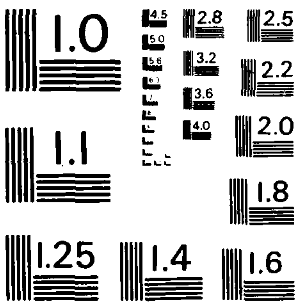
AD-A136 920 CONTINUED EXPERIMENTAL INVESTIGATION OF DYNAMIC STALL (U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF ENGINEERING S J SCHRECK DEC 83 1/2

UNCLASSIFIED AFIT/GAE/AA/83D-21 F/G 20/4 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

CONTINUED EXPERIMENTAL INVESTIGATION

OF DYNAMIC STALL

THESIS

Scott J. Schreck
First Lieutenant, USAF

AFIT/GAE/AA/83D-21

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

84 01 17 070

AFIT/GAE/AA/83D-21

CONTINUED EXPERIMENTAL INVESTIGATION
OF DYNAMIC STALL

THESIS

Scott J. Schreck
First Lieutenant, USAF

AFIT/GAE/AA/83D-21

Approved for public release; distribution unlimited

AFIT/GAE/AA/83D-21

CONTINUED EXPERIMENTAL INVESTIGATION OF DYNAMIC STALL

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Aeronautical Engineering

Scott J. Schreck, B.S.
First Lieutenant, USAF

December 1983

Acknowledgements

In performing the experiment and writing this thesis I have had a great deal of help from others. I would like to express my sincere gratitude to Maj. Eric Jumper, my thesis advisor, for his continual patience and assistance during the course of this study. I would also like to thank Dr. Peter Torvik and Lt. Col. Michael Smith for their valuable assistance. The technical expertise of Steve Coates in the areas of electronic hardware and computer software proved indispensable in this study, and the willing assistance given by Leroy Cannon and Harley Linville was also appreciated. Nick Yardich should be commended for his efforts in ensuring that the smoke tunnel remained operational, and Maj. Daniel Daley could be always counted on for practical advice concerning overall system operation.

Table of Contents

List of Figures

Figure		Page

Figure		Page

List of Tables

List of Symbols

$\dot{\alpha}$	angle of attack angular rate
$\dot{\alpha}_{ND}$	nondimensional angular rate
V_{∞}	freestream velocity
c	airfoil chord
$C_{l_{max\ dyn}}$	maximum dynamic lift coefficient
$C_{l_{max\ st}}$	maximum static lift coefficient
$\alpha_{stall\ dyn}$	dynamic stall angle of attack
$\alpha_{stall\ st}$	static stall angle of attack
c_p	pressure coefficient
p_{loc}	airfoil local static pressure
p_{∞}	freestream static pressure
ρ_{∞}	freestream density
Δp_{tran}	differential pressure sensed by transducer
p_a	ambient pressure
p_o	freestream stagnation pressure
ε_{SB}	solid blockage coefficient
ε_{WB}	wake blockage coefficient
$\Delta\alpha_{SC}$	streamline curvature correction
t	airfoil maximum thickness
h	test section height
mV	millivolts
psi	pounds per square inch
psig	pounds per square inch gauge

AFIT/GAE/AA/83D-21

Abstract

The flow over an NACA 0015 airfoil undergoing a constant rate of change of angle of attack was experimentally studied over a range of tunnel speeds and rotation rates. Surface-pressure transducers coupled with a microcomputer-based data acquisition system were used to collect surface-pressure data at the rate of 4000 samples per second. Data reduction was also microcomputer-based. ~~The~~ data was reduced in two forms. ~~First,~~ C_ℓ versus α curves through stall were determined for each dynamic experimental configuration. This was accomplished by numerical integration of ~~the~~ pressure data at a number of angles through stall, each data point representing the average of five experiments at the same experimental conditions. These curves indicated a slight decrease in $C_\ell - \alpha$ slope with increasing $\dot{\alpha}$. ~~Secondly, the~~ increase in stall angle of attack of the dynamic over the static case was ~~also~~ plotted against a nondimensional angular rate parameter (defined as the product of one-half the chord length and angular rotation rate, divided by the freestream velocity). This comparison gave rise to an apparently universal curve of nondimensional angular rotation rate versus increase in stall angle of attack. This curve was in agreement in some sense with previous experiments using stall indicators other than the actual stall. Data was collected in the range of nondimensional angular rates between .006 and .032.

CONTINUED EXPERIMENTAL INVESTIGATION OF DYNAMIC STALL

I. Introduction

Background

Dynamic stall is a physical phenomenon that occurs when an airfoil undergoes an uninterrupted, dynamic rotation through its static-stall angle of attack. Over a large range of angle-of-attack rate and freestream velocity combinations, the lift curve continues to increase beyond the static-stall point. Though this is a transient event, the momentary increase in maximum unstalled angle of attack yields a corresponding increase in the lift generated by the airfoil. This greater lift is of sufficient magnitude to render the dynamic stall effect of some possible practical use, and therefore worthy of further investigation.

The first formal investigation of dynamic stall was performed by Max Kramer in 1932, and was prompted by pilot reports of unexplained high lift values occurring in turbulent air (Ref 1:1). In his experiment, Kramer mounted a wing on a balance in the test section of a wind tunnel. He then used movable guide vanes, located upstream of the wing, to produce a rotating freestream flow in the test section. The resulting angle of attack experienced by the wing ranged from 0 to 30 degrees (Ref 1:2-3). Kramer repeated the experiment for three wings: the first two were Gottingen 459 airfoil cross-sections (a symmetric airfoil), with different chord lengths, and the third was a Gottingen 398 airfoil cross-section (a cambered airfoil).

Kramer's results showed a direct relationship between maximum lift coefficient and the rate of angle-of-attack change, $\dot{\alpha}$, and an inverse relationship to the test section velocity, V_∞. He introduced a non-dimensional angular rate parameter $\frac{c\dot{\alpha}}{V_\infty}$, where c is the chord length of the airfoil. This parameter allowed Kramer to collapse all his data onto a single curve given by:

$$C_{l_{max\ dyn}} = C_{l_{max\ st}} + 0.36 \frac{c\dot{\alpha}}{V_\infty} \qquad (1)$$

Though various attempts to model Kramer's experiment have been made, the first successful model was developed in 1980 by R. G. Docken, Jr., E. J. Jumper and J. E. Hitchcock (Ref 2). Their method involved applying the momentum-integral equation to a transient boundary layer control volume in unsteady potential flow. The method was then applied to the case of an 11% thick symmetrical Joukowski airfoil (J011), and yielded the result, for $\dot{\alpha}$ in radians per second:

$$\alpha_{stall\ dyn} = \alpha_{stall\ st} + 0.096 \frac{1/2c\dot{\alpha}}{V_\infty} \qquad (2)$$

As it stands, Eq. 2 is not directly comparable to Eq. 1. It can be transformed to an equivalent version, which is directly comparable, by assuming that the only difference between the static and dynamic lift curves is the increased stall angle of attack for the dynamic case. This implies that both static and dynamic lift curves have the same slope. Correcting this slope for the aspect ratio of Kramer's wing, Eq. 3 is obtained, which is directly comparable to Eq. 1 (Ref 2:3):

$$C_{l_{max\ dyn}} = C_{l_{max\ st}} + 0.301 \frac{c\dot{\alpha}}{V_\infty} \tag{3}$$

Comparison of Eq. 1 and Eq. 3 reveals a remarkable agreement between Kramer's experimental findings and the analytical result of Docken, et. al. The fact that Kramer used a Gottingen 459 and Docken a J011 detracts little, if at all, from the credibility of the comparison, since both are symmetrical airfoils and have similar leading edge geometries.

Since the Kramer experiment, which was the original investigation in dynamic stall, a great deal of research, both analytical and experimental, has been done in this area. However, the preponderance of work, unlike Kramer's, has been for the case of an airfoil undergoing a dynamic angle-of-attack change in a constant-direction freestream, and most of this work has been done for the case of a sinusoidally oscillating airfoil. The reasons for this are fairly obvious. Situations where dynamic stall research would likely find an application are those such as helicopter blades, turbomachinery, and aircraft wing-flutter, and the angle of attack variations displayed in cases such as these are likely to be sinusoidal, or at least approximately so.

The advent of digital flight control systems, however, promises an application for the case of an airfoil which undergoes an angle of attack variation that can be described by a ramp function. A distinct advantage of the ramp angle-of-attack variation is the comparative ease and physical clarity with which a mathematical model may be developed, as opposed to the case of the sinusoidally oscillating airfoil. The mathematical model for the sinusoidal case lies in the realm of full Navier-Stokes solutions, and amounts to little more than a numerical experiment. While such an approach succeeds fairly well in modelling the results of a corresponding experiment, the sheer mathematical complexity overwhelms any attempt to truly understand the physics of the phenomenon.

In 1979, Deekens and Kuebler (Ref 3) undertook an investigation of dynamic stall in which the effects of constant airfoil angular rate on dynamic-stall angle of attack were evaluated. Smoke-trace flow visualization was employed in conjunction with simultaneous movie filming in order to characterize the dynamic stall phenomenon on an NACA 0015 airfoil, which was rotated about its midchord in a constant-velocity freestream. They concluded that the increase in unstalled angle of attack above that for static stall was directly related to airfoil angular rate, and inversely related to the freestream velocity. Moreover, they were able to predict the dynamic stall angle of attack within the scope of their experiment, which was the Reynolds number range between 14,500 and 32,500.

Introducing the nondimensional angular rate parameter, as did both Kramer and Docken, et. al., Deekens and Kuebler were able to collapse their data onto a single curve given by:

$$\alpha_{stall\ dyn} = \alpha_{stall\ st} + 2.5\,\frac{1/2c\dot{\alpha}}{V_\infty} \tag{4}$$

A plot of these results, showing dynamic stall angle of attack as a function of nondimensional rotation rate parameter is shown in Fig. 1, on the following page.

