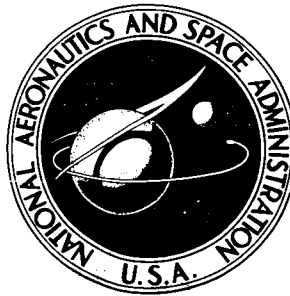


**NASA CONTRACTOR
REPORT**



NASA CR-2322

NASA CR-2322

**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

1. Report No. NASA CR-2322	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle ANALYSIS OF STALL FLUTTER OF A HELICOPTER ROTOR BLADE		5. Report Date November 1973	6. Performing Organization Code
		8. Performing Organization Report No.	
7. Author(s) Peter Crimi		10. Work Unit No.	
9. Performing Organization Name and Address AVCO Systems Division Wilmington, Massachusetts 01887		11. Contract or Grant No. NAS1-11378	
		13. Type of Report and Period Covered Contractor Report	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		14. Sponsoring Agency Code	
		15. Supplementary Notes The contract research effort which has lead to the results in this report was financially supported by USAAMRDL (Langley Directorate). This is a final report.	
16. 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.			
17. Key Words (Suggested by Author(s)) Helicopter rotor, Aeroelasticity, Dynamic stall, Torsional stability		18. Distribution Statement Unclassified - Unlimited	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 133	22. Price* Domestic, \$4.50 Foreign, \$7.00

ANALYSIS OF STALL FLUTTER
OF A HELICOPTER ROTOR BLADE

By Peter Crimi
Avco Systems Division

SUMMARY

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

INTRODUCTION

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

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

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

A method was recently developed for analyzing dynamic stall of an airfoil undergoing arbitrary pitching and plunging motions which provides an ideal tool for analyzing the stall problem in forward flight. The method, which is described in detail in Ref. 7, employs models for each of

the basic flow elements contributing to the unsteady stall of a two-dimensional airfoil. Calculations of the loading during transient and sinusoidal pitching motions are in good qualitative agreement with measured loads. Dynamic overshoot, or lift in excess of the maximum static value, as well as unstable moment variation, are in clear evidence in the computed results.

This study was directed to analyzing the aeroelastic stability of a helicopter rotor, particularly as it relates to stall, using the method of Ref. 7 to compute aerodynamic loading. The representation of the elastomechanical system includes flapping and flapwise bending degrees of freedom as well as torsion. A listing of the computer program used to perform the calculations is given in Appendix A.

SYMBOLS

b	blade semichord, m
\bar{C}_L	mean lift coefficient, ratio of time average of l to $\rho \Omega^2 R^2 b$
C_1	lift coefficient, $C_1 = c_1 / (\rho U^2 b)$
C_m c/4	moment coefficient referred to quarterchord, C_m c/4 = $m_{c/4} / (2 \rho U^2 b^2)$
c	blade chord, m
f_θ	mode shape of first uncoupled torsional mode, unit tip deflection
f_ϕ	mode shape of first uncoupled flapwise bending mode, unit tip deflection
h_β	tip deflection due to flapping, semichords
h_ϕ	tip deflection due to bending, semichords
h_i	translational coordinates of 2-D system ($i = 1, 2$), semichords
I_0	moment of inertia of 2-D system about pitch axis, kg - m
I'_θ	blade moment of inertia about elastic axis per unit span, kg - m
k_i	translational spring stiffnesses of 2-D system ($i = 1, 2$), N/m ²
k_θ	torsional spring stiffness of 2-D system, N/rad
l	lift per unit span at aerodynamic reference radius, N/m
l_{s_i}	offsets of springs from pitch axis of 2-D system ($i = 1, 2$), m
M_b	total blade mass, kg
m	blade mass per unit span, kg/m
m c/4	aerodynamic moment per unit span at aerodynamic reference radius, N

m_1	masses of 2-D system, kg/m
R	rotor radius, m
r_0	inner radius of blade lifting surface, m
r_R	aerodynamic reference radius, m
U	instantaneous free-stream speed at aerodynamic reference section, m/sec
U_0	reference speed, $U_0 = \Omega r_R$, m/sec
x_m	distance aft of elastic axis of blade section mass center, m
\bar{x}	distance aft of pitch axis of mass center of m_1 , m
Z_β	generalized coordinate of 2-D system, equivalent to h_β , semichords
Z_ϕ	generalized coordinate of 2-D system, equivalent to h_ϕ , semichords
α	angle of attack, deg
δ	flapping hinge offset, m
θ_0	collective pitch angle, deg or rad
θ_1	blade tip torsional deflection, rad
$\tilde{\theta}$	angle of zero restraint of 2-D system torsion spring, rad
μ	advance ratio, ratio of forward speed to ΩR
ρ	free-stream density, kg/m ³
τ	dimensionless time, $\tau = U_0 t/b$
ψ	blade azimuth angle measured from downwind direction, deg or rad
Ω	rotor rotational speed, rad/sec
Ω^*	dimensionless rotor speed, $\Omega^* = \Omega R / (\omega_{\theta_0} b)$
ω_f	flutter frequency, rad/sec

ω_{θ_0}

frequency of first uncoupled, nonrotating
torsion mode, rad/sec

ω_{ϕ_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 1a): (1) the laminar boundary layer from the stagnation point to separation near the leading-edge, (2) the small leading-edge separation bubble; and, (3) a potential flow, including a vortex wake generated by the variation with time of the circulation about the airfoil. When the airfoil is stalled, as indicated in Figure 1b, the flow elements are: (1) the laminar boundary layer, (2) a dead-air region extending from the separation point to the pressure recovery point; and, (3) a potential flow external to the airfoil and dead-air region, again including a vortex wake. The analytic representations of these elements are described briefly below. Details are given in Ref. 7.

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

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

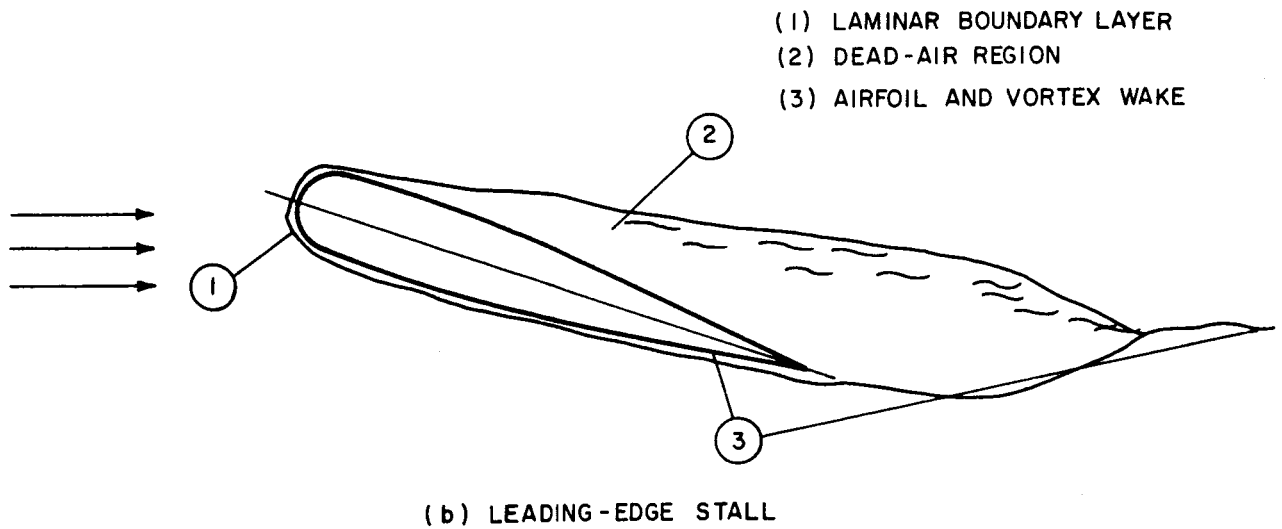
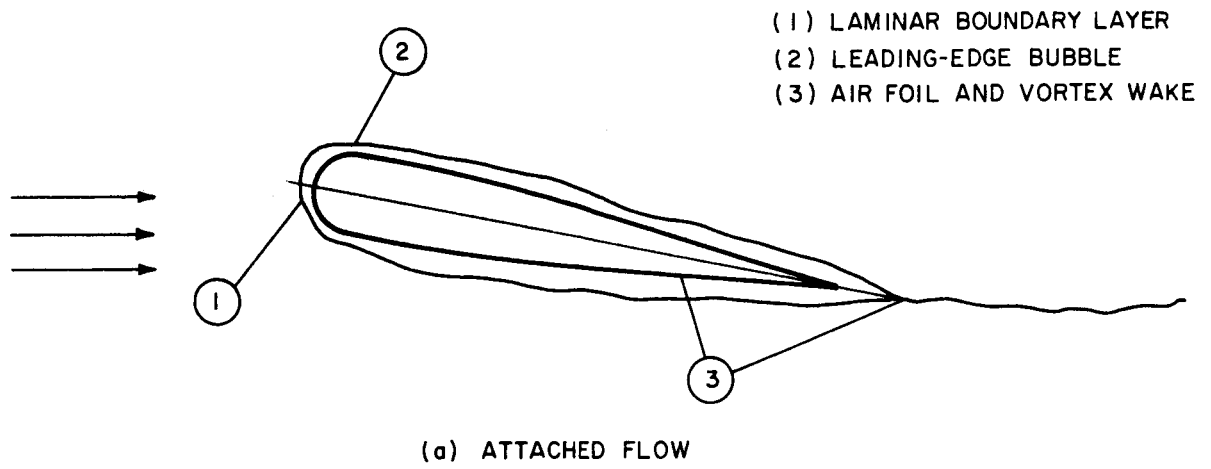


Figure 1 FLOW ELEMENTS

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 unstart, 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)

$$\begin{aligned} \frac{d^2 h_\beta}{d\tau^2} + \frac{R}{b} \frac{M_{\beta\theta}}{M_{\beta\beta}} \frac{d^2 \theta_1}{d\tau^2} + \bar{\omega}_\beta^2 h_\beta - \frac{R}{b} \bar{\Omega}^2 \frac{T_{\beta\theta}}{M_{\beta\beta}} \theta_1 \\ = \frac{Rb}{U_o^2} \frac{F_\beta}{M_{\beta\beta}} \end{aligned}$$

$$\begin{aligned} \frac{d^2 h_\phi}{d\tau^2} + \frac{M_{\phi\theta}}{b M_{\phi\phi}} \frac{d^2 \theta_1}{d\tau^2} + \bar{\omega}_\phi^2 h_\phi - \bar{\Omega}^2 \frac{T_{\phi\theta}}{M_{\phi\phi}} \theta_1 \\ = \frac{b}{U_o^2} \frac{F_\phi}{M_{\phi\phi}} \end{aligned}$$

$$\begin{aligned} \frac{d^2 \theta_1}{d\tau^2} + \frac{b}{R} \frac{M_{\beta\theta}}{M_{\theta\theta}} \frac{d^2 h_\beta}{d\tau^2} + \frac{b M_{\phi\theta}}{M_{\theta\theta}} \frac{d^2 h_\phi}{d\tau^2} + \bar{\omega}_\theta^2 \theta_1 \\ - \frac{b}{R} \bar{\Omega}^2 \frac{T_{\beta\theta}}{M_{\theta\theta}} h_\beta - \bar{\Omega}^2 \frac{b T_{\phi\theta}}{M_{\theta\theta}} h_\phi \\ = \frac{b^2 F_\theta}{U_o^2 M_{\theta\theta}} \end{aligned}$$

where h_β and h_ϕ are tip displacements due to flapping and bending, respectively, in semichords, θ_1 is torsional displacement at the blade tip and the frequencies* are the following functions of rotational speed:

$$\bar{\omega}_\beta^2 = - \bar{\Omega}^2 \frac{T_{\beta\beta}}{M_{\beta\beta}}, \quad \bar{\omega}_\phi^2 = \bar{\omega}_{\phi_0}^2 - \bar{\Omega}^2 \frac{T_{\phi\phi}}{M_{\phi\phi}},$$

$$\bar{\omega}_\theta^2 = \bar{\omega}_{\theta_0}^2 - \bar{\Omega}^2 \frac{T_{\theta\theta}}{M_{\theta\theta}}$$

The inertial and centrifugal-force coefficients are given by

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

$$M_{\theta\theta} = \int_\delta^R I'_\theta f_\theta^2 dr,$$

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

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

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

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

$$T_{\phi\phi} = - \int_{\delta}^R f_{\phi}'^2 \left\{ \int_r^R r_1 m(r_1) dr_1 \right\} dr,$$

$$T_{\theta\theta} = - M_{\theta\theta}, \quad T_{\beta\theta} = - M_{\beta\theta},$$

$$T_{\phi\theta} = \int_{\delta}^R (r - \delta) f_{\phi}' f_{\theta}' m x_m dr$$

The complexity of the aerodynamic representation precludes evaluation of the generalized forces F_{β} , F_{ϕ} and F_{θ} 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 θ_1 should correspond to blade torsional displacement at the blade tip. The counterparts of flapping and bending, Z_{β} and Z_{ϕ} , respectively, are defined by

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

$$\text{where } A_1 = \frac{\bar{\omega}_{\beta}^2 - \bar{\omega}_2^2}{\bar{\omega}_{\phi}^2 - \bar{\omega}_{\beta}^2}, \quad A_2 = \frac{\bar{\omega}_2^2 - \bar{\omega}_{\phi}^2}{\bar{\omega}_{\phi}^2 - \bar{\omega}_{\beta}^2},$$

$$B = \frac{(\bar{\omega}_2^2 - \bar{\omega}_{\phi}^2)(\bar{\omega}_2^2 - \bar{\omega}_{\beta}^2)}{(\bar{\omega}_{\phi}^2 - \bar{\omega}_{\beta}^2) \bar{\omega}_2^2} \quad (1)$$

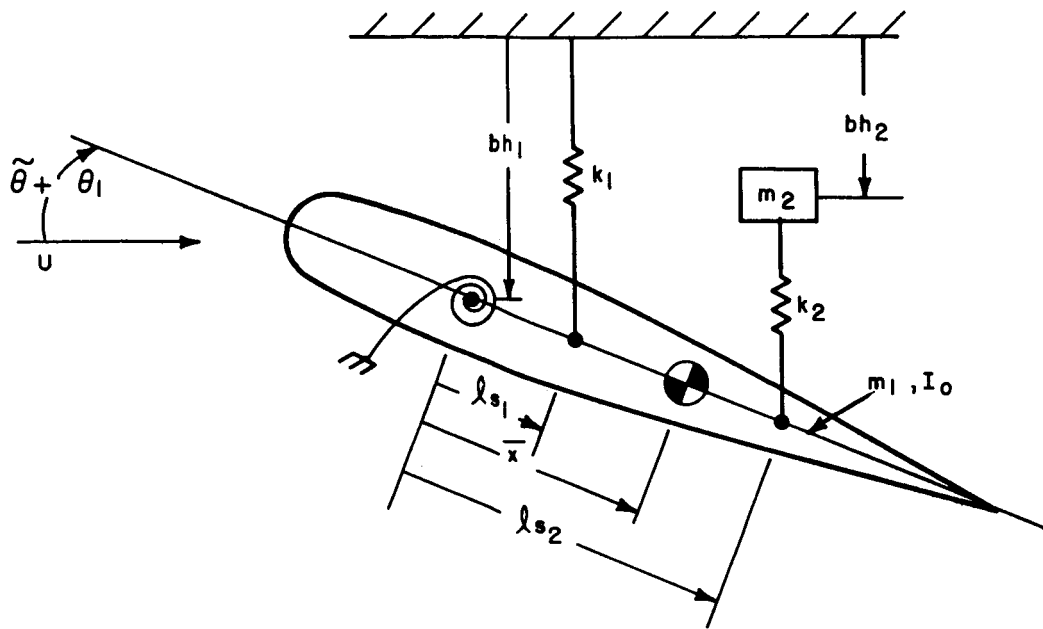


Figure 2 TWO-DIMENSIONAL ELASTOMECHANICAL SYSTEM

$$\text{and } \bar{\omega}_1^2 = (k_1/m_1)(b/U_0)^2, \quad i = 1, 2.$$

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

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

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

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

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

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

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

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

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

$$m_1 R/(-A_1) = 4 \frac{r_R}{R} \frac{M_{\beta\beta}}{R^2 [1 - (r_o/R)^4]}$$

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

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

$$\tilde{\theta} = \theta_o [1 - 2 (R/r_R) \mu \sin \psi]$$

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

BLADE PARAMETERS FOR NOMINAL CONFIGURATION

<u>Parameter</u>	<u>Value</u>
b/R	.0435
δ/R	.0543
r_o/R	.174
$\omega_{\theta_o}/\omega_{\phi_o}$	3.69
$\rho R b^2/M_b$.00431
x_m/b	.216
$m R/M_b$	1.055
$I'_{\theta}/M_b R$	3.51×10^{-4}

Two elastomechanical configurations in addition to the nominal one were analyzed. One of these had $\omega_{\theta_o}/\omega_{\phi_o} = 2.5$, with all other parameters as listed in Table 1. The third configuration had $x_m/b = .108$, with the remaining parameters as listed in Table 1.

The bending mode shape, which was computed by a finite-element method, was found not to vary appreciably over the range of rotational speeds of interest. The mode shape for $\omega_{\phi_o}/\Omega = 1.26$, which is plotted in Figure 3, was used for all computations. The torsional mode shape for the nonrotating blade, also shown in Figure 3, was used to evaluate torsional inertia parameters.

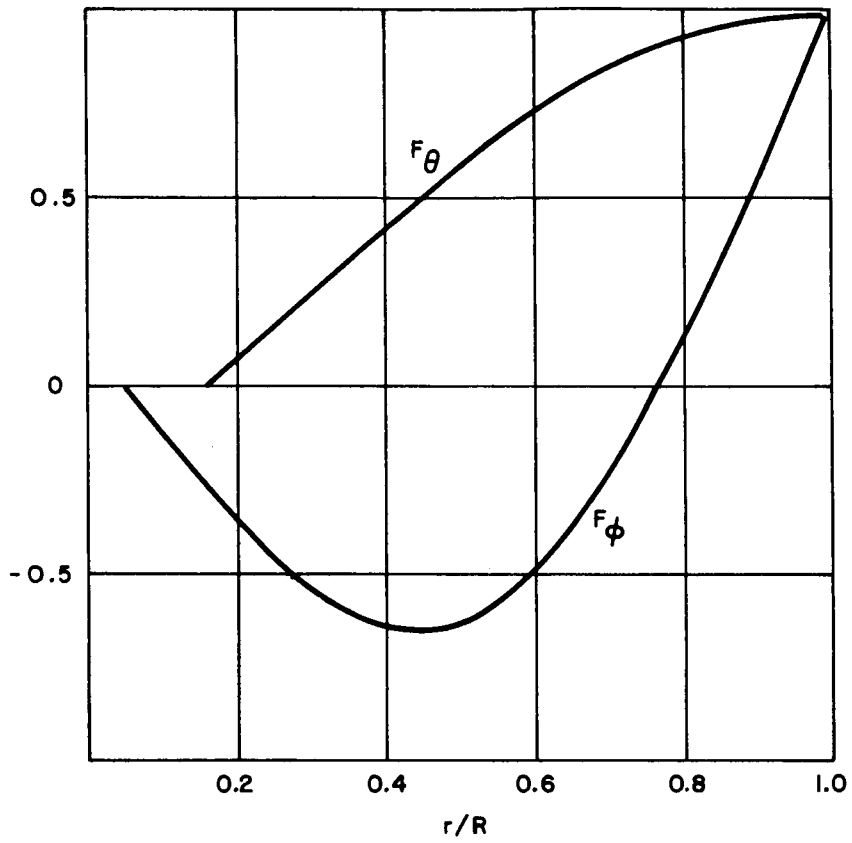


Figure 3 BENDING AND TORSION MODE SHAPES

The test blade had a NACA 23012 section. The variation of static lift and moment coefficients with angle of attack for this section were computed from a series of transient pitch calculations, and are shown in Figure 4, together with the measured section characteristics, from Ref. 13. The aerodynamic model is seen to give nearly the correct maximum lift, but at a slightly lower angle of attack, and, as indicated from the variation of $C_m c/4$, the computed center of pressure is somewhat further aft than that of the actual airfoil section below the stall angle.

Stability in Hover

Initial calculations were performed for hovering flight, with the nominal configuration, to allow a direct comparison with the test results of Ref. 2. First, rotor speed was varied parametrically, with the collective pitch at a value well below the stall incidence. A classical bending-torsion instability was encountered at $\Omega^* \equiv \Omega R / (\omega_{\theta_0} b) = 5.3$ 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 Ω^* 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 Ω^* as low as 3.4. Computed blade motions for stall flutter at Ω^* of 3.5 are shown in Figure 6.

For Ω^* below 3.4, a limit cycle could not be set up, regardless of the initial conditions or the collective pitch angle. Severe oscillations involving repeated stall and unstall could be made to occur by imposing a large initial bending deflection. However, the flapping response modulated the torsional response, and caused continuous stall and/or unstall of the blade over a significant portion of

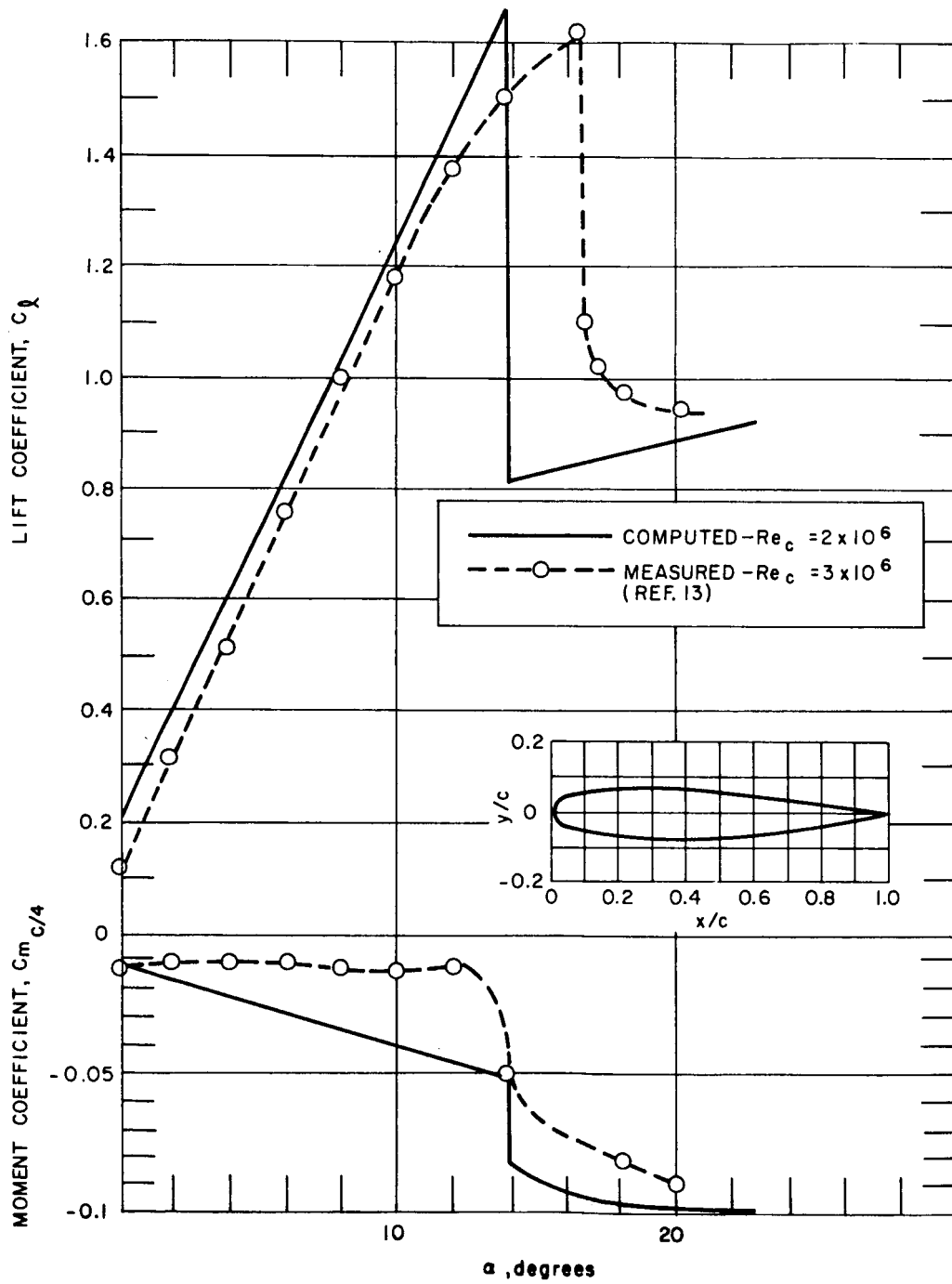


Figure 4 AIRFOIL SECTION CHARACTERISTICS FOR NACA 23012

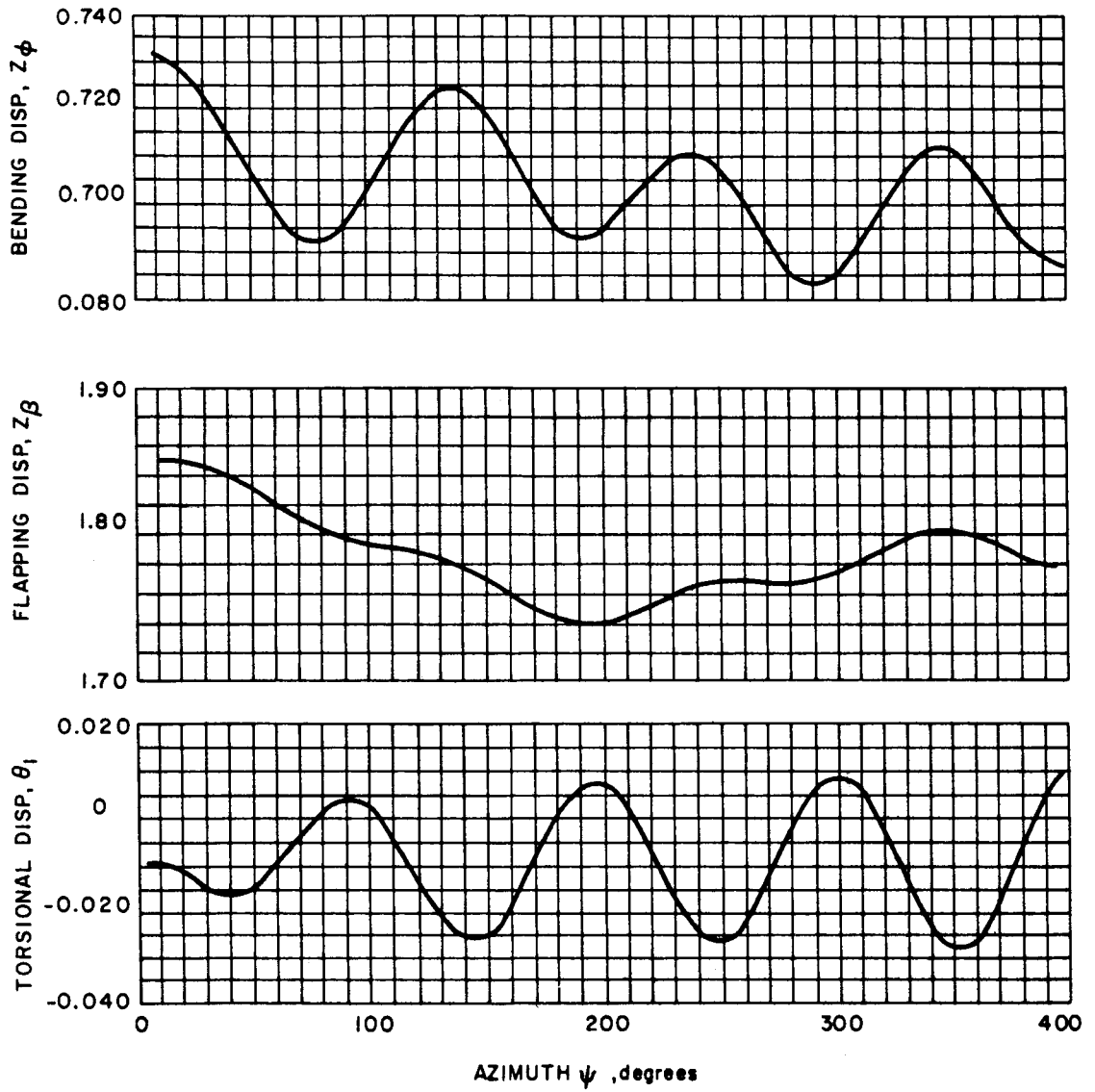


Figure 5 DISPLACEMENT TIME HISTORIES AT CLASSICAL FLUTTER ONSET
 $\Omega^* = 5.3, \theta_0 = 11 \text{ deg}, \mu = 0$

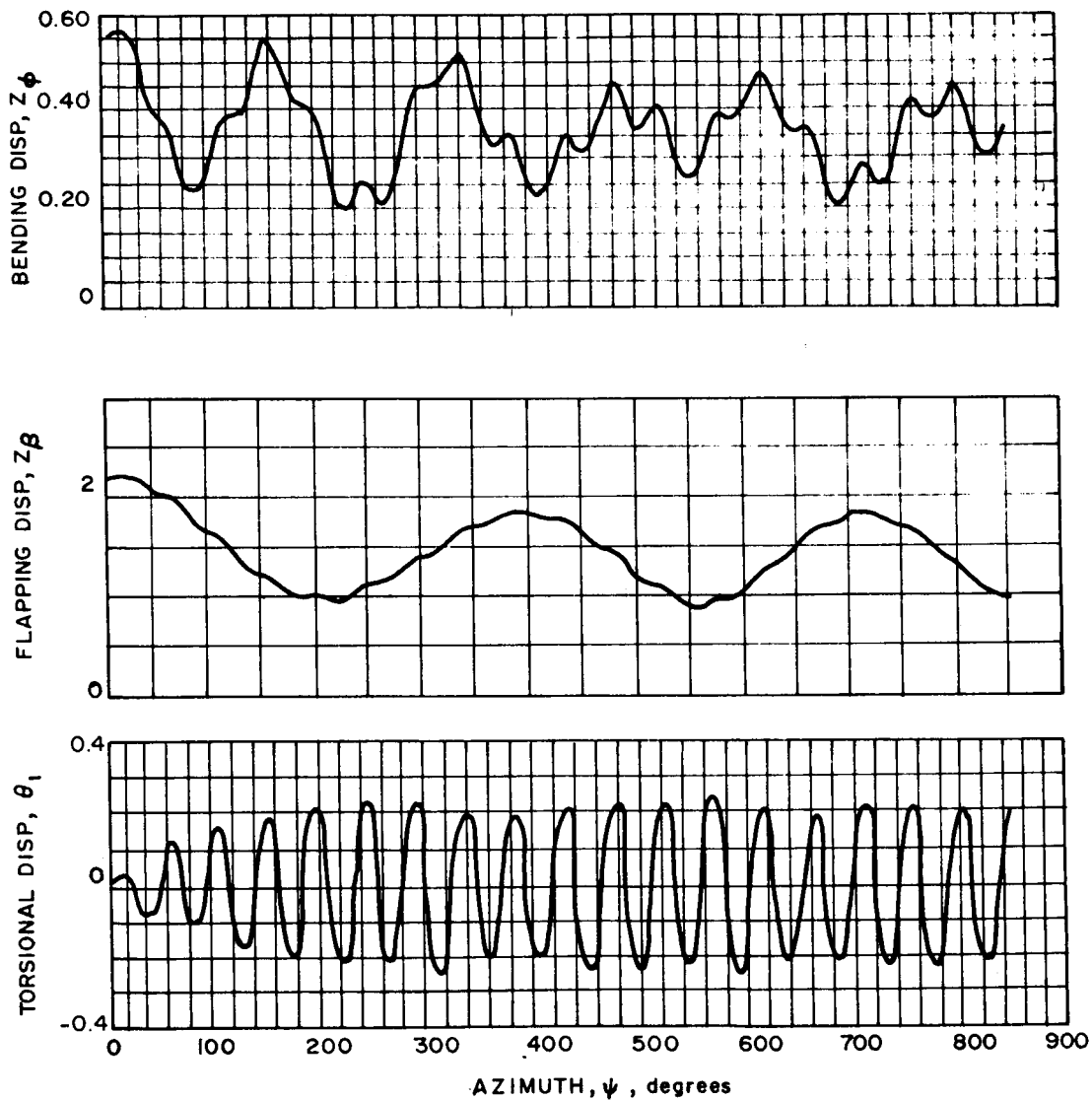


Figure 6 DISPLACEMENT TIME HISTORIES FOR STALL FLUTTER
 $\Omega^* = 3.5, \theta_0 = 15.0 \text{ deg}, \mu = 0$

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. 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 Ω^* 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

