which compares computed and measured flutter speed and frequency, plotted against collective pitch angle. No upper limit in collective pitch angle for the occurrence of stall flutter was calculated, since that limit would depend strongly on initial conditions, and so would be arbitrary. Quantitative differences between the computed and measured stability boundaries of Figure 8 can be attributed in large part to the use of a two-dimensional aerodynamic model, which cannot precisely reproduce the aerodynamic coupling between the rotational and translational degrees of freedom.

From the basic similarity of the computed and measured stability boundaries and the character of the computed instabilities (Figures 5 and 6) it can be concluded that the aerodynamic and dynamic models formulated are capable of reproducing both classical and stall flutter as experienced by a rotor blade, and so can be employed to investigate the forward-flight problem.

**Stability in Forward Flight**

The nominal configuration was analyzed next for an advance ratio of .1. Computations were carried out in the same sequence as for hovering. First, the rotational speed at which classical flutter occurs was determined. Then, stall-related instabilities were investigated.

A linear bending-torsion instability of the Floquet type (Ref. 14) was encountered at $\Omega^* = 5.2$. Blade motions as a function of azimuth angle at flutter onset are shown in Figure 9. The torsional and bending displacements are seen to display the aperiodic character typical of this type of instability. The flapping motion is the steady-state response to the cyclic pitch variation.

An instability analogous to stall flutter in hover was found to occur for $\Omega^*$ as low as about 4.4, with collective pitch angle greater than 12 deg. Blade motions for $\Omega^* = 4.8$ are shown in Figure 10. The torsional displacement time history, while not strictly periodic, is nonetheless
Figure 7 BLADE RESPONSE BELOW STALL FLUTTER BOUNDARY

$\Omega^* = 3.1, \theta_0 = 15.0 \text{ deg}, \mu = 0$
Figure 8 FLUTTER SPEED AND FREQUENCY VARIATION WITH COLLECTIVE PITCH ANGLE FOR A HOVERING ROTOR
Figure 9  DISPLACEMENT TIME HISTORIES AT LINEAR INSTABILITY ONSET
\[ \Omega^* = 5.2, \theta_0 = 6 \text{ deg}, \mu = 0.1 \]
Figure 10 DISPLACEMENT TIME HISTORIES FOR STALL FLUTTER

\[ \Omega^* = 4.8, \theta_0 = 13 \text{ deg}, \mu = 0.1 \]
brought about by successive stall and unstall. The azimuth positions at which those events occur are marked by (S) and (U), respectively, on the $\psi$-scale.

The blade motions for the type of instability shown in Figure 10 are not of the same character as those of particular concern in the limiting of helicopter performance, in that the excessive torsional displacements shown in Figure 10 persist over a complete revolution of the blade. The control load time history, taken from flight test (Ref. 6), shown in Figure 11 illustrates the type of stall-related blade motions usually encountered at a thrust level or forward speed near the upper limit of an aircraft. Large oscillations in the control loads, presumably deriving from blade torsional oscillations, are seen from Figure 11 to persist only between about $\psi = 270$ deg and $\psi = 400$ deg, rather than throughout a complete revolution of the blade.

A torsional displacement time history closely resembling the variation of control loads in Figure 11 was obtained for $\Omega^* < 4.4$, for collective pitch angles between 12 and 13 deg. Results for two typical cases are shown in Figures 12 and 13. The occurrences of stall and unstall are indicated on the abscissas. The large oscillations in torsion are clearly related to stall, but their persistence is not the result of successive stalling and unstalling, as would be the case for true stall flutter. The blade appears to be responding to the sudden changes in aerodynamic moment at stall onset and unstall, as can be seen by comparing the variation of moment coefficient shown in Figures 12 and 13 with that of torsional displacement, and noting the azimuth positions at which stall and unstall occur. There is some cyclic stall-unstall within the stall zone evident in the results, particularly at the higher rotor speed ($\Omega^* = 4.15$, Figure 13). However, the major contributors to the oscillations appear to be the initial and final pulses associated with stall and unstall upon entering and leaving that zone. There are, in general, two cycles of torsional oscillation of excessive amplitude after the blade unstalls the last time on a given revolution. The response can be regarded as transient, on a localized time scale, or forced, when viewed on a scale of several rotor revolutions. The severity of the response is apparently due in part to the suddenness of load changes at stall and unstall, and partly to the relative lack of aerodynamic damping in pitch, particularly when the blade is not stalled.

If the collective pitch angle is increased, the blade does undergo stall flutter, as seen from the time history plotted in Figure 14. These results are for the same rotor
Figure 11 VARIATION OF PITCH LINK LOAD IN FLIGHT TEST OF CH-47 AT 123 KNOTS
(from Ref. 6)
Figure 12 DISPLACEMENT AND MOMENT TIME HISTORIES FOR EXCESSIVE TORSIONAL RESPONSE

\[ \Omega^* = 3.89, \, \theta_0 = 12 \text{ deg, } \mu = 0.1 \]
Figure 13 DISPLACEMENT AND MOMENT TIME HISTORIES FOR EXCESSIVE TORSIONAL RESPONSE

$\Omega^* = 4.15$, $\theta_0 = 12$ deg, $\mu = 0.1$
Figure 14 DISPLACEMENT TIME HISTORIES FOR STALL FLUTTER AT LOW ROTOR SPEED
\[ \Omega^* = 3.89, \theta_0 = 14.3 \text{ deg}, \mu = 0.1 \]
speed as those of Figure 12, but with $\theta_0$ increased from 12 deg to 14.3 deg. Successive stall and unstall persists over the whole revolution of the blade for this case.

It could be argued that the blade torsional oscillations of Figures 12 and 13 are still a manifestation of stall flutter, even though successive stall and unstall is not taking place, since the aerodynamic moment can undergo unstable variations when the blade remains stalled throughout a cycle (Ref. 4). It may, in fact, be the case that the large deflections do result partly from that effect, so choosing to term them as simply a response may be somewhat misleading. On the other hand, the solutions are distinctly different from what is definitely stall flutter obtained both in hover (Figure 6) and in forward flight (Figures 10 and 14) so that label would seem to be even less appropriate. Further, the persistence of the oscillations after exit from the stall zone is clearly symptomatic of a response, so, for lack of a more precise term, solutions of the type shown in Figures 12 and 13 are identified in what follows as excessive response.

Linear Stability Boundaries

The value of $\Omega^*$ at the onset of linear instability was determined for the three configurations considered, for advance ratios of 0, .1, .2, and .3. The effects of advance ratio and torsion-bending frequency ratio on linear stability are shown in Figure 15, where $\Omega^*$ is plotted against $\mu$ for two different frequency ratios. Increasing advance ratio is seen to cause some decrease in flutter rotational speed, with most of the decrease occurring between advance ratios of .1 and .2. The substantial decrease in frequency ratio, from 3.69 to 2.5, caused only about a 4 percent reduction in flutter speed over the range of advance ratios considered. The insensitivity to frequency ratio can be attributed to the large chordwise mass imbalance, which produces the same effect in classical binary flutter of a wing (Ref. 15).

The effect of chordwise mass imbalance on linear stability is shown in Figure 16, where $\Omega^*$ at flutter onset is plotted against $\mu$ for values of $x_m$ of .216 and .108 semichords. As one would expect, the reduction in $x_m$, and hence in the coupling between bending and torsion, causes a substantial increase in the flutter rotational speed.
Figure 15 EFFECT OF ADVANCE RATIO AND TORSION-BONDING FREQUENCY RATIO ON LINEAR STABILITY - $X_{mb} = 0.216$
Figure 16 EFFECT OF Xm ON LINEAR STABILITY.

$$\frac{\omega_{\theta_0}}{\omega_{\phi_0}} = 3.69$$
Stall Flutter and Response Boundaries

The effect of forward speed on stall-related instabilities for the three configurations was investigated by systematically varying the collective pitch angle and advance ratio, with \( \Omega^* \) equal to 3.89. In order to relate the results to rotor performance, a mean lift coefficient \( \bar{C}_L \) is defined, according to

\[
\bar{C}_L = \frac{\bar{l}}{\rho \Omega^2 R^2 b}
\]

where \( \bar{l} \) is the time-averaged lift per unit span at the aerodynamic reference radius. This coefficient is, to a good approximation, directly proportional to the thrust coefficient (see Ref. 16). The two-dimensional aerodynamic model does not provide a good measure of \( \bar{C}_L \) when the rotor is partially stalled, so \( \bar{C}_L \) was computed assuming it varies linearly with the collective pitch angle, using the formula

\[
\bar{C}_L = a(\mu)(\theta_0 + .0217)
\]

The slope \( a \) and zero-lift collective pitch angle of -.0217 rad were obtained from calculations of \( \bar{C}_L \) for the nominal configuration with stall precluded. The variation of \( a \) with \( \mu \) is shown in Figure 17.

The results obtained for the nominal configuration are summarized in Figure 18 as a plot of \( \bar{C}_L \) vs \( \mu \). As thrust is increased at a given \( \mu \), the rotor is seen to first encounter a region of excessive response, of the type discussed previously, and then, for \( \mu \) of .2 or less, a region where stall flutter occurs. Increasing advance ratio has the effect of suppressing the tendency for stall flutter. At \( \mu = .2 \), stall flutter occurs at \( \bar{C}_L = .85 \), but a further increase in \( \bar{C}_L \) results in excessive response again. At \( \mu = .3 \) a limit-cycle type of oscillation could not be triggered at all. As a result, stall flutter is confined to a region somewhat as indicated by the shaded area in Figure 18.

The suppression of stall flutter at high advance ratio is apparently caused by an effect similar to the one encountered at low rotor speed in hover, whereby the flapping motion prevented a limit cycle from occurring. This can be seen from the blade motions obtained for \( \mu = .3 \) and
Figure 17 VARIATION OF $a = \frac{d \bar{C}_L}{d \theta}$ WITH ADVANCE RATIO
Figure 18 STALL STABILITY BOUNDARIES FOR $\Omega^* = 3.89$, $\omega_0/\omega_0 = 3.69$ AND $Xm/b = 0.216$
\( \bar{C}_L = .78 \), plotted in Figure 19. On the first revolution, as the blade enters the stall zone on the retreating side, it appears that a limit cycle is being set up, with repeated stall and unstall occurring. However, at about \( \psi = 420 \) deg, the flapping motion has built up in response to the large cyclic pitch changes, producing a negative plunging rate sufficient to keep the blade unstalled over the remainder of its passage on the advancing side. Then, when the blade again enters the stall zone, the large positive flap-induced plunging rate precludes unstall until exit from the stall zone at about \( \psi = 670 \) deg. As a result, the blade subsequently undergoes excessive torsional response, rather than stall flutter.

The effect of torsion-bending frequency ratio on stall-related instabilities can be seen from Figure 20, where \( \bar{C}_L \) is plotted against \( \mu \) for \( \omega \theta_0/\omega \phi_0 = 2.5 \). No instance of excessive torsional response occurred with this configuration for an advance ratio of .2 or less. Instead, limit-cycle type oscillations were set up, with almost no evidence of suppression by the flapping motion, even at relatively high values of \( \bar{C}_L \) with \( \mu = .2 \). At \( \mu = .3 \), however, only excessive response was obtained, similar to the results for \( \omega \theta_0/\omega \phi_0 = 3.69 \).

The marked deterioration in stability at the lower frequency ratio is apparently associated with the lessened linear stability of the system. The configuration with \( x_m/b = .108 \), which is more stable, in the linear sense, than the nominal one, exhibited a trend opposite to the one resulting from a decrease in frequency ratio. The results for the smaller mass center offset, shown in Figure 21, are similar to those of the nominal configuration, Figure 18, but the region in which stall flutter occurs is somewhat reduced, there being no occurrence of stall flutter at an advance ratio of .2. Also, the amplitude of the torsional oscillations in the region of excessive response is considerably reduced, as evidenced by comparing the blade motions plotted in Figure 22, which are for \( \mu = .1 \), \( \bar{C}_L = .95 \) and \( x_m/b = .108 \), with those of the nominal configuration plotted in Figure 12.
Figure 19 DISPLACEMENT TIME HISTORIES AT HIGH ADVANCE RATIO —
\( \Omega^* = 3.89, C_L = 0.78, \mu = 0.3 \)
Figure 20 STALL STABILITY BOUNDARIES FOR $\Omega^* = 3.89$, $\omega_\theta / \omega_\phi = 2.5$
AND $X_m/b = 0.216$
Figure 21  STALL STABILITY BOUNDARIES FOR $\Omega^* = 3.89$, $\omega_{\theta_o}/\omega_{\phi_o} = 3.69$ AND $X_m/b = 0.108$
Figure 22 DISPLACEMENT TIME HISTORIES FOR EXCESSIVE TORSIONAL RESPONSE.
$\Omega^* = 3.89$, $\ddot{C}_L = 0.95$, $\mu = 0.1$, AND $Xm/b = 0.108$
CONCLUSIONS

An analysis has been performed of the aeroelastic stability of a helicopter rotor blade in hovering and forward flight. An analytical model of an airfoil undergoing unsteady stall and an elastomechanical representation including flapping, flapwise bending and torsional degrees of freedom were employed in the study. The following conclusions can be drawn from the results obtained.

1. Analysis of aeroelastic stability for a hovering rotor demonstrated that the aerodynamic and dynamic representations developed are capable of reproducing classical and stall flutter.

2. While stall flutter is an instability involving a single rotational degree of freedom, the minimum rotational speed for its occurrence, in hover, is determined from coupling with a translational degree of freedom.

3. In forward flight, the rotor can undergo a linear instability analogous to classical flutter and a stall-induced flutter which, while not manifested by a strictly periodic limit cycle, has the same basic mechanism for its occurrence as stall flutter of a hovering rotor.

4. The large stall-related torsional oscillations which limit forward speed and thrust are primarily the response to the rapid changes in aerodynamic moment which accompany stall and unstall, rather than the result of an aeroelastic instability.

5. Linear stability is relatively insensitive to advance ratio for advance ratios as large as .3.

6. While excessive response due to stall occurs at high advance ratio, stall flutter is precluded by the large flap-induced plunging rates.
7. The severity of stall-related instabilities and response depends to some extent on linear stability. Increasing linear stability lessens the susceptibility to stall flutter and reduces the magnitude of the torsional response to stall and unstall.
APPENDIX A

PROGRAM LISTING
APPENDIX A

PROGRAM LISTING

A listing of the FORTRAN coding of the computer program follows. The program was written in FORTRAN IV for use on an IBM 360/75 computer.
PROGRAM TO ANALYZE UNSTEADY AIRFOIL STALL

COMMON /RBL1/ NTIME, NDIRC, ISTD
   COMMON /CLCMRL / CLVB, CMVR, CMPAVB

COMMON /INPTVB/ FTVB(64), FPVB(64), FPPRVRB(64), DDIRVB(64), SETUPS17
   COMMON /INPPVB/ FNVB(64), DELVB, XMUVB, FOVB, XMUVAB, SETUPS18
   COMMON /INPVVB/ AQVB, ATCVB, ATSVB, ROVB, RVB(64), SETUPS19
   COMMON /INPVB/ MVB(64), NVR, SETUPS20

COMMON /INPUTS/ NSBL, NZ, NOFF, NGAM, NSIG, SETUPS21
   COMMON /INPVS/ NCST, NCORD, LOWER, MSTOP, MAXT, MTR, SETUPS22
   COMMON /INPVS/ NOTBL, INDV, ELSIG, DXI, REB, ROB, SETUPS23
   COMMON /INPVS/ FRZ, ARR, AMPLU, FREQU, ALPH1, ALPH2, SETUPS24
   COMMON /INPVS/ HEAVE, AROT, FREQF, PHIH, NY, RV1, SETUPS25
   COMMON /INPVS/ DRY, Y(100), TEST, UPRIM, XU(30), YU(30), SETUPS26
   COMMON /INPVS/ XL(30), YL(30), ER1, ER2, ER3, BD3R, SETUPS27
   COMMON /INPVS/ RRDR, SETUPS28
   COMMON /INPVS/ H, CMPA, CMPAS, BARG, EML, HVOR, NVOR, SSPA, SVOR, TORF, X1VOR
   COMMON /INPVS/ I, PLOTOPI, PSILOW, PSIUP, J, NOUT

   COMMON /ZZZ/ (3)

DIMENSION USAV(300,100), SCALS(300)
DIMENSION USAV(300), SCALS(300)
DIMENSION CAMBR(24), THICK(24)
DIMENSION XGAM(30), XSIG(100), XSIGA(100), XSIGB(100), X(300), X(300),
   LSBL(300), XSIG(100)
DIMENSION ACAP(30,3), BCAP(100,3), ASZ(30), AS(30,30), BS(30,30), ASZH
   L1(100), ASH(30,30), ASH(30,30), ARH(100), UE(300,3)
DIMENSION ALAM(30), VZIP(30), FPRES(100), GAMMA(1000), XIW(1000)
DIMENSION BLAM(30), FLAM(10), XFLAM(10)
DIMENSION SCALE(300,2), U(1,1,1), UC(100,3), V(100,2),
   1, P(200,7)

DOUBLE PRECISION CHAT(60,60), RMAT(130)

DATA IN, MOUT, NF/ 5,6, 24/
DATA PI, TIME, UINF, RENE, UST0P/3.14159,0.1,4.7564,2.8/
DATA FLAM /1.75,1.75,1.724,1.527,1.354,1.354,663,452,25/, MAIN 18
DATA 14, 21/
DATA XFLAM /-100, -11.26, -7.01, -3.48, -1.76, 6, 0.1, 888, 9/, MAIN 20
DATA DEG0S /1.74 53292 51994 3300-2/, MAIN 22
DATA DEG0S /1.74 53292 51994 3300-2/, SUPPL 38

EQUATION (CHAT(1), USAV(1), ASH(1), SCALS(1))

IF ISTO = -1, TIME DERIVATIVES NOT USED
ISTD = 1
RAD = 180. /PI
IL = 8888
NDIMC = 60

CALL SETUPS

IF(ISTD .EQ. 1) GO TO 40
D0 100 J = 1,300
SCAL$ = 1.
D0 101 I = 1,100
100 USE (J, I) = 0
CONTINUE

CALL READIN (IL, N)

C NOTE - OFFSETS ARE PUT IN AS LISTED IN THEORY OF WING SECTIONS, I.E.
C AS A FRACTION OF TOTAL CHORD, XI BEING MEASURED FROM THE
C LEADING EDGE. ENSURE NF IS AN EVEN NUMBER.
C
TIME = 0.
NTIME = 0
NMAKE = 999
ISEP = 0
ISEP = 0
ISSEP = 0
IWASH = 2
UNF = 1.

