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

**Report Date**
November 1973

**Performing Organization Name and Address**
AVCO Systems Division
Wilmington, Massachusetts 01887

**Sponsoring Agency Name and Address**
National Aeronautics and Space Administration
Washington, D.C. 20546

**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**
Helicopter rotor, Aeroelasticity, Dynamic stall, Torsional stability

**Distribution Statement**
Unclassified - Unlimited

---

*For sale by the National Technical Information Service, Springfield, Virginia 22151*
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.
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.
SYMBOLS

\( b \) \hspace{1cm} \text{blade semichord, m}

\( \overline{C_L} \) \hspace{1cm} \text{mean lift coefficient, ratio of time average of } 1 \text{ to } \rho \Omega^2 R^2 b

\( C_1 \) \hspace{1cm} \text{lift coefficient, } C_1 = c_1 / (\rho U^2 b)

\( C_{m c/4} \) \hspace{1cm} \text{moment coefficient referred to quarterchord, } C_{m c/4} = m_{c/4}/(2 \rho U^2 b^2)

\( c \) \hspace{1cm} \text{blade chord, m}

\( f_\theta \) \hspace{1cm} \text{mode shape of first uncoupled torsional mode, unit tip deflection}

\( f_\phi \) \hspace{1cm} \text{mode shape of first uncoupled flapwise bending mode, unit tip deflection}

\( h_\beta \) \hspace{1cm} \text{tip deflection due to flapping, semichords}

\( h_\phi \) \hspace{1cm} \text{tip deflection due to bending, semichords}

\( h_i \) \hspace{1cm} \text{translational coordinates of 2-D system (} i = 1, 2), \text{ semichords}

\( I_0 \) \hspace{1cm} \text{moment of inertia of 2-D system about pitch axis, kg } \cdot \text{ m}

\( I_\theta' \) \hspace{1cm} \text{blade moment of inertia about elastic axis per unit span, kg } \cdot \text{ m}

\( k_i \) \hspace{1cm} \text{translational spring stiffnesses of 2-D system (} i = 1, 2), \text{ N/m}^2

\( k_\theta \) \hspace{1cm} \text{torsional spring stiffness of 2-D system, N/rad}

\( l \) \hspace{1cm} \text{lift per unit span at aerodynamic reference radius, N/m}

\( l_{s_1} \) \hspace{1cm} \text{offsets of springs from pitch axis of 2-D system (} i = 1, 2), \text{ m}

\( M_b \) \hspace{1cm} \text{total blade mass, kg}

\( m \) \hspace{1cm} \text{blade mass per unit span, kg/m}

\( m_{c/4} \) \hspace{1cm} \text{aerodynamic moment per unit span at aerodynamic reference radius, N}
\( m_1 \) masses of 2-D system, kg/m
\[ R \] rotor radius, m
\[ r_o \] inner radius of blade lifting surface, m
\[ r_R \] aerodynamic reference radius, m
\[ U \] instantaneous free-stream speed at aerodynamic reference section, m/sec
\[ U_o \] reference speed, \( U_o = \Omega \ r_R \), m/sec
\[ x_m \] distance aft of elastic axis of blade section mass center, m
\[ \bar{x} \] 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
\[ Z_\theta \] generalized coordinate of 2-D system, equivalent to \( h_\theta \), semichords
\[ a \] angle of attack, deg
\[ \delta \] flapping hinge offset, m
\[ \theta_o \] collective pitch angle, deg or rad
\[ \theta_t \] blade tip torsional deflection, rad
\[ \bar{\theta} \] 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/m³
\[ \tau \] dimensionless time, \( \tau = U_o \ t/b \)
\[ \psi \] blade azimuth angle measured from downwind direction, deg or rad
\[ \Omega \] rotor rotational speed, rad/sec
\[ \Omega^* \] dimensionless rotor speed, \( \Omega^* = \Omega \ R/(\omega \theta_o \ b) \)
\[ \omega_f \] 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
PROBLEM FORMULATION

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 lb, 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)

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

\[
= \frac{R b}{U_o^2} \frac{F_\beta}{M_\beta}
\]

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

\[
= \frac{b}{U_o^2} \frac{F_\phi}{M_\phi \phi}
\]

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

\[
- \frac{b}{R} \bar{\Omega}^2 \frac{2 T_\beta \theta \theta_1}{M_\theta \theta} h_\beta - \bar{\Omega}^2 \frac{b T_\theta \phi \theta_1}{M_\theta \theta} h_\phi
\]

\[
= \frac{b^2}{U_o^2} \frac{F_\theta}{M_\theta \theta}
\]
where $h_\beta$ and $h_\varphi$ 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:

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

$$\bar{\omega}_\theta^2 = \bar{\omega}_\theta^2 - \bar{\Omega}^2 \frac{T_{\theta\theta}}{M_{\theta\theta}}.$$

The inertial and centrifugal-force coefficients are given by

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

$$M_{\theta\theta} = \int_{\delta}^{R} I_\theta f_\theta^2 dr,$$

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

$$M_{\varphi\theta} = -\int_{\delta}^{R} m x_m f_\varphi f_\theta dr,$$

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

*Barred quantities are dimensionless frequencies, $U_0/b$ being reference frequency; e.g., $\bar{\Omega} = \Omega b/U_0$.  

12
The complexity of the aerodynamic representation precludes evaluation of the generalized forces $F_{\beta}$, $F_\theta$ and $F_\phi$ 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

$$Z_{\beta} = A_1 h_1 + B h_2, \quad Z_\phi = A_2 h_1 - B h_2$$

where

$$A_1 = \frac{\overline{\omega}_\beta^2 - \overline{\omega}_\phi^2}{\overline{\omega}_\phi^2 - \overline{\omega}_\beta^2}, \quad A_2 = \frac{\overline{\omega}_2^2 - \overline{\omega}_\phi^2}{\overline{\omega}_\phi^2 - \overline{\omega}_\theta^2},$$

$$B = \frac{(\overline{\omega}_2^2 - \overline{\omega}_\phi^2)(\overline{\omega}_2^2 - \overline{\omega}_\beta^2)}{(\overline{\omega}_\phi^2 - \overline{\omega}_\theta^2)\overline{\omega}_2^2}$$

(1)
Figure 2 TWO-DIMENSIONAL ELASTOMECHANICAL SYSTEM
and $\bar{\omega}_i^2 = (k_1/m_1)(b/U_0)^2$, $i = 1, 2$.

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

$$
\left( \frac{k_\theta + k_1 l_{s1}^2 + k_2 l_{s2}^2}{I_0} \right) \left( \frac{b}{U_0} \right)^2 = \bar{\omega}_\theta^2
$$

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

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

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

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

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

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

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

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

Equating the corresponding coefficients of the characteristic equations of the two systems provides three additional relations, which can be solved for the coupling parameters $\bar{x}$, $l_{s1}$, $l_{s2}$. 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

\[ m_1 \frac{R}{(-A_1)} = 4 \frac{r_R}{R} \frac{M_R}{R^2 [1 - (r_0/R)^4]} \]

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

where \( \Omega r_R = U_o \). 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

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

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 untwisted, 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 for Nominal Configuration</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>b/R</td>
<td>0.0435</td>
</tr>
<tr>
<td>δ/R</td>
<td>0.0543</td>
</tr>
<tr>
<td>r_o/R</td>
<td>0.174</td>
</tr>
<tr>
<td>ω_e0/ω_0</td>
<td>3.69</td>
</tr>
<tr>
<td>ρ 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_e'/M_b R</td>
<td>3.51 x 10^-4</td>
</tr>
</tbody>
</table>

Two elastomechanical configurations in addition to the nominal one were analyzed. One of these had ω_e0/ω_0 = 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 ω_0/Ω = 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 $C_m c/H$, 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 $\Omega^* = \Omega R / (\omega \theta_0 b) = 5.3$ with $\omega_f / \omega \theta_0 = .803$. 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 $\omega_f / \omega \theta_0 = .72$.

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 un stall 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 un stall 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

$\Omega^* = 5.3, \theta_0 = 11\,\text{deg}, \mu = 0$
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 1.1. 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

$\Omega^* = 4.8$, $\theta_0 = 13$ 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 ψ -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 ψ = 270 deg and ψ = 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 Ω* less than 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 (Ω* = 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

$\Omega^* = 4.15, \theta_0 = 12 \text{ deg}, \mu = 0.1$
Figure 14 DISPLACEMENT TIME HISTORIES FOR STALL FLUTTER AT LOW ROTOR SPEED
$\Omega^* = 3.89$, $\theta_0 = 14.3$ deg, $\mu = 0.1$
speed as those of Figure 12, but with $\theta_o$ 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 $X_m$ 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

\[
\bar{C}_L = \frac{\bar{1}}{\rho \Omega^2 R^2 b}
\]

where \( \bar{1} \) 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

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

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 $a = d\bar{C}_L/d\theta_o$ WITH ADVANCE RATIO
Figure 18 STALL STABILITY BOUNDARIES FOR $\Omega^* = 3.89$, $\omega_\theta_0 / \omega_\theta_0 = 3.69$ AND $X_m/b = 0.216$
$\bar{C}_L = .78$, 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 un stalled 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 $\omega_{\theta_0}/\omega_\phi = 2.5$. 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 $\omega_{\theta_0}/\omega_\phi = 3.69$.

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 -
\[ \Omega^* = 3.89, \bar{C}_L = 0.78, \mu = 0.3 \]
<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>TYPE OF SOLUTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>STALL FLUTTER</td>
</tr>
<tr>
<td>Δ</td>
<td>EXCESSIVE RESPONSE</td>
</tr>
<tr>
<td>O</td>
<td>STABLE (NO STALL)</td>
</tr>
</tbody>
</table>

**Figure 20** STALL STABILITY BOUNDARIES FOR $\Omega^* = 3.89$, $\omega_{\theta_0}/\omega_{\phi_0} = 2.5$
AND $Xm/b = 0.216$
Figure 21 STALL STABILITY BOUNDARIES FOR $\Omega^* = 3.89$, $\omega_\theta / \omega_\phi = 3.69$ AND $Xm/b = 0.108$
Figure 22  DISPLACEMENT TIME HISTORIES FOR EXCESSIVE TORSIONAL RESPONSE.

\[ \Omega^* = 3.89, \bar{C}_L = 0.95, \mu = 0.1, \text{ AND } X_m/b = 0.108 \]
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 /BL1/, NTIME, NDIMC, ISTD
COMMON /CLCHBL, CLVB, CMVR, CMPAVB

COMMON /INPTVB/, FTVB(64), FPVB(64), FPPRVB(64), DIDRVB(64), SETPS17
A X4VB(64), DELVB, XMUVB, FOVB, XMUAVB, SETPS18
B ATOVB, ATCVB, ATSVB, ROVB, RVB(64), SETPS19
C MVB(64), NVR

COMMON /INPUTS/, NSBL, NZ, NOFF, NGAM, NSIG, SETPS20
A NC01, NCOORD, LOWER, MSTOP, MXT, MTR, SETPS21
B NOTBL, INDV, ELSIG, DXI, REB, RRB, SETPS22
C FRZ, ARR, AMPLU, FREQU, ALPH1, ALPH2, SETPS23
D HEAVE, AROT, FREQV, PHIM, NY, RY1, SETPS24
E DRY, Y(100), TEST, UPRIM, XU(30), YU(30), SETPS25
F XL(30), YL(30), ER1, ER2, ER3, BD3R, SETPS26
G RRDRB
H, CMPA, CMPAS, BARG, EML, HVOR, NVR, SSPA, SVOR, TDF, X1VOR
I, PLOTOP, PSTLOW, PSTUP
J, NOUT
COMMON /ZZZ/ Z(3)

DIMENSION USAV(300,100), SCALS(300)
DIMENSION USAV(1,1), SCALS(300)
DIMENSION CAMBR(24), THICK(24)
DIMENSION XGAM(301), XSIG(100), XSIGA(100), XSIGB(100), XCI(300), X(300), MAIN 7
LSBL(300), XRSIG(100)
DIMENSION ACAP(30,3), BCAP(100,31), AS2(30), AS(30,30), BS(30,30), ASH2MAIN 9
1(100), ASH(30,30), BSH(30,30), AR(130), ARH(100), UE(300,3) MAIN 10
DIMENSION ALAM(30), VZIPI(30), FPPRES(100), GAMMA(1000), X(1000) MAIN 11
DIMENSION BLAM(30), FLAM(10), XFLAM(10) MAIN 12
DIMENSION SCALE(300,2), UC(1,1,1), UC(100,3), V(100,2) MAIN 13
M, P(200,7)

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

DATA IN, MOUT, NF/ 5, 6, 24/
DATA PI, TIME, UINF, TRENEL, USTOP/3.14159, 0., 1., 4.75E4, 2.8/
DATA FLAM /1.75, 1.75, 1.724, 1.527, 1.354, 1., 663., 452., 25/ MAIN 19
DATA XFLAM /-100., -11.26, -7., 0., -3.48, -1.766, 0., 1.888, 4./ MAIN 20
DATA 103., 6.77, 7.19/
DATA DEGRES /1.74, 53292, 51994, 330D-2/

EQUIVALENCE (CMAT(1), USAV(1)), (ASH(1), SCALS(1)) MAIN 16

IF ISTO =1 TIME DERIVATIVES NOT USED
ISTD = 1
RAD = 180. /PI
IL = 8888
NDIMC = 60
CALL SFTUPS
IF (ISTD .EQ. 1) GO TO 40
DO 100 J = 1,300
SCALJS(J) =0.
DO 100 I = 1,100
L =0
INDV=INDV+1
WRITE(MOUT,6)
PITCH = ALPH1
IF (INDV + MODR .LE. 2) PITCH = PITCH - ALPH2
X AMPLU = 1.33333*XMUAVB * (1. - ROVB**3)/(1. - ROVB**4)
IF (INDV .EQ. 2) FREQU = BDBR/KRDBR
IF (INDV .GE. 2) GO TO 343
WRITE(MOUT,25) NVOR,SVOR,HVOR,BARG,XIVOR,EMI,YORF,SSPA
RY=RY1
HVOR=HVOR+2
BARG=BARG/6.2832
CALL SECTK(XU,YU,XL,YL,NOFF,NF,ROBB,THCBB,CMDBG,THICK,CAMBR)
DO 7875 N=1,NF
CAMBR(N)=CAMBR(N)+CMDBG
THICK(N)=THICK(N)+TMDBG
WRITE(MOUT,4)
WRITE(MOUT,7) AMPLU,FREQU,ALPH1,ALPH2,HEAVE,AROT,FREQF,ROBB,REB
WRITE(MOUT,8)
WRITE(MOUT,9) (N,CAMBR(N),THICK(N),N=1,NF)
MX=NSBL+NS-1
CALL SCAL(SBL,NSBL,FRZ,ARR,ROBB)
CALL CORDX(NSBL,NZ,ROBB,SBL,XC1)
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
LE(MX+1,1) = 1.
EPSLF=2.*(N(NZ)-X(NZ-1))
PPSTF=X(MX)-X(MX-1)
ALTC=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.
2422 U(M,N,2) = 0.
50 CONTINUE
NSIGA=NSIG
NSIGA=NSIG
NSIGI=NSIG+1
M0TR=M0TR+1
M0TR=M0TR+1
XMAX=1.-ELSIG
CCNA=3.75*PI/DXI
ANGS=PI/FLOAT(NSIG)
CALL_SETSX(NSIG1,1,1,2, ,XSIG, ANGS)
XSEP=1.1
DO 2430 N=1,NSIG
XSIGA(N)=XSIG(N)
2430 XSIGA(N)=XSIG(N)
DO 2431 N=1,NSIG
DO 2431 N=1,3
2431 BCAPIN,NUI = 0.
PINT=2./FLOAT(NCORD)
NCIP=NCORD+1
THXI=1.5/DXI
NGPI=NGAM+1
NWMI=NWAKE-1
COUNT=0.
DO 8456 N=1,NWAKE
GAMAW(N) = 0.
XIW(N) = 1.+COUNT
8456 COUNT=COUNT+DXI
ANGLE=PI/FLOAT(NGAM)
COUNT=0.
DO 1002 M=1,NGP1
PHIM = COUNT*ANGLE
XGAM(M) = COS(PHIM)
COUNT=2.
DO 1001 N=1,NGAM
AS(M,N)=COS(COUNT*PHIM)
1001 COUNT=COUNT+1.
1002 COUNT=COUNT+1.
CALL_WASH1(XGAM, NGAM, TIME, ALPH1, ALPH2, HEAVE, AROT, FREQ, PHIM, UINF, CAMAIN
1MNR, NPI, ZIPPI, 11)
DO 8458 M=1,NGP1
CPAT(M,1) = 1.
TEMP=2.*ZIP(M)
RMAT(M) = TEMP
CMAT(M,2) = XGAM(1)
DO 8457 N = 3, NPG1
8457 CMAT(M, N) = AS(M, N - 1)
8458 CONTINUE
CALL ALSO(NGPI, CMAT, RMAT)
DO 8459 N = 1, NGPI
ACAP(N,1) = RMAT(N)
ACAP(N,3) = RMAT(N)
8459 ACAP(N,2) = ACAP(N,1)
DO 2784 M = 1, MX
SIGN = 1.
IF (M-NZ) 2774, 2775, 2776
2774 SIGN = -SIGN
2775 CALL QECELL(GP, NGAM, NSIG, NF, XSIG, ACAP, BCAP, THICK, RCRB, GAMAN(1), U1)
1NF, X(1), UF, N, SIGN, THICK, RCRB, GAMAN(1), U1)
2784 UF(M,Z) = UE(M, 1)
DO 1004 M = 2, NGAM
1004 BBLAM(M) = (1.125*XGAM(M)+1.875*(1.+XGAM(M)))/(1.3*XGAM(M))
14*XGAM(M))/((1.-XGAM(M))/DXI
BBLAM(NGPI) = -1.125/DXI
CALL CLCMNTCR(TTSFP, NGAM, NSIG, NSIGA, NSIGB, NSIGB, ACAP, RMAT
1AP, THICK, RCRB, GAMAN, U1NF, UD0T, DXI, AR0T, CMP4)
IF (INDV .EQ. 2) CALL SUPPL
C INDEXING IN TIME IS CARRIED OUT AT THIS POINT.
C CONTINUE
C CALL ACUCPU (IACU )
IF (IACU .LT. 35000 ) GO TO 99
C
C NOTE - FOR READ-IN CF FCIL MOTIONS, MAKE ALPH1 = ALPHA,
C ALPH2 = ALPHA-DOT, AND HEAVE = H-DOT.
C
C IF (MCTR .EQ. 2)
XREAD(IN, 2, EN=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 (MAXT - NTIME) 8989, 8800, 8800
8800 SAVFU = U1NF
L = L + 1
PIL (1) = BCRB / (RCRBR * TIME * RAD
PSI360 = AMOD (PIL, IT, 360)
U1NF = 1.0 + AMPLU * SIN (FREQ * TIME)
IF (INDV .EQ. 2) CALL SUPPL (U1NF)
PITCH = ALPH1
IF (INDV + MCTR .LE. 2) PITCH = PITCH - ALPH2 * COS (FREQ * TIME)
UD0T = FREQ - AMP * COS (FREQ * TIME)
STEPX = 5 * DXI * (U1NF + SAVFU)
DO 1003 JZ = 2, NWAKE