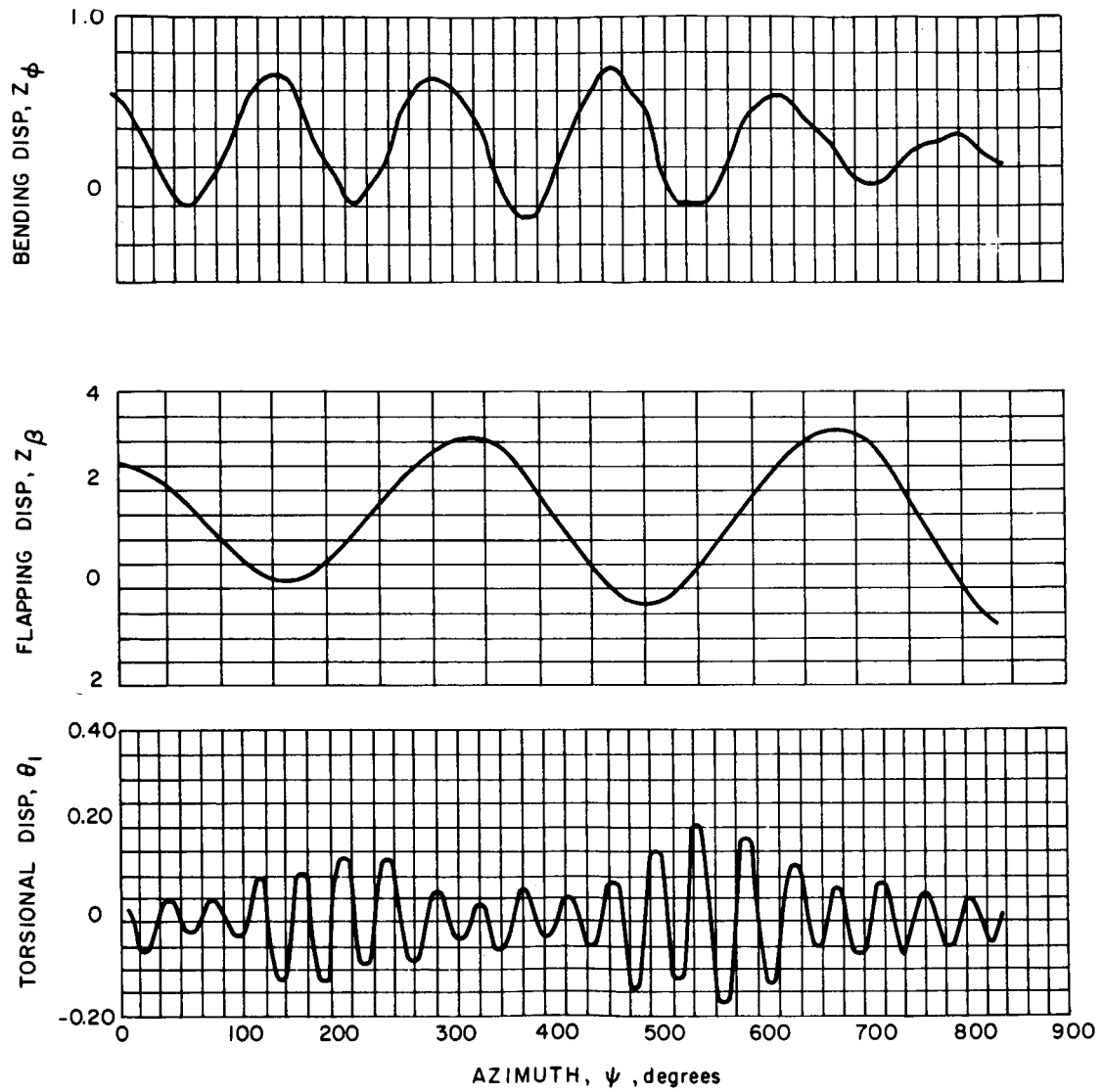


Figure 7 BLADE RESPONSE BELOW STALL FLUTTER BOUNDARY
 $\Omega^* = 3.1, \theta_0 = 15.0 \text{ deg}, \mu = 0$

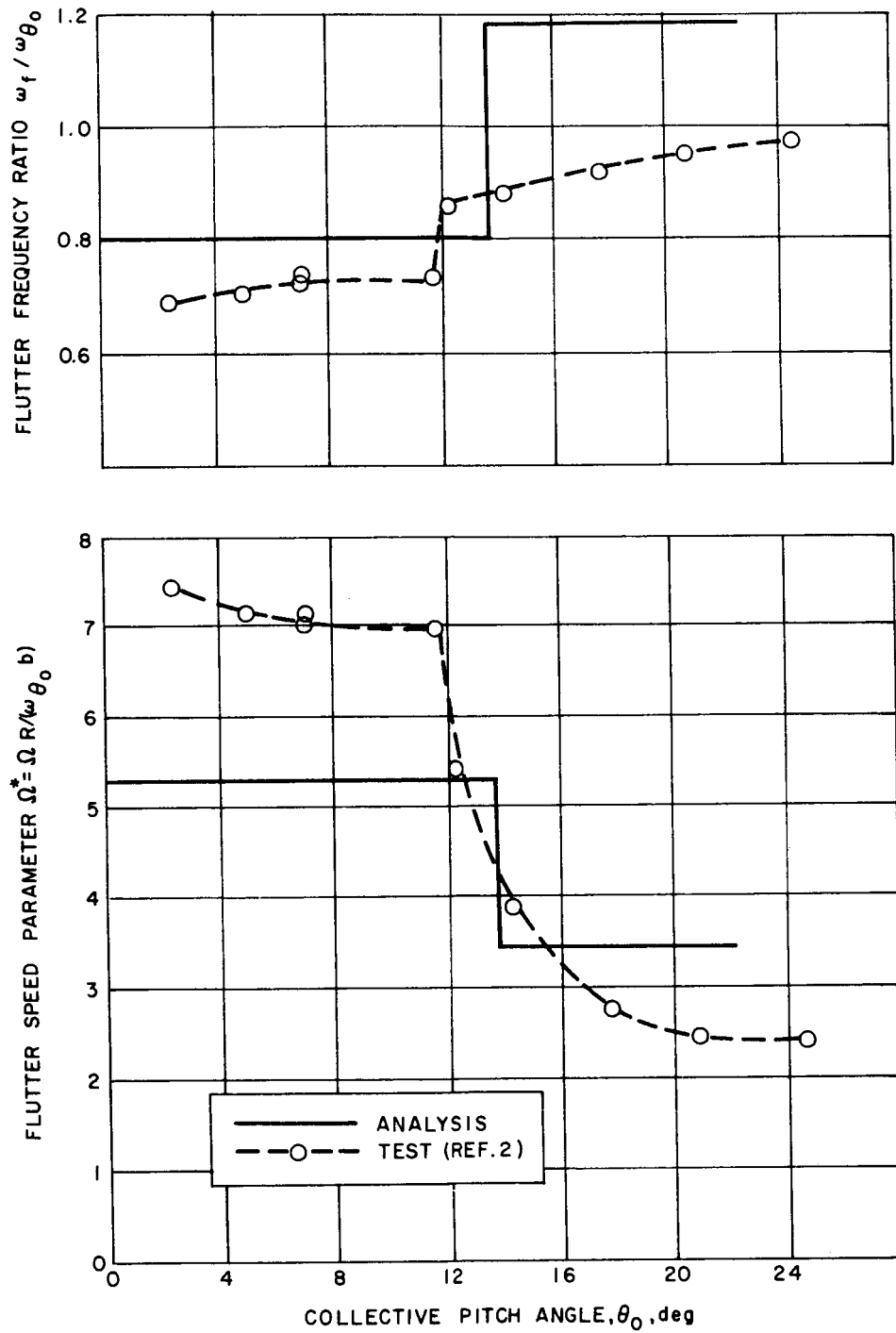


Figure 8 FLUTTER SPEED AND FREQUENCY VARIATION WITH COLLECTIVE PITCH ANGLE FOR A HOVERING ROTOR

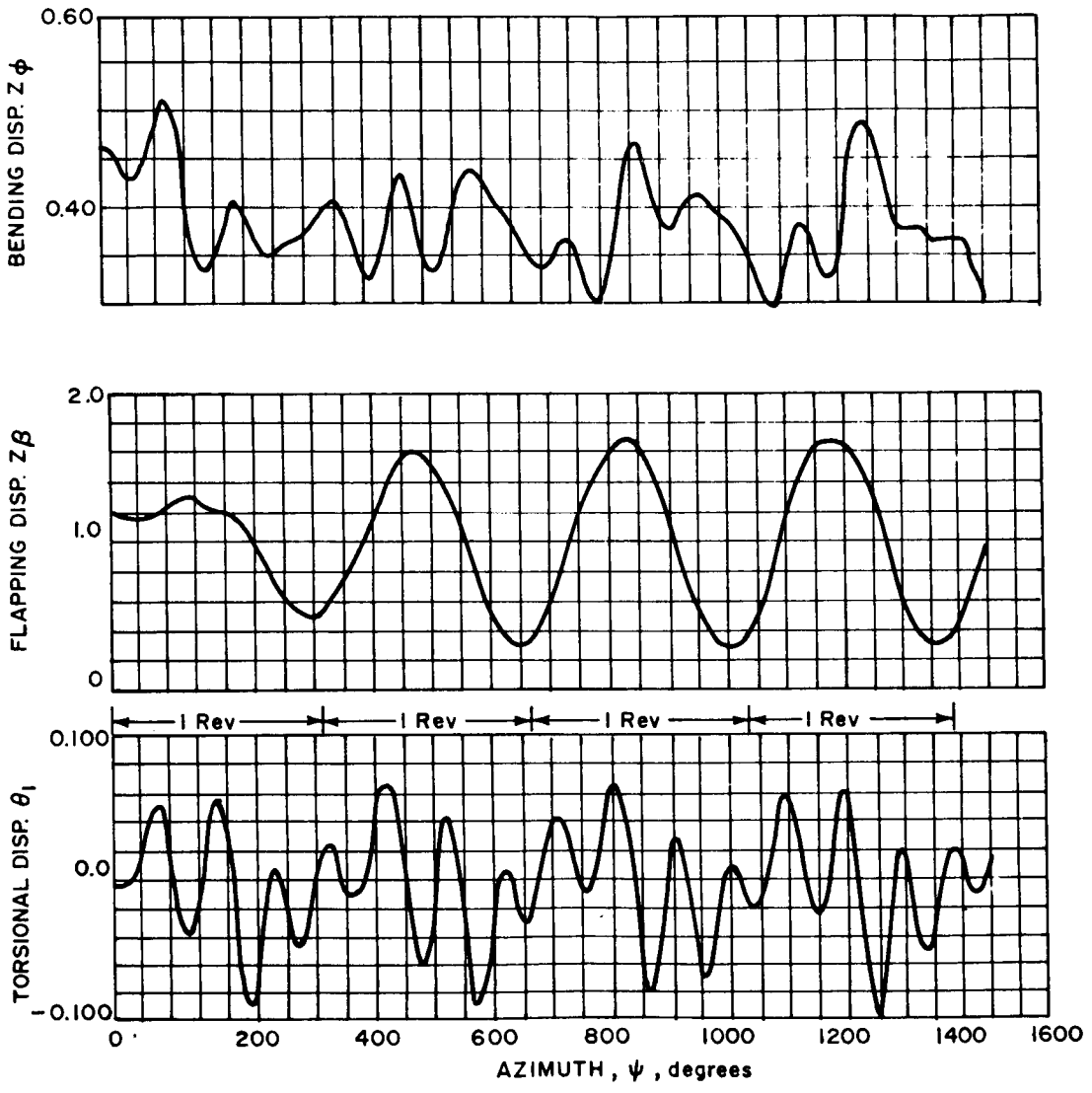


Figure 9 DISPLACEMENT TIME HISTORIES AT LINEAR INSTABILITY ONSET
 $\Omega^* = 5.2, \theta_0 = 6 \text{ deg}, \mu = 0.1$

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 $\psi = 270$ deg and $\psi = 400$ deg, rather than throughout a complete revolution of the blade.

A torsional displacement time history closely resembling the variation of control loads in Figure 11 was obtained for 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 ($\Omega^* = 4.15$, Figure 13). However, the major contributors to the oscillations appear to be the initial and final pulses associated with stall and unstall upon entering and leaving that zone. There are, in general, two cycles of torsional oscillation of excessive amplitude after the blade unstalls the last time on a given revolution. The response can be regarded as transient, on a localized time scale, or forced, when viewed on a scale of several rotor revolutions. The severity of the response is apparently due in part to the suddenness of load changes at stall and unstall, and partly to the relative lack of aerodynamic damping in pitch, particularly when the blade is not stalled.

If the collective pitch angle is increased, the blade does undergo stall flutter, as seen from the time history plotted in Figure 14. These results are for the same rotor

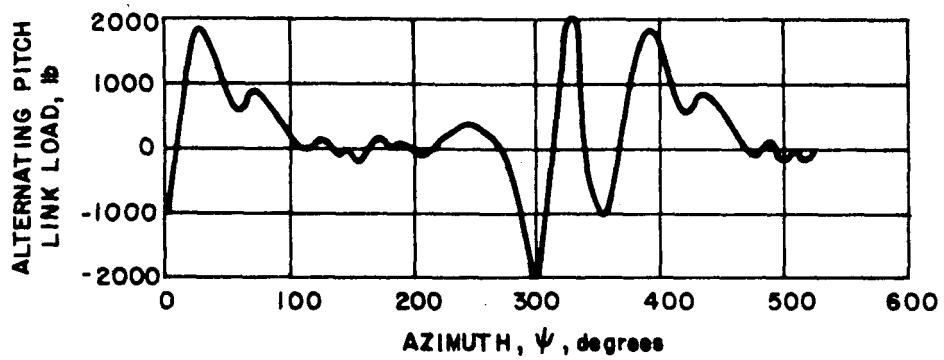


Figure 11 VARIATION OF PITCH LINK LOAD IN FLIGHT
 TEST OF CH-47 AT 123 KNOTS
 (from Ref. 6)

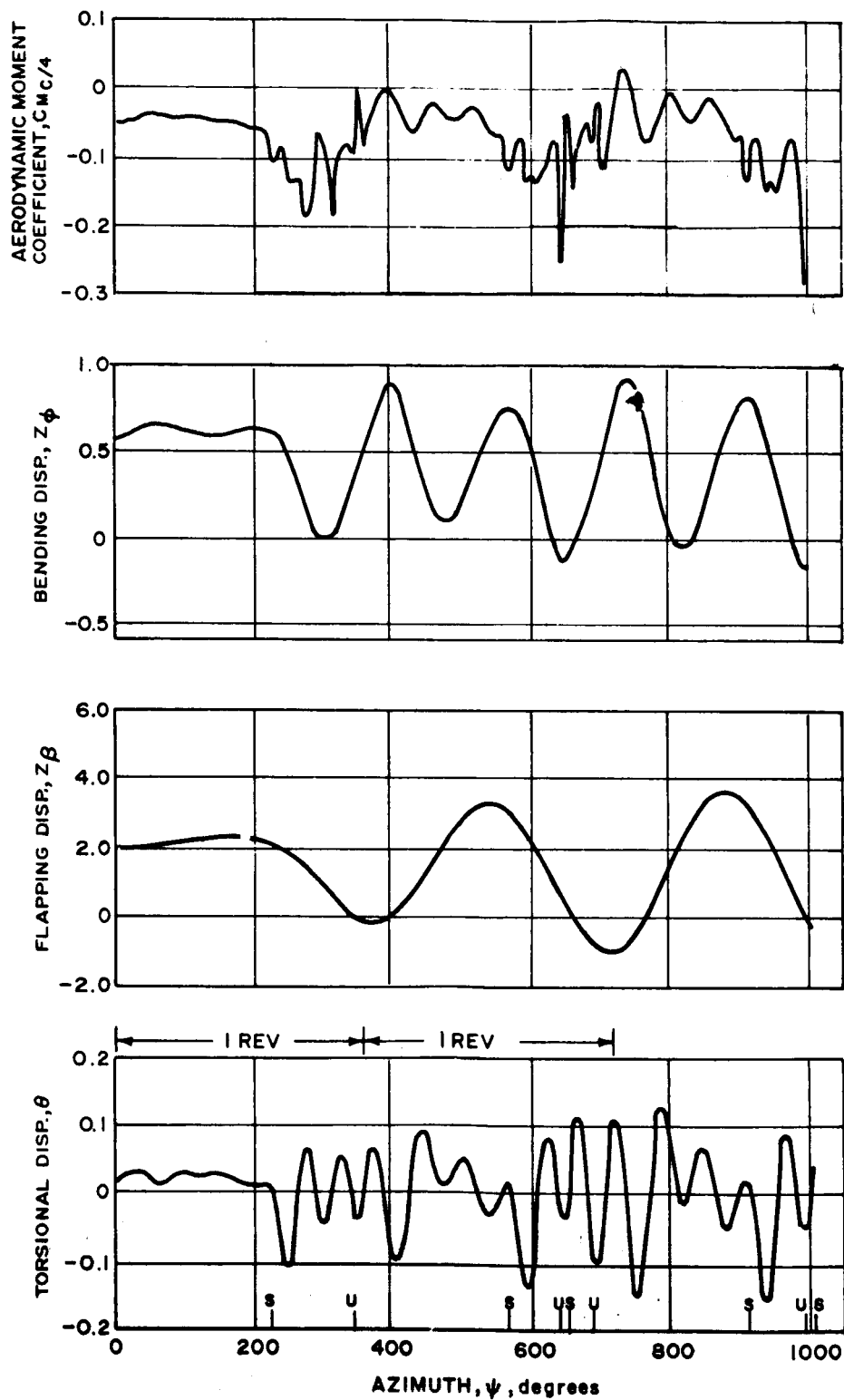


Figure 12 DISPLACEMENT AND MOMENT TIME HISTORIES FOR EXCESSIVE TORSIONAL RESPONSE
 $\Omega^* = 3.89, \theta_0 = 12 \text{ deg}, \mu = 0.1$

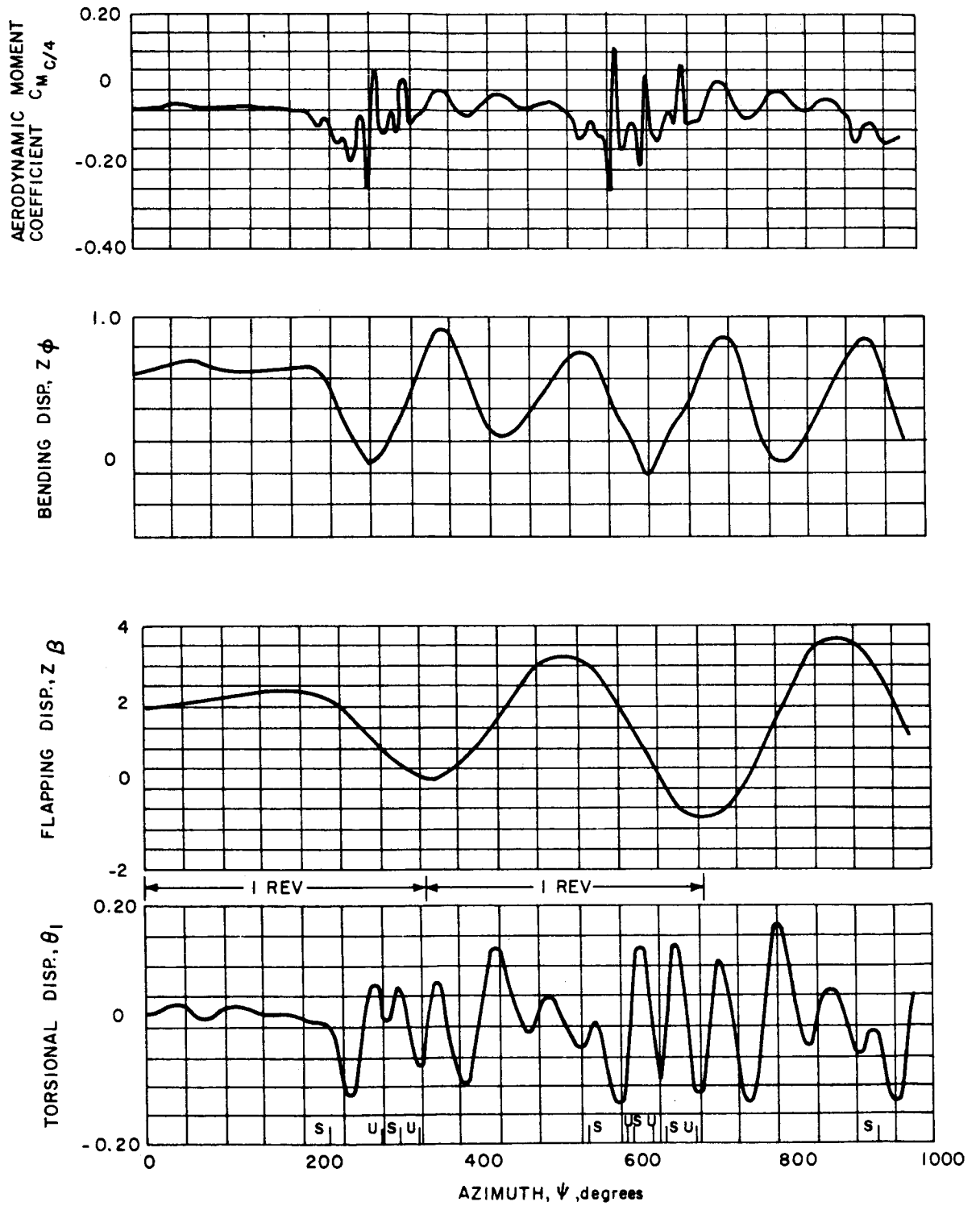


Figure 13 DISPLACEMENT AND MOMENT TIME HISTORIES FOR EXCESSIVE TORSIONAL RESPONSE
 $\Omega^* = 4.15, \theta_0 = 12 \text{ deg}, \mu = 0.1$

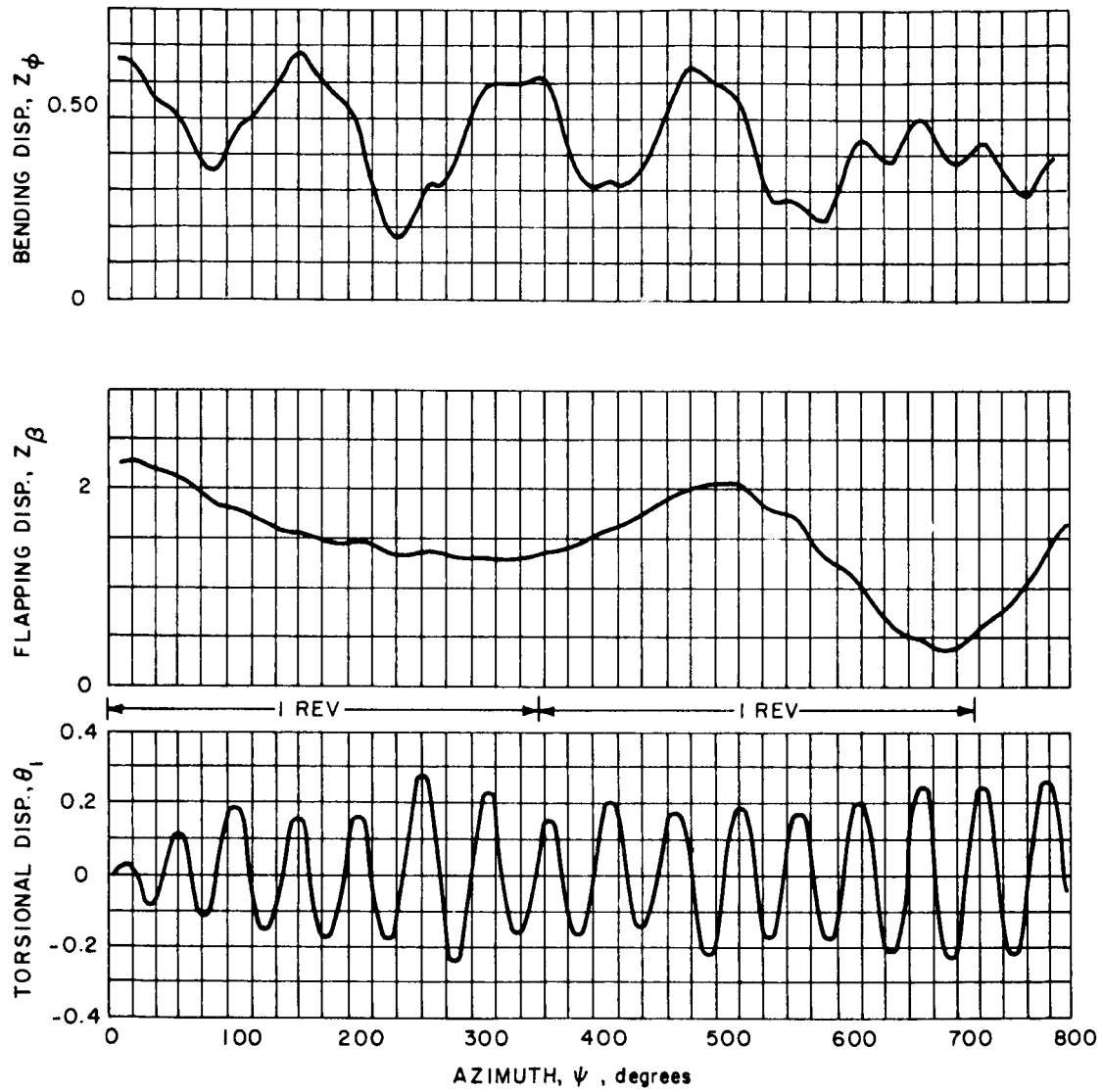


Figure 14 DISPLACEMENT TIME HISTORIES FOR STALL FLUTTER AT LOW ROTOR SPEED
 $\Omega^* = 3.89, \theta_0 = 14.3 \text{ deg}, \mu = 0.1$

speed as those of Figure 12, but with θ_0 increased from 12 deg to 14.3 deg. Successive stall and unstall persists over the whole revolution of the blade for this case.

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

Linear Stability Boundaries

The value of Ω^* 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 Ω^* is plotted against μ 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 Ω^* at flutter onset is plotted against μ 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.

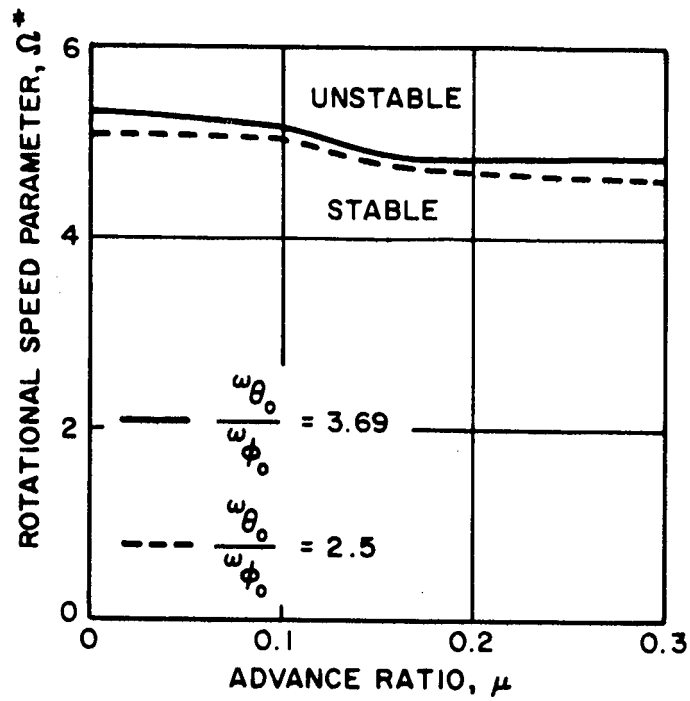


Figure 15 EFFECT OF ADVANCE RATIO AND TORSION-BONDING FREQUENCY RATIO ON LINEAR STABILITY - $X_m/b = 0.216$

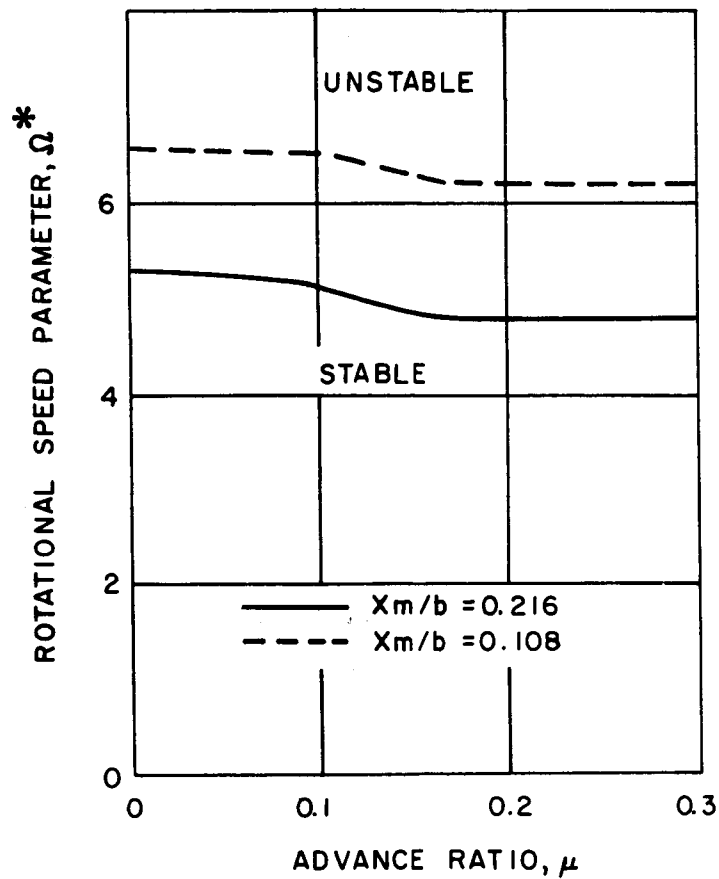


Figure 16 EFFECT OF X_m ON LINEAR STABILITY -

$$\omega_{\theta_0} / \omega_{\phi_0} = 3.69$$

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 Ω^* equal to 3.89. In order to relate the results to rotor performance, a mean lift coefficient \bar{C}_L is defined, according to

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

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

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

The slope a and zero-lift collective pitch angle of $-.0217$ rad were obtained from calculations of \bar{C}_L for the nominal configuration with stall precluded. The variation of a with μ 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 μ . As thrust is increased at a given μ , the rotor is seen to first encounter a region of excessive response, of the type discussed previously, and then, for μ 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

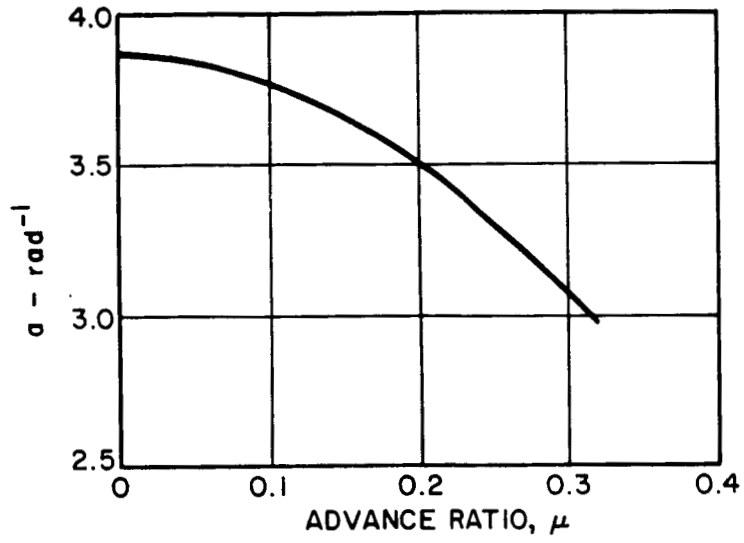


Figure 17 VARIATION OF $a = d\bar{C}_L/d\theta_0$ WITH ADVANCE RATIO

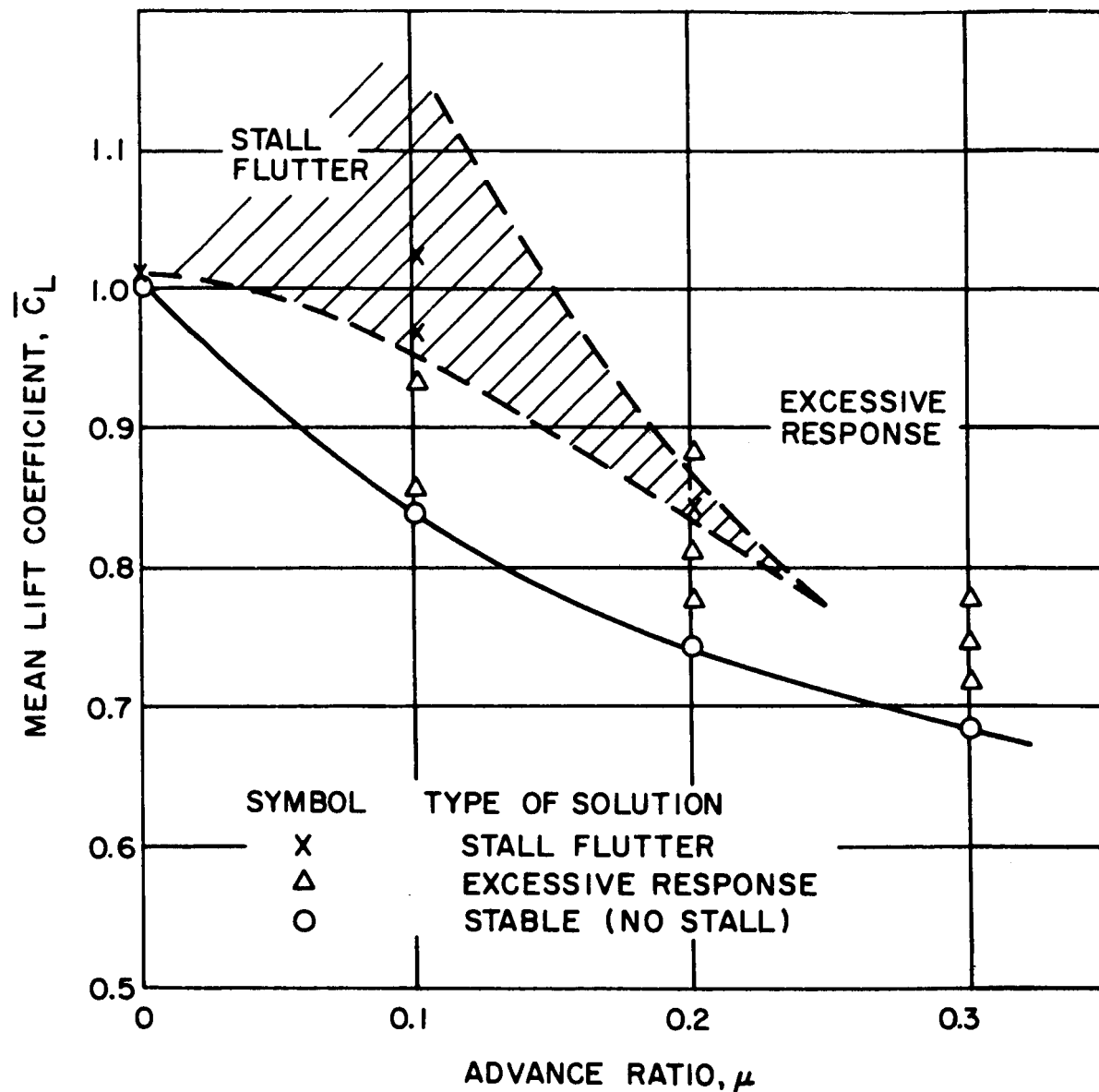


Figure 18 STALL STABILITY BOUNDARIES FOR $\Omega^* = 3.89$, $\omega_{\theta_0}/\omega_{\theta_0} = 3.69$
AND $Xm/b = 0.216$

$\bar{C}_L = .78$, plotted in Figure 19. On the first revolution, as the blade enters the stall zone on the retreating side, it appears that a limit cycle is being set up, with repeated stall and unstall occurring. However, at about $\psi = 420$ deg, the flapping motion has built up in response to the large cyclic pitch changes, producing a negative plunging rate sufficient to keep the blade unstalled over the remainder of its passage on the advancing side. Then, when the blade again enters the stall zone, the large positive flap-induced plunging rate precludes unstall until exit from the stall zone at about $\psi = 670$ deg. As a result, the blade subsequently undergoes excessive torsional response, rather than stall flutter.