By now introducing the same assumptions and corrections as were used previously in transforming Eq. 2 into Eq. 3, Eq. 5 is arrived at, which is directly comparable to Eq. 1 and Eq. 3.

$$C_{l_{max\ dyn}} = C_{l_{max\ stat}} + 4.8\,\frac{\dot{\alpha}}{V_\infty} \tag{5}$$

Comparing Eq. 1 and Eq. 3 with Eq. 5, it is immediately obvious that the dynamic lift curve slope implied by Deekens and Kuebler is significantly greater than that given by the work of either Kramer or Docken, et. al. An error, either experimental or computational, of such a magnitude to explain this apparent discrepancy can be ruled out since the results of Deekens and Kuebler are substantiated by the work of Francis (Ref 4), and by Scheubel, as well (Ref 5:1-4). In

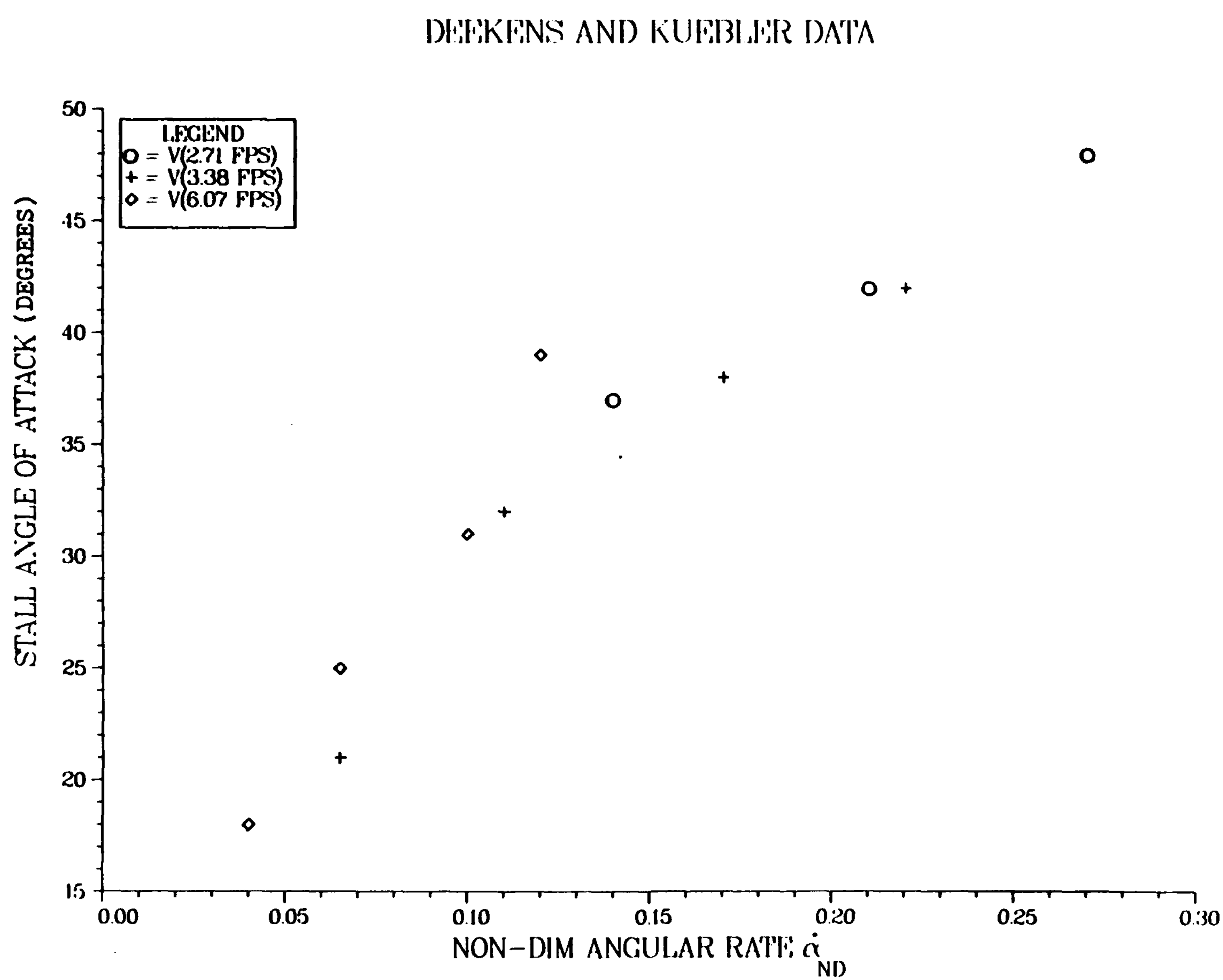


Figure 1. Data Summary of Deekens' and Kuebler's Results

addition, Kramer's work seems to have been verified in an experiment mentioned by Scheubel (Ref 5:1).

At this point it might be profitable to pause for a moment to point out and emphasize an important distinction. With Kramer's experiment configured as previously described, the airfoil was fixed in inertial space and encountered a gust. Therefore, a mathematical model of the flow over the airfoil could justly assume a Newtonian, or nonaccelerating, control volume. However for a constant-velocity freestream with the airfoil being rotated dynamically therein, the airfoil is rotating with respect to inertial space. In this case, mathematical analysis of the flow over the airfoil cannot be carried out using a Newtonian control volume. The previously mentioned order-of-magnitude disagreement between Kramer's results and those of Deekens and Kuebler could, therefore, be due to the effect of the accelerating control volume.

The results of Deekens and Kuebler were recently reconfirmed by Daley. Like Deekens and Kuebler, Daley rotated an NACA 0015 airfoil section about its midchord at a constant angular rate in a constant-velocity freestream. He also used smoke-trace flow visualization along with high-speed cinematography as a medium for recording and later analyzing his results. Daley added a new dimension to the experiment, however, by embedding four piezo-resistive pressure transducers in the airfoil around its quarter-chord. This modification enabled him to simultaneously gather two types of data regarding the same phenomenon. Using both movies and electronically-gathered pressure information, Daley possessed an extremely accurate and sensitive indicator of flow separation at the quarter-chord. Adopting quarter-chord flow separation as the criterion for stall, he then proceeded to verify a major portion of the results obtained by Deekens and Kuebler. He also extended the range of results into the region of lower nondimensional angular rate, as shown in Fig. 2, and, at the same time, expanded the Reynolds number range of the experiment.

Objectives

The previous research in dynamic stall for the constant rate angle-of-attack change provided a foundation on which to base further investigation. In view of this previous work, the objectives of this experiment were as follows:

1. Equip an NACA 0015 airfoil with 16 piezo-resistive pressure transducers, and integrate this with an automated, microcomputer-controlled data collection system for gathering time, angle-of-attack, and pressure information.
2. Develop a FORTRAN software package to automate both the acquisition and reduction of time, angle-of-attack, and pressure data.
3. Verify the operation of this system, and then use it to conduct a parametric investigation of the effects of airfoil angular rate and freestream velocity on dynamic stall.

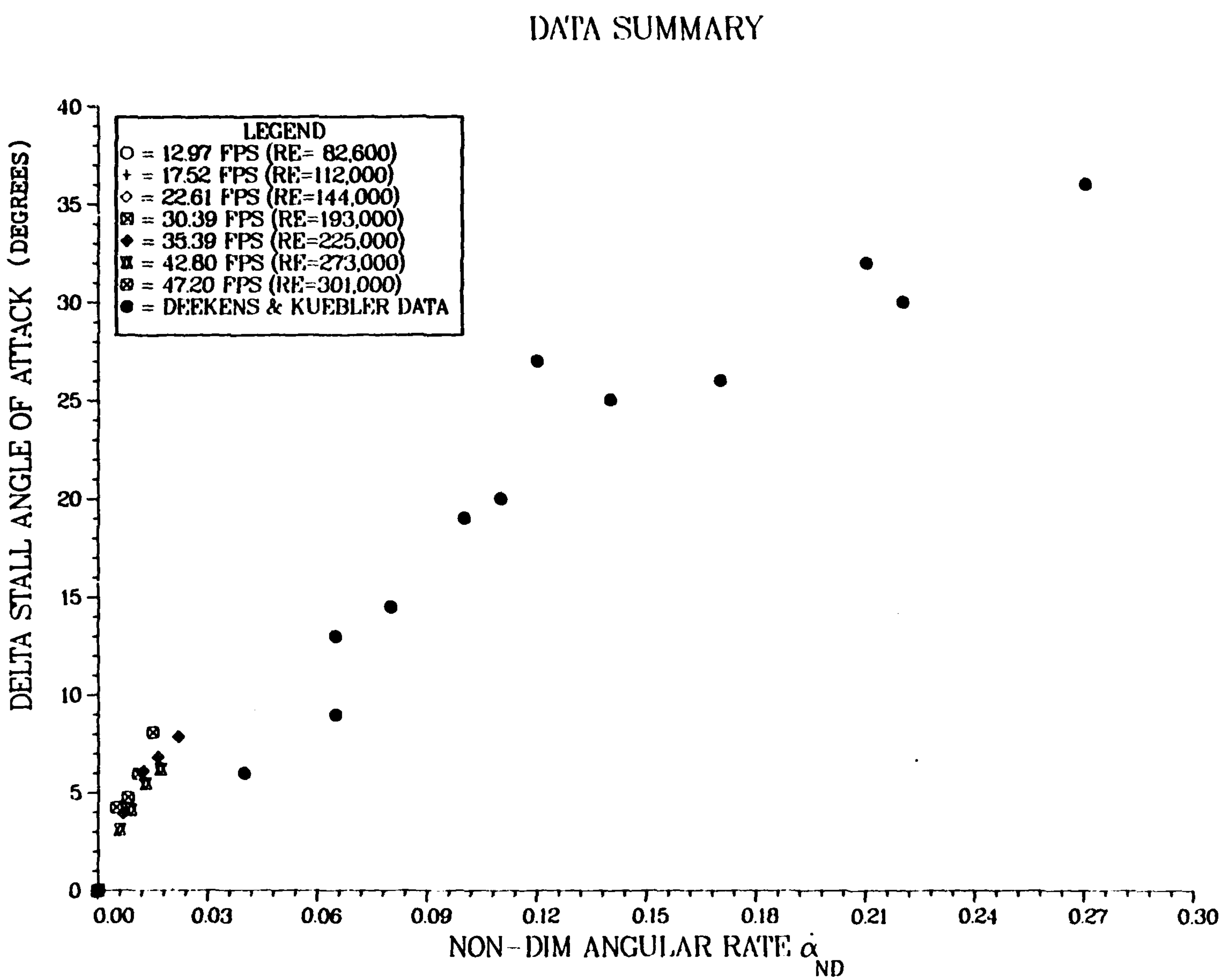


Figure 2. Data Summary of Daley's Results

II. Theory and Approach

The following theory and approach section is composed of six subsections. Each of these subsections, in turn, presents a brief discussion of the way in which existing dynamic stall theory or previous research influenced the experimental approach in the current investigation. The first subsection provides a more detailed qualitative description of dynamic stall than was given in the introduction. The second discusses the determination of pressure coefficients using ensemble-averaged data. Then, the third subsection covers discretization of the pressure distribution defined by these pressure coefficients, while the fourth describes the integration of this discretized pressure distribution. The fifth subsection considers the computation of force coefficients using the results of the integration, and the sixth presents a brief narrative concerning the problem of data acquisition.