51
JC=NWAKE-J+2
GAMAW(JC)=GAMAW(JC-1)
1003 XIW(JC)=XIW(JC-1)+STFPX
IF (ISEP) 2009,2009,2007
2007 D2 2008 N=1,NSIG
RCAP(N,3)=RCAP(N,2)
2008 RCAP(N,2)=RCAP(N,1)
DO 4433 N=1,NSIG1
XSIG(N)=XSIG(N)
4433 $XSIG(N)=XSIG(N)$
GO TO 2010
2009 DEADL=0.
ELODT=UNF
210 DO 1014 M=1,MX
UE(M,3)=UE(M,2)
1014 UE(M,2)=UE(M,1)
DEADL=0DEADL
ELODT=ELODT
ALAM(1)=(1.125+.75*ALOG(STEPX*.5))/DXI
1005 ALAM(M)=BLAM(M-1)+1.125+.75*ALOG(STEPX*.5)/DXI
11/1-1-XGAM(M))/DXI
DC 2006 M=1,NGP1
ACAP(M,3)=ACAP(M,2)
2006 ACAP(M,2)=ACAP(M,1)
ACFACT=8.0*(ACAP(1,2)+.5*ACAP(2,2))-2.0*(ACAP(1,3)+.5*ACAP(2,3))
ALPHS=ZVIP(J)
CALL WASH(XGAM,NGAM,TIME,ALPH1,ALPH2,HEAVE,AROT,FREQF,PHIH,UNIF,CAMAIN
IMBR,NF,ZVIP,MODT,INDV)
DO 1006 M=1,NGP1
ASZ(M)=1.+2.*ALAM(M)
ASZ(M,1)=XGAM(M)+ALAM(M)
SUM=0.
DO 4343 J=2,NWM1
4343 SUM=SUM+(GAMAW(J+1)-GAMAW(J))*(XGAM(M)-XIW(J))/(XIW(J+1)
1-XIW(J))*ALOG((XIW(J+1)-XGAM(M))/(XIW(J)-XGAM(M)))
1-1-XGAM(M))/XIW(J)
ELX1=1.-XGAM(M)
IF (M=1) 1006,2130,1006
2130 ELX=1.
1006 AR(M)=2.*ZVIP(M)+ALAM(M)+ACFACT/3.+SUM-GAMAW(2)*(1.-XGAM(M))*ALOG(MAIN
11.1-STEPX-XGAM(M)))/ELX1
DO 1006,2130,1006
C
C THE FOLLOWING CALCULATIONS, THROUGH STATEMENT 4444, ARE PERFORMED
C ONLY IF THE AIRFOIL IS STALLED. THE AIRFOIL IS DESIGNATED TO BE
C STALLED IF INTEGER ISEP IS NONZERO.
C
C IF (ISEP) 3247,3247,3247
3247 GO TO 3344
3344 XSEP*XSEP+DXI
IF (XSEP-XMAX) 3248,3249,3247
3347 XSEP=2
ISEP=0
XSEP=I,1
DO 3015 K=1,3
3015 N=1,NSIG

52
3015 RCAP(N,K)=0.
GO TO 4444
3345 IF(INOT) 3349,3348,3248
3349 IF(NITS-1) 3248,3349,3248
3349 IF(INOT.EQ.2) GO TO 6349
IF(VIP(1)-ALPHS) 6349,6348,6348
6348 NITS=2
GO TO 3248
6349 CALL UNDPINGAM,AR,ALAM,AFAC,RM,CMAT,CMAT,XMAG,AS,ACAP,4X,NZ,IF,XSIG
MAIPJ
MAIN 251
MAIN 252
MAIN 253
MAIN 254
MAIN 255
MAIN 256
MAIN 257
MAIN 258
MAIN 259
1,RAPI,THICK,RORB,UINF,XC,UE
GO TO 2785
3248 XATT=XSEP+DEADL+.5*(EL01+ELDOT)*XI
DEADL=XATT-XSEP
DIFF=1.-XATT
XTEST = XSEP + 3. * EPSLE
CALL SETS(XSIG1,XSEP,XATT,XSIG,ANGS)
GO TO 2785
4434 XBSIG(N)=-.5*(XSIG(N)+XSIG(N+1))
DO 3086 M=1,NGP1
DO 3086 N=1,NSIG
3085 RSM(N)=0.
DO 3087 M=1,NGP1
IF(XGAM(M)-XSEP) 3088,3087,3086
3087 XGAM(M)=XSEP
IF(XGAM(M)-XSIG(1)) 3093,3092,3088
3088 RSM(N)=0.
GO TO 3087
3093 MARK=1
GO TO 3094
3192 CONTINUE
3194 CSK=XSIG(MARK)-XSIG(MARK-1)
RSM(MARK-1)=(XSTG(MARK)-XGAM(M))/WIDES
RSM(MARK)=(XGAM(M)-XSIG(MARK-1))/WIDES
RSM(M) = SQRT((XGAM(M)-XSEP)/(XATT-XGAM(M)))
3088 IF(DIFF<1.5-6.) 3087,3098,3099
3195 RSM(M) = DIFF**(-1)*SQRT(DEADL)*(2.+DIFF*(SQRT((1.-XGAM(M))/XATT-XGAM(M))))
GO TO 3087
3197 RSM(M) = DIFF**(-1.5)*SQRT(DEADL)*(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
IF(XRSIG(K)-1.) 4348,4349,4349
4348 CSK=XSIG(K)
SINK=SQR1(-COSK*COSK)
THEK=ARC1(COSK)
TANT=SIGN(5.*THEK)/COS(K..5.*THEK)
ASHZ(K)=TANT*CONA(1.*COSK)**(1.3.*COSK)/UINF**THXI*(PI+THEK+SINK)
ICONA(1.*COSK)*SINK**2./UINF
ASHK,T1=5.*(ASHZ(K)-TANT1)*SINK
COUNT=1.
DO 4355 N=2,NGAM
COUNT=COUNT+1.
4355 ASHK,N=SIGN(COUNT+THEK+75.*SIGN(COUNT+1.)*THEK1/(COUNT+1.))-S1
MAIN 290
MAIN 291
MAIN 292
MAIN 293
MAIN 302
MAIN 294
MAIN 295
MAIN 296
MAIN 297
MAIN 298
MAIN 299
MAIN 300
MAIN 301
MAIN 302
MAIN 303
MAIN 304
IN((COUNT-1.)*THETK)/(COUNT-1.))/(DXI*UINF)
GO TO 4350

4349 ASH2(K)=0.
DO 4359 N=1,NGAM

4355 ASH2(K,N)=0.
DO 4360 N=1,NGAM

4362 CONTINUE

IF(DIFF-1.E-6) 5005,5006,5006

5005 PREC=0.
GO TO 5007

5006 CALL ATTIR(PREC,XSIG,NSIG,ASZ,AS,AR,CMAT,RMAT,NGAM,NSIG,INOT,DEL1,THETM)
11,REF,USEP,X4,CP1)
CMAT=CMAT(H)*ASZ(M)
DO 4360 N=1,NGAM

CMAT2=CMAT2(H)*ASZ2(M)
DO 6481 N=1,NGAM

IF(SEP) 6486,6500,6486

6486 DO 6499 N=1,NSIG
NGG=NGG+NGPI

6499 CONTINUE

TF(TSEP) 6502,6501,6502
NTO=NGPI
GO TO 6751

6501 CONTINUE

6502 DO 6750 K=1,NSIG

6750 CONTINUE

6751 CONTINUE

6758 DO 6748 K=1,NSIG

6748 CONTINUE

C C C
C C C C C
C C
8562 CALL RUBR(DEL,THETI,RFB,XSEP,USEP,X5,DCP,DEL5,X,XC,MX,NZ,X5,U5,MAIN 416
    IF/YLT,RFNEL,USTOP)
    USEP=USEP*0.2046*USEP*3
    PDIFF=(USEP-USEP+USEP+USEP)
    WRITE(MO,22) PDIFF,DCP
    IF(DCP=PDIFF) 8263,8365,8366
8263  ISEP=0
    GO TO 8463
8366 IF(ISEP) 8368,8369,8369
8369 IF(ISEP=0) 8467,8467,8368
8467 IWSH=1
    NITS=2
    GO TO 3344
8368 GO TO (8169,1786),NOTRA
8169 CALL RFATTUC(V,X,Y,MX,NY,RY,DY,UE,XS,DEL5,MST,REB)
    LAMQ=0
    GO TO 8367
8463 IF(ISEP) 7741,7741,7742
7741 ISEP=1
    NITS=NITS+1
    IF(NORD) 7743,7743,7643
7643 ISEP=1
    OSXEP=XSEP
    OSXEP=6*XSEP+.4
    CALL CPC(ISEP,NGAM,NF,NSIG,NSIG,NSIGA,NSIGB,NSIGA,NSIGB,ACAP,BCAP,MAIN 440
      LTHICK,RDRD,GMAM,UNINF,UDOT,1,XSEP,DXI,CPL)
    GO TO 3248
7742 CALL FLDER(BCAP,XSIG,NSIG,UNINF,ELDOT,NSIG2,NSIG2,NSIG2,NSIG2)
    IF(ISEP).EQ.1.AND.ISEP.EQ.0.AND.NITS.EQ.1) GO TO 9210
    IF(ISEP=1) 7841,7842,7842
7841 EPS=EPSLE
    GO TO 7843
7843 EPS=EPSLE
    GO TO 7843
    IF(DXSEP-EPS) 7834,7834,9210
7834 IF(ISEP-XMAX) 1786,1786,7835
7635 ISEP=0
    ISEP=0
    GO 7836 K=1,3
    GO 7836 N=1,NSIG
7836 BCAP(N,K)=0.
    GO TO 1786
9210 NITS=NITS+1
    IF(NITS.EQ.2.AND.NORD.EQ.0) XSEPS=XSEP
    IF(NORD) 9211,9211,1786
9211 IF(ISEP-XSEPS) 9335,9305,9306
9305 XSEP=6*XSEPS+.4*XSEP
    GO TO 9307
9307 XSEP=6*XSEPS+.4*XSEPS
9307 IF(ISEP-XMAX) 9212,9212,7835
9212 CALL CPC(ISEP-NGAM,NF,NSIG,NSIG,NSIGA,NSIGB,NSIGA,NSIGB,ACAP,BCAP,MAIN 440
      LTHICK,RDBB,GMAM,UNINF,UDOT,1,XSEP,DXI,CPL)
    IF(NORD) 9212,9212,9212
7743 IF(NORD-1) 7737,7737,3248
    GO TO 3248
    GO TO 3248
DECLARE INT N1,N2
DECLARE REAL X,Y,Z

IF (N1 .GE. 0.1) READ(YOUT20) XSIG(1)

WRITE(10,9001) Z,ALDEG,ALPH1,ALPH2,HEAVE
IF (PSI1360.GE.0.1) READ(YOUT912) XSIG(N)

IF (PSI1360.LE.0.1) READ(YOUT912) XSIG(N)

DO 7102 NS=NCPL
CALL OECALHSEP,NGAMHNF*XSIGNS
IGSIGNSIGBACAP,BCAP,THICKRCRYA

7102 CONTINUE
CALL OECALHSEP,NGAMHNF*XSIGNS
IGSIGNSIGBACAP,BCAP,THICKRCRYA

IF (N-1) 7546,7545,7546

7485 NOUT=NOUT+1
7486 WRITE(MOUT,20) NTIME
7487 WRITE(MOUT,7) NIVOR
7488 PITCH=PITCH+1.0
7489 WRITE(MOUT,10) TIME,UNINF,XSEP,XATT,PITCHNIVOR
7490 NIVOR=ALPH1/DEGRFS
7491 WRITE(6,9001) Z,ALDEG,ALPH1,ALPH2,HEAVE
IF (PSI360.GE.0.1) READ(YOUT912) XSIG(N)

IF (PSI360.LE.0.1) READ(YOUT912) XSIG(N)

DO 7102 NS=NCPL
CALL OECALHSEP,NGAMHNF*XSIGNS
IGSIGNSIGBACAP,BCAP,THICKRCRYA

7102 CONTINUE
CALL OECALHSEP,NGAMHNF*XSIGNS
IGSIGNSIGBACAP,BCAP,THICKRCRYA

IF (N-1) 7546,7545,7546

7545 CPL=CPU
7546 DLIFT=CPL-CPU
7547 WRITE(MOUT,16) XPC,QEL,CPUCPL,GEU,CPU,DLIFT
7548 XPC=XPC+QEL
7549 P(L2)=PITCH
P(L3)=Z(3)

P(L6)=CLVAR
P(L7)=CMPA

IF (L .LT. 200) GO TO 98

CALL PLTSB(PLOTNP,P,L)

L=0

98 CONTINUE

IF (ISTN.EQ.1) GO TO 9999

DO 7950 N=1,M
SCALE(M,2)=SCALE(M,1)

7950 CONTINUE
SCALE(M,1)=SCAIS(M)
MAIN 508
DC 7950 N1,N2
MAIN 509
U(M,N,2)=U(M,N,1)
MAIN 510
7550 U(M,N,1)=USAV(M,N)
MAIN 511
GO TO 9999
MAIN 512
8589 CONTINUE
MAIN 513
99 CONTINUE
MAIN 514
     CALL PLOTTB( PLOTOP , P , L )
MAIN 515
     CALL ACUCPU1 , IACU , 1
MAIN 516
     IF ( IACU .LT. 35000 ) GO TO 60
MAIN 517
     GO TO 40
MAIN 518
60 CONTINUE
MAIN 519
     IF ( PLOTOP.EQ. 0 .) CALL EXIT
MAIN 520
     CALL PLTNOD
MAIN 521
     CALL EXIT
MAIN 522
     RETURN
MAIN 523

C
MAIN 23
C
C
1  FORMAT(1315)
MAIN 24
2  FORMAT(3F10.4)
MAIN 25
3  FORMAT(2F10.4)
MAIN 26
4  FORMAT(1H1//1)
MAIN 27
5  FORMAT(6F10.4)
MAIN 28
6  FORMAT(1H1,50X,34HANALYSIS OF UNSTEADY AIRFOIL STALL///)
MAIN 29
7  FORMAT(8X,6HUMBAR =E13.5/7X,7HUFREQ =E13.5//3X,11HALPHA ONE =E13.5/MAIN 30
1E13.5/7X,6HRO/9 =E13.5///9X,5HREP =E13.5///)
MAIN 32
8  FORMAT(29X,1HN,25X,4HC(N),26X,4HT(N)/)
MAIN 33
9  FORMAT(30,2E30.5)
MAIN 34
1X,4HPA =E13.5///7)
MAIN 36
11 FORMAT(4X,1HN,1IX,1HX,14X,SHVZ(X),12X,5HRN(X),12X,4HAI(N),21X,3HMAIN 37
IIX,14X,5HGMMA/)  
MAIN 38
12 FORMAT(15,4E17.5,8X,2E17.5)
MAIN 39
13 FORMAT(1H1,8X,1HN,20X,1HX,21X,5HFPI(X),22X,5HRM(N),21X,4HBN(N)/)
MAIN 40
14 FORMAT(54,9H,9DCT =E13.5///5X,27HMPRESSURES IN SEPARATED FLOWMAIN 41
1/55X,14X,19X,2HCF/)
MAIN 42
15 FORMAT(1H1,11X,1MX,16X,3HQEL,15X,3HCPL,15X,3HQUEU,15X,3HCPU,13X,9HCMAIN 43
MP - CPU/)
MAIN 44
16 FORMAT(6E18.5)
MAIN 45
17 FORMAT(10,4E25.5)
MAIN 46
18 FORMAT(140X,2E20.5//))
MAIN 47
19 FORMAT(15,5F10.4)
MAIN 48
20 FORMAT(1H1,50X,12HTIME STEP NO13//)
MAIN 49
21 FORMAT(1H1,50X,12HTIME STEP NO13//)
MAIN 50
22 FORMAT(//40X,26HINCREASE IN CP REQUIRED 15E13.5///40X,26HINCREASE MAIN 51
1IN CP POSSIBLE ISE13.5)
MAIN 52
23 FORMAT(///45X,23HPOTENTIAL FLOW XS =E12.4//60X,8HCFPI(XS) =E12.4/MAIN 53
1/45X,23BOUNDARY LAYER XS =E12.4/)
MAIN 54
24 FORMAT(5,4F10.4//5F10.4)
MAIN 55
25 FORMAT(12X,4HNV =12,3X,3HS =E12.4,3X,3HM =E12.4,3X,3HG =E12.4,3MAIN 56
X)
MAIN 57
26 FORMAT(4H4X,1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 58
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 59
28 FORMAT(4H4X,1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 60
29 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 61
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 62
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 63
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 64
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 65
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 66
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 67
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 68
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 69
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 70
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 71
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)}
MAIN 72
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 73
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 74
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 75
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
MAIN 76
27 FORMAT(1H1 =E12.4//12X,4HMI =E12.4,3X,4HNY =E12.4,3X,4HPA =E12.4/7)
9CC1A  //  T5,  'FLAP DISP =',  G14.5
9CC1P  ,  T47,  'BENDING DISP =',  G14.5
9CC1C  ,  T39,  'TORSONAL DISP =',  G14.5
9CC1D  /  T38,  'SECTION PITCH ANGLE =',  F9.3,  'DEGREES OR',  SUPPL384
9CC1E  /  T38,  'SECTION PITCH ANGLE =',  F9.4,  'RADIANS'
9CC1F  /  T21,  'SECTION PITCH RATE =',  G14.5
9CC1G  ,  T71,  'SECTION PLUNGING RATE =',  G14.5  //
9CC1H  /  T38,  'SECTION PITCH ANGLE =',  F9.3,  'DEGREES OR',  SUPPL385
9CC1I  ,  T47,  'BENDING DISP =',  G14.5
9CC1J  ,  T39,  'TORSONAL DISP =',  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