INDV = INDV + 1
WRITE (MOUT, 6)
PITCH = ALPH
IF (INDV + MODR, LE. 2) PITCH = PITCH - ALPH
IF (INDV .EQ. 2)
X AMPLU = 1.33333 * XNUAV8 * (1. - ROVB**3) / (1. - ROVB**4)
IF (INDV .EQ. 2) FREQU = BDBR/RQRDB
IF (INDV .GE. 2) GO TO 343
WRITE (MOUT, 235) XNOR, SVOR, HVOR, BARG, XIVOR, EMI, TOFR, SSPA
RY = RY1
HVOR1/2 = 0
BARG = BARG/6.2832
343 CALL SECT (XU, YU, XL, YL, NOFF, NF, RDBB, TMBBB, MBDBB, THICK, CAMBR)
DQ 7875 N = 1, NF
CAMBR(N) = CAMBR(N) + CHDBB
7EH7 THICK(N) = THICK(N) + TMDBB
WRITE (MOUT, 41)
WRITE (MOUT, 7) AMPLU, FREQU, ALPH, ALPH2, HEAVE, AROT, FREDF, RDBB, REB
WRITE (MOUT, 8)
WRITE (MOUT, 9) (N, CAMBR(N), THICK(N), N = 1, NF)
MX = NSBL + NZ - 1
CALL SCALE (SBL, NSBL, FRZ, AROB, RDBB)
CALL CORDX (NSBL, NZ, RDBB, SBL, X, X1)
DO 2420 M = 1, MX
IF (XCM) = 1.7 2420, 2419, 2419
2419 MEND = M - 1
GO TO 2421
2420 CONTINUE
2421 MX = MEND
MXM1 = MX-1
IF (MX+1,1) = 1.
EPSLF = 2.*(X(NZ)-X(NZ-1))
FPS1F = X(MX)-X(MX-1)
ALTG = 8.3664/SQRT(REB)
  IF (ISTD.EQ.1) GO TO 50
  DO 2422 M = 1, MX
  SCALE(M,1) = 0.
  SCALE(M,2) = 0.
  DO 2422 N = 1, NY
  U(M,N,1) = 0.
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
  CONTINUE
  NSIGA = NSIG
  NSIG = NSIG
  NSIG = NSIG + 1
  MTR = MTR + 1
  NOTR = NOTRL + 1
  NMAX = NELS
  CCNA = 3.75*PI/DOXI
  ANGS = PI/DOATINSIG
  CALL SET3X(NSIG1,1,1,2,,XSIG,ANGS)
  XSEP = 1.1
  DO 2430 N = 1, NSIG
  XSIGN(N) = XSIG(N)
CMAT(M,2)=XGAM(M)  
DO 8457 N=3,NGP1  
8457 CMAT(M,N)=AS(M,N-1)  
8458 CONTINUE  
8458 CONTINUE  
CALL ALSOL(NGP1,CMAT,RMAT)  
DO 8459 N=1,NGP1  
ACAP(N,1)=RMAT(N)  
ACAP(N,3)=RMAT(N)  
8459 ACAP(N,2)=ACAP(N,1)  
DO 2784 M=1,MAX  
SIGN=1.  
IF(M-N) 2774,2775,2775  
2774 SIGN=-SIGN  
2777 CALL QCAL(ISEP,NGAM,NSIG,NSIG,ACAP,BCAP,THICK,RCBR,GAMAV(1),UI)MAIN 162  
1NF,XP(M)=UF(M,1),SIGN  
2784 UF(M,2)=UE(M,1)  
DO 1004 M=2,NGAM  
1004 BLAM(M)=(1.125*XGAM(M)+1.875*(1.1*XGAM(M)))*ALOG((1+XGAM(M)))/(1-XGAM(M))  
RM=NGAM-1.125/DXI  
CALL CLCMTNCST,TSFP,NGAM,NSIG,NSIG,NSIG,NSIG,NSIG,NSIG,ACAP,RM)MAIN 504  
1AP,THICK,RCBR,GAMAV,UIF,UDOT,DXI,ARDT,CMP)  
IF (INDV.EQ. 2)  
1CALL SUPPL  
C  
INDEXING IN TIME IS CARRIED OUT AT THIS POINT.  
C  
9999 CONTINUE  
CALL ACUPCU(IACU)  
IF (IACU .LT. 35000) GO TO 99  
C  
NOTE - FOR READ-IN OF FCIL MOTIONS, MAKE ALPH1 = ALPHA,  
C  
ALPH2 = ALPHA-DOT, AND HEAVE = H-DOT.  
C  
IF(WCTR.EQ. 2)  
XRAD(1N,2,N=8989) ALPH1,ALPH2,HEAVE  
158 NITS=1  
TIME=TIME+DXI  
NTIME=NTIME+1  
NWAKE=NTIME+2  
IF(NWAKE=-998) 202,201,201  
201 NWAKE=-998  
202 IF(MAX-NTIME) 8989,8800,8800  
8800 SAFU=UIF  
L=L+1  
PIL(1) = BCBR / RRDBR * TIME * RAD  
PSI360= AMOD( PIL(1), 360.)  
UIF=L+AMPLU*SN(FREQ*TIME)  
IF(INDV.EQ. 2)  
XCALL SUPPL(UINF)  
PITCH = ALPH1  
IF(INDV + MOTR .LE. 2) PITCH = PITCH - ALPH2*COS(FREQ*TIME)  
UDOT=FREQU*AMPLU*COST(FREQ*TIME)  
STEPX=5*DXI*(UINF+SAVEU)  
DO 1003 J=2,NWAKE
JC=NOWAFF-J+2
GAMAW(JC)=GAMAW(JC-1)

1CC3 XIW(JC)=XIW(JC-1)+STFPX
IF (ISEP) 209, 209, 207

2CC7 DC 2008 N=1,NSIG
RCAP(N,3)=RCAP(N,2)
2CC8 RCAP(N,2)=RCAP(N,1)
DO 4433 N=1,NSIG1
XSIG(N)=XSIG(N)

4433 XSIGA(N)=XSIG(N)
GO TO 2010

2009 DEADL=0.
ELDNT=UNF
2010 DO 1014 M=1,MX
UE(M,3)=UE(M,2)
1014 UE(M,2)=UE(M,1)
DEADL=DEADL
ELDNT=ELDNT
ALAM(1)=(1.125+.75*ALOG(STEPX*.5))/DXI
TO 1005 N=1,NGP1
1005 ALAM(M)=BLAM(M)+.75*(1+1-XGAM(M))/STEPX*ALOG(1+STEPX-XGAM(M))
11/(1-XGAM(M))/DXI

DC 2006 M=1,NGP1
ACAP(M,3)=ACAP(M,2)
2006 ACAP(M,2)=ACAP(M,1)
ACAP(M,1)=ACAP(M,2)+ACAP(2,21)-2.*ACAP(1,3)+.5*ACAP(2,11)

ALPHS=VZIP(1)
CALL WASH(XGAM,NGAM,TIME,ALPH1,ALPH2,HEAVE,AROT,FREQF,PHIH,UINF,CAMAIN

1MBR,NF,VZIP,MOTR,INDV)
DO 1006 M=1,NGP1
ASZ(M)=1.*XGAM(M)
ASZ(M,1)=XGAM(M)+ALAM(M)
SUM=0.
DO 4343 J=2,NWM1
4343 SUM=SUM+(GAMAW(J)+GAMAW(J+1)-GAMAW(J))*XGAM(M)-XIM(J)/XIM(J+1)

1-XIM(J))/ALOG((XI(J)+1-XGAM(M))/(XIM(J)-XGAM(M))

ELX=1.-(XGAM(M)
IF (M=1) 1006,2130,1006
2130 ELX=1.

1006 AR(M)=2.*VZIP(M)+ALAM(M)*ACAP/3.*SUM-GAMAW(2)*XGAM(M)*ALOG

IF (ISEP) 3247,4444,3247
3247 GO TO (3344,3349),IWASH
3344 XSPIX+XSIP+DXI
IF (XSIPX=XMAX) 3248,3347,3347
3347 IWASH=2
ISEP=0
XSEP=1.1
DO 3015 K=1,3
DO 3015 N=1,NSIG

52
3015 RCAP(N,K)=0.
GO TO 4444
3345 IF (INDT) 3349, 3348, 3248
3349 IF (NITS-1) 3248, 3349, 3248
3349 IF (INDV.EQ.2) GO TO 6349
IF (VZIP(1)-ALPHS) 6349, 6348, 6348
6348 NITS=2
GO TO 3248
6349 CALL UNPUPINGAM, AR, ALAM, AFACR, RMAT, CMAT, XGAM, AS, ACAP, M, NZ, IF, XSIG
GO TO 2785
3248 XATT=XSEP+DEADL+5*(ELD1+ELD0)*NXI
DEADL=XATT-XSEP
DIFF=1.-XATT
XTEST = XSEP + 3.* EPSL
CALL SETSX(NSIG1, XSEP, XATT, XSIG, ANGS)
DO 4434 N=1, NSIG
4434 XBSIG(N)=.5*(XSIG(N)+XSIG(N+1))
DO 3086 M=1, NGPI
DO 3086 N=1, NSIG
3086 BM(N)=0.
DO 3087 M=1, NGPI
IF(XGAM(M)-XSEP) 3088, 3088, 3088
3088 BS(M,N)=0.
DO 3087 M=1, NGPI
IF(XGAM(M)-XSIG(1)) 3092, 3090, 3090
3090 MARK=I
GO TO 3094
3094 CONTINUE
3155 CALL=XSIG(MARK)-XSIG(MARK-1)
RS(M,MARK-1)=(XSG(MARK)-XGAM(M))/WIDES
BS(M,MARK)=(XGAM(M)-XSIG(MARK-1))/WIDES
RS(M,1)=SQRT((XGAM(M)-XSEP)/(XATT-XGAM(M)))
3085 IF(DIFF1.5-6) 3087, 3098, 3098
3087 IF(DIFF-1.5)*SQR(DMADL)*(2.+DIFF*(SQR((1.-XGAM(M))/XATT-XGAM(M))))
GO TO 3087
3167 BS(M,1)=DIFF*(1.5)*SQR(DMADL)*(3.+XATT-4.*XGAM(M))
3087 CONTINUE
C
C SET-UP OF THE SECOND SET OF EQUATIONS STARTS HERE.
C
DO 4350 K=1, NSIG
4350 IF(XSIG(K)-1) 4348, 4349, 4349
4348 CSK=XBSIG(K)
SINK=SQR((1.-COSK*COSK)
THETK=ARCT(COSK)
TAN=(SIN(5.*THETK)/COS(5.*THETK)
ASHZ(K)=TAN+(*COSK)(1.3*COSK)/UNIF+THXI*(PI-THETK+SINK)
ICONA(1.+COSK)*SINK2)/UNIF
ASH(K,1)=5.*(ASHZ(K)-TANT)*SINK
COUNT=1.
DO 4355 N=2, NGAM
COUNT=COUNT+1.
4355 ASHTK,NI=SIGN(COUNT+1.)*THETK/(COUNT+1.)*SINK
IN((COUNT-1.)*THETK)/(COUNT-1.)/(DXi*UINF)
GO TO 4350
4349 ASHZ(K)=0.
DO 4359 N=1,NGAM
4355 ASH(K,N)=0.
4350 CONTINUE
IF(DIFF-1.E-6) 5005,5006,5006
5005 PREC=0.
GO TO 5007
5006 CALL ATPR(PREC,NSIG,NSIG,ASZ,AS,AR,CMAT,MRAT,NGAM,TF,ACAP,THICK,RMAIN)
IF(DDB,GAAMAW,UIIF,JDOT,DXI,RACAP)
5007 CALL MXER(IFPRES,PREC,UIIF,JDOT,THICK,NSIG,NSIG,IND,DEL1,THETMAIN)
11,REF,USEP,X4,CP1)
CPCT=CP1
DO 4800 K=1,NSIG
CORD=XSIG(K)
BSh(K)=1-.THXi*BINT(XSEP,XATT,CORD)/UINF
DO 4808 N=2,NSIG
4808 BSh(K,N)=F(K)(SIG(N-1),XSIG(N),XSIG(N+1),CORD)+THXIG(XSIG(N-1),XSIG(N),XSIG(N+1),CORD)/UINF
CALL ESIGI(2,NSIG,XSIG,BCAP,CORD,VALI)
CALL ESIGI(3,NSIG,XSIG,BCAP,CORD,VAL2)
ARH(K)=FPRST(K)+T2.*VALI-.5*VAL2)/(DXi*UINF)
IF(CORD-1.) 5009,4800,4800
5008 CALL EGAMI(2,NGAM,ACAP,BCAP,1,2),XSIGA(1),XSIGA(NGAM+1),GAAMAW(2),MAIN
1CORD,VALI)
CALL EGAMI(3,NGAM,ACAP,BCAP,1,3),XSIGB(1),XSIGB(NGAM+1),GAAMAW(3),MAIN
ICORD,VAL2)
ARH(K)=ARH(K)+T2.*VALI-.5*VAL2)/(DXi*UINF)+.0625*FACT*PI*(1.+CORD/Main)
4850 CONTINUE
4444 CONTINUE
C
CALCULATIONS FROM THIS POINT ON COMBINE THE
CASES OF STALLED AND UNSTALLED AIRFOILS.
4
DO 6500 H=1,NGPI
CMMAT(H,1)=ASZ(H)
DO 6485 N=1,NGAM
64E5 CMAT(H,N)=ASZ(N)
IF(1SEP) 6486,6500,6486
64E6 DO 6499 N=1,NSIG
NGG=N+NGPI
64E9 CMAT(H,NGG)=BS(N,N)
6500 CONTINUE
TF(1SEP) 6502,6501,6502
6501 NTO=NGPI
GO TO 6751
6502 DO 6750 K=1,NSIG
KK=K+NGPI
RMA(T(KK))=ARH(K)
CMAT(KK,1)=ASHZ(K)
DO 6748 N=1,NGAM
6748 CMAT(KK,N+1)=ASH(Z,N)
DO 6750 N=1,NSIG
NGG=N+NGP1
6750 CMAT(K),NGG)=RSH(K,N)
NTOT=NSIG+NGP1
6751 CALL ALSOL(NTOT,CMAT,RMAT)
DO 6800 N=1,NGP1
6800 ACP(N,1)=RMTAT(N)
IF(ISFP).EQ.6805,6820,6805
68C5 DO 6810 N=1,NSIG
NGG=N+NGP1
6810 BCAP(V,1)=RMTAT(NGG)
6820 CONTINUE
GAMA(1)=GAM1(ACAP,IXI,PI)
1 IF (PSI360 .GE. PSILW .AND. PSI360 .LE. PSIUP) GO TO 1736
DO 1785 M=1,MX
SIGN=1.
IF(M-NZ) 1780,1785,1785
1780 SIGN=-SIGN
1785 CALL QECAL(ISEP,NG4M,NSIG,NF,NSIG,ACAP,BCAP,THICK,ROBR,GAMA M(1),UI)
DO 8886 I=1,2
US2=UE(1+1)
DO 8886 M=1,MXMI
USL=UE(M+1)
UF(M+1)=(USL+US2+UE(M+1,1,1))/3.
8866 USZ=US1
GO TO (8351,8352),(8562,8562),(8562)
8351 DO 8352 M=1,4X
8352 SCALSM)=0.
GO TO 1786
8353 CALL YSETRY1,Y(1),NY,Y)
RY=RY1
DO 8354 M=1,MX
8354 SCALSM)=0.
IF(INDV.EQ.2) GO TO 8370
IF(ISEP.EQ.0.AND.VZIP(1).LT.ALPHS) GO TO 1786
8370 CALL STAGMX,NY,NSTOP,MST,DXI,RY,DRY,X,Y,UE,UC,USAV,SCALS,ISEP)
LAMQ=1
XSEPS=XSEP
DXX=DXI
IF(ISEP.EQ.1.AND.ISEP.EQ.0.AND.NITS.EQ.1) DXX=1.E-30
8367 CALL CLC(X,Y,MST,MCNo,NY,RY,DRY,DX,REB,UPRIM,FLAM,XFLAM,TESTU,SC)
14ALF,UE,UC,V,XSEP,USEP,DISP,THETA,LOWER,LAMQ,MSEP,SCS,USA,SCALS,NITEMAIN 403
15,NITEM,NOTBL,XTEST,NZ,NOTU)
IF(TXSEP-XMAX) 7735,7735,7735
7735 IF(ISEP) 1786,1786,7736
7736 DELI=DISP
THETI=THETA
INDT=LAMQ
IF(INOT.EQ.1.AND.NOTAL.EQ.2) GO TO 1786
WRITE(MOUT,23) XSIG(I),CPLT,XSEP
IF(INOT) 8462,8462,8463
8462 IF(ISEP) 8562,8562,8563
8563 IF(NITS=1) 8562,8562,8562
9662 IF(ISEP) 7742,7742,8562
55
8562 CALL RUBB(DEL1, THET1, RFB, XSEP, USEP, XCS, DCP, DEL5, X, XC, MX, NZ, X5, U5, UMAIN 416
IF(ALC, RFNEL, USTOP)
USEP=USEP*0.002046*USEP**3
PDIFF = (USEP-USEP-U5)*USEP
WRITE(MOUT, 22) PDIFF, DCP
IF(DCP-PDIFF) 8263, 8366, 8367
8263 ISEPT=0
GO TO 8463
8366 IF(ISEP) 8368, 8369, 8369
8369 IF(ISEP) 8467, 8467, 8468
8467 I = 1
NITS=2
GO TO 3344
8368 GO TO ((165, 1786), NOTAL
8168 CALL RATT(UV, V, Y, M, X, Z, RYM, USE, X5, DEL5, MST, REB) LAMQ=0
GO TO 8367
8463 IF(ISEP) 7741, 7741, 7742
7741 ISEP=1
NITS=NITS+1
IF(NIT1) 7743, 7743, 7643
7643 ISEP=1
DXSEP=1.*XSEP
DXSEP=6.*XSEP
CALL CPC(ISEP, NGAM, NF, XSIG, NSIG, XSIGA, NSIGA, XSIGB, NSIGB, ACP, BCAP, MAIN 440
LTHICK, RAD, GAMAM, UINF, UDOT, 1., XSEP, DX1, CPL)
GO TO 3248
7742 CALL FLDER(BCAP, XSIG, NSIG, UINF, ELDCT, SIGSUM, BMX) IF(ISEP.EQ.1.AND.ISEP.EQ.0.AND.NITS.EQ.1) GO TO 9210
IF(ISEP*.5) 7841, 7842, 7842
7841 EPS=EPSLE
GO TO 7843
7842 EPS=EPSLE
GO TO 7843
7843 DXSEP=ABS(XSEP-XSEPS)
IF(DXSEP-EPS) 7836, 7836, 9210
7834 IF(ISEP=MAXI) 1786, 1786, 7835
7835 ISEP=0
ISEPT=0
DO 7836 K=1,3
7836 BCAP(N, K)=0.
GO TO 1786
9210 NITS=NITS+1
IF(NITS.EQ.2.AND.ISEP.EQ.0) XSEPS=XSEP
IF(NITS=4) 9211, 9211, 1786
9211 IF(XSEP-XSEPS) 9335, 9305, 9306
9305 XSEP=6.*XSEPS+.4*XSEP
GO TO 9307
9307 XSEP=6.*XSEPS+.4*XSEPS
9308 IF(XSEP-XMAXI) 9212, 9212, 7835
9212 CALL CPC(ISEP, NGAM, NF, XSIG, NSIG, XSIGA, NSIGA, XSIGB, NSIGB, ACP, BCAP, MAIN 466
LTHICK, RAD, GAMAM, UINF, UDOT, 1., XSEP, DX1, CPL)
IF(NOTBL.EQ.2.AND.XSEP.GT.0.1) XSEP=-.98
GO TO 3248
7743 IF(NITS=1) 7737, 7737, 3248
7836 N=1, NSIG
GO TO 8368
7737 N1=MT1+1
ELDOT=ELDI
GO TO 3248
1786 WRITE(MOUT,20) NTIME
WRITE(MOUT,26) XIVOR
PITCH=PITCH+100.*PI
2C5 WRITE(MOUT,10) TIME,UNIF,XSEP,XATT,PITCH
ALDFG=ALPH1/DEGIFS
WRITE(6,9001) Z,ALDEG,ALPH1,ALPH2,HEAVE
IF(PL1360.GE.,PSIL,AND,PSI360.LE.,PSIUP) GO TO 101
IF(NOUT.GE.0,PSIL,AND,PSI360.LE.,PSIUP) GO TO 101
1WRITE(MOUT,11)
IF(NOUT.EQ.0) WRITE(MOUT,12)
WRITE(MOUT,14) ELDOT
WRITE(MOUT,13) XSIG(1),CPOT,X4,CPOT,XATT,PREC
7432 WRITE(MOUT,15) XPC=-1.
DO 7302 N1=1,NCPI
CALL QECAL(ISEP,NGAM,NSIG,NF,NSIG,ACAP,BCAP,THICK,RCBB,GAM(N),MAIN 490
IF(XPC.NE.-1)CALL QECAL(ISEP,NGAM,NSIG,NF,NSIG,ACAP,BCAP,THICK,RCBB,GAM(N),MAIN 491
IF(XPC.NE.-1)CALL CPC(ISEP,NGAM,NSIG,NF,NSIG,ACAP,BCAP,THICK,RCBB,GAM(N),MAIN 492
7433 WRITE(MOUT,16) XPC=1.
CALL CPC(ISEP,NGAM,NSIG,NF,NSIG,ACAP,BCAP,THICK,RCBB,GAM(N),MAIN 493
CALL CPC(ISEP,NGAM,NSIG,NF,NSIG,ACAP,BCAP,THICK,RCBB,GAM(N),MAIN 494
THICK,RCBB,GAM(N),UINF,UDOT,DX1,CPL)CALL CPC(ISEP,NGAM,NSIG,NF,NSIG,ACAP,BCAP,THICK,RCBB,GAM(N),MAIN 495
101 CONTINUE
103 CALL CLCMNCOL(ISEP,NGAM,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,RCMA IN 503
1AP,THICK,RCBB,GAM(N),UINF,UDOT,DX1,AROT,CMPA)CALL CLCMNCOL(ISEP,NGAM,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,RCMAIN 504
P(L,2)=PITC
P(L,3)=Z(3)
P(L,4)=Z(1)
P(L,5)=Z(2)
P(L,6)=CLVAR
P(L,7)=CMPA
IF(L.LT.200)GO TO 98
CALL PLTSB(PLTSB,P,L)
L=0
58 CONTINUE
IF(ISYM.EQ.1)GO TO 9999
DO 7950 M=1,MX
SCALE(M,2)=SCALE(M,1)
SCALE(M,1) = SCALS(M)
DC 7950 N1, NY
U(M,N,2) = U(M,N,1)
7550 U(M,N,1) = USAV(M,N)
GO TO 9999
8589 CONTINUE
99 CONTINUE
 CALL PLOTSB( PLOTOP, P, L )
 CALL ACUCPU(! IACU )
 IF(IACU .LT. 35000) GO TO 60
 GO TO 40
60 CONTINUE
 IF(IACU .EQ. 0.) CALL EXIT
 CALL PLTND
 CALL EXIT
 RETURN
C
C
C
1 FORMAT(13I5) MAIN 23
2 FORMAT(3F10.4) MAIN 24
3 FORMAT(2F10.4) MAIN 25
4 FORMAT(1H1//) MAIN 26
5 FORMAT(6F10.4) MAIN 27
6 FORMAT(1H1,5OX,34HANALYSIS OF UNSTEADY AIRFOIL STALL///) MAIN 28
7 FORMAT(1X,6HUBAR =E13.5/7X,7HUFREQ =E13.5/3X,11HALPHA ONE =E13.5/MAIN 29
  1E13.5/9X,6HRO99 =E13.5///) MAIN 31
8 FORMAT(29X,1HN,25X,4HC(N),26X,4HT(N)/) MAIN 32
9 FORMAT(13O,2E30.5) MAIN 33
10 FORMAT(5X,3HT =E13.5/5X,3HU =E13.5/4X,4HX5 =E13.5/4X,4HXO =E13.5/4MAIN 34
  1X,4HPA =E13.5////) MAIN 35
11 FORMAT(///4X,1HN,11X,1HX,14X,SHVZ(X),12X,5HRN(X),12X,4HAN(N),21X,3HMAIN 36
  3XW,14X,5HGMMA//) MAIN 37
12 FORMAT(15,4E17.5,8X,2E17.5) MAIN 38
13 FORMAT(1H1,8X1LHN,20X,1HX,21X,5HFP(X),22X,5HRH(N),21X,4HB(N)/) MAIN 39
14 FORMAT(///54X,9H, L-DCT =E13.5///51X,27Hpressures in separated flowMAIN 40
  1/55X,1HX,19X,2HCP//) MAIN 41
15 FORMAT(1H1,11X,1HX,16X,3HQEL,15X,3HCPL,15X,3HqeU,15X,3HCP,13X,9HCMAIN 42
1PL - CPU//) MAIN 43
16 FORMAT(6E18.5) MAIN 44
17 FORMAT(10,4E25.5) MAIN 45
18 FORMAT(10X,2E20.5//)) MAIN 46
19 FORMAT(15,5F10.4) MAIN 47
20 FORMAT(1H1,5O,12HTIME STEP NO13//) MAIN 48
21 FORMAT(///40X,26HINCREASE IN CP REQUIRED IS1E13.5///40X,26HINCREASE MAIN 49
  1IN CP POSSIBLE IS1E13.5) MAIN 50
22 FORMAT(///45X,23HPOTENTIAL FLOW XS =E12.4/60X,8HCP(XS) =E12.4/MAIN 51
  1/45X,23BOUNDARY LAYER XS =E12.4//) MAIN 52
23 FORMAT(///45X,45FL0USION FLOW,3H =E12.4/60X,8HCP(XS) =E12.4/MAIN 53
  112X,4HIN) MAIN 54
24 FORMAT(15,5F10.4) MAIN 55
25 FORMAT(12X,4HNN =E12.4,3X,3HM =E12.4,3X,3HM =E12.4,3X,3HM =E12.4,3X MAIN 56
  1HX1 =E12.4///12X,4HM =E12.4,3X,4HNN =E12.4,3X,4HPPA =E12.4///) MAIN 57
26 FORMAT(4X,4X1 =E12.5) MAIN 58
9001 FORMAT(0, 750, "EQUIVALENT ROTOR BLADE RESPONSE") SUPPL380
9CC1A // T 5, "FLAP DISP =", G14.5
9CC1B // T47, "RENDING DISP =", G14.5
9CC1C // T39, "TORSIONAL DISP =", G14.5
9CC1D // T38, "SECTION PITCH ANGLE =", F9.3, ' DEGREES OR ','
9CC1E // F9.4, ' RADIANS '
9CC1F // T21, "SECTION PITCH RATE =", G14.5
9CC1G // T71, "SECTION PLUNGING RATE =", G14.5 //
END
SUBROUTINE SUPPL
IMPLICIT REAL*8 (A-H,P-F,L-Z)
REAL*4 FR1S, FR2S, FR3S, ANSX, OMS
REAL*8 CLVB, CMVB, CMPAVB