Dynamic Stall in Contrast to Static Stall

Stall, whether static or dynamic, is defined as having occurred when the boundary layer has separated from the airfoil to such a degree that the resulting pressure distribution ceases to yield any increase in lift for further angle-of-attack increase. Obviously, the boundary layer interactions for static and dynamic stall must differ significantly to produce the dramatic dissimilarities previously mentioned in the introduction.

In the familiar case for static stall, a boundary layer under the influence of an adverse pressure gradient will eventually separate from the surface of the airfoil at the point where the shear stress at the wall vanishes. The point where flow separates and where it reverses are coincident for static stall. Thus, the wake formed by this viscous interation is large, and appreciably distorts the potential flow field around the airfoil. For the static case, the stall angle of attack is relatively constant, being, at most, a weak function of Reynolds number (Ref 6:248).

In dynamic stall, as with static stall, the boundary layer under the influence of an adverse pressure gradient also eventually reaches the condition of vanishing wall shear. The similarity ends here, however, as the point of reversed flow is no longer coincident with the point of zero wall shear, but is delayed for some distance downstream of the zero-shear point. The point of separation for the case of dynamic stall is determined by the MRS (Moore-Rott-Sears) criterion (Ref 7:113-144). This difference substantially reduces the wake size and corresponding potential flow field distortion in comparison to the static case (Ref 8:2-3). In addition to the MRS separation criterion, other effects appear to be at work, as well (Ref 15). Thus, it is clear that the dynamic stall angle-of-attack is a complex function of freestream velocity, airfoil angular rate, and even airfoil section geometry.

The fact that the wake size and corresponding potential flow field distortion associated with dynamic stall are small relative to their static-stall counterparts is favorable to the present study. Tunnel wall interference effects are therefore correspondingly small compared to those encountered in the same flow regime for steady-state phenomena. This implies that the effects of streamline curvature and wake blockage can probably be considered negligible in dynamic stall testing. A more detailed discussion of these test section wall effects is presented in Appendix A.

<u>Determination of Pressure Coefficients</u>

Because of inevitable freestream irregularities in the test section, pressure measurements taken at the same location on the airfoil, but at different times, are not constant. These irregularities can be filtered out using ensemble-averaging which, at the same time, preserves those pressure fluctuations due only to the dynamic stall phenomenon. For this experiment, pressure data from five different airfoil rotations at the same angular rate and freestream velocity were averaged to yield an ensemble-averaged data set.

Using this ensemble-averaged pressure data, a method of calculating the pressure coefficient at any chord location on the airfoil must be found. This method should use as its inputs physical parameters which can be readily sensed or measured. The standard equation for the computation of pressure coefficient is given by:

$$c_p = \frac{p_{loc} - p_\infty}{1/2\rho_\infty \; V_\infty^2} \tag{6}$$

where p_{loc} is the local static pressure at some location on the airfoil, p_∞ is the freestream static pressure, and ρ_∞ and V_∞ are the freestream density and velocity, respectively. The local pressure anywhere on the airfoil can be expressed as:

$$p_{loc} = \Delta p_{tran} + p_a \tag{7}$$

where p_{loc} retains the same definition it had in Eq. 6, p_a is some reference pressure, and Δp_{tran} is the differential pressure between these two. Substituting p_{loc} from Eq. 7 into Eq. 6 yields the relationship:

$$c_p = \frac{(\Delta p_{tran} + p_a) - p_\infty}{1/2\rho_\infty \; V_\infty^2} \tag{8}$$

Regrouping the terms in the numerator and noting that the denominator is equivalent to $p_o - p_\infty$ under the incompressible Bernoulli equation, Eq. 8 becomes:

$$c_p = \frac{\Delta p_{tran} + (p_a - p_\infty)}{p_o - p_\infty} \tag{9}$$

Eq. 9 requires that three quantities be known in order to determine the corresponding pressure coefficient. The first, Δp_{tran}, is the difference in pressure between some constant reference pressure, p_a, and the pressure at a certain point on the surface of the airfoil. This differential pressure was sensed by a transducer mounted in the airfoil. The second, $p_a - p_\infty$, is the pressure difference between the reference pressure and test section static pressure, while the third, $p_o - p_\infty$, is the pressure difference between test section stagnation pressure and static pressure. Since the reference pressure must be easily accessible as well as constant, ambient room pressure constituted a good choice, although any constant, easily accessible pressure source would have been acceptable.

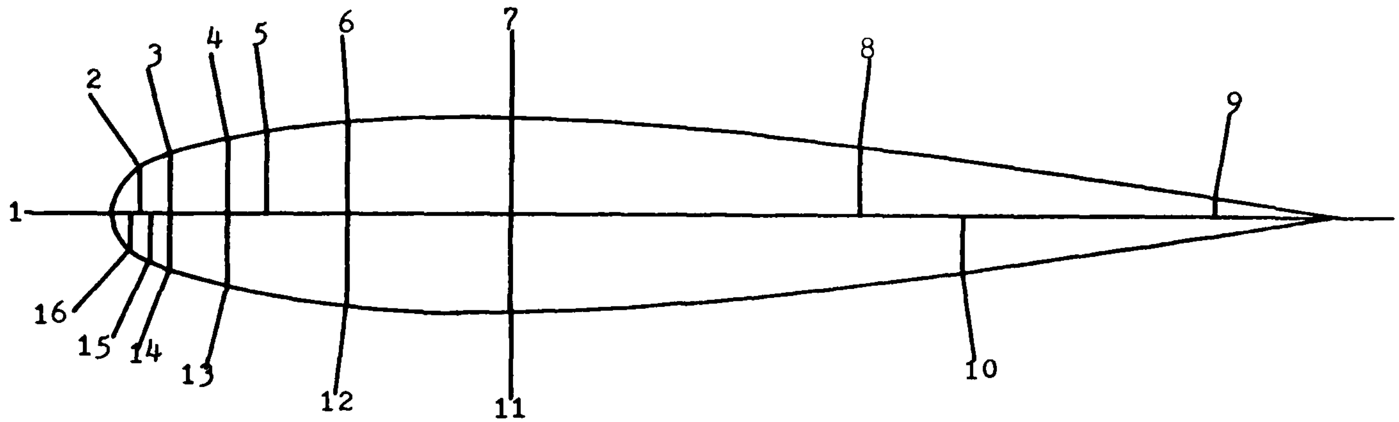
Discretization of the Pressure Distribution

The mathematical procedure developed in the preceding subsection facilitates pressure coefficient determination at any point on the airfoil where a transducer is located. Hence, to minimize the error inherent in discretizing the dynamic stall pressure distribution, two basic issues must be addressed. First, an acceptable number of transducers must be determined and, second, the optimum distribution of these transducers has to be established.

Obviously, a greater number of transducers yields a reduced discretization error. However, an upper limit on this number is eventually reached, and is dictated mainly by practical considerations. In this experiment, 16 transducers were used, the number being limited chiefly by the capacity of the electronic data acquisition system (c.f. below).

Placement of these 16 transducers on the airfoil was governed by the requirement that the resulting discretized pressure distribution portray as accurately as possible the actual pressure distribution. Thus, the transducers were concentrated in the region of the airfoil where the dynamic stall pressure distribution was anticipated to have the largest gradient. The pressure distribution obtained by McAlister, Carr and McCroskey (Ref 9:51) for an NACA 0012 airfoil oscillating at a nondimensional angular rate comparable to that for the present study was useful here as a guide. Accordingly, the transducers were distributed most densely on the upper surface of the airfoil near the leading edge. Fig. 3 illustrates the transducer placement chosen to minimize the error inherent in discretizing the anticipated pressure distribution.

As shown in Fig. 3, there is no transducer at the trailing edge of the airfoil and, as a consequence, no way to determine the pressure at the trailing edge. However, the results of McAlister, et. al., were obtained using an airfoil that had a pressure transducer located at the 98 percent chord position (Ref 9:51). Again using their results (Ref 9:51) as a guide, it appears that the trailing edge pressure coefficient can be approximated in this experiment by using a linear curve fit involving the rearmost two transducers on the upper surface of the airfoil.



x/c (1) = 0.000	x/c (9) = 0.902
x/c (2) = 0.025	x/c (10) = 0.697
x/c (3) = 0.049	x/c (11) = 0.328
x/c (4) = 0.098	x/c (12) = 0.197
x/c (5) = 0.131	x/c (13) = 0.098
x/c (6) = 0.197	x/c (14) = 0.049
x/c (7) = 0.328	x/c (15) = 0.033
x/c (8) = 0.615	x/c (16) = 0.016

Figure 3. Upper and Lower Surface Pressure Transducer Locations

Integration of the Pressure Distribution

A numerical integration was performed on the discretized pressure distribution to obtain the corresponding force coefficients. McAlister, et. al., found that cubic and variable power splines applied to the discrete data points did not yield acceptable accuracy. The spline fits caused large overshoots that made this method unsatisfactory in general application (Ref 9:12). Therefore, following McAlister, et. al., all integrations in this investigation were based on the trapezoidal rule.

Determination of Force Coefficients

The body axis normal force can be decomposed into its constituent lift and drag forces in the wind axis system, and the drag force can be further separated into pressure drag and skin-friction drag. The integration of the pressure distribution properly accounts for the lift and pressure drag forces, but is incapable of predicting the skin-friction drag (a viscous effect). This is not a serious source of error, however, for the following reasons.