C

REAL*4 A, B, C, D

1 DUMMY, PLOTOP
REAL FTVB, FPVB, FPPRVB, DIDRVB, XMVB, DELVB, XMUVB, SUPPL 1
A FOVB, XMUAVR, ATOVB, ATCVB, ATSVB, ROVB, RVB, VVB, SUPPL 2
C W0XI, PSI, UNIF
REAL ELSIG, DXI, REB, RDBR, FRZ, ARR, AMPLU, FREQU,
A ALP1H, ALPH2, HEAVE, AR1T, FREQF, PH1H, RY1, DRY, SUPPL 3
B X, TEST, UPRIM, XU, YU, XL, YL, ER1, ER2, ER3, RDBR, SUPPL 4
C RDBR
REAL SUM(8), YCLD(8), YNEW(8), DEL(3,3), CMPA(3), CL(3), S(3), SUPPL 5
A Z, ZPR(3), SMALLG(3), Y(3,3), YPR(3,3), GCAP(3,3), SUPPL 6
C

COMMON /BL1/ NTIME, NDIMC
COMMON /CLCMHL/ CLVB, CMVB, CMPAVB
COMMON /Z27/ Z(3)
COMMON /INPTVB/ FTVB(64), FPVB(64), FPPRVB(64), DIDRVB(64), SUPPL 7
A XMVB(64), DELVB, XMUAVB, FOVB, XMUAVB, SUPPL 8
B ATOVB, ATCVB, ATSVB, ROVB, RVB, VVB, SUPPL 9
C MVB(64), NVB
COMMON /INPUTS/ NSBL, NZ, NOFF, NGAM, NSIG, SUPPL 10
A NCOI, NCORD, LOWER, MSTOP, MAXT, MJTR, SUPPL 11
B NOTBL, INDV, ELSIG, DXI, REB, RDBR, SUPPL 12
C FRZ, ARR, AMPLU, FREQU, ALPH1, ALPH2, SUPPL 13
D HEAVE, ARCT, FREQF, PH1H, NY, RY1, SUPPL 14
E DRY, X(100), TEST, UPRIM, XU(30), YU(30), SUPPL 15
F XL(30), YL(30), ER1, ER2, ER3, BDBR, SUPPL 16
G RDBR
H, DUMMY(10), PLOTOP
DIMENSION DELTA(3,3), SUPPL 17
DIMENSION ALPHA(3,3), BETA(3,3), GAMMA(3,3), OMS(3), OMEGA(3), C(3,3), SUPPL 18
DIMENSION AA(10), AB(10), ANB(20), ANT(20), AAX(10), ANSX(20), SORT(3) SUPPL 19

1, NT(2)

CF4(X) = F4 - B4 + (B4 * C6 - C4) * XX
SUPPL 20
Z1(X) = H1 * (CF4(X) / GB) ** 2 * (CF4(X) * FR1S * (1 - C6 * XX) * GB - F2) * XX
SUPPL 21
Z2(X) = F2 / FR1S * FR1S * CF4(X) - F2 * (1 - C6 * XX) * (GB - GB) / FR1S * XX
SUPPL 22
SUPPL 23
SUPPL 24
SUPPL 25
SUPPL 26
SUPPL 27
SUPPL 28
SUPPL 29
SUPPL 30
SUPPL 31
SUPPL 32
SUPPL 33
SUPPL 34
SUPPL 35
SUPPL 36
SUPPL 37
SUPPL 38
SUPPL 39
SUPPL 40
SUPPL 41
SUPPL 42
SUPPL 43
SUPPL 44
SUPPL 45
SUPPL 46
SUPPL 47
SUPPL 48
SUPPL 49
SUPPL 50

C

DATA BB, REL, NPOL/1.1, 1.4-7, 1.1-6.3/

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 51
C AND S2LB ARE FRACTIONS OF SEMICHORD. XBAR, SIL, AND S2L ARE

C FRACTIONS OF ROTOR RADIUS.

C

ADIMC = 3
DO 63 K = 1, 8
SUPPL 52
SUM(K) = 0.
SUPPL 53
63 YNEW(K) = 0.
SUPPL 54
DO 69 I = 1, NVB
SUPPL 55

60
DO 66 K = 1, 8
YOLD(K) = YNEW(K)
CALL YB3(YNEW, I)
IF (I .LE. 1) GO TO 69
DO 67 K = 1, 8
SUM(K) = (YNEW(K) + YOLD(K)) * (RVRA(I) - RVB(I-1)) / 2. + SUM(K)
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 / RRDBR
BDS = BDBR ** 2
T11 = H11 * BDS
T22 = H22 * BDS
T33 = H33 * BDS
T13 = H13 * BDS
T23 = H23 * BDS
FR1S = BDS * ER1**2 - T11 / EH11
FR2S = ER2**2 * BDS - T22 / EH22
FR3S = FR3**2 * BDS - T33 / EH33
FR1 = DSQRT(FR1S)
FR2 = DSQRT(FR2S)
FR3 = DSQRT(FR3S)
RATH = EM11 / EM22
ZETA = (1. + RATH) * (RATH * FR1S + FR2S ** 2) / (RATH * FR1S + FR2S) ** 2
RM = ZETA ** 1.
SUMS = FR1S + FR2S
HIGHS = (SUMS + DSQRT(SUMS ** 2 - 4. * ZETA ** FR1S ** FR2S)) / (2. * ZETA)
SMALS = FR1S + FR2S / HIGHS
DEN = FR2S - FR1S
A1 = (HIGHS - FR1S) / DEN
A2 = 1. - A1
B = A1 * DEN / HIGHS
SLAMB = EM11 * BDBR / EM33
SLAMZ = - A1 * SLAMB
SUM3 = SUMS + FR3S
ADDZ = FR1S * FR2S * FR3S + FR2S * FR3S
ADD2 = FR1S * FR2S * FR3S
HB3R = 1. - (EM13**2 * EM11 * EM23**2 / EM22 / EM33)
IEM13**2 / EM11**2 / EM33
B4 = A4 / RBAR
I23**2 / EM22**2 / EM33
B2 = HZ / RBAR
B2 = ADDZ * (FR2S * T13**2 / EM11**2 + FR1S * T37 / EM22**2 / EM33
HZ = BZ / RBAR

C6=(EM11*A1**2*EM22*A2**2)/EM33
F4=SUM3
C4=(FR2*S*EM11*A1**2*FR1*S*EM22*A2**2)/EM33
GA=2*(EM11*A1/EM33)
GB=2*(EM22*A2/EM33)
F2=ADD2
HA=EM11/EM33
HB=EM22/EM33
FZ=ADD2
R1=-HA-HR*(GA/GR)**2
R2=HA*(FR2S/FR1S-1)
ZLAM=F4-B4
TILAM=HA*(C6-C4)
FZHAT=HB*(TILAM/GB)**2
F2HAT=R2-FZ+FR1S*ZLAM+2.07LAM*TILAM*HB/GB**2
F4HAT=C6**2*FR1S*TILAM*HB*(TILAM/GB)**2
G2HAT=R2-FZ+IFZ-BZ)/FR1S*FRIS*ZLAM
G4HAT=C6**2*FB-BZ)/FR1S*TILAM
SIG2=2.0*HB*ZLAM/GB**2
SIG2=GA*(FRIS-FR25+2.0*HB*TILAM/GB**2)
GAM2=GA*(FRIS-FR25)
UZ=-R2*FZHAT
U1=R1*G2HAT-R2*F2HAT
U2=R1*G4HAT-R2*F4HAT
U3=-R2*SIG2
U4=R1*GAM2-R2*SIG2
U5=SIG2*G2HAT-GAM2*FZHAT
U6=SIG2*G4HAT-SIG2*G2HAT-GAM2*F2HAT
U7=SIG2*G4HAT-GAM2*F4HAT
AAX(1)=U2**2
AAX(2)=U2*U1+U3*U5
AAX(3)=U1**2+2.0*U2+U3*U6+U4*U5
AAX(4)=U2**2+U2*U4+U5
AAX(5)=U2**2+U4*U5
CALL POLLY(4,RBS,RCS,ANSX,AAX)
XBAR=1.25
DO 86 I=1,4
IP=2*I
IM=IP-1
IF(ABS(ANKX(I*IM)).GT.1.D-10) GO TO 86
IF(ANKX(IP).LE.0.) GO TO 86
XBAR=DSQRT(ANKX(IP))
IF(XBAR.LT.XBAR) XBAR=XBAR
CONTINUE
86 IF(XBAR.LT.5E25) Go To 88
WRITE(16,87)
STOP
87 FORMAT(H11.1,0X,0C,SOLUTION FOR XBAR)
STOP
88 CONTINUE
15 ALow=(R1+Z2(XBAR)-R2+Z1(XBAR))/(R1*S2(XBAR)-R2*S1(XBAR))
ALow=ALow/XBAR
BLOW=(CF6(XBAR)-GA*ALow/XBAR)/(XBAR*GB)
XI=ALow-BLOW
ETA=(BLOW*A1-ALow*A2)/(A1-A2)
S2L=TA/(B*RTGHS)
SIL=(XI-RM*HIGHS*S2L)*HIGHS/(FR1S*FR2S)
WRITE(6,4)FRL,ER2,EF3,RF
WRITE(4,721)FRL,FR2,FR3,ALOW,BLOW
WRITE(6,5)EM11,EM22,EM33,EM13,EM23
WRITE(6,6)H11,H22,H33,H13,H23
C13=ALOW/BDIR
C23=BLOW/BDIR
XBAR=XBAR/BDIR
SILR=SIL/BDIR
S2LR=S2L/BDIR
WRITE(6,7)XBAR,X9AB,SIL,SILB,S2L,S2LB,SMAL,S,HIGHS
AA(1)=B7
AA(2)=B2
AA(3)=R4
AA(4)=1.
CALL POLLY(NPOL,BRS,REL,ANB,AA)
SSX=SLAMZ*XBAB
DIV=1.-SLAMZ*XBAR**2
BETA(3,1)=(SLAM1*C13*SSX*FR1S)/DIV
BETA(3,2)=(SLAM2*C23*SSX*FR2S)/DIV
BETA(3,3)=(FR3S*SSX*(C13+C23))/DIV
AXB=A1*XBAB
BETA(1,1)=FR1S-AXB*BETA(3,1)
BETA(1,2)=AXB*BETA(3,2)
BETA(1,3)=C13-AXB*BETA(3,3)
AXR=AX*XBAB
BETA(2,1)=A1*AXB*BETA(3,1)
BETA(2,2)=FR2S-AXB*BETA(3,2)
BETA(2,3)=C23-AXB*BETA(3,3)
AB(4)=1.
AB(3)=BETA(1,1)+BETA(2,1)+BETA(3,3)
AB(2)=BETA(1,1)*(BETA(2,2)+BETA(3,3)) + BETA(2,2)*BETA(3,3) - BETA(3,2)*BETA(3,3)
AB(1)=BETA(1,1)*(BETA(2,2)+BETA(3,3)) + BETA(2,2)*BETA(3,2)*BETA(3,3)
AB(0)=BETA(1,1)*BETA(3,2)*BETA(3,3)
IF(TA(2,3)=BETA(1,3)*BETA(2,2))
CALL POLLY(NPOL,BRS,REL,ANT,AB)
WRITE(6,44)
DO 45 I=1,4
45 WRITE(6,46)IM,AA(I),AB(I)
WRITE(6,47)
DO 48 I=1,3
48 WRITE(6,49)ANB(TTT),ANB(TTM),ANT(TTT),ANT(TTM)
DO 301 I=1,3
301 OMS(I)=ANT(I)
MAX=3
DO 70 I=1,2
70 IF(OMS(I).GE.OMS(MAX)) MAX=I
CONTINUE
GO TO (71,72,73),MAXI
361 IF NUMS(I1).GT.OMS(I2) GO TO 75
    MINI=1
    MIDI=2
    GO TO 76
75    MINI=2
    MIDI=1
    GO TO 77
76    SORT(1)=OMS(MINI)
    SORT(2)=OMS(MIDI)
    SORT(3)=OMS(MAXI)
    DO 77 I=1,3
    OMS(I)=SORT(I)
    DO 302 I=1,3
302    OMEGA(I)=DSQRT(OMS(I))
362    DO 381 I=1,3
    ALPHA(I,1)=1.
    DENB=RETA(2,1)*BETA(3,2)-BETA(3,1)*BETA(2,1)*OMS(I1)
    ALPHAS(1,2)=(BETA(1,2)*BETA(3,1)-BETA(3,2)*BETA(1,1)*OMS(I1))/DENB
    ALPHAS(1,3)=((BETA(2,2)*OMS(I1))*BETA(1,1)-OMS(I1)*BETA(1,2)*BETA(2,2)/DENB)
    CHKS(1)=BETA(1,3)*ALPHA(1,1)+BETA(2,3)*ALPHA(1,2)+BETA(3,3)*OMS(I1)
    1)*ALPHA(1,3)
    ALPHAS(2,1)=BETA(1,2)*BETA(3,1)-BETA(3,2)*BETA(1,1)*OMS(I1)/DENB
    ALPHAS(2,2)=BETA(1,2)*BETA(3,1)-BETA(3,2)*BETA(1,1)*OMS(I1)/DENB
    ALPHAS(2,3)=BETA(1,2)*BETA(3,1)-BETA(3,2)*BETA(1,1)*OMS(I1)/DENB
    CHKS(2)=ALPHA(1,3)*ALPHA(2,1)+BETA(2,3)*ALPHA(2,2)*BETA(3,3)*/OMS(I2)
    1)*ALPHA(2,3)
    ALPHAS(3,1)=BETA(1,2)*BETA(3,1)-BETA(3,2)*BETA(1,1)*OMS(I3)/DENB
    ALPHAS(3,2)=BETA(1,2)*BETA(3,1)-BETA(3,2)*BETA(1,1)*OMS(I3)/DENB
    ALPHAS(3,3)=BETA(1,2)*BETA(3,1)-BETA(3,2)*BETA(1,1)*OMS(I3)/DENB
    CHKS(3)=ALPHA(1,3)*ALPHA(3,1)+(BETA(2,2)*OMS(3)*ALPHA(3,2)+BETA(3,3)*ALPHA(3,3)
    1)*ALPHA(3,3)
    WRITE(6,487)
    WRITE(6,489) (1,OMEGA(I),BETA(1,1),BETA(1,2),BETA(1,3),ALPHA(1,1),SUPPL256
    ALPHA(1,2),ALPHA(1,3),CHKS(1),I=1,3)
    1)*ALPHA(1,3)
    SORT(1)=0.
    SORT(2)=0.
    SORT(3)=0.
    DO 432 J=1,3
    GO TO (381,382,383), J
382    SORT(1)=0.
    SORT(2)=1.
    SORT(3)=0.
    GO TO 381
383    SORT(1)=0.
    SORT(2)=0.
    SORT(3)=1.

      11=2
      12=3
      GO TO 74
      11=1
      12=3
      GO TO 74
      11=1
      12=2
      GO TO 75
      MINI=1
      MIDI=2
      GO TO 76
      MINI=2
      MIDI=1
      GO TO 77
      SORT(1)=OMS(MINI)
      SORT(2)=OMS(MIDI)
      SORT(3)=OMS(MAXI)
      DO 77 I=1,3
      OMS(I)=SORT(I)
      DO 302 I=1,3
      OMEGA(I)=DSQRT(OMS(I))
      DO 381 I=1,3
      ALPHA(I,1)=1.
      DENB=RETA(2,1)*BETA(3,2)-BETA(3,1)*BETA(2,1)*OMS(I1)
      ALPHAS(1,2)=(BETA(1,2)*BETA(3,1)-BETA(3,2)*BETA(1,1)*OMS(I1))/DENB
      ALPHAS(1,3)=((BETA(2,2)*OMS(I1))*BETA(1,1)-OMS(I1)*BETA(1,2)*BETA(2,2))/DENB
      CHKS(1)=BETA(1,3)*ALPHA(1,1)+BETA(2,3)*ALPHA(1,2)+BETA(3,3)*OMS(I1)
      1)*ALPHA(1,3)
      11)*ALPHA(1,3)
      ALPHAS(2,1)=BETA(1,2)*BETA(3,1)-BETA(3,2)*BETA(1,1)*OMS(I1)/DENB
      ALPHAS(2,2)=BETA(1,2)*BETA(3,1)-BETA(3,2)*BETA(1,1)*OMS(I1)/DENB
      ALPHAS(2,3)=BETA(1,2)*BETA(3,1)-BETA(3,2)*BETA(1,1)*OMS(I1)/DENB
      CHKS(2)=ALPHA(1,3)*ALPHA(2,1)+BETA(2,3)*ALPHA(2,2)*BETA(3,3)/OMS(I2)
      1)*ALPHA(2,3)
      11)*ALPHA(2,3)
      ALPHAS(3,1)=BETA(1,2)*BETA(3,1)-BETA(3,2)*BETA(1,1)*OMS(I3)/DENB
      ALPHAS(3,2)=BETA(1,2)*BETA(3,1)-BETA(3,2)*BETA(1,1)*OMS(I3)/DENB
      ALPHAS(3,3)=BETA(1,2)*BETA(3,1)-BETA(3,2)*BETA(1,1)*OMS(I3)/DENB
      CHKS(3)=ALPHA(1,3)*ALPHA(3,1)+(BETA(2,2)*OMS(3)*ALPHA(3,2)+BETA(3,3)*ALPHA(3,3)
      1)*ALPHA(3,3)
      WRITE(6,487)
      WRITE(6,489) (1,OMEGA(I),BETA(1,1),BETA(1,2),BETA(1,3),ALPHA(1,1),SUPPL256
      ALPHA(1,2),ALPHA(1,3),CHKS(1),I=1,3)
      1)*ALPHA(1,3)
      SORT(1)=0.
      SORT(2)=0.
      SORT(3)=0.
      DO 432 J=1,3
      GO TO (381,382,383), J
382    SORT(1)=0.
      SORT(2)=1.
      SORT(3)=0.
      GO TO 381
383    SORT(1)=0.
      SORT(2)=0.
      SORT(3)=1.