1. DUMMY, PLOTOP
REAL FTVB, FPVB, FPPRVB, DIDRVB, XMVB, DELVB, XMUVB,
A FOVB, XMUAVR, ATOVB, ATCVB, ATSVG, ROVB, RVB, MV
C WOL, PST, UNIF
REAL ELSIG, DXI, REB, RDBR, FRZ, ARR, AMPLU, FREQU,
A ALPH, ALPH2, HEAVE, ARSF, FREOF, PHPH, RY1, DRY
B X, TEST, UPRIM, XU, YU, XL, YL, ER1, ER2, ER3, NDBR,
C RDBR
REAL SUM(8), YCLD(8), YNEW(8), DEL(3,3), CMPA(3), CL(3), G(3)
A Z, ZPR(3), SMALL(3), Y(3,3), YPR(3,3), GCAP(3,3)

COMMON /BL1/ NTIME, NDMC
COMMON /CLCMRL/ CLVB, CMVB, CMPAVB

COMMON /Z3/ Z(3)

COMMON /INPTVB/ FTVB(64), FPVB(64), FPPRVB(64), DIDRVB(64),
A XMVB(64), DELVB, XMUVB, FOVB, XMUAVB,
B ATOVB, ATCVB, ATSVG, ROVB, RVB(64), MV

COMMON /INPUTS/ NSBL, NZ, NOFF, NGAM, NSIG,
A NCOR, NCORD, LOWER, MOSTP, MAXT, MJTR,
B NOTBL, INDV, ELSIG, DXI, REB, RDBR,
C FRZ, ARR, AMPLU, FREQU, ALPH1, ALPH2, ALPH3,
D HEAVE, ARSF, FREOF, PHPH, NY, RY1,
E DRY, X(100), TEST, UPRIM, XU(30), YU(30),
F XL(30), YL(30), ER1, ER2, ER3, BDBR,
G RDBR

H, DUMMY(10), PLOTOP
DIMENSION DELTA(3,3)

DIMENSION ALPHA(3,3), BETA(3,3), GAMMA(3,3), OMS(3), OMEGA(3), C(3,3)

DIMENSION AA(10), AB(10), ANB(20), ANT(20), AAX(10), ANS(20), SORT(3)

1, NT(2)
CF4(X)=F*-B4+(B4*C6-C4)*X**X
Z1(X)=H*CF4(X)/B1**2+(CF4(X)*FR1S+1.*C6*X**1.*B2-F2)*X**X
Z2(X)=FZ/FRLS*FR1S*CF4(X)-F2+1.*C6*X**1.*FR(2-BZ)/FR1S)**X
S1(X)=(2.*HB*CF4(X)/7GB**2+(FR1S-FR2S)**X)**GA
S2(X)=(FR1S-FR2S)*GAM*X
FUna(X)=R1*Z1(X)-R2*Z1(X)**2+R1*S1(X)-R2*S1(X)*Z1(X)*S1(X)-Z1(X)**2
X=S2(X)

DATA BBS, REL, NPOL/1.1E-7, 1.1E-6, 3/
C
C MASSES AND H'S ARE NONDIMENSIONAL, WITH BLADE MASS AND RADIUS
C AS REFERENCES. NONROTATING NATURAL FREQUENCIES ARE
C DIMENSIONLESS, USING ROTOR SPEED AS REFERENCE. DISTANCES XBAB, SILB, SUPPL 43
C AND S2L ARE FRACTIONS OF SEMICHORD. XBAB, SILB, AND S2L ARE
C FRACTIONS OF ROTOR RADIUS.
C
NOMC=3
DO 69 K = 1, 8
69 SUM(K) = 0.
DO 69 K = 1, NVB
DO 66 K = 1, 8
66 YOLD(K) = YNEW(K)
CALL YB(YNEW, K)
IF (1 .LE. 1) GO TO 64
DO 67 K = 1, 8
67 SUM(K) = (YNEW(K) + YOLD(K)) * (RV1(K) - RV1(K-1)) / 2. + SUM(K)
69 CONTINUE

EM11 = SUM(1)
EM22 = SUM(2)
EM33 = SUM(3)
EM13 = SUM(4)
EM23 = SUM(5)
H11 = SUM(6)
H22 = SUM(7)
H33 = - EM33
H13 = - EM13
H23 = SUM(8)
BDBBR = BDBR / RDBR
BDS = BDBR ** 2
T11 = H11 * BDS
T22 = H22 * BDS
T33 = H33 * BDS
T13 = H13 * BDS
T23 = H23 * BDS
FR1S = BDS * EM11 * T11 / EM11
FR2S = EM22 * BDS * T22 / EM22
FR3S = EM33 * BDS * T33 / EM33
FR1 = DSQR(FR1S)
FR2 = DSQR(FR2S)
FR3 = DSQR(FR3S)
RATM = EM11 / EM22
ZETA = (1. + RATM) * (RATM * FR1S * T11 / FR1S + FR2S * T22 / FR2S + FR3S * T33 / FR3S)
RM = ZETA - 1.
SUM = FR1S * FR2S
HIGHS = SUMS * DSQR((SUMS**2 / SUMS + ZETA * FR1S * FR2S**2 / (RATM * FR1S + FR2S)**2))
SMAL = FR1S * FR2S / HIGHS
DEN = FR2S - FR1S
A1 = (HIGHS - FR1S) / DEN
A2 = 1. - A1
B = A1 * DEN / HIGHS
SLAM1 = EM11 * BDBR ** 2 / EM33
SLAM2 = A1 * SLAM1
SLAM3 = SLAM2 / A2
SUM3 = SUMS * FR3S
ADD3 = FR1S * FR2S * FR3S
ADD2 = FR1S * FR2S * FR3S
ADD1 = FR1S * FR2S * FR3S
ADD = FR1S * FR2S * FR3S
BRAK = 1. - (EM11 * T11 * EM22 * T22 * EM33 * T33 / EM11 - FR1S * EM22 - FR2S * EM33)
B4 = SUM3 + T2 * EM22 * T33 + EM11 * T11 * EM11 - FR1S * EM22 * T22 - FR2S * EM33
B2 = ADD * (EM11 * T11 * EM11 * EM22 * T22 * EM33)
B1 = ADD3 * (EM11 * T11 * EM11 * EM22 * T22 * EM33)
B0 = R4 / RBAR
B2 = ADD2 * (EM11 * T11 * EM11 * EM22 * T22 * EM33)
B1 = ADD3 * (EM11 * T11 * EM11 * EM22 * T22 * EM33)
B0 = RBAR / RBAR

61
\[ C_6 = EM_{11}A_1A_2 + EM_{22}A_2A_2 + EM_{33} \]

\[ F_4 = SUP_3 \]

\[ C_4 = FR_{25}E_{11}A_1A_2 + FR_{15}F_{10}EM_{22}A_2A_2 + EM_{33} \]

\[ GA = EM_{11}A_1 + EM_{33} \]

\[ GB = EM_{22}A_2 + EM_{33} \]

\[ F_2 = ADD_2 \]

\[ HA = EM_{11} + EM_{33} \]

\[ HB = EM_{22} + EM_{33} \]

\[ F_2 = ADD_2 \]

\[ R_1 = HA - HB \] *(GA/GR) **2

\[ R_2 = HA * (FR_{25} - FR_{15} - 1) \]

\[ ZLAM = F_4 + B_4 \]

\[ TWLAM = B_4 * C_6 - C_4 \]

\[ FZHAT = HB * (ZLAM/GB) **2 \]

\[ F2HAT = R_2 - F_2 + FR_{15}ZLAM + 2 * 7LAM * TWLAM * HB / GB ** 2 \]

\[ F4HAT = C_6 * R_2 + FR_{15}TWLAM + HB * (TWLAM/GB) ** 2 \]

\[ GZHAT = R_2 - F_2 + IF_{25} / FR_{15} * FR_{15} * ZLAM \]

\[ SIG2 = ZLAM * GA / GB ** 2 \]

\[ SIG2 = GA * (FR_{15} * FR_{25} + 2 * HB * TWLAM / GB ** 2) \]

\[ GAM2 = GA * (FR_{15} - FR_{25}) \]

\[ UZ = - R_2 * FZHAT \]

\[ U1 = R_1 * GZHAT + R_2 * F2HAT \]

\[ U2 = R_1 * GZHAT + R_2 * F4HAT \]

\[ U3 = - R_2 * SIG2 \]

\[ U4 = R_1 * GAM2 - R_2 * SIG2 \]

\[ U5 = SIG2 * GZHAT - GAM2 * FZHAT \]

\[ U6 = SIG2 * GAM2 + SIG2 * GZHAT + GAM2 * FZHAT \]

\[ U7 = SIG2 * GZHAT + GAM2 * F4HAT \]

\[ AAX(1) = UZ ** 2 \]

\[ AAX(2) = UZ * U1 + U3 * U5 \]

\[ AAX(3) = U1 ** 2 + U2 * U3 + U3 * U6 + U4 * U5 \]

\[ AAX(4) = U2 * U2 + U4 * U7 \]

\[ AAX(5) = U2 ** 2 + U4 * U7 \]

\[ CALL POLLY(4, RBS, REL, ANSX, AAX) \]

\[ XBAR = 1.25 \]

DO 86 I = 1, 4

IP = 2 * I

IM = IP - 1

IF (ABS(ANSX(IM)).GT.1.D-10) GO TO 86

IF (ANsx(IP).LE.0) GO TO 86

XBART = 0.7071

WRITE (16, 87)

FORMAT (99, 10X, *NO SOLUTION FOR XBAR*)

STOP

88 CONTINUE

15 ALow = (R1 * Z2(XBAR) - R2 * Z1(XBAR)) / (R1 * S2(XBAR) - R2 * S1(XBAR))

LOW = ALow * XBAR

BLOW = (CF(XBAR) - GA * ALow * XBAR) / (XBAR * GB)

XI = ALow * BLOW

ETA = (BLOW + A1 - ALOW * A2) / (A1 - A2)

S2L = FTA / (B * RTGHS)

62
71 11=2
    12=3
    GO TO 74
72 11=1
    12=3
    GO TO 74
73 11=1
    12=2
74 IF IOMS(11).GT.OMS(12)) GO TO 75
    MINI=1
    MIDI=12
    GO TO 76
75 MINI=12
    MIDI=11
76 SORT(1)=OMS(MINI)
    SORT(2)=OMS(MIDI)
    SORT(3)=OMS(MAXI)
    DO 77 1=1,3
    OMS(1)=SORT(1)
77 OMEGA(I)=DSQRT(CMS(I))
    DO 302 I=1,3
302 ALPHAI(I,1)=1.
    DENB=RETA(2,1)*BETA(3,2)-BETA(3,1)*BETA(2,2)-OMS(I1))
    ALPHA(1,2)=(BETA(1,2)*BETA(3,1)-BETA(3,2)*BETA(1,1)-OMS(I3)))/DENB
    ALPHA(1,3)=((-BETA(2,2)-OMS(1)) * (BETA(1,1)-OMS(1)) - BETA(1,2)*BETA(2,2))/SUPPL241
    1,11)/DENB
    CHK(1)=BETA(1,3)*ALPHA(1,1)*BETA(2,3)*ALPHA(1,2)*BETA(3,3)-OMS(I1))
SUPPL243
    1)*ALPHA(1,3))
SUPPL244
    1)*ALPHA(3,3)
SUPPL245
    1)*ALPHA(2,3)
SUPPL246
    1)*ALPHA(3,3)
SUPPL247
    1)*ALPHA(2,2)
SUPPL248
    1)*ALPHA(3,2)
SUPPL249
    1)*ALPHA(2,3)
SUPPL250
    1)*ALPHA(3,3)
SUPPL251
    1)*ALPHA(2,2)
SUPPL252
    1)*ALPHA(3,2)
SUPPL253
    1)*ALPHA(3,3)
SUPPL254
    WRITE(6,4888)
    WRITE(6,4899) (I,OMEGA(I),BETA(I,1),BETA(I,2),BETA(I,3),ALPHA(I,1))
SUPPL255
    (ALPHAI(2),ALPHAI(3,1),CHK(I),I=1,3)
    SORT(1)=1.
    SORT(2)=0.
78 DO 432 J=1,3
    GO TO 381,382,383, J
432 SORT(1)=0.
    SORT(2)=1.
    SORT(3)=0.
    GO TO 381
382 SORT(1)=0.
    SORT(2)=0.
    SORT(3)=1.
CALL ALSOL(3, DELTA, SRT, 3)
GO TO 384

384 DELTA(I,K) = ALPH(A(I,K))
WRITE(6,11) (I, GAMMA(I,1), GAMMA(I,2), GAMMA(I,3), I=1,3)
AMPLU = XMUAVR * (1. - ROV**3) / (1. - ROV**4) * 1.33333333330
SA = SMALS * SILA * RM * S2LB * HIGHS
SB = SMALS * SILA**2 * RM * S2LR**2 * HIGHS
DEL(1,1) = XMUAVR * (1. - ROV**4) / (4. * (1. - SLAMZ * XB4**2))
A * RROBR * E411
DEL(1,2) = 2. * SLAMZ * XBAB * DEL(1,1)
DEL(1,3) = (1. - SLAMZ * X3AB**2)
DEL(1,4) = A1 * (SLAMZ * XBAB * SB - SA)
DEL(1,5) = 2. * SLAMZ * XBAB * DEL(1,1)
DEL(1,6) = A2 / (SLAMZ * XBAB * SB - SA)
DEL(1,7) = AL * (SLAMZ * XBAB * SB - SA)
DEL(1,8) = AL / (SLAMZ * XBAB**2)
DEL(1,9) = A1 / (SLAMZ * XBAB**2)

50 SMALLG(I) = DEL(I,1) * CLVB * DEL(I,2) * CMPAVR

51 GCAP(I,J) = 0.
52 GCAP(I,J) = GCAP(I,J) + ALPH(I,J) * SMALLG(J)

51 Y(1,1) = Y(1,1) / CMS(I)

51 IF (PLOTPR == 0.) THEN
WRITE(6,90000) TO, X, TOPR, ZPR, Y, YPR, DEL, SMALLG
FORMAT(75/T0 = '1PE13.6', ' Z = '1PE13.6, ' TOPR = '1PE13.6,
* ZPR = '1PE13.6, ' YPR = '1PE13.6, ' SMALLG = '1PE13.6)
RETURN

ENTRY SUPPI (UNIT)