At low angles of attack, where flow remains attached over most of the airfoil and skin friction is the major source of drag, the angle between the normal force vector and the drag force vector is large. Since this angle is large, the projection of the drag force vector onto that of the normal force is correspondingly small, thereby making any error in the drag force negligible with respect to the magnitude of the normal force vector.

At higher angles of attack, the projection of the drag force vector onto the normal force vector and the normal force vector, itself, are of comparable magnitude. In this case, however, the flow is no longer fully attached to the airfoil. As a result, skin-friction drag is overshadowed by pressure drag. Thus, only the drag force coefficient would be significantly in error, and even then, only at low angles of attack.

To accurately take into account skin-friction drag, direct force measurement could be used. The preferred method, however, still consists of integrating the pressure distribution on the airfoil. Proceeding in this manner not only yields the normal force coefficient, but characterizes the pressure distribution as a necessary intermediate step. This characterization, in turn, constitutes an indispensable tool in the analysis of the flow over the airfoil as it undergoes dynamic stall.

The Problem of Data Acquisition

Measurement of the physical parameters associated with dynamic stall presents a unique problem, due to the transient nature of the phenomenon. The measurement system used must be not only accurate, but fast, as well. The solution to this problem has taken many forms, beginning with Kramer, who used a balance system to measure the aerodynamic forces on the wing as the freestream flow was rotated past it.

Deekens and Kuebler used high-speed cinemaphotography of smoke traces to ascertain airfoil rotation rate and the angle of attack at which dynamic stall occurred. Daley, again, used movies of smoke traces, but simultaneously gathered digital electronic position and pressure data using four electronic transducers embedded in the quarter-chord region of the airfoil. McAlister, et. al., used an airfoil equipped with 16 pressure transducers, and collected analog electronic position and pressure data.

In the present study, digital position and pressure information was collected using an airfoil instrumented with 16 pressure transducers (c.f. above). To gather digital information for the purpose of later constructing dynamic lift curves, it was insured that the sample rate was high enough to yield the resolution required for such a purpose. The lower threshold on sample rate, in this case, was approximately 300 data samples per second (Ref 10:7). The data acquisition system used in the present study had the capability to meet and exceed this criterion by a wide margin (c.f. below).

III. Facilities and Instrumentation

Smoke Tunnel

The present study was conducted in the AFIT smoke tunnel located in Building 640, Area B, Wright-Patterson AFB, Ohio. The test section is 59 inches long, 39.5 inches high, and 2.75 inches deep. This facility is capable of test section velocities between, approximately, 10 and 50 feet per second. The smoke tunnel and its capabilities are further described by Sisson (Ref 11), and Baldner (Ref 12).

Velocity Measurement

Test section static and total pressure were measured using a standard hemispherical-head Pitot-static probe in conjunction with a Meriam A-937 water micromanometer. These pressures were used to establish the test section velocity during data collection and, later, to determine the pressure coefficients when reducing the data. This made the accurate measurement of these pressures a crucial step in the experiment. It was therefore necessary to locate the probe as close to the airfoil as possible, without creating mutual interference between the two, due to the presence of a significant longitudinal static pressure gradient (see Appendix B).

Considering this, the probe hole was drilled into the floor of the tunnel at a location 31 inches downstream of the point where the test section begins. With the airfoil at the zero angle-of-attack position, the tip of the probe was directly below the leading edge of the airfoil, and the probe static ports were coplanar with the 13 percent chord station on the airfoil.

Airfoil

The NACA 0015 airfoil used in this experiment measured 12.2 inches chord and 2.63 inches span. It consisted of a hollow mahogany shell closed on both sides by aluminum endplates, which were sealed to the shell with silicone rubber adhesive sealant. To reduce the frictional effects of the tunnel walls on the rotating airfoil, Teflon bearing surfaces were fitted to both the front and rear endplates. The rear endplate was rigidly attached to a 14 inch tubular aluminum shaft with an outside diameter of .75 inches. This aluminum shaft was slotted at its midpoint to admit ambient atmospheric pressure to the interior of the airfoil, as well as the transducer leads. Sixteen ports were bored into the wooden shell using a number 38 drill, to accept the 16 transducers (c.f. below).

Transducers

The transducers chosen for use in this experiment were Endevco 8506-2 and 8507-2 miniature piezo-resistive pressure transducers, the only difference between the 8506 and 8507 being that the 8506 had a threaded mounting. Both types of transducers had a maximum range of plus or minus two psig, and required an excitation voltage of 10.00 volts DC. Excitation voltage was provided by a Hewlett-Packard 6205B Dual DC Power Supply, and monitored by a Hewlett-Packard 34701A DC voltmeter with a 34740A digital display insert, which allowed voltage to be read to three decimal places. Resonance frequency for both types of transducers was 45,000 Hertz. Thus, the frequency response of the transducer had a negligible effect on the results obtained in this experiment.

The transducers were mounted in the ports in the airfoil according to the specifications provided by Endevco (Ref 13). General Electric RTV silicone rubber adhesive sealant was used here as a bonding agent. After completing the electrical connections between the transducers and the microcomputer, the transducers were recalibrated (c.f. below). In many cases, the sensitivities obtained in the recalibration were significantly different from those provided by Endevco.

Drive Mechanism

The airfoil was rotated using a TRW Globe Model 5A2298-4 12 volt DC, constant-speed planetary gearmotor with a 525:1 reduction ratio. The attributes of this unit gave it two definite advantages over the alternatives considered. First, its power requirements could be met using a standard laboratory bench power supply, which, in this experiment, was the Hewlett-Packard 6205C Dual DC Power Supply. By adjusting the supply voltage between 6 and 12 volts, the rotation rate at the output shaft could be varied between approximately 30 and 95 degrees per second. Second, the high reduction ratio gave a high output torque. This high output torque, in turn, spun the output shaft up to constant speed in less than .01 seconds, which was negligible compared to the time required to rotate the airfoil through the dynamic stall angle of attack. Thus, airfoil angular rate was easily governed and, once the motor voltage was set, the variation in the output angular rate was negligible. The test case was therefore easily repeatable. A spring-loaded double-pole, double-throw toggle switch was used to rotate the airfoil, stop it, and then return it to the initial position in preparation for the next run.

The airfoil angle-of-attack transducer consisted of a Helipot #7216 1000 ohm, ten-turn potentiometer. The potentiometer was coupled to the airfoil shaft through a gear train having a 33:1 ratio, giving a full ten turns at the potentiometer for 109.1 degrees airfoil rotation. The potentiometer was excited at 10 volts, simultaneously using the same voltage source as used for transducer excitation, and its output was fed to one of two analog-to-digital conversion boards in the computer (c.f. below).

Data Acquisition System

The microcomputer system consisted of a Heathkit H-19 terminal, a Tarbell Model VDS-IID dual eight-inch floppy disk drive, and an Electronic Control Technology S-100 bus equipped with an SD Systems SBU-100 Single Board Computer, an SD Systems Expandoram II board, and an MD2022 Tarbell Disk Controller board. To perform its digital data gathering function, this system was augmented with two Dual Systems Control Corporation AIM-12 analog input module boards.

The AIM-12 is a high speed, multiplexed analog-to-digital data acquisition module compatible with the standard S-100 bus. The analog-to-digital conversion subsystem on the board can be operated in one of two modes. The unipolar mode requires that the input voltage to the A/D converter be between 0 and 10 volts, while the bipolar mode accepts input voltages from -5 volts to +5 volts.

Since the analog signal to be digitized will usually not give full-scale deflection in either of these modes, the board has a preconditioning subsystem consisting of a multiplexed, precision instrumentation amplifier. By changing the resistance in the feedback loop of the amplifier, gains from 1 to 100 can be selected. Single-ended amplifier operation allows 32 separate analog inputs to the multiplexer, while differential operation limits the multiplexer to 16 inputs. Differential operation, however, takes advantage of the high common mode rejection ratio of the amplifier, which is 80 decibels at a gain of 1 and increases to 114 decibels for the maximum gain of 100.

As mentioned previously, two of these AIM-12 boards were used in the data acquisition system. The board responsible for the collection and digitization of the electrical signals from the 16 pressure transducers was configured for gain 100 amplification in the differential mode, and bipolar A/D conversion. The electrical signals originating at the pressure transducers had a magnitude of, approximately, 15 millivolts. Although the gain of 100 resulted in no more than 30 percent of full-scale on the A/D converter, the high common mode rejection ratio was very effective in cancelling noise in the system and the resulting overall system accuracy was quite good.

Another point that should be discussed is the reason for choosing bipolar A/D conversion. The highest pressure that will be experienced in any flow condition is that at stagnation. Since the smoke tunnel is of the drawdown variety, the stagnation pressure in the test section is the same as ambient pressure if inlet losses are assumed negligible, and lower if they are taken into account. Therefore, it is reasonable to assume that the highest pressure that could possibly occur in the test section is somewhat lower than ambient pressure. The transducers in the present configuration sense the difference between some local pressure on the airfoil and ambient, or room, pressure. With the highest pressure in the test section being somewhat lower than ambient pressure, it seems that the transducers would output only negative voltages and the unipolar A/D conversion mode could be used, thereby cutting the worst case error in half. However, due to zero drift, the transducers did, in some instances, output a voltage in the positive range, obviating the need to use the bipolar A/D conversion mode.

The other AIM-12 board was responsible for the collection and digitization of the signal from the position potentiometer, which varied between 0 and 10 volts. This board was configured for gain 1 amplification and unipolar A/D conversion.

IV. Experimental Procedure

Transducer Calibration

All 16 transducers in the airfoil were calibrated prior to the first data collection run. This calibration procedure was repeated every three weeks until data gathering was complete. A more complete description of transducer calibration is presented in Appendix C.

Data Collection

To prepare the system for a data collection run, all three voltmeters, both power supplies, and the computer were allowed to warm up at for a minimum of four hours before any data was taken. This was done to allow any large electrical transients in the system to die out, insuring steady-state operation during data collection.

The first step in making a data collection run was to execute the data acquisition program, BIG (see Appendix D). The remainder of the experimental procedure was then automatically indicated at the computer terminal by requests for input, or as simple instructions concerning equipment operation. The following discussion constitutes a summary of the data collection sequence.