361 DO 384 I=1,3
384 DELTA(I,KI)=ALPHA(I,KI)
CALL ALSOL(3,DELTA,SPRT,3)
DO 431 I=1,3
431 GAMMA(I,J)=SORT(I)
432 CONTINUE
WRITE(6,11) I,GAMMA(I,1),GAMMA(I,2),GAMMA(I,3),I=1,3
AMPLU = XMUVB * (1. - ROVB**3) / (1. - ROVB**4) * 1.3333333330 SUPPL284
SA = SMALS * SILA + RM * S2LB * HIGHS
SB = SMALS * S2LA**2 + RM * S2LR**2 * HIGHS
DELI(1,1) = XMUVB * (1. - ROVB**4) / (4. * (1. - SLAMZ * XB4**2)) SUPPL287
A = RROBR / F411
DELI(1,2) = 2. * SLAMZ * XBAB + DELI(1,1)
DELI(1,3) = A1 * (SLAMZ * XBAB + SB - SA) / (1. - SLAMZ * XB4**2) SUPPL290
A + B * HIGHS * S2LB
DELI(2,1) = A2 / A1 * DELI(1,1)
DELI(2,2) = A2 / A1 * DELI(1,2)
DELI(2,3) = A2 * (SLAMZ * XBAB + SB - SA) / (1. - SLAMZ * XB4**2) SUPPL294
A - R * SMALS * S2LB
DELI(3,1) = - SLAMZ * XBAB - DELI(1,1) / A1
DELI(3,2) = - 2. * SLAMZ * DELI(1,1) / A1
DELI(3,3) = (BDOR / RRRR)**2 + SLAMZ * (XBAB + SA - SP) / (1. - SLAMZ * XB4**2) SUPPL297
A = (1. - SLAMZ * XB4**2)
CMPA(2) = CMPAVB
CLR(2) = CLVB
NDIMC= 63
COSPSI= 1.
SINPSI= 0.
TO = ATOVB + ATCVB * COS PSI + ATSVB * SIN PSI
TOT(I) = TO - ATOVB
DO 50 I=1,3
50 SMALLG(I) = DELI(I,1) * CLVB + DELI(I,2) * CMPAVB
DO 51 I=1,3
51 GCAP(I,1)=0.
DO 52 J=1,3
52 Y(I,J)= Y(I,1)
IF (PLOP**LJ LT 0.)
1 WRITE(6,99001) TO, Z, TOPR, ZPR, Y, YPR, DEL, SMALLG
99001 FORMAT(/** TO=' ', IP3E13.6, ' Z= ', IP3E13.6, ' TOPR=' , IP3E13.6
1 , ' ZPR=' , IP3E13.6, '/ Y= ', IP3E13.6, ' YPR= ', IP3E13.6
2 * DEL= ' , IP3E13.6/* SMALLG= ' , IP3E13.6/)
RETURN
C
ENTRY SUPPI (UINF)
C
CMPPA(3) = CMPPA(2)
CMPPA(2) = CMPPA
CL(1) = 2. * CL(2) - CL(3)
CL(2) = CLVB
CL(1) = 2. * CL(2) - CL(3)
PSI = (BDRR / RDRR) * NTIME * DXI
SIN PSI = SIN(PSI)
COS PSI = COS(PSI)
TOTA(2) = TOT(1)
TOTA(1) = TO - ATVVB
TO PR = (BDRR / RDRR) * (ATVVB * COS PSI - ATCVB * SIN PSI)
DO 60 K = 1, 3
64 SMALL G(I) = UNIF * 2 * (DEL(I,1) * CL(K) + DEL(I,2) * CMPA(K))
A + DEL(I,3) * TOT(K)
DO 65 I = 1, 3
G CAP(I, K) = 0.
DO 65 J = 1, 3
65 G CAP(I, K) = GCAP(I, K) + ALPHA(I,J) * SMALLG(J)
60 CONTINUE
DO 62 I = 1, 3
Y(I,2) = Y(I,1)
YPR(I,2) = YPR(I,1)
WDXI = OMEGA(I) * DXI
SNUX = SIN(WDXI)
CDX = COS(WDXI)
Y(I,1) = Y(I,2) * CDX + YPR(I,2) + SWDXI / OMEGA(I)
A + (GCAP(I,2) - GCAP(I,1)) * (SWDXI - WDXI * CDX) / WDXI
B + GCAP(I,1) * (1. - SWDXI) / OMEGA(I)**2
62 YPR(I,1) = YPR(I,2) * CDX - OMEGA(I) * Y(I,2) * SWDXI
A + (GCAP(I,2) - GCAP(I,1)) * (WDXI * SWDXI + CDX - 1.)
B + WDXI * GCAP(I,1) * CDX / OMEGA(I)
DC 61 I = 1, 3
Z(I) = 0.
ZPR(I) = 0.
DO 61 J = 1, 3
Z(I) = Z(I) + GAMMA(I,J) * Y(J,1)
61 ZPR(I) = ZPR(I) + GAMMA(I,J) * YPR(J,1)
ALPH1 = TO + Z(3)
ALPH2 = TO + ZPR(3)
HEAVE = ZPR(1) - ZPR(2)
IF (PLOTOP .LT. 0.)
1 WRITE(6,9090) TO, Z, TOPR, ZPR, Y, YPR, DEL, SMALLG
2 , TOT
RETURN
1 FORMAT(5F10.4)
2 FORMAT(5F10.4)
3 FORMAT(1HL10X,"ITERATION FOR XBAR DIVERGED")
4 FORMAT(1HL5X,4HFI =E13.5,5X,4HF3 =E13.5//5X,4HRM =E13.5/5//4HRM
5 FORMAT(5X,5HM11 =E13.5,5X,5HM22 =E13.5,5X,5HM33 =E13.5,5X,5HM13 =E13.5
7 FORMAT(20X,6HX8/R =E13.5,10X,6HX8/B =E13.5/20X,6HL1/R =E13.5,10X,6SUPPL361
8 SUPPL355
9 SUPPL356
10 SUPPL357
11 SUPPL358
12 SUPPL359
13 SUPPL360
14 SUPPL361
SUBROUTINE SETUPS

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

REAL FTVB, FPVB, FPVRVB, DIDRVB, XMVB, DELVB, XMUVB, A, FOVB, XMUVB, ATOVB, ATCVB, ATSVB, ROVB, RVB, NVH
REAL ELSIG, DXI, RFB, ROBB, FRZ, ARR, AMPLU, FREQU, A, ALPH1, ALPH2, HEAVE, AROT, FREQF, PHIH, Y1, DRY, B, Y, TEST, UPRIM, XU, YJ, XL, YL, ER1, ER2, ER3, BDPR, H, CMPA, CMPAS, BARG, EMI, HVOR, SSPA, SVOR, TORF, XIVOR
I, PLOTOP, PSILOW, PSIUP

INTEGER TABLE(7, 80) /560 */

COMMON /BLI/ NTIME

COMMON /INPTVB/ FTVB(64), FPVB(64), FPVRVB(64), DIDRVB(54), XMVB(64), DELVB, XMUVB, FOVB, XMUVB, ATOVB, ATCVB, ATSVB, ROVB, RVB, NVH

COMMON /INPUTS/ NSBL, NZ, NOFF, NGAM, NSIG, NCOI, NCORD, LOER, MSTOP, MXT, MTR, RFR, ARR, AMPLU, FREQU, ALPH1, ALPH2, HEAVE, AROT, FREQF, PHIH, DRY, Y1, Y2, Y3, Y4, Y5, Y6, Y7, Y8, Y9, Y10, Y11, Y12, Y13, Y14, Y15, Y16, Y17, Y18, Y19, Y20, Y21, Y22, Y23, Y24, Y25, Y26, Y27, Y28, Y29, Y30, Y31, Y32, Y33, Y34, Y35, Y36, Y37, Y38, Y39, Y40, Y41, Y42, Y43, Y44, Y45, Y46, Y47, Y48, Y49, Y50, Y51, Y52, Y53, Y54, Y55, Y56, Y57, Y58, Y59, Y60, Y61, Y62, Y63, Y64, Y65, Y66, Y67, Y68, Y69, Y70, Y71, Y72, Y73, Y74, Y75, Y76, Y77, Y78, Y79, Y80

CALL WHERE(TABLE)
CALL ZERODIN

CALL SETUP('ALPH1', 'X', 'ALPH1')
CALL SETUP('ALPH2', 'X', 'ALPH2')
CALL SETUP('AMPLU', 'X', 'AMPLU')
CALL SETUP('ARR', 'X', 'ARR')
CALL SETUP('AROT', 'X', 'AROT')
CALL SETUP('ATCVB', 'X', 'ATCVB')
CALL SETUP('ATSVB', 'X', 'ATSVB')
CALL SETUP('BARG', 'X', 'BARG')
CALL SETUP('BDPR', 'X', 'BDPR')
CALL SETUP('CMPA', 'X', 'CMPA')
CALL SETUP('CMPAS '4, CMPAS ) SETUPS449
CALL SETUP('DELVB '4, DELVB ) SETUPS450
CALL SFTUP('IDRV9 '4, IDRV9, 64 ) SETUPS51
CALL SFTUP('DRY '4, DRY ) SETUPS52
CALL SETUP('OXI '4, OXI ) SETUPS53
CALL SETUP('ELSIG '4, ELSIG ) SETUPS54
CALL SETUP('EMI '4, EMI ) SETUPS55
CALL SETUP('ER1 '4, ER1 ) SETUPS56
CALL SETUP('ER2 '4, ER2 ) SETUPS57
CALL SETUP('ER3 '4, ER3 ) SETUPS58
CALL SETUP('FPVB '4, FPVB, 64 ) SETUPS59
CALL SETUP('FPRVRB '4, FPRVRB, 64 ) SETUPS60
CALL SETUP('FRZ '4, FRZ ) SETUPS61
CALL SETUP('FREQU '4, FREQU ) SETUPS62
CALL SETUP('FTVB '4, FTVB, 64 ) SETUPS63
CALL SETUP('FOVB '4, FOVB ) SETUPS64
CALL SETUP('HEAVE '4, HEAVE ) SETUPS65
CALL SETUP('HVOR '4, HVOR ) SETUPS66
CALL SETUP('INDV '4, INDV ) SETUPS67
CALL SETUP('LOWER '4, LOWER ) SETUPS68
CALL SETUP('MXT '4, MXT ) SETUPS69
CALL SETUP('MCTR '4, MCTR ) SETUPS70
CALL SETUP('MSTOP '4, MSTOP ) SETUPS71
CALL SETUP('MVR '4, MVR, 64 ) SETUPS72
CALL SETUP('NCOI '4, NCOI ) SETUPS73
CALL SETUP('NCORD '4, NCORD ) SETUPS74
CALL SETUP('AGAM '4, AGAM ) SETUPS75
CALL SETUP('NOFF '4, NOFF ) SETUPS76
CALL SETUP('NOTBL '4, NOTBL ) SETUPS77
CALL SETUP('NOUY '4, NOUY ) SETUPS78
CALL SETUP('NSBL '4, NSBL ) SETUPS79
CALL SETUP('NSIG '4, NSIG ) SETUPS80
CALL SETUP('NV8 '4, NV8 ) SETUPS81
CALL SETUP('NVOR '4, NVOR ) SETUPS82
CALL SETUP('NY '4, NY ) SETUPS83
CALL SETUP('NZ '4, NZ ) SETUPS84
CALL SETUP('PHIH '4, PHIH ) SETUPS85
CALL SETUP('PLOTOP '4, PLOTOP ) SETUPS86
CALL SETUP('PSILOW '4, PSILOW ) SETUPS87
CALL SETUP('PSIUP '4, PSIUP ) SETUPS88
CALL SETUP('RVB '4, RVB, 64 ) SETUPS89
CALL SETUP('RRAB '4, RRAB ) SETUPS90
CALL SETUP('REFB '4, REB ) SETUPS91
CALL SETUP('RRDRBR '4, RRDRBR ) SETUPS92
CALL SETUP('ROVB '4, ROVB ) SETUPS93
CALL SETUP('RYI '4, RYI ) SETUPS94
CALL SETUP('SSPA '4, SSPA ) SETUPS95
CALL SFTUP('SVOR '4, SVOR ) SETUPS96
CALL SETUP('TEST '4, TEST ) SETUPS97
CALL SETUP('TORF '4, TORF ) SETUPS98
CALL SETUP('UPRIM '4, UPRIM ) SETUPS99
CALL SETUP('XIVOR '4, XIVOR ) SETUPS100
CALL SETUP('XIL '4, XL, 30 ) SETUPS101
CALL SETUP('XMVB '4, XMVB, 64 ) SETUPS102
CALL SETUP(*XMUVR   *4,   XMUVR   )
CALL SETUP(*XMUAVB  *4,   XMJAVB  )
CALL SETUP(*XU      *4,   XU,  30   )
CALL SETUP(*Y      *4,   Y,    100   )
CALL SETUP(*YL     *4,   YL,   30   )
CALL SETUP(*YU     *4,   YU,   30   )

C
C
C
C
C
C
C

PSILOW= 1.E10
PSIUP= -1.E10
PLOTNP = 1.

RETURN

END

SETUPS92
SFUPS93
SETUPS94
SETUPS95
SETUPS96
SETUPS97
SETUPS98
SETUPS99
SETUP100
SETUP101
SETUP102
SETUP103
SETUP104
SETUP105
SETUP106
GO TO (543,550), LOWER
543 IF(LAMQ) 544,544,545
544 WRITE(MOUT,11) NTIME,NITS
GO TO 550
BLC 44
550 WRITE(MOUT,10) NTIME,NITS
545 CONTINUE
BLC 48
BLC 47
YTR = SQRT(RER)
BLC 50
UC(1,1) = 0.
BLC 53
V(1,1) = 0.
BLC 54
NV = NY - 2
BLC 55
NV1 = NV + 1
BLC 56
CALL YDIFF(NY,ALPHA,BETA,GAMMA,DELTA,SD,SE,SE,C2,C3,C4,Y)
BLC 57
DO 41 N=1,NV1
BLC 58
VISC(N,1) = 1.
BLC 59
41 VISC(N,2) = 1.
BLC 60
DO 42 M=MSTI,MST2
BLC 61
L = MSTI-M+2
BLC 62
DO 50 N=1,NV
BLC 63
GRAD(N+1) = SD(N+1)*UC(N+1,L)+SE(N+1)*UC(N+1,L)-SF(N+1)*UC(N,L)
BLC 64
GRAD(1) = C2*UC(2,L)+C3*UC(3,L)+C4*UC(4,L)
BLC 65
MM=M-1
BLC 66
CALL PGRAD(M,M,X,UE,DXI,PRESS,SA,SB,SC,SR,SS)
BLC 67
DO 456 N=1,NY
BLC 68
UC(N,1)=UC(N,L)
BLC 69
CALL SETIT(LAMQ,M,NV,REB,X,Y,UC,PRESS,GRAD,DELT,DISP,THETA,VISC,MTB3)
BLC 70
457 I=0.
BLC 71
DO 42 M=MSTI,MEND1
CALL CONT
BLC 72
BLC 73
MEND1 = MEND - 1
BLC 74
GRADS=GRAD(1)
BLC 75
GRADSS=GRAD(1)
BLC 76
C
THE MAIN CALCULATION STARTS HERE.
BLC 77
BLC 78
C
DO 99 M=MST1,MEND1
BLC 79
ITER=0.
BLC 80
WALLG=0.
BLC 81
MPI=M+1
BLC 82
DELT = DELT/YTR
BLC 83
DISP = DISP*YTR
BLC 84
THEY = THEY*YTR
BLC 85
SHEAR = GRAD(1)/YTR
BLC 86
IF(MOD(M,MOUT1).NE.MSTIMD) GO TO 225
BLC 87
GO TO (561,562), LOWER
BLC 88
561 WRITE(MOUT,12) X(M),X2(M),UE(M,1),PRESS,DELT,DISP,THETA,SHEAR
BLC 89
1                    MAXIT
GO TO 225
BLC 90
562 WRITE(MOUT,20) M,X(M),UE(M,1),PRESS,REB,UPRIM
BLC 91
WRITE(MOUT,24) DELT,DISP,THETA,DELTA,DISP,THEYT
BLC 92
WRITE(MOUT,21)
BLC 93
WRITE(MOUT,22) (YIN),UC(N,2),VIN,1),GRAD(N),VISC(N,1),N=1,NVP1
BLC 94
WRITE(MOUT,25) SHEAR
BLC 95
225 IF(GrADSS>GrADS-1.E-6) 229,229,408
BLC 96
4C8 XSV=X(M-2)+X(M-1-X(M-2))GRADSS/1GRADSS-GRADS)
BLC 97
IF(XSV-X(M)) 409,409,229
BLC 98
425 \text{WFS} = (X_{SX} - X(M-1)) / (X(M) - X(M-1)) \\
\text{GO TO 224} \\
229 \text{IF (GRAD(1)) 227, 227, 273} \\
273 \text{IF (NISP > GT. 0. AND. THETA > GT. 0.) GO TO 273} \\
283 \text{CONTINUE} \\
XSF = X(m-1) \\
USF = U(m-1, 1) \\
\text{XRL = X(m-1)} \\
\text{WRITE (MOUT, 21) XBL, XSEP} \\
\text{RETURN} \\
227 \text{WFS} = \text{GRAD(1)/GRADS-GRAD(1)} \\
224 \text{WFS} = 1.0 - \text{WFS} \\
\text{XSEP} = \text{WFS} * X(m-1) + WFS * X(m) \\
\text{XRL} = \text{WFS} * X(m-1) + \text{WFS} * X(m) \\
\text{USF} = \text{WFS} * U(m-1, 1) + \text{WFS} * U(m, 1) \\
\text{WFP} = (\text{XRL} - X(m-2)) / (X(m-1) - X(m-2)) \\
\text{WFP2} = 1.0 / \text{WFP} \\
\text{DISS} = \text{DISS} * \text{WFP2} + \text{DISS} * \text{WFP} \\
\text{THE} = \text{THE} * \text{WFP2} + \text{THE} * \text{WFP} \\
\text{WRITE (MOUT, 23) XBL, XSEP} \\
\text{IF (LAMQ, EQ. 0. AND. } \text{M. LT. MTRAN+5) LAMQ = 1} \\
\text{GO TO 222} \\
223 \text{CONTINUE} \\
\text{IF (NOTBAL, EQ. 2. AND. NITS > GT. 1. AND. M. GT. NZ AND.} \\
\text{X(m) > GT. XTEST) GO TO 283} \\
\text{IF (LAMQ) 801, 801, 802} \\
802 \text{CALL TRANS (UPRIM, PRESS, THETA, REB, UC, NY, FLAM, XFLAM, LAMQ)} \\
\text{IF (LAMQ) 805, 805, 801} \\
801 \text{CONTINUE} \\
\text{IF (Y(NV)-DELT) 620, 641, 641} \\
620 \text{RY = RY*DRY} \\
641 \text{C} \\
\text{C RE-SCALING CALCULATION STARTS HERE.} \\
\text{C} \\
\text{DO 632 N = 1, NY} \\
\text{YB(N) = YT(N)} \\
\text{VAR1(N) = UC(N, 2)} \\
632 \text{VAR2(N) = UC(N, 3)} \\
\text{CALL YSET (RY, YSUB2, NY, Y)} \\
\text{WRITE (MOUT, 351) YBL(NY), Y(NY)} \\
\text{DO 633 N = 2, NVP1} \\
\text{YN = YT(N)} \\
\text{CALL TERP (YN, YB1, VAR1, NY, UPAS1)} \\
\text{UC(N, 2) = UPAS1} \\
\text{CALL TERP (YN, YB1, VAR2, NY, UPAS2)} \\
633 \text{UC(N, 3) = UPAS2} \\
\text{CALL YDIFF (NY, ALPHA, BETA, GAMMA, DELTA, SO, SE, SF, C2, C3, C4, Y)} \\
\text{IF (LAMQ) 700, 700, 701} \\
700 \text{DO 635 N = 2, NVP1} \\
635 \text{VAR1(N) = VISC (N, 1)} \\
\text{VAR2(N) = VISC (N, 2)} \\
\text{DO 636 N = 2, NVP1} \\
73
\[ YIN = Y(N) \]