C
C
C
C
C
C
C
SUPPL308
CMPA(1) = 2. * CMPA(2) - CMPA(3)
SUPPL309
CL(3) = CL(2)
SUPPL310
CL(2) = CLVB
MAIN
CL(1) = 2. * CL(2) - CL(3)
SUPPL311
PSI = (BDAR / RDBR) * NTIME * DXI
SUPPL312
SIN PSI = SIN(PSI)
SUPPL313
COS PSI = COS(PSI)
SUPPL314
TOT(2) = TOT(1)
SUPPL315
TO = ATOVB + ATCVB * COS PSI + ATSVB * SIN PSI
SUPPL316
TOT(1) = TO - ATOVB
SUPPL317
DO 66 K = 1, 2
SUPPL318
DO 65 I = 1, 3
SUPPL319
64 SMALL G(I) = UINF **2 * (DEL(I,1) + CL(K) + DEL(I,2) * CMPA(K))
SUPPL320
A + DEL(I,3) * TOT(K)
SUPPL321
DO 65 I = 1, 3
SUPPL322
65 G CAP(I, K) = GCAP(I, K) + ALPHA(I,J) * SMALLG(J)
SUPPL323
DO 66 I = 1, 3
SUPPL324
Y(I,2) = Y(I,1)
SUPPL325
SUPPL326
SUPPL327
SUPPL328
SUPPL329
SUPPL330
SUPPL331
SUPPL332
SUPPL333
SUPPL334
SUPPL335
SUPPL336
SUPPL337
SUPPL338
SUPPL339
SUPPL340
SUPPL341
SUPPL342
SUPPL343
SUPPL344
SUPPL345
SUPPL346
SUPPL347
SUPPL348
SUPPL349
SUPPL350
IF (PLOTOP .LT., 0.)
SUPPL351
1 WRITE(6,9000) TO, Z, TOPR, ZPR, Y, YPR, DEL, SMALLG
SUPPL352
2, TOT
SUPPL353
RETURN
SUPPL354
1 FORMAT(5F10.4)
SUPPL355
2 FORMAT(5F10.4)
SUPPL356
3 FORMAT(I1L,10X,*ITERATION FOR XBAR DIVERGED*)
SUPPL357
4 FORMAT(I1L,5X,4HF1 =E13.5,5X,4HF2 =E13.5,5X,4HF3 =E13.5,5X,4HRM =E13.5,5X
SUPPL358
113.5,///)
SUPPL359
SUPPL360
113.5,5X,5HM23 =E13.5,///)
SUPPL361
SUPPL362
113.5,5X,5HY23 =E13.5,///)
SUPPL363
7 FORMAT(20X,6HX8/R =E13.5,10X,6HX8/R =E13.5,10X,6HX8/R =E13.5,10X,6SUPPL361
1H1/B =E13.5/720X, 6HL2/R =E13.5/10X, 6HL2/B =E13.5/9X, 7HK1/M1 =E13.5 SUPPL362
1/9X, 74K2/M2 =E13.5) SUPPL363
41 FORMAT(//10X, 5HRR =E13.5, 20X, 6HRR/R =E13.5) SUPPL364
44 FORMAT(1HI, 20X, 'POLYNOMIAL COEFFICIENTS' ///7X, 5HPower, 12X, 5+BLADE, SUPPL365
126X, 3H2-07) SUPPL366
46 FORMAT(10, 2030, 9) SUPPL367
47 FORMAT(1HI, 20X, 'ROOTS OF POLYNOMIALS' ///30X, 'BLADE', 60X, '2-1' //20X, SUPPL368
14HREAL, 21X, 4HIMAG, 31X, 6HREAL, 21X, 4HIMAG/) SUPPL369
49 FORMAT(2D25.9, 10X, 2D25.9) SUPPL370
11 FORMAT(///9X, 1HI, 15X, 10HGamma(I, 1), 15X, 10HGamma(I, 2), 15X, 10HGamma SUPPL371
1A(I, 3)/) SUPPL372
12 FORMAT(10, 3E25.5) SUPPL373
488 FORMAT(1HI, 8X, 1HI, 7X, 5Homega, 4X, 9Hbeta(I, 1), 4X, 9Hbeta(I, 2), 4X, 9Hbeta SUPPL374
1A(I, 3), 3X, 10Halpha(I, 1), 3X, 10Halpha(I, 2), 3X, 10Halpha(I, 3), 9X, 3HCHE SUPPL375
1K/) SUPPL376
489 FORMAT(10, 8E13.5) SUPPL377
721 FORMAT(77//10X, 5Hfr1 =E13.5/10X, 5Hfr2 =E13.5/10X, 5Hfr3 =E13.5/) SUPPL378
14HSA =E13.5/10X, 4HSB =E13.5//1) SUPPL379
END
SUPROUTINE SETUPS

IMPLICIT REAL*8 (A-H,O-Z)

REAL FTVB, FPVR, FPPVRV, DIDRVB, XMVB, DELV, XMUVB, 
A FOVR, XMUVR, ATOV, ATCV, ATSVB, ROVR, RVB, NVH 
REAL ELSIG, DXI, RF, ROBB, FRZ, ARR, AMPLU, FREQU, 
A ALPHI, ALPH2, HEAVE, AROF, FREQF, PHI, RY1, DRY, 
B Y, TEST, UPRIM, XU, YJ, XL, YL, ER1, ER2, ER3, ROBR, 
H CMPA, CMPAS, BAR, EMN, HVOR, SSPA, SVOR, TORF, XIVOR 
I, PLOTOP, PSILR, PSIUP

INTEGER TABLE(7,80)/560*

COMMON /BL1/ NTIME.

COMMON /INPTVB/ FTVB(64), FPVR(64), FPPVRV(64), DIDRVB(5), 
A XMVB(64), DELV, XMUVB, FOVR, XMUVR, 
B ATOV, ATCV, ATSVD, ROVR, RVB, NVH 
COMMON /INPUTS/ NSBL, NZ, NOFF, NGAM, NSIG, 
A NDOI, NDCOR, LOER, MSTOP, MXT, MTR, 
B NOTB, INDY, ELSIG, DXI, RER, ROBB, 
C FRZ, ARR, AMPLU, FREQU, ALPHI, ALPH2, 
D HEAVE, AROF, FREQF, PHI, RY1, 
E DRY, Y(100), TEST, UPRIM, XU(30), YU(30), 
F XL(30), YL(30), ER1, ER2, ER3, ROBR, 
G ROBR 
H CMPA, CMPAS, BAR, EMN, HVOR, NVOR, SSPA, SVOR, TORF, XIVOR 
I, PLOTOP, PSILR, PSIUP

J, MCUT

CALL WHERE(TABLE)
CALL ZERDIN

CALL SETUP('ALPHI', 1, ALPHI)
CALL SETUP('ALPHA1', 1, ALPHI)
CALL SETUP('ALPH2', 1, ALPH2)
CALL SETUP('ALPHA2', 1, ALPH2)
CALL SETUP('AMPLU', 1, AMPLU)
CALL SETUP('ARR', 1, ARR)
CALL SETUP('AROT', 1, AROF)
CALL SETUP('ATOV', 1, ATOV)
CALL SETUP('ATCV', 1, ATCV)
CALL SETUP('ATSV', 1, ATSV)
CALL SETUP('BARG', 1, BARG)
CALL SETUP('BDNB', 1, BDNR)
CALL SETUP('CMPS', 1, CMPA)

CALL SETUP('ALMA2', 949)
CALL SETUP('ARR', 949)
CALL SETUP('ARR1', 949)
CALL SETUP('ATCVB', 949)
CALL SETUP('ATSVB', 949)
CALL SETUP('BDNR', 949)
CALL SETUP('BARG', 949)
CALL SETUP('CMPS', 949)

 CALL SETUP('ALMA2', 949)
 CALL SETUP('ARR', 949)
 CALL SETUP('ARR1', 949)
 CALL SETUP('ATCVB', 949)
 CALL SETUP('ATSVB', 949)
 CALL SETUP('BDNR', 949)
 CALL SETUP('BARG', 949)
 CALL SETUP('CMPS', 949)

68
SUBROUTINE BLC(TX,Y,MST,MEND,NY,RX,DXI,RFR,JFR,IM,FLAM,XFLAM,TESTLC)
1T,U,SCALE,H,UC,V,SEH,USEH,DISS,THET,S,LSTER,AMOG,MSF,EP,UC,USA,SCARLC
ILS,NIT,NTIM,NTC30,XTX,NZ,NOUT)

C PROGRAM FOR ANALYZING LAMINAR AND TURBULENT BOUNDARY LAYERS
C BY THE METHOD OF FINITE DIFFERENCES. IF THE INTEGER LAMQ
C IS GREATER THAN ZERO, THE BOUNDARY LAYER IS LAMINAR.
C
COMM /RAN/  NODUMMY, NDINC, ISTD
DIMENSION USB(100,300),SCAL(300)
DIMENSION X(300),Y(100),UC(100,3),V(100,2),XC(300)
DIMENSION S(100),SEH(100),SF(100),VISC(100,2),GRAD(100)
DIMENSION A(100),R(100),C(100),Q(100),F(100)
DIMENSION ALPHA(100),RETA(100),GAMMA(100),DELTA(100)
DIMENSION SCALE(300,2),VAR1(100),VAR2(100)
DIMENSION FLAM(10),XFLAM(10),YR1(100),YR2(100)
DIMENSION U(300,10,2)
DIMENSION CAP(100),CAPH(100),CAPJ(100),CAPX(100)
DOUBLE PRECISION AP(100),BP(100),CP(100),DP(100),FP(100),UP(100)
FORMAT(10,41X,36H ANALYSIS OF LAMINAR BOUNDARY LAYER///51X,12HTILAC
1ME STEP NO13//51X,12HTILAC NO13///4X,1HM,8X,1HX,13X,2HX,12X,2RLC
1HM,10X,6H-DP/DX,9X,5HDELTA,9X,5HTHETA,9X,5HSHEA///)
FORMAT(11,41X,36H ANALYSIS OF TURBULENT BOUNDARY LAYER///51X,12HTILAC
1ME STEP NO13//51X,12HTILAC NO13///4X,1HM,8X,1HX,13X,2HX,12X,2RLC
1HM,10X,6H-DP/DX,9X,5HDELTA,9X,5HTHETA,9X,5HSHEA///)
3 'T'

FORMAT(15,8E14.4,13)
FORMAT(14H1,2X,3HM =14//2X,3HX =E14.5//2X, 4HUE =E14.5//10X,17+ (1/RBLAC
1H10TP/DX) =E14.5,10X,5HREB =E14.5,10X,4HJ =E14.5//)
FORMAT(2X,25HPHYSICAL DELTA =E14.5,8X,12HDELTA STAR =E14.5,8X,12HDELTA
15,8X,7THETA =E14.5,772X,25HTHETA TRANSFORMED DELTA =E14.5,8X,12HDELTA
ILTA STAR =E14.5,8X,7THETA =E14.5//)
FORMAT(22X,5E20.5)
FORMAT(730X,1HSEPARATION AT X =E13.5,6H, XC =E13.5)
FORMAT(40X,12HWEAR =E14.5//)
FORMAT(750X,1HTHREE HARD WALL AT X =E14.5)
FORMAT(20X,35HSCALE CHANGE - Y-MAX INCREASED FROM 12.4,3H DE12.4RLC
77)
FORMAT(10X,7HSTEPS,22H, THE WALL GRADIENT IS 12.4)
BCON = 1.57075X
FCON = 1.0/(2.*DXI)
IF(ISTD.NE. 0) GO TO 900
DXI=1.0
BCON = 0.
FCON = 0.

CONTINUE
MOUT =6
MTRAN=1
YSUR=2*Y(12)
MST=2
MST1=MST-1
MSTL=MST-1
NOUTL = NOUT +1
MSTLMD = MOD(MST1, NOUTL)
MAXIT=0
GO TO (543,550),LIMER
543 IF(LAMQ) 544,544,545
544 WRITE(MOUT,11) NTIME,NITS
GO TO 550
545 WRITE(MOUT,10) NTIME,NITS
550 CONTINUE
       YTR = SQRT(RER)
       UC(1,1) = 0.
       V(1,1) = 0.
       NV = NY - 2
       NV1 = NV + 1
       CALL YDIFF(NY,ALPHA,BETA,GAMMA,DELTA,SD,SE,SF,C2,C3,C4,Y)
       DO 41 N=1,NV1
           VISCN,1 = 1.
        41 VISCN,2 = 1.
       DO 42 M=MST2,MST1
           L = MST1-M+2
           DO 50 N=1,NV
           GRAD(N+1) = SD(N+1)*UC(N+2,L)+SE(N+1)*UC(N+1,L)-SF(N+1)*UC(N,L)
           GRAD1 = C2*UC(2,L)+C3*UC(3,L)+C4*UC(4,L)
           MM=M-1
           CALL PGRAD(MM,X,Y,DXI,PRESS,SA,SB,SC,SR,SS)
           DO 456 N=1,NY
           CALL SETY(LAMQ,M,NV,REB,X,Y,UC,PRESS,GRAD,DELT,DISP,THETA,VISC,MTBL3)
        456 I=AIN
       42 CONTINUE
          MEND1 = MEND - 1
          GRADS=GRAD1
          GRADSS=GRAD1
   C THE MAIN CALCULATION STARTS HERE.
   C
       DO 99 M=MST1,MEND1
           ITER=0.
           WALLG=0.
           MPI=M+1
           DELTP = DELT/YTR
           DISPT = DISP*YTR
           THEYT = THEYT+YTR
           SHEAR = GRAD1/YTR
           IF (MOD(M,NOUT1).NE. MSTIMD) GO TO 225
       GO TO (561,562),LOWER
   561 WRITE(MOUT,12) M,X(M),XC(M),UE(M,1),PRESS,DELTP,DISP,THETA,SHEAR
       1, MAXIT
           GO TO 225
   562 WRITE(MOUT,20) M,X(M),UE(M,1),PRESS,REB,UPRIM
       WRITE(MOUT,24) DELTP,DISP,THETA,DELT,DISP,THEYT
       WRITE(MOUT,21)
       WRITE(MOUT,22) (YIN,NY,UC(N,2),VIN,N1,GRAD(N),VISC(N,1),N=1,NVP1)
       WRITE(MOUT,25) SHEAR
   225 IF (GRADSS-GRADS-1.E-6) 229,229,408
       408 XSY=M(X-2)+M(X-1)-X(M-2)*GRADSS/1GRADSS-GRADS)
       IF (XSY-X(M)) 409,409,229
       72
4CS WFS = (XSX - X(M-1)) / (X(M) - X(M-1))
GO TO 224
229 IF (GRAD(1)) 227, 227, 273
273 IF (NISP, GT. 0, * AND, THETA, GT. 0, *) GO TO 273
283 CONTINUE
XSFP = XC(M-1)
USEP = UE(M-1)
XRL = X(M-1)
WRITE (MOUT,23) XBL, XSEP
RETURN
227 WFS = GRADS / (GRADS - GRAD(1))
224 WFSL = 1. - WFS
XSEP = WFSL * XC(M-1) + WFS * XC(M)
XRL = WFSL * X(M-1) + WFS * X(M)
USEP = WFSL * UE(M-1) + WFS * UE(M, 1)
WFP = (XRL = X(X-2)) / (X(M-1) - X(M-2))
DP1 = 1. - WFP
DISS = DISS * WFP1 + DISS * WFP
THET5 = THET5 * WFP1 + THET5 * WFP
WRITE (MOUT, 23) XBL, XSEP
IF (LAMQ, EQ. 0, AND, M. LT. MTRAN+5) LAMQ = 1
GO TO 222
223 CONTINUE
IF (NOTBL, EQ. 2, AND, NITS, GT. 1, AND, M, GT, NZ, AND, 1, XC(M), GT, XTEST) GO TO 283
IF (LAMQ) 801, 801, 902
802 CALL TRANS (UPRIM, PRESS, THETA, REB, UC, NY, FLAM, XFLAM, LAMQ)
803 WRITE (MOUT, 30) X(N)
801 CONTINUE
IF (Y(NV) - DELT) 620, 641, 641
620 RY = RY + DRY
C
C RECALCULATION STARTS HERE.
C
DO 632 N = 1, NY
YBL(N) = Y(N)
VAR1(N) = UC(N, 2)
632 VAR2(N) = UC(N, 3)
CALL YSET (RY, YSUB2, NY, Y)
WRITE (MOUT, 35) YBL(NY), Y(NY)
DO 633 N = 2, NVP1
YIN = Y(NI)
CALL TERP (YIN, YBL, VAR1, NY, UPAS1)
UC(N, 2) = UPAS1
CALL TERP (YIN, YBL, VAR2, NY, UPAS2)
633 UC(N, 3) = UPAS2
CALL YDIFF(NY, ALPHA, BETA, GAMMA, DELTA, SD, SE, SF, C2, C3, C4, Y)
IF (LAMQ) 700, 700, 701
700 DO 635 N = 2, NVP1
VAR1(N) = VISC(N+1)
635 VAR2(N) = VISC(N+2)
DO 636 N = 2, NVP1
73
YIN = Y(N)
CALL TERP(YIN,YB1,VAR1,NV1,UPAS1)
VISC(N+1) = UPAS1
CALL TERP(YIN,YB1,VAR2,NV1,UPAS2)
636 VISC(N+2) = UPAS2
7C1 DO 637 N=2,NV1
VAR1(N) = Y(N-1)
637 VAR2(N) = Y(N+1)
DO 638 N=2,NV1
YIN = Y(N)
CALL TERP(YIN,YB1,VAR1,NV1,UPAS1)
V(N+1) = UPAS1
CALL TERP(YIN,YB1,VAR2,NV1,UPAS2)
638 V(N+2) = UPAS2
641 CONTINUE
C
C RESCALING CALCULATION ENDS HERE.
C CALL PGRAD(MX,UE,DXI,PRESS,SA,SB,SC,SR,SS)
C
C RECURSION RELATIONS ARE SET UP HERE.
C
IF (ISTD.EQ.1) GO TO 820
IF (SCALE(M+1,1)-1.) LACU=1
521 IF (SCALE(M-1,2)-1.) LACU=2
FACU1=UE(M+1,2)/UE(M+1,1)
522 FACU2=UE(M-1,1)/UE(M+1,1)
GO TO 820
523 LACU=2
DO 610 N=1,NV1
VAR1(NN) = U(M+1,NN)
610 VAR2(NN) = U(M+1,NN)
CALL YSET(SCALE(M+1,1),YSUB2,NV1)
CALL YSET(SCALE(M+1,2),YSUB2,NV2)
820 DO 88 N=2,NV
CALL CAPSTER(N,CAPG,CAPR,CAPJ,CAPK,SR,SS,SD,SE,SF,VISC,V,UC)
A(N) = SF(N) *CAPG(N) - DELTA(N) *CAPH(N) + SF(N) *CAPJ(N)
B(N) = BCON*SA*CAPK(N)+SF(N)*CAPG(N)-GAMMA(N)*CAPH(N)-SE(N)*CAPJ(N)
C(N) = SD(N) *CAPG(N) - BETA(N) *CAPH(N) - SD(N) *CAPJ(N)
DT(N) = -ALPHA(N)*CAPH(N)
IF (ISTD.EQ.1) GO TO 576
GO TO (574,575), LACU
574 UPAS1=FACU1*UC(N+1)
575 YIN = Y(N)
CALL TERP(YIN,YB1,VAR1,NV1,UPAS1)
CALL TERP(YIN,YB2,VAR2,NV1,UPAS2)
576 F(N) = PRESS+FCON*(4.*UPAS1-UPAS2)*CAPK(N)*(SB*UC(N-2)-SC*UC(N,3))
88 CONTINUE
C
C SOLUTION FOR VELOCITY PROFILE STARTS HERE.
C
DO 89 N=2,NV
AP(N) = A(N)
BP(N) = B(N)
CP(N) = C(N)
DP(N) = D(N)

89
FP(N) = F(N)
DO 77 N=2,NVML
CP(N) = CP(N)/BP(N)
DP(N) = DP(N)/BP(N)
FP(N) = FP(N)/BP(N)
RP(N+1) = BP(N+1) - CP(N)*AP(N+1)
CP(N+1) = CP(N+1) - DP(N)*AP(N+1)