The first set of inputs to the computer consisted of date, time, temperature, and barometer. These were then echoed back to the operator for verification and, if desired, corrections to these inputs could then be made before they were written to the disk file. Next, the zero-input readings for the 16 transducers were taken, displayed on the terminal screen, and written to disk. At this point the operator was instructed by the computer to turn on the tunnel motors and bring the test section flow up to the desired velocity.

The next set of inputs consisted of the two different micromanometer readings, the voltages corresponding to the 90 and 0 degree angle-of-attack positions, and the voltage that was to be supplied to airfoil drive motor. The first micromanometer reading was the difference between ambient pressure and test section static pressure. This was obtained by connecting the tube from the Pitot-static probe static ports to one leg of the micromanometer and leaving the other leg open to the ambient air. The second micromanometer reading was the difference between the test section total and static pressures, and was obtained by connecting the tube from the probe total pressure port to the other leg of the micromanometer.

The voltages corresponding to 90 and 0 degrees angle-of-attack were measured using a digital voltmeter connected to the position potentiometer during this phase of the data collection run. The 90 and 0 degree angle-of-attack positions were indicated by markers attached to the front glass wall of the test section. After these voltages were input to the computer, the motor voltage was input, and all values input in this set were echoed at the terminal screen for verification. The operator then had the option to write these values to disk or to enter them again.

The next part of the program did the actual dynamic stall data collection for five consecutive airfoil rotations at one motor voltage and one test section velocity. The following description is representative of the sequence that the operator performed for one such airfoil rotation.

The operator would first input the number of samples to be taken, as well as his choice of manual or automatic trigger. The number of samples and choice of trigger remained consistent over the five consecutive rotations to avoid difficulty in data reduction. After the airfoil had been rotated through dynamic stall and returned to zero angle of attack, the computer would output, to the terminal, the number of samples actually taken and the angular rotation rate of the airfoil in degrees per second. At this point, the operator had the option to write the data set just collected to disk, or to repeat the rotation to get a new data set. The rotation was repeated if the rotation rate was not within two degrees per second of the angular rates for the rotations previously completed at the same test section velocity and motor voltage. If it did meet this criterion, the data set just collected was written to disk.

After five satisfactory dynamic stall data files had been obtained, a static lift curve was determined for the same test section velocity. The next part of the program did this by first instructing the operator to rotate the airfoil to the desired static angle of attack. Then, at the command of the operator, the transducers were sampled the specified number of times, and the resulting normal force coefficient was computed and displayed at the terminal. This value was recorded and the procedure was repeated a sufficient number of times at successively higher angles of attack to define a static lift curve. The static angle of attack was read from a clear plastic protractor taped to the front wall of the test section. When the static lift curve had been determined to the satisfaction of the operator, the data collection program, BIG, was terminated, and the tunnel shut down until the next run.

Velocities and Reynolds Numbers

Using the procedure outlined above, test runs were conducted at five test section velocities between 26 and 48 feet per second, inclusive. Although the smoke tunnel was capable of test section velocities as low as 10 feet per second, any data gathered at velocities below approximately 25 feet per second was not of acceptable quality for two reasons. First, the magnitude of the resulting signal was of low enough magnitude to render the resulting signal-to-noise ratio unacceptable. Second, the percent full-scale deflection at the analog-to-digital converter was small enough to make the resulting analog-to-digital resolution unacceptable. At each test section velocity, a run was accomplished for each of four motor voltages, giving a total of 20 test runs. The resulting Reynolds numbers, based on airfoil chordlength, ranged from 1.58×10^5 to 2.81×10^5. As such, all data was collected in a flow regime generally considered laminar, based on Reynolds number.

V. Data Reduction and Discussion of Results

Data Reduction

The raw data files generated during the test runs and stored on disk were later reduced using the data reduction program RED, a copy of which is included in Appendix D. This program first ensemble averaged the five data files (all five of which were gathered at one test section velocity and one motor voltage), then converted the averaged digital voltages to angles of attack and pressures. These pressures were converted to pressure coefficients which were, in turn, used to construct a pressure distribution. This pressure distribution was then integrated using the trapezoidal rule, which gave a dynamic normal force coefficient for a corresponding angle of attack. RED reduced every tenth data set, a data set being the elapsed time, position voltage, and the voltages from the 16 transducers which were collected in one sampling pass.

Since RED reduced every tenth data set, another program was needed to better resolve the point at which dynamic stall actually occurred. This program, DEN, reduced every data set within a specified interval on either side of the anticipated dynamic stall angle of attack. The anticipated dynamic stall angle of attack was obtained by first running RED and finding the approximate angle of attack at which dynamic stall occurred. A copy of DEN is included in Appendix D. Programs RED and DEN both include comments to facilitate understanding of the data reduction routines used.

Discussion of Results

The reduced data obtained using RED and DEN was plotted, giving a dynamic lift curve (that is, for $\dot{\alpha} \neq 0$) for each of the twenty different combinations of motor voltage and test section velocity. A representative set of these plots for 34.4 feet per second test section velocity and the full range of four motor voltages is presented in Figs. 4 through 7. A static lift curve for the same test section velocity is also included in this set, in Fig. 8. Appendix E contains the remainder of the plotted results and the associated static lift curves.

Comparison of the static lift curve in Fig. 8 with the one given in Ref 6 for a NACA 0015 at a similar Reynolds number leads to the following three observations. First, the slope of the curve obtained in this experiment is slightly lower than that shown in Ref 6. Second, the character of the curve in the range between eight and fourteen degrees angle of attack differs somewhat from the one in Ref 6. Finally, the static-stall angle of attack is approximately six degrees higher than would be expected based on Ref 6.

The difference in lift curve slope could be due, in part, to the influence of three-dimensional effects on the airfoil. One source of three-dimensional effects was the

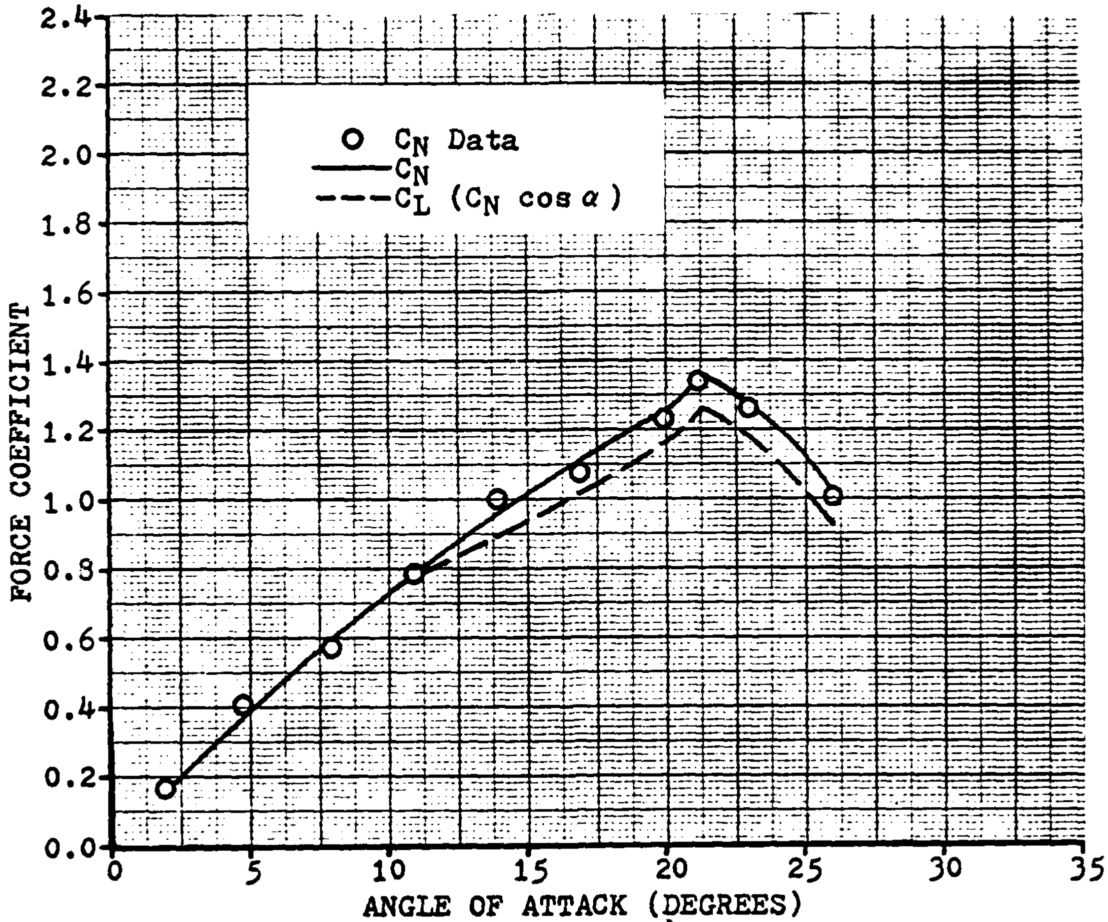


Figure 4. Force Coefficient vs. Angle of Attack for $\dot{\alpha}=32.6$ Deg/Sec and $V_\infty=34.4$ Ft/Sec ($\dot{\alpha}_{ND}=0.009$)