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

The marked deterioration in stability at the lower frequency ratio is apparently associated with the lessened linear stability of the system. The configuration with $x_m/b = .108$, which is more stable, in the linear sense, than the nominal one, exhibited a trend opposite to the one resulting from a decrease in frequency ratio. The results for the smaller mass center offset, shown in Figure 21, are similar to those of the nominal configuration, Figure 18, but the region in which stall flutter occurs is somewhat reduced, there being no occurrence of stall flutter at an advance ratio of .2. Also, the amplitude of the torsional oscillations in the region of excessive response is considerably reduced, as evidenced by comparing the blade motions plotted in Figure 22, which are for $\mu = .1$, $\bar{C}_L = .95$ and $x_m/b = .108$, with those of the nominal configuration plotted in Figure 12.

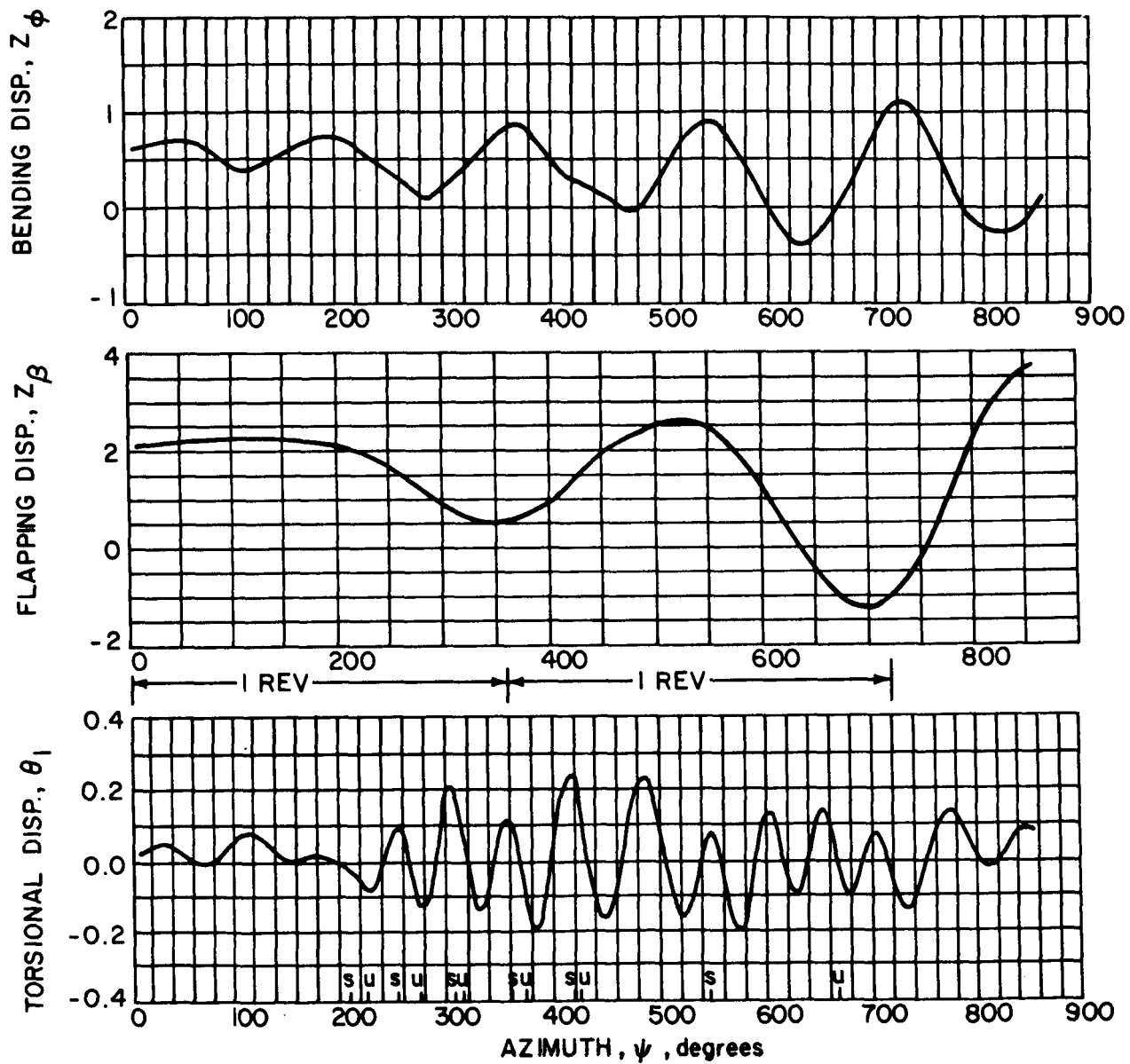


Figure 19 DISPLACEMENT TIME HISTORIES AT HIGH ADVANCE RATIO -
 $\Omega^* = 3.89, \bar{C}_L = 0.78, \mu = 0.3$

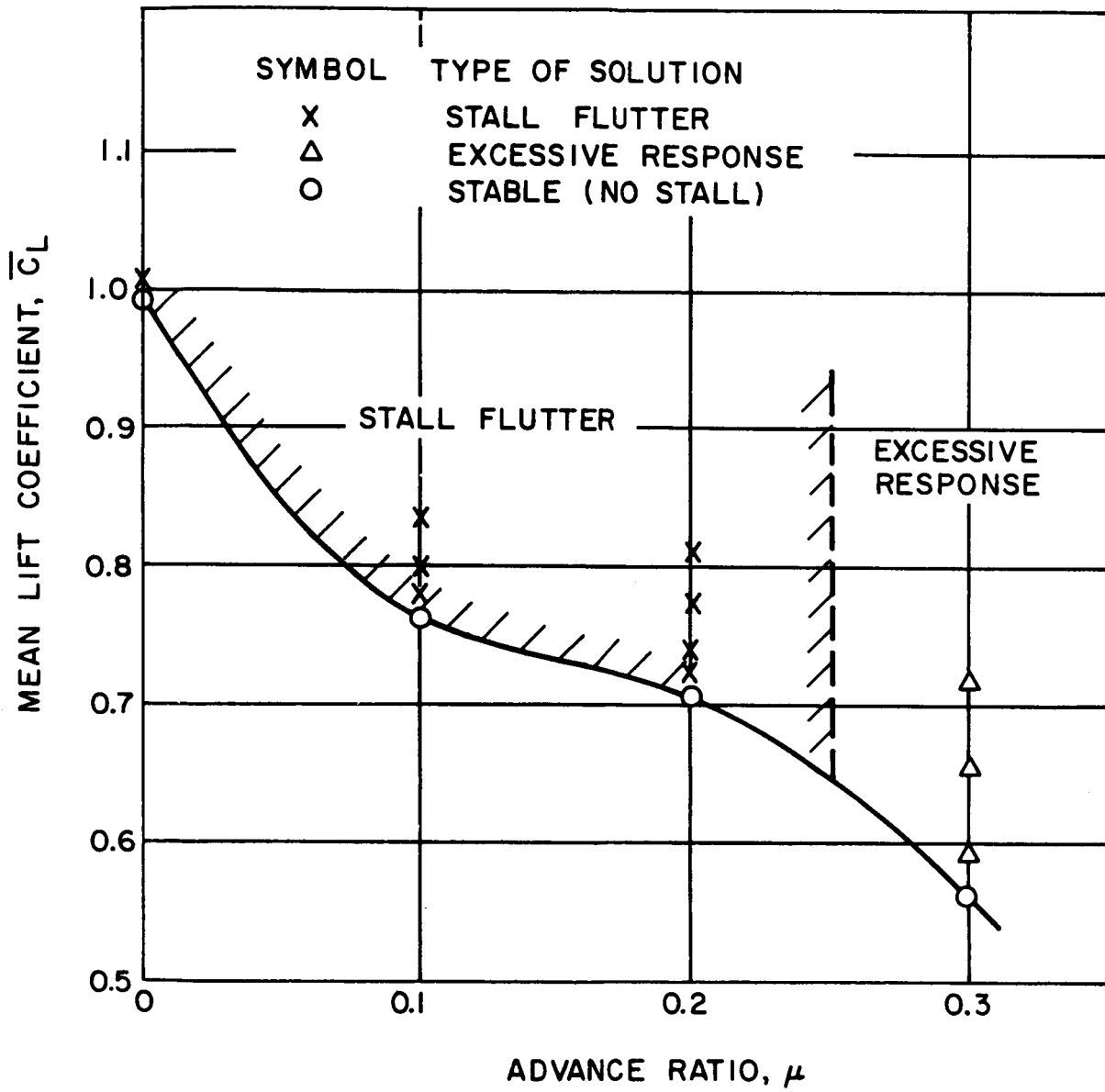


Figure 20 STALL STABILITY BOUNDARIES FOR $\Omega^* = 3.89$, $\omega_{\theta_0}/\omega_{\phi_0} = 2.5$ AND $X_m/b = 0.216$

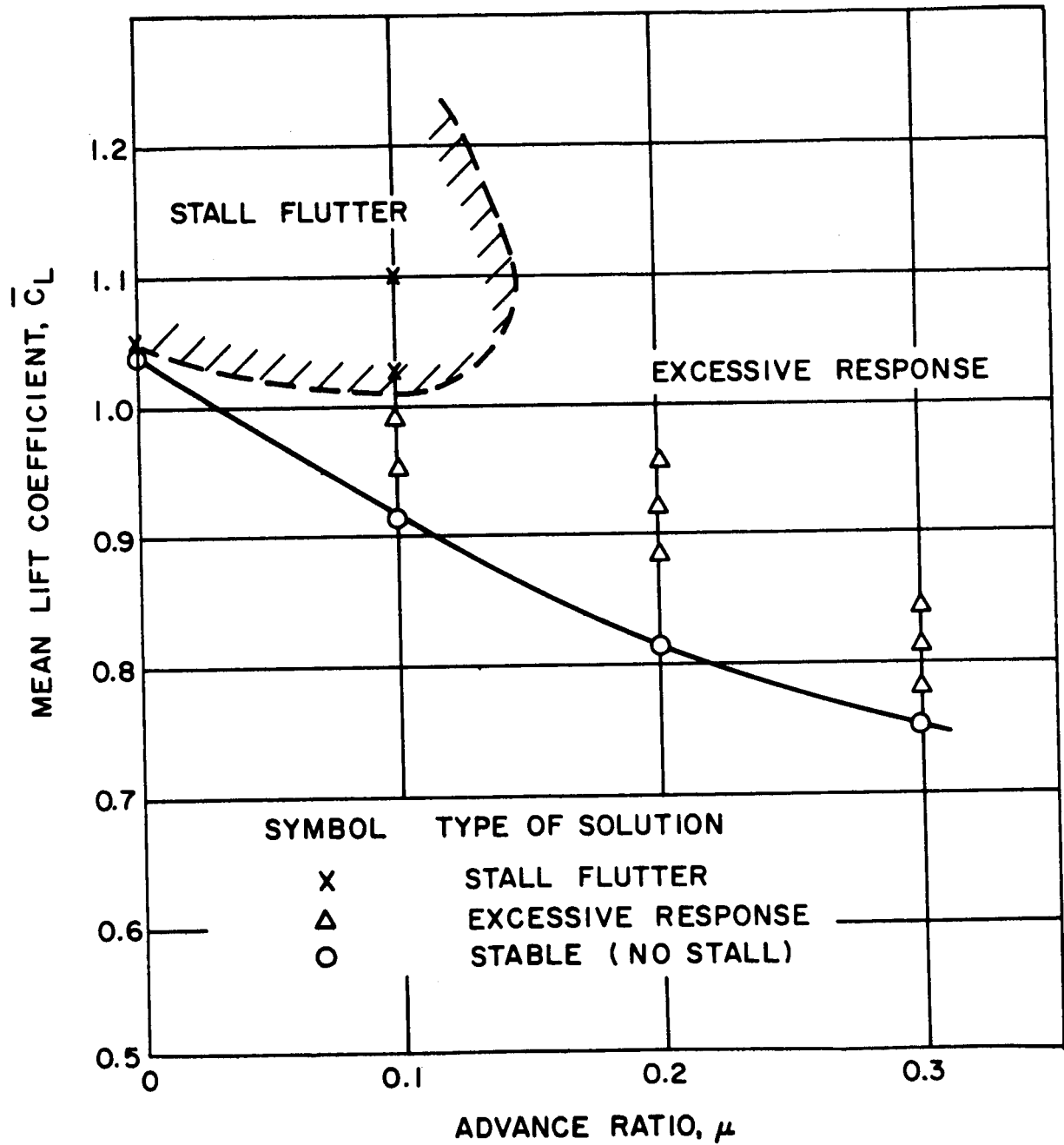


Figure 21 STALL STABILITY BOUNDARIES FOR $\Omega^* = 3.89, \omega_{\theta_0}/\omega_{\phi_0} = 3.69$ AND $X_m/b = 0.108$

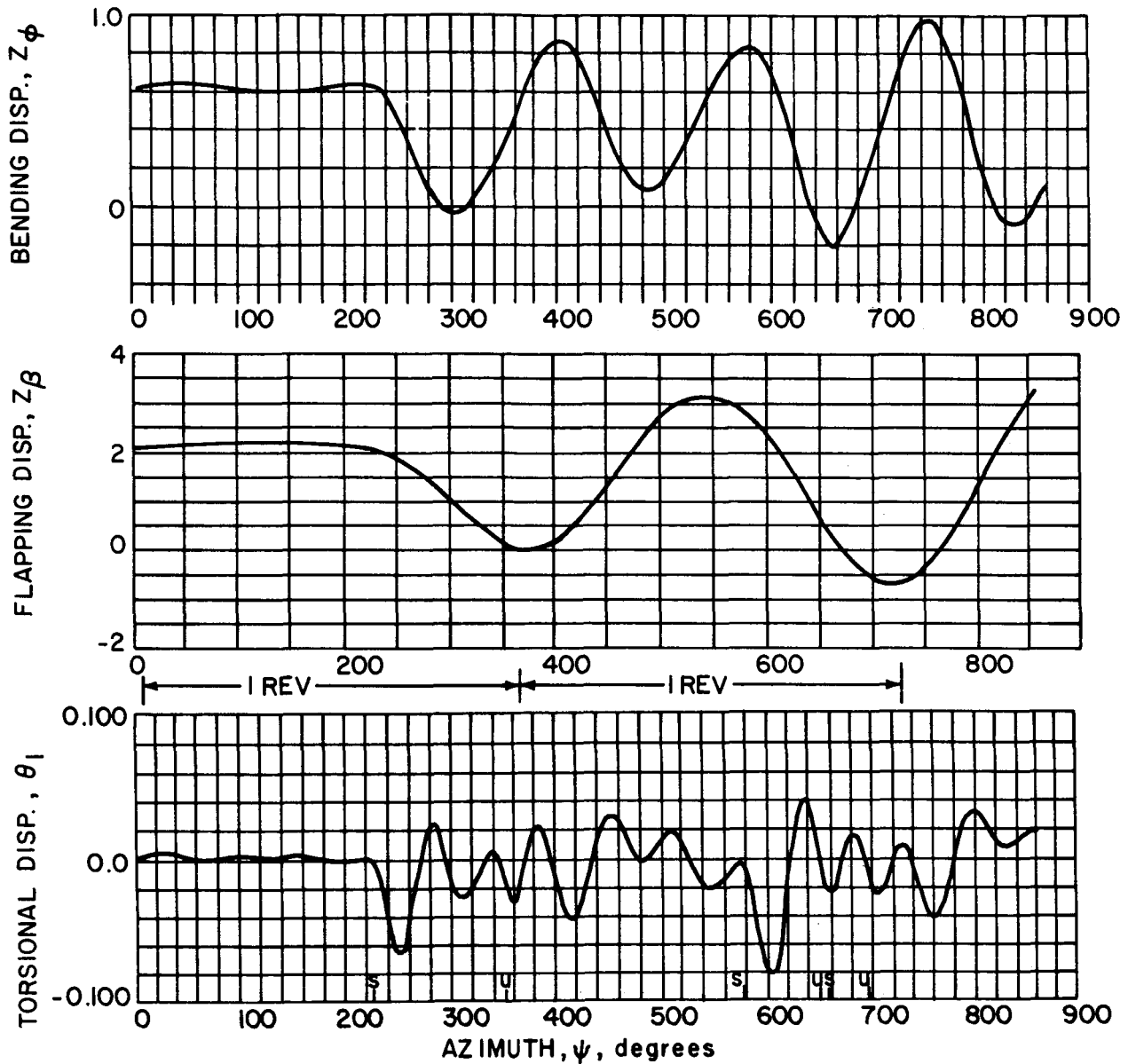


Figure 22 DISPLACEMENT TIME HISTORIES FOR EXCESSIVE TORSIONAL RESPONSE.
 $\Omega^* = 3.89$, $\bar{C}_L = 0.95$, $\mu = 0.1$, AND $X_m/b = 0.108$

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.


```

C
C PROGRAM TO ANALYZE UNSTEADY AIRFOIL STALL MAIN 2
C MAIN 3
C MAIN 4
COMMON /BL1/ NTIME, NDIMC, ISTD
COMMON /CLCMBL / CLVB, CMVB, CMPAVB
C
COMMON /INPTVB/ FTVB(64), FPVB(64), FPPRVB(64), DIORVB(64),
A XMVB(64), DELVB, XMUVB, FOVB, XMUAVB,
B ATOVB, ATCVB, ATSVB, ROVB, RVB(64),
C MVB(64), NVR
C
COMMON /INPUTS/ NSBL, NZ, NOFF, NGAM, NSIG,
A NCOT, NCORD, LOWER, MSTOP, MAXT, MCTR,
B NOTBL, INDV, ELSIG, DXI, REB, RDBB,
C FRZ, ARR, AMPLU, FREQU, ALPH1, ALPH2,
D HEAVE, AROT, FREQF, PHIH, NY, RY1,
E DRY, Y(100), TEST, UPRIM, XU(30), YU(30),
F XL(30), YL(30), ER1, ER2, ER3, BDBR,
G RRDBR
H, CMPA, CMPAS, BARG, EMI, HVOR, NVOR, SSPA, SVOR, TORF, X1VOR
I, PLOTOP, PSILOW, PSTUP
J, NOUT
COMMON / ZZZ/ Z(3)
C
C SETUPS17
C SETUPS18
C SETUPS19
C SETUPS20
C SETUPS21
C SETUPS22
C SETUPS23
C SETUPS24
C SETUPS25
C SETUPS26
C SETUPS27
C SETUPS28
C SETUPS29
C SETUPS30
C
C SETUPS31
C
C DIMENSION USAV(300,100),SCALS(300) MAIN 5
C DIMENSION USAV(1,1),SCALS(300) MAIN 5
C DIMENSION CAMBR(24),THICK(24) MAIN 6
C DIMENSION XGAM(30),XSIG(100),XSIGA(100),XSIGB(100),XC(300),X(300),
C 1SBL(300),XBSIG(100) MAIN 7
C DIMENSION ACAP(30,3),BCAP(100,3),ASZ(30),AS(30,30),BS(30,30),ASHZ
C 1(100),ASH(30,30),BSHF(30,30),AR(30),ARH(100),UE(300,3) MAIN 8
C DIMENSION ALAM(30),VZIP(30),FPRES(100),GAMAW(1000),XTW(1000) MAIN 9
C DIMENSION BLAM(30),FLAM(10),XFLAM(10) MAIN 10
C DIMENSION SCALE(300,2),U(1,1,1),UC(100,3),V(100,2) MAIN 11
C 1, P(200,7) MAIN 12
C
C
C DOUBLE PRECISION CMAT(60,60),RMAT(130) MAIN 13
C
C
C DATA IN, MOUT, NF/ 5,6, 24/
C DATA PI, TIME, UINF, RENEL, USTOP/3.14159, 0., 1., 4.75E4, 2.87/ MAIN 14
C DATA FLAM /1.75, 1.75, 1.724, 1.527, 1.354, 1., .663, .452, .25/ MAIN 15
C 14., 21/ MAIN 16
C DATA XFLAM /-100., -11.26, -7.01, -3.48, -1.766, 0., 1.888, 4./ MAIN 17
C 103, 6.77, 7.19/ MAIN 18
C DATA DEGRES /1.74 53292 51994 3300-2/ SUPPL 19
C
C
C EQUIVALENC (CMAT(1),USAV(1)),(ASH(1),SCALS(1)) MAIN 20
C
C
C IF ISTD =1 TIME DERIVATIVES NOT USED

```

```

      ISTD= 1
      RAD = 180. /PI
      IL= 8888
      NDIMC= 60
      CALL SETUPS
      IF (ISTD .EQ. 1) GO TO 40
      DO 100 J = 1,300
      SCALS(J) =0.
      DO 100 I = 1,100
100   USAV (J,I) =0
40    : CONTINUE
C
      CALL READIN ( IL,& 60)
C
C   NOTE - OFFSETS ARE PUT IN AS LISTED IN THEORY OF WING SECTIONS,
C   AS A FRACTION OF TOTAL CHORD, X1 BEING MEASURED FROM THE
C   LEADING EDGE. BE SURE NF IS AN EVEN NUMBER.
C
      TIME=0.
      NTIME=0
      NWAKE= 999
      ISEP=0
      ISEPT =0
      IWASH =2
      UINF =1.
      L=0
      INDV=INDV+1
      WRITE(MOUT,6)
      PITCH = ALPH1
      IF(INDV + MOTR .LE. 2) PITCH = PITCH - ALPH2
      IF(INDV .EQ. 2)
      X   AMPLU = 1.33333* XMJAVB * (1.-ROVB**3) / (1. - ROVB**4)
      IF( INDV.EQ. 2) FREQU= BDBR/RDDBR
      IF(INDV .GE. 2) GO TO 343
      WRITE(MOUT,25) NVOR,SVOR,HVOR,BARG,XIVOR,EMI,TORF,SSPA
      RY=RY1
      HVOR=HVOR**2
      BARG=BARG/6.2832
343  CALL SECT(XU,YU,XL,YL,NOFF,NF,RDBB,TMDBB,CMDBB,THICK,CAMBR)
      DO 7875 N=1,NF
      CAMBR(N)=CAMBR(N)*CMDBB
7875 THICK(N)=THICK(N)*TMDBB
      WRITE(MOUT,4)
      WRITE(MOUT,7) AMPLU,FREQU,ALPH1,ALPH2,HEAVE,AROT,FREQF,RDBB,REB
      WRITE(MOUT,8)
      WRITE(MOUT,9) (N,CAMBR(N),THICK(N),N=1,NF)
      MX=NSBL+NZ-1
      CALL SCAL(SBL,NSBL,FRZ,ARR,RDBB)
      CALL CORDX(NSBL,NZ,RDBB,SBL,X,XC)
      DO 2420 M=1,MX
      IF(XC(M)-1.) 2420,2419,2419
2419 MEND=M-1
      GO TO 2421
2420 CONTINUE
2421 MX=MEND

```

MAIN 4

MAIN 65
I.E. MAIN 66
MAIN 67
MAIN 68
MAIN 69

MAIN 59
MAIN 72

MAIN 75
MAIN 64
MAIN 76
MAIN 77
MAIN 78
MAIN 79
MAIN 80
MAIN 81
MAIN 82
MAIN 83
MAIN 84
MAIN 85
MAIN 86
MAIN 87
MAIN 88
MAIN 89
MAIN 90
MAIN 91
MAIN 92
MAIN 93
MAIN 94

MXM1=MX-1	MAIN	95
UE(MX+1,1)=1.	MAIN	96
EPSLF=2.*(X(NZ)-X(NZ-1))	MAIN	97
FPSTF=X(MX)-X(MX-1)	MAIN	98
ALTC=8.36F4/SQRT(REB)	MAIN	99
IF(ISTD.EQ. 1) GO TO 50		
DO 2422 M=1,MX	MAIN	100
SCALE(M,1)=0.	MAIN	101
SCALE(M,2)=0.	MAIN	102
DO 2422 N=1,NY	MAIN	103
U(M,N,1)=0.	MAIN	104
2422 U(M,N,2)=0.	MAIN	105
5C CONTINUE		
NSIGA=NSIG	MAIN	106
NSIGB=NSIG	MAIN	107
NSIG1=NSIG+1	MAIN	108
MOTR=MOTR+1	MAIN	109
NOTBL=NOTBL+1	MAIN	110
XMAX=1.-ELSIG	MAIN	111
CCNA=.375*PI/DXI	MAIN	112
ANGS=PI/FLOAT(NSIG)	MAIN	113
CALL SETSX(NSIG1,1,1,2.,XSIG,ANGS)	MAIN	114
XSEP=1.1	MAIN	115
DO 2430 N=1,NSIG1	MAIN	116
XSIGB(N)=XSIG(N)	MAIN	117
2430 XSIGA(N)=XSIG(N)	MAIN	118
DO 2431 N=1,NSIG	MAIN	119
DO 2431 NU=1,3	MAIN	120
2431 BCAP(N,NU)=0.	MAIN	121
PINT=2./FLOAT(NCORD)	MAIN	122
NCPI=NCORD+1	MAIN	123
THXI=1.5/DXI	MAIN	124
NGPI=NGAM+1	MAIN	125
NWMI=NWAKE-1	MAIN	126
COUNT=0.	MAIN	127
DO 8456 N=1,NWAKE	MAIN	128
GAMAW(N)=0.	MAIN	129
XIW(N)=1.+COUNT	MAIN	130
8456 COUNT=COUNT+DXI	MAIN	131
ANGLE=PI/FLOAT(NGAM)	MAIN	132
COUNT=0.	MAIN	133
DO 1002 M=1,NGPI	MAIN	134
PHIM=COUNT*ANGLE	MAIN	135
XGAM(M)=COS(PHIM)	MAIN	136
DOUNT=2.	MAIN	137
DO 1001 N=2,NGAM	MAIN	138
AS(M,N)=COS(DOUNT*PHIM)	MAIN	139
1001 DOUNT=DOUNT+1.	MAIN	140
1002 COUNT=COUNT+1.	MAIN	141
CALL WASH(XGAM,NGAM,TIME,ALPH1,ALPH2,HEAVE,AROT,FREQF,PHIH,UINF,CAMAIN	MAIN	142
IMBR,NF,VZIP,1,1)	MAIN	143
DO 8458 M=1,NGPI	MAIN	146
CPAT(M,1)=1.	MAIN	147
TEMP=2.*VZIP(M)	MAIN	148
RPAT(M)=TEMP	MAIN	149

	CMAT(M,2)=XGAM(M)	MAIN 150
	DO 8457 N=3,NGP1	MAIN 151
8457	CMAT(M,N)=AS(M,N-1)	MAIN 152
8458	CONTINUE	MAIN 153
	CALL ALSOL(NGP1,CMAT,RMAT)	MAIN 154
	DO 8459 N=1,NGP1	MAIN 155
	ACAP(N,1)=RMAT(N)	MAIN 156
	ACAP(N,3)=RMAT(N)	MAIN 156
8459	ACAP(N,2)=ACAP(N,1)	MAIN 157
	DO 2784 M=1,MX	MAIN 158
	SIGN=1.	MAIN 159
	IF(M-NZ) 2774,2775,2775	MAIN 160
2774	SIGN=-SIGN	MAIN 161
2775	CALL QECAL(ISEP,NGAM,NSIG,NF,XSIG,ACAP,BCAP,THICK,RCBB,GAMAW(1),U	MAIN 162
	INF,XC(M),UF(M,1),SIGN)	MAIN 163
2784	UF(M,2)=UE(M,1)	MAIN 164
	DO 1004 M=2,NGAM	MAIN 165
1004	BLAM(M)=(1.125*XGAM(M)+.1875*(1.+XGAM(M))*(1.-3.*XGAM(M))*ALOG((1.	MAIN 166
	1+XGAM(M))/((1.-XGAM(M))))/DXI	MAIN 167
	BLAM(NGP1)=-1.125/DXI	MAIN 168
	CALL CLCM(NCOT,ISEP,NGAM,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,RC	MAIN 504
	IAP,THICK,RDDB,GAMAW,UINF,UDOT,DXI,AROT,CMPA)	MAIN 505
	IF (INDV.EQ. 2)	
	ICALL SUPPL	MAIN
C		MAIN 169
C	INDEXING IN TIME IS CARRIED OUT AT THIS POINT.	MAIN 170
C		MAIN 171
	9999 CONTINUE	MAIN 172
	CALL ACUCPU(IACU)	
	IF(IACU.LT. 35000) GO TO 99	
C		MAIN 175
C	NOTE - FOR READ-IN CF FCIL MOTIONS, MAKE ALPH1 = ALPHA,	MAIN 176
C	ALPH2 = ALPHA-DOT, AND HEAVE = H-DOT.	MAIN 177
C		MAIN 178
	IF(MCTR.EQ. 2)	
	XREAD(IN,2,END=8989) ALPH1,ALPH2,HEAVE	MAIN 174
158	NITS=1	MAIN 182
	TIME=TIME+DXI	MAIN 183
	NTIME=NTIME+1	MAIN 184
	NWAKE=NTIME+2	MAIN 185
	IF(NWAKE-998) 202,201,201	MAIN 186
201	NWAKE=998	MAIN 187
202	IF(MAXT-NTIME) 8989,8800,8800	MAIN 188
8800	SAVFU=UINF	MAIN 189
	L=L+1	
	P(L,1) = BCBB / RRDBR * TIME * RAD	
	PSI360= AMOD(P(L,1) , 360.)	
	UINF=1.+AMPLU*SIN(FREQJ*TIME)	MAIN 190
	IF(INDV.EQ. 2)	
	XCALL SUPP1(UINF)	MAIN
	PITCH = ALPH1	
	IF(INDV + MOTR .LE. 2) PITCH = PITCH - ALPH2*COS(FREQF*TIME)	MAIN 475
	UDOT=FREQU*AMPLU*COS(FREQU*TIME)	MAIN 191
	STEPX=.5*DXI*(UINF+SAVEU)	MAIN 192
	DO 1003 J=2,NWAKE	MAIN 193

JC=NWAKE-J+2	MAIN 194
GAMAW(JC)=GAMAW(JC-1)	MAIN 195
1003 XIW(JC)=XIW(JC-1)+STFPX	MAIN 196
IF(ISEP) 2009,2009,2007	MAIN 197
2007 DO 2008 N=1,NSIG	MAIN 198
BCAP(N,3)=BCAP(N,2)	MAIN 199
2008 BCAP(N,2)=BCAP(N,1)	MAIN 200
DO 4433 N=1,NSIG1	MAIN 201
XSIGR(N)=XSIGA(N)	MAIN 202
4433 XSIGA(N)=XSIG(N)	MAIN 203
GO TO 2010	MAIN 204
2009 DEADL=0.	MAIN 205
ELDOT=UINF	MAIN 206
2010 DO 1014 M=1,MX	MAIN 207
UE(M,3)=UE(M,2)	MAIN 208
1014 UE(M,2)=UF(M,1)	MAIN 209
DEAD1=DEADL	MAIN 210
ELD1=ELDOT	MAIN 211
ALAM(1)=(1.125+.75*ALOG(STEPX*.5))/DXI	MAIN 212
DO 1005 M=2,NGPI	MAIN 213
1005 ALAM(M)=BLAM(M)+.75*(1.+(1.-XGAM(M))/STEPX)*ALOG((1.+STEPX-XGAM(M)	MAIN 214
1)/(1.-XGAM(M))/DXI	MAIN 215
DO 2006 M=1,NGPI	MAIN 216
ACAP(M,3)=ACAP(M,2)	MAIN 217
2006 ACAP(M,2)=ACAP(M,1)	MAIN 218
AFACT=8.*(ACAP(1,2)+.5*ACAP(2,2))-2.*(ACAP(1,3)+.5*ACAP(2,3))	MAIN 219
ALPHS=VZIP(1)	MAIN 220
CALL WASH(XGAM,NGAM,TIME,ALPH1,ALPH2,HEAVE,AROT,FREQF,PHIH,UINF,CAMAIN	MAIN 221
IMBR,NF,VZIP,MOTR,INDV)	MAIN 222
DO 1006 M=1,NGPI	MAIN 225
ASZ(M)=1.+2.*ALAM(M)	MAIN 226
AS(M,1)=XGAM(M)+ALAM(M)	MAIN 227
SUM=0.	MAIN 228
DO 4343 J=2,NWMI	MAIN 229
4343 SUM=SUM+(GAMAW(J)+(GAMAW(J+1)-GAMAW(J))*(XGAM(M)-XIW(J))/(XIW(J+1)	MAIN 230
1-XIW(J))*ALOG((XIW(J+1)-XGAM(M))/(XIW(J)-XGAM(M)))	MAIN 231
ELX=1.-XGAM(M)	MAIN 232
IF(M-1) 1006,2130,1006	MAIN 233
2130 ELX=1.	MAIN 234
1006 AR(M)=2.*VZIP(M)+ALAM(M)*AFACT/3.+(SUM-GAMAW(2))*(1.-XGAM(M))*ALOG(MAIN 235
1/(1.+STEPX-XGAM(M))/ELX)/STEPX)/PI	MAIN 236
C	MAIN 237
C THE FOLLOWING CALCULATIONS, THROUGH STATEMENT 4444, ARE PERFORMED	MAIN 238
C ONLY IF THE AIRFOIL IS STALLED. THE AIRFOIL IS DESIGNATED TO BE	MAIN 239
C STALLED IF INTEGER ISEP IS NONZERO.	MAIN 240
C	MAIN 241
IF(ISEP) 3247,4444,3247	MAIN 242
3247 GO TO (3344,3345),IWASH	MAIN 243
3344 XSEP=XSEP+DXI	MAIN 244
IF(XSEP-XMAX) 3248,3347,3347	MAIN 245
3347 IWASH=2	MAIN 246
ISEP=0	MAIN 247
XSEP=1.1	MAIN 248
DO 3015 K=1,3	MAIN 249
DO 3015 N=1,NSIG	MAIN 250