77
FP(N+1) = FP(N+1) - FP(N)*AP(N+1)
UP(NY) = UE(N+1,1)
UP(NV1) = UP(NV)
UP(NV) = (FP(NV) - UP(NV) * (DP(NV) + CP(NV1))/RP(NV))
DO 56 N=3,NV
NN=NV+2-N
UP(NN) = FP(NN) - DP(NN)*UP(NN+2) - CP(NN)*UP(NN+1)
DO 65 N=2,NY
UC(N,1) = UP(N)
IF (ITER) 843,841,843

841
DO 842 N=2,NVPI
VIN(N,2) = VIN(N,1)

842
VISC(N,2)=VISC(N,1)
DISS=DISS
DISS=DISS
THETS=THETS
THETS=THETS
GRADSS=GRADSS
GRADS=GRAD(1)

843
DO 55 N=2,NVPI
VIN(N,1) = VIN(N-1,1) + S*(Y(N)-Y(N-1))*SA*UC(N,1) + UC(N-1,1) - 3*UC(N,1)
IN(N,2) = UC(N-1,2) + UC(N,3) + UC(N-1,3)
DO 56 N=1,NV

56
GRAD(N+1) = SD(N+1) + UC(N+2,1) + UC(N+1,1) - SF(N+1) + UC(N,1)
GRAD(1) = C2*UC(2,1) + C3*UC(3,1) + C4*UC(4,1)
CALL SFITIT(LAMQ, NVPI, NV, REB, X, UC, PRESS, GRAD, DELT, DISP, THETA, VISC, BL)
IMTRAN

ITER=ITER+1
GO TO (330, 809), LOWER

809
WRITE (MOUT, 810) ITER, GRAD(1)
83C IF (ITER=9) 811, 811, 812
811 FPW=ABS(GRAD(1)-WALLG)
IF (WALLG<1.) 120, 120, 119
119 EPW=EPW/WALLG
120 IF (EPW-TEST) 812, 814, 814
814 WALLG=GRAD(1)
GO TO 820

812 DO 44 N=1,NY
UC(N,3) = UC(N,2)
UC(N,2) = UC(N,1)
CONTINUE

MAXIT=ITER
IF (ISTD.EQ.1) GO TO 99
DO 48 N=1,NY
```
48 USAV(N+1,N) = UC(N+1)
SCALS(N+1) = RY
99 CONTINUE
XSEP = 1.1
USEP = UE(MX, 1)
222 CONTINUE
RETURN
END
```
SOURCE: PLTSR(PLOTOP, P, L)
REAL*8, ORD(6)
DIMENSION P(200, 7), TIT(56)
N.F.(5, 4)

1, NFP(6)
DATA NL, N2, NO, N42
1 / 1, 2, 0, 42 /
DATA ORD', 'THETA-P', 'TORS', 'FLAP-H', 'BEND-H',
1 'CL', 'CM-A' /
IF(PLOTOP. EQ. 0.) RETURN
IF (L = L.T. 2) RETURN
IF (PLOTOP. EQ. 2.) GO TO 2

PLOT.P = 2,
CALL IFRMV( 'CRIMI-PETE', '30', '5100' )
CONTINUE
3
NL = 1
DO 1 J = 1, 6
CALL EXPLOT(9, N1, N1, P, P(J+1), L, -N1, N2
1, N42, 1, 12, 'PSI-DEGREES', 8, ORD(J)
2, N1, N1, XL, XU, N1, YL, YU, N1, NO, NL)
1 CONTINUE
NFP(1) = -1
NFP(2) = 66
NFP(3) = 50
NFP(4) = 50
NFP(5) = 680
CALL EXPLOT(9, N1, N1, P, P(1, 2), L, -N1, N2
1, N42, 1, 12, 'PSI-DEGREES', 8, ORD(1)
2, NFP, N1, XL, XU, N1, YL, YU, N1, NO, N1)
NFP(1) = -2
NFP(2) = 66
NFP(3) = 350
NFP(4) = 380
CALL EXPLOT(9, N1, N1, P, P(1, 6), L, -N1, N2
1, N42, 1, 12, 'PSI-DEGREES', 8, ORD(5)
2, NFP, N1, XL, XU, N1, YL, YU, N1, NO, N1)
NFP(1) = 50
NFP(2) = 690
NFP(3) = 40
CALL EXPLOT(9, N1, N1, P, P(1, 7), L, -N1, N2
1, N42, 1, 12, 'PSI-DEGREES', 8, ORD(6)
2, NFP, N1, XL, XU, N1, YL, YU, N1, NO, N1)
NFP(1) = -1
NFP(2) = 50
NFP(3) = 50
NFP(4) = 50
NFP(5) = 690
CALL EXPLOT(9, N1, N1, P, P(1, 3), L, -N1, N2
1, N42, 1, 12, 'PSI-DEGREES', 8, ORD(2)
2, NFP, N1, XL, XJ, N1, YL, YU, N1, NO, N1)
NFP(1) = -2
NFP(2) = 66
NFP(3) = 350
NFP(4) = 380
CALL EXPLOT(9, N1, N1, P, P(1, 4), L, -N1, N2
1, N42, 1, 12, 'PSI-DEGREES', 8, ORD(3)
2, NFP, N1, XL, XJ, N1, YL, YU, N1, NO, N1)
2, NFP, N1, XL, XU, N1, YL, YU, N1, N0, N1)
  NFP(2) = 50
  NFP(4) = 690
  NFP(5) = 43
CALL EZPL0T(9, N1, N1, P, P(1, 5), L, -N1, N2
1, N42, 1, 12, 8, ORD(4)
2, NFP, N1, XL, XU, N1, YL, YU, N1, N0, N1)
  RETURN
  END
SUBROUTINE STAG(NX, NY, MSTOP, MST, DXI, RY, DRY, X, Y, UE, UC, V, USAV, SCALS, STAG)

C PROGRAM FOR CALCULATING THE BOUNDARY LAYER PROFILE NEAR
C THE STAGNATION POINT
C
COMM N /H11/ YTIME, ADIMC, ISTD
DIMENSION USAV(300,100), SCALS(300)
DIMENSION PHI7(24), PHIP(24), FTAP(24)
DIMENSION X(300), Y(100), UE(300,3), UC(100,3), V(100,2)
DIMENSION FF(100), EFPI(100)
DATA FTAP /9,2,4,6,8,11,1,2,1,4,1,6,1,8,2,2,1,2,2,2,2,6,2,8,3,20
13,3,3,4,4,4,4,4,4,6/
DATA PHIP /9,0,023,0,381,0,1867,0,3124,0,4592,0,622,0,7967,0,9793,1,1164/
19,1,362,1,5578,1,7553,1,9538,2,153,2,3,526,2,5523,2,7522,2,9521,3,1/
1521,3,3521,3,5521,3,7521,3,9521/
DATA PHIF /0,0,2265,0,4145,0,5663,0,6859,0,7779,0,8467,0,8968,0,9231,0,956/
18,0,9732,0,9839,0,9935,0,9946,0,9971,0,9984,0,9992,0,9996,0,9998,0,9999,1,1/
1,1,1,1/
RAG = 0.08
IF (ISTPF) 10, 10, 5
5 RAG = 3
10 IF (EF(I)) 10, 10, 5
20 EF(I) = 0.
22 DO 20 M = 1, MX
23 IF (UF(M, 1)) 20, 20, 19
25 MSP = M
27 GO TO 21
20 CONTINUE
21 ASTAG = (UF(MSP + 2, 1) - UF(MSP + 1, 1)) / (X(MSP + 2) - X(MSP + 1))
29 IF (ASTAG) 22, 22, 23
22 ASTAG = (UF(MSP, 1) - UF(MSP - 1, 1)) / (X(MSP) - X(MSP - 1))
26 SQAS = SQAS(AFSTAG)
28 DELT = 2.675 * SQAS
31 IF (DELTA - YNY(3)) 31, 31, 310
30 RY = RY + DRY
34 CALL YSF7(RY, Y(2), NY, Y)
38 GO TO 309
31 CONTINUE
36 DO 30 N = 2, NY
38 YET = Y(N) * SQAS
40 DO 33 NN = 1, 24
42 IF (Y(N) - ETAP(NN)) 408, 408, 33
41 40 MARK = NN
45 GO TO 410
33 CONTINUE
37 FFP(N) = YET - 6479
39 FFP(N) = 1.
43 GO TO 80
40 FRAC7 = (YET - ETAP(MARK - 1)) / ETAP(MARK) - ETAP(MARK - 1)
48 FRAC1 = 1. - FRAC7
50 FFP(N) = PHITZ(MARK - 1) * FRAC1 * PHIZ(MARK) * FRAC7
52 FFP(N) = PHIP(MARK - 1) * FRAC1 * PHIP(MARK) * FRAC7
54 80 CONTINUE
58 M1 = MSP - MSTOP
60 M2 = MSP + MSTOP

79
M=M1-1
M=M+1
MST=M+1
SCAL5(M)=RY
DO 71 N=1, N
UC(N,2) = UC(N+2)
UC(N,2) = UE(M,1)*EFF(N)
V(N,2) = V(N,1)
V(N,1) = -SQAS*EFF(N)
IF(ISTD.EQ. 1) GO TO 71
USAV(N,N)=UC(N,2)
CONTINUE
IF(M-M2) 50, 55, 55
55 IF(UF(M,1)-BAG) 50, 50, 81
81 CONTINUE
RETURN
END
SURROGATE ATTPR(PREC, XSIG, NSIG, ASZ, AS, AR, CMAT, RMAT, NGAM, NF, ACAP, THICK, RATTPR)
DIMENSION XSIG(100), ASZ(30), AS(30, 30), AR(30), ACAP(100, 3)
DIMENSION ACAP(30, 3), THICK(24), GAMMA(1000)
DECIMAL PRECISION CMAT(50, 50), RMAT(130)
PI = 3.14159
NGAM = NGAM + 1
DO 50 M = 1, NGAM
  CMAT(M, 1) = ASZ(M)
  RMAT(M) = AR(M)
  DC 25 N = 1, NGAM
  CMAT(M, N + 1) = AS(M, N)
  50 CONTINUE
  CALL ALSOL(NGAM, CMAT, RMAT)
  DO 75 M = 1, NGAM
  75 ACAP(M, 1) = RMAT(M)
  GAMMA(L) = GAMMA(ACAP, DXI, PI)
  SAVE = XSIG(NSIG + 1)
  XSIG(NSIG + 1) = 2.
  CALL CPC(0, NGAM, NF, XSIG, NSIG, XSIG, NSIG, XSIG, NSIG, ACAP, BCAP, THICK, RATTPR)
  100 BB = GAMMA, UINF, UDQ, 1, SAVE, DXI, PREC
  XSIG(NSIG + 1) = SAVE
RETURN
END
SUBROUTINE UNPOP(NGAM, AR, ALAM, AFACT, RMAT, CMAT, XGAM, AS, ACAP, MX, NZ, NUNPOP)

IF, XSIG, RCAP, THICK, RDBB, UINF, XC, UF)

DIMENSION AR(30), ALAM(30), XGAM(30), AS(30, 30), ACAP(30, 31), XSIG(130), UNPOP

IRCAP(100, 3), THICK(24), XC(300), JEL(300, 3)

DOUBLE PRECISION RMAT(130), CMAT(60, 60)

NGP1=NGAM+1

DO 5 M=1, NGP1

SUB=AR(M)-ALAM(M)*AFAC1/3.

RMAT(M)=SUB

CMAT(M, 11)=1.

CMAT(M, 2)=XGAM(M)

DO 5 N=2, NGAM

CMAT(M, N+1)=AS(M, N)

CALL ALSCL(NGP1, CMAT, RMAT)

DO 10 N=1, NGP1

ACAP(N, 1)=RMAT(N)

DO 15 M=1, MX

SIGN=1.

IF(M-N2) 12, 14, 14

SIGN=-SIGN

12 CONTINUE

CALL QECAL(0, NGAM, NGAM, NF, XSIG, ACAP, BCAP, THICK, RDBB, OF, UINF, XC(M), UNPOP)

1UE(M, 11), SIGN)

15 CONTINUE

RETURN

END
SUBROUTINE ALSL(NL, C, R)
 DOBLE PRECISION C(NDIMC,NDIMC), R(130)
 DOUBLE PRECISION CMAX,SAVE, SUM
 COMMON / RL1/ NTIL, NTIME, NDIMC
 NTIL = NT-1
 DC 99 J=1,NTIL
 CMAX = C(NL,J)
 L=NT
 DC 10 I=J,NTIL
 IF (DARS(CMAX)-DARS(C(I,J))) 5,10,10
 CMAX = C(I,J)
 L=I
 5 CONTINUE
 DC 15 JJ=J,NT
 SAVE = C(L, JJ)
 C(L, JJ) = C(I, JJ)
 15 C(I, JJ) = SAVE/CMAX
 SAVE = R(L)
 R(L) = R(J)
 R(J) = SAVE/CMAX
 JPI = J+1
 DO 25 I=JPI,NT
 20 JJ=JPI,NT
 C(I, JJ) = C(I, JJ) - C(I, J)*C(I, JJ)
 25 R(I) = R(I) - R(J)*C(I, J)
 CONTINUE
 R(NT) = R(NT)/C(NT, NT)
 DO 150 K=1,NTIL
 1 I=NT-K
 150 RETURN
 END
SUBROUTINE CPC(ISEP, NGAM, NF, XSIG, NSIG, XSIGA, NSIGA, XSIGB, NSIGB, ACAP, CPC)
1, BCAP, THICK, ROBB, GAMAW, UINF, UDOT, SIGN, XC, DXI, CP)
DIMENSION XSIG(100), XSIGA(100), XSIGB(100), ACAP(30, 3), BCAP(100, 3)
DIMENSION GAMAW(1000), THICK(24)
THETA = ARCTHINC
RECIIP = 1./(UINF*UINF)
SUM = 0.
ANGLE = 0.
DC = 5 N = 1, NF
ANGLE = ANGLE + THETA
5
SUM = SUM + THICK(N) * COS(ANGLE)
CP = UDOT*RECIIP*(THICK(1)+2.0* (1.-XC) *SUM)
CALL QECAL (ISEP, NGAM, NSIG, NF, XSIG, ACAP, BCAP, THICK, RCRR, GAMAW(1), UICPC
1NF, XC, U, SIGN)
CP = CP + 2.0 *(SIGN*U/UINF-1.)
CALL EQAMI (1, NGAM, ACAP, BCAP(1, 1), XSIG(1), XSIG(1)*SIG+1), GAMAW(1, XC, CPC
1VAL1)
CALL EQAMI (2, NGAM, ACAP, RCAP(1, 2), XSIGA(1), XSIGB(1)*SIGA+1), GAMAW(2, CPC
1XC, VAL2)
CALL EQAMI (3, NGAM, ACAP, BCAP(1, 3), XSIGB(1), XSIGB(1)*SIGB+1), GAMAW(3, CPC
1XC, VAL3)
CP = CP + SIGN*RECIIP*(1.5*VAL1-2.*VAL2+.5*VAL3)/DXI
10
IF (ISEP) 20, 20, 10
CALL FSGI (1, NSIG, XSIG, BCAP, XC, VAL1)
CALL FSGI (2, NSIGA, XSIGA, BCAP, XC, VAL2)
CALL FSGI (3, NSIGB, XSIGB, BCAP, XC, VAL3)
CP = CP + RECIIP*(1.5*VAL1-2.*VAL2+.5*VAL3)/DXI
20
CP = CP
RETURN
END
SUBROUTINE CLCM(NC1,IESP,NGAM,XSIG,NSIG,XSIGA,NSIGA,ACAP,BCAP,THICK,RDRE,GAMAU,UINF,YDOT,DXI,AROT,CMXR)
COMMON /CLCMAL/ CLVR, CMVR, CMPAR
DIMENSION ARG(21), ARG4(121)
DIMENSION GAMAY(1000),THICK(24)
DIMENSION XSIGA(100),XSIG(100),ACAP(30,3),HCAP(100,3)
FORMAT(//40X,4HCL =E13.5/40X,4HCM =E13.5,17H (ABOUT MIDCHORD))/40X,CLU
HCM =E13.5,24H (ABOUT PITCH AXIS - A =F7.4,1H)
NCLT=6
SAVE=THICK(1)
THICK(1)=0.
D=3.14159/FLOAT(NC01)
CL=0.
C=0.
XI=1.
ANGLE=0.
FLI=0.
FM1=0.
IF(IESP) 5,5,7
XATX=XSIG(NSIG+1)
IF(ABS(XATX) .LE. 95) 8,5,5
XATX=XATX+.5
8 XATX=.5*(XAT+XATX)
XAT=.5*(XAT+XATX)
C2=CL+XAT
C1=.5*(1.+XAT)
C1P=.5*(1.-XAT)
C2P=C1+XAP
DC 10 I=1,NC01
ANGLE=ANGLE+DT
XIPI=CI*COS(ANGLE)+C2
CALL CPCITSEP,NGAM,1,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,BCAP,TCLCM
THICK,RDRE,GAMAU,UINF,YDOT,DXI,CMXR
CALL CPCITSEP,NGAM,1,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,BCAP,TCLCM
THICK,RDRE,GAMAU,UINF,YDOT,DXI,CMXR
IF(MV)CPL-CPU
FMPI=XIPI*FLIP1
CL=CL+XIPI*XAT*(FLIP1+FLI)
CM=CM+XIPI*(FLIP1+FM1)
XI=XIPI
FLI=FLIP1
FM1=FMPI
XI=1.
FLI=0.
FM1=0.
ANGLE=0.
DC 15 I=1,NC01
ANGLE=ANGLE+DT
XIPI=CL+CM*COS(ANGLE)+C2
CALL CPCITSEP,NGAM,1,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,3CAP,TCLCM
THICK,ROBL,GAMAU,UINF,YDOT,DXI,CMXR
CALL CPCITSEP,NGAM,1,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,ACAP,TCLCM
THICK,ROBL,GAMAU,UINF,YDOT,-1.,XIPI,DXI,CMXR
FLIP1=CPL-CPU
FMPI=XIPI*FLIP1
CL=CL+XIPI*XAT*(FLIP1+FLI)
CLCM 7
CLCM 8
CLCM 9
CLCM 10
CLCM 11
CLCM 12
CLCM 13
CLCM 14
CLCM 15
CLCM 16
CLCM 17
CLCM 18
CLCM 19
CLCM 20
CLCM 21
CLCM 22
CLCM 23
CLCM 24
CLCM 25
CLCM 26
CLCM 27
CLCM 28
CLCM 29
CLCM 30
CLCM 31
CLCM 32
CLCM 33
CLCM 34
CLCM 35
CLCM 36
CLCM 37
CLCM 38
CLCM 39
CLCM 40
CLCM 41
CLCM 42
CLCM 43
CLCM 44
CLCM 45
CLCM 46
CLCM 47
CLCM 48
CLCM 49
CLCM 50
CLCM 51
CLCM 52
CLCM 53
CLCM 54
C = C - (XIPL - XI) * (FMIPL + FMI)  
XI = XIPL  
FMI = FMIPL  
FLI = FLIPL  

15  
FMI = FMIPL  
XIPL = XA  
DO 16 I = 1, 21  
CALL CPC(I), SEP, NGAM, 1, XSIG, NSIG, XSIGA, NSIGA, XSIGB, NSIGB, ACAP, BCAP, TCLCM  
IHCk, RDBR, GAMAW, UINF, UDOT, 1, 0, XIPL, DXI, CPU)  
CALL CPC(I), SEP, NGAM, 1, XSIG, NSIG, XSIGA, NSIGA, XSIGB, NSIGB, ACAP, BCAP, TCLCM  
IHCk, RDBR, GAMAW, UINF, UDOT, -1, XIPL, DXI, CPU  
ARGL1 = CPL - CPU  
ARGL = XIPL * ARGL(I)  

16  
XIPL = XIPL * 0.00125  
SUML = 0  
SUMM = 0  
DO 17 I = 1, 19, 2  
SUML = SUML + 2 * ARGL1 + 4 * ARGL1 + 1  
SUMM = SUMM + 2 * ARGM1 + 4 * ARGM1 + 1  
CL = CL + 0.333333E-3 * (SUML + ARGL1 - ARGL1)  
CM = CM + XATT * RC(AVR)  
RCON = 16 * RAPC(I, 1) * SQRT(5. E - 4 * (XATT * XSIGA(I))) / UINF  
CL = CL + RCON  
C = C + XATT * RCON  
GO TO 130  