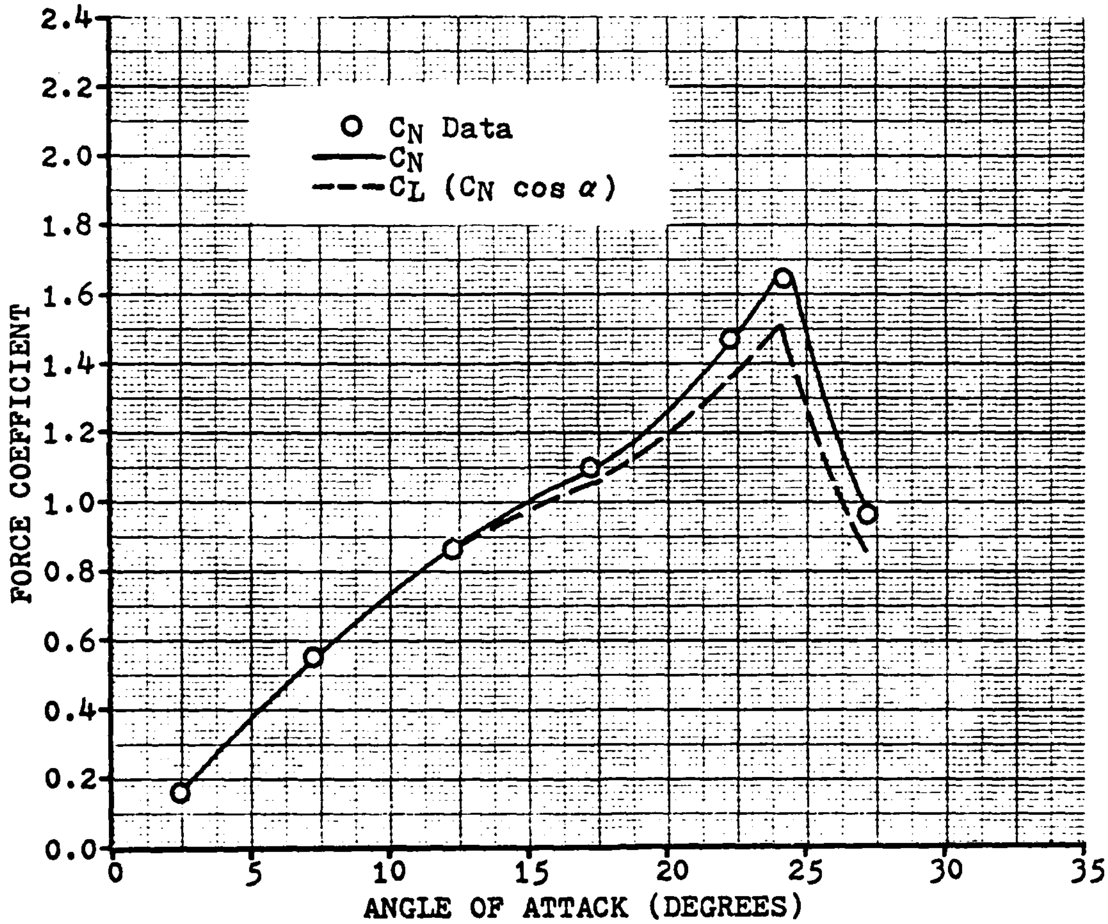


Figure 5. Force Coefficient vs. Angle of Attack for $\dot{\alpha}=52.7$ Deg/Sec and $V_\infty=34.4$ Ft/Sec ($\dot{\alpha}_{ND}=0.014$)

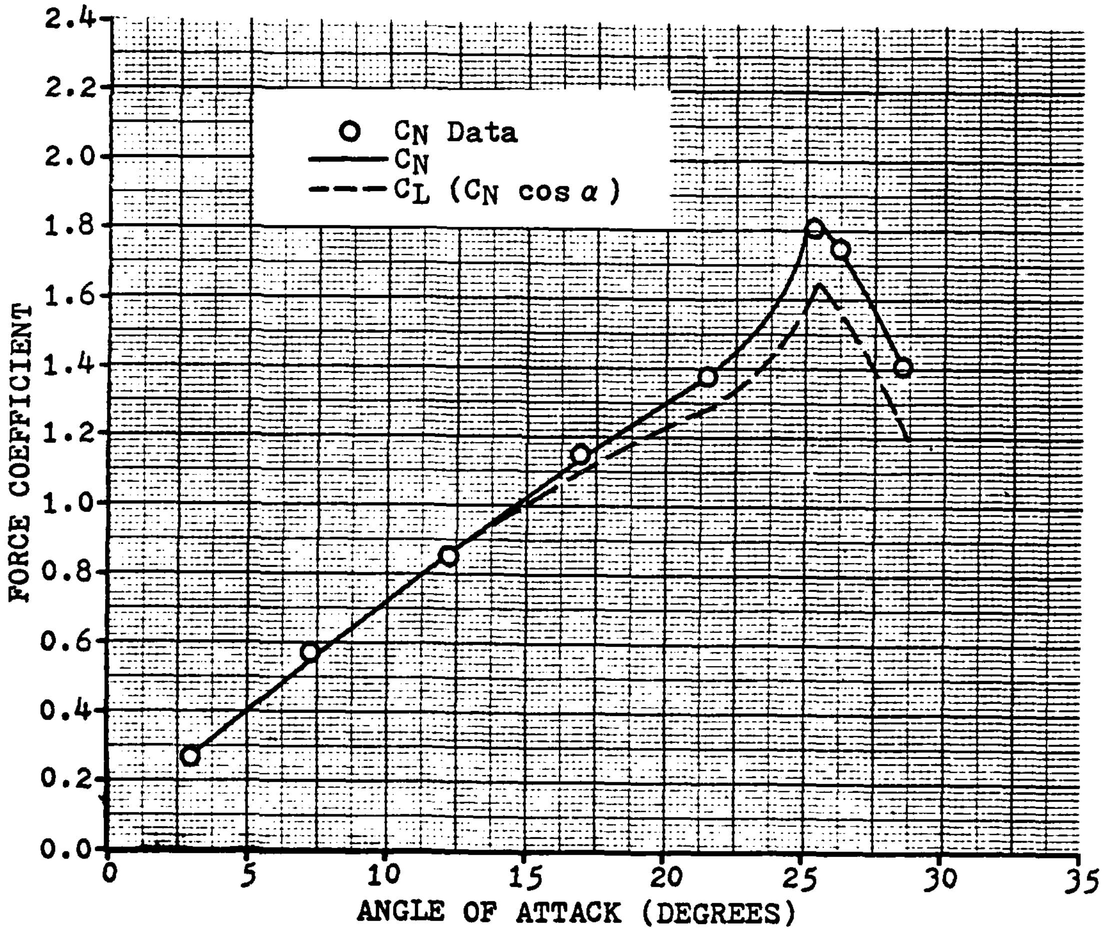


Figure 6. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=74.4 Deg/Sec and V_∞=34.4 Ft/Sec ($\dot{\alpha}_{ND}$=0.019)

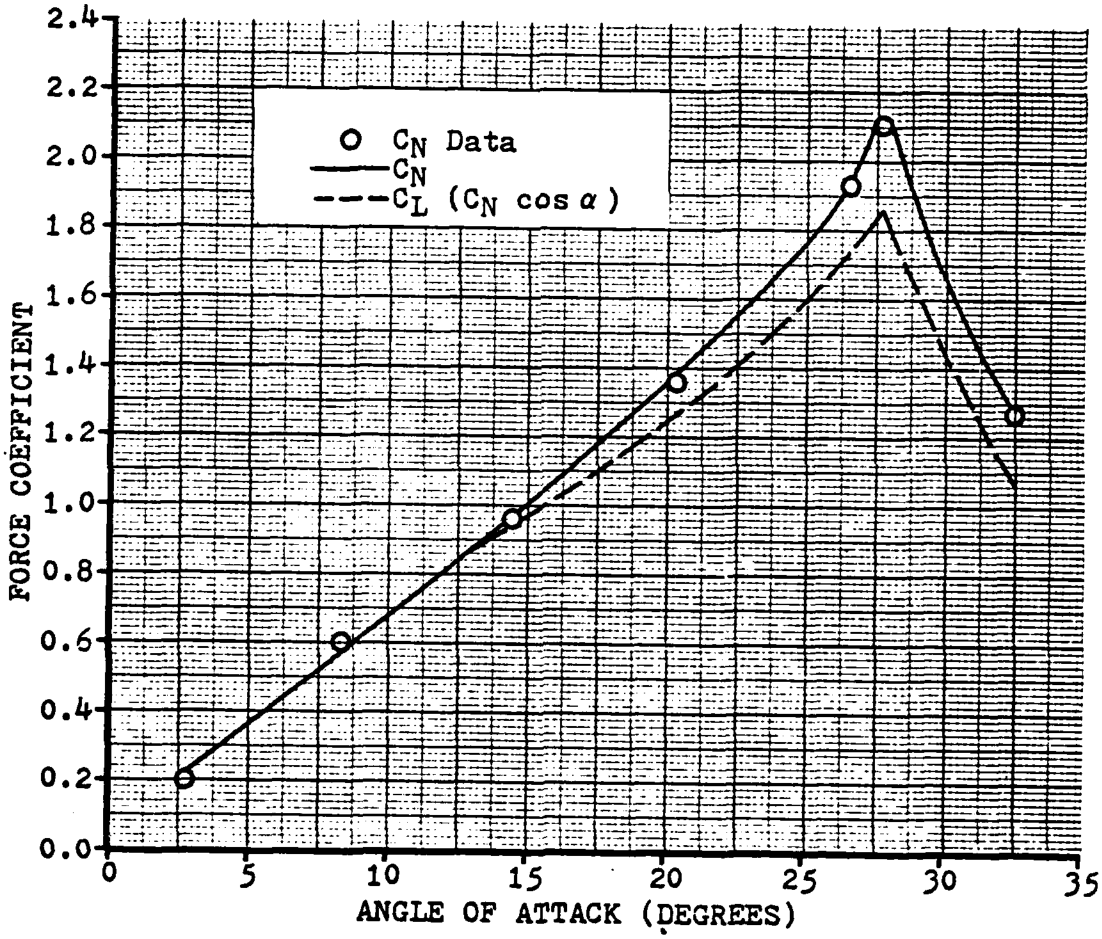


Figure 7. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=95.9 Deg/Sec and V_∞=34.4 Ft/Sec ($\dot{\alpha}_{ND}$=0.025)

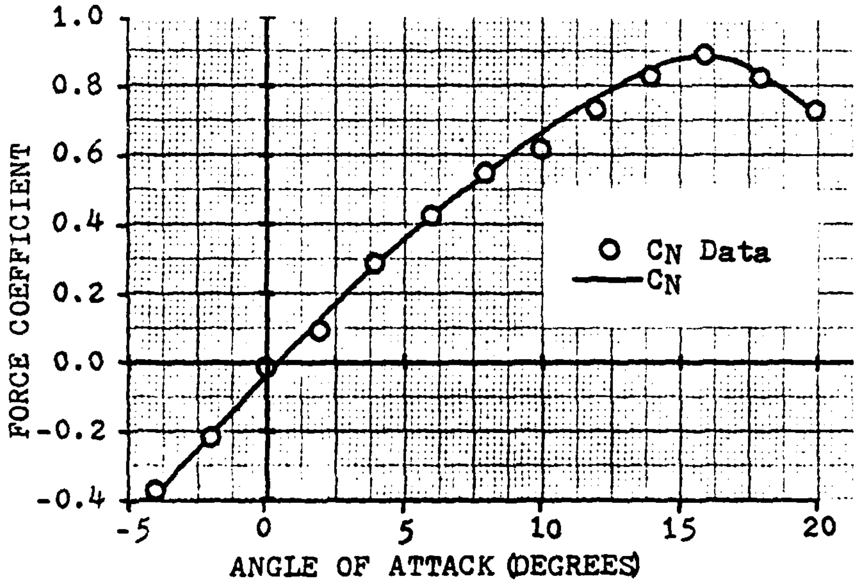


Figure 8. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=0.0 Deg/Sec and V_∞=34.4 Ft/Sec

interaction between the airfoil and the boundary layer along the walls of the test section (see Appendix B). Another source of these effects was the clearance between the airfoil and the walls of the test section necessary to allow airfoil rotation. This clearance varied from .125 inches with the tunnel shut down, to less than .05 inches for a test section velocity of 48 feet per second. To minimize the leakage through this space, felt was glued to both sides of the airfoil.