3015	BCAP(N,K)=0.	MAIN	251
	GO TO 4444	MAIN	252
3345	IF(INDT) 3348,3348,3248	MAIN	253
3348	IF(NITS-1) 3248,3349,3248	MAIN	254
3349	IF(INDV.EQ.2) GO TO 6349	MAIN	255
	IF(VZIP(1)-ALPHS) 6349,6348,6348	MAIN	256
6348	NITS=2	MAIN	257
	GO TO 3248	MAIN	258
6349	CALL UNPOP(NGAM,AR,ALAM,AFAC,RMAT,CMAT,XGAM,AS,ACAP,MX,NZ,IF,XSIGMAIN	MAIN	259
	1,BCAP,THICK,RDBB,UINF,XC,UE)	MAIN	260
	GO TO 2785	MAIN	261
3248	XATT=XSEP+DEAD1+.5*(ELD1+ELDOT)*DXI	MAIN	262
	DEADL=XATT-XSEP	MAIN	263
	DIFF=1.-XATT	MAIN	264
	XTEST = XSEP + 3. * EPSLE		
	CALL SETSX(NSIG1,XSEP,XATT,XSIG,ANGS)	MAIN	265
	DO 4434 N=1,NSIG	MAIN	266
4434	XBSIG(N)=.5*(XSIG(N)+XSIG(N+1))	MAIN	267
	DO 3086 M=1,NGPI	MAIN	268
	DO 3086 N=1,NSIG	MAIN	269
3086	BS(M,N)=0.	MAIN	270
	DO 3087 M=1,NGPI	MAIN	271
	IF(XGAM(M)-XSEP) 3088,3088,3089	MAIN	272
3089	IF(XATT-XGAM(M)) 3187,3087,3091	MAIN	273
3091	DO 3092 I=1,NSIG1	MAIN	274
	IF(XGAM(M)-XSIG(I)) 3093,3092,3092	MAIN	275
3093	MARK=I	MAIN	276
	GO TO 3094	MAIN	277
3092	CONTINUE	MAIN	278
3094	WIDES=XSIG(MARK)-XSIG(MARK-1)	MAIN	279
	BS(M,MARK-1)=(XSIG(MARK)-XGAM(M))/WIDES	MAIN	280
	BS(M,MARK)=(XGAM(M)-XSIG(MARK-1))/WIDES	MAIN	281
	BS(M,I)=SQRT((XGAM(M)-XSEP)/(XATT-XGAM(M)))	MAIN	282
3088	IF(DIFF-1.E-6) 3087,3098,3098	MAIN	283
3098	BS(M,I)=BS(M,I)+DIFF**(-1.5)*SQRT(DEADL)*(2.*DIFF+(SQRT((1.-XGAM(M)	MAIN	284
	I))/(XATT-XGAM(M))-1.)*(4.*XGAM(M)-1.-3.*XATT))	MAIN	285
	GO TO 3087	MAIN	286
3187	BS(M,I)=DIFF**(-1.5)*SQRT(DEADL)*(3.+ XATT-4.*XGAM(M))	MAIN	287
3087	CONTINUE	MAIN	288
C		MAIN	289
C	SET-UP OF THE SECOND SET OF EQUATIONS STARTS HERE.	MAIN	290
C		MAIN	291
	DO 4350 K=1,NSIG	MAIN	292
	IF(XBSIG(K)-1.) 4348,4349,4349	MAIN	293
4348	COSK=XBSIG(K)	MAIN	294
	SINK=SQRT(1.-COSK*COSK)	MAIN	295
	THETK=ARCT(COSK)	MAIN	296
	TANT=SINI.5*THETK/COS(.5*THETK)	MAIN	297
	ASHZ(K)=TANT+CINA*(1.+COSK)*(1.-3.*COSK)/UINF+THX I*(PI-THETK+SINK+	MAIN	298
	ICINA*(1.+COSK)*SINK**2)/UINF	MAIN	299
	ASH(K,I)=.5*(ASHZ(K)-TANT)+SINK	MAIN	300
	COUNT=1.	MAIN	301
	DO 4355 N=2,NGAM	MAIN	302
	COUNT=COUNT+1.	MAIN	303
4355	ASH(K,N)=SINI(COUNT*THETK)+.75*(SINI(COUNT+1.)*THETK)/(COUNT+1.)-SI	MAIN	304

	IN((COUNT-1.)*THETK)/(COUNT-1.)/(DXI*UINF)	MAIN 305
	GO TO 4350	MAIN 306
4349	ASHZ(K)=0.	MAIN 307
	DO 4359 N=1,NGAM	MAIN 308
4359	ASH(K,N)=0.	MAIN 309
4350	CONTINUE	MAIN 310
	IF(DIFF-1.E-6) 5005,5006,5006	MAIN 311
5005	PREC=0.	MAIN 312
	GO TO 5007	MAIN 313
5006	CALL ATTPR(PRFC, XSIG, NSIG, ASZ, AS, AR, CMAT, RMAT, NGAM, NF, ACAP, THICK, RMAIN IDBB, GAMAW, UINF, UDOT, DXI, BCAP)	MAIN 314 MAIN 315
5007	CALL MIXER(FPRES, PREC, UINF, UDOT, THICK, NF, XBSIG, NSIG, INDT, DEL1, THETMAIN 11, REB, USEP, X4, CPI)	MAIN 316 MAIN 317
	CPCT=CPI	MAIN 318
	DO 4800 K=1, NSIG	MAIN 319
	CORD=XBSIG(K)	MAIN 320
	BSH(K,1)=-1.+THXI*BINT(XSEP, XATT, CORD)/UINF	MAIN 321
	DO 4808 N=2, NSIG	MAIN 322
4808	BSH(K,N)=FB(XSIG(N-1), XSIG(N), XSIG(N+1), CORD)+FHXI*GB(XSIG(N-1), XSM 1IG(N), XSIG(N+1), CORD)/UINF	MAIN 323 MAIN 324
	CALL ESIGI(2, NSIGA, XSIGA, BCAP, CORD, VAL1)	MAIN 325
	CALL ESIGI(3, NSIGB, XSIGB, BCAP, CORD, VAL2)	MAIN 326
	ARH(K)=FPRES(K)+(2.*VAL1-.5*VAL2)/(DXI*UINF)	MAIN 327
	IF(CORD-1.) 5008,4800,4800	MAIN 328
5008	CALL EGAMI(2, NGAM, ACAP, BCAP(1,2), XSIGA(1), XSIGA(NSIGA+1), GAMAW(2), 1CORD, VAL1)	MAIN 329 MAIN 330
	CALL EGAMI(3, NGAM, ACAP, BCAP(1,3), XSIGB(1), XSIGB(NSIGB+1), GAMAW(3), 1CORD, VAL2)	MAIN 331 MAIN 332
	ARH(K)=ARH(K)+(2.*VAL1-.5*VAL2)/(DXI*UINF)+.0625*AFACT*PI*(1.+CORD 1)*(1.-3.*CORD+THXI*(1.-CORD*CORD))/(DXI*UINF)	MAIN 333 MAIN 334
4800	CONTINUE	MAIN 335
4444	CONTINUE	MAIN 336
C		MAIN 337
C	CALCULATIONS FROM THIS POINT ON COMBINE THE	MAIN 338
C	CASES OF STALLED AND UNSTALLED AIRFOILS.	MAIN 339
C		MAIN 340
	DO 6500 M=1, NGPI	MAIN 341
	RMAT(M)=AR(M)	MAIN 342
	CMAT(M,1)=ASZ(M)	MAIN 343
	DO 6485 N=1, NGAM	MAIN 344
6485	CMAT(M,N+1)=AS(M,N)	MAIN 345
	IF(ISEP) 6486,6500,6486	MAIN 346
6486	DO 6499 N=1, NSIG	MAIN 347
	NGG=N+NGPI	MAIN 348
6499	CMAT(M,NGG)=BS(M,N)	MAIN 349
6500	CONTINUE	MAIN 350
	IF(ISEP) 6502,6501,6502	MAIN 351
6501	NTOT=NGPI	MAIN 352
	GO TO 6751	MAIN 353
6502	DO 6750 K=1, NSIG	MAIN 354
	KK=K+NGPI	MAIN 355
	RMAT(KK)=ARH(K)	MAIN 356
	CMAT(KK,1)=ASHZ(K)	MAIN 357
	DO 6748 N=1, NGAM	MAIN 358
6748	CMAT(KK,N+1)=ASH(K,N)	MAIN 359

DO 6750 N=1,NSIG	MAIN 360
NGG=N+NGPI	MAIN 361
6750 CMAT(KK,NGG)=BSH(K,N)	MAIN 362
NTOT=NSIG+NGPI	MAIN 363
6751 CALL ALSOL(NTOT,CMAT,RMAT)	MAIN 364
DO 6800 N=1,NGPI	MAIN 365
6800 ACAP(N,1)=RMAT(N)	MAIN 366
IF(ISEP) 6805,6820,6805	MAIN 367
6805 DO 6810 N=1,NSIG	MAIN 368
NGG=N+NGPI	MAIN 369
6810 BCAP(N,1)=RMAT(NGG)	MAIN 370
6820 CONTINUE	MAIN 371
GAMAW(1)=GAMI(ACAP,DXI,PI)	MAIN 372
IF(PSI360 .GE. PSILOW .AND. PSI360 .LE. PSIUP) GO TO 1736	
DO 1785 M=1,MX	MAIN 373
SIGN=1.	MAIN 374
IF(M-NZ) 1780,1785,1785	MAIN 375
1780 SIGN=-SIGN	MAIN 376
1785 CALL QECAL(ISEP,NGAM,NSIG,NF,XSIG,ACAP,BCAP,THICK,RDRB,GAMAW(1),UI	MAIN 377
INF,XC(M),UE(M,1),SIGN)	MAIN 378
2785 DO 8886 I=1,2	MAIN 379
US2=UE(I,1)	MAIN 380
DO 8886 M=1,MXMI	MAIN 381
US1=UE(M,1)	MAIN 382
UE(M,1)=(US1+US2+UE(M+1,1))/3.	MAIN 383
8886 US2=US1	MAIN 384
GO TO (8351,8353),IWASH	MAIN 386
8351 DO 8352 M=1,MX	MAIN 387
8352 SCALS(M)=0.	MAIN 388
GO TO 1786	MAIN 389
8353 CALL YSET(RY1,Y(2),NY,Y)	MAIN 390
RY=RY1	MAIN 391
DO 8354 M=1,MX	MAIN 392
8354 SCALS(M)=0.	MAIN 393
IF(INDV.EQ.2) GO TO 8370	MAIN 395
IF(ISEP.EQ.0.AND.VZIP(1).LT.ALPHS) GO TO 1786	MAIN 396
8370 CALL STAG(MX,NY,MSTOP,MST,DXI,RY,DRY,X,Y,UE,UC,V,USAV,SCALS,ISEP)	MAIN 397
LAMQ=1	MAIN 398
XSEPS=XSEP	MAIN 399
DXX=DXI	MAIN 400
IF(ISEP.EQ.1.AND.ISEPT.EQ.0.AND.NITS.EQ.1) DXX=1.E30	MAIN 401
8367 CALL BLC(X,Y,MST,MEND,NY,RY,DRY,DXX,REB,UPRIM,FLAM,XFLAM,TEST,U,SC	MAIN 402
IALE,UE,UC,V,XSEP,USEP,DISP,THETA,LOWER,LAMQ,MSEP,XC,USAV,SCALS,NIT	MAIN 403
MAIN IS,NTIME,NOTBL,XTEST,NZ,NOUT)	
IF(XSEP-XMAX) 7736,7735,7735	MAIN 405
7735 IF(ISEP) 1786,1786,7736	MAIN 406
7736 DELI=DISP	MAIN 407
THET1=THETA	MAIN 408
INDT=I-LAMQ	MAIN 409
IF(INDT.EQ.1.AND.NOTBL.EQ.2) GO TO 1786	MAIN 410
WRITE(MOUT,23) XSIG(I),CPCT,XSEP	MAIN 411
IF(INDT) 8462,8462,8463	MAIN 412
8462 IF(ISEP) 8562,8562,8563	MAIN 413
8563 IF(NITS-1) 8562,8562,8562	MAIN 414
8562 IF(ISEPT) 7742,7742,8562	MAIN 415

8562	CALL RUB8(DEL1,THET1,RF8,XSEP,USEP,XC5,DCP,DEL5,X,XC,MX,NZ,X5,U5,UMAIN	416
	IF,ALTC,RFNEL,USTOP)	MAIN 417
	USEP=USEP+.002046*USEP**3	MAIN 418
	PDIFF=(USEP-U5)*(IUSEP+U5)	MAIN 419
	WRITE(MOUT,22) PDIFF,DCP	MAIN 420
	IF(DCP-PDIFF) 8263,8366,8366	MAIN 421
8263	I SEPT=0	MAIN 422
	GO TO 8463	MAIN 423
8366	IF(I SEPT) 8368,8368,8369	MAIN 424
8369	IF(I SEPT) 8467,8467,8368	MAIN 425
8467	I WASH=1	MAIN 426
	NITS=2	MAIN 427
	GO TO 3344	MAIN 428
8368	GO TO (8168,1786),NOTBL	MAIN 429
8168	CALL RFAAT(UC,V,X,Y,MX,NY,RY,DRY,UE,X5,DEL5,MST,REB)	MAIN 430
	LAMQ=0	MAIN 431
	GO TO 8367	MAIN 432
8463	IF(I SEPT) 7741,7741,7742	MAIN 433
7741	I SEPT=1	MAIN 434
	NITS=NITS+1	MAIN 435
	IF(INDT) 7743,7743,7643	MAIN 436
7643	I SEPT=1	MAIN 437
	DXSEP=1.-XSEP	MAIN 438
	XSEP=.6*XSEP+.4	MAIN 439
	CALL CPC(I SEPT,NGAM,NF,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,BCAP,	MAIN 440
	ITHICK,ROBB,GAMAW,UINF,UDOT,1.,XSEP,DXI,CP1)	MAIN 441
	GO TO 3248	MAIN 442
7742	CALL ELDER(BCAP,XSIG,NSIG,UINF,ELDOT,SIGSUM,YMX)	MAIN 443
	IF(I SEPT.EQ.1.AND.I SEPT.EQ.0.AND.NITS.EQ.1) GO TO 9210	MAIN 444
	IF(XSEP+.5) 7841,7842,7842	MAIN 445
7841	EPS=EPSLE	MAIN 446
	GO TO 7843	MAIN 447
7842	EPS=EPSTE	MAIN 448
7843	DXSEP=ABS(XSEP-XSEPS)	MAIN 449
	IF(DXSEP-EPS) 7834,7834,9210	MAIN 450
7834	IF(XSEP-XMAX) 1786,1786,7835	MAIN 451
7835	I SEPT=0	MAIN 452
	I SEPT=0	MAIN 453
	DO 7836 K=1,3	MAIN 454
	DO 7836 N=1,NSIG	MAIN 455
7836	BCAP(N,K)=0.	MAIN 456
	GO TO 1786	MAIN 457
9210	NITS=NITS+1	MAIN 458
	IF(NITS.EQ.2.AND.INDT.EQ.0) XSEPS=XSEP	MAIN 459
	IF(NITS-4) 9211,9211,1786	MAIN 460
9211	IF(XSEP-XSEPS) 9305,9305,9306	MAIN 461
9305	XSEP=.6*XSEPS+.4*XSEP	MAIN 462
	GO TO 9307	MAIN 463
9306	XSEP=.6*XSEP+.4*XSEPS	MAIN 464
9307	IF(XSEP-XMAX) 9212,9212,7835	MAIN 465
9212	CALL CPC(I SEPT,NGAM,NF,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,BCAP,	MAIN 466
	ITHICK,ROBB,GAMAW,UINF,UDOT,1.,XSEP,DXI,CP1)	MAIN 467
	IF(NOTBL .EQ. 2 .AND. XSEP .GT. 0.) XSEP=-.98	
	GO TO 3248	MAIN 468
7743	IF(NITS-1) 7737,7737,3248	MAIN 469

```

7737 NITS=NITS+1                                MAIN 470
      ELDOT=ELDI                                  MAIN 471
      GO TO 3248                                  MAIN 472
1786 WRITE(MOUT,20) NTIME                        MAIN 473
      WRITE(MOUT,26) XIVOR                        MAIN 477
      PIRC = PITCH * 180. / PI
209  WRITE(MOUT,10) TIME,UINF,XSEP,XATT,PIRC     MAIN 473
      ALDFG= ALPH1/DEGRFS                        SUPPL349
      WRITE(6,9001) Z,ALDEG,ALPH1, ALPH2, HEAVE SUPPL350
      IF( PSI360 .GE. PSILOW .AND. PSI360 .LE. PSIUP) GO TO 101
      IF( NOUT .EQ. 0)
1WRITE(MOUT,11)                                MAIN 479
      IF( NOUT .EQ. 0)
1WRITE(MOUT,12) (N,XGAM(N),VZIP(N),AR(N),ACAP(N,1),XIW(N),GAMAW(N),MAIN 480
2N=1,NGPI)                                     MAIN 481
      IF(ISEP) 7432,7433,7432                    MAIN 482
7432  IF( NOUT .EQ. 0)
1WRITE(MOUT,13)                                MAIN 483
      IF( NOUT .EQ. 0)
1WRITE(MOUT,17) (N,XBSIG(N),FPRES(N),ARH(N),BCAP(N,1),N=1,NSIG) MAIN 484
      WRITE(MOUT,14) ELDOT                        MAIN 485
      WRITE(MOUT,18) XSIG(1),CPOT,X4,CPOT,XATT,PREC MAIN 486
7433  WRITE(MOUT,15)                             MAIN 487
      XPC=-1.                                    MAIN 488
      DO 7102 N=1,NCPI                           MAIN 489
      CALL QECAL(ISEP,NGAM,NSIG,NF,XSIG,ACAP,BCAP,THICK,RDBB,GAMAW(1),UI MAIN 490
INF,XPC,QFL,-1.)                               MAIN 491
      CALL QECAL(ISEP,NGAM,NSIG,NF,XSIG,ACAP,BCAP,THICK,RDBB,GAMAW(1),UI MAIN 492
INF,XPC,QUE,1.)                                 MAIN 493
      CALL CPC(ISEP,NGAM,NF,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,BCAP,MAIN 494
1THICK,RDBB,GAMAW,UINF,JDOT,1.0,XPC,DXI,CPU)   MAIN 495
      CALL CPC(ISEP,NGAM,NF,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,BCAP,MAIN 496
1THICK,RDBB,GAMAW,UINF,UDOT,-1.,XPC,DXI,CPL)   MAIN 497
      IF(N=1) 7546,7545,7546                     MAIN 498
7545  CPL=CPU                                    MAIN 499
7546  DLIFT=CPL-CPU                              MAIN 500
      WRITE(MOUT,16) XPC,QEL,CPL,QUE,CPU,DLIFT   MAIN 501
7102  XPC=XPC+PINT                               MAIN 502
101   CONTINUE
      CMPAS=CMPA                                  MAIN 503
      CALL CLCM(NCOI,ISEP,NGAM,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,RCMAIN 504
1AP,THICK,RDBB,GAMAW,UINF,UDOT,DXI,AROT,CMPA)   MAIN 505
      P(L,2) = PIRC
      P(L,3) = Z(3)
      P(L,4) = Z(1)
      P(L,5) = Z(2)
      P(L,6) = CLVR
      P(L,7) = CMPA
      IF( L .LT. 200 ) GO TO 98
      CALL PLOTSB( PLOTOP , P , L )
      L= 0
98   CONTINUE
      IF(ISTD .EQ. 1) GO TO 9999
      DO 7950 M=1,MX                                MAIN 506
      SCALE(M,2)=SCALE(M,1)                       MAIN 507

```

	SCALE(M,1)=SCALS(M)	MAIN 508
	DC 7950 N=1,NY	MAIN 509
	U(M,N,2)=U(M,N,1)	MAIN 510
7950	U(M,N,1)=USAV(M,N)	MAIN 511
	GO TO 9999	MAIN 512
8989	CONTINUE	MAIN 513
99	CONTINUE	
	CALL PLOTSB(PLOTOP , P , L)	
	CALL ACUCPU(IACU)	
	IF(IACU .LT. 35000) GO TO 60	
	GO TO 40	
60	CONTINUE	
	IF(PLOTOP.EQ. 0.) CALL EXIT	
	CALL PLTND	
	CALL EXIT	
	RETURN	
C		
C		
C		
C		
1	FORMAT(13I5)	MAIN 23
2	FORMAT(3F10.4)	MAIN 24
3	FORMAT(2F10.4)	MAIN 25
4	FORMAT(1H1//)	MAIN 26
5	FORMAT(6F10.4)	MAIN 27
6	FORMAT(1H1,50X,34HANALYSIS OF UNSTEADY AIRFOIL STALL///)	MAIN 28
7	FORMAT(8X,6HUBAR =E13.5/7X,7HUFREQ =E13.5//3X,11HALPHA ONE =E13.5/MAIN 29	
	13X,11HALPHA TWO =E13.5/8X,6HHBAR =E13.5/11X,3HA =E13.5/8X,6HFREQ =MAIN 30	
	1E13.5//8X,6HRO/B =E13.5//9X,5HREB =E13.5//)	MAIN 31
8	FORMAT(29X,1HN,25X,4HC(N),26X,4HT(N)/)	MAIN 32
9	FORMAT(130,2E30.5)	MAIN 33
10	FORMAT(5X,3HT =E13.5/5X,3HU =E13.5/4X,4HXS =E13.5/4X,4HXO =E13.5/4MAIN 34	
	1X,4HPA =E13.5//)	MAIN 35
11	FORMAT(///4X,1HN,11X,1HX,14X,5HVZ(X),12X,5HRN(X),12X,4HA(N),21X,3HMAIN 36	
	1XIW,14X,5HGAMMA/)	MAIN 37
12	FORMAT(15,4E17.5,8X,2E17.5)	MAIN 38
13	FORMAT(1H1,8X,1HN,20X,1HX,21X,5HFP(X),22X,5HRH(N),21X,4HB(N)/)	MAIN 39
14	FORMAT(//54X,9H L-DCT =E13.5//51X,27HPRESSURES IN SEPARATED FLOWMAIN 40	
	1//55X,1HX,19X,2HCP/)	MAIN 41
15	FORMAT(1H1,11X,1HX,16X,3HQEL,15X,3HCPL,15X,3HQEU,15X,3HCPU,13X,9HCMAN 42	
	1PL - CPU/)	MAIN 43
16	FORMAT(6E18.5)	MAIN 44
17	FORMAT(110,4E25.5)	MAIN 45
18	FORMAT(3(40X,2E20.5//))	MAIN 46
19	FORMAT(15,5F10.4)	MAIN 47
20	FORMAT(1H1,50X,12HTIME STEP NO13//)	MAIN 48
22	FORMAT(///40X,26HINCREASE IN CP REQUIRED ISE13.5//40X,26HINCREASE MAIN 49	
	1IN CP POSSIBLE ISE13.5)	MAIN 50
23	FORMAT(///45X,23HPOTENTIAL FLOW XS =E12.4/60X,8HCP(XS) =E12.4/MAIN 51	
	1/45X,23HBOUNDARY LAYER XS =E12.4)	MAIN 52
24	FORMAT(15,4F10.4/5F10.4)	MAIN 53
25	FORMAT(12X,4HNV =E12.4,3X,3HS =E12.4,3X,3HM =E12.4,3X,3HG =E12.4,3X,4MAIN 54	
	1HX1 =E12.4//12X,4HMI =E12.4,3X,4HWT =E12.4,3X,4HPA =E12.4//)	MAIN 55
26	FORMAT(4X,4HX1 =E13.5)	MAIN 56
9001	FORMAT('0', Y50, 'EQUIVALENT ROTOR BLADE RESPONSE'	SUPPL380

9001A	//	T 5,	'FLAP DISP =',	G14.5		SUPPL381
9001B	,	T47,	'RENDING DISP =',	G14.5		SUPPL382
9001C	,	T39,	'TORSIONAL DISP =',	G14.5		SUPPL383
9001D	/	T38,	'SECTION PITCH ANGLE =',	F9.3,	' DEGREES OR ',	SUPPL384
9001E				F9.4,	' RADIANS '	SUPPL385
9001F	/	T21,	'SECTION PITCH RATE =',	G14.5		SUPPL386
9001G	,	T71,	'SECTION PLUNGING RATE =',	G14.5	//)	SUPPL387
		END				MAIN 515

	SUBROUTINE SUPPL	SUPPL 1
	IMPLICIT REAL*8 (A-H,O-Z)	SUPPL 2
	REAL*8 FR1S, FR2S, FR3S, ANSX, OMS	SUPPL 3
C		SUPPL 4
	REAL*4 CLVB, CMVB, CMPAVB	
	I, DUMMY, PLOTOP	
	REAL FTVB, FPVB, FPPRVB, DIDRVB, XMVB, DELVB, XMUVB,	SUPPL 5
A	FOVB, XMUAVB, ATOVB, ATCVB, ATSVB, ROVB, RVB, MVB,	SUPPL 6
C	WDXI, PSI, UINF	SUPPL 7
	REAL ELSIG, DXI, REB, RDBR, FRZ, ARR, AMPLU, FREQU,	SUPPL 8
A	ALPH1, ALPH2, HEAVE, AROT, FREQF, PHIH, RY1, DRY,	SUPPL 9
B	X, TEST, UPRIM, XU, YU, XL, YL, ER1, ER2, ER3, BDBR,	SUPPL 10
C	RROBR	SUPPL 11
	REAL SUM(8), YCLD(8), YNEW(8), DEL(3,3), CMPA(3), CL(3), G(3),	SUPPL 12
A	Z, ZPR(3), SMALLG(3), Y(3,3), YPR(3,3), GCAP(3,3)	SUPPL 13
	COMMON /BL1/ NTIME, NDMC	
	COMMON /CLCMBL/ CLVB, CMVB, CMPAVB	MAIN
	COMMON /ZZZ/ Z(3)	
	COMMON /INPTVB/ FTVB(64), FPVB(64), FPPRVB(64), DIDRVB(64),	SUPPL 15
A	XMVB(64), DELVB, XMUVB, FOVB, XMUAVB,	SUPPL 16
B	ATOVB, ATCVB, ATSVB, ROVB, RVB(64),	SUPPL 17
C	MVB(64), NVB	SUPPL 18
	COMMON /INPUTS/ NSBL, NZ, NOFF, NGAM, NSIG,	SUPPL 19
A	NCOI, NCORD, LOWER, MSTOP, MAXT, MCTR,	SUPPL 20
B	NOTBL, INDV, ELSIG, DXI, REB, RDBR,	SUPPL 21
C	FRZ, ARR, AMPLU, FREQU, ALPH1, ALPH2,	SUPPL 22
D	HEAVE, AROT, FREQF, PHIH, NY, RY1,	SUPPL 23
E	DRY, X(100), TEST, UPRIM, XU(30), YU(30),	SUPPL 24
F	XL(30), YL(30), ER1, ER2, ER3, BDBR,	SUPPL 25
G	RROBR	SUPPL 26
	H, DUMMY(10), PLOTOP	
	DIMENSION DELTA(3,3)	SUPPL 27
	DIMENSION ALPHA(3,3), BETA(3,3), GAMMA(3,3), OMS(3), OMEGA(3), CHK(3)	SUPPL 28
	DIMENSION AA(10), AB(10), ANB(20), ANT(20), AAX(10), ANSX(20), SORT(3)	SUPPL 29
	I, TOT(2)	
	CF4(X)=F4-B4+(B4*G6-C4)*X*X	SUPPL 30
	Z1(X)=HB*(CF4(X)/GB)**2+(CF4(X)*FR1S+(1.-C6*X*X)*B2-F2)*X*X	SUPPL 31
	Z2(X)=(FZ/FR1S+FR1S*CF4(X)-F2+(1.-C6*X*X)*(B2-BZ/FR1S))*X*X	SUPPL 32
	S1(X)=(2.*HB*CF4(X)/GB**2+(FR1S-FR2S)*X*X)*GA	SUPPL 33
	S2(X)=(FR1S-FR2S)*GA*X*X	SUPPL 34
	FUN(X)=(R1*Z2(X)-R2*Z1(X))**2+(R1*S2(X)-R2*S1(X))*(Z2(X)*S1(X)-Z1(X)*S2(X))	SUPPL 35
	DATA BBS,REL,NPOL/1.E-7,1.E-6,3/	SUPPL 36
		SUPPL 39
C		SUPPL 40
C	MASSSES AND H'S ARE NONDIMENSIONAL, WITH BLADE MASS AND RADIUS	SUPPL 41
C	AS REFERENCES. NONROTATING NATURAL FREQUENCIES ARE	SUPPL 42
C	DIMENSIONLESS, USING ROTOR SPEED AS REFERENCE. DISTANCES XBAB, SILB,	SUPPL 43
C	AND S2LB ARE FRACTIONS OF SEMICORD. XBAR, SIL, AND S2L ARE	SUPPL 44
C	FRACTIONS OF ROTOR RADIUS.	SUPPL 45
C		SUPPL 46
	NDIMC=3	
	DO 63 K = 1, 8	SUPPL 47
	SUMTKI = 0.	SUPPL 48
63	YNEW(K) = 0.	SUPPL 49
	DO 69 I = 1, NVB	SUPPL 50

	DO 66 K = 1, 8	SUPPL 51
66	YOLD(K) = YNEW(K)	SUPPL 52
	CALL YVB(YNEW,I)	SUPPL 53
	IF(I .LE. 1) GO TO 69	SUPPL 54
	DO 67 K = 1, 8	SUPPL 55
67	SUM(K) = (YNEW(K) + YOLD(K)) * (RVB(I) - RVB(I-1)) / 2. + SUM(K)	SUPPL 56
69	CONTINUE	SUPPL 57
	EM11 = SUM(1)	SUPPL 59
	EM22 = SUM(2)	SUPPL 60
	EM33 = SUM(3)	SUPPL 61
	EM13 = SUM(4)	SUPPL 62
	EM23 = SUM(5)	SUPPL 63
	H11 = SUM(6)	SUPPL 64
	H22 = SUM(7)	SUPPL 65
	H33 = - EM33	SUPPL 66
	H13 = -EM13	SUPPL 67
	H23 = SUM(8)	SUPPL 68
	BDBRR=BDBR/RRDBR	SUPPL 69
	BDS=BDBRR**2	SUPPL 70
	T11=H11*BDS	SUPPL 71
	T22=H22*BDS	SUPPL 72
	T33=H33*BDS	SUPPL 73
	T13=H13*BDS	SUPPL 74
	T23=H23*BDS	SUPPL 75
	FR1S=BDS*ER1**2-T11/EM11	SUPPL 76
	FR2S=ER2**2*BDS-T22/EM22	SUPPL 77
	FR3S=ER3**2*BDS-T33/EM33	SUPPL 78
	FR1=DSQRT(FR1S)	SUPPL 79
	FR2=DSQRT(FR2S)	SUPPL 80
	FR3=DSQRT(FR3S)	SUPPL 81
	RATM=EM11/EM22	SUPPL 82
	ZETA=(1.+RATM)*(RATM*FR1S**2+FR2S**2)/(RATM*FR1S+FR2S)**2	SUPPL 83
	RM=ZETA-1.	SUPPL 84
	SUMS=FR1S+FR2S	SUPPL 85
	HIGHS=(SUMS+DSQRT(SUMS**2-4.*ZETA*FR1S*FR2S))/(2.*ZETA)	SUPPL 86
	SMALS=FR1S*FR2S/HIGHS	SUPPL 87
	DEN=FR2S-FR1S	SUPPL 88
	A1=-(HIGHS-FR1S)/DEN	SUPPL 89
	A2=-1.-A1	SUPPL 90
	B=-A1*A2*DEN/HIGHS	SUPPL 91
	SLAM1=EM11*BDBR**2/EM33	SUPPL 92
	SLAMZ=-A1*SLAM1	SUPPL 93
	SLAM2=-SLAMZ/A2	SUPPL 94
	SUM3=SUMS+FR3S	SUPPL 95
	ADD2=FR1S*(FR2S+FR3S)+FR2S*FR3S	SUPPL 96
	ADDZ=FR1S*FR2S*FR3S	SUPPL 97
	BBAR=1.-(EM13**2/EM11+EM23**2/EM22)/EM33	SUPPL 98
	B4=SUM3+(2.*EM23*T23/EM22+2.*EM13*T13/EM11-FR1S*EM23**2/EM22-FR2S*	SUPPL 99
	EM13**2/EM11)/EM33	SUPPL100
	B4=B4/BBAR	SUPPL101
	B2=ADD2+T2.*FR2S*EM13*T13/EM11+2.*FR1S*EM23*T23/EM22-T13**2/EM11-T	SUPPL102
	123**2/EM22)/EM33	SUPPL103
	B2=B2/BBAR	SUPPL104
	BZ=ADDZ-(FR2S*T13**2/EM11+FR1S*T23**2/EM22)/EM33	SUPPL105
	BZ=BZ/BBAR	SUPPL106

	C6=(EM11*A1**2+EM22*A2**2)/EM33	SUPPL107
	F4=SUM3	SUPPL108
	C4=(FR2S*EM11*A1**2+FR1S*EM22*A2**2)/EM33	SUPPL109
	GA=2.*EM11*A1/EM33	SUPPL110
	GB=2.*EM22*A2/EM33	SUPPL111
	F2=ADD2	SUPPL112
	HA=EM11/EM33	SUPPL113
	HB=EM22/EM33	SUPPL114
	FZ=ADDZ	SUPPL115
	R1=-HA-HB*(GA/GB)**2	SUPPL116
	R2=HA*(FR2S/FR1S-1.)	SUPPL117
	ZLAM=F4-B4	SUPPL118
	TWLAM=B4*C6-C4	SUPPL119
	FZHAT=HB*(ZLAM/GB)**2	SUPPL120
	F2HAT=B2-F2+(FR1S*ZLAM+2.*ZLAM*TWLAM*HB/GB)**2	SUPPL121
	F4HAT=-C6*R2+FR1S*TWLAM+HB*(TWLAM/GB)**2	SUPPL122
	G2HAT=B2-F2+(FZ-BZ)/FR1S+FR1S*ZLAM	SUPPL123
	G4HAT=-C6*(B2-BZ/FR1S)+FR1S*TWLAM	SUPPL124
	SIGZ=2.*HB*ZLAM*GA/GB**2	SUPPL125
	SIG2=GA*(FR1S-FR2S+2.*HB*TWLAM/GB)**2	SUPPL126
	GAM2=GA*(FR1S-FR2S)	SUPPL127
	UZ=-R2*FZHAT	SUPPL128
	U1=R1*G2HAT-R2*F2HAT	SUPPL129
	U2=R1*G4HAT-R2*F4HAT	SUPPL130
	U3=-R2*SIGZ	SUPPL131
	U4=R1*GAM2-R2*SIG2	SUPPL132
	U5=SIGZ*G2HAT-GAM2*FZHAT	SUPPL133
	U6=SIGZ*G4HAT+SIG2*G2HAT-GAM2*F2HAT	SUPPL134
	U7=SIG2*G4HAT-GAM2*F4HAT	SUPPL135
	AAX(1)=UZ**2	SUPPL136
	AAX(2)=2.*UZ*U1+U3*U5	SUPPL137
	AAX(3)=U1**2+2.*UZ*U2+U3*U6+U4*U5	SUPPL138
	AAX(4)=2.*U1*U2+U3*U7+U4*U6	SUPPL139
	AAX(5)=U2**2+U4*U7	SUPPL140
	CALL POLLY(4,RBS,REL,ANSX,AAX)	SUPPL141
	XBAR=1.E25	SUPPL142
	DO 86 I=1,4	SUPPL143
	IP=2*I	SUPPL144
	IM=IP-1	SUPPL145
	IF(DABS(ANSX(IM)).GT.1.D-10) GO TO 86	SUPPL146
	IF(ANSX(IP).LE.0.) GO TO 86	SUPPL147
	XBART=DSQRT(ANSX(IP))	SUPPL148
	IF(XBART.LT.XBAR) XBAR=XBART	SUPPL149
86	CONTINUE	SUPPL150
	IF(XBAR.LT..5E25) GO TO 88	SUPPL151
	WRITE(6,87)	SUPPL152
87	FORMAT(IH1,IOX,'NO SOLUTION FOR XBAR')	SUPPL153
	STOP	SUPPL154
88	CONTINUE	SUPPL155
15	ALOW=(R1*Z2(XBAR)-R2*Z1(XBAR))/(R1*S2(XBAR)-R2*S1(XBAR))	SUPPL156
	ALOW=ALOW/XBAR	SUPPL157
	BLOW=(CF4(XBAR)-GA*ALOW*XBAR)/(XBAR*CB)	SUPPL158
	XI=-ALOW-BLOW	SUPPL159
	ETA=(BLOW*A1-ALOW*A2)/(A1-A2)	SUPPL160
	S2L=FYA/(B*HTGHS)	SUPPL161