CALL TERP(YIN, YB1, VAR1, NV1, UPAS1)

\[ \text{VISC}(N+1) = \text{UPAS1} \]

CALL TERP(YIN, YB1, VAR2, NV1, UPAS2)

\[ \text{VISC}(N+2) = \text{UPAS2} \]

DO 637 N = 2, NV1

\[ \text{VAR1}(N) = V(N,1) \]

VAR2(N) = V(N,2)

DO 638 N = 2, NV1

\[ YIN = Y(N) \]

CALL TERP(YIN, YB1, VAR1, NV1, UPAS1)

\[ V(N+1) = \text{UPAS1} \]

CALL TERP(YIN, YB1, VAR2, NV1, UPAS2)

\[ V(N+2) = \text{UPAS2} \]

638 CONTINUE

C

RESCALING CALCULATION ENDS HERE.

C

CALL PGRAD(MX, UE, DXI, PRESS, SA, SB, SC, SR, SS)

C

RECURSION RELATIONS ARE SET UP HERE.

C

IF (ISTD.EQ. 1) GO TO 820

IF (SCALE(M+1,1)-1) 522, 522, 521

521 IF (SCALE(M+1,2)-1) 522, 522, 523

LACKU=1

FACU1=UE(M+1,2)/UE(M+1,1)

FACU2=UE(M+1,3)/UE(M+1,1)

GO TO 820

LACKU=2

DO 610 N = 2, NV1

VAR1(NN) = U(M+1,NN,1)

VAR2(NN) = U(M+1,NN,2)

CALL YSET(SCALE(M+1,1), YSUB2, NY, YB1)

CALL YSET(SCALE(M+1,2), YSUB2, NY, YB2)

820 DO 88 N = 2, NV

CALL CAPSITER(N, CAPG, CAPH, CAPJ, SR, SS, SD, SE, SF, VISC, V, UC)

A(N) = SF(N) * CAPG(V) - DELTA(N) * CAPH(N) + SF(N) * CAPJ(N)

R(N) = BCONSA * 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, LACKU

UPAS1=FACU1*UC(N,1)

UPAS2=FACU2*UC(N,1)

GO TO 576

575 YIN = Y(N)

CALL TERP(YIN, YB1, VAR1, NV, UPAS1)

CALL TERP(YIN, YB2, VAR2, NV, UPAS2)

576 F(N) = PRESS + FC(4, UPAS1-UPAS2) * CAP(4) * (SA*UC(N,2) - SC*UC(N,3))

88 CONTINUE

C

SOLUTION FOR VELOCITY PROFILE STARTS HERE.

C

DO 89 N = 2, NV
AP(N) = A(N)
RP(N) = B(N)
CP(N) = C(N)
DP(N) = D(N)

FP(N) = F(N)
DO 77 N=2,NVMP
    CP(N) = CP(N)/RP(N)
    DP(N) = DP(N)/RP(N)
    FP(N) = FP(N)/RP(N)
    RP(N+1) = BP(N+1) - CP(N)*AP(N+1)
    CP(N+1) = CP(N) - DP(N)*AP(N+1)
77    FP(N+1) = FP(N+1) - RP(N+1) - CP(N)*AP(N+1)
    UP(NY) = UE(N+1,1)
    UP(NVP1) = UP(NV)
    UP(NV) = (FP(NV)-UP(NY)*(DP(NV)+CP(NV)))/RP(NV)
DO 56 N=3,NVMP
    NN=NN+2-N
66    UP(NN) = FP(NN) - DP(NN)*UP(NN+2) - CP(NN)*UP(NN+1)
DO 65 N=2,N
65    UC(N,1) = UP(N)
IF (ITER) 843,841,843
841    DO 842 N=2,NVP1
        VIN(2) = VIN(1)
842    VISC(N,2) = VISC(N,1)
        DISS = DISS
        DISS = DISS
        THESS = THESS
        GRADS = GRAD(1)
        GRADS = GRAD(1)
459    DO 55 N=2,NVP1
        VIN(N,1) = VIN(N-1,1) - \[\frac{A(N-1) - A(N)}{B(N-1) - B(N)}\] * (SA*UC(N,1) + UC(N-1,1) - 3*(UC(N,1) - UC(N-1,1)))
55    INN+1 = IN(N+1,1) - \[\frac{A(N-1) - A(N)}{B(N-1) - B(N)}\] * (SA*UC(N,1) + UC(N-1,1) - 3*(UC(N,1) - UC(N-1,1)))
466    DO 56 N=1,NVMP
467    GRAD(N+1) = SD(N+1)*UC(N+1,1) + SE(N+1) - SF(N+1)*UC(N,1)
468    GRAD(1) = C2*UC(2,1) + C3*UC(3,1) + C4*UC(4,1)
469    CALL SFITIT(LAMQ,MP1,MP2,REB,X,Y,UC,PRESS,GRAD,DEL0,M0,DISP,THET,VISC,NCMP,ITER)
470    IMTRAN
471    ITER=ITER+1
472    GO TO (330,809),LOWER
809    WRITE(MOUT,810) ITER,GRAD(1)
830    IF (ITER-9) 811,811,812
811    FPW=ABSTGRAD(1)-WALLG)
812    IF (WALLG=1.) 120,120,119
119    EPW=EPW+WALLG
120    IF (EPW-TEST) 812,814,814
814    WALLG = GRAD(1)
75    GO TO 820
812    DO 44 N=1,NY
        UC(N,3) = UC(N,2)
        UC(N,2) = UC(N,1)
44    CONTINUE
44    MAXIT=ITER
    IF (TSTAD .EQ. 1) GO TO 99
    DTY 48 N=1,NY
48  USAW(M+1,N) = UC(N,1)
    SCALS(M+1) = RY
 99  CONTINUE
    XSEP=1.1
    USEP=UE(MX,1)
222  CONTINUE
    RETURN
    END
SUBROUTINE PLOTS(P, PLOT, P, L)
REAL * 9 ORD(6)
DIMENSION P(200,7), TIT(56)

1, NFP(6)
DATA NL, N2, NO, N42
1/1, 2, 0, 42 /
DATA ORD, THETA-P, TORS, FLAP-H, REN-H, CM-A /
1, CL, CM-A /
IF(PLOT=EQ. 0.) RETURN
IF(L=LT. 2) RETURN
IF(PLOT=EQ. 2.) GO TO 2
PLOT=2,
CONTINUE

2 NL=1,
DO 1 J = 1, 6
CALL EPLEXT(9, Ni, N1, P, P(1,J+1), L, -N1, N2
1, N42, 1, 12, PSI-DEGREES, 8, ORD(J)
2, Ni, Ni, X1, X2, Ni, Yi, Yi, Ni, NO, NL
1 CONTINUE
NFP(1)= -1
NFP(2)= 66
NFP(3)= 50
NFP(4)= 50
NFP(5)= 680
CALL ELEXPT(9, Ni, N1, P, P(1,2), L, -N1, N2
1, N42, 1, 12, PSI-DEGREES, 8, ORD(1)
2, NFP, Ni, X1, X2, Ni, Yi, Yi, Ni, Yo, Vi
NFP(1)= -2
NFP(2)= 66
NFP(3)= 350
NFP(4)= 380
CALL ELEXPT(9, Ni, N1, P, P(1,6), L, -N1, N2
1, N42, 1, 12, 8, ORD(5)
2, NFP, Ni, X1, X2, Ni, Yi, Yi, Ni, NO, Vi
NFP(1)= 50
NFP(2)= 690
NFP(3)= 40
CALL ELEXPT(9, Ni, N1, P, P(1,7), L, -N1, N2
1, N42, 1, 12, 8, ORD(6)
2, NFP, Ni, X1, X2, Ni, Yi, Yi, Ni, NO, Vi
NFP(1)= -1
NFP(2)= 50
NFP(3)= 50
NFP(4)= 50
NFP(5)= 690
CALL ELEXPT(9, Ni, N1, P, P(1,3), L, -N1, N2
1, N42, 1, 12, PSI-DEGREES, 8, ORD(2)
2, NFP, Ni, X1, X2, Ni, Yi, Yi, Ni, NO, Vi
NFP(1)= 2
NFP(2)= 66
NFP(3)= 350
NFP(4)= 380
CALL ELEXPT(9, Ni, N1, P, P(1,4), L, -N1, N2
1, N42, 1, 12, 8, ORD(3)

77
2, NFP, N1, XL, XU, N1, YL, YU, N1, N0, N1
NFP(2) = 50
NFP(4) = 690
NFP(5) = 43
CALL EZPLT(9, N1, N1, P, P(1,5), L, -N1, N2
1, N42, 1, 12, 8, 4)
2, NFP, N1, XL, XU, N1, YL, YU, N1, N0, N1
RETURN
END
SUBROUTINE STAG(X, NY, MSTOP, MST, DXI, RY, DRY, X, Y, UE, UC, V, USAV, SCALS, STAG)
115P)
5  C PROGRAM FOR CALCULATING THE BOUNDARY LAYER PROFILE NEAR
6  C THE STAGNATION POINT
7  C
8
9 COMMON /HL1/ NTIME, ADIME, LSTD
10 DIMENSION USAV(300, 100), SCALS(300)
11 DIMENSION PHI7(24), PHIP(24), FTAP(24)
12 DIMENSION X(300), Y(100), UE(300, 3), UC(100, 3), V(100, 2)
13 DIMENSION FF(100), EFPP(100)
14 DATA FTAP /9.4, 4.1, 2.6, 1.8, 1.2, 1.4, 1.6, 2.6, 2.2, 2.4, 2.6, 2.8, 3.0, 3.2, 3.4, 3.6, 3.8, 4.0, 4.2, 4.4, 4.6/.
15 DATA PHII /9.1, 0.033, 0.981, 1.867, 3.124, 4.592, 6.22, 7.967, 9.973, 1.16, STAG/ 16 19, 1.362, 1.5578, 1.7553, 1.9538, 2.153, 2.3526, 2.5523, 2.7522, 2.9521, 3.1 STAG/ 17 1521, 3.3521, 3.5521, 3.7521, 3.9521/.
18 DATA PHIP /0.01, 0.266, 0.4145, 0.5663, 0.6859, 0.7779, 0.8467, 0.8968, 0.9123, 0.956 STAG/ 19 18, .9732, .9839, .9935, .9946, .997, .9984, .9992, .9996, .9998, .9999, 1.1 STAG/ 20 1.1, 1.1/.
21 RAG = .08
22 IF(115) 10, 10, 5
23 BAG = .5
24 10 EF(I) = 0.
25 DO 20 M = 1, MX
26 IF(UF(M, 1)) 20, 20, 19
27 15 MSP = M
28 GO TO 21
29 CONTINUE
30 21 ASTAG = (UE(MSP + 2, 1) - UE(MSP + 1, 1)) / (X(MSP + 2) - X(MSP + 1))
31 IF(ASTAG) 22, 22, 23
32 22 ASTAG = (UE(MSP) - UE(MSP - 1, 1)) / (X(MSP) - X(MSP - 1))
33 SQAS = SQRT(ASTAG)
34 DELT = 2.675 SQAS
35 3C9 IF(DELT - YNY - 3) 311, 310, 310
36 310 RY = RY + DRY
37 CALL YSFT(RY, Y(2), NY, Y)
38 GO TO 309
39 311 CONTINUE
40 DO 33 NN = 2, NY
41 YET = Y(N)*SQAS
42 GO TO 33
43 33 CONTINUE
44 4C3 308, 408, 33
45 EF(N) = YET - 6479
46 EF(N) = 1.
47 GO TO 80
48 410 FRAC1 = (YET - ETAP(MARK - 1)) / ETAP(MARK) - ETAP(MARK - 1)
49 FRAC2 = 1. - FRAC1
50 EF(N) = PHITZ(MARK - 1)*FRAC1 + PHIZ(MARK)*FRAC2
51 EF(N) = PHIP(MARK - 1)*FRAC1 + PHIP(MARK)*FRAC2
52 CONTINUE
53 M1 = MSP - MSTOP
54 M2 = MSP + MSTOP
M=M1–1
50  M=M+1
MST=M+1
SCALS(M)=RY
DO 71 N=1,NV
UC(N,3) = UC(N,2)
UC(N,2) = UE(M,1)*EFP(N)
V(N,2) = V(N,1)
V(N,1) = –SQA5*EF(N)
IF(ISTD .EQ. 1) GO TO 71
USAV(M,N)=UC(N,2)
71  CONTINUE
IF(M-M2) 50,55,55
55  IF(UF(M,1)=BAG) 50,50,81
81  CONTINUE
RETURN
END
SURROUTINE ATTPR(PREC, XSIG, NSIG, ASZ, AS, AR, CMAT, RMAT, NGAM, NF, ACAP, TAMPR) 1
THICK, ND9R, GAMAW, UINF, UDOT, DXI, BCAP) 2
DIMENSION XSIG(100), ASZ(30), ASI(30, 30), AR(30), ACAP(100, 3) 3
DIMENSION ACAP(30, 3), THICK(24), GAMAW(1000) ATTPR 4
DECIMAL PRECISION CMAT(60, 60), RMAT(130) ATTPR 5
PI=3.14159 6
NGAM=NGAM+1 ATTPR 7
DO 50 M=1, NGAM ATTPR 8
CMAT(M,1)=ASZ(M) ATTPR 9
RMAT(M)=AR(M) ATTPR 10
DC 25 N=1, NGAM ATTPR 11
25 CMAT(M,N+1)=AS(M,N) ATTPR 12
50 CONTINUE ATTPR 13
CALL ALSOL(NGAM, CMAT, RMAT) ATTPR 14
DO 75 M=1, NGAM ATTPR 15
75 ACAP(M,1)=RMAT(M) ATTPR 16
GAMAW(L)=GAMAW(L) ATTPR 17
SAVE=XSIG(NSIG+1) ATTPR 18
XSIG(NSIG+1)=2. ATTPR 19
CALL CPC(0, NGAM, NF, XSIG, NSIG, XSIG, NSIG, XSIG, NSIG, ACAP, BCAP, THICK, TAMPR) 20
1000 GAMAW, UINF, UDOT, 1, SAVE, DXI, PREC) ATTPR 21
XSIG(NSIG+1)=SAVE ATTPR 22
RETURN ATTPR 23
END ATTPR 24
SUBROUTINE UNPOP(NGAM, AR, ALAM, AFACT, RMAT, CMAT, XGAM, AS, ACAP, 4X, NZ, NUNPOP)
!
DIMENSION AR(30), ALAM(30), XGAM(30), AS(30, 30), ACAP(30, 3), XSIG(120), NUNPOP
!
DCURLE PRECISION RYAT (130), CMAT(60, 60)
!
NUNPOP
!
DO 5 M = 1, NGAM
!
CMAT(M, 1) = 1.
!
CMAT(M, 2) = XGAM(M)
!
DO 5 N = 2, NGAM
!
CMAT(M, N + 1) = AS(M, N)
!
CALL ALSCL(NGAM, CMAT, RMAT)
!
DO 10 N = 1, NGAM
!
ACAP(N, 1) = RMAT(N)
!
DO 15 M = 1, MX
!
SIGN = 1.
!
IF (M - N2) 12, 14, 14
!
12 SIGN = -SIGN
!
14 CALL QECAL(0, NGAM, NGAM, NF, XSIG, ACAP, BCAP, THICK, RDBB, 0., UINF, XC(M)), NUNPOP
!
15 CONTINUE
!
RETURN
!
END