The slight differences in the character of the static lift curve at higher angles of attack might suggest a possible leading edge separation bubble (Ref 14:2). However, this difference is so slight that it could be attributed solely to data averaging effects (c.f. above). Further, smoke-on films made by Daley (Ref 10) did not appear to indicate the presence of such a bubble up to stall, at least for the dynamic cases. Since Daley filmed only the stall location for the static case, it can only be stated that a separation bubble was apparently not present in the dynamic cases.

Another factor possibly contributing to the difference in the static lift curve is tunnel blockage due to the airfoil itself as well as the wake it generates at the higher angles of attack. An explanation of these blockage effects is included in Appendix A.

Finally, it is difficult to formulate a hypothesis as to why the stall angle of attack in the present study was higher than that given for the same airfoil in Ref 6. Higher effective Reynolds number, whether due to turbulence in the test section or airfoil roughness, can probably be ruled out, though. While there was an increase in the angle of attack at which the maximum lift coefficient occurred, the maximum lift coefficient itself did not increase. The increase in stall angle of attack is probably, therefore, another manifestation of the effect depressing the lift curve slope (Ref 19) (c.f. above).

Comparing Fig. 8, the static lift curve, with Figs. 4 through 7, the dynamic lift curves for the same test section velocity, the following can be noted. The most obvious difference between the two types of lift curves is the shape of the curve at the stall point. The static lift curve exhibits a gradual buildup of lift with increasing angle of attack and levels off as the stall angle of attack is approached. When the stall condition is reached, further increases in angle of attack cause a gradual decline in lift. In contrast, the dynamic lift curve shows a rapid lift increase as angle of attack increases, with no indication of levelling off as the stall angle of attack is approached. When the stall condition is reached, the loss of lift is quite abrupt. Physically, the phenomenon responsible for this behavior is the formation and growth of a vortex near the leading edge of the airfoil, followed by its movement down the upper surface of the airfoil toward the trailing edge. Stall occurs abruptly as this vortex strips off the trailing edge into the freestream (Ref 20). A general appreciation for this effect can be gained by comparing the pressure distribution on the airfoil immediately before and after dynamic stall occurs. Fig. 9 shows the pressure distribution on the airfoil just before dynamic stall occurs, while Fig. 10 shows it a short time after dynamic stall. Although these pressure distributions have been excerpted from the case for 34.4 feet per second test section velocity and a motor voltage of 12.00 volts, they are typical of those observed in all cases in this study.

Clearly, these pressure distributions for the dynamic experiment are different from those observed for the static case at the same angle of attack. The most notable feature of the pressure distribution for the dynamic experiment is the pressure spike located on the upper surface of the airfoil, just behind the leading edge. This is seen to grow rapidly with increasing angle of attack until just before stall occurs, when it undergoes catastrophic collapse. A spike also occurs for the static case at a comparable angle of attack, and ceases to grow when stall angle of attack is reached. Compared to the dynamic case, however, the spike

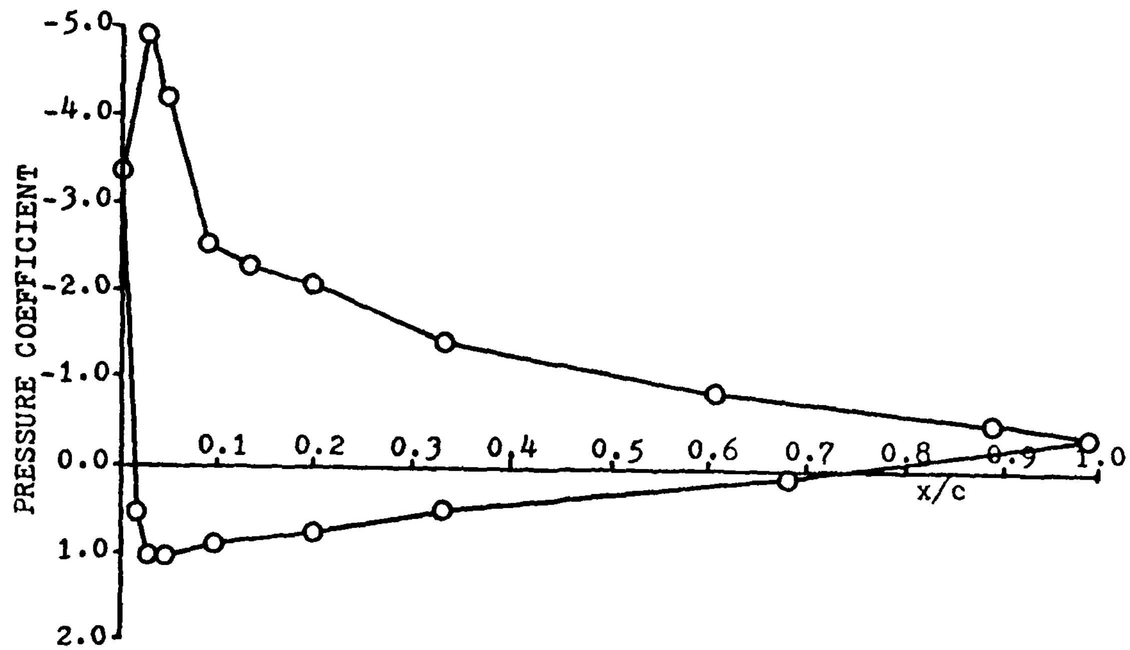


Figure 9. Pressure Distribution for Dynamic Angle-of-Attack Increase Prior to Dynamic Stall

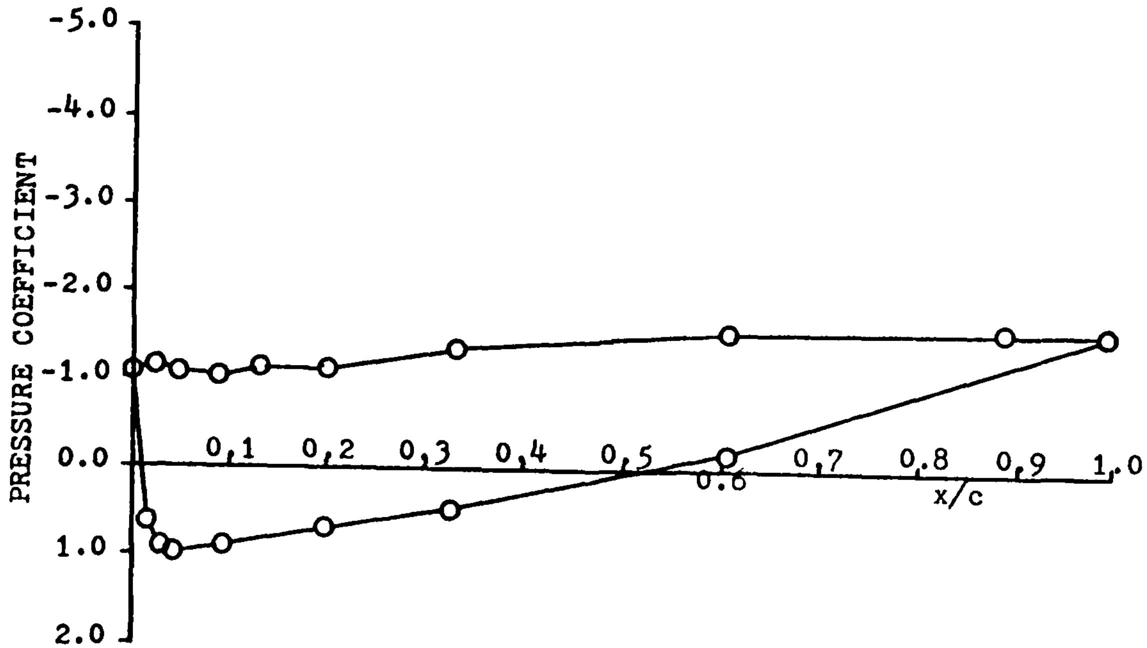


Figure 10. Pressure Distribution for Dynamic Angle-of-Attack Increase Following Dynamic Stall

for the static case shrinks gradually as angle of attack increases beyond stall.

In comparing the static and dynamic lift curves further, it becomes evident that, not only is the general character of the curve changed in the neighborhood of the stall point, but the point at which stall occurs is changed, as well. More precisely, the angle of attack at which dynamic stall occurs is significantly greater than that at which static stall occurs. This delay in stall results in a corresponding maximum lift coefficient which is also significantly greater than that encountered in the static case. The relationship between maximum unstalled angle of attack and nondimensional angular rate parameter for the case of constant-rate angle of attack increase was first determined by Deekens and Kuebler (Ref 3), and later verified by Daley (Ref 10). Table I summarizes the results of the present study in this respect, and Fig. 11 shows a plot of these results. Delta stall angle of attack is defined as the difference between the dynamic stall angle of attack and the static stall angle of attack at the same Reynolds number.