	SIL=(XI-RM*HIGHS*S2L)*HIGHS/(FR1S*FR2S)	SUPPL162
	WRITE(6,4) ER1,ER2,EF3,RM	SUPPL163
	WRITE(6,721) FR1,FR2,FR3,ALOW,BLOW	SUPPL164
	WRITE(6,5) EM11,EM22,EM33,FM13,EM23	SUPPL165
	WRITE(6,6) H11,H22,H33,H13,H23	SUPPL166
	C13=ALOW/BDBR	SUPPL167
	C23=BLOW/BDBR	SUPPL168
	XBAR=XPAB/BDBR	SUPPL169
	SILB=SIL/BDBR	SUPPL170
	S2LB=S2L/BDBR	SUPPL171
	WRITE(6,41) BDBR,RRDBR	SUPPL172
	WRITE(6,7) XBAR,XBAB,SIL,SILB,S2L,S2LB,SMALS,HIGHS	SUPPL173
	AA(1)=B7	SUPPL174
	AA(2)=B2	SUPPL175
	AA(3)=B4	SUPPL176
	AA(4)=1.	SUPPL177
	CALL POLLY(NPOL,BBS,REL,ANB,AA)	SUPPL178
	SSX=SLAMZ*XBAB	SUPPL179
	DIV=1.-SLAMZ*XBAR**2	SUPPL180
	BETA(3,1)=(SLAM1*C13+SSX*FR1S)/DIV	SUPPL181
	BETA(3,2)=(SLAM2*C23+SSX*FR2S)/DIV	SUPPL182
	BETA(3,3)=(FR3S+SSX*(C13+C23))/DIV	SUPPL183
	AXB=A1*XBAB	SUPPL184
	BETA(1,1)=FR1S-AXB*BETA(3,1)	SUPPL185
	BETA(1,2)=-AXB*BETA(3,2)	SUPPL186
	BETA(1,3)=C13-AXB*BETA(3,3)	SUPPL187
	AAXB=A2*XBAB	SUPPL188
	BETA(2,1)=-AAXB*BETA(3,1)	SUPPL189
	BETA(2,2)=FR2S-AAXB*BETA(3,2)	SUPPL190
	BETA(2,3)=C23-AAXB*BETA(3,3)	SUPPL191
	AB(4)=1.	SUPPL192
	AB(3)=BETA(1,1)+BETA(2,2)+BETA(3,3)	SUPPL193
	AB(2)=BETA(1,1)*(BETA(2,2)+BETA(3,3))+BETA(2,2)*BETA(3,3)-BETA(3,2)	SUPPL194
	1)*BETA(2,3)-BETA(1,2)*BETA(2,1)-BETA(1,3)*BETA(3,1)	SUPPL195
	AB(1)=BETA(1,1)*(BETA(2,2)*BETA(3,3)-BETA(3,2)*BETA(2,3))-BETA(2,1)	SUPPL196
	1)*(BETA(1,2)*BETA(3,3)-BETA(3,2)*BETA(1,3))+BETA(3,1)*(BETA(1,2)*B	SUPPL197
	1)FTA(2,3)-BETA(1,3)*BETA(2,2)	SUPPL198
	CALL POLLY(NPOL,BBS,REL,ANT,AB)	SUPPL199
	WRITE(6,44)	SUPPL200
	DO 45 I=1,4	SUPPL201
	ITM=(I-1)*2	SUPPL202
45	WRITE(6,46) IM,AA(I),AB(I)	SUPPL203
	WRITE(6,47)	SUPPL204
	DO 48 I=1,3	SUPPL205
	ITT=2*I	SUPPL206
	ITM=ITT-1	SUPPL207
48	WRITE(6,49) ANB(ITT),ANB(ITM),ANT(ITT),ANT(ITM)	SUPPL208
	DO 301 I=1,3	SUPPL209
	II=2*I	SUPPL210
301	OMS(I)=-ANT(II)	SUPPL211
	MAXI=3	SUPPL212
	DO 70 I=1,2	SUPPL213
	IF(OMS(I).GT.OMS(MAXI)) MAXI=I	SUPPL214
70	CONTINUE	SUPPL215
	GO TO (71,72,73),MAXI	SUPPL216

71	I1=2	SUPL217
	I2=3	SUPL218
	GO TO 74	SUPL219
72	I1=1	SUPL220
	I2=3	SUPL221
	GO TO 74	SUPL222
73	I1=1	SUPL223
	I2=2	SUPL224
74	IF(OMS(I1).GT.OMS(I2)) GO TO 75	SUPL225
	MINI=I1	SUPL226
	MIDI=I2	SUPL227
	GO TO 76	SUPL228
75	MINI=I2	SUPL229
	MIDI=I1	SUPL230
76	SORT(1)=OMS(MINI)	SUPL231
	SORT(2)=OMS(MIDI)	SUPL232
	SORT(3)=OMS(MAXI)	SUPL233
	DO 77 I=1,3	SUPL234
	OMS(I)=SORT(I)	SUPL235
77	OMEGA(I)=DSQRT(CMS(I))	SUPL236
	DO 302 I=1,3	SUPL237
302	ALPHA(I,I)=1.	SUPL238
	DENB=BETA(2,1)*BETA(3,2)-BETA(3,1)*(BETA(2,2)-OMS(1))	SUPL239
	ALPHA(1,2)=(BETA(1,2)*BETA(3,1)-BETA(3,2)*(BETA(1,1)-OMS(1)))/DENB	SUPL240
	ALPHA(1,3)=((BETA(2,2)-OMS(1))*(BETA(1,1)-OMS(1))-BETA(1,2)*BETA(2,1,1))/DENB	SUPL241
	CHK(1)=BETA(1,3)*ALPHA(1,1)+BETA(2,3)*ALPHA(1,2)+(BETA(3,3)-OMS(1))*ALPHA(1,3)	SUPL242
	DENB=BETA(3,2)*(BETA(1,1)-CMS(2))-BETA(3,1)*BETA(1,2)	SUPL243
	ALPHA(2,1)=(BETA(3,1)*(BETA(2,2)-OMS(2))-BETA(2,1)*BETA(3,2))/DENB	SUPL244
	ALPHA(2,3)=(BETA(2,1)*BETA(1,2)-(BETA(1,1)-OMS(2))*(BETA(2,2)-OMS(2)))/DENB	SUPL245
	CHK(2)=BETA(1,3)*ALPHA(2,1)+BETA(2,3)*ALPHA(2,2)+(BETA(3,3)-OMS(2))*ALPHA(2,3)	SUPL246
	DENB=BETA(2,3)*(BETA(1,1)-CMS(3))-BETA(1,3)*BETA(2,1)	SUPL247
	ALPHA(3,1)=(BETA(2,1)*(BETA(3,3)-OMS(3))-BETA(3,1)*BETA(2,3))/DENB	SUPL248
	ALPHA(3,2)=(BETA(3,1)*BETA(1,3)-(BETA(1,1)-OMS(3))*(BETA(3,3)-OMS(3)))/DENB	SUPL249
	CHK(3)=BETA(1,2)*ALPHA(3,1)+(BETA(2,2)-OMS(3))*ALPHA(3,2)+BETA(3,2)*ALPHA(3,3)	SUPL250
	WRITE(6,488)	SUPL251
	WRITE(6,489) (I,OMEGA(I),BETA(I,1),BETA(I,2),BETA(I,3),ALPHA(I,1),ALPHA(I,2),ALPHA(I,3),CHK(I),I=1,3)	SUPL252
	SORT(1)=1.	SUPL253
	SORT(2)=0.	SUPL254
	SORT(3)=0.	SUPL255
	DO 432 J=1,3	SUPL256
	GO TO (381,382,383), J	SUPL257
382	SORT(1)=0.	SUPL258
	SORT(2)=1.	SUPL259
	SORT(3)=0.	SUPL260
	GO TO 381	SUPL261
383	SORT(1)=0.	SUPL262
	SORT(2)=0.	SUPL263
	SORT(3)=1.	SUPL264
		SUPL265
		SUPL266
		SUPL267
		SUPL268
		SUPL269
		SUPL270
		SUPL271

```

3E1 DO 384 I=1,3 SUPPL272
DO 384 K=1,3 SUPPL273
384 DELTA(I,K)=ALPHA(I,K) SUPPL274
CALL ALSOL(3,DELTA,SCRT,3) SUPPL275
DO 431 I=1,3 SUPPL276
431 GAMMA(I,J)=SCRT(I) SUPPL277
432 CONTINUE SUPPL278
WRITE(6,11) SUPPL279
WRITE(6,12) (I,GAMMA(I,1),GAMMA(I,2),GAMMA(I,3),I=1,3) SUPPL280
AMPLU = XMUAVB * (1. - ROVB**3) / (1. - ROVB**4) * 1.3333333333333333 SUPPL284
SA = SMALS * S1LB + RM * S2LB * HIGHS SUPPL285
SB = SMALS * S1LB**2 + RM * S2LB**2 * HIGHS SUPPL286
DEL(1,1) = XMUVB * (1. - ROVB**4) / (4. * (1. - SLAMZ * XBAB**2) SUPPL287
A * RRDBR * FM11 ) SUPPL288
DEL(1,2) = 2. * SLAMZ * XBAB * DEL(1,1) SUPPL289
DEL(1,3) = A1 * (SLAMZ * XBAB * SB - SA) / (1. - SLAMZ * XBAB**2) SUPPL290
A + B * HIGHS * S2LB SUPPL291
DEL(2,1) = A2 / A1 * DEL(1,1) SUPPL292
DEL(2,2) = A2 / A1 * DEL(1,2) SUPPL293
DEL(2,3) = A2 * (SLAMZ * XBAB * SB - SA) / (1. - SLAMZ * XBAB**2) SUPPL294
A - R * SMALS * S2LB SUPPL295
DEL(3,1) = - SLAMZ * XBAB * DEL(1,1) / A1 SUPPL296
DEL(3,2) = -2. * SLAMZ * DEL(1,1) / A1 SUPPL297
DEL(3,3) = (BDBR / RRDBR)**2 + SLAMZ * (XBAB * SA - SR) / SUPPL298
A (1. - SLAMZ * XBAB**2) SUPPL299
CMPA(2) = CMPAVB MAIN
CL(2) = CLVB MAIN
NDIMC = 6)
COSPSI = 1.
SINPSI = 0.
TO = ATOVB + ATCVB * COS PSI + ATSVB * SIN PSI
TOT(1) = TO - ATOVB
DO 50 I=1,3
50 SMALLG(I) = DEL(I,1) * CLVB + DEL(I,2) * CMPAVB
DO 51 I=1,3
GCAP(I,1) = 0.
DO 52 J=1,3
YPR(I,J) = 0.
52 GCAP(I,1) = GCAP(I,1) + ALPHA(I,J) * SMALLG(J)
GCAP(I,2) = GCAP(I,1)
Y(I,1) = GCAP(I,1) / CMS(I)
51 Y(I,2) = Y(I,1)
IF ( PLOTOP .LT. 0.)
1 WRITE( 6,9000) TO, Z, TOPR, ZPR, Y, YPR, DEL, SMALLG
9000 FORMAT(// ' TO=', IP1E13.6, ' Z=', IP3E13.6, ' TOPR=', IP1E13.6
1 , ' ZPR=', IP3E13.6 / ' Y=', IP9E13.6 / ' YPR= ', IP9E13.6
2 / ' DEL= ', IP9E13.6 / ' SMALLG= ', IP9E13.6 / )
RETURN SUPPL300
C SUPPL301
C SUPPL302
ENTRY SUPPI (UINF) SUPPL303
C SUPPL304
C SUPPL305
C SUPPL306
CMPA(3) = CMPA(2) MAIN
CMPA(2) = CMPAVB

```



```

1HL1/B =E13.5/20X,6HL2/R =E13.5,10X,6HL2/B =E13.5/9X,7HK1/M1 =E13.5SUPPL362
1/9X,74K2/M2 =E13.5) SUPPL363
41 FCRMAT(//10X,5HRR/R =E13.5,20X,6HRR/R =E13.5//) SUPPL364
44 FORMAT(1H1,20X,'POLYNOMIAL COEFFICIENTS'///7X,5HPDWER,12X,5HBLADE,SUPPL365
126X,3H2-D//) SUPPL366
46 FORMAT(I10,2D30.9) SUPPL367
47 FORMAT(1H1,20X,'ROOTS OF POLYNOMIALS'///30X,'BLADE',60X,'2-)'//20X,SUPPL368
14HREAL,21X,4HIMAG,31X,4HREAL,21X,4HIMAG//) SUPPL369
49 FORMAT(2D25.9,10X,2D25.9) SUPPL370
11 FORMAT(/////9X,1H1,15X,10HGAMMA(I,1),15X,10HGAMMA(I,2),15X,10HGAMMSUPPL371
IA(I,3)//) SUPPL372
12 FORMAT(I10,3E25.5) SUPPL373
488 FORMAT(1H1,8X,1H1,7X,5HOMEGA,4X,9HBETA(I,1),4X,9HBETA(I,2),4X,9HBSUPPL374
1TA(I,3),3X,10ALPHA(I,1),3X,10ALPHA(I,2),3X,10ALPHA(I,3),8X,3HCHSUPPL375
IK//) SUPPL376
489 FORMAT(I10,8E13.5) SUPPL377
721 FORMAT(//10X,5HFR1 =E13.5,10X,5HFR2 =E13.5,10X,5HFR3 =E13.5//10X,SUPPL378
14HSA =E13.5,10X,4HSB =E13.5//) SUPPL379
END

```

```

SURROUTINE SETUPS
C
IMPLICIT REAL*8 (A-H,O-Z)
C
REAL FTVB, FPVB, FPPRVB, DIDRVB, XMVB, DELVB, XMUVB,
A FOVB, XMUAVB, ATOVB, ATCVB, ATSVB, ROVB, RVB, MVB
REAL ELSIG, DXI, REB, RDBB, FRZ, ARR, AMPLU, FREQU,
A ALPH1, ALPH2, HEAVE, AROT, FREQF, PHIH, RY1, DRY,
B Y, TEST, UPRIM, XU, YU, XL, YL, ER1, ER2, ER3, BDBR,
C RRDBR
H, CMPA, CMPAS, BARG, EMI, HVOR, SSPA, SVOR, TORF, XIVOR
I, PLOTOP, PSILOW, PSIUP
C
INTEGER TABLE(7, 80) /560 * ' ' /
C
COMMON /BL1/ NTIME
C
COMMON /INPTVB/ FTVB(64), FPVB(64), FPPRVB(64), DIDRVB(64),
A XMVB(64), DELVB, XMUVB, FOVB, XMUAVB,
B ATOVB, ATCVB, ATSVB, ROVB, RVB(64),
C MVB(64), NVB
C
COMMON /INPUTS/ NSBL, NZ, NOFF, NGAM, NSIG,
A NCOI, NCORD, LOWER, MSTOP, MAXT, MTR,
B NOTBL, INDV, ELSIG, DXI, REB, RDBB,
C FRZ, ARR, AMPLU, FREQU, ALPH1, ALPH2,
D HEAVE, AROT, FREQF, PHIH, NY, RY1,
E DRY, Y(100), TEST, UPRIM, XU(30), YU(30),
F XL(30), YL(30), ER1, ER2, ER3, BDBR,
G RRDBR
H, CMPA, CMPAS, BARG, EMI, HVOR, NVOR, SSPA, SVOR, TORF, XIVOR
I, PLOTOP, PSILOW, PSIUP
J, NOUT
C
CALL WHERE(TABLE)
CALL ZEROIN
C
CALL SETUP('ALPH1', 4, ALPH1)
CALL SETUP('ALPHA1', 4, ALPH1)
CALL SETUP('ALPH2', 4, ALPH2)
CALL SETUP('ALPHA2', 4, ALPH2)
CALL SETUP('AMPLU', 4, AMPLU)
CALL SETUP('ARR', 4, ARR)
CALL SETUP('AROT', 4, AROT)
CALL SETUP('ATOV', 4, ATOVB)
CALL SETUP('ATCV', 4, ATCVB)
CALL SETUP('ATSV', 4, ATSVB)
CALL SETUP('BARG', 4, BARG)
CALL SETUP('BDBR', 4, BDBR)
CALL SETUP('CMPA', 4, CMPA)

```

```

SETUPS 1
SETUPS 2
SETUPS 3
SETUPS 4
SETUPS 5
SETUPS 6
SETUPS 7
SETUPS 8
SETUPS 9
SETUPS10
SETUPS11
SETUPS12
SETUPS14
SETUPS15
SETUPS16
SETUPS17
SETUPS18
SETUPS19
SETUPS20
SETUPS21
SETUPS22
SETUPS23
SETUPS24
SETUPS25
SETUPS26
SETUPS27
SETUPS28
SETUPS29
SETUPS30
SETUPS31
SETUPS32
SETUPS33
SETUPS34
SETUPS35
SETUPS36
SETUPS37
SETUPS38
SETUPS39
SETUPS40
SETUPS41
SETUPS42
SETUPS43
SETUPS44
SETUPS45
SETUPS46
SETUPS47
SETUPS48

```

CALL SETUP('CMPAS	' , 4 ,	CMPAS)	
CALL SETUP('DELVB	' , 4 ,	DELVB)	SETUPS49
CALL SFTUP('DIDRVB	' , 4 ,	DIDRVB , 64)	SETUPS50
CALL SFTUP('DRY	' , 4 ,	DRY)	SETUPS51
CALL SFTUP('DXI	' , 4 ,	DXI)	SETUPS52
CALL SETUP('ELSIG	' , 4 ,	ELSIG)	SETUPS53
CALL SETUP('EMI	' , 4 ,	EMI)	
CALL SETUP('ER1	' , 4 ,	ER1)	SETUPS54
CALL SFTUP('ER2	' , 4 ,	ER2)	SETUPS55
CALL SETUP('ER3	' , 4 ,	ER3)	SETUPS56
CALL SETUP('FPVB	' , 4 ,	FPVB , 64)	SETUPS57
CALL SETUP('FPPRVB	' , 4 ,	FPPRVB , 64)	SETUPS58
CALL SETUP('FRZ	' , 4 ,	FRZ)	SETUPS59
CALL SETUP('FREQU	' , 4 ,	FREQU)	SETUPS60
CALL SETUP('FREQF	' , 4 ,	FREQF)	SETUPS61
CALL SETUP('FTVB	' , 4 ,	FTVB , 64)	SETUPS62
CALL SETUP('FOVB	' , 4 ,	FOVB)	SETUPS63
CALL SETUP('HEAVE	' , 4 ,	HEAVE)	SETUPS64
CALL SETUP('HVOR	' , 4 ,	HVOR)	
CALL SETUP('INDV	' , 4 ,	INDV)	SETUPS65
CALL SETUP('LOWER	' , 4 ,	LOWER)	SETUPS66
CALL SETUP('MAXT	' , 4 ,	MAXT)	SETUPS67
CALL SFTUP('MCTR	' , 4 ,	MCTR)	SETUPS68
CALL SETUP('MSTOP	' , 4 ,	MSTOP)	SETUPS69
CALL SETUP('MVB	' , 4 ,	MVB , 64)	SETUPS70
CALL SETUP('NCOI	' , 4 ,	NCOI)	SETUPS71
CALL SETUP('NCORD	' , 4 ,	NCORD)	SETUPS72
CALL SETUP('NGAM	' , 4 ,	NGAM)	SETUPS73
CALL SETUP('NOFF	' , 4 ,	NOFF)	SETUPS74
CALL SFTUP('NOTBL	' , 4 ,	NOTBL)	SETUPS75
CALL SETUP('NOUJ	' , 4 ,	NOUJ)	
CALL SETUP('NSBL	' , 4 ,	NSBL)	SETUPS76
CALL SETUP('NSIG	' , 4 ,	NSIG)	SETUPS77
CALL SETUP('NVB	' , 4 ,	NVB)	SETUPS78
CALL SETUP('NVOR	' , 4 ,	NVOR)	
CALL SETUP('NY	' , 4 ,	NY)	SETUPS79
CALL SETUP('NZ	' , 4 ,	NZ)	SETUPS80
CALL SETUP('PHIH	' , 4 ,	PHIH)	SETUPS81
CALL SETUP('PLOTOP	' , 4 ,	PLOTOP)	
CALL SETUP('PSILOW	' , 4 ,	PSILOW)	
CALL SETUP('PSTUP	' , 4 ,	PSTUP)	
CALL SETUP('RVB	' , 4 ,	RVB , 64))	SETUPS82
CALL SETUP('ROBB	' , 4 ,	ROBB)	SETUPS83
CALL SETUP('REB	' , 4 ,	REB)	SETUPS84
CALL SETUP('RRDBR	' , 4 ,	RRDBR)	SETUPS85
CALL SETUP('ROVB	' , 4 ,	ROVB)	SETUPS86
CALL SETUP('RYI	' , 4 ,	RYI)	SETUPS87
CALL SETUP('SSPA	' , 4 ,	SSPA)	
CALL SFTUP('SVOR	' , 4 ,	SVOR)	
CALL SETUP('TEST	' , 4 ,	TEST)	SETUPS88
CALL SETUP('TORF	' , 4 ,	TORF)	
CALL SETUP('UPRIM	' , 4 ,	UPRIM)	SETUPS89
CALL SETUP('XIVOR	' , 4 ,	XIVOR)	
CALL SETUP('XL	' , 4 ,	XL , 30)	SETUPS90
CALL SETUP('XMVB	' , 4 ,	XMVB , 64)	SETUPS91

	CALL SETUP('XMUVR	' ,4, XMUVR)	SETUPS92
	CALL SETUP('XMUAVB	' ,4, XMJAVB)	SETUPS93
	CALL SETUP('XU	' ,4, XU, 30)	SETUPS94
	CALL SETUP('Y	' ,4, Y, 100)	SETUPS95
	CALL SETUP('YL	' ,4, YL, 30)	SETUPS96
	CALL SETUP('YU	' ,4, YU, 30)	SETUPS97
C				SETJPS98
C				SETUPS99
C				SETUP100
C				SETUP101
C				SETUP102
C				SETUP103
				SETUP104
	PSILOW=	1.E10		
	PSIUP=	-1.E10		
	PLOTOP =	1.		
	NOUT=	0		
	RETURN			SETUP105
C				SETUP106
	END			

```

SUBROUTINE BLC(X,Y,MST,MEND,NY,RY,DRY,DXI,REF,UPRIM,FLAM,XFLAM,TESBLC 1
IT,U,SCALE,UE,UC,V,XSEP,USEP,DISS,THETS,LOWER,LAMQ,MSEP,XC,USAV,SCARBLC 2
ILS,NITS,NTIME,NOTBL,XTEST,NZ,NOUT)
C
C PROGRAM FOR ANALYZING LAMINAR AND TURBULENT BOUNDARY LAYERS BLC 4
C BY THE METHOD OF FINITE DIFFERENCES. IF THE INTEGER LAMQ BLC 5
C IS GREATER THAN ZERO, THE BOUNDARY LAYER IS LAMINAR. BLC 6
C BLC 7
C BLC 8
COMMON /BL1/ NDUMMY,NDIMC,ISTD
DIMENSION USAV(300,100),SCALS(300) BLC 9
DIMENSION X(300),Y(100),UE(300,3),UC(100,3),V(100,2),XC(300) BLC 10
DIMENSION SO(100),SE(100),SF(100),VISC(100,2),GRAD(100) BLC 11
DIMENSION A(100),B(100),C(100),D(100),F(100) BLC 12
DIMENSION ALPHA(100),BETA(100),GAMMA(100),DELTA(100) BLC 13
DIMENSION SCALE(300,2),VAR1(100),VAR2(100) BLC 14
DIMENSION FLAM(10),XFLAM(10),YB1(100),YB2(100) BLC 15
DIMENSION U(300,100,2) BLC 16
DIMENSION CAPG(100),CAPH(100),CAPJ(100),CAPK(100) BLC 17
DOUBLE PRECISION AP(100),BP(100),CP(100),DP(100),FP(100),UP(100) BLC 18
10 FORMAT(1H1,41X,36H ANALYSIS OF LAMINAR BOUNDARY LAYER//51X,12HTIBLC 19
1ME STEP NOI3//51X,12HITERATION NOI3//4X,1HM,8X,1HX,13X,2HXC,12X,2BLC 20
1HUE,10X,6H-DP/DX,9X,5HDELTA,9X,5HDISPL,9X,5HTHETA,9X,5HSHEAR/) BLC 21
11 FORMAT(1H1,41X,36H ANALYSIS OF TURBULENT BOUNDARY LAYER//51X,12HTIBLC 22
1ME STEP NOI3//51X,12HITERATION NOI3//4X,1HM,8X,1HX,13X,2HXC,12X,2BLC 23
1HUE,10X,6H-DP/DX,9X,5HDELTA,9X,5HDISPL,9X,5HTHETA,9X,5HSHEAR,4X, BLC 24
3 'I'//)
12 FORMAT(I5,8F14.4,I3)
20 FORMAT(1H1,2X,3HM =I4//2X,3HX =E14.5//2X, 4HUE =E14.5,10X,17H-(1/RBLC 26
1H0)(DP/DX) =E14.5,10X,5HREB =E14.5,10X,4HUF =E14.5//) BLC 27
24 FORMAT(2X,25H PHYSICAL DELTA =E14.5,8X,12HDELTA STAR =E14.5 BLC 28
15,8X,7HTHETA =E14.5//2X,25H TRANSFORMED DELTA =E14.5,8X,12HDEBLC 29
1LTA STAR =E14.5,8X,7HTHETA =E14.5//) BLC 30
21 FORMAT(25X,1HV,19X,1HU,19X,1HV,16X,5H DU/DY,14X,6HNUE/NU/) BLC 31
22 FORMAT(10X,5E20.5) BLC 32
23 FORMAT(//30X,17HSEPARATION AT X =E13.5,6H, XC =E13.5) BLC 33
25 FORMAT(///40X,12HWALL SHEAR =E14.5//) BLC 34
30 FORMAT(//50X,17HTRANSITION AT X =E14.5) BLC 35
35 FORMAT(//20X,35HSCALE CHANGE - Y-MAX INCREASED FROME12.4,3H TOE12.4 BLC 36
I/) BLC 37
810 FORMAT(10X,7HAT STEPI3,22H, THE WALL GRADIENT ISEL2.4) BLC 38
BCON = 1.57DXI BLC 51
FCON = 1./(2.*DXI) BLC 52
IF(ISTD.NE.1) GO TO 900
DXI=1.E30
BCON=0.
FCON= 0.
900 CONTINUE
MOUT=6 BLC 39
MTRAN=-1 BLC 40
YSUB2=Y(2) BLC 41
MST2 = MST - 2 BLC 42
MST1=MST-1 BLC 43
NOUT1= NOUT +1
MST1MD= MOD( MST1, NOUT1)
MAXIT=0

```


	GO TO (543,550),LOWER	BLC	44
543	IF(LAMQ) 544,544,545	BLC	45
544	WRITE(MOUT,11) NTIME,NITS	BLC	46
	GO TO 550	BLC	47
545	WRITE(MOUT,10) NTIME,NITS	BLC	48
550	CONTINUE	BLC	49
	YTR = SQRT(REF)	BLC	50
	UC(1,1) = 0.	BLC	53
	V(1,1) = 0.	BLC	54
	NV = NY - 2	BLC	55
	NVM1 = NV - 1	BLC	56
	NVPI = NV + 1	BLC	57
	CALL YDIFF(NY,ALPHA,BETA,GAMMA,DELTA,SD,SE,SF,C2,C3,C4,Y)	BLC	58
	DO 41 N=1,NVPI	BLC	59
	VISC(N,1) = 1.	BLC	60
41	VISC(N,2) = 1.	BLC	61
	DO 42 M=MST2,MST1	BLC	62
	L = MST1-M+2	BLC	63
	DO 50 N=1,NV	BLC	64
50	GRAD(N+1) = SD(N+1)*UC(N+2,L)+SE(N+1)*UC(N+1,L)-SF(N+1)*UC(N,L)	BLC	65
	GRAD(1) = C2*UC(2,L)+C3*UC(3,L)+C4*UC(4,L)	BLC	66
	MM=M-1	BLC	67
	CALL PGRAD(MM,X,UE,DXI,PRESS,SA,SB,SC,SR,SS)	BLC	68
	DO 456 N=1,NY	BLC	69
456	UC(N,1)=UC(N,L)	BLC	70
	CALL SETIT(LAMQ,M,NV,REB,X,Y,UC,PRESS,GRAD,DELT,DISP,THETA,VISC,MTBL3	BLC	71
	IRAN)	BLC	72
42	CONTINUE	BLC	73
	MEND1 = MEND - 1	BLC	74
	GRADS=GRAD(1)	BLC	75
	GRADSS=GRAD(1)	BLC	76
C		BLC	77
C	THE MAIN CALCULATION STARTS HERE.	BLC	78
C		BLC	79
	DO 99 M=MST1,MEND1	BLC	80
	ITER=0	BLC	81
	WALLG=0.	BLC	82
	MPI=M+1	BLC	83
	DELTP = DELT/YTR	BLC	84
	DISPT = DISP*YTR	BLC	85
	THETT = THETA*YTR	BLC	86
	SHEAR = GRAD(1)/YTR	BLC	87
	IF(MOD(M, NOUT1).NE. MSTIMD) GO TO 225		
	GO TO (561,562),LOWER	BLC	88
561	WRITE(MOUT,12) M,X(M),XC(M),UE(M,1),PRESS,DELTP,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) DELTP,DISP,THETA,DELT,DISPT,THETT	BLC	92
	WRITE(MOUT,21)	BLC	93
	WRITE(MOUT,22) (Y(N),UC(N,2),V(N,1),GRAD(N),VISC(N,1),N=1,NVPI)	BLC	94
	WRITE(MOUT,25) SHEAR	BLC	95
225	IF(GRADSS-GRADS-I.E-6) 229,229,408	BLC	96
408	XSX=X(M-2)+(X(M-1)-X(M-2))*GRADSS/(GRADSS-GRADS)	BLC	97
	IF(XSX-X(M)) 409,409,229	BLC	98

409	WFS=(X _{SX} -X(M-1))/(X(M)-X(M-1))	BLC	99
	GO TO 224	BLC	100
229	IF (GRAD(1)) 227, 227, 273		
273	IF (DISP .GT. 0. .AND. THETA .GT. 0.) GO TO 223		
283	CONTINUE		
	XSEP= XC(M-1)		
	USEP=UE(M-1,1)		
	XRL=X(M-1)		
	WRITE(MOUT,23) XBL, XSEP		
	RETURN		
227	WFS=GRADS/(GRADS-GRAD(1))	BLC	102
224	WFS1=1.-WFS	BLC	103
	XSEP=WFS1*XC(M-1)+WFS*XC(M)	BLC	104
	XBL=WFS1*X(M-1)+WFS*X(M)	BLC	105
	USEP=WFS1*UE(M-1,1)+WFS*UE(M,1)	BLC	106
	WFP=(XRL-X(M-2))/(X(M-1)-X(M-2))	BLC	107
	WFP1=1.-WFP	BLC	108
	DISS=DISSS*WFP1+DISS*WFP	BLC	109
	THETS=THETSS*WFP1+THETS*WFP	BLC	110
	WRITE(MOUT,23) XBL,XSEP	BLC	111
	IF(LAMQ.EQ.0.AND.M.LT.MTRAN+5) LAMQ=1	BLC	112
	GO TO 222	BLC	113
223	CONTINUE	BLC	114
	IF(NOTBL .EQ. 2 .AND. NITS .GT. 1 .AND. M.GT. NZ .AND.		
1	XC(M) .GT. XTEST) GO TO 283		
	IF(LAMQ) 801,801,802	BLC	115
802	IF(NOTBL .EQ. 2) GO TO 801		
	CALL TRANS(UPRIM,PRESS,THETA,REB,UC,NY,FLAM,XFLAM,LAMQ)	BLC	116
	IF(LAMQ) 805,805,801	BLC	117
805	WRITE(MOUT,30) X(M)	BLC	118
	MTRAN = M+1	BLC	119
801	CONTINUE	BLC	120
	IF(Y(NV)-DELT) 620,641,641	BLC	121
620	RY=RY+DRY	BLC	122
C		BLC	123
C	RESCALING CALCULATION STARTS HERE.	BLC	124
C		BLC	125
	DO 632 N=1,NY	BLC	126
	YBI(N) = Y(N)	BLC	127
	VARI(N) = UC(N,2)	BLC	128
632	VAR2(N) = UC(N,3)	BLC	129
	CALL YSET(RY,YSUB2,NY,Y)	BLC	130
	WRITE(MOUT,35) YBI(NY),Y(NY)	BLC	131
	DO 633 N=2,NVPI	BLC	132
	YIN = Y(N)	BLC	133
	CALL TERP(YIN,YBI,VARI,NY,UPAS1)	BLC	134
	UC(N,2) = UPAS1	BLC	135
	CALL TERP(YIN,YBI,VAR2,NY,UPAS2)	BLC	136
633	UC(N,3) = UPAS2	BLC	137
	CALL YDIFF(NY,ALPHA,BETA,GAMMA,DELTA,SD,SE,SF,C2,C3,C4,Y)	BLC	138
	IF(LAMQ) 700,700,701	BLC	139
700	DO 635 N=2,NVPI	BLC	140
	VARI(N) = VISC(N,1)	BLC	141
635	VAR2(N) = VISC(N,2)	BLC	142
	DO 636 N=2,NVPI	BLC	143