82
SURROUTINE ALSOL(NT, C, R)
DOUBLE PRECISION C    NDIMC, NDIMC), R(130)
DOUBLE PRECISION CMAX,SAVE, SUM
COMMON /R11/ NTIME, NDIMC
NT1 = NT-1
DC 99 J=1,NT1
CMAX = C(NT,J)
L=NT
DC 10 I=J,NT1
IF (DARS(J,CMAX)-DARS(C(I,J))) 5,10,10
   CMAX = C(I,J)
   L=1
10    CONTINUE
   DC 15 JJ=J,NT
   SAVE = C(L, JJ)
   C(L, JJ) = C(J, JJ)
   15    C(J, JJ) = SAVE/CMAX
   SAVE = R(L)
   R(L) = R(J)
   R(J) = SAVE/CMAX
   JPI = J+1
   DO 25 I=JPI,NT
   DO 20 JJ=JPI,NT
   20    CJ = C(I, JJ) - C(I, J)*C(J, JJ)
   25    R(I) = R(I) - R(J)*C(I, J)
   CONTINUE
   R(NT) = R(NT)/C(NT, NT)
   DO 150 K=I,NT1
   I=NT-K
   JPI = I+1
   SUM = 0.
   DO 125 J=JPI,NT
   125    SUM = SUM + R(JT)*C(I, J)
   150    R(I) = R(I) - SUM
   RETURN
END
```
SURF PUTNF CPC(ISEP,NGAM,NF,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,CPC
1,BCAP,THICK,ROBB,GAMAW,UNIF,UDDOT,SIGN,XC,DXI,CP)
DIMENSION XSIG(100),XSIGA(100),XSIGB(100),ACAP(30,3),BCAP(100,3)
DIMENSION GAMAW(1000),THICK(24)
THETA=ARCT(XC)
RECIPI(UINF*UINF)
SUM=0.
ANGLE=0.
DC=5 N=1,NF
ANGLE=ANGLE+THETA
SUM=SUM+THICK(N)*COS(ANGLE)
CP=UDDOT*RECIPI(THICK(1)+2.*(1.-XC)*SUM)
CALL QECAL(ISEP,NGAM,NSIG,NF,XSIG,ACAP,BCAP,THICK,RCRR,GAMAW(1),UI CPC
1INF,XC,U,SIGN)
CP=CP+2.*(SIGN*U/UINF-1.)
CALL EGAMI(1,NGAM,ACAP,BCAP(1,1),XSIG(1),XSIG(XSIG+1),GAMAW(1),XC,CP
1
IVAL1)
CALL EGAMI(2,NGAM,ACAP,BCAP(1,2),XSIGA(1),XSIGA(NGSIG+1),GAMAW(2),CP
1XC,VAL2)
CALL EGAMI(3,NGAM,ACAP,BCAP(1,3),XSIGB(1),XSIGB(NGSIG+1),GAMAW(3),CP
1XC,VAL3)
CP=CP+SIGN*RECIPI(1.5*VAL1-2.*VAL2+.5*VAL3)/DXI
IF(ISEP) 20,20,10
CALL FSGI1,NSIG,XSIG,BCAP,XC,VAL1)
CALLSIGI(2,NSIGA,XSIGA,BCAP,XC,VAL2)
CALL SIG3(3,NSIGB,XSIGB,BCAP,XC,VAL3)
CP=CP+RECIPI(1.5*VAL1-2.*VAL2+.5*VAL3)/DXI
20 CP=CP
RETURN
END
```
SUBROUTINE CLCM(NC01, ISEP, NGAM, XSIG, NSIG, XSIGA, NSIGA, XSIGB, NSIGB, ACAP, BCAP, TCM)

COMMON /CLCMAL/ CLV, CMV, CMPAV

DIMENSION ARGL(21), ARGH(21)

DIMENSION GAMAW(1000), THICK(24)

DIMENSION XSIG(100), XSIGA(100), XSIGB(100), ACAP(30, 3), HCAP(100, 3)

FORMAT(/40X,4H4CL =E13.5/40X, 4HCM =E13.5, 17H (ABOUT MIDCHORD))/40X, CLM

4

ICAP, BCAP, THICK, ROBB, GAMAW, UINF, UDUT, DXI, AROT, CM18A)

CM1

CM2

CM3

CM4

CM5

CM6

CM7

CM8

CM9

CM10

CM11

CM12

CM13

CM14

CM15

CM16

CM17

CM18

CM19

CM20

CM21

CM22

CM23

CM24

CM25

CM26

CM27

CM28

CM29

CM30

CM31

CM32

CM33

CM34

CM35

CM36

CM37

CM38

CM39

CM40

CM41

CM42

CM43

CM44

CM45

CM46

CM47

CM48

CM49

CM50

CM51

CM52

CM53

CM54
CM = CM - (XIPI - XI)* (FMIPI + FMI)  
XI = XIPI  
FMI = FMIPI  
15  
DO 16 I = 1, 21  
CALL CPC(I) SEP, NGAM, 1, XSIG, NSIG, XSIGA, NSIGA, XSIGB, NSIGB, ACAP, BCAP, TCLCM  
1HICK, R DBR, GAMAW, UNIF, UDOT, 1.0, XIPI, DXI, CPU)  
CALL CPC(I) SEP, NGAM, 1, XSIG, NSIG, XSIGA, NSIGA, XSIGB, NSIGB, ACAP, RCAP, TCLCM  
1HICK, R DBR, GAMAW, UNIF, UDOT, -1.0, XIPI, DXI, CPU)  
SUML = 0.  
SUMM = 0.  
DO 17 I = 1, 19, 2  
SUML = SUML + 2. * ARGL(I) + 4. * ARGL(I + 1)  
SUMM = SUMM + 2. * ARGX(I) + 4. * ARGX(I + 1)  
17  
CL = CL + 0.833 33 3 3 * (SUML + ARGL(I) - ARGL(I + 1))  
CM = CM + 0.833 33 3 3 * (SUMM + ARGX(I) - ARGX(I + 1))  
BCON = 16. * RCAP(1, 1) + SQRT(5. E-4 * (XATT - XSIG(I))) / UINF  
CL = CL + BCON  
CM = CM + XATT * BCON  
GO TO 130  
5  
DO 99 I = 1, NCDO  
ANGLE = ANGLE + DT  
XIPI = COS(ANGLE)  
CALL CPC(I) SEP, NGAM, 1, XSIG, NSIG, XSIGA, NSIGA, XSIGB, NSIGB, ACAP, BCAP, TCLCM  
1HICK, R DBR, GAMAW, UNIF, UDOT, 1.0, XIPI, DXI, CPU)  
CALL CPC(I) SEP, NGAM, 1, XSIG, NSIG, XSIGA, NSIGA, XSIGB, NSIGB, ACAP, RCAP, TCLCM  
1HICK, R DBR, GAMAW, UNIF, UDOT, -1.0, XIPI, DXI, CPU)  
FMIPI = XIPI * FLIPI  
CL = CL + (XIPI - XI) * (FMIPI + FMI)  
XI = XIPI  
FMI = FMIPI  
99  
DO 100 I = 25, CL  
CL = CL + 1.25 * CM  
CM = -1.25 * CM  
CMPA = CM + AROT * CL + 5  
WRITE(MOUT, 4) CL, CM, CMPA, AROT  
THICK(I) = SAVE  
CLEVR = CL  
CHVB = CM  
CMPA = CM  
RETURN  
END
SUBROUTINE QECL (ISEP, NGAM, NSIG, NF, XSIG, ACAP, RCAP, THICK, RDHA, GAMMA, QECL)
1, UNINF, XC, U, SIGN)
DIMENSION ACAP (30, 31), RCAP (100, 3), XSIG (100)
DIMENSION THICK (24)
EPS 1.0, 6
CORR = 7.24767 + 1L, -.6362 * SQRT (RDHA) + .25 * RDHA
SINT = SQRT (1 - XC * XC)
THETA = ARCT (XC)
COUNT = 0,
SUM = 0.
SINT2 = SIN (.5 * THETA)
COST2 = COS (.5 * THETA)
IF (SINT - EPS) 4, 5, 6
4 FACT = THETA * .5
GO TO 8
6 FACT = (1 - XC) / SINT
8 DO 10 N = 1, NF
COUNT = COUNT + 1.
ANGLE = THETA * COUNT
SUM = SUM + THICK * N * (COUNT * FACT * SIN (ANGLE) - COS (ANGLE))
10 CONTINUE
U = 2 * SIGN * UNINF * COST2 * SUM + ACAP (1, 1) * SINT2 + .25 * COST2 * (1 - XC) * (3 - XC)
11 * GAMMA
SUM = 0.
ANGLE = 0.
DO 12 N = 1, NGAM
ANGLE = ANGLE + THETA
12 SUM = SUM + ACAP (N - 1, 1) * SIN (ANGLE)
U = U * COST2 * SUM
IF (ISEP) 25, 99, 25
25 SUM = 0.
XSEP = XSIG (1)
XATT = XSIG (NSIG + 1)
DO 40 N = 2, NSIG
40 SUM = SUM + RCAP (N, 1) * F8 (SIG (N - 1), XSIG (N), XSIG (N + 1), XC)
IF (XATT - EPS) 45, 44, 46
46 FACT = (1 - XATT)**(-1.5) * SQRT (TXATT - XSEP) * (1 - XC) / (XC - XATT) * (1 + 3 * FACT)
LXATT = 6 * XC - SIGN * (1 - SQRT (XSEP - XC) / (XATT - XC))
GO TO 55
45 IF (XSEP - XC) 49, 49, 48
48 FACT = -SIGN * (1 - SQRT (XSEP - XC) / (XATT - XC))
GO TO 55
49 FACT = -SIGN
55 U = U * COST2 * (BCAP (1, 1) * FACT * SIGN * SJM)
99 U = U * SIGN * UNINF * SQRT (1 - XC) + CORR * U / SQRT (1 + XC + 5 * RDHA)
RETURN
END
SUBROUTINE YVB(Y, I)
REAL Y(I)
REAL MVB
COMMON /IVPTVB/ FTVB(64), FPVR(64), FPPRVB(64), DIDRVB(64), YVB
A XMVB(64), DELVB, XMVB, FOXR, XMVRB, YVB
B ATVII, ATCVB, ATSVR, ROVB, RVB(64), YVB
C MVB(64), YVB
Y(1) = (RVB(I) - DELVB)**2 * MAVR(I)
Y(2) = FPVB(I)**2 * MAVR(I)
Y(3) = FTVB(I)**2 * DDRVII
Y(4) = (DELVB - MV(R(I))) * FTVB(I) * XMVB(I) * MAVR(I)
Y(5) = FPVB(I) * FTVB(I) * XMVRB(I) * MAVR(I)
Y(6) = RVB(I) * (DELVB - RVB(I))/MVRB(I)
Y(7) = (RVB(I) - DELVB) * FPVRB(I) * FTVB(I) * XMVRB(I) * 4VBI
IPI = I+1
IF(IPI .GE. NVB) GO TO 12
SUM = 0.
DD 10 J = IPI, NVB
10 SUM = SUM - (RVB(4+1) - RVB(4)) * (RVB(4+1) - MAVB(J+1))
A + RVB(J) * MAVB(J)
12 Y(7) = FPVRB(I)**2 * SUM / 2.
RETURN
END
SUBROUTINE POLLY(y, A, B, C, N, M)
C
IMPLICIT REAL*8 (A-H, O-Z)
C
COMPLEX X ROOTS OF A POLYNOMIAL BAIRSTOW'S METHOD
DIMENSION A(30), AN(60), C(26), ABR(26), B(30), AA(30)
I=1
7 NPI=N+1
NPP1=N+2
DO 66 I=1,NPI
LLL=NPP1-I
601 A(I)=A(I)(LLL)
13 DO 14 K=1,NPI
14 ABR(K)=A(K)
ABSSQ=ABS(*ABS
REL SQ=REL*REL
NBAR=N
B(I)=A(I)
C(I)=A(I)
15 IF(NBAR-2).GE.00,210,17
17 P1=2
Q1=1
18 IFR=0
19 P1=P1*5.
Q1=Q1/16.
33 P=P1
Q=Q1
NRP1=NBAR+1
34 L=1
[last=nb
DIST=9.99936
C
BASTOW ITERATION
37 B(2)=ABR(2T)-P B1)
DO 40 K=3,NRP1
40 B(K)=ABR(K)-P B(K-1)-Q B (K-2)
45 C(2)=B(2)-P C(I)
DO 50 K=3,LAST
50 C(I)=C(I)-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)
DQ=SQR(D)
60 IF(DQ>-1.0D36).GE.19,19,60
65 DELP=REAL(LAST)*C(LAST-1)*C(LAST-2)*D
DELQ=(B(LAST-1)+C(LAST-1)-B(LAST)*C(LAST))/D
C
TEST FOR CONVERGENCE
RELP=DEL/P
Q=Q+DELQ
RELPS=RELPS+RELP
RELQS=RELQS+RELQ
65 IF(RELPS/DELQ>70).GE.70,70,65
70 IF(RELQS/DELQ>70).GE.70,70,65
75 IF(Delp*DELQ-ABSSQ)70,70,80
80 GO TO 90;100;L
50 ITER = ITER + 1
   IF (250 - ITER) .GE. 310, 37, 37
100 IF (DTEST - DELSQ) .GE. 34, 34, 110
110 DTEST = DELSQ
   R (2) = A (2) - P * B (1)
   DO 115 K = 3, NRP1
   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
   NRP1 = NBAR + 1
   ABAR (2) = ABAR (2) - P * ABAR (1)
   DO 150 K = 3, NRP1
   150 ABAR (K) = ABAR (K) - P * ABAR (K - 1) - Q * ABAR (K - 2)
   GO TO 250
C Solve linear equation
200 NBAR = NBAR - 1
   RL = -ABAR (2) / ABAR (1)
   R2 = 0.
   GO TO 262
C Normalize quadratic
210 P = ABAR (2) / ABAR (1)
   Q = ABAR (3) / ABAR (1)
   NBAR = NBAR - 2
C Solve normalized quadratic
250 RL = -P / 2.
   C1 = RL * RL - 1
   IF (C1) .LE. 270, 280, 260
   260 C1 = DSQRT (C1)
   R2 = RL - C1
   R1 = RL + C1
   262 C1 = 0.
   GO TO 290
270 C1 = -C1
   C1 = DSQRT (C1)
280 R2 = RL
290 C2 = -C1
   AN (III + 1) = C1
   AN (III + 4) = R1
   AN (III + 2) = C2
   AN (III + 3) = R2
   III = III + 4
   IF (NBAR - 1) .LE. 4, 200, 15
C Special Conditions
310 WRITE (6, 600)
   600 FORMAT (1X, 50X, 'CONVERGENCE IN 250 ITERATIONS, POLLY HAS SPEAKEN')
4 CONTINUE
90 RETURN
END
SUBROUTINE SETIT(LGO,M,NV,REB,X,Y,UC,PRESS,GRAD,DELT,DISP,THETA,VISETUP)
C
C SUBROUTINE FOR CALCULATION OF BOUNDARY LAYER THICKNESS,
C DISPLACEMENT THICKNESS, MOMENTUM THICKNESS AND FIDDY VISCOSITY.
C
DIMENSION X(300),Y(100),UC(100,3),VISC(100,2),GRAD(100)
RTR=SQR(T(Reb))
NY = NV + 2
UEDGE = .995*UC(NY,1)
DO 10 N=1,NV
10 CONTINUE
UEDGE = UC(N+1,1)
GO TO 20

20 DELT = Y(NDELT)*(UEDGE-UC(NDELT,1))*(Y(NDELT+1)-Y(NDELT))/UC(NDELT)

DO 60 N=1,NV
60 SUM = SUM+Y(N)-Y(N-1)*UC(N,1)+UC(N-1,1)
DISP = (Y(NV)-.5*SUM/UC(NV,1))/RTR
SUM = 0.
UEDGE = UC(NV,1)
DO 50 N=1,NV
50 SUM = SUM+Y(N)-Y(N-1)*UC(N-1,1)+UC(N,1)*UEDE-UC(N-1,1)
THETA = .5*SUM/(RTR*UEDE**2)
IF(LGO) 53,53,56
53 NVPI=NV+1
EASE = 1.
IF(M-MRAN) 31,32,32
31 CONTINUE
INNER=0
FAC1 = .16*RTR/EASE
FAC2 = .016*UEDE*DISP*REB/EASE
FAC1 = -RTR/26.
EFAC1 = PRESS*RTR
TAUW = GRAD(1)/RTR
DO 160 N=2,NVPI
160 ALTER = 1.*FAC2/(1.+5.5*(Y(N)/DELT)**6)
IF(INNER) 402,401,402
402 VISC(N,1)=ALFTR
GO TO 160
401 CONTINUE

TAUW=TAUW-Y(N)*EFAC2
IF(TAUXY) 701,701,702
7C1 VISC(N,1)=1.
GO TO 703
702 PX=Y(N)*EFAC1*SQR(TAUW)
VISC(N,1) = 1.+FAC1*Y(N)*Y(N)*ABS(GRAD(N))*(1.-EXP(EX)**2
7C3 IF(VISC(N,1)-ALTER) 150,160,521
521 VISC(N,1)=ALTER
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 SAVE=RAVE
56 CONTINUE
RETURN
END
C IF INDT IS NONZERO, THE BOUNDARY LAYER IS TURBULENT
C AT SEPARATION.