Fig. 11 shows a definite disagreement between the results of the present study and those of Daley. This is, in a large part, due to the difference in the definition of stall. The present study defines stall angle of attack as

TABLE I

Data Summary

V_∞ (Ft/Sec)	$\alpha_{stall\ st}$ (Degrees)	$\alpha_{stall\ dyn}$ (Degrees)	$\dot{\alpha}_{ND}$
26.7	16	22.6	.011
26.7	16	25.6	.018
26.7	16	27.5	.025
26.7	16	30.7	.032
30.1	16	23.5	.010
30.1	16	24.6	.016
30.1	16	26.4	.022
30.1	16	28.7	.029
34.4	16	21.2	.009
34.4	16	24.2	.014
34.4	16	25.4	.019
34.4	16	27.7	.025
39.9	17	22.8	.008
39.9	17	25.0	.013
39.9	17	26.0	.017
39.9	17	28.6	.022
47.8	17	22.6	.006
47.8	17	23.2	.010
47.8	17	25.5	.014
47.8	17	27.1	.018

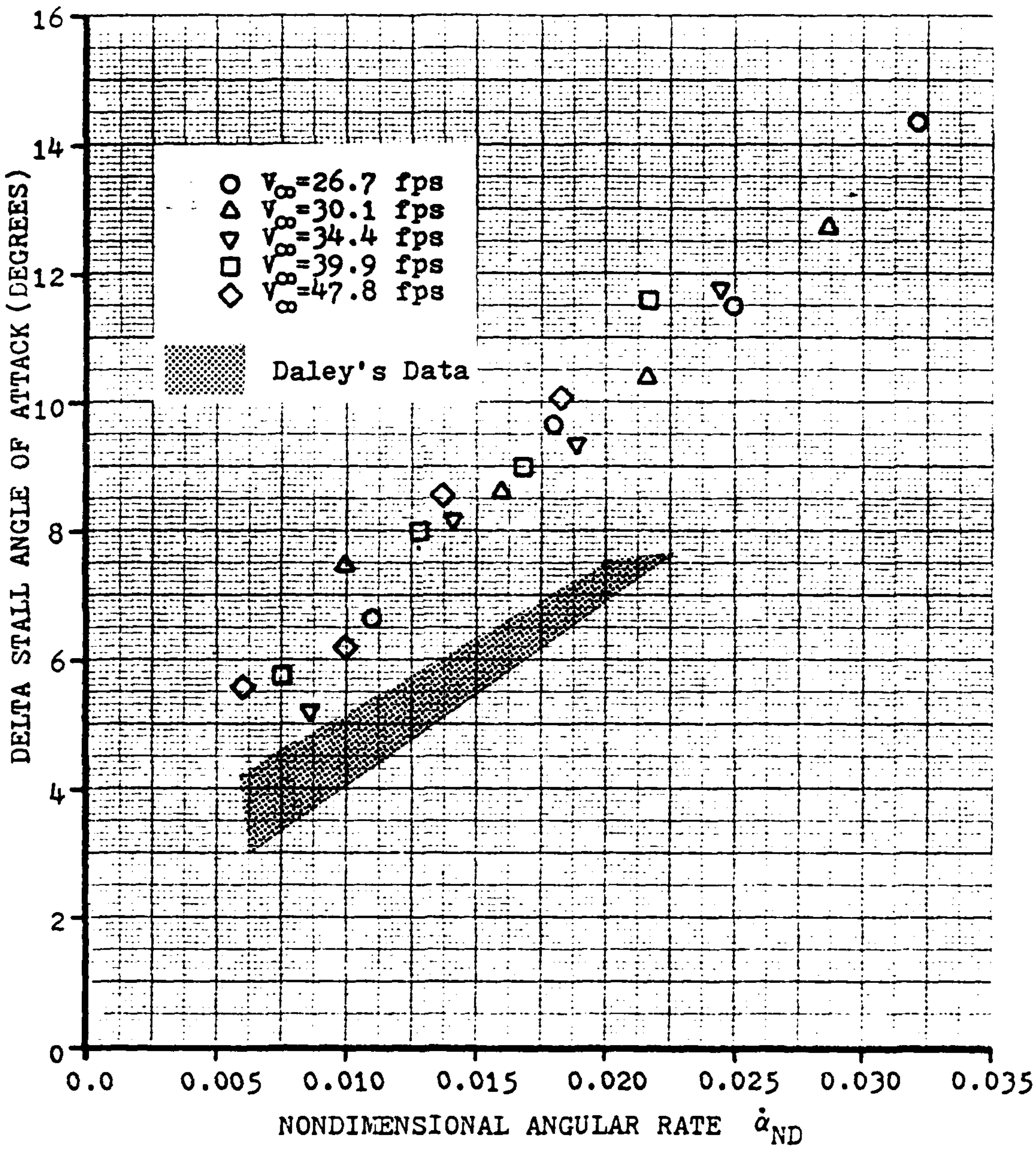


Figure 11. Data Summary Including Daley's Data

the angle of attack at which any further increase in angle of attack yields no further increase in lift coefficient. Daley, however, was unable to determine lift coefficient, and therefore defined flow separation at the airfoil quarter-chord as the stall criterion. It is perhaps significant that the two sets of results exhibit a constantly increasing divergence with increasing nondimensional angular rate. If Daley's results show the angle of attack at which quarter-chord separation occurs, and the results of the present study indicate the actual dynamic stall angle of attack, then this divergence may constitute evidence that the length of time between quarter-chord separation and actual stall is a function of nondimensional angular rate. Lawrence (Ref 15) has recently completed theoretical work aimed at predicting the dynamic stall point for the case of constant-rate angle of attack increase. These predictions are generally verified by the results of this experiment.

The final distinction between the static and dynamic stall lift curves is the difference in slope. A clearly discernible lift curve slope is not present in all twenty of the dynamic cases, but in the cases where a dynamic lift curve slope is recognizable, it seems to be depressed in comparison to the static case. It should be emphasized, however, that the difference between the dynamic and static lift curve slopes is approaching the noise level of the experiment. As a result, conclusions requiring extensive quantitative comparisons should probably not be drawn using these results. This is not to imply, however, that general qualitative trends cannot be seen here. Tupper (Ref 16) has recently completed theoretical work intended to predict the dynamic lift curve slope for the case of constant-rate angle of attack increase. The general trend in the results mentioned above seems to confirm these theoretical predictions.

An additional investigation was undertaken to experimentally determine the effect of initiating the airfoil rotation at angles of attack other than zero, as was done for the main investigation. Airfoil rotation was begun at -5, 0,5, and 10 degrees angle of attack while test section velocity and motor voltage were held constant at 29.4 feet per second and 12.00 volts, respectively. Since test section velocity and motor voltage were held constant, the associated nondimensional rotation parameter remained constant, as well. The dynamic lift curves obtained in this investigation are presented in Figs. 12 through 15.

The variation of dynamic stall angle of attack and maximum lift coefficient seems negligible with respect to the angle of attack at which airfoil rotation was begun. This result is in good agreement with the results of Law-

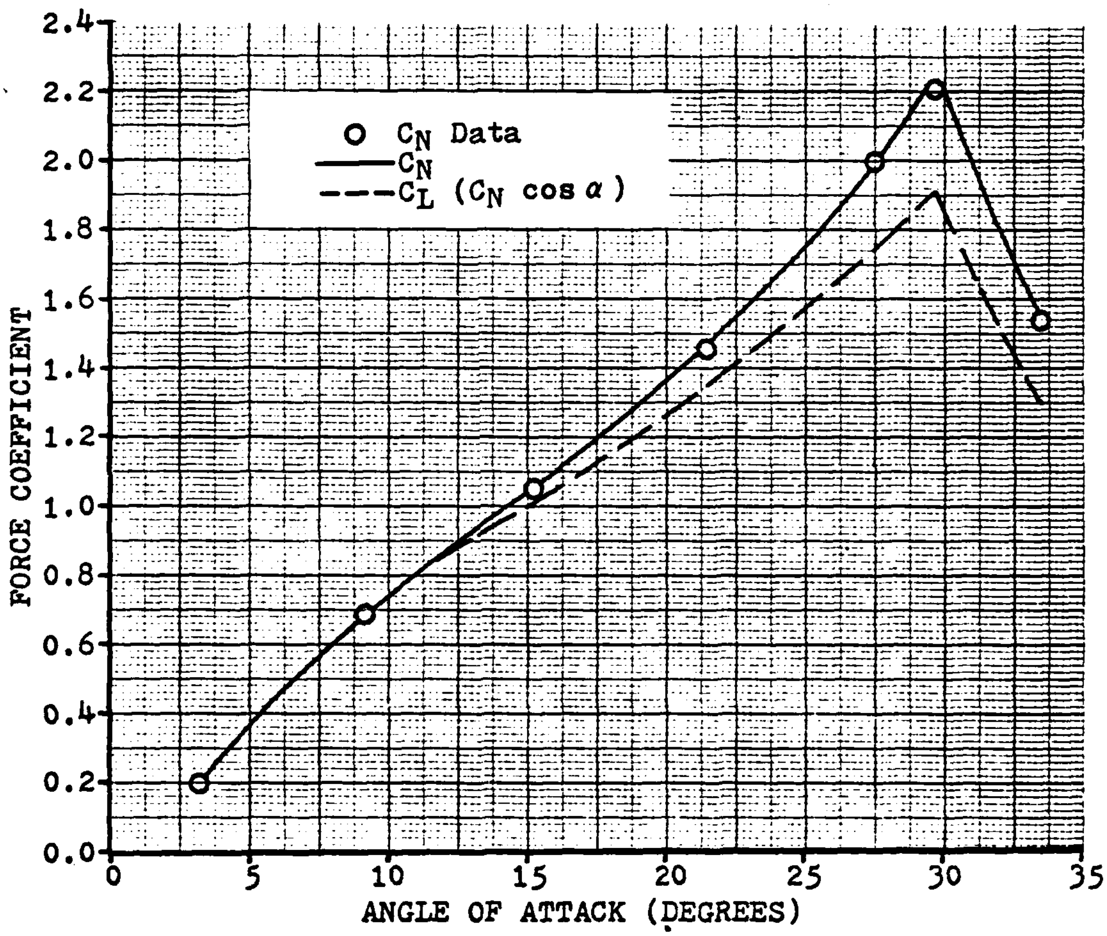


Figure 12. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=94.6 Deg/Sec and V_∞=29.4 Ft/Sec ($\dot{\alpha}_{ND}$=0.029)
Airfoil Rotation Initiated at α=-5 Degrees