5  
DO 99 I = 1, 191  
ANGLE = ANGLE + DT  
XIPL = COS(ANGLE)  
CALL CPC(I), SEP, NGAM, 1, XSIG, NSIG, XSIGA, NSIGA, XSIGB, NSIGB, ACAP, BCAP, TCLCM  
IHCk, RDBR, GAMAW, UINF, UDOT, 1, 0, XIPL, DXI, CPU)  
CALL CPC(I), SEP, NGAM, 1, XSIG, NSIG, XSIGA, NSIGA, XSIGB, NSIGB, ACAP, BCAP, TCLCM  
IHCk, RDBR, GAMAW, UINF, UDOT, -1, XIPL, DXI, CPU  
FLIPL = CPL - CPU  

99  
FMIPL = FMIPL  
CL = CL + (XIPL - XI) * (FLIPL + FLI)  
C = C + (XIPL - XI) * (FLIPL + FLI)  
XI = XIPL  
FLI = FLIPL  

99  
CMPA = CM + AROT * CL * 5  
WRITE(MOUT, 4) CL, CM, CMPA, AROT  
THICK(I) = SAVE  
CLVR = CL  
CMVR = CM  
CMPAVR = CMPA  
RETURN
END
SUBROUTINE QECAL(ISEP, NGAM, NSIG, NF, XSIG, ACAP, RCAP, THICK, RDHA, GAMMAQECAL
1, UINF, XC, U, SIG)
DIMENSION ACAP(30, 31), RCAP(100, 3), XSIG(100)
DIMENSION THICK(24)
EPS=1.0E-6
CRR=7.71707/1.0-.63662*SQRT(RDHA)+.25*RDBA)
SINT=SQRT(1.0-XC*XC)
THETA=ARCT(THICK(N))
COUNT=0.
SUM=0.
SINT2=SIGN(N*THETA)
COST2=SIGN(SINT2)
IF(SINT - EPS) 4, 5, 6
4 FACT=THETA*.5
GO TO 8
6 FACT=(1.0-XC)*SINT
8 DO 10 N=1, NF
COUNT=COUNT+1.
ANGLE=THETA/COUNT
SUM=SUM+THICK(N) * (COUNT*FACT*SIGN(ANGLE) - COST(ANGLE))
10 CONTINUE
U=2.*SIGN(UINF)*COST2*SUM+ACAP(1, 1)*SINT2+.25*COST2*(1.0*XC)*(3.0*XC-ACAP)
11.*GAMMA
SUM=0.
ANGLE=0.
DO 12 N=1, NGAM
ANGLE=ANGLE+THETA
12 SUM=SUM+ACAP(N+1, 1)*SIGN(ANGLE)
U=UJ*COST2*SUM
IF(ISEP) 25, 99, 25
25 SUM=0.
XSEP=SIG(1)
XATT=SIG(NSIG+1)
DO 40 N=2, NSIG
40 SUM=SUM+ACAP(N, 1)*FB(SIG(N-1), SIG(N), SIG(N+1), XC)
IF(XC-XATT-EPS) 45, 45, 46
45 FACT=(1.0-XATT)**(-1.5)*SQRT(XATT-XSEP)*(1.0-XC)/XATT-XATT)* (1.0+1.*XSEA-CAP)
46 LXATT=6.*XC-SIGN*(1.0-SQRT((XSEP-XC)/(XATT-XC))
GO TO 55
4E FACT=-SIGN*(1.0-SQRT((XSEP-XC)/(XATT-XC))
GO TO 55
49 FACT=-SIGN
55 U=UJ*COST2*(ACAP(N, 1)*SIGN*SIGNSJM)
99 U=SIGN(UINF)*SQRT(XC)+CORR*UT5/SQRTTI+X5*RCBA
RETURN
END
SUBROUTINE YVR(Y, I)
REAL Y(10)
REAL MVB
COMMON /XPTVB/ FTVB(64), FPVB(64), FPPRVA(64), DIDRVA(64), YVB
A XMVB(64), DELVB, XMUVB, FOVB, XMUAVB,
B ATQVB, ATCVB, ATSVB, ROVB, RVB(64), YVB
C MVB(64), NVB
Y(1) = (RVB(I) - DELVB)**2 * MVB(I)
Y(2) = FPVB(I)**2 * MVB(I)
Y(3) = FTVB(I)**2 * DIDRVA(I)
Y(4) = (DELVB - RVB(I) * FTVB(I) * XMVB(I) * MVB(I)
Y(5) = FPVB(I) * FTVB(I) * XMVB(I) * MVB(I)
Y(6) = RVB(I) * (DELVB - RVB(I)) * MVB(I)
Y(8) = (RVB(I) - DELVB) * FPPRVA(I) * FTVB(I) * XMVB(I) * MVB(I)
IPI = I+1
IF(IPI .GE. NVB) GO TO 12
SUM = 0.
DO 10 J = IPI, NVB
10 SUM = SUM - (RVB(4+1) - RVB(4)) * (RVB(4+1) * MVB(J+1)
A = RVB(J) * MVB(J))
12 Y(7) = FPPRVA(I)**2 * SUM / 2.
RETURN
END
SUBROUTINE POLLY(N,ABS,REL,AN,AA)
IMPLICIT REAL*8 (A-H,O-Z)
C COMPLEX ROOTS OF A POLYNOMIAL BAIRSTOWS METHOD
DIMENSION A(30),AN(60),C(26),ABAR(26),B(30),AA(30)
III=1
7 NPI=N+1
NPP1=N+2
DO 6 GCI I=1,NPI
LLL=NPP1-I
601 A(I)=AA(LLL)
13 DO 14 K=1,NPI
14 ABAR(K)=A(K)
ABSSQ=ABS**ABS
RELSQ=REL**REL
NBAR=N
B(1)=A(1)
C(1)=A(1)
15 IF(NBAR-2).gt.00,210,17
17 PI=2
Q1=1
18 IFR=0
19 PI=PI+9,
QT=QT*10.*
33 P=P1
Q=Q1
NRP1=NBAR+1
34 L=1
LAST=NBAR
OTST=9.99D36
C BAIRSTOW ITERATION
37 B(2)=ABAR(2T)-P*B(B1)
DO 40 K=3,NRP1
40 B(K)=ABAR(K)-P*B(K-1)-Q*B(K-2)
45 C(2)=B(2)-P*C(1)
DO 50 K=3,LAST
50 C(K)=B(K)-P*C(K-1)-Q*C(K-2)
C(LAST)=C(LAST)-B(LAST)
D=C(LAST-1)*C(LAST-1)-C(LAST)*C(LAST-2)
DSQR=D**D
IF(DSQR.DT.3619.19.60)
65 DELP=RT(LAST)*C(LAST)*I-RT(LAST+1)*C(LAST-2)/D
DELQ=(B(LAST+1)*C(LAST-1)-B(LAST)*C(LAST))/D
C TEST FOR CONVERGENCE
RELP=DELP/P
RELP=DELP/P
RELP=RELSQ/RELSQ
RELP=RELSQ/RELSQ
DELSQ=RELSQ/RELSQ
P=P*DELP
Q=Q*DELQ
IF(RELPS.RELPS).lt.7C,70,65
65 IF(RELPS.RELPS).le.70,70,80
70 IFRELSQ.RELSQ.120.75
75 IF(DELQ.120,120,80
80 GO TO 190,1007,L
POLLY 1
POLLY 2
POLLY 3
POLLY 4
POLLY 5
POLLY 6
POLLY 7
POLLY 8
POLLY 9
POLLY 10
POLLY 11
POLLY 12
POLLY 13
POLLY 14
POLLY 15
POLLY 16
POLLY 17
POLLY 18
POLLY 19
POLLY 20
POLLY 21
POLLY 22
POLLY 23
POLLY 24
POLLY 25
POLLY 26
POLLY 27
POLLY 28
POLLY 29
POLLY 30
POLLY 31
POLLY 32
POLLY 33
POLLY 34
POLLY 35
POLLY 36
POLLY 37
POLLY 38
POLLY 39
POLLY 40
POLLY 41
POLLY 42
POLLY 43
POLLY 44
POLLY 45
POLLY 46
POLLY 47
POLLY 48
POLLY 49
POLLY 50
POLLY 51
POLLY 52
POLLY 53
POLLY 54
POLLY 55
\begin{verbatim}
50 ITER=ITER+1
100 IF(DTEST=DELSQ)34,34,110
110 DTEST=DELSQ
    P(2)=A(2)-P*B(1)
    DO 115 K=3,NPI
    115 R(K)=A(K)-P*R(K-1)-Q*A(K-2)
    GO TO 45
C ITERATION HAS CONVERGED
120 GO TO (130,140) ; L
130 L=2
    LAST=N
    GO TO 110
C FACTOR OUT QUADRATIC
140 NBAR=NBAR-2
    NBPI=NBAR+1
    Aabar(2)=Aabar(2)-P*abar(1)
    DO 150 K=3,NBPI
    150 Aabar(K)=Aabar(K)-P*Aabar(K-1)-Q*Aabar(K-2)
    GO TO 250
C SOLVE LINEAR EQUATION
200 NBAR=NBAR-1
    R1=-Aabar(2)/Aabar(1)
    R2=0.
    GO TO 262
C NORMALIZE QUADRATIC
210 P=Aabar(2)/Aabar(1)
    Q=Aabar(3)/Aabar(1)
    NBAR=NBAR-2
C SOLVE NORMALIZED QUADRATIC
250 R1=-P/2.
    C1=R1*R1-O
    IF(C1>270,280,260
    260 C1=DSORT(C1)
    R2=R1-C1
    R1=R1+C1
    262 C1=0.
    GO TO 290
    270 C1=-C1
    C1=DSORT(C1)
    280 R2=R1
    290 C2=-C1
        AN(I)=C1
        AN(I+1)=R1
        AN(I+2)=C2
        AN(I+3)=R2
    310 EXIT (6,600)
    IF(NBAR-14,200,15
    340 FORMAT(1X,50HNO CONVERGENCE IN 250 ITERATIONS, POLLY HAS SPEKEN)
7 CONTINUE
RETURN
END
\end{verbatim}
SUBROUTINE SETIT(IG0,M,NV,REB,X,Y,UC,PRESS,GRAD,DELT,DISP,THETA,VISSETUP)
15C,MTTRAN)
C
SUBROUTINE FOR CALCULATION OF BOUNDARY LAYER THICKNESS,
C DISPLACEMENT THICKNESS, MOMENTUM THICKNESS AND EDdy VISCOSITY.
C
DIMENSION X(300),Y(100),UC(100,3),VIS(100,2),GRAD(100)

RTR=SQR(REB)
NY=NV+2
UEDGE=.995*UC(NY,1)
DO 10 N=1,NV
10 CONTINUE
41 NOELT=N

GO TO 20
10 CONTINUE
20 DELT=Y(NDELT)*(UEDGE-UC(NDELT,1))*(Y(NDELT+1)-Y(NDELT))/UC(NDELT)
1*UC(NDELT,1))
SUM=0.
DO 50 N=2,NV
50 SUM=SUM+(Y(N)-Y(N-1))*UC(N,1)+UC(N-1,1))
SUM=0.
UEDGE=UC(NY,1)
DO 60 N=2,NV
60 SUM=SUM+(Y(N)-Y(N-1))*UC(N-1,1))*UC(N-1,1)+UEDE-UC(N-1,1))
THE='QTED=0.5*SUM/RTR*UEDGE**2.
IF(LGO)53,53,56

53 NVP=NV+1
EASE=1.
IF(M=MTTRAN)31,32,32
32 IF(M=MTTRAN+5-M)31,31,33
33 EASE=(X(NM)-X(MTRAN))/X(MTRAN+5-X(MTRAN))
31 CONTINUE
INNER=0
FAC1=.16*RTR*EASE
FAC2=.016*EDGE*DISP*REB*EASE
FFAC=-RTR/26.
EFFAC=PRES/RTR
TAUW=GRAD/I/RTR
DO 160 N=2,NV
160 INNER=INNER+1
ALTER=1.0+FAC2/(1.0+5.5*Y(N)/DELT)**6.
IF(INNER)402,401,402
402 VISC(N,1)=ALTFR
GO TO 160

401 CONTINUE
TAUW=TAUW-Y(N)*EFAC2
IF(TAUMY)701,701,702
7C1 VISC(N,1)=1.
GO TO 703
702 FX=Y(N)*EFAC2*SQR(TAUMY)
VISC(N,1)=1.0+FAC1*Y(N)*Y(N)*ABS(GRAD(N))*Y(N-1.-EXP(EX1)**2
7C3 IF(VISC(N,1)-ALTER)150,160,521
521 VISC(N,1)=ALTER
INNER=INNER+1
16C CONTINUE
SAVE=1.
DO 162 N=2,NV
RAVE=VISC(N,1)
VISC(N,1)=(VISC(N+1,1)+RAVE+SAVE)/3.
162 SAVF=RAVE
56 CONTINUE
RETURN
END
CALL H4X4(INOT,XSEP,DFL,THET1,XATT,REB,USEP,X3,H3,X4,H4)

IF (XSF-P-1.) 24, 25, 25

25 CP4=0.
GO TO 27

24 URAT=EXP(-.08712-UI1(H4)-.24123*(.3255+UI2(H4)))
CP4=1.-(1.-PREC)/URAT**2
DEADL=XATT-XSEP
IF (DEADL-2.) 5, 6, 6

5 G=1.5*DEADL/2.
GO TO 7

6 G=1.

7 CP4=PREC+CP4-(CP4-PREC)/(1.-G*XSEP)

27 CONTINUE
COEF=(PREC-CP4)/(XATT-X4)
C2=-2.*UINF
DO 20 H=1,NSIG

20 X=XBSIG(H)
IF (X-1.) 2, 2, 3

2 THEATA = ARC(T(X))
TANT = SIN(.5*THETA)/COS(.5*THETA)
CI = -CZ*(1.-COS(THETA))/1.
DO 10 N=1,NF
COUNT=COUNT+1.
ANGLE=COUNT*THETA

10 SUM=SUM+THICK(IN)*CI*COS(ANGLE)+C2*(COUNT+TANT)*SIN(ANGLE)-C3*ANGLE
IF (.) 41

41 SUM=SUM-5.*CZ*THICK(IN)
GO TO 35

3 CI=CZ*(1.-X)
XRAD=1./(X+SQRT(X*X-1.))

46 CF=CZ*(1.-X)
RF=SQRT((1.-X)/(1.*X))
SUM=THICK(IN)*XRAD*FC*(RF-1.)-CF*(1.-.5*XRAD)
FRAD=XRAD

COUNT=1.
DO 30 N=2,NF
COUNT=COUNT+1.
FRAD=FRAD*XRAD

30 SUM=SUM+THICK(IN)*FRAD*FC*(COUNT+RF-1.)*CI
35  CP=CP4  
50  CP=CP+(X-X4)*COEF  
55  CONTINUE  
   FPRES(M)=-UNF*CP*SUM  
20  CONTINUE  
   RETURN  
END

MIXER 56  
MIXER 57  
MIXER 58  
MIXER 59  
MIXER 60  
MIXER 61  
MIXER 62  
MIXER 63
SUBROUTINE RUBAL(DFLL, THETI, RE9, XC1, UI, XC5, DCP, DEL5, X, XC, MX, NZ, X5, UMUR)

15, UF, ALTC, RENFL, USTOP

I1, IMASIN, IN(300), UC(300), UE(300, 3)

FCAK(X) =-1.9656*1.0755*XX-336.33**3+581.1**4-295.94*XX**5

UI(X)=-1.4692*X+.64825*XX-.65293*XX**3+.6592*XX**4

UI(X)=-.045929*X1.91615*X2.98143*X**2.542125*X**4

FDRE(X) =FRE(2.5773-3.5252*X-4.379*XX-.076511*XX**3-.0937.7**4)RUBA

FAK(X) =FRE(-3.7491+0.038772*X+.41967*XX+.071046*XX**3+.023162*X**3)

1**4

DEL1(X) =-.045929*ALOG1(X) -3.9242*X+.54535*XX-1.39147*XX**3-1.3423*XX**4

L**4

FORMAT(1H1, 44X, 3HANALYSIS OF LEADING-EDGE RUBBLE/////34X, 1H1, 19X, 1H1RUBA

1H1, 19X, 1H1RUBA, 18X, 1H4DIP/)

10 FORMAT(20X, 4F20.5)

MOUT=6

H1=.25

H5=.429

DO 5 M=1,NZ,1

IF(XC1>XC(M)) 4,4,5

M1=M

CONTINUE

6 XI=X(M-1)+XI(XI-1)/(XI(XI-1)*XI(XI-1))/XI(XI-1)

ARG=ALOG((X4-XI)/1 infusion DEL1/DELI U11)

H4=.25*FAK(ARG)

DELI=.58*FDEL1(ARG)*DELI

X5=X4+5*DELI (1-4H/4.29)**2

IF(U1-USTOP) '41, 41, 40'

CONTINUE

40 ALTL=ALTC*DELI

IF(X5-XL, LT ALTL) X5=XL+ALTL

41 UREX=X-P/0.08712*(H4)-2.4723*(3.355+U12(H4))

DPL=U1*I1-U1**2

DREX=X-P*(2.24374-FCAK(H4)) +2.4723*(2.0214*DELI H4))

DEL5=DREA*DELI

DC 7 NZ, MX

IF(X5-X(M)) 16, 16, 7

16 M5=M

CONTINUE

7 CONTINUE

8 FACT=X5-X(M-1)+X1(X1-1)*X5-M-1)

FACT1=1.-FACT

XC5=XC(M-1)+FACT1+XC(M)*FACT

U5=UF(M-1)+FACT1+UF(M+1)*FACT

WRITE(MOUT, '251')

WRITE(MOUT, '10) XI, U1, H1, DEL1

WRITE(MOUT, '10) X5, U5, H5, DEL5

RETURN

END
SUBROUTINE YSET(R, A, NY, Y)
DIMENSION Y(100)
RPI = 1. * R
Y(1) = 0.
Y(2) = A
DO 10 N = 3, NY
    Y(N) = RPI * Y(N-1) - R * Y(N-2)
10 CONTINUE
RETURN
END
SURROUTINE H4X4(INDT,X1,DEL1,THET1,X5,REB,U1,X3,H3,X4,H4) H4X4 1
CURFL(H)=26.703/H+0.033*ALOG(H)-211,3*H+327.8*H*H-24.03*H**3 H4X4 2
FDELT(X)=EXP(2.5773-3.4252*X-4.379*X*X-0.076511*X**3-0.0039737*X**4) H4X4 3
FATCH(X)=EXP(-3.7481+0.038772*X+4.1967*X*X+0.71046*X**3+0.032162*X**4) H4X4 4
1*4) H4X4 5
10 FORMAT('20X,54HA SOLUTION FOR X4 COULD NOT BE OBTAINED IN 1000 TRY') H4X4 6
11(ALS)) H4X4 7
MOUT=6 H4X4 8
C
C IF IND1 IS NONZERO, THE BOUNDARY LAYER IS TURRULFNT H4X4 9
C AT SEPERATION. H4X4 10
C
2 IF(INDT)$2,5,2 H4X4 11
H3=THET1/DEL1 H4X4 12
X3=X1 H4X4 13
DEL3=DEL1 H4X4 14
GO TO 20 H4X4 15
5 X3=X3+5,F4/(U1*REB) H4X4 16
ARG=ALOG((X3-X1)/(REB*DEL1*DEL1)) H4X4 17
H3=THET1*FATCH(ARG)/DEL1 H4X4 18
DEL3=5.5*FDELT(ARG)*DEL1 H4X4 19
IF((X3-X5) .GT. 20,15,15) H4X4 20
15 H4=.429 H4X4 21
X4=X5 H4X4 22
GO TO 50 H4X4 23
20 CONTINUE H4X4 24
IG0=0 H4X4 25
DIST=X5-X1 H4X4 26
UNDER=0 H4X4 27
H4=H3+H3 H4X4 28
COEF1=DEL3*H3 H4X4 29
COEF2=10.5*DEL3*H3 H4X4 30
SUB=X3-COEFL*CURFL(H3) H4X4 31
95 OVER=H4 H4X4 32
H4=.5*(H4+UNDER) H4X4 33
X4=CURFL(H4)*COEF1+SUB H4X4 34
ALTER=X5-COEFL*(1.-(H4/.429)**21)/H4 H4X4 35
IG0=IG0+1 H4X4 36
IF((X4-ALTER) .LT. 41,50,42) H4X4 37
41 IF((IG0.1000) .LT. 95,61,61) H4X4 38
42 IF((ABS(X4-ALTER)/DIST-.001) .LT. 50,50,43) H4X4 39
43 UNDER=H4 H4X4 40
H4=.5*(OVER/H4) H4X4 41
X4=CURFL(H4)*COEF1+SUB H4X4 42
ALTER=X5-COEFL*(1.-(H4/.429)**21)/H4 H4X4 43
IG0=IG0+1 H4X4 44
IF((X4-ALTER) .LT. 52,50,51) H4X4 45
51 IF((IG0.1000) .LT. 43,61,61) H4X4 46
52 IF((ABS(X4-ALTER)/DIST-.001) .LT. 50,50,95) H4X4 47
61 H4=.429 H4X4 48
X4=X5 H4X4 49
WRITE(MOUT,10) H4X4 50
50 CONTINUE H4X4 51
RETURN H4X4 52
END H4X4 53
SUBROUTINE SFTSX(NSPI, XSEP, XATT, XSIG, ANGLE)