CALL H4X4(INOT, XSEP, DFL1, THET1, XATT, REB, USEP, X3, H3, X4, H4)
IF (XSF-1, 1) 24, 25, 25
25 CP4=0.
GO TO 27
24 URAT=EXP(-.08712-U11(H4)-.24723*(.3255+U12(H4))
CP4=-1.-(1.-PREC)/URAT**2
DEADL=XATT-XSEP
IF (DEADL<2.1) 3, 6, 6
5 G=1.5*DEADL**2
GO TO 7
6 G=1.
7 CP4=PREC+(CP4-PREC)*(1.-G*XSEP)
27 CONTINUE
COEF=(PREC-CP4)/(XATT-X4)
CZ=2.*UDOT/UNF
DC=2.*UINF
DO 20 H=INDF-
20 IF(X-1.1 2, 2, 3
2 THETA = ARCTAN
TANT = SIN(5.*THETA)/COS(5.*THETA)
C1 = -CZ*(1.-COS(THETA))
DO 10 N=1, NF
10 COUNT=COUNT+1.
C0UNGLE=COUNT*THETA
10 SUM=SUM+THICK(NF)*(C1*COS(ANGLE)+CZ*(COUNT*TANT*SIND(ANGLE)-C1*SIND(ANGLE))
11)
SUM=SUM-5.*CZ*THICK(1)
GO TO 35
3 C1=CZ*(1.-X)
XRAD=1./(X+SQRT(X*X-1.))
C1=CZ*(X-1.)
RF=SQRT((X-1.)/(X+1.))
SUM=THICK(NF)*XRAD*(CZ*(RF-1.))-CZ*(1.-.5*XRAD)
FRAD=XRAD
COUNT=1.
DO 30 N=2, NF
COUNT=COUNT+1.
FRAD=FRAD*XRAD
30 SUM=SUM+THICK(NF)*FRAD*(CZ*(COUNT*RF-1.)+C1

35 CP = CP4
50 IF (X - X4) 55, 50, 50
55 CP = CP + (X - X4) * COEF
55 CONTINUE
FPRES(M) = -UINF * 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,THET1,RE9,XC1,UL,XC5,DOP,DEL5,X,XC4,UX,XZ,X5,UBUR3)
15,UF,ALTC,RENFL,USTOP

DIMENSION XL(300),UX(300),UF(300,3)

FCAP(X)=-19.556*X*107.535*X*X*336.33*X*X*3*508.1*X*X*4-205.06*X*X
UI1(X)=-.4652*X*.48425*X*X-.45293*X*X+.6592*X*X
UI2(X)=-.04592*X*1.91615*X*X*.98143*X*X-.54212*X*X
FDLT(X)=XP2(2.5773-.34252*X-.4379*X*X-.076511*X*X+.303707*X*X)
FAICH(X)=XP3(-3.7491+.038772*X+.41967*X*X+.071046*X*X+3.3032162*X*3)

DELI(X)=-.04592*X*ALOG(X)-.39242*X*.54535*X*X-1.39147*X*X-1)*3425*UBUR3

IUH,10X,14X,18X,40H substitute

END

25 FORMAT(1MH1,44X,3HANALYSIS OF LEADING-EDGE RUBBLE///34X,1IH<19X,1HUBUR3

MOUT=6
HI=25
H5=.429
IF(XC1-XC(M))4,4,5
ML=M
GO TO 6
CONTINUE

6 XL=X(M1-1)+XC(M1)-XC(M1-1)+XC(M1-1)--XC(M1-1)

ARG=ALOS(X4-X1//TREP/DELT1*U11)

H4=.25*FAICH(ARG)

DELT4=.58*FDLT(ARG)*DELT1

IF(U1-USTOP)41,41,40

IF(X5-XL,1LT,ALTL)X5=XL+ALTL

DOP=LI*U1*2-UWAT**2

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

M5=M
GO TO 8
CONTINUE

8 FACT=(X5-XM5-1177)(XTM5-XM5-11)
FACT1=1.-FACT

WRITE(MOUT+30)XI,UI,HI,DELT1

MOUT=30
X5,UX,H5,DFLT

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
10 Y(N)=RPI*Y(N-1)-R*Y(N-2)
RETURN
END
SURROUTINE HAX4 {INDT, X1, DEL1, THET1, X5, REA, U1, X3, H3, X4, H4}
CURF {H} = 26.703 / H + 305.33 * ALOG(H) - 2111.3 * H + 3327.8 * H * H - 24.03 * H * H * H + 9 * H * H * H
FDEL {X} = EXP(2.573 - 34252.4 * X - 4379.9 * X * X - 0.765111 * X * X * X - 0.0039737 * X * X * X * X)
FAICH {X} = EXP(-3.74811 + 0.038772 * X + 4.1967 * X * X + 0.071046 * X * X * X + 0.0032162 * X * X * X * X)
10 FORMAT //20X,54HA SOLUTION FOR X4 COULD NOT BE OBTAINED IN 1000 TRIALS
11 {MOUT} = 6
C IF IND{NOT} IS NONZERO, THE BOUNDARY LAYER IS TURBULENT
C AT SEPARATION.
C
IF {INDT} = 2, 5, 2
2 H3 = THET1 / DEL1
X3 = X1
DEL3 = DEL1
GO TO 20
5 X3 = X1 * 5, F4 / (U1 * REA)
ARG = ALOG({X3 - X1}) / (REA * DEL1 * DEL1)
H3 = THET1 * FAICH {ARG} / DEL1
DEL3 = 58 * FDEL {ARG} * DEL1
IF {X3 - X5} 20, 15, 15
15 H4 = 0.429
X4 = X5
GO TO 50
20 CONTINUE
C
IGN = 0
DIST = X5 - X1
UNDER = 0
H4 = H3 / H3
COEF1 = DEL3 * H3
COEF2 = 10.5 * DEL3 * H3
SUB = X3 - COEF1 * CURLF {H3}
95 OVER = H4
H4 = 5 * (H4 + UNDER)
X4 = CURLF {H4} * COEF1 + SUB
ALTER = X5 - COEF2 * (1. - (H4 / 0.429) ** 21) / H4
IGN = IGN + 1
IF {X4 - ALTER} 41, 50, 42
41 IF {IGN - 1000} 95, 61, 61
42 IF {ABS {X4 - ALTER} / DIST} 0.011 50, 50, 43
43 UNDER = H4
H4 = 5 * (OVER + H4)
X4 = CURLF {H4} * COEF1 + SUB
ALTER = X5 - COEF2 * (1. - (H4 / 0.429) ** 21) / H4
IGN = IGN + 1
IF {X4 - ALTER} 52, 50, 51
51 IF {IGN - 1000} 43, 61, 61
52 IF {ABS {X4 - ALTER} / DIST} 0.011 50, 50, 95
61 H4 = 0.429
X4 = X5
WRITE {MOUT, 10}
50 CONTINUE
RETURN
END
SUBROUTINE SFTSX(NSPI, XSEP, XATT, XSIG, ANGLE)
DIMENSION XSIG(100)
A = 0.5*(XSEP + XATT)
B = 0.5*(XATT - XSEP)
ARG = 0.
DO 5 N = 1, NSPI
   XSIG(N) = A - B*COS(ARG)
      ARG = ARG + ANGLE
RETURN
END

SFTSX 1
SFTSX 2
SFTSX 3
SFTSX 4
SFTSX 5
SFTSX 6
SFTSX 7
SFTSX 8
SFTSX 9
SFTSX 10
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
    CONTINUE
    RETURN
END
FUNCTION GAM1(ACAP, DX1, PI)

DIMENSION ACAP(30,3)

GAM1 = PI * (-1.5 * ACAP(1,1) - .75 * ACAP(1,3) + 2. * ACAP(1,2) + ACAP(2,2) - .5 * ACGAM1)

1AP(1,3) = 2.5 * 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)*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.
   SUM = SUM + A(N + 1, NU) * (SINT(COUNT + 1.) * THETA) / (COUNT + 1.) - SINT(COUNT - 1.) * THETA / (COUNT - 1.)
   GI = (3. 14159 - THETA * SINT) * (A(1, NU) + 5*A(2, NU)) + 5*SUM - 0.25*GAMMA*(1. + GI)
6   IF(Y - XATT) .LT. B, 8, 7
   IF(DIFF - 1.E-6) .GT. 8, A, 9
   GI = GI + 2.*B*DIFF**(-1.5)*SQRT((XATT - XSEP)*(1. - Y)*(Y - XATT))
   CONTINUE
RETURN
END

102
SUBROUTINE ESIGI(NU, NX, XS, A, Y, SI)
DIMENSION XS(100), A(10, 3)
SUM=0.
DC IC=2, NX
10 SUM=SUM+R(I, NU)*GA(XS(I-1), XS(I), XS(I+1), Y)
SI=R(I, NU)*AINT(XS(I), 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)
ARGA=ABS(X-A)
ARGB=ABS(X-B)
COEF=2.*(B-A)
API=A+1.
BP1=R+1.
IF (ARGA-1.E-6) 2,3,3
2 CA=0.
GO TO 5
3 CA=ALOG(ARGA)
IF(ARGA-1.E-6) 4,5,5
4 CR=0.
GO TO 6
5 CR=ALOG(ARGA)
6 ABINT=(CA-.5)*ARGA**2-(CB-.5)*ARGB**2-(ALOG(API)-.5)*API**2+(ALOG(ARGB)-.5)*BP1**2-COEF*((X-B)**(CB-1.))+BP1*(ALOG(BP1)-1.)
ABINT=ABINT/COEF
RETURN
FND
FUNCTION BINT(XS, XZ, X)
RTS=SQRT(1+XS)
RTZ=SQRT(1+XZ)
BINT=-1-X+RTS*RTZ
IF(XZ-X) 2,3,3
2
RTSX=SQRT(X-XS)
RTZX=SQRT(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((SQRT(XZ-XS)/(RTS+RTZ)))
GO TO 50
5
RTSX=SQRT(XS-X)
RTZX=SQRT(XZ-X)
BINT=BINT+((XZ-XS)*ALCG((RTSX+RTZX)/(RTS+RTZ))-RTSX*RTZX
50
CONTINUE
RETURN
END
SUBROUTINE SCAL(SAL, NSAL, FRZ, ARR, ROBR)
DIMENSION SAL(300)
DELZ=FRZ*ROBR
EN=ARR/FRZ
DO 5 N=1,300
IF(EN-N) 4,4,5
4 NE=N
GO TO 6
5 CONTINUE
6 N=NSAL-NE
EN=FLOAT(NG)
NG1=NG-1
SBL(1)=Q.
DO 7 N=2,NE
7 SAL(N)=SAL(N-1)+DELZ
FRACT=2.2/DELZ
FRAC1=FRAC-1.
R=FRAC**(1./FLOAT(NG1))
SAVE=R
R=R-(R**NG-FRAC*R+FRAC)/(EN*R**NG1-FRAC)
IF(ABS(SAVE-R)<1.E-6) 9,9,8
9 RPL=R+1.
DO 10 N=NE,NSBL
10 SBL(N+1)=RPL*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)
IF(XI-.0001) 2,2,10
2 GC TO (3,4,5,6),J
3 FP=2.53-2.439*ALOG(XI)
GO TO 99
4 FP=3.54-1.725*ALOG(.7071*X1)
GO TO 99
5 FP=4.58-1.725*ALOG(.5*X1)
GO TO 99
6 FP=.10.12
GO TO 99
10 IF(IXI-XITAB(NJ)) 11,11,12
11 NX=N
GO TO 13
12 CONTINUE
13 TX=(XI-XITAB(NX-1))/(XITAB(NX)-XITAB(NX-1))
TXI=1-TX
GO TO (14,15,16,17),J
14 FP=TXI*TAB1(NX-1)+TX*TAB1(NX)
GO TO 99
15 FP=TXI*TAB2(NX-1)+TX*TAB2(NX)
GO TO 99
16 FP=TXI*TAB3(NX-1)+TX*TAB3(NX)
GO TO 99
17 FP=TXI*TAB4(NX-1)+TX*TAB4(NX)
CONTINUE
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.)  
THE = 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

SIMP 1
SIMP 2
SIMP 3
SIMP 4
SIMP 5
SIMP 6
SIMP 7
SIMP 8
SIMP 9
SIMP 10
SUBROUTINE SCRT(XU,YU,XY,YL,NOFF,RCDC,TMAX,CMAX,ST,SC) SECT 1
C PROGRAM TO COMPUTE COEFFICIENTS TN AND CN OF THE FOURIER SERIES SECT 2
C REPRESENTATION OF SECTION THICKNESS AND CAMBER DISTRIBUTIONS SECT 3
DIMENSION XU(30),YU(30),XY(30),YL(30),YUC(30),YLC(30),ST(24),SC(24) SECT 4
1)NUM(50),TBAR(50),CRAR(50) SECT 5
12 FORMAT(///47X,26HINPUT AND COMPUTED OFFSETS/) SECT 6
15,SHYLC/C/) SECT 8
14 FORMAT(3X,3F16.5,8X,3F16.5) SECT 9
NA=6 SECT 10
RNA=6 SECT 11
 AF=FLOAT(NF) SECT 12
MCUT=6 SECT 13
PI = 3.14159 SECT 14
DELT = PI/(2.*RNF) SECT 15
NTC = 2*NF - 1 SECT 16
NINT = NTC + 2 SECT 17
NSIMP = NTC + 1 SECT 18
RDCR=5*RCDC SECT 19
VARY = 0 SECT 20
CA = 0 SECT 21
CB = 0 SECT 22
THETA = 0 SECT 23
DO 89 K=1,NTC SECT 24
THFTA = THETA + DELT SECT 25
XL = .5*(1. + COS(THFTA)) SECT 26
DO 90 LAM=2,NOFF SECT 27
IF(X1-XU(LAM)) 110,90,90 SECT 28
110 YUINT = YULAM-L1 + (X1 - XU(LAM-1))*)YU(LAM) - YU(LAM-1))/XU(LAM) SECT 29
1 - XU(LAM-1)) SECT 30
GC TO 111 SECT 31
5C CONTINUE SECT 32
111 DO 80 LAM=2,NOFF SECT 33
IF(X1-XL(LAM)) 210,80,80 SECT 34
210 YLINT = YL(LAM-1) + (X1 - XL(LAM-1))*)YL(LAM) - YL(LAM-1))/XL(LAM) SECT 35
1 - XL(LAM-1)) SECT 36
GC TO 112 SECT 37
80 CONTINUE SECT 38
112 TBAR(K+1) = .5*(YUINT - YLINT) SECT 39
89 CRAR(K+1) = .5*(YUINT + YLINT) SECT 40
TMAX = 0 SECT 41
CMAX = 0 SECT 42
DO 79 K = 2,NSIMP SECT 43
IF(TBAR(K)-TMAX) 801,802,802 SECT 44
801 TMAX = TBAR(K) SECT 45
802 IF(CRAR(K)-CMAX) 79,702,702 SECT 46
702 CMAX = CRAR(K) SECT 47
75 CONTINUE SECT 48
IF(CMAX-1.E-51) 1201,1202,1202 SECT 49
1201 CMAX=1 SECT 50
1202 CONTINUE SECT 51
IF((TMAX-1.E-51) 1140,1141,1141 SECT 52
1140 TMAX=1 SECT 53
1141 DO 69 K=2,NSIMP SECT 54
TBAR(K) = TBAR(K)/TMAX SECT 55
65 \text{CBAR(K)} = \text{CRAR(K)}/\text{CMAX} \\
70 \text{TBAR(1)} = 0. \\
75 \text{CRAR(1)} = 0. \\
80 \text{TBAR(NINT)} = 0. \\
85 \text{CRAR(NINT)} = 0. \\
90 \text{TTA} = \text{TBAR(NA)} \\
95 \text{TRR} = \text{TBAR(NA+1)} \\
100 \text{TTG} = \text{TBAR(NA+2)} \\
105 \text{TAA} = \text{DEL}*(\text{RNA}-1.1) \\
110 \text{TRR} = \text{TAA} + \text{DEL} \\
115 \text{TCC} = \text{TRR} + \text{DEL} \\
120 \text{XA} = 0.5\times\text{COS(TAA)} \\
125 \text{XB} = 0.5\times\text{COS(TRR)} \\
130 \text{XC} = 0.5\times\text{COS(TCC)} \\
135 \text{SLOPE} = ((\text{TTG}-\text{TBG})*(\text{XB}-\text{XA})/(\text{XC}-\text{XB}) + (\text{TTG}-\text{TAA})*(\text{XC}-\text{XB})/(\text{XC}-\text{XB}))/(\text{SECT} 70) \\
140 \text{IC(XA)} = \text{SECT} 71 \\
145 \text{THETA} = 0. \\
150 \text{COSB} = \text{COS(TBB)} \\
155 \text{DO 456 I=2,NA} \\
160 \text{THETA} = \text{THETA} + \text{DEL} \\
165 \text{COST} = \text{COS(THETA)} \\
170 \text{TBAR(I)} = (\text{SQR}(1.0-\text{COSR})*(1.0-\text{COSB})*0.5*(\text{TTG}-1.0*(\text{COST}-2.0*\text{COSB})))/(\text{SECT} 77) \\
175 1.0-\text{COSR} + 0.5*\text{SLOPE}*(\text{COST}-\text{COSB})) \\
180 \text{NEL} = 2*\text{NF} + 1 - \text{NA} \\
185 \text{COSR} = 1.0 + \text{COS}(\text{PI}-\text{RNA})*\text{DEL} \\
190 \text{THETA} = \text{PI} \\
195 \text{SINAS} = \text{SIN}*(\text{RNA}*\text{DEL})**2 \\
200 \text{COSAS} = \text{COS}*(\text{RNA}*\text{DEL}) \\
205 \text{ANG} = 0. \\
210 \text{DO 457 I=2,NA} \\
215 \text{IND} = 2*\text{NF} + 2 - 1 \\
220 \text{THETA} = \text{THETA} + \text{DEL} \\
225 \text{COST} = 1.0 + \text{COS(THETA)} \\
230 \text{ANG} = \text{ANG} + \text{DEL} \\
235 \text{COEF} = (\text{SINAS} - \text{SIN}*(\text{ANG})*2.0)/(\text{COSR} - \text{COS}*(\text{ANG})*\text{COSAS}) \\
240 \text{TBAR(IND)} = (\text{SQR}*(\text{OBG})*\text{COEF})*\text{MAX} + \text{TBAR(NLE)}*\text{COST} - \text{COSR})**1 \\
245 \text{1.5})/(2.0-\text{COST})) \\
250 \text{THETA} = \text{TAA} \\
255 \text{NA} = \text{NA} + 1 \\
260 \text{DO 458 I = NA, NLE} \\
265 \text{THETA} = \text{THETA} + \text{DEL} \\
270 \text{TBAR(I)} = \text{TBAR(I)}/(1.0-\text{COS(THETA)}) \\
275 \text{THETA} = 0. \\
280 \text{DO 459 I=2,NSIMP} \\
285 \text{THETA} = \text{THETA} + \text{DEL} \\
290 \text{CBAR(I)} = \text{CBAR(I)}*\text{SIN(THETA)} \\
295 \text{RKK} = 0. \\
300 \text{DO 59 K=1,NF} \\
305 \text{RKK} = \text{RKK} + 1. \\
310 \text{THETA} = 0. \\
315 \text{DO 777 I=1, NINT} \\
320 \text{DUM(I) = TBARI(I) * SIN(THETA* RKK)} \\
325 \text{777 THETA = THETA + DEL} \\
330 \text{CALL SIMP( NSIMP, DELT, DUM, VARY) \\
335 \text{STKJ = 2.0*VARY/PI) \\
340 SECT 56 SECT 57 SECT 58 SECT 59 SECT 60 SECT 61 SECT 62 SECT 63 SECT 64 SECT 65 SECT 66 SECT 67 SECT 68 SECT 69 SECT 70 SECT 71 SECT 72 SECT 73 SECT 74 SECT 75 SECT 76 SECT 77 SECT 78 SECT 79 SECT 80 SECT 81 SECT 82 SECT 83 SECT 84 SECT 85 SECT 86 SECT 87 SECT 88 SECT 89 SECT 90 SECT 91 SECT 92 SECT 93 SECT 94 SECT 95 SECT 96 SECT 97 SECT 98 SECT 99 SECT 100 SECT 101 SECT 102 SECT 103 SECT 104 SECT 105 SECT 106 SECT 107 SECT 108 SECT 109 SECT 110
THETA = 0.
DO 888 I = 1, NINT

DUM(I) = CRAR(I) * SIN(THETA*RKK)
END

THETA = THETA + DELT
CALL SIMP(NSIMP, DELT, DUM, VARY)

69  SC(K) = 2. * VARY / PI
    DO 969 I = 1, NOFF
    X = XU(I)
    CALL FVAL(NF, X, SC, ST, CR, TB, CMAX, TMAX)
969  X = XU(I)
    NO 869 T = 1, NOFF
    X = XL(I)
    CALL FVAL(NF, X, SC, ST, CR, TB, CMAX, TMAX)

69  YLC(I) = CB + TB
    SUM1 = 0.
    COUNT = 0.
    DO 659 I = 1, NF
    COUNT = COUNT + 1.
659  SUM1 = SUM1 - ST(I) * COUNT * (-1.) ** I
    RDCBC = 8. * (TMAX * SUM1) ** 2
    RDCAC = 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, NF)
    RETURN
END
SUBROUTINE CORDX(NSRL,NZ,RDBB,SRL,XC)

C BOUNDARY LAYER COORDINATES AND CORRESPONDING CHORDAL
COORDINATES ARE COMPUTED HERE.