	YIN = Y(N)	BLC	144
	CALL TERP(YIN,YB1,VARI,NVPI,UPAS1)	BLC	145
	VISC(N,1) = UPAS1	BLC	146
	CALL TERP(YIN,YB1,VAR2,NVPI,UPAS2)	BLC	147
636	VISC(N,2) = UPAS2	BLC	148
7C1	DO 637 N=2,NVPI	BLC	149
	VARI(N) = V(N,1)	BLC	150
637	VAR2(N) = V(N,2)	BLC	151
	DO 638 N=2,NVPI	BLC	152
	YIN = Y(N)	BLC	153
	CALL TERP(YIN,YB1,VARI,NVPI,UPAS1)	BLC	154
	V(N,1) = UPAS1	BLC	155
	CALL TERP(YIN,YB1,VAR2,NVPI,UPAS2)	BLC	156
638	V(N,2) = UPAS2	BLC	157
641	CONTINUE	BLC	158
C		BLC	159
C	RESCALING CALCULATION ENDS HERE.	BLC	160
C		BLC	161
	CALL PGRAD(M,X,UE,DXI,PRESS,SA,SB,SC,SR,SS)	BLC	162
C		BLC	163
C	RECURSION RELATIONS ARE SET UP HERE.	BLC	164
C		BLC	165
	IF (ISTD.EQ. 1) GO TO 820		
	IF(SCALE(M+1,1)-1.) 522,522,521	BLC	166
521	IF(SCALE(M+1,2)-1.) 522,522,523	BLC	167
522	LACKU=1	BLC	168
	FACU1=UE(M+1,2)/UE(M+1,1)	BLC	169
	FACU2=UE(M+1,3)/UE(M+1,1)	BLC	170
	GO TO 820	BLC	171
523	LACKU=2	BLC	172
	DO 610 NN=1,NY	BLC	173
	VARI(NN) = U(M+1,NN,1)	BLC	174
610	VAR2(NN) = U(M+1,NN,2)	BLC	175
	CALL YSET(SCALE(M+1,1),YSUB2,NY,YB1)	BLC	176
	CALL YSET(SCALE(M+1,2),YSUB2,NY,YB2)	BLC	177
820	DO 88 N=2,NV	BLC	178
	CALL CAPS(ITER,N,CAPG,CAPH,CAPJ,CAPK,SR,SS,SD,SE,SF,VISC,V,UC)	BLC	179
	A(N)=-SF(N)*CAPG(N)-DELTA(N)*CAPH(N)+SF(N)*CAPJ(N)	BLC	180
	B(N)=BCON+SA*CAPK(N)+SF(N)*CAPG(N)-GAMMA(N)*CAPH(N)-SE(N)*CAPJ(N)	BLC	181
	C(N)=SD(N)*CAPG(N)-BETA(N)*CAPH(N)-SD(N)*CAPJ(N)	BLC	182
	D(N) = -ALPHA(N)*CAPH(N)	BLC	183
	IF (ISTD .EQ. 1) GO TO 576		
	GO TO (574,575),LACKU	BLC	184
574	UPAS1=FACU1*UC(N,1)	BLC	185
	UPAS2=FACU2*UC(N,1)	BLC	186
	GO TO 576	BLC	187
575	YIN = Y(N)	BLC	188
	CALL TERP(YIN,YB1,VARI,NY,UPAS1)	BLC	189
	CALL TERP(YIN,YB2,VAR2,NY,UPAS2)	BLC	190
576	F(N) = PRESS+FCON*(4.*UPAS1-UPAS2)+CAPK(N)*(SB*UC(N,2)-SC*UC(N,3))	BLC	191
88	CONTINUE	BLC	192
C		BLC	193
C	SOLUTION FOR VELOCITY PROFILE STARTS HERE.	BLC	194
C		BLC	195
	DO 89 N=2,NV	BLC	196

	AP(N) = A(N)	BLC	197
	BP(N) = B(N)	BLC	198
	CP(N) = C(N)	BLC	199
	DP(N) = D(N)	BLC	200
89	FP(N) = F(N)	BLC	201
	DO 77 N=2,NVMI	BLC	202
	CP(N) = CP(N)/BP(N)	BLC	203
	DP(N) = DP(N)/BP(N)	BLC	204
	FP(N) = FP(N)/BP(N)	BLC	205
	BP(N+1) = BP(N+1) - CP(N)*AP(N+1)	BLC	206
	CP(N+1) = CP(N+1) - DP(N)*AP(N+1)	BLC	207
77	FP(N+1) = FP(N+1) - FP(N)*AP(N+1)	BLC	208
	UP(NY) = UE(M+1,1)	BLC	209
	UP(NVP1) = UP(NY)	BLC	210
	UP(NV) = (FP(NV)-UP(NY)*(DP(NV) + CP(NV)))/BP(NV)	BLC	211
	DO 66 N=3,NV	BLC	212
	NN=NV+2-N	BLC	213
66	UP(NN) = FP(NN) - DP(NN)*UP(NN+2) - CP(NN)*UP(NN+1)	BLC	214
	DO 65 N=2,NY	BLC	215
65	UC(N,1) = UP(N)	BLC	216
	IF(ITER) 843,841,843	BLC	217
841	DO 842 N=2,NVPI	BLC	218
	V(N,2) = V(N,1)	BLC	219
842	VISC(N,2)=VISC(N,1)	BLC	220
	DISS=DISS	BLC	221
	DISS=DISP	BLC	222
	THETSS=THETS	BLC	223
	THETS=THETA	BLC	224
	GRADSS=GRADS	BLC	225
	GRADS=GRAD(1)	BLC	226
843	DO 55 N=2,NVPI	BLC	227
55	V(N,1) = V(N-1,1) - .5*(Y(N)-Y(N-1))*(SA*(UC(N,1)+UC(N-1,1)) - SB*(UC(N,2)+UC(N-1,2)) + SC*(UC(N,3)+UC(N-1,3)))	BLC	228
	DO 56 N=1,NV	BLC	229
56	GRAD(N+1) = SD(N+1)*UC(N+2,1)+SE(N+1)*UC(N+1,1)-SF(N+1)*UC(N,1)	BLC	231
	GRAD(1) = C2*UC(2,1)+C3*UC(3,1)+C4*UC(4,1)	BLC	232
	CALL SFTIT(LAMQ,MPI,NV,REB,X,Y,UC,PRESS,GRAD,DELT,DISP,THETA,VISC,IMTRAN)	BLC	233
	ITER=ITER+1	BLC	234
	GO TO (830,809),LOWER	BLC	235
809	WRITE(MOUT,810) ITER,GRAD(1)	BLC	236
830	IF(ITER-9) 811,811,812	BLC	237
811	EPW=ABS(GRAD(1)-WALLG)	BLC	238
	IF(WALLG-1.) 120,120,119	BLC	239
119	EPW=EPW/WALLG	BLC	240
120	IF(EPW-TEST) 812,814,814	BLC	241
814	WALLG=GRAD(1)	BLC	242
	GO TO 820	BLC	243
820	DO 44 N=1,NY	BLC	244
	UC(N,3) = UC(N,2)	BLC	245
	UC(N,2) = UC(N,1)	BLC	246
44	CONTINUE	BLC	247
	MAXIT=ITER		
	IF(ISTD .EQ. 1) GO TO 99		
	DO 48 N=1,NY		

48	USAV(M+1,N)=UC(N,1)	
	SCALS(M+1)=RY	BLC 249
99	CONTINUE	BLC 250
	XSEP=1.1	BLC 251
	USEP=UE(MX,1)	BLC 252
222	CONTINUE	BLC 253
	RETURN	BLC 254
	END	

```

SUBROUTINE PLOTSB( PLOTOP , P , L )
  REAL * 8 ORD(6)
  DIMENSION P(200,7), TIT1(56) , NF(5,4)
1 , NFP(6)
  DATA N1 , N2 , NO , N42
1 / 1 , 2 , 0 , 42 /
  DATA ORD/ ' THETA-P' , ' TORS ' , ' FLAP-H ' , ' BEND-H ' ,
1 ' CL ' , ' CM-A ' /
  IF (PLOTOP .EQ. 0.) RETURN
  IF ( L .LT. 2) RETURN
  IF ( PLOTOP .EQ. 2.) GO TO 2
  PLOTOP = 2.
  CALL IOFRMV ( 'CRIMI -PETE ' , '30' , '5100' )
2 CONTINUE
3 NL=1
  DO 1 J = 1, 6
  CALL EZPLOT(9. , N1 , N1, P , P(1,J+1), L , -N1 , N2
1 , N42 , 1 , ' ' , 12 , ' PSI-DEGREES' , 8 , ORD( J)
2 , N1 , N1 , XL , XU , N1 , YL , YU ,N1 , NO , NL)
1 CONTINUE
  NFP(1)= -1
  NFP(2)= 66
  NFP(3)= 50
  NFP(4)= 50
  NFP(5)= 680
  CALL EZPLOT(9. , N1 , N1, P , P(1,2 ) , L , -N1 , N2
1 , N42 , 1 , ' ' , 12 , ' PSI-DEGREES' , 8 , ORD( 1)
2 , NFP , N1 , XL , XU , N1 , YL , YU ,N1 , NO , N1)
  NFP(1)= -2
  NFP(2)= 66
  NFP(4)= 350
  NFP(5)= 380
  CALL EZPLOT(9. , N1 , N1, P , P(1,6 ) , L , -N1 , N2
1 , N42 , 1 , ' ' , 12 , ' ' , 8 , ORD( 5)
2 , NFP , N1 , XL , XU , N1 , YL , YU ,N1 , NO , N1)
  NFP(2)= 50
  NFP(4)= 690
  NFP(5)= 40
  CALL EZPLOT(9. , N1 , N1, P , P(1,7 ) , L , -N1 , N2
1 , N42 , 1 , ' ' , 12 , ' ' , 8 , ORD( 6)
2 , NFP , N1 , XL , XU , N1 , YL , YU ,N1 , NO , N1)
  NFP(1)=-1
  NFP(2)= 50
  NFP(3)=50
  NFP(4)=50
  NFP(5)= 690
  CALL EZPLOT(9. , N1 , N1, P , P(1,3 ) , L , -N1 , N2
1 , N42 , 1 , ' ' , 12 , ' PSI-DEGREES' , 8 , ORD( 2)
2 , NFP , N1 , XL , XU , N1 , YL , YU ,N1 , NO , N1)
  NFP(1)=-2
  NFP(2)= 66
  NFP(4)= 350
  NFP(5)= 380
  CALL EZPLOT(9. , N1 , N1, P , P(1,4 ) , L , -N1 , N2
1 , N42 , 1 , ' ' , 12 , ' ' , 8 , ORD( 3)

```

```

2 , NFP , N1 , XL , XU , N1 , YL , YU ,N1, NO, N1)
  NFP(2)= 50
  NFP(4)= 690
  NFP(5)= 40
CALL EZPLOT(9. , N1 , N1, P , P(1,5 ) , L , -N1 , N2
1 , N42 , 1 , ' ' , 12 , ' ' , 8 , ORD( 4)
2 , NFP , N1 , XL , XU , N1 , YL , YU ,N1, NO, N1)
  RETURN
  END

```

```

SUBROUTINE STAG(MX,NY,MSTOP,MST,DXI,RY,DRY,X,Y,UE,UC,V,USAV,SCALS,STAG 1
  ISEF) STAG 2
C PROGRAM FOR CALCULATING THE BOUNDARY LAYER PROFILE NEAR STAG 3
C THE STAGNATION POINT STAG 4
C STAG 5

COMMON /BL1/ NTIME,NDIMC,ISTD
DIMENSION USAV(300,100),SCALS(300) STAG 6
DIMENSION PHI7(24),PHIP(24),FTAP(24) STAG 7
DIMENSION X(300),Y(100),UE(300,3),UC(100,3),V(100,2) STAG 8
DIMENSION EF(100),EFP(100) STAG 9
DATA FTAP /0.,.2,.4,.6,.8,1.,1.2,1.4,1.6,1.8,2.,2.2,2.4,2.6,2.8,3. STAG 10
1,3.2,3.4,3.6,3.8,4.,4.2,4.4,4.6/ STAG 11
DATA PHIZ /0.,.0233,.0881,.1867,.3124,.4592,.622,.7967,.9793,1.168 STAG 12
19,1.362,1.5578,1.7553,1.9538,2.153,2.3526,2.5523,2.7522,2.9521,3.1 STAG 13
1521,3.3521,3.5521,3.7521,3.9521/ STAG 14
DATA PHIP /0.,.2265,.4145,.5663,.6859,.7779,.8467,.8968,.9323,.956 STAG 15
18,.9732,.9839,.9905,.9946,.997,.9984,.9992,.9996,.9998,.9999,1.,1. STAG 16
1,1.,1./ STAG 17
BAG=.08 STAG 18
IF(ISEF) 10,10,5 STAG 19
5 BAG=.5 STAG 20
10 EF(1) = 0. STAG 21
EFP(1) = 0. STAG 22
DO 20 M=1,MX STAG 23
IF(UE(M,1)) 20,20,19 STAG 24
15 MSP = M STAG 25
GO TO 21 STAG 26
20 CONTINUE STAG 27
21 ASTAG = (UE(MSP+2,1)-UE(MSP+1,1))/(X(MSP+2)-X(MSP+1)) STAG 28
IF(ASTAG) 22,22,23 STAG 29
22 ASTAG=(UE(MSP,1)-UE(MSP-1,1))/(X(MSP)-X(MSP-1)) STAG 30
23 SQAS = SQRT(ASTAG) STAG 31
DELT = 2.6/SQAS STAG 32
309 IF(DELT-Y(NY-3)) 311,310,310 STAG 33
310 RY=RY+DRY STAG 34
CALL YSET(RY,Y(2),NY,Y) STAG 35
GO TO 309 STAG 36
311 CONTINUE STAG 37
DO 80 N=2,NY STAG 38
YET = Y(N)*SQAS STAG 39
DO 33 NN=1,24 STAG 40
IF(YET-ETAP(NN)) 408,408,33 STAG 41
400 MARK = NN STAG 42
GO TO 410 STAG 43
33 CONTINUE STAG 44
EF(N) = YET-.6479 STAG 45
EFP(N) = 1. STAG 46
GO TO 80 STAG 47
410 FRACT = (YET-ETAP(MARK-1))/(ETAP(MARK)-ETAP(MARK-1)) STAG 48
FRAC1 = 1.-FRACT STAG 49
EF(N) = PHIZ(MARK-1)*FRAC1+PHIZ(MARK)*FRACT STAG 50
EFP(N) = PHIP(MARK-1)*FRAC1+PHIP(MARK)*FRACT STAG 51
80 CONTINUE STAG 52
M1 = MSP-MSTOP STAG 53
M2 = MSP+MSTOP STAG 54

```


	M=M1-1	STAG	55
50	M=M+1	STAG	56
	MST=M+1	STAG	57
	SCALS(M)=RY	STAG	58
	DO 71 N=1,NY	STAG	59
	UC(N,3) = UC(N,2)	STAG	60
	UC(N,2) = UE(M,1)*EFP(N)	STAG	61
	V(N,2) = V(N,1)	STAG	62
	V(N,1) = -SQAS*EF(N)	STAG	63
	IF(ISTD .EQ. 1) GO TO 71		
	USAV(M,N)=UC(N,2)	STAG	64
71	CONTINUE	STAG	65
	IF(M-M2) 50,55,55	STAG	66
55	IF(UF(M,1)-BAG) 50,50,81	STAG	67
81	CONTINUE	STAG	68
	RETURN	STAG	69
	END	STAG	70

```

SUBROUTINE ATTPR(PREC,XSIG,NSIG,ASZ,AS,AR,CMAT,RMAT,NGAM,NF,ACAP,TATTPR 1
THICK,RDBB,GAMAW,UINF,UDOT,DXI,BCAP) ATTPR 2
DIMENSION XSIG(100),ASZ(30),AS(30,30),AR(30),BCAP(100,3) ATTPR 3
DIMENSION ACAP(30,3),THICK(24),GAMAW(1000) ATTPR 4
DOUBLE PRECISION CMAT(60,60),RMAT(130) ATTPR 5
PI=3.14159 ATTPR 6
NGPI=NGAM+1 ATTPR 7
DO 50 M=1,NGPI 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(NGPI,CMAT,RMAT) ATTPR 14
DO 75 M=1,NGPI ATTPR 15
75 ACAP(M,1)=RMAT(M) ATTPR 16
GAMAW(1)=GAMI(ACAP,DXI,PI) 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,RATTPR 20
10BB,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,MX,NZ,NUNPOP	1
	IF,XSIG,BCAP,THICK,RDBB,UINF,XC,UF)	UNPJP 2
	DIMENSION AR(30),ALAM(30),XGAM(30),AS(30,30),ACAP(30,3),XSIG(100),UNPOP	3
	BCAP(100,3),THICK(24),XC(300),JE(300,3)	UNPOP 4
	DOUBLE PRECISION RMAT(130),CMAT(60,60)	UNPOP 5
	NGPI=NGAM+1	UNPOP 6
	DO 5 M=1,NGPI	UNPOP 7
	SUB=AR(M)-ALAM(M)*AFACT/3.	UNPJP 8
	RMAT(M)=SUB	UNPJP 9
	CMAT(M,1)=1.	UNPOP 10
	CMAT(M,2)=XGAM(M)	UNPOP 11
	DO 5 N=2,NGAM	UNPOP 12
5	CMAT(M,N+1)=AS(M,N)	UNPOP 13
	CALL ALSOL(NGPI,CMAT,RMAT)	UNPOP 14
	DO 10 N=1,NGPI	UNPJP 15
10	ACAP(N,1)=RMAT(N)	UNPOP 16
	DO 15 M=1,MX	UNPOP 17
	SIGN=1.	UNPOP 18
	IF(M-NZ) 12,14,14	UNPOP 19
12	SIGN=-SIGN	UNPJP 20
14	CALL QECAL(0,NGAM,NGAM,NF,XSIG,ACAP,BCAP,THICK,RDBB,O.,UINF,XC(M),UNPJP	21
	IUE(M,1),SIGN)	UNPJP 22
15	CONTINUE	UNPOP 23
	RETURN	UNPOP 24
	END	UNPOP 25

```

SUBROUTINE ALSOL(NT, C, R)
  DCUBLE PRECISION C, NDIMC, NOIMC), R(130)
  DOUBLE PRECISION CMAX,SAVE,SUM
  COMMON /BL1/ NTIME, NOIMC
  NT1 = NT-1
  DO 99 J=1,NT1
    CMAX = C(NT,J)
    L=NT
    DO 10 I=J,NT1
      IF (DABS(CMAX)-DABS(C(I,J))) 5,10,10
5     CMAX = C(I,J)
      L=I
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
    JP1 = J+1
    DO 25 I=JP1,NT
      DO 20 JJ=JP1,NT
20     C(I,JJ) = C(I,JJ) - C(I,J)*C(J,JJ)
25     R(I) = R(I) - R(J)*C(I,J)
99    CONTINUE
    R(NT) = R(NT)/C(NT,NT)
    DO 150 K=1,NT1
      I=NT-K
      IP1 = I+1
      SUM = 0.
      DO 125 J=IP1,NT
125     SUM = SUM + R(J)*C(I,J)
150     R(I) = R(I) - SUM
    RETURN
  END

```

ALSOL

```

SUBROUTINE CPC(ISEP,NGAM,NF,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,CPC 1
1,BCAP,THICK,RDBB,GAMAW,UINF,UDOT,SIGN,XC,DXI,CP) CPC 2
DIMENSION XSIG(100),XSIGA(100),XSIGB(100),ACAP(30,3),BCAP(100,3) CPC 3
DIMENSION GAMAW(1000),THICK(24) CPC 4
THETA=ARCT(XC) CPC 5
RECIP=1./(UINF*UINF) CPC 6
SUM=0. CPC 7
ANGLE=0. CPC 8
DO 5 N=1,NF CPC 9
ANGLE=ANGLE+THETA CPC 10
5 SUM=SUM+THICK(N)*COS(ANGLE) CPC 11
CP=UDOT*RECIP*(THICK(1)+2.*(1.-XC)*SUM) CPC 12
CALL DECAL(ISEP,NGAM,NSIG,NF,XSIG,ACAP,BCAP,THICK,RDBB,GAMAW(1),UICPC 13
INF,XC,U,SIGN) CPC 14
CP=CP+2.*(SIGN*U/UINF-1.) CPC 15
CALL EGAMI(1,NGAM,ACAP,BCAP(1,1),XSIG(1),XSIG(NSIG+1),GAMAW(1),XC,CPC 16
IVAL1) CPC 17
CALL EGAMI(2,NGAM,ACAP,BCAP(1,2),XSIGA(1),XSIGA(NSIGA+1),GAMAW(2),CPC 18
IXC,VAL2) CPC 19
CALL EGAMI(3,NGAM,ACAP,BCAP(1,3),XSIGB(1),XSIGB(NSIGB+1),GAMAW(3),CPC 20
IXC,VAL3) CPC 21
CP=CP+SIGN*RECIP*(1.5*VAL1-2.*VAL2+.5*VAL3)/DXI CPC 22
IF(ISEP) 20,20,10 CPC 23
10 CALL FSI(1,NSIG,XSIG,BCAP,XC,VAL1) CPC 24
CALL ESI(2,NSIGA,XSIGA,BCAP,XC,VAL2) CPC 25
CALL ESI(3,NSIGB,XSIGB,BCAP,XC,VAL3) CPC 26
CP=CP+RECIP*(1.5*VAL1-2.*VAL2+.5*VAL3)/DXI CPC 27
20 CP=-CP CPC 28
RETURN CPC 29
END

```

```

SUBROUTINE CLCM(NCOI,ISEP,NGAM,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACLCM 1
ICAP,BCAP,THICK,RDBB,GAMAW,UINF,UDOT,DXI,AROT,CMPA) CLCM 2
COMMON /CLCMBL/ CLVB,CMVB, CMPAVB MAIN
DIMENSION ARGL(21),ARGM(21) CLCM 3
DIMENSION GAMAW(1000),THICK(24) CLCM 4
DIMENSION XSIG(100),XSIGA(100),XSIGB(100),ACAP(30,3),BCAP(100,3) CLCM 5
4 FORMAT(/740X,4HCL =E13.5/40X,4HCM =E13.5,17H (ABOUT MIDCHDR)/40X,CLCM 6
14HCM =E13.5,24H (ABOUT PITCH AXIS - A =F7.4,1H) CLCM 7
MOUT=6 CLCM 8
SAVE=THICK(1) CLCM 9
THICK(1)=0. CLCM 10
DT=3.14159/FLOAT(NCOI) CLCM 11
CL=0. CLCM 12
CM=0. CLCM 13
XI=-1. CLCM 14
ANGLE=0. CLCM 15
FLI=0. CLCM 16
FMI=0. CLCM 17
IF (ISEP) 5,5,7 CLCM 18
7 XATT=XSIG(NSIG+1) CLCM 19
IF (XATT-.95) 8,5,5 CLCM 20
8 XAQ=XATT+.5,F-4 CLCM 21
XAP=XAQ+.025 CLCM 22
C1=-.5*(1.+XATT) CLCM 23
C2=C1+XATT CLCM 24
C1P=.5*(1.-XAP) CLCM 25
C2P=C1P+XAP CLCM 26
DO 10 I=1,NCOI CLCM 27
ANGLE=ANGLE+DT CLCM 28
XIPI=C1*COS(ANGLE)+C2 CLCM 29
CALL CPC(ISEP,NGAM,1,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,BCAP,TCCLCM 30
1THICK,RDBB,GAMAW,UINF,UDOT,1.0,XIPI,DXI,CPU) CLCM 31
CALL CPC(ISEP,NGAM,1,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,BCAP,TCCLCM 32
1THICK,RDBB,GAMAW,UINF,UDOT,-1.,XIPI,DXI,CPL) CLCM 33
FLIPI=CPL-CPU CLCM 34
FMIPI=XIPI*FLIPI CLCM 35
CL=CL+(XIPI-XI)*(FLIPI+FLI) CLCM 36
CM=CM+(XIPI-XI)*(FMIPI+FMI) CLCM 37
XI=XIPI CLCM 38
FLI=FLIPI CLCM 39
10 FMI=FMIPI CLCM 40
XI=1. CLCM 41
FLI=0. CLCM 42
FMI=0. CLCM 43
ANGLE=0. CLCM 44
DO 15 I=1,NCOI CLCM 45
ANGLE=ANGLE+DT CLCM 46
XIPI=C1P*COS(ANGLE)+C2P CLCM 47
CALL CPC(ISEP,NGAM,1,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,BCAP,TCCLCM 48
1THICK,RDBB,GAMAW,UINF,UDOT,1.0,XIPI,DXI,CPU) CLCM 49
CALL CPC(ISEP,NGAM,1,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,BCAP,TCCLCM 50
1THICK,RDBB,GAMAW,UINF,UDOT,-1.,XIPI,DXI,CPL) CLCM 51
FLIPI=CPL-CPU CLCM 52
FMIPI=XIPI*FLIPI CLCM 53
CL=CL-(XIPI-XI)*(FLIPI+FLI) CLCM 54

```

	CM=CM-(XIPI-XI)*(FMIPI+FMI)	CLCM	55
	XI=XIPI	CLCM	56
	FLI=FLIPI	CLCM	57
15	FMI=FMIPI	CLCM	58
	XIPI=XAQ	CLCM	59
	DO 16 I=1,21	CLCM	60
	CALL CPC(I SEP,NGAM,1,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,BCAP,TCLCM	CLCM	61
	IHICK,RDBB,GAMAW,UINF,UDDT,1.0,XIPI,DXI,CPU)	CLCM	62
	CALL CPC(I SEP,NGAM,1,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,BCAP,TCLCM	CLCM	63
	IHICK,RDBB,GAMAW,UINF,UDDT,-1.,XIPI,DXI,CPL)	CLCM	64
	ARGL(I)=CPL-CPU	CLCM	65
	ARGM(I)=XIPI*ARGL(I)	CLCM	66
16	XIPI=XIPI+.00125	CLCM	67
	SUML=0.	CLCM	68
	SUMM=0.	CLCM	69
	DO 17 I=1,19,2	CLCM	70
	SUML=SUML+2.*ARGL(I)+4.*ARGL(I+1)	CLCM	71
17	SUMM=SUMM+2.*ARGM(I)+4.*ARGM(I+1)	CLCM	72
	CL=CL+0.833333E-3*(SUML+ARGL(21)-ARGL(1))	CLCM	73
	CM=CM+0.833333E-3*(SUMM+ARGM(21)-ARGM(1))	CLCM	74
	BCON=16.*BCAP(1,1)*SQRT(5.E-4*(XATT-XSIG(1)))/UINF	CLCM	75
	CL=CL+BCON	CLCM	76
	CM=CM+XATT*BCON	CLCM	77
	GO TO 100	CLCM	78
5	DO 99 I=1,NCOT	CLCM	79
	ANGLE=ANGLE+DT	CLCM	80
	XIPI=-COS(ANGLE)	CLCM	81
	CALL CPC(I SEP,NGAM,1,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,BCAP,TCLCM	CLCM	82
	IHICK,RDBB,GAMAW,UINF,UDDT,1.0,XIPI,DXI,CPU)	CLCM	83
	CALL CPC(I SEP,NGAM,1,XSIG,NSIG,XSIGA,NSIGA,XSIGB,NSIGB,ACAP,BCAP,TCLCM	CLCM	84
	IHICK,RDBB,GAMAW,UINF,UDDT,-1.,XIPI,DXI,CPL)	CLCM	85
	FLIPI=CPL-CPU	CLCM	86
	FMIPI=XIPI*FLIPI	CLCM	87
	CL=CL+(XIPI-XI)*(FLIPI+FLI)	CLCM	88
	CM=CM+(XIPI-XI)*(FMIPI+FMI)	CLCM	89
	XI=XIPI	CLCM	90
	FLI=FLIPI	CLCM	91
99	FMI=FMIPI	CLCM	92
100	CL=.25*CL	CLCM	93
	CM=-.125*CM	CLCM	94
	CMPA=CM+AROT*CL*.5	CLCM	95
	WRITE(MOUT,4) CL,CM,CMPA,AROT	CLCM	96
	THICK(1)=SAVE	CLCM	97
	CLVB = CL	MAIN	
	CMVB = CM	MAIN	
	CMPAVB = CMPA	MAIN	
	RETURN	CLCM	98
	END		

```

SUBROUTINE QECAL (ISEP,NGAM,NSIG,NF,XSIG,ACAP,BCAP,THICK,RDBB,GAMMA,QECAL  1
1,UINF,XC,U,SIGN) QECAL  2
DIMENSION ACAP(30,3),BCAP(100,3),XSIG(100) QECAL  3
DIMENSION THICK(24) QECAL  4
EPS=1.E-6 QECAL  5
CORR=.707107/(1.-.63662*SQRT(RDBB)+.25*RDBB) QECAL  6
SINT=SQRT(1.-XC*XC) QECAL  7
THETA=ARCT(XC) QECAL  8
COUNT=0. QECAL  9
SUM=0. QECAL 10
SINT2=SIN(.5*THETA) QECAL 11
COST2=COS(.5*THETA) QECAL 12
IF(SINT -EPS) 4,6,6 QECAL 13
4 FACT=THETA*.5 QECAL 14
GO TO 8 QECAL 15
6 FACT=(1.-XC)/SINT QECAL 16
8 DO 10 N=1,NF QECAL 17
COUNT=COUNT+1. QECAL 18
ANGLE=THETA*COUNT QECAL 19
SUM=SUM+THICK(N)*(COUNT*FACT*SIN(ANGLE)-COS(ANGLE)) QECAL 20
10 CONTINUE QECAL 21
U=2.*SIGN*UINF*COST2*SUM+ACAP(1,1)*SINT2+.25*COST2*(1.+XC)*(3.*XC- QECAL 22
11.)*GAMMA QECAL 23
SUM=0. QECAL 24
ANGLE=0. QECAL 25
DO 12 N=1,NGAM QECAL 26
ANGLE=ANGLE+THETA QECAL 27
12 SUM=SUM+ACAP(N+1,1)*SIN(ANGLE) QECAL 28
U=U+COST2*SUM QECAL 29
IF(ISEP) 25,99,25 QECAL 30
25 SUM=0. QECAL 31
XSEP=XSIG(1) QECAL 32
XATT=XSIG(NSIG+1) QECAL 33
DO 40 N=2,NSIG QECAL 34
40 SUM=SUM+BCAP(N,1)*FB(XSIG(N-1),XSIG(N),XSIG(N+1),XC) QECAL 35
IF(XC-XATT-EPS) 45,45,46 QECAL 36
46 FACT=(1.-XATT)**(-1.5)*SQRT((XATT-XSEP)/(1.-XC)/(XC-XATT))*(1.+3.* QECAL 37
1XATT-4.*XC)-SIGN*(1.-SQRT((XSEP-XC)/(XATT-XC))) QECAL 38
GO TO 55 QECAL 39
45 IF(XSEP-XC) 49,49,48 QECAL 40
48 FACT=-SIGN*(1.-SQRT((XSEP-XC)/(XATT-XC))) QECAL 41
GO TO 55 QECAL 42
49 FACT=-SIGN QECAL 43
55 U=U+COST2*(BCAP(1,1)*FACT+SIGN*SJM) QECAL 44
99 U=(SIGN*UINF*SQRT(1.+XC)+ CORR*U)/SQRT(1.+XC+.5*RDBB) QECAL 45
RETURN QECAL 46
END

```


	SLBRoutine YVR(Y, I)	YVR	1
	REAL Y(10)	YVR	2
	REAL MVR	YVR	3
	COMMON /INPTVR/ FTVB(64), FPVB(64), FPPRVB(64), DIDRVB(64),	YVB	4
A	XMVB(64), DELVB, XMUVB, FOVB, XMUAVR,	YVB	5
B	ATOVB, ATCVB, ATSVB, ROVB, RVB(64),	YVB	6
C	MVB(64), NVB	YVR	7
	Y(1) = (RVB(I) - DELVB)**2 * MVB(I)	YVB	8
	Y(2) = FPVB(I)**2 * MVB(I)	YVB	9
	Y(3) = FTVB(I)**2 * DIDRVB(I)	YVB	10
	Y(4) = (DELVB - RVB(I)) * FTVB(I) * XMVB(I) * MVR(I)	YVB	11
	Y(5) = FPVB(I) * FTVB(I) * XMVB(I) * MVR(I)	YVB	12
	Y(6) = RVB(I) * (DELVB - RVB(I)) * MVB(I)	YVB	13
	Y(8) = (RVB(I) - DELVB) * FPPRVB(I) * FTVB(I) * XMVB(I) * MVR(I)	YVB	14
	IPI = I+1	YVR	15
	IF(IPI .GE. NVB) GO TO 12	YVB	16
	SUM = 0.	YVR	17
	DO 10 J = IPI, NVB	YVB	18
10	SUM = SUM - (RVB(4+1) - RVB(4)) * (RVB(4+1) * MVB(J+1)	YVB	19
A	+ RVB(J) * MVB(J))	YVB	20
12	Y(7) = FPPRVB(I) ** 2 * SUM / 2.	YVB	21
	RETURN	YVB	22
	END		

	SUBROUTINE POLLY(N,BBS,REL,AN,AA)	POLLY 1
	IMPLICIT REAL*8 (A-H,O-Z)	POLLY 2
C	COMPLEX ROOTS OF A POLYNOMIAL BAIRSTOWS METHOD	POLLY 3
	DIMENSION A(30),AN(60),C(26),ABAR(26),B(30),AA(30)	POLLY 4
	III=1	POLLY 5
	7 NPI=N+1	POLLY 6
	NPPI=N+2	POLLY 7
	DO 6 C1 I=1,NPI	POLLY 8
	LLL=NPPI-I	POLLY 9
601	A(I)=AA(LLL)	POLLY 10
13	DO 14 K=1,NPI	POLLY 11
14	ABAR(K)=A(K)	POLLY 12
	ABSSQ=BBS*BBS	POLLY 13
	RELSQ=REL*REL	POLLY 14
	NBAR=N	POLLY 15
	B(1)=A(1)	POLLY 16
	C(1)=A(1)	POLLY 17
15	IF(NBAR-2)200,210,17	POLLY 18
17	P1=.2	POLLY 19
	Q1=.1	POLLY 20
18	ITFR=0	POLLY 21
19	P1=P1*5.	POLLY 22
	Q1=Q1*10.	POLLY 23
33	P=P1	POLLY 24
	Q=Q1	POLLY 25
	NRP1=NBAR+1	POLLY 26
34	L=1	POLLY 27
	LAST=NBAR	POLLY 28
	DTFST=9.99D36	POLLY 29
C	BAIRSTOW ITERATION	POLLY 30
37	B(2)=ABAR(2)-P*B(1)	POLLY 31
	DO 40 K=3,NRP1	POLLY 32
40	B(K)=ABAR(K)-P*B(K-1)-Q*B(K-2)	POLLY 33
45	C(2)=B(2)-P*C(1)	POLLY 34
	DO 50 K=3, LAST	POLLY 35
50	C(K)=B(K)-P*C(K-1)-Q*C(K-2)	POLLY 36
	C(LAST)=C(LAST)-B(LAST)	POLLY 37
	D=C(LAST-1)*C(LAST-1)-C(LAST)*C(LAST-2)	POLLY 38
	DSQR=D*D	POLLY 39
	IF(DSQR-1.D-36)19,19,60	POLLY 40
60	DELP=(B(LAST)*C(LAST-1)-B(LAST+1)*C(LAST-2))/D	POLLY 41
	DELQ=(B(LAST+1)*C(LAST-1)-B(LAST)*C(LAST))/D	POLLY 42
C	TEST FOR CONVERGENCE	POLLY 43
	RELP=DELP/P	POLLY 44
	RELQ=DELQ/Q	POLLY 45
	RELPS=RELP*RELP	POLLY 46
	RELQS=RELQ*RELQ	POLLY 47
	DELSQ=RELPS+RELQS	POLLY 48
	P=P+DELP	POLLY 49
	Q=Q+DELQ	POLLY 50
	IF(RELPS-RELSQ)70,70,65	POLLY 51
65	IF(DELP*DELP-ABSSQ)70,70,80	POLLY 52
70	IF(RELQS-RELSQ)120,120,75	POLLY 53
75	IF(DELQ*DELQ-ABSSQ)120,120,80	POLLY 54
80	GO TO (90,100),L	POLLY 55

90	ITER=ITER+1	POLLY 56
	IF(250-ITER)310,37,37	POLLY 57
100	IF(DTEST-DELSQ)34,34,110	POLLY 58
110	DTEST=DELSQ	POLLY 59
	R(2)=A(2)-P*B(1)	POLLY 60
	DO 115 K=3,NP1	POLLY 61
115	R(K)=A(K)-P*B(K-1)-Q*B(K-2)	POLLY 62
	GO TO 45	POLLY 63
C	ITERATION HAS CONVERGED	POLLY 64
120	GO TO (130,140),L	POLLY 65
130	L=2	POLLY 66
	LAST=N	POLLY 67
	GO TO 110	POLLY 68
C	FACTOR OUT QUADRATIC	POLLY 69
140	NBAR=NBAR-2	POLLY 70
	NBPI=NBAR+1	POLLY 71
	ABAR(2)=ABAR(2)-P*ABAR(1)	POLLY 72
	DO 150 K=3,NBPI	POLLY 73
150	ABAR(K)=ABAR(K)-P*ABAR(K-1)-Q*ABAR(K-2)	POLLY 74
	GO TO 250	POLLY 75
C	SOLVE LINEAR EQUATION	POLLY 76
200	NBAR=NBAR-1	POLLY 77
	R1=-ABAR(2)/ABAR(1)	POLLY 78
	R2=0.	POLLY 79
	GO TO 262	POLLY 80
C	NORMALIZE QUADRATIC	POLLY 81
210	P=ABAR(2)/ABAR(1)	POLLY 82
	Q=ABAR(3)/ABAR(1)	POLLY 83
	NBAR=NBAR-2	POLLY 84
C	SOLVE NORMALIZED QUADRATIC	POLLY 85
250	R1=-P/2.	POLLY 86
	C1=R1*R1-Q	POLLY 87
	IF(C1)270,280,260	POLLY 88
260	C1=DSQRT(C1)	POLLY 89
	R2=R1-C1	POLLY 90
	R1=R1+C1	POLLY 91
262	C1=0.	POLLY 92
	GO TO 290	POLLY 93
270	C1=-C1	POLLY 94
	C1=DSQRT(C1)	POLLY 95
280	R2=R1	POLLY 96
290	C2=-C1	POLLY 97
	AN(III)=C1	POLLY 98
	AN(III+1)=R1	POLLY 99
	AN(III+2)=C2	POLLY100
	AN(III+3)=R2	POLLY101
	III=III+4	POLLY102
	IF(NBAR-1)4,200,15	POLLY103
C	SPECIAL CONDITIONS	POLLY104
310	WRITE (6,600)	POLLY105
600	FORMAT(IX,50HNO CONVERGENCE IN 250 ITERATIONS ,POLLY HAS SPOKEN)	POLLY106
	4 CONTINUE	POLLY107
	RETURN	POLLY108
	END	