DIMENSION XSIG(100)
A = .5*(XSEP+XATT)
B = .5*(XATT-XSEP)
ARG = 0.
DO 5 N=1, NSPI
XSIG(N) = A - B*COS(ARG)
5 ARG = ARG + ANGLE
RETURN
END
FUNCTION ARCT(X)

PI=3.14159

IF(ABS(X)-1.E-6) 1,2,2
1  ARCT=.5*PI
   GO TO 6
2  IF(X+.99999) 3,4,4
3  ARCT=PI
   GO TO 6
4  ARCT=ATAN(SQRT(1.-X*X)/X)
   IF(ARCT) 5,6,6
5  ARCT=ARCT+PI
6  CONTINUE
RETURN
END
FUNCTION GAM1(ACAP, DX1, PI)
DIMENSION ACAP(30,3)
GAM1=PI*(-1.5*ACAP(1,1)-.75*ACAP(2,1)+2.*ACAP(1,2)+ACAP(2,2)-.5*ACGAM1
1AP(1,3)-.25*ACAP(2,3)) / DX1
RETURN
END
FUNCTION FB(X1, X2, X3, Y)
D1=1./(X2-X1)
D2=1./(X3-X2)
T1=ABS(Y-X1)
T2=ABS(Y-X2)
T3=ABS(Y-X3)
EPS=1.E-6
IF(T1-EPS) 2,3,3
2 F1=0.
F2=ALOG(T2)
F3=ALOG(T3)
GO TO 10
3 F1=ALOG(T1)
IF(T2-EPS) 4,5,5
4 F2=0.
F3=ALOG(T3)
GO TO 10
5 F2=ALOG(T2)
IF(T3-EPS) 6,7,7
6 F3=0.
GO TO 10
7 F3=ALOG(T3)
10 FB=(Y-X1)*F1*D1+(D1+D2)*(X2-Y)*F2+(Y-X3)*F3*D2)/3.14159
RETURN
END
SUBROUTINE EGAMI(NU, NG, A, B, XSEP, XATT, GAMMA, Y, GI)

DIMENSION A(30,3)
SINT=SQRT(1.-Y*Y)
THETA=ARCT(Y)
SUM=0.
COUNT=1.
DO 6 N=2,NG
COUNT=COUNT+1.
6 SUM=SUM+A(N+1,NU)*(SINT(COUNT+1.)*THETA)/(COUNT+1.)*SINV(COUNT-1.)
THETA/(COUNT-1.))
GI=(3.14159-THETA*SIGN(A(1,NU)+.5*A(2,NU))+.5*SUM-.25*GAMMA*(1.+EGAMI 1)

IF(Y-XATT) 8,8,7
DIFF=I.-XATT
7 IF(DIFF-1.E-6) 8,8,9
9 GI=GI+2.*B*DIFF**(-1.5)*SQRT((XATT-XSEP)*(1.-Y)*(Y-XATT))
CONTINUE
RETURN
END
SUBROUTINE ESIGI(NU, NX, XS, A, Y, SI)
DIMENSION XS(100), A(10, 3)
SUM = 0.
DC IC I = 2, NX
10 SUM = SUM + A(I, NU) * G9(XS(I-1), XS(I), XS(I+1), Y)
SI = A(I, NU) * AINT(XS(1), XS(NX+1), Y) + SUM
RETURN
END
FUNCTION GB(X1, X2, X3, X)
GB = ABINT(X1, X2, X) - ABINT(X3, X2, X)
GB = GB / 3.14159
RETURN
END
FUNCTION ABINT(A,B,X)  
enerima 
A = ABS(X-A)  
B = ABS(X-B)  
COEF = 2.0*(B-A)  
APL = A+1  
BPL = B+1  
IF (ARGA - 1.E-6) .GT. 3  
        CA = 0  
        GO TO 5  
3  
        CA = ALOG(ARGA)  
        IF (ARGA - 1.E-6) .GT. 3  
1  
        CR = 0  
        GO TO 6  
5  
        CR = ALOG(ARGB)  
        IF (ARGB - 1.E-6) .GT. 3  
6  
        ABINT = (CA - .5)*ARCB**2 - (CB - .5)*ARGB**2 - (ALOG(AP1) - .5)*AP1**2 + (ALOG(BP1) - .5)*BP1**2 - COEF*((X-A)*(X-B) + BPL*(ALOG(BP1) - 1.))  
        ABINT = ABINT/COEF  
        RETURN  
FND
FUNCTION BINT(XS, XZ, X)
RTS=SORT(1.+XS)
RTZ=SORT(1.+XZ)
BINT=1.-X*RTS*RTZ
IF(XZ-X)2,3,3
2RTSX=SQR(T(X-XS))
RTZX=SQR(T(X-XZ))
BINT=BINT+(XZ-XS)*ALCG((RTSX+RTZX)/(RTS+RTZ))+RTSX*RTZX
GO TO 50
BINT 1
BINT 2
BINT 3
BINT 4
BINT 5
BINT 6
BINT 7
BINT 8
BINT 9
BINT 10
BINT 11
BINT 12
BINT 13
BINT 14
BINT 15
BINT 16
BINT 17
BINT 18
GO TO 50
3 IF(X-XS)5,5,4
4 BINT=BINT+(XZ-XS)*ALCG(SQR(T(XZ-XS))/(RTS+RTZ))
GO TO 50
5 RTSX=SQR(T(XS-X))
RTZX=SQR(T(XZ-X))
BINT=BINT+(XZ-XS)*ALCG((RTSX+RTZX)/(RTS+RTZ))-RTSX*RTZX
50 CONTINUE
RETURN
END
SUBROUTINE SCAL(SAL*NSAL,FRZ,ARR,ROBAR)
DIMENSION SAL(300)
DELT=FRZ*ROBAR
EN=ARR/FRZ
DO 5 N=1,300
IF(EN-N) 4,4,5
4 NE=N
GO TO 6
5 CONTINUE
6 NG=NSAL-NE
EN=FLOAT(NG)
NGM1=NG-1
SBL(1)=0.
DO 7 N=2,NE
7 SAL(N)=SAL(N-1)+DELZ
FRAC=2.2/DELZ
FRAC1=FRAC-1.
R=FRAC**(1./FLOAT(NGM1))
8 SAFE=R
R=R-(R**NG-FRAC*R+FRAC1)/(EN*R**NGM1-FRAC)
IF(ABS(SAVE-R)-1.F-6) 9,9,8
9 RP1=R+1.
DO 10 N=NE,NSAL
10 SBL(N+1)=RP1*SBL(N)-R*SBL(N-1)
RETURN
END
SUBROUTINE TERPF(XI,J,TAB1,TAB2,TAB3,TAB4,XITAR,FP)
DIMENSION TAB1(24),TAB2(24),TAB3(24),TAB4(24),XITAR(24)

I=IF(XI-.0001) 2,2,10
2 GC TC (3,4,5,6),J
3 FP=2.53-2.439*ALOG(XI)
GO TO 99
4 FP=3.54-1.725*ALOG(.7071*XI)
GO TO 99
5 FP=4.58-1.2195*ALOG(.5*XI)
GO TO 99
6 FP=10.12
GO TO 99
10 DO 12 N=1,24
IF(XI-XITAB(N)) 11,11,12
11 NX=N
GO TO 13
12 CONTINUE
13 TX=(XI-XITAB(NX))/XITAB(NX)-XITAR(NX-1)
TX1=1.-TX
GO TO (14,15,16,17),J
14 FP=TX1*TAB1(NX-1)+TX*TAB1(NX)
GO TO 99
15 FP=TX1*TAB2(NX-1)+TX*TAB2(NX)
GO TO 99
16 FP=TX1*TAB3(NX-1)+TX*TAB3(NX)
GO TO 99
17 FP=TX1*TAB4(NX-1)+TX*TAB4(NX)
CONTINUE
99 RETURN
END
SUBROUTINE EVAL(NNF,XX,SSC,SST,CCA,TTB,CCM,TTM)
DIMENSION SSC(50),SST(50)
COST = 2.*XX - 1.
COSTS = COST**2
IF(COSTS-1.E-8) 303,304,304
304 TANT = SQRT(1./COSTS - 1.)
THF = ATAN(TANT)
GO TO 305
303 THE = 1.5708
305 IF(COST) 403,404,404
403 THE = 3.14159 - THE
404 ARG = 0.
SUM1 = 0.
SUM2 = 0.
DO 551 N=1,NNF
ARG = ARG + THE
SUM1 = SUM1 + SSC(N)*SIN(ARG)
551 SUM2 = SUM2 + SST(N)*SIN(ARG)
CCM = SUM1*SIN(THE)*CCM
TTB = (1. - COS(THE))*SUM2*TTM
RETURN
END
SUBROUTINE SIMP(NS, DX, ORD, FIND)
DIMENSION ORD(50)

C INTEGRATION OF NS + 1 EQUALLY SPACED ORDINATE VALUES
C BY SIMPSON'S RULE. NS MUST BE EVEN
SUM = 0.
DC 88 I=2,NS+2
 88 SUM = SUM + 2.*ORD(I-1) + 4.*ORD(I)
FIND = DX*(SUM - ORD(1) + ORD(NS+1))/3.
RETURN
END
SUBROUTINE SECT(XU,YU,XL,YL,NOFF,NF,RODC,TMAX,CMAX,ST,SC)

C PROGRAM TO COMPUTE COEFFICIENTS TN AND CN OF THE FOURIER SERIES

C REPRESENTATION OF SECTION THICKNESS AND CAMBER DISTRIBUTIONS

DIMENSION XU(30),YU(30),XL(30),YL(30),YUC(30),YLC(30),ST(24),SC(24)
1,NUM(50),TBAR(50),CPR(50)

12 FORMAT(///47X,26HINPUT AND COMPUTED OFFSETS/) SEC
1,5HYLC/C/) SEC

14 FORMAT(3X,3F16.5,8X,3F16.5)
NA=6
RNA=6.*
RNF=FLOAT(NF)
MCUT=6
PI = 3.14159
DELT = PI/(2.*RNF)
NTC = 2*NF - 1
NINT = NTC + 2
NSIMP = NTC + 1
RDBC=5.*RODC
VARY = 0.*
CA = 0.*
TB = 0.*

THETA = 0.*
DO 89 K=1,NTC
THFTA = THETA + DELT
XI = .5*(1. + COS(THFTA))
DO 90 LAM=2,NOFF

IF(XI-XU(LAM)) L10,90,90
10 YUINT = YULAM-1) + (XI - XU(LAM-1))YU(LAM) - YU(LAM-1))/XU(LAM) SECT 29
1) - XI(LAM-1))
GC TO 111
SECT 30
5C CONTINUE
SECT 31
111 DO 80 LAM=2,NOFF

SECT 32
80 CONTINUE

SECT 33
112 TARTK+1) = .5*(YUINT - YLINT)
89 CINTK+1) = .5*(YUINT + YLINT)

SECT 34

SECT 35
80 CONTINUE

SECT 36
SECT 37
112 TARTK+1) = .5*(YUINT - YLINT)
89 CINTK+1) = .5*(YUINT + YLINT)

SECT 38
SECT 39

SECT 40
TMAX = 0.*
CMAX = 0.*
DO 79 K = 2,NSIMP

SECT 41
SECT 42
IF(TBAR(K)-TMAX) 801,802,802

SECT 43
SECT 44
802 TMAX = TBARTK)
SECT 45
801 IF(CINTK-CMAX) 79,702,702
SECT 46
702 CMAX = CINTK)
SECT 47
75 CONTINUE

SECT 48
IF(CMAX-1.E-5) 1201,1202,1202
1201 CMAX=1.
SECT 49
SECT 50
1202 CONTINUE

SECT 51
IF(TMAX-1.E-5) 1140,1141,1141
1140 TMA=1.
SECT 52
SECT 53
1141 DO 69 K=2,NSIMP

SECT 54
TBARTK) = TBARTK)/TMAX
SECT 55

111
CBAR(K) = CBAR(K)/CMAX
TBAR(1) = 0.
CRAR(1) = 0.
TBAR(NINT) = 0.
CRAR(NINT) = 0.
TTA = TBAR(NA)
TTR = TBAR(NA+1)
TTC = TBAR(NA+2)
TAA = DELT*(RNA-1.
TTB = TAA + DELT
TCC = TBB + DELT
XP = .5*COS(TAA)
XB = .5*COS(TBB)
XC = .5*COS(TCC)
SLOPE = ((TTC-TTB)*(XR-XA)/(XC-XB) + (TTB-TTA)*(XC-XR)/(XB-XA))/XSECT
XC = .5*COS(TCC)

DATA (M,NNA,NNT,NTIN,NTNL,NTNL2)
M = 456
NNA = 2
NNT = 1
NTIN = 1
NTNL = 1
NTNL2 = 1

2 56
TBAR(1) = (SQRT(1.-COSR)**1.5)*(TTB*(1.+COST-2.*COSB))/1.*SIN(THETA)
1.5)/(2.*COST)
THETA = TAA
NAP = NA + 1
DO 458 I = NAP,NLE
THETA = THETA + DELT
TBAR(I) = TRAR(I)/(1.-COS(TA))
THETA = 0.
DO 459 I = 2,NSIMP
THETA = THETA + DELT
CBAR(I) = CBAR(I)/SIN(THETA)
RKK = 0.
DO 59 K = 1,NF
RKK = RKK + 1.
THETA = 0.
DO 777 I = 1,NINT
DUM(I) = TBAR(I)*SIN(THETA*RKK)
THETA = THETA + DELT
CALL SIMP(NSIMP,DEL,T,DUM,VARY)
SY(K) = 2.*VAR/P
THETA = 0.
DO 888 I=1,NINT
  DUM(I) = CRAAI(I)*SIN(THETA*RXK)
END
THETA = THETA + DELT
CALL SIMP(NSIMP,DELT,DUM,VARY)
59 SC(K) = 2.*VARY/PI
DO 569 I=1,NOFF
  X = XU(I)
  CALL FVAL(NF,X,SC,ST,CR,TC,CMAX,TMAX)
569 YUC(I) = CB + T9
  NO 869 T=1,NOFF
  X = XL(I)
  CALL FVAL(NF,X,SC,ST,CR,TC,CMAX,TMAX)
END YLC(I) = CB - TR
  SUM1 = 0.
  COUNT = 0.
DO 659 I=1,NF
  COUNT = COUNT + 1.
659 SUM1 = SUM1 - ST(I)*COUNT*(-1.)**I
  RDCBC = R.*(TMAX*SUM1)**2
  RDCBC=2.*RDCBC
  TMAX=2.*TMAX
  CMAX=2.*CMAX
WRITE(MOUT,12)
WRITE(MOUT,13)
WRITE(MOUT,14) (XI(I),YU(I),YUC(I),XL(I),YL(I),YLC(I),I=1,NOFF)
RETURN
END
SUBROUTINE CORDX(NSRL,NZ,RDBB,SRL,XC)
C
C BOUNDARY LAYER COORDINATES AND CORRESPONDING CHORDAL
COORDINATES ARE COMPUTED HERE.
C
DIMENSION SRL(301),X(300),XC(300)

336 FORMAT(10X,3I1) ITERATION TO COMPUTE XC FOR M =15,32 H DID NOT CONVERGE
1E8 IN 1000 STEPS.:

337 FORMAT(1H1,25X,1H4,20X,1HS,25X,1HX,24X,2HXC/1)

338 FORMAT(22X,1S,3E25.5)

MOUT=6
MX = NSBL + NZ - 1
RZERO = RDBB/2.
XC(NZ) = -1.
DO 255 M=1,NZ
MM = NZ + 1 - M
255 X(M) = SRL(NZ) - SRL(MM)
DO 256 M=NZ,MX
MM = M + 1 - NZ
256 X(M) = SRL(NZ) + SRL(MM)
DO 257 M=1,MX
IF(NZ-M) 333,257,335
257 MM = NZ + 1 - M
258 L=1,1000
SAVE = XC(M)
CALC1 = SQRT((1.+XC(M))/RZERO)
CALC2 = SQRT(1.+((1.+XC(M))/RZERO))
XC(M) = XC(M)*CALC1*SRL(K) - RZERO*(CALC1*CALC2+ALOG(CALC1+CALC2))/CALC2
L /CALC2
IF(ABS(SAVE-XC(M))-L.E-6) 257,257,258
258 CONTINUE
WRITE(MOUT,336) M
CONTINUE
WRITE(MOUT,337)
DO 264 M=1,MX
IF(NZ-M) 261,261,262
261 K=NZ-M+1
GO TO 263
262 K=M+1-NZ
263 WRITE(MOUT,338) M,SRL(K),X(M),XC(M)
264 CONTINUE
RETURN
END
SUBROUTINE PGRAD(M, X, UE, DXI, PRESS, SA, SR, SC, SR, SS)

C
C SUBROUTINE FOR CALCULATION OF PRESSURE GRADIENT AND
C DERIVATIVE COEFFICIENTS.
C
DIMENSION X(300), UE(300, 3)
D1Z=X(M+1)-X(M)
D2Z=X(M+2)-X(M)
D21=X(M+1)-X(M+1)
D1M1=X(M+1)-X(M-1)
DZM1=X(M)-X(M-1)
XIM=D1Z/(D2Z*D21)
ETAM=1./D1Z-1./D21
ZETAM=D21/(D1Z*D2Z)
PRESS = (3.*UE(M+1, 1)-4.*UE(M+1, 2)+UE(M+1, 3))/(2.*DXI)+UE(M+1, 1)
1XIM*UE(M+2, 1)+ETAM*UE(M+1, 1)-ZETAM*UE(M, 1))

SA=1./D1Z+1./D1M1
SR=D1M1/(D1Z*D2M1)
SC=D1Z/(D1M1*D2M1)
SR=0
SS=D1Z/D2M1
RETURN
END
SUBROUTINE TRANS(UPRIM, PRESS, THETA, RE8, UC, NY, FLAM, XFLAM, LAM2)

C SUBROUTINE TO TEST FOR TRANSITION IN A LAMINAR BOUNDARY LAYER.

C DIMENSION UC(100,3),FLAM(10),LAMINAR

F(X) = 0.11746 - 1.0582E-3*X - 1.1023E-4*X*X

TKAY = PRESS*REP*THETA**2/UC(NY,2)

IF(TKAY-.077) 2,2,99

2 IF(ARS(TKAY)-.07011) 3,3,4

3 ARG = TKAY**72.49

GO TO 5

4 ARG = 0.

DO 6 N=1,1000

SAVE = ARG

ARG = ARG - (ARG**2(TKAY)**2-TKAY)/(F(ARG)**0.11746-ARG*3.1746E-3 - ARG**5*5.1195E-4)

6 IF(ARS(TKAY)-SAVE/ARG)-1.E-6) 7,7,6

CCONTINUE

7 IF(ARG**11.) 8,8,5

8 EF = 1.75

GO TO 10

5 DO 15 N=1,10

IF(ARG-XFLAM(N)) 24,24,15

24 NBAR = N

GO TO 16

15 CONTINUE

16 EF = FLAM(NBAR-1)+(ARG-XFLAM(NBAR-1))*(FLAM(NBAR)-FLAM(NBAR-1))/XTRANS 27

IGNS(NBAR)-XFLAM(NBAR-1)

10 R = .5*EF

A = 3.36*(UPRIM/UC(NY,2))**2

RTH = F(ARG)**0.9860.*A**0.9860.