DIMENSION SRL(300),X(300),XC(300)

336 FORMAT(1OHX,31HITERATION TO COMPUTE XC FOR M =15.32H DID NOT CONVERGENCE)
1ERGE IN 1000 STEPS. )

337 FORMAT(1H1.25X,1H*M,20X,1HS,25X,1HX,24X,2HXC//)

338 FORMAT(22X,1S3E25.5)

MOUT=6
MX = NSRL + 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,MM
MM = M + 1 - NZ
256 X(M) = SRL(NZ) + SRL(MM)
DO 257 M=1,MM
IF(NZ-M) 333,257,335
257 CONTINUE

333 N = M + 1 - NZ
GO TO 334

334 N = NZ - M + 1
GO TO 335

336 X(M) = -1. + SRL(K)
IF(SRL(K)-RZERO) 341,341,342
341 X(M) = -1. + SRL(K)**2/(4.*RZERO)
342 CONTINUE

DO 258 L=1,1000
SAVE = XC(M)
CALC1 = SQRT((1.+XC(M))/RZERO)
CALC2 = SQRT((1.+XC(M))/RZERO)
X(M) = XC(M) + CALC1*(SRL(K) - RZERO*CALC1*CALC2 + ALOG(CALC1+CALC2) )/CALC2
IF(ABS(SAVE-XC(M))<1.E-6) 257,257,258
258 CONTINUE

WRITE(MOUT,336) M

257 CONTINUE

WRITE(MOUT,337)

DO 264 M=1,MM
IF(NZ-M) 261,261,262
262 K=NZ-M+1
GO TO 263

261 K=M+1-NZ
263 WRITE(MOUT,338) M,SRL(K),X(M),XC(M)
264 CONTINUE

RETURN

END
SUBROUTINE PGRAD(*, X, UE, DXI, PRESS, SA, SR, SC, SR, SS)

C SUBROUTINE FOR CALCULATION OF PRESSURE GRADIENT AND DERIVATIVE COEFFICIENTS.

C

DIMENSION X(300), UE(300, 3)
DIZ = X(M+1) - X(M)
D2Z = X(M+2) - X(M)
D21 = X(M+2) - X(M+1)
D1M1 = X(M+1) - X(M-1)
D2M1 = X(M) - X(M-1)
XIM = DIZ / (D2Z * D21)
ETAM = 1 / (DIZ - 1 / D21)
ZETAM = D21 / (DIZ * 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 / DIZ + 1 / D1M1
SR = D1M1 / (D1Z * D2M1)
SC = D1Z / (D1M1 * D2M1)
SR = D1M1 / D2M1
SS = D1Z / D2M1
RETURN

END

PGRAD 1
PGRAD 2
PGRAD 3
PGRAD 4
PGRAD 5
PGRAD 6
PGRAD 7
PGRAD 8
PGRAD 9
PGRAD 10
PGRAD 11
PGRAD 12
PGRAD 13
PGRAD 14
PGRAD 15
PGRAD 16
PGRAD 17
PGRAD 18
PGRAD 19
PGRAD 20
PGRAD 21
PGRAD 22
PGRAD 23

115
SUBROUTINE TRANS(UPRIM, PRESS, THETA, REB, UC, NY, FLAM, XFLAM, LAM, J)
TRANS 1
C
C SUBROUTINE TO TEST FOR TRANSITION IN A LAMINAR BOUNDARY LAYER.
TRANS 2
C
DIMENSION UC(100,3), FLAM(10), XFLAM(10)
TRANS 3
F(X) = 1.11746 - 1.0582E-3*X - 1.1023E-4*X*X
TRANS 4
TKAY = PRESS*REP*THETA**2/UC(NY,2)
TRANS 5
IF(TKAY < 0.077) 2,2,99
TRANS 6
2 IF(ARS(TKAY) < 0.011) 3,3,4
TRANS 7
3 ARG = TKAY*72.49
TRANS 8
GO TO 5
TRANS 9
5 ARG = 0.
TRANS 10
DO 6 N=1,1000
TRANS 11
SAVE = ARG
TRANS 12
ARG = ARG - (ARG**F(ARG)**2-TKAY)/F(ARG)*1.11746-ARG*3.1746E-3 - ATRANS 13
1RG*ARG*5.5115E-4)
TRANS 14
6 IF(ARS(1.-SAVE/ARG)-1.E-6) 7,7,6
TRANS 15
7 IF(ARG<11.) 8,8,5
TRANS 16
8 EF = 1.75
TRANS 17
GO TO 10
TRANS 18
5 DO 15 N=1,10
TRANS 19
15 IF(ARG-XFLAM(N)) 24,24,15
TRANS 20
24 NBAR = N
TRANS 21
GO TO 16
TRANS 22
16 EF = FLAM(NBAR-1)+(ARG-XFLAM(NBAR-1))*(FLAM(NBAR)-FLAM(NBAR-1))/(XTRANS 23
FLAM(NBAR)-XFLAM(NBAR-1))
TRANS 24
10 R = (.5*EF
TRANS 25
A = 3.36*UPRIM/UC(NY,2)**2
TRANS 26
RTH = F(ARG)*SQR(R*R+9860.*A)-R/A
TRANS 27
IF(REB*THETA-RTH) 99,50,50
TRANS 28
50 LAMQ = 0
TRANS 29
95 CONTINUE
TRANS 30
RETURN
TRANS 31
END
TRANS 32

SUBROUTINE CAPS(ITER,N,CAPG,CAPH,CAPJ,CAPK,SR,SS,SD,SE,SF,VISC,V,UCAPS)
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) 4,2,4
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(N)*VISC(N-1,1)) - SCAPS
    LSN=(SD(N)*VISC(N+1,2)+SE(N)*VISC(N,2)-SF(N)*VISC(N-1,2))
    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(N,1)-SF(N)*VISC(N-1,1))
    CAPK(N)=.5*(CAPK(N)+UC(N,1))
6 CONTINUE
RETURN
END
SUBROUTINE TERP(YIN,YBASE,VARY,NY,VALUE)
C
C SUBROUTINE FOR DETERMINING INTERPOLATED VALUE OF THE
C FUNCTION VARY AT Y = YIN.
C
DIMENSION YBASE(100), VARY(100)
IF(YIN-YBASE(NY-1)) 2,3,3
3 VALUE = VARY(NY)
GO TO 10
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)
   LD2A*DA1*VARY(NBAR+1)/(D31*D32)
10 CONTINUE
RETURN
END

TERP 1
TERP 2
TERP 3
TERP 4
TERP 5
TERP 6
TERP 7
TERP 8
TERP 9
TERP 10
TERP 11
TERP 12
TERP 13
TERP 14
TERP 15
TERP 16
TERP 17
TERP 18
TERP 19
TERP 20
TERP 21
TERP 22
TERP 23
TERP 24
TERP 25
SUBROUTINE YDIFF(Y, ALPHA, BETA, GAMMA, DELTA, SD, SE, SF, C2, C3, C4, Y)  
DIMENSION ALPHA(100), BETA(100), GAMMA(100), DELTA(100)  
DIMENSION SD(100), SE(100), SF(100), Y(100)  
NV = NY + 2  
NVPI = NV + 1  
DC 40 N = 2, NV  
ALPHA(N) = 2. * (sin(N) - Y(N-1) - Y(N)) / ((Y(N+2) - Y(N-1)) * (Y(N+2) - Y(N)))  
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)))  
1 - 1) * (Y(N) - Y(N-1))  
BETA(N) = (DELTA(N) * (Y(N) - Y(N-1))) ** 3 - ALPHA(N) * (Y(N+2) - Y(N)) ** 3 / YDIFF  
1(N+1) - Y(N)) ** 3  
GAMMA(N) = -ALPHA(N) - BETA(N) - DELTA(N)  
CONTINUE  
DC 39 N = 2, NVPI  
SD(N) = (Y(N) - Y(N-1)) / ((Y(N+1) - Y(N)) * (Y(N+1) - Y(N)))  
SE(N) = 1. / (Y(N) - Y(N-1)) - 1. / (Y(N+1) - Y(N))  
SF(N) = (Y(N+1) - Y(N)) / ((Y(N) - Y(N-1)) * (Y(N+1) - Y(N-1)))  
CONTINUE  
C2 = Y(3) * Y(4) / (Y(2) * Y(3) * Y(2))  
C3 = -Y(2) * Y(4) / (Y(3) * Y(4) * Y(2))  
C4 = Y(2) * Y(3) / (Y(4) * Y(3) * Y(2))  
RETURN  
END
SUBROUTINE ELDER(BCAP, XSIG, NSIG, UINF, ELD, Y, YMAX)
DIMENSION RCAP(100, 3), XSIG(100)
BCAP(NSIG+1, 1) = 0.
XS = XSIG(1)
XZ = XSIG(NSIG+1)
IF (XZ - XS) 16, 16, 1
1 DEADL = XZ - XS
YMAX = 1., E-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) - UINF) 20, 20, 12
12 IF (ELD) 16, 16, 16
14 ELD = UINF
GO TO 20
16 ELD = UINF
CONTINUE
RETURN
END
SUBROUTINE REATT (UC, V, X, Y, XM, YM, DRY, UE, X5, DELS, MST, REB) 
DIMENSION UC(100,2), V(100,2), YM(100)
DIMENSION X(300), IF(300,3)
DIMENSION TAR1(24), TAR2(24), TAB3(24), TAB4(24), X(300)
DATA TAR1 /24.98, 23.29, 21.04, 19.33, 17.61, 15.29, 13.46, 11.54, 10.36, 9
DATA TAR2 /20.05, 18.85, 17.25, 16.04, 14.77, 13.03, 10.35, 9.02, 8.54, 7.87, 7.00, 6.24, 5.43, 4.58, 3.63, 2.67, 1.61, 0.55, 0.49, 0.42, 0.35, 0.28, 0.21, 0.14, 0.07, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00
DATA TAB3 /15.81, 14.67, 13.52, 12.37, 11.22, 10.07, 8.92, 7.77, 6.62, 5.47, 4.32, 3.17, 2.02, 0.87, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00
DATA TAB4 /2.00, 3.00, 4.00, 5.00, 6.00, 7.00, 8.00, 9.00, 10.00, 11.00, 12.00, 13.00, 14.00, 15.00, 16.00, 17.00, 18.00, 19.00, 20.00
DATA X(300) /0.00, 1.00, 2.00, 3.00, 4.00, 5.00, 6.00, 7.00, 8.00, 9.00, 10.00, 11.00, 12.00, 13.00, 14.00, 15.00, 16.00, 17.00, 18.00, 19.00, 20.00, 21.00, 22.00, 23.00, 24.00, 25.00, 26.00, 27.00, 28.00, 29.00, 30.00
DATA IF(300,3) /300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300, 300
FORMAT(///40%923HAT REATTACHMENT? RFT4 =E13051 REATT
REATT 1 5
REATT 2 3
REATT 3 2
REATT 4 4
REATT 5 5
REATT 6 6
REATT 7 7
REATT 8 8
REATT 9 9
REATT 10 10
REATT 11 11
REATT 12 12
REATT 13 13
REATT 14 14
REATT 15 15
REATT 16 16
REATT 17 17
REATT 18 18
REATT 19 19
REATT 20 20
REATT 21 21
REATT 22 22
REATT 23 23
REATT 24 24
REATT 25 25
REATT 26 26
REATT 27 27
REATT 28 28
REATT 29 29
REATT 30 30
REATT 31 31
REATT 32 32
REATT 33 33
REATT 34 34
REATT 35 35
REATT 36 36
REATT 37 37
REATT 38 38
REATT 39 39
REATT 40 40
REATT 41 41
REATT 42 42
REATT 43 43
REATT 44 44
REATT 45 45
REATT 46 46
REATT 47 47
REATT 48 48
REATT 49 49
REATT 50 50
REATT 51 51
REATT 52 52
REATT 53 53
REATT 54 54
REATT 55 55
\[ V(N,1) = V(N-1,1) - (Y(N) - Y(N-1)) \times (\text{DUDXP} + \text{DUDX}) \]

\[ V(N,2) = V(N,1) \]

\[ \text{DUDX} = \text{DUDXP} \]

RETURN

END
SUBROUTINE ELPIT(ALPH1, ALPH2, EMI, TORF, THETZ, UINF, DXI, CMPA, CMPAS)

SAVF = ALPH1
STEP = TORF + DXI
SINS = SIN(STEP)
COSS = COS(STEP)

CONST = 2. * EMI * (UINF / TORF)**2

ALPH1 = THETZ + (ALPH1 - THETZ) * COSS + ALPH2 * SINS + TORF * CONST * (2. * CMPA - CMPAS)

ALPH2 = ALPH2 * COSS - TORF + SINS * (SAVF - THETZ) + CONST * (CMPA - CMPAS) * (1. - COE)

1SS) / DXI + CONST * CMPA * TCRA * SINS

RETURN

END
SUBROUTINE VWASH(BARG,H,S,NVOR,X1,UINF,ZIPI,XGA4,NGPI,DXI)
DIMENSION ZIPI(30),XGM(3))
DO 10 N=1,NGPI
   DIFF=XGM(N)-X1
   SUM=0.
   DO 5 K=1,NVOR
      SUM=SUM+DIFF/(DIFF*DIFF+H)
   5   DIFF=DIFF-S
10  ZIPI(N)=ZIPI(N)+SUM*BARG
RETURN
END
SLAROUTINE WASH(XGAM,NGAM,TIME,ALPH1,ALPH2,HEAVE,AROT,FREQF,PHIH,UVASI 1
1INF,CAMRR,NF,VZIP,MOTR,INDV)
DIMENSION XGAM(301),VZIP(301),CAMRR(24)
NGP1 = NGAM+1
ANGLE = FREQF*TIME
GO TO (100,120), INDV
108 GO TO (110,120), MOTR
110 CNST = -ALPH2*COS(ANGLE)*UINF + HEAVE*COS(ANGLE + PHIH) + ALPH1*UINF
FACT = -ALPH2*FREQF*SIN(ANGLE)*JINF
GO TO 130
120 CNST = UIINF*ALPH1 + HEAVE
FACT = UIINF*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 * CAMRR(N) * CCS(COUNT * THETA)
IF(M-1) 2,4,2
IF(NP1=M) 3,4,3
4 SUM = SUM + SUM
GO TO 50
3 COUNT = 0.
COUNT = COUNT + THETA
50 CONTINUE
COTT = X/SIN(THETA)
DO 30 N=1,NF
COUNT = COUNT + THETA
30 SUM = SUM + COTT * CAMRR(N) * SIN(COUNT)
50 VZIP(M) = UINF*SUM+CNST+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

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

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

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

$$B_6 = 1 - \frac{M_{\beta \theta}^2}{M_{\beta \beta} M_{\theta \theta}} - \frac{M_{\beta \phi}^2}{M_{\beta \phi} M_{\theta \theta}}$$
The characteristic equation for the two-dimensional system is found to be

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

where

$$D_{0} = f_{0} - \bar{\omega} \phi^{2} a_{2} - \bar{\omega}^{2} h_{b} b_{2}$$

$$D_{2} = f_{2} - \bar{\omega} \phi^{2} g_{a} \bar{x} a_{2} - \bar{\omega}^{2} g_{b} \bar{x} b_{2}$$

$$D_{4} = f_{4} - C_{4} \bar{x}^{2} - g_{a} \bar{x} a_{2} - g_{b} \bar{x} b_{2}$$

$$D_{6} = 1 - C_{6} \bar{x}^{2}$$

in which

$$h_{a} = \frac{M_{\beta \beta}}{R^{2} M_{\theta \theta}} \quad h_{b} = \frac{M_{\phi \phi}}{M_{\theta \theta}}$$
\[ g_a = 2 h_a A_1 \quad g_b = 2 h_b A_2 \]

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

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

\[ a_1 = A_1 \left( \bar{\omega}_\beta^2 l_{s_1} + r_m \bar{\omega}^2 l_{s_2} \right) - B \bar{\omega}_\beta^2 l_{s_2} \]

\[ b_1 = A_2 \left( \bar{\omega}_\beta^2 l_{s_1} + r_m \bar{\omega}^2 l_{s_2} \right) + B \bar{\omega}_\beta^2 l_{s_2} \]

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 \( \bar{x} \), \( l_{s_1} \), and \( l_{s_2} \). If \( a_1 \) and \( b_1 \) are eliminated, the following equation for \( \bar{x} \) 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

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

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

\[ t_1 = (1 - c_6 \bar{x}^2) B_2/B_6 - f_2 + \bar{\omega}_\beta^2 F + \frac{h_b F^2}{g_b^2 \bar{x}^2} \]
\[ t_2 = (1 - c_6 \bar{x}^2)(B_2 - B_0/\bar{\omega}_\beta^2)/B_6 - f_2 + \bar{\omega}_\beta^2 F + f_\sigma/\bar{\omega}_\beta^2 \]

in which

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

With some algebraic manipulation, a polynomial of fourth degree in $\bar{x}^2$ 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 $l_{s1}$ and $l_{s2}$ are readily obtained.
REFERENCES


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