```

SUBROUTINE SETIT(LGO,M,NV,REB,X,Y,UC,PRESS,GRAD,DELTA,DISP,THETA,VISSETUP 1
1SC,MTRAN) SETUP 2
C SETUP 3
C SUBROUTINE FOR CALCULATION OF BOUNDARY LAYER THICKNESS, SETUP 4
C DISPLACEMENT THICKNESS, MOMENTUM THICKNESS AND EDDY VISCOSITY. SETUP 5
C SETUP 6
DIMENSION X(300),Y(100),UC(100,3),VISC(100,2),GRAD(100) SETUP 7
RTR=SQRT(REB) SETUP 8
NY = NV + 2 SETUP 9
UEGE = .995*UC(NY,1) SETUP 10
DO 10 N=1,NV SETUP 11
IF(UEGE-UC(N+1,1)) 41,41,10 SETUP 12
41 NDELTA = N SETUP 13
GO TO 20 SETUP 14
10 CONTINUE SETUP 15
20 DELTA = Y(NDELTA)+(UEGE-UC(NDELTA,1))*(Y(NDELTA+1)-Y(NDELTA))/(UC(NDELTA
Y+1,1)-UC(NDELTA,1)) SETUP 16
SUM = 0. SETUP 17
DO 50 N=2,NY SETUP 18
50 SUM = SUM+(Y(N)-Y(N-1))*(UC(N,1)+UC(N-1,1)) SETUP 20
DISP = (Y(NY)-.5*SUM/UC(NY,1))/RTR SETUP 21
SUM = 0. SETUP 22
UEGE = UC(NY,1) SETUP 23
DO 60 N=2,NY SETUP 24
60 SUM = SUM+(Y(N)-Y(N-1))*((UEGE-UC(N,1))*UC(N,1)+(UEGE-UC(N-1,1))
1*UC(N-1,1)) SETUP 25
THETA = .5*SUM/(RTR*UEGE**2) SETUP 27
IF(LGO) 53,53,56 SETUP 28
53 NVPI=NV+1 SETUP 29
EASE = 1. SETUP 30
IF(M-MTRAN) 31,32,32 SETUP 31
32 IF(MTRAN+5-M) 31,31,33 SETUP 32
33 EASE = (X(M)-X(MTRAN))/(X(MTRAN+5)-X(MTRAN)) SETUP 33
31 CONTINUE SETUP 34
INNER=0 SETUP 35
FAC1 = .16*RTR*EASE SETUP 36
FAC2 = .0168*UEGE*DISP*REB*EASE SETUP 37
FFAC1 = -RTR/26. SETUP 38
EFAC2 = PRESS/RTR SETUP 39
TAUW = GRAD(1)/RTR SETUP 40
DO 160 N=2,NVPI SETUP 41
ALTER = 1.+FAC2/(1.+5.5*(Y(N)/DELTA)**6) SETUP 42
IF(INNER) 402,401,402 SETUP 43
402 VISC(N,1)=ALTER SETUP 44
GO TO 160 SETUP 45
401 CONTINUE SETUP 46
TAUMY=TAUW-Y(N)*EFAC2 SETUP 47
IF(TAUMY) 701,701,702 SETUP 48
701 VISC(N,1)=1. SETUP 49
GO TO 703 SETUP 50
702 FX=Y(N)*EFAC1*SQRT(TAUMY) SETUP 51
VISC(N,1) = 1.+FAC1*Y(N)*Y(N)*ABS(GRAD(N))*(1.-EXP(-FX))**2 SETUP 52
703 IF(VISC(N,1)-ALTER) 160,160,521 SETUP 53
521 VISC(N,1)=ALTER SETUP 54
INNER=1 SETUP 55

```

160	CONTINUE	SFTUP 56
	SAVE=1.	SETUP 57
	DO 162 N=2,NV	SETUP 58
	RAVE=VISC(N,1)	SETUP 59
	VISC(N,1)=(VISC(N+1,1)+RAVE+SAVE)/3.	SETUP 60
162	SAVE=RAVE	SETUP 61
56	CONTINUE	SETUP 62
	RETURN	SETUP 63
	END	

```

SUBROUTINE MIXER(FPRES,PREC,UINF,UDOT,THICK,NF,XBSIG,NSIG,INDT,DELMIXER 1
11,THET1,REB,USEP,X4,CP4) MIXER 2
DIMENSION FPRES(100),THICK(24),XBSIG(100) MIXER 3
FCAP(X)=-19.556*X+107.535*X*X-336.33*X**3+508.1*X**4-295.96*X**5 MIXER 4
UI1(X)=-.46532*X+.68425*X*X-.45293*X**3+.6592*X**4 MIXER 5
UI2(X)=-.045929*X-1.91615*X*X+2.91843*X**3-5.42125*X**4 MIXER 6
DIST=.5*(XBSIG(2)-XBSIG(1)) MIXER 7
XSEP=XBSIG(1)-DIST MIXER 8
XATT=XBSIG(NSIG)+DIST MIXER 9
C MIXER 10
C IF INDT IS NONZERO, THE BOUNDARY LAYER IS TURBULENT MIXER 11
C AT SEPARATION. MIXER 12
C MIXER 13
CALL H4X4(INDT,XSEP,DEL1,THET1,XATT,REB,USEP,X3,H3,X4,H4) MIXER 14
IF(XSEP-1.) 24,25,25 MIXER 15
25 CP4=0. MIXER 16
GO TO 27 MIXER 17
24 URAT=EXP(-.08712-UI1(H4)-.24723*(.3255+UI2(H4))) MIXER 18
CP4=1.-(1.-PREC)/URAT**2 MIXER 19
DEADL=XATT-XSEP MIXER 20
IF(DEADL-2.) 5,6,6 MIXER 21
5 G=(.5*DEADL)**2 MIXER 22
GO TO 7 MIXER 23
6 G=1. MIXER 24
7 CP4=PREC+(CP4-PREC)*(1.-G*XSEP) MIXER 25
27 CONTINUE MIXER 26
COEF=(PREC-CP4)/(XATT-X4) MIXER 27
CZ=2.*UDOT/UINF MIXER 28
C2=-2.*UINF MIXER 29
DO 20 M=1,NSIG MIXER 30
SUM=0. MIXER 31
COUNT=0. MIXER 32
X=XBSIG(M) MIXER 33
IF(X-1.) 2,2,3 MIXER 34
2 THETA = ARCT(X) MIXER 35
TANT = SIN(.5*THETA)/COS(.5*THETA) MIXER 36
CI = -CZ*(1.-COS(THETA)) MIXER 37
DO 10 N=1,NF MIXER 38
COUNT=COUNT+1. MIXER 39
ANGLE=COUNT*THETA MIXER 40
10 SUM=SUM+THICK(N)*(CI*COS(ANGLE)+CZ*(COUNT*TANT*SIN(ANGLE)-C)S(ANGL MIXER 41
1E))) MIXER 42
SUM=SUM-.5*CZ*THICK(1) MIXER 43
GO TO 35 MIXER 44
3 CI=CZ*(1.-X) MIXER 45
XRAD=1./(X+SQRT(X*X-1.)) MIXER 46
CI=CZ*(X-1.) MIXER 47
RF=SQRT((X-1.)/(X+1.)) MIXER 48
SUM=THICK(1)*XRAD*(CZ*(RF-1.)-CZ*(1.-.5*XRAD)) MIXER 49
FRAD=XRAD MIXER 50
COUNT=1. MIXER 51
DO 30 N=2,NF MIXER 52
COUNT=COUNT+1. MIXER 53
FRAD=FRAD*XRAD MIXER 54
30 SUM=SUM+THICK(N)*FRAD*(CZ*(COUNT*RF-1.)+CI) MIXER 55

```

35	CP=CP4	MIXER 56
	IF (X-X4) 55,50,50	MIXER 57
50	CP=CP+(X-X4)*COEF	MIXER 58
55	CONTINUE	MIXER 59
	FPRÉS(M)=-UTNF*CP+SUM	MIXER 60
20	CONTINUE	MIXER 61
	RETURN	MIXER 62
	END	MIXER 63

```

SUBROUTINE BUBB(DEL1,THET1,REB,XC1,U1,XC5,DCP,DEL5,X,XC,MX,NZ,X5,UBUBB 1
15,UF,ALTC,RENFL,USTOP) BUBB 2
DIMENSION X(300),XC(300),UE(300,3) BUBB 3
FCAP(X)=-19.556*X+107.535*X*X-336.33*X**3+508.1*X**4-295.06*X**5 BUBB 4
UI1(X)=-.46532*X+.68425*X*X-.45293*X**3+.6592*X**4 BUBB 5
UI2(X)=-.045929*X-1.91615*X*X+2.91843*X**3-5.42125*X**4 BUBB 6
FDEL1(X)=EXP(2.5773-.34252*X-.4379*X*X-.076511*X**3-.0039707*X**4) BUBB 7
FAICH(X)=EXP(-3.7481+.038772*X+.41967*X*X+.071046*X**3+.0032162*X**3 BUBB 8
1*4) BUBB 9
DEL1(X)=-.045929*ALOG(X)-3.9242*X+.54535*X*X-1.39147*X**3-10.8425*BUBB 10
1X**4 BUBB 11
25 FORMAT(1H1,44X,31H ANALYSIS OF LEADING-EDGE BUBBLE////34X,1HX,19X,1 BUBB 12
1HU,19X,1HH,18X,4H DISP/) BUBB 13
30 FORMAT(20X,4E20.5) BUBB 14
MOUT=6 BUBB 15
H1=.25 BUBB 16
H5=.429 BUBB 17
DC 5 M=NZ,MX BUBB 18
IF(XC1-XC(M)) 4,4,5 BUBB 19
4 M1=M BUBB 20
GO TO 6 BUBB 21
5 CONTINUE BUBB 22
6 X1=X(M1-1)+X(M1)-X(M1-1))*X(XC1-XC(M1-1))/(XC(M1)-XC(M1-1)) BUBB 23
X4=X1+RENFL/(U1*REB) BUBB 24
ARG=ALOG((X4-X1)/(REB*DEL1*DEL1*U1)) BUBB 25
H4=.25*FAICH(ARG) BUBB 26
DEL4=.53*FDEL1(ARG)*DEL1 BUBB 27
X5=X4+10.5*DEL4*(1.-(H4/.429)**2) BUBB 28
IF(U1-USTOP) 41,41,40 BUBB 29
40 ALTL=ALTC*DEL1 BUBB 30
IF(X5-X1.LT.ALTL) X5=X1+ALTL BUBB 31
41 URAT=EXP(-.08712-UI1(H4)-.24723*(.3255+UI2(H4))) BUBB 32
DCP=U1*U1*(1.-URAT**2) BUBB 33
DRAT=EXP(-2.24374-FCAP(H4)+.24723*(2.0214+DEL1(H4))) BUBB 34
DEL5=DRAT*DEL4 BUBB 35
DC 7 M=NZ,MX BUBB 36
IF(X5-X(M)) 16,16,7 BUBB 37
16 M5=M BUBB 38
GO TO 8 BUBB 39
7 CONTINUE BUBB 40
8 FACT=(X5-X(M5-1))/(X(M5)-X(M5-1)) BUBB 41
FACT1=1.-FACT BUBB 42
XC5=XC(M5-1)*FACT1+XC(M5)*FACT BUBB 43
U5=UF(M5-1,1)*FACT1+UE(M5,1)*FACT BUBB 44
WRITE(MOUT,25) BUBB 45
WRITE(MOUT,30) X1,U1,H1,DEL1 BUBB 46
WRITE(MOUT,30) X4,U1,H4,DEL4 BUBB 47
WRITE(MOUT,30) X5,U5,H5,DEL5 BUBB 48
RETURN BUBB 49
END BUBB 50

```



```
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
```

```
YSET 1
YSET 2
YSET 3
YSET 4
YSET 5
YSET 6
YSET 7
YSET 8
YSET 9
```

	SUBROUTINE H4X4 (INDT,X1,DEL1,THET1,X5,REB,U1,X3,H3,X4,H4)	H4X4	1
	CURLF(H)=26.703/H+305.03*ALOG(H)-2111.3*H+3327.8*H*H-2403.9*H**3	H4X4	2
	FDEL1(X)=EXP(2.5773-.34252*X-.4379*X*X-.076511*X**3-.0039707*X**4)	H4X4	3
	FAICH(X)=EXP(-3.7481+.038772*X+.41967*X*X+.071046*X**3+.0032162*X**4)	H4X4	4
	1*4)	H4X4	5
10	FORMAT(/20X,54HA SOLUTION FOR X4 COULD NOT BE OBTAINED IN 1000 TR	H4X4	6
	IALS)	H4X4	7
	MOUT=6	H4X4	8
C		H4X4	9
C	IF INDT IS NONZERO, THE BOUNDARY LAYER IS TURBULENT	H4X4	10
C	AT SEPARATION.	H4X4	11
C		H4X4	12
	IF (INDT) 2,5,2	H4X4	13
2	H3=THET1/DEL1	H4X4	14
	X3=X1	H4X4	15
	DEL3=DEL1	H4X4	16
	GO TO 20	H4X4	17
5	X3=X1+5.F4/(U1*REB)	H4X4	18
	ARG=ALOG((X3-X1)/(REB*DEL1*DEL1))	H4X4	19
	H3=THET1*FAICH(ARG)/DEL1	H4X4	20
	DEL3=.58*FDEL1(ARG)*DEL1	H4X4	21
	IF (X3-X5) 20,15,15	H4X4	22
15	H4=.429	H4X4	23
	X4=X5	H4X4	24
	GO TO 50	H4X4	25
20	CONTINUE	H4X4	26
	IG0=0	H4X4	27
	DIST=X5-X1	H4X4	28
	UNDER=0.	H4X4	29
	H4=H3+H3	H4X4	30
	COEF1=DEL3*H3	H4X4	31
	COEF2=10.5*DEL3*H3	H4X4	32
	SUB=X3-COEF1*CURLF(H3)	H4X4	33
95	OVER=H4	H4X4	34
	H4=.5*(H4+UNDER)	H4X4	35
	X4=CURLF(H4)*COEF1+SUB	H4X4	36
	ALTER=X5-COEF2*(1.-(H4/.429)**2)/H4	H4X4	37
	IG0=IG0+1	H4X4	38
	IF (X4-ALTER) 41,50,42	H4X4	39
41	IF (IG0-1000) 95,61,61	H4X4	40
42	IF (ABS(X4-ALTER)/DIST-.001) 50,50,43	H4X4	41
43	UNDER=H4	H4X4	42
	H4=.5*(OVER+H4)	H4X4	43
	X4=CURLF(H4)*COEF1+SUB	H4X4	44
	ALTER=X5-COEF2*(1.-(H4/.429)**2)/H4	H4X4	45
	IG0=IG0+1	H4X4	46
	IF (X4-ALTER) 52,50,51	H4X4	47
51	IF (IG0-1000) 43,61,61	H4X4	48
52	IF (ABS(X4-ALTER)/DIST-.001) 50,50,95	H4X4	49
61	H4=.429	H4X4	50
	X4=X5	H4X4	51
	WRITE (MOUT,10)	H4X4	52
50	CONTINUE	H4X4	53
	RETURN	H4X4	54
	END	H4X4	55

	SUBROUTINE SETSX(NSP1,XSEP,XATT,XSIG,ANGLE)	SETS X 1
	DIMENSION XSIG(100)	SETS X 2
	A=.5*(XSEP+XATT)	SETS X 3
	B=.5*(XATT-XSEP)	SETS X 4
	ARG=0.	SETS X 5
	DO 5 N=1,NSP1	SETS X 6
5	XSIG(N)=A-B*COS(ARG)	SETS X 7
	ARG=ARG+ANGLE	SETS X 8
	RETURN	SETS X 9
	END	SETS X 10

	FUNCTION ARCT(X)		ARCT	1
	PI=3.14159		ARCT	2
	IF (ABS(X)-1.E-6) 1,2,2		ARCT	3
1	ARCT=.5*PI		ARCT	4
	GO TO 6		ARCT	5
2	IF (X+.99999) 3,4,4		ARCT	6
3	ARCT=PI		ARCT	7
	GO TO 6		ARCT	8
4	ARCT=ATAN(SQRT(1.-X*X)/X)		ARCT	9
	IF (ARCT) 5,6,6		ARCT	10
5	ARCT=ARCT+PI		ARCT	11
6	CONTINUE		ARCT	12
	RETURN		ARCT	13
	END		ARCT	14

FUNCTION GAMI(ACAP,DXI,PI)	GAMI	1
DIMENSION ACAP(30,3)	GAMI	2
GAMI=PI*(-1.5*ACAP(1,1)-.75*ACAP(2,1)+2.*ACAP(1,2)+ACAP(2,2)-.5*AC	GAMI	3
IAP(1,3)-.25*ACAP(2,3))/DXI	GAMI	4
RETURN	GAMI	5
END	GAMI	6

	FUNCTION FB(X1,X2,X3,Y)	FB	1
	D1=1./(X2-X1)	FB	2
	D2=1./(X3-X2)	FB	3
	T1=ABS(Y-X1)	FB	4
	T2=ABS(Y-X2)	FB	5
	T3=ABS(Y-X3)	FB	6
	EPS=1.E-6	FB	7
	IF(T1-EPS) 2,3,3	FB	8
2	F1=0.	FB	9
	F2=ALOG(T2)	FB	10
	F3=ALOG(T3)	FB	11
	GO TO 10	FB	12
3	F1=ALOG(T1)	FB	13
	IF(T2-EPS) 4,5,5	FB	14
4	F2=0.	FB	15
	F3=ALOG(T3)	FB	16
	GO TO 10	FB	17
5	F2=ALOG(T2)	FB	18
	IF(T3-EPS) 6,7,7	FB	19
6	F3=0.	FB	20
	GO TO 10	FB	21
7	F3=ALOG(T3)	FB	22
10	FB=((Y-X1)*F1+D1+(D1+D2)*(X2-Y)*F2+(Y-X3)*F3+D2)/3.14159	FB	23
	RETURN	FB	24
	END	FB	25

	SUBROUTINE EGAMI (NU,NG,A,B,XSEP,XATT,GAMMA,Y,GI)	EGAMI 1
	DIMENSION A(30,3)	EGAMI 2
	SINT=SQRT(1.-Y*Y)	EGAMI 3
	THETA=ARCT(Y)	EGAMI 4
	SUM=0.	EGAMI 5
	CCUNT=1.	EGAMI 6
	DO 6 N=2,NG	EGAMI 7
	CCUNT=COUNT+1.	EGAMI 8
6	SUM=SUM+A(N+1,NU)*(SIN((COUNT+1.)*THETA)/(COUNT+1.)-SIN((COUNT-1.)	EGAMI 9
	1*THETA)/(COUNT-1.))	EGAMI 10
	GI=(3.14159-THETA+SINT)*(A(1,NU)+.5*A(2,NU))+.5*SUM-.25*GAMMA*(1.+	EGAMI 11
	1Y)*SINT*SINT	EGAMI 12
	IF(Y-XATT) 8,8,7	EGAMI 13
7	DIFF=1.-XATT	EGAMI 14
	IF(DIFF-1.E-6) 8,8,9	EGAMI 15
9	GI=GI+2.*B*DIFF**(-1.5)*SQRT((XATT-XSEP)*(1.-Y)*(Y-XATT))	EGAMI 16
8	CONTINUE	EGAMI 17
	RETURN	EGAMI 18
	END	EGAMI 19

	SUBROUTINE ESIGI (NU,AX,XS,B,Y,SI)	ESIGI 1
	DIMENSION XS(100),B(100,3)	ESIGI 2
	SUM=0.	ESIGI 3
	DC 10 I=2,NX	ESIGI 4
10	SUM=SUM+B(I,NU)*GR(XS(I-1),XS(I),XS(I+1),Y)	ESIGI 5
	SI=B(I,NU)*RINT(XS(1),XS(NX+1),Y)+SUM	ESIGI 6
	RETURN	ESIGI 7
	END	ESIGI 8