IF(RE8*THETA-RTH) 99,50,50

50 LAMQ = 0

95 CONTINUE

RETURN

END
SUBROUTINE CAPS(ITER,N,CAPG,CAPH,CAPJ,CAPK,SR,SS,SD,SE,SF,VISC,V,UCAPS)
1.
2.
3.
4.
5.
6.
GO TO 6
CONTINUE
RETURN
END

DIMENSION CAPG(100),CAPH(100),CAPJ(100),CAPK(100)
DIMENSION VISC(100,2),V(100,2),UC(100,3),SD(100),SE(100),SF(100)

IF(ITER) 2,6,1

2.
CAPG(N) = SR*V(N,1) - SS*V(N,2)
CAPH(N) = SR*VISC(N,1) - SS*VISC(N,2)
CAPJ(N) = SR*(SD(N)*VISC(N+1,1)+SE(N)*VISC(N,1)-SF(V)*VISC(N-1,1)) - SCAPS
CAPK(N) = SR*UC(N,2) - SS*UC(N,3)

GO TO 6

4.
CAPG(N) = .5*(CAPG(N)+V(N,1))
CAPH(N) = .5*(CAPH(N)+VISC(N,1))
CAPJ(N) = .5*(CAPJ(N)+SD(N)*VISC(N+1,1)+SE(N)*VISC(V,1)-SF(N)*VISC(NCAPS))
CAPK(N) = .5*(CAPK(N)+UC(N,1))

CONTINUE
RETURN
END
SUBROUTINE TERP(YIN,YBASE,VARY,NY,VALUE)

C SUBROUTINE FOR DETERMINING INTERPOLATED VALUE OF THE
FUNCTION VARY AT Y = YIN.

C

DIMENSION YBASE(100),VARY(100)
IF(YIN-YBASE(NY-1))2,3,3
VALUE = VARY(NY)
GO TO 10
10 CONTINUE

2 DO 15 N=1,NY
IF(YIN-YBASE(N))24,24,15
24 NRAR=N
GO TO 16
15 CONTINUE

16 D21=YBASE(NBAR)-YBASE(NBAR-1)
D31=YBASE(NBAR+1)-YBASE(NBAR-1)
D32=D31-D21
D3A=YBASE(NBAR+1)-YIN
D2A=YBASE(NBAR)-YIN
DA1=YIN-YBASE(N3AR-1)
VALUE=D3A*D2A*VARY(NBAR-1)/(D21*D31)+D3A*DA1*VARY(NBAR)/((D21*D32)-TERP 21
1D2A*DA1*VARY(NBAR+1)/(D31*D32)
GO TO 16
10 CONTINUE
RETURN
END
SURREOUTINE YDIFF(Y, ALPHA, RETA, GAMMA, DELTA, SD, SF, C2, C3, C4, Y)

DIMENSION ALPHA(100), RETA(100), GAMMA(100), DELTA(100)

DIMENSION SD(100), SF(100), C(100), Y(100)

NV=NY-2

NV1=NV+1

DC 40 N=2, NV

ALPHA(N) = 2.*((2.*Y(N)-Y(N-1)-Y(N+1))/((Y(N+2)-Y(N-1))*Y(N+2)) - Y(N) YDIFF 7

1+1)*(Y(N+2)-Y(N)))

DELTA(N) = 2.*((Y(N+2)-Y(N+1)-2.*Y(N))/((Y(N+2)-Y(N-1))*Y(N+1)) - Y(N) YDIFF 9

1-1)*(Y(N)-Y(N-1))

RETA(N) = (DELTA(N)*Y(N)-Y(N-1))**3 - ALPHA(N)*(Y(N+2)-Y(N))**3)/YDIFF 11

1(N+1)-Y(N))**3

GAMMA(N) = -ALPHA(N)-RETA(N)-DELTA(N)

CONTINUE YDIFF 13

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

RETURN

END

119
SUBROUTINE ELDER(BCAP, XSIG, NSIG, UNF, ELD, Y, YMAX)
DIMENSION RCAP(100,3), XSIG(100)
BCAP(NSIG+1,1)=0.
XS=XSIG(1)
XZ=XSIG(NSIG+1)
IF(XZ-1.) 16,16,1
1
154
DEADL=XZ-XS
YMAX=1.0E-10
SUM=5*(XSIG(2)-XS)*RCAP(2,1)
DO 10 N=2,NSIG
X=XSIG(N+1)
SUM=SUM+5*(X-XSIG(N))*RCAP(N+1,1)*BCAP(N,1)
10 IF(N-NSIG) 4,2,4
2 ANGLE=1.5708
GO TO 6
4 ANGLE=ATAN(SQRT((X-XS)/(XZ-X)))
6 Y=SUM+BCAP(1,1)*DEADL*ANGLE-SQRT((X-XS)*(XZ-X))
IF(Y-YMAX) 10,10,8
8 YMAX=Y
10 CONTINUE
ELD=Y/YMAX
IF(ABS(ELD)-UNF) 20,20,12
12 IF(ELD) 16,16,16
14 ELD=-UNF
GO TO 20
16 ELD=UNF
20 CONTINUE
RETURN
END
SUBROUTINE REATT (UC, VX, Y, XM, YR, DRY, UE, X5, DEL5, MST, REB)
DIMENSION UC(100, 3), VX(100, 2), Y(100)
DIMENSION X(300, 3), IF(300, 3)
DIMENSION TAI(24), TAH2(24), TAB3(24), TAB4(24), XITAB(24)
DATA TAI1 /24, 98, 23, 29, 21, 04, 19, 33, 17, 61, 15, 29, 13, 46, 11, 54, 13, 36, 9REATT/ 5
1, 38, 8, 35, 7, 32, 6, 72, 5, 31, 4, 4, 3, 5, 7, 2, 2, 2, 1, 2, 66, 31, 1, 14, 0, 1, 1, 0, / REATT 6
DATA TAI2 /20, 05, 18, 85, 17, 25, 15, 04, 14, 13, 12, 11, 77, 10, 3, 9, 3, 30, 5, 65REATT/ 7
1, 7, 95, 7, 7, 6, 43, 5, 56, 4, 9, 4, 18, 2, 39, 1, 86, 1, 11, 1, 62, 3, 3, 04, 0, 7, / REATT 8
DATA TAI3 /16, 65, 15, 8, 14, 67, 13, 8, 12, 91, 11, 66, 13, 65, 5, 43, 8, 71, 11, 11, / REATT 9
1, 7, 59, 7, 016, 6, 41, 5, 77, 5, 13, 4, 5, 3, 31, 2, 28, 1, 49, 9, 51, 0, 09, 0, 1, 0, / REATT 10
DATA TAI4 /10, 12, 10, 05, 9, 9, 3, 9, 78, 9, 58, 9, 17, 8, 72, 8, 08, 7, 6, 7, 7, 6, 83, REATT 11
16, 53, 6, 18, 5, 79, 5, 36, 4, 91, 3, 98, 3, 05, 2, 21, 1, 5, 95, 22, 03, 0, / REATT 12
DATA XI TAB/ /3001, 0002, 0005, 0001, 0002, 0005, 01, 02, 03, 04, 05, / REATT 13
106, .07, .08, .09, .11, .12, .14, .16, .18, .2, .25, .3, .35/ REATT 14
3 FORMAT (1/40X, 23HAT REATTACHMENT, BETA =E13.5)
MOUT = 6
MOUT = 6
RTR = SCRT (REFR)
UC(1, 2) = 0.
UC(1, 3) = 0.
V(1, 1) = 0.
V(1, 2) = 0.
DO 5 M = 1, XM
5 IF X5 = XMT) = 4, 4, 5
MST = M + 2
GO TO 6
CONTINUE
6 XA = X(MST - 2)
XB = X(MST - 1)
UA = UE(MST - 2, 1)
UR = UE(MST - 1, 1)
ZA = ALOG (UA * DEL5 * REB)
PGRAD = UA - UR / (UA * UB) * (XB - XA)
RETM2 = (1.0974 * SCRT (DEL5 * PGRAD)) / (1.049 * 0.04565 * TA)
IF (RETM2 < 1.) 8, 7, 7
7 BETM2 = 1.
GO TO 10
8 IF (RETM2 < -3) 9, 9, 10
9 BETM2 = 3
10 BET = 1.7 (BETM2 * BETM2)
WRITE (MOUT, 3) BETA
AGAM = .097 * BETM2 - 0.0497 * BETA
9GAM = .004565 * BETA
AH = .75 * 3.9 * BETM2 + 0.0974 - 0.0497 * BETM2
GAM = AGAM - BGAM * ZA
DERIV = UA * FRFR * EXP(-ZA) * GAM * GAM * (1. + BETAM + 1. + AH + BH * ZA) / (AH + BH + BR)
IF (DERIV = Y(Y, NY = 3)) 14, 12, 12
12 RY = RY + DRY
CALL YSET (RY, Y(2), NY, Y)
GO TO 11
14 IF(\beta=4.) 102,101,101
101 TERPA=1.-4.\beta
INDEX=3
GO TO 110
102 IF(\beta=2.) 104,103,103
103 TERPA=5.\beta-1.
INDEX=2
GO TO 110
104 TFRPA=\beta-1.
INDEX=1
110 K=0
TFRP1=1.-TERPA
50 K=K+1
GO TO (16,17,99),K
16 G=\gammaM
DELTA=DLS
UEDGE=UA
L=3
GO TO 18
17 G=\gammaM
DELTA=DEL
UEDGE=UA
L=2
18 XI=\gamma/(DELTA*RTR*\beta+M2)
UCW=RTR*(UEDGE*G)**2
EFRM=\gamma/BETM2
NLAM=NY
DO 75 N=2,NY
XI=Y(N)*XI
IF(XI<350) 20,19,19
19 UC(N,L)=UEDGE
GO TO 75
20 CALL TERPF(XI,INDEX,TAB1,TAB2,TAB3,TAB4,XITAB,FP1)
INDEX=INDEX+1
CALL TERPF(XI,INDEX,TAB1,TAB2,TAB3,TAB4,XITAB,FP2)
FP=TERPL*FP1+TERPA*FP2
UC(N,L)=UEDGE*(1.-EFRM*FP)
IF(N-NLAM) 21,75,75
21 ALTER=UCOW*Y(N)
IF(ALTER=UC(N,L)) 33,33,32
32 UC(N,L)=ALTER
GO TO 79
33 NLAM=N
CONTINUE
GO TO 50
99 DO 60 K=2,3
SAVE2=0.
DO 60 N=3,NY
SAVE1=UC(N-1,K)
UC(N-1,K)=(SAVE2+SAVE1+UC(N,K))/3.
60 SAVE2=SAVE1
DUDX=0.
C00=3.5/(XN-X1)
DO 65 N=3,NY
DUDXP=C00*(UC(N,2)-UC(N,3))
\[ V(N,1) = V(N-1,1) - (Y(N) - Y(N-1)) \cdot (U \cdot \Delta x^2 + \Delta x) \]
\[ V(N,2) = V(N,1) \]
65
\[ \Delta x \cdot \Delta x \cdot U \]
RETURN
END
SUBROUTINE ELPIT(ALPH1, ALPH2, EMI, TORF, THETZ, UINF, DXI, CMPA, CMPAS)  
SAVE=ALPH1  
STEP=TORF*DXI  
SINS=SIN(STEP)  
COSST=COS(STEP)  
CONST=2.*EMI*(UINF/TORF)**2  
ALPH1=THETZ+(ALPH1-THETZ)*COSST+ALPH2*SINS/TORF+CONST*(2.*CMPA-CMPA)  
SINS/SINS-STEP*COSST/TORF*DXI  
ALPH2=ALPH2*COSST-TORF*SINS*(SAVET-THETZ)+CONST*(CMPA-CMPA)*(1.-COEL)  
RETURN  
END
SUBROUTINE VWASH(BARG,H,S,NVOR,X1,UNIF,VZIP,XGA4,NGPL,DX1)

DIMENSION VZIP(30),XGA4(3)

DO 10 N=1,NGPL
DIFF=XGA4(N)-X1
SUM=0,
DO 5 K=1,NVOR
SUM=SUM+DIFF/(DIFF**2+H)
5 DIFF=DIFF-S
10 VZIP(N)=VZIP(N)+SUM*BARG
RETURN
END

VWASH 1
VWASH 2
VWASH 3
VWASH 4
VWASH 5
VWASH 6
VWASH 7
VWASH 8
VWASH 9
VWASH 10
VWASH 11
```
SLURoutine WASI(XGAM, NGAM, TIME, ALPH1, ALPH2, HEAVE, AROT, FREQF, PHIH, UWASI)
DIMENSION XGAM(30), VZIP(30), CAMR(24)
NGP1 = NGAM+1
ANGLE = FREQF*TIME
GO TO (108, 120), INDV
GO TO (110, 120), MOTR
110 CONST = -ALPH2*COS(ANGLE)*UINF+HEAVE*COS(ANGLE+PHIH)*ALPH1*UINF
FACT = -ALPH2*FREQF*SIN(ANGLE)*JINF
GO TO 130
120 CONST=UINF*ALPH1*HEAVE
FACT=-UINF*ALPH2
130 DO 10 M=1, NGP1
X=XGAM(M)
THETA = ARCT(X)
SUM=0.
DO 20 N=1, NF
COUNT=COUNT+1.
20 SUM=SUM+COUNT*CAMR(N)*CCS(COUNT*THETA)
IF(M-1) 2, 4, 2
2 IF(NGP1-M) 3, 4, 3
4 SUM = SUM + SUM
GO TO 50
3 COUNT = 0.
COUNT = COUNT*THETA
DO 30 N=1, NF
COUNT = COUNT*THETA
30 SUM=SUM+COUNT*CAMR(N)*SIN(COUNT)
50 VZIP(M) = UINF*SUM+CONST+FACT*(AROT-X)
10 CONTINUE
RETURN
END
```
APPENDIX B

DETERMINATION OF COUPLING PARAMETERS
APPENDIX B

DETERMINATION OF COUPLING PARAMETERS

The characteristic equation for the rotor blade is

$$\sum_{k=0}^{3} B_{2k} \lambda^{2k} = 0$$

where

$$B_0 = f_0 - \frac{\bar{\omega}^2 T_\beta \phi^2}{M_{\beta\phi} M_{\phi\phi}} - \frac{\bar{\omega}^2 T \phi^2}{M_{\phi\phi} M_{\phi\phi}}$$

$$B_2 = f_2 + 2 \frac{\bar{\omega}^2 M_\beta \phi T_\beta \phi}{M_{\beta\phi} M_{\phi\phi}} + 2 \frac{\bar{\omega}^2 M_\phi \phi T \phi}{M_{\phi\phi} M_{\phi\phi}}$$

$$+ \frac{T_\beta \phi^2}{M_{\beta\phi} M_{\phi\phi}} - \frac{T \phi^2}{M_{\phi\phi} M_{\phi\phi}}$$

$$B_4 = f_4 - \frac{\bar{\omega}^2 M_\beta \phi}{M_{\beta\phi} M_{\phi\phi}} - \frac{\bar{\omega}^2 M_\phi \phi}{M_{\phi\phi} M_{\phi\phi}}$$

$$+ 2 \frac{M_\beta \phi T_\beta \phi}{M_{\beta\phi} M_{\phi\phi}} + 2 \frac{M_\phi \phi T \phi}{M_{\phi\phi} M_{\phi\phi}}$$

$$B_6 = 1 - \frac{M_\beta \phi^2}{M_{\beta\phi} M_{\phi\phi}} - \frac{M_\phi \phi^2}{M_{\phi\phi} M_{\phi\phi}}$$
in which

\[ f_0 = \bar{\omega}_\beta^2 \bar{\omega}_\phi^2 \bar{\omega}_\theta^2 \]

\[ f_2 = \bar{\omega}_\beta^2 \bar{\omega}_\phi^2 + \bar{\omega}_\beta^2 \bar{\omega}_\theta^2 + \bar{\omega}_\phi^2 \bar{\omega}_\theta^2 \]

\[ f_4 = \bar{\omega}_\beta^2 + \bar{\omega}_\phi^2 + \bar{\omega}_\theta^2 \]

The characteristic equation for the two-dimensional system is found to be

\[ \sum_{k=0}^{3} D_{2k} \lambda^{2k} = 0 \]

where

\[ D_0 = f_0 - \bar{\omega}_\phi^2 h_a a_1^2 - \bar{\omega}_\beta^2 h_b b_1^2 \]

\[ D_2 = f_2 - \bar{\omega}_\phi^2 g_a \bar{x} a_1 - \bar{\omega}_\beta^2 g_b \bar{x} b_1 - h_a a_1^2 - h_b b_1^2 \]

\[ D_4 = f_4 - c_4 \bar{x}^2 - g_a \bar{x} a_1 - g_b \bar{x} b_1 \]

\[ D_6 = 1 - c_6 \bar{x}^2 \]

in which

\[ h_a = \frac{M_{\beta\beta}}{R^2 M_{\theta\theta}} \quad h_b = \frac{M_{\phi\phi}}{M_{\theta\theta}} \]
\( g_a = 2 h_a A_1 \quad g_b = 2 h_b A_2 \)

\[ c_4 = \bar{\omega}^2 h_a A_1^2 + \bar{\omega}^2 h_b A_2^2 \]

\[ c_6 = h_a A_1^2 + h_b A_2^2 \]

\[ a_1 = A_1 (\bar{\omega}^2 l_s_1 + r_m \bar{\omega}^2 l_s_2) - B \bar{\omega}^2 l_s_2 \]

\[ b_1 = A_2 (\bar{\omega}^2 l_s_1 + r_m \bar{\omega}^2 l_s_2) + B \bar{\omega}^2 l_s_2 \]

Equating \( D_0/D_6 \) to \( B_0/B_6 \), \( D_2/D_6 \) to \( B_2/B_6 \) and \( D_4/D_6 \) to \( B_4/B_6 \) provides three relations in the three unknowns \( x, l_s_1 \), and \( l_s_2 \). If \( a_1 \) and \( b_1 \) are eliminated, the following equation for \( x \) is obtained:

\[
(r_1 t_2 - r_2 t_1)^2 + (r_1 s_2 - r_2 s_1)(t_2 s_1 - t_1 s_2) = 0
\]

where

\[
r_1 = - \left[ h_a + \frac{h_b g_a^2}{g_b^2} \right] \quad r_2 = \left[ \frac{\bar{\omega}^2}{\bar{\omega}^2} - 1 \right] h_a
\]

\[
s_2 = (\bar{\omega}^2 - \bar{\omega}^2) g_2 x, \quad s_1 = s_2 + \frac{2 h_b g_a f}{g_b^2 x}
\]

\[
t_1 = (1 - c_6 x^2) B_2/B_6 - f_2 + \bar{\omega}^2 f + \frac{h_b f^2}{g_b^2 x^2}
\]
With some algebraic manipulation, a polynomial of fourth degree in $\bar{x}$ can be extracted from that equation. The value of $\bar{x}$ is taken to be the square root of the smallest positive root of that polynomial. The original equations are then used to solve for $a_1$ and $b_1$, from which $l_{s_1}$ and $l_{s_2}$ are readily obtained.

\[ t_2 = (1 - c_6 \bar{x}^2)(B_2 - B_0/\omega^2)/B_6 - f_2 + \omega^2 F + f_0/\omega^2 \]

in which
\[ F = f_4 - B_4/B_6 + (B_4 c_6/B_6 - c_4) \bar{x}^2 \]
REFERENCES


15. Theodorsen, T.; and Garrick, I. E.: Mechanism of Flutter--A Theoretical and Experimental Investigation of the Flutter Problem. NACA TR 685, 1940.