```
FUNCTION GB(X1,X2,X3,X)
GB=ABINT(X1,X2,X)-ABINT(X3,X2,X)
GB=GB/3.14159
RETURN
END
```

```
GB 1
GB 2
GB 3
GB 4
GB 5
```

	FUNCTION ABINT(A,B,X)	ABINT 1
	ARGA=ABS(X-A)	ABINT 2
	ARGB=ABS(X-B)	ABINT 3
	COEF=2.*(B-A)	ABINT 4
	AP1=A+1.	ABINT 5
	BP1=B+1.	ABINT 6
	IF (ARGA-1.E-6) 2,3,3	ABINT 7
2	CA=0.	ABINT 8
	GO TO 5	ABINT 9
3	CA=ALOG(ARGA)	ABINT 10
	IF (ARGB-1.E-6) 4,5,5	ABINT 11
4	CB=0.	ABINT 12
	GO TO 6	ABINT 13
5	CB=ALOG(ARGB)	ABINT 14
6	ABINT=(CA-.5)*ARGA**2-(CB-.5)*ARGB**2-(ALOG(AP1)-.5)*AP1**2+(ALOG(BP1)-.5)*BP1**2-COEF*((X-B)*(CB-1.)+BP1*(ALOG(BP1)-1.))	ABINT 15
	ABINT=ABINT/COEF	ABINT 16
	RETURN	ABINT 17
	FND	ABINT 18
		ABINT 19

	FUNCTION BINT(XS,XZ,X)	BINT	1
	RTS=SQRT(1.+XS)	BINT	2
	RTZ=SQRT(1.+XZ)	BINT	3
	BINT=-1.-X+RTS*RTZ	BINT	4
	IF(XZ-X) 2,3,3	BINT	5
2	RTSX=SQRT(X-XS)	BINT	6
	RTZX=SQRT(X-XZ)	BINT	7
	BINT=BINT+(XZ-XS)*ALOG((RTSX+RTZX)/(RTS+RTZ))+RTSX*RTZX	BINT	8
	GO TO 50	BINT	9
3	IF(X-XS) 5,5,4	BINT	10
4	BINT=BINT+(XZ-XS)*ALOG(SQRT(XZ-XS)/(RTS+RTZ))	BINT	11
	GO TO 50	BINT	12
5	RTSX=SQRT(XS-X)	BINT	13
	RTZX=SQRT(XZ-X)	BINT	14
	BINT=BINT+(XZ-XS)*ALOG((RTSX+RTZX)/(RTS+RTZ))-RTSX*RTZX	BINT	15
50	CONTINUE	BINT	16
	RETURN	BINT	17
	END	BINT	18

	SUBROUTINE SCAL(SBL, NSBL, FRZ, ARR, RDBR)	SCAL	1
	DIMENSION SBL(300)	SCAL	2
	DELZ=FRZ*RDBR	SCAL	3
	EN=ARR/FRZ	SCAL	4
	DO 5 N=1,300	SCAL	5
	IF(EN-N) 4,4,5	SCAL	6
4	NE=N	SCAL	7
	GO TO 6	SCAL	8
5	CONTINUE	SCAL	9
6	NG=NSBL-NE	SCAL	10
	EN=FLOAT(NG)	SCAL	11
	NGM1=NG-1	SCAL	12
	SBL(1)=0.	SCAL	13
	DO 7 N=2,NE	SCAL	14
7	SBL(N)=SBL(N-1)+CELZ	SCAL	15
	FRACT=2.2/DELZ	SCAL	16
	FRAC1=FRACT-1.	SCAL	17
	R=FRACT**(1./FLOAT(NGM1))	SCAL	18
8	SAVE=R	SCAL	19
	R=R-(R**NG-FRACT*R+FRAC1)/(EN*R**NGM1-FRACT)	SCAL	20
	IF(ABS(SAVE-R)-1.E-6) 9,9,8	SCAL	21
9	RPI=R+1.	SCAL	22
	DO 10 N=NE,NSBL	SCAL	23
10	SBL(N+1)=RPI*SBL(N)-R*SBL(N-1)	SCAL	24
	RETURN	SCAL	25
	END	SCAL	26

	SUBROUTINE TERPF(XI,J,TAB1,TAB2,TAB3,TAB4,XITAB,FP)	TERPF 1
	DIMENSION TAB1(24),TAB2(24),TAB3(24),TAB4(24),XITAB(24)	TERPF 2
	IF(XI-.0001) 2,2,10	TERPF 3
2	GO TO (3,4,5,6),J	TERPF 4
3	FP=2.53-2.439*ALOG(XI)	TERPF 5
	GO TO 99	TERPF 6
4	FP=3.54-1.725*ALOG(.7071*XI)	TERPF 7
	GO TO 99	TERPF 8
5	FP=4.58-1.2195*ALOG(.5*XI)	TERPF 9
	GO TO 99	TERPF 10
6	FP=10.12	TERPF 11
	GO TO 99	TERPF 12
10	DO 12 N=1,24	TERPF 13
	IF(XI-XITAB(N)) 11,11,12	TERPF 14
11	NX=N	TERPF 15
	GO TO 13	TERPF 16
12	CONTINUE	TERPF 17
13	TX=(XI-XITAB(NX-1))/(XITAB(NX)-XITAB(NX-1))	TERPF 18
	TX1=1.-TX	TERPF 19
	GO TO (14,15,16,17),J	TERPF 20
14	FP=TX1*TAB1(NX-1)+TX*TAB1(NX)	TERPF 21
	GO TO 99	TERPF 22
15	FP=TX1*TAB2(NX-1)+TX*TAB2(NX)	TERPF 23
	GO TO 99	TERPF 24
16	FP=TX1*TAB3(NX-1)+TX*TAB3(NX)	TERPF 25
	GO TO 99	TERPF 26
17	FP=TX1*TAB4(NX-1)+TX*TAB4(NX)	TERPF 27
99	CONTINUE	TERPF 28
	RETURN	TERPF 29
	END	TERPF 30

SUBROUTINE EVAL(NNF,XX,SSC,SST,CCB,TTB,CCM,TTM)	EVAL	1
DIMENSION SSC(50),SST(50)	EVAL	2
COST = 2.*XX - 1.	EVAL	3
COSTS = COST**2	EVAL	4
IF(COSTS-1.E-8) 303,304,304	EVAL	5
304 TANT = SQRT(1./COSTS - 1.)	EVAL	6
THE = ATAN(TANT)	EVAL	7
GO TO 305	EVAL	8
303 THE = 1.5708	EVAL	9
305 IF(COST) 403,404,404	EVAL	10
403 THE = 3.14159 - THE	EVAL	11
404 ARG = 0.	EVAL	12
SUM1 = 0.	EVAL	13
SUM2 = 0.	EVAL	14
DO 551 N=1,NNF	EVAL	15
ARG = ARG + THE	EVAL	16
SUM1 = SUM1 + SSC(N)*SIN(ARG)	EVAL	17
551 SUM2 = SUM2 + SST(N)*SIN(ARG)	EVAL	18
CCB = SUM1*SIN(THE)*CCM	EVAL	19
TTB = (1. - COS(THE))*SUM2*TTM	EVAL	20
RETURN	EVAL	21
END	EVAL	22

	SUBROUTINE SIMP(NS,DX,ORD,FIND)	SIMP	1
	DIMENSION ORD(50)	SIMP	2
C	INTEGRATION OF NS + 1 EQUALLY SPACED ORDINATE VALUES	SIMP	3
C	BY SIMPSON'S RULE. NS MUST BE EVEN	SIMP	4
	SUM = 0.	SIMP	5
	DC 88 I=2,NS,2	SIMP	6
88	SUM = SUM + 2.*ORD(I-1) + 4.*ORD(I)	SIMP	7
	FIND = DX*(SUM - ORD(1) + ORD(NS+1))/3.	SIMP	8
	RETURN	SIMP	9
	END	SIMP	10

	SUBROUTINE SECT(XU,YU,XL,YL,NOFF,NF,RDRC,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),XL(30),YL(30),YUC(30),YLC(30),ST(24),SC(24)	SECT	4
	1),DUM(50),TBAR(50),CBAR(50)	SECT	5
12	FORMAT(////47X,26HINPUT AND COMPUTED OFFSETS/)	SECT	6
13	FORMAT(19X,4HX1/C,12X,4HYU/C,11X,5HYUC/C,20X,4HX1/C,12X,4HYL/C,11X	SECT	7
	1,5HYLC/C/)	SECT	8
14	FORMAT(9X,3F16.5,8X,3F16.5)	SECT	9
	NA=6	SECT	10
	RNA=6.	SECT	11
	RNF=FLOAT(NF)	SECT	12
	MCUT=6	SECT	13
	PI = 3.14159	SECT	14
	DELT = PI/(2.*RNF)	SECT	15
	NTC = 2*RNF - 1	SECT	16
	NINT = NTC + 2	SECT	17
	NSIMP = NTC + 1	SECT	18
	RDRC=.5*RDRC	SECT	19
	VARY = 0.	SECT	20
	CB = 0.	SECT	21
	TB = 0.	SECT	22
	THETA = 0.	SECT	23
	DO 89 K=1,NTC	SECT	24
	THETA = THETA + DELT	SECT	25
	XI = .5*(1. + COS(THETA))	SECT	26
	DO 90 LAM=2,NOFF	SECT	27
	IF(XI-XU(LAM)) 110,90,90	SECT	28
110	YUINT = YU(LAM-1) + (XI - XU(LAM-1))*(YU(LAM) - YU(LAM-1))/(XU(LAM	SECT	29
	1) - XU(LAM-1))	SECT	30
	GO TO 111	SECT	31
90	CONTINUE	SECT	32
111	DO 80 LAM=2,NOFF	SECT	33
	IF(XI-XL(LAM)) 210,80,80	SECT	34
210	YLINT = YL(LAM-1) + (XI - XL(LAM-1))*(YL(LAM) - YL(LAM-1))/(XL(LAM	SECT	35
	1) - XL(LAM-1))	SECT	36
	GO TO 112	SECT	37
80	CONTINUE	SECT	38
112	TBAR(K+1) = .5*(YUINT - YLINT)	SECT	39
89	CBAR(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
802	TMAX = TBAR(K)	SECT	45
801	IF(CBAR(K)-CMAX) 79,702,702	SECT	46
702	CMAX = CBAR(K)	SECT	47
79	CONTINUE	SECT	48
	IF(CMAX-1.E-5) 1201,1202,1202	SECT	49
1201	CMAX=1.	SECT	50
1202	CONTINUE	SECT	51
	IF(TMAX-1.E-5) 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

69	CBAR(K) = CBAR(K)/CMAX	SECT 56
	TBAR(1) = 0.	SECT 57
	CBAR(1) = 0.	SECT 58
	TBAR(NINT) = 0.	SECT 59
	CBAR(NINT) = 0.	SECT 60
	TTA = TBAR(NA)	SECT 61
	TTB = TBAR(NA+1)	SECT 62
	TTC = TBAR(NA+2)	SECT 63
	TAA = DELT*(RNA-1.)	SECT 64
	TBB = TAA + DELT	SECT 65
	TCC = TBB + DELT	SECT 66
	XA = .5*COS(TAA)	SECT 67
	XB = .5*COS(TBB)	SECT 68
	XC = .5*COS(TCC)	SECT 69
	SLOPE = ((TTC-TTB)*(XB-XA)/(XC-XB) + (TTB-TTA)*(XC-XB)/(XB-XA))/(XSECT	70
	1C-XA)	SECT 71
	THETA = 0.	SECT 72
	COSB = COS(TBB)	SECT 73
	DO 456 I=2,NA	SECT 74
	THETA = THETA + DELT	SECT 75
	COST = COS(THETA)	SECT 76
456	TBAR(I) = (SQRT(1.-COST)/(1.-COSB)**1.5)*(TTB*(1.+COST-2.*COSB)/(1	SECT 77
	1.-COSB) + .5*SLOPE*(COST-COSB))	SECT 78
	NLE = 2*NF + 1 - NA	SECT 79
	COSR1 = 1. + COS(PI-RNA*DELT)	SECT 80
	THETA = PI	SECT 81
	SINAS=SIN(RNA*DELT)**2	SECT 82
	COSAS=COS(RNA*DELT)	SECT 83
	ANG=0.	SECT 84
	DO 457 I=2,NA	SECT 85
	IND = 2*NF + 2 - I	SECT 86
	THETA = THETA - DELT	SECT 87
	COST1 = 1. + COS(THETA)	SECT 88
	ANG=ANG+DELT	SECT 89
	COEF=(SINAS-SIN(ANG)**2)/(COSR1*(COS(ANG)+COSAS))	SECT 90
457	TBAR(IND) = (SQRT(RDBC*COST1)*COEF/TMAX+TBAR(NLE)*(COST1/COSR1)**1	SECT 91
	1.5)/(2.-COST1)	SECT 92
	THETA = TAA	SECT 93
	NAPI = NA + 1	SECT 94
	DO 458 I = NAPI,NLE	SECT 95
	THETA = THETA + DELT	SECT 96
458	TBAR(I) = TBAR(I)/(1.-COS(THETA))	SECT 97
	THETA = 0.	SECT 98
	DO 459 I=2,NSIMP	SECT 99
	THETA = THETA + DELT	SECT 100
459	CBAR(I) = CBAR(I)/SIN(THETA)	SECT 101
	RKK = 0.	SECT 102
	DO 59 K=1,NF	SECT 103
	RKK = RKK + 1.	SECT 104
	THETA = 0.	SECT 105
	DO 777 I=1,NINT	SECT 106
	DUM(I) = TBAR(I)*SIN(THETA*RKK)	SECT 107
777	THETA = THETA + DELT	SECT 108
	CALL SIMP(NSIMP,DELT,DUM,VARY)	SECT 109
	ST(K) = 2.*VARY/PI	SECT 110

THETA = 0.	SECT 111
DO 888 I=1,NINT	SECT 112
DUM(I) = CBAR(I)*SIN(THETA*RKK)	SECT 113
888 THETA = THETA + DELT	SECT 114
CALL SIMP(NSIMP,DELT,DUM,VARY)	SECT 115
59 SC(K) = 2.*VARY/PI	SECT 116
DO 969 I=1,NOFF	SECT 117
X = XU(I)	SECT 118
CALL EVAL(NF,X,SC,ST,CB,TB,CMAX,TMAX)	SECT 119
569 YUC(I) = CB + TB	SECT 120
DO 869 I=1,NOFF	SECT 121
X = XL(I)	SECT 122
CALL EVAL(NF,X,SC,ST,CB,TB,CMAX,TMAX)	SECT 123
869 YLC(I) = CB - TB	SECT 124
SUM1 = 0.	SECT 125
COUNT = 0.	SECT 126
DO 699 I=1,NF	SECT 127
COUNT = COUNT + 1.	SECT 128
699 SUM1 = SUM1 - ST(I)*COUNT*(-1.)**I	SECT 129
RCDBC = 8.*(TMAX*SUM1)**2	SECT 130
RCDBC=2.*RCDBC	SECT 131
TMAX=2.*TMAX	SECT 132
C MAX=2.*CMAX	SECT 133
WRITE(MOUT,12)	SECT 134
WRITE(MOUT,13)	SECT 135
WRITE(MOUT,14) (XU(I),YU(I),YUC(I),XL(I),YL(I),YLC(I),I=1,NOFF)	SECT 136
RETURN	SECT 137
END	SECT 138

	SUBROUTINE CORDX(NSBL,NZ,RDBB,SBL,X,XC)	CORDX 1
C		CORDX 2
C	BOUNDARY LAYER COORDINATES AND CORRESPONDING CHORDAL	CORDX 3
C	COORDINATES ARE COMPUTED HERE.	CORDX 4
C		CORDX 5
	DIMENSION SBL(300),X(300),XC(300)	CORDX 6
336	FORMAT(/10X,31H ITERATION TO COMPUTE XC FOR M = 15,32H DID NOT CONVCORDX 7	
	ERGE IN 1000 STEPS.)	CORDX 8
337	FORMAT(1H1,25X,1HM,20X,1HS,25X,1HX,24X,2HXC//)	CORDX 9
338	FORMAT(22X,I5,3E25.5)	CORDX 10
	MOUT=6	CORDX 11
	MX = NSBL + NZ - 1	CORDX 12
	RZERO = RDBB/2.	CORDX 13
	XC(NZ) = -1.	CORDX 14
	DO 255 M=1,NZ	CORDX 15
	MM = NZ + 1 - M	CORDX 16
255	X(M) = SBL(NZ) - SBL(MM)	CORDX 17
	DO 256 M=NZ,MX	CORDX 18
	MM = M + 1 - NZ	CORDX 19
256	X(M) = SBL(NZ) + SBL(MM)	CORDX 20
	DO 257 M=1,MX	CORDX 21
	IF(NZ-M) 333,257,335	CORDX 22
333	K = M + 1 - NZ	CORDX 23
	GO TO 334	CORDX 24
335	K = NZ - M + 1	CORDX 25
334	XC(M) = -1. + SBL(K)	CORDX 26
	IF(SBL(K)-RZERO) 341,341,342	CORDX 27
341	XC(M) = -1. + SBL(K)**2/(4.*RZERO)	CORDX 28
342	CONTINUE	CORDX 29
	DO 258 L=1,1000	CORDX 30
	SAVE = XC(M)	CORDX 31
	CALC1 = SQRT((1.+XC(M))/RZERO)	CORDX 32
	CALC2 = SQRT(1.+(1.+XC(M))/RZERO)	CORDX 33
	XC(M)=XC(M)+CALC1*(SBL(K) - RZERO*(CALC1*CALC2+ALOG(CALC1+CALC2)	CORDX 34
	I)/CALC2	CORDX 35
	IF(ABS(SAVE-XC(M))-1.E-6) 257,257,258	CORDX 36
258	CONTINUE	CORDX 37
	WRITE(MOUT,336) M	CORDX 38
257	CONTINUE	CORDX 39
	WRITE(MOUT,337)	CORDX 40
	DO 264 M=1,MX	CORDX 41
	IF(NZ-M) 261,261,262	CORDX 42
262	K=NZ-M+1	CORDX 43
	GO TO 263	CORDX 44
261	K=M+1-NZ	CORDX 45
263	WRITE(MOUT,338) M,SBL(K),X(M),XC(M)	CORDX 46
264	CONTINUE	CORDX 47
	RETURN	CORDX 48
	END	CORDX 49

	SUBROUTINE PGRAD(M,X,UE,DXI,PRESS,SA,SB,SC,SR,SS)	PGRAD 1
C		PGRAD 2
C	SUBROUTINE FOR CALCULATION OF PRESSURE GRADIENT AND	PGRAD 3
C	DERIVATIVE COEFFICIENTS.	PGRAD 4
C		PGRAD 5
	DIMENSION X(300),UE(300,3)	PGRAD 6
	D1Z=X(M+1)-X(M)	PGRAD 7
	D2Z=X(M+2)-X(M)	PGRAD 8
	D21=X(M+2)-X(M+1)	PGRAD 9
	D1M1=X(M+1)-X(M-1)	PGRAD 10
	DZM1=X(M)-X(M-1)	PGRAD 11
	XIM=D1Z/(D2Z*D21)	PGRAD 12
	ETAM=1./D1Z-1./D21	PGRAD 13
	ZETAM=D21/(D1Z*D2Z)	PGRAD 14
	PRESS = (3.*UE(M+1,1)-4.*UE(M+1,2)+UE(M+1,3))/(2.*DXI)+UE(M+1,1)*	PGRAD 15
	1XIM*UE(M+2,1)+ETAM*UE(M+1,1)-ZETAM*UE(M,1))	PGRAD 16
	SA=1./D1Z+1./D1M1	PGRAD 17
	SB=D1M1/(D1Z*DZM1)	PGRAD 18
	SC=D1Z/(D1M1*DZM1)	PGRAD 19
	SR=D1M1/DZM1	PGRAD 20
	SS=D1Z/DZM1	PGRAD 21
	RETURN	PGRAD 27
	END	PGRAD 23

	SUBROUTINE TRANS (UPRIM, PRESS, THETA, REB, UC, NY, FLAM, XFLAM, LAMQ)	TRANS 1
C		TRANS 2
C	SUBROUTINE TO TEST FOR TRANSITION IN A LAMINAR BOUNDARY LAYER.	TRANS 3
C		TRANS 4
	DIMENSION UC(100,3), FLAM(10), XFLAM(10)	TRANS 5
	F(X) = .11746 - 1.0582E-3*X - 1.1023E-4*X*X	TRANS 6
	TKAY = PRESS*REP*THETA**2/UC(NY,2)	TRANS 7
	IF(TKAY-.077) 2,2,99	TRANS 8
2	IF(ABS(TKAY)-.0701) 3,3,4	TRANS 9
3	ARG = TKAY*72.48	TRANS 10
	GO TO 5	TRANS 11
4	ARG = 0.	TRANS 12
	DO 6 N=1,1000	TRANS 13
	SAVE = ARG	TRANS 14
	ARG = ARG - (ARG*(F(ARG)**2-TKAY)/(F(ARG)*(.11746-ARG*3.1746E-3 - A	TRANS 15
	IRG*ARG*5.5115E-4))	TRANS 16
	IF(ABS(1.-SAVE/ARG)-1.E-6) 7,7,6	TRANS 17
6	CONTINUE	TRANS 18
7	IF(ARG+11.) 8,8,5	TRANS 19
8	EF = 1.75	TRANS 20
	GO TO 10	TRANS 21
5	DO 15 N=1,10	TRANS 22
	IF(ARG-XFLAM(N)) 24,24,15	TRANS 23
24	NBAR = N	TRANS 24
	GO TO 16	TRANS 25
15	CONTINUE	TRANS 26
16	EF = FLAM(NBAR-1)+(ARG-XFLAM(NBAR-1))*(FLAM(NBAR)-FLAM(NBAR-1))/(X	TRANS 27
	IFLAM(NBAR)-XFLAM(NBAR-1))	TRANS 28
10	B = .5*EF	TRANS 29
	A = 3.36*(UPRIM/UC(NY,2))**2	TRANS 30
	RTH = F(ARG)*(SQRT(B*B+9860.*A)-B)/A	TRANS 31
	IF(REB*THETA-RTH) 99,50,50	TRANS 32
50	LAMQ = 0	TRANS 33
99	CONTINUE	TRANS 34
	RETURN	TRANS 35
	END	TRANS 36

```

SUBROUTINE CAPS(ITER,N,CAPG,CAPH,CAPJ,CAPK,SR,SS,SD,SE,SF,VISC,V,UCAPS 1
1C) CAPS 2
DIMENSION CAPG(100),CAPH(100),CAPJ(100),CAPK(100) CAPS 3
DIMENSION VISC(100,2),V(100,2),UC(100,3),SD(100),SE(100),SF(100) CAPS 4
IF(ITER) 4,2,4 CAPS 5
2 CAPG(N)= SR*V(N,1) - SS*V(N,2) CAPS 6
CAPH(N)=SR*VISC(N,1)-SS*VISC(N,2) CAPS 7
CAPJ(N)=SR*(SD(N)*VISC(N+1,1)+SE(N)*VISC(N,1)-SF(N)*VISC(N-1,1))-SCAPS 8
1S*(SD(N)*VISC(N+1,2)+SE(N)*VISC(N,2)-SF(N)*VISC(N-1,2)) CAPS 9
CAPK(N)= SR*UC(N,2)-SS*UC(N,3) CAPS 10
GO TO 6 CAPS 11
4 CAPG(N)=.5*(CAPG(N)+V(N,1)) CAPS 12
CAPH(N)=.5*(CAPH(N)+VISC(N,1)) CAPS 13
CAPJ(N)=.5*(CAPJ(N)+SD(N)*VISC(N+1,1)+SE(N)*VISC(N,1)-SF(N)*VISC(NCAPS 14
1-1,1)) CAPS 15
CAPK(N)=.5*(CAPK(N)+UC(N,1)) CAPS 16
6 CONTINUE CAPS 17
RETURN CAPS 18
END CAPS 19

```

	SUBROUTINE TERP(YIN,YBASE,VARY,NY,VALUE)	TERP	1
C		TERP	2
C	SUBROUTINE FOR DETERMINING INTERPOLATED VALUE OF THE	TERP	3
C	FUNCTION VARY AT Y = YIN.	TERP	4
C		TERP	5
	DIMENSION YBASE(100),VARY(100)	TERP	6
	IF(YIN-YBASE(NY-1)) 2,3,3	TERP	7
3	VALUE = VARY(NY)	TERP	8
	GO TO 10	TERP	9
2	DO 15 N=1,NY	TERP	10
	IF(YIN-YBASE(N)) 24,24,15	TERP	11
24	NBAR=N	TERP	12
	GO TO 16	TERP	13
15	CONTINUE	TERP	14
16	D21=YBASE(NBAR)-YBASE(NBAR-1)	TERP	15
	D31=YBASE(NBAR+1)-YBASE(NBAR-1)	TERP	16
	D32=D31-D21	TERP	17
	D3A=YBASE(NBAR+1)-YIN	TERP	18
	D2A=YBASE(NBAR)-YIN	TERP	19
	DA1=YIN-YBASE(NBAR-1)	TERP	20
	VALUE=D3A*D2A*VARY(NBAR-1)/(D21*D31)+D3A*DA1*VARY(NBAR)/(D21*D32)-	TERP	21
	1D2A*DA1*VARY(NBAR+1)/(D31*D32)	TERP	22
10	CONTINUE	TERP	23
	RETURN	TERP	24
	END	TERP	25

```

SUBROUTINE YDIFF(NY,ALPHA,BETA,GAMMA,DELTA,SD,SE,SF,C2,C3,C4,Y)      YDIFF  1
DIMENSION ALPHA(100),BETA(100),GAMMA(100),DELTA(100)           YDIFF  2
DIMENSION SD(100),SE(100),SF(100),Y(100)                        YDIFF  3
NV=NY-2                                                            YDIFF  4
NVPI=NV+1                                                          YDIFF  5
DC 40 N=2,NV                                                       YDIFF  6
ALPHA(N) = 2.*(2.*Y(N)-Y(N-1)-Y(N+1))/((Y(N+2)-Y(N-1))*(Y(N+2)-Y(N
1+1))*(Y(N+2)-Y(N)))                                             YDIFF  7
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)))                                             YDIFF  8
BETA(N) = (DELTA(N)*Y(N)-Y(N-1))**3-ALPHA(N)*(Y(N+2)-Y(N))**3)/(Y
1(N+1)-Y(N))**3                                                  YDIFF  9
GAMMA(N) = -ALPHA(N)-BETA(N)-DELTA(N)                            YDIFF 10
CONTINUE                                                            YDIFF 11
DO 39 N=2,NVPI                                                     YDIFF 12
SD(N) = (Y(N)-Y(N-1))/((Y(N+1)-Y(N-1))*(Y(N+1)-Y(N)))          YDIFF 13
SE(N) = 1./(Y(N)-Y(N-1))-1./(Y(N+1)-Y(N))                        YDIFF 14
SF(N) = (Y(N+1)-Y(N))/((Y(N)-Y(N-1))*(Y(N+1)-Y(N-1)))          YDIFF 15
CONTINUE                                                            YDIFF 16
C2 = Y(3)*Y(4)/(Y(2)*(Y(3)-Y(2))*(Y(4)-Y(2)))                  YDIFF 17
C3 = -Y(2)*Y(4)/(Y(3)*(Y(4)-Y(3))*(Y(3)-Y(2)))                 YDIFF 18
C4 = Y(2)*Y(3)/(Y(4)*(Y(4)-Y(3))*(Y(4)-Y(2)))                  YDIFF 19
RETURN                                                            YDIFF 20
END                                                                  YDIFF 21

```


	SUBROUTINE ELDER(BCAP, XSIG, NSIG, UINF, ELD, Y, YMAX)	ELDER 1
	DIMENSION BCAP(100,3), XSIG(100)	ELDER 2
	BCAP(NSIG+1,1)=0.	ELDER 3
	XS=XSIG(1)	ELDER 4
	XZ=XSIG(NSIG+1)	ELDER 5
	IF(XZ-1.) 16,16,1	ELDER 6
1	DEADL=XZ-XS	ELDER 7
	YMAX=1.E-10	ELDER 8
	SUM=.5*(XSIG(2)-XS)*BCAP(2,1)	ELDER 9
	DO 10 N=2, NSIG	ELDER 10
	X=XSIG(N+1)	ELDER 11
	SUM=SUM+.5*(X-XSIG(N))*(BCAP(N+1,1)+BCAP(N,1))	ELDER 12
	IF(N-NSIG) 4,2,4	ELDER 13
2	ANGLE=1.5708	ELDER 14
	GO TO 6	ELDER 15
4	ANGLE=ATAN(SQRT((X-XS)/(XZ-X)))	ELDER 16
6	Y=SUM+BCAP(1,1)*(DEADL*ANGLE-SQRT((X-XS)*(XZ-X)))	ELDER 17
	IF(Y-YMAX) 10,10,8	ELDER 18
8	YMAX=Y	ELDER 19
10	CONTINUE	ELDER 20
	ELD=Y/YMAX	ELDER 21
	IF(ABS(ELD)-UINF) 20,20,12	ELDER 22
12	IF(ELD) 14,16,16	ELDER 23
14	ELD=-UINF	ELDER 24
	GO TO 20	ELDER 25
16	ELD=UINF	ELDER 26
20	CONTINUE	ELDER 27
	RETURN	ELDER 28
	END	ELDER 29

```
SUBROUTINE REATT(UC,V,X,Y,MX,NY,RY,DRY,UE,X5,DEL5,MST,REB)      REATT  1
DIMENSION UC(100,3),V(100,2),Y(100)                             REATT  2
DIMENSION X(300),IF(300,3)                                       REATT  3
DIMENSION TAB1(24),TAB2(24),TAB3(24),TAB4(24),XITAB(24)       REATT  4
DATA TAB1 /24.98,23.29,21.04,19.33,17.61,15.29,13.46,11.54,10.36,9.      REATT  5
1.38,8.35,7.32,6.2,5.31,4.4,3.57,2.22,1.26,.66,.31,.14,.01,0.,0./ REATT  6
DATA TAB2 /20.05,18.85,17.25,15.04,14.8,13.12,11.77,10.3,9.36,8.65REATT  7
1,7.95,7.2,6.43,5.66,4.9,4.18,2.89,1.86,1.11,.62,.32,.04,0.,0./  REATT  8
DATA TAB3 /16.65,15.8,14.67,13.8,12.91,11.66,10.65,9.48,8.71,8.11,REATT  9
17.59,7.01,6.41,5.77,5.13,4.5,3.31,2.28,1.48,.9,.51,.09,.01,0./  REATT 10
DATA TAB4 /10.12,10.05,9.93,9.78,9.58,9.17,8.72,8.08,7.6,7.2,6.85,REATT 11
16.53,6.18,5.79,5.36,4.91,3.98,3.05,2.21,1.5,.95,.22,.03,0./   REATT 12
DATA XITAB /,.0001,.0002,.0005,.001,.002,.005,.01,.02,.03,.04,.05,.REATT 13
106,.07,.08,.09,.1,.12,.14,.16,.18,.2,.25,.3,.35/             REATT 14
3  FORMAT(///40X,23HAT REATTACHMENT, BETA =E13.5)                REATT 15
MOUT=6                                                            REATT 16
RTR=SQRT(REB)                                                    REATT 17
UC(1,2)=0.                                                         REATT 18
UC(1,3)=0.                                                         REATT 19
V(1,1)=0.                                                         REATT 20
V(1,2)=0.                                                         REATT 21
DO 5 M=1,MX                                                       REATT 22
IF(X5-X(M)) 4,4,5                                               REATT 23
4  MST=M+2                                                         REATT 24
GO TO 6                                                           REATT 25
5  CONTINUE                                                       REATT 26
6  XA=X(MST-2)                                                    REATT 27
  XB=X(MST-1)                                                    REATT 28
  UA=UE(MST-2,1)                                                REATT 29
  UB=UE(MST-1,1)                                                REATT 30
  ZA=ALOG(UA*DEL5*REB)                                           REATT 31
  PGRAD=2.*(UA-UB)/((UA+UB)*(XB-XA))                             REATT 32
  BETM2=(.0974-SQRT(DEL5*PGRAD))/(.0249+.004565*Z A)          REATT 33
  IF(BETM2-1.) 8,7,7                                             REATT 34
7  BETM2=1.                                                       REATT 35
GO TO 10                                                         REATT 36
8  IF(BETM2-.3) 9,9,10                                           REATT 37
9  BETM2=.3                                                       REATT 38
10 BETA=1.7*(BETM2*BETM2)                                         REATT 39
WRITE(MOUT,3) BETA                                             REATT 40
AGAM=.0974*BETM2-.0249/BETA                                       REATT 41
BGAM=.004565/BETA                                               REATT 42
AH=1.-(5.3+3.9*BETM2)*(.0974-.0249*BETM2)                     REATT 43
BH=BETM2*(5.3+3.9*BETM2)*.004565                               REATT 44
GAMA=AGAM-BGAM*ZA                                             REATT 45
DERIV=UA*REB*EXP(-ZA)*GAMA*GAMA*(1.+BETA*(1.+AH+BH*ZA))/(AH+BH*BH*      REATT 46
IZA)                                                            REATT 47
ZR=ZA+DERIV*(XB-XA)                                             REATT 48
DELB=EXP(ZR)/(UB*REB)                                           REATT 49
GAMB=AGAM-BGAM*ZR                                             REATT 50
DELL=.35*DELB*RTR*BETM2/GAMB                                       REATT 51
11 IF(DELL-Y(NY-3)) 14,12,12                                       REATT 52
12 RY=RY+DRY                                                    REATT 53
CALL YSET(RY,Y(2),NY,Y)                                         REATT 54
GO TO 11                                                         REATT 55
```

14	IF(BETA-4.) 102,101,101	REATT 56
101	TERPB=1.-4./BETA	REATT 57
	INDEX=3	REATT 58
	GO TO 110	REATT 59
102	IF(BETA-2.) 104,103,103	REATT 60
103	TERPB=.5*BETA-1.	REATT 61
	INDEX=2	REATT 62
	GO TO 110	REATT 63
104	TERPB=BETA-1.	REATT 64
	INDEX=1	REATT 65
110	K=0	REATT 66
	TERP1=1.-TERPB	REATT 67
50	K=K+1	REATT 68
	GO TO (16,17,99),K	REATT 69
16	G=GAMA	REATT 70
	DELTA=DELS	REATT 71
	UEDGE=UA	REATT 72
	L=3	REATT 73
	GO TO 18	REATT 74
17	G=GAMB	REATT 75
	DELTA=DELR	REATT 76
	UEDGE=UB	REATT 77
	L=2	REATT 78
18	XICO=G/(DELTA*RTR*BETM2)	REATT 79
	UCOW=RTR*(UEDGE*G)**2	REATT 80
	EFCO=G/BETM2	REATT 81
	NLAM=NY	REATT 82
	DO 75 N=2, NY	REATT 83
	XI=Y(N)*XICO	REATT 84
	IF(XI-.35) 20,19,19	REATT 85
19	UC(N,L)=UEDGE	REATT 86
	GO TO 75	REATT 87
20	CALL TERPF(XI,INDEX,TAB1,TAB2,TAB3,TAB4,XITAB,FP1)	REATT 88
	INDP1=INDEX+1	REATT 89
	CALL TERPF(XI,INDP1,TAB1,TAB2,TAB3,TAB4,XITAB,FP2)	REATT 90
	FP=TERP1*FP1+TERPB*FP2	REATT 91
	UC(N,L)=UEDGE*(1.-EFCO*FP)	REATT 92
	IF(N-NLAM) 21,75,75	REATT 93
21	ALTER=UCOW*Y(N)	REATT 94
	IF(ALTER-UC(N,L)) 33,33,32	REATT 95
32	UC(N,L)=ALTER	REATT 96
	GO TO 75	REATT 97
33	NLAM=N	REATT 98
75	CONTINUE	REATT 99
	GO TO 50	REATT100
99	DO 60 K=2,3	REATT101
	SAVE2=0.	REATT102
	DO 60 N=3, NY	REATT103
	SAVE1=UC(N-1,K)	REATT104
	UC(N-1,K)=(SAVE2+SAVE1+UC(N,K))/3.	REATT105
60	SAVE2=SAVE1	REATT106
	DUDX=0.	REATT107
	COO=.5/(XB-XA)	REATT108
	DO 65 N=2, NY	REATT109
	DUDXP=COO*(UC(N,2)-UC(N,3))	REATT110

```
65 V(N,1)=V(N-1,1)-(Y(N)-Y(N-1))*(DUDXP+DUDX)  
V(N,2)=V(N,1)  
DUDX=DUDXP  
RETURN  
END
```

```
REATT111  
REATT112  
REATT113  
REATT114  
REATT115
```

```

SUBROUTINE ELPIT (ALPH1, ALPH2, EMI, TORF, THETZ, UINF, DXI, CMPA, CPAS)  ELPIT  1
SAVE T=ALPH1  ELPIT  2
STEP=TORF*DXI  ELPIT  3
SINS=SIN(STEP)  ELPIT  4
COSS=COS(STEP)  ELPIT  5
CONST=2.*EMI*(UINF/TORF)**2  ELPIT  6
ALPH1=THETZ+(ALPH1-THETZ)*COSS+ALPH2*SINS/TORF+CONST*(2.*CMPA-CMPAELPIT  7
IS)*(1.-COSS)+CONST*(CPAS-CMPA)*(SINS-STEP*COSS)/(TORF*DXI)  ELPIT  8
ALPH2=ALPH2*COSS-TORF*SINS*(SAVE T-THETZ)+CONST*(CMPA-CPAS)*(1.-COELPIT  9
ISS)/DXI+CONST*CMPA*TCRF*SINS  ELPIT 10
RETURN  ELPIT 11
END  ELPIT 12

```

	SUBROUTINE VWASH(BARG,H,S,NVQR,X1,UINF,VZIP,XGAM,NGPI,DXI)	VWASH 1
	DIMENSION VZIP(30),XGAM(30)	VWASH 2
	DO 10 N=1,NGPI	VWASH 3
	DIFF=XGAM(N)-X1	VWASH 4
	SUM=0.	VWASH 5
	DO 5 K=1,NVQR	VWASH 6
	SUM=SUM+DIFF/(DIFF*DIFF+H)	VWASH 7
5	DIFF=DIFF-S	VWASH 8
10	VZIP(N)=VZIP(N)+SUM*BARG	VWASH 9
	RETURN	VWASH 10
	END	VWASH 11

	SUBROUTINE WASH(XGAM,NGAM,TIME,ALPH1,ALPH2,HEAVE,AROT,FREQF,PHIH,UWASH	1
	UINF,CAMBR,NF,VZIP,MOTR,INDV)	2
	DIMENSION XGAM(30),VZIP(30),CAMBR(24)	3
	NGPI = NGAM+1	4
	ANGLE = FREQF*TIME	5
	GO TO (108,120), INDV	6
108	GO TO (110,120),MOTR	7
110	CONST = -ALPH2*COS(ANGLE)*UINF+HEAVE*COS(ANGLE+PHIH)+ALPH1*UINF	8
	FACT = -ALPH2*FREQF*SIN(ANGLE)*UINF	9
	GO TO 130	10
120	CONST=UINF*ALPH1+HEAVE	11
	FACT=-UINF*ALPH2	12
130	DO 10 M=1,NGPI	13
	X=XGAM(M)	14
	THETA = ARCT(X)	15
	SUM=0.	16
	CCUNT=0.	17
	DO 20 N=1,NF	18
	COUNT=COUNT+1.	19
20	SUM=SUM+COUNT*CAMBR(N)*CCS(COUNT*THETA)	20
	IF(M-1) 2,4,2	21
2	IF(NGPI-M) 3,4,3	22
4	SUM = SUM + SUM	23
	GO TO 50	24
3	COUNT = 0.	25
	COTT = X/SIN(THETA)	26
	DO 30 N=1,NF	27
	COUNT = COUNT+THETA	28
30	SUM=SUM+COTT*CAMBR(N)*SIN(COUNT)	29
50	VZIP(M) = UINF*SUM+CONST+FACT*(AROT-X)	30
10	CONTINUE	31
	RETURN	32
	END	33

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

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

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

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

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

in which

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

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

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

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

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

where

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

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

$$D_4 = f_4 - C_4 \bar{x}^2 - g_a \bar{x} a_1 - g_b \bar{x} b_1$$

$$D_6 = 1 - C_6 \bar{x}^2$$

in which

$$h_a = \frac{M_{\beta\beta}}{R^2 M_{\theta\theta}} \quad h_b = \frac{M_{\phi\phi}}{M_{\theta\theta}}$$

$$g_a = 2 h_a A_1 \qquad g_b = 2 h_b A_2$$

$$C_4 = \bar{\omega}_{\phi}^2 h_a A_1^2 + \bar{\omega}_{\beta}^2 h_b A_2^2$$

$$C_6 = h_a A_1^2 + h_b A_2^2$$

$$a_1 = A_1 (\bar{\omega}_{\beta}^2 l_{s_1} + r_m \bar{\omega}_{\phi}^2 l_{s_2}) - B \bar{\omega}_{\phi}^2 l_{s_2}$$

$$b_1 = A_2 (\bar{\omega}_{\beta}^2 l_{s_1} + r_m \bar{\omega}_{\phi}^2 l_{s_2}) + B \bar{\omega}_{\phi}^2 l_{s_2}$$

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

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

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

$$t_1 = (1 - C_6 \bar{x}^2) B_2/B_6 - r_2 + \bar{\omega}_{\beta}^2 F + \frac{h_b F^2}{g_b^2 \bar{x}^2}$$

$$t_2 = (1 - C_6 \bar{x}^2)(B_2 - B_0/\bar{\omega}_\beta^2)/B_6 - f_2 + \bar{\omega}_\beta^2 F + f_0/\bar{\omega}_\beta^2$$

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

1. DuWaldt, F.; Gates, C.; and Piziali, R.: Investigation of Helicopter Rotor Blade Flutter and Flapwise Bending Response in Hovering. WADC Tech. Report 59-403, August 1959.
2. Brooks, G. W.; and Baker, J. E.: An Experimental Investigation of the Effects of Various Parameters Including Tip Mach Number on the Flutter of Some Model Helicopter Rotor Blades. NACA TN 4005, September 1958.
3. Gates, C. A.; Piziali, R. A.; and DuWaldt, F. A.: Comparison of Theoretical and Experimental Flutter Characteristics for a Model Rotor in Translational Flight. J. Amer. Helicopter Soc., Vol. 8, No. 2, April 1963, pp. 14-27.
4. Liiva, J.; et al: Two-Dimensional Tests of Airfoils Oscillating Near Stall, USAAVLAB Tech, Report 68-13A, April 1968.
5. Ericsson, L.; and Reding, J.: Dynamic Stall of Helicopter Blades. J. Am. Helicopter Soc., Vol. 17, No. 1, January 1972, pp. 11-19.
6. Tarzanin, F.: Prediction of Control Loads Due to Blade Stall. J. Am. Helicopter Soc., Vol. 17, No. 2, April 1972, pp. 33-46.
7. Crimi, P.; and Reeves, B. L.: A Method for Analyzing Dynamic Stall of Helicopter Rotor Blades. NASA CR-2009, May 1972.
8. Johnson, W.; and Ham, N. D.: On the Mechanism of Dynamic Stall. J. Am. Helicopter Soc., Vol. 17, No. 4, October 1972, pp. 36-45.
9. Sisto, F.: Stall Flutter in Cascades. J. Aero. Sci., Vol. 20, No. 9, September 1953, pp. 598-604.
10. Carta, F. O.; and Niebanck, C. F.: Prediction of Rotor Instability at High Forward Speeds, Vol. III, Stall Flutter. USAAVLABS Tech. Report 68-18C, February 1969.
11. Ham, N. D.; and Young, M. I.: Torsional Oscillation of Helicopter Blades Due to Stall. J. Aircraft, Vol. 3, No. 3, May-June 1966, pp. 218-224.

12. Reeves, B. L.; and Lees, L.: Theory of Laminar Near Wake of Blunt Bodies in Hypersonic Flow. AIAA J., Vol. 3, No. 11, November 1965, pp. 2061-2074.
13. Abbott, I. H.; and von Doenhoff, A. E.: Theory of Wing Sections. Dover, New York, 1959.
14. Crimi, P.: Stability of Dynamic Systems with Periodically Varying Parameters. AIAA J., Vol. 8, No. 10, October 1970, pp. 1760-1764.
15. Theodorsen, T.; and Garrick, I. E.: Mechanism of Flutter--A Theoretical and Experimental Investigation of the Flutter Problem. NACA TR 685, 1940.
16. Gessow, A.; and Meyers, G.: The Aerodynamics of the Helicopter. Ungar, New York, 1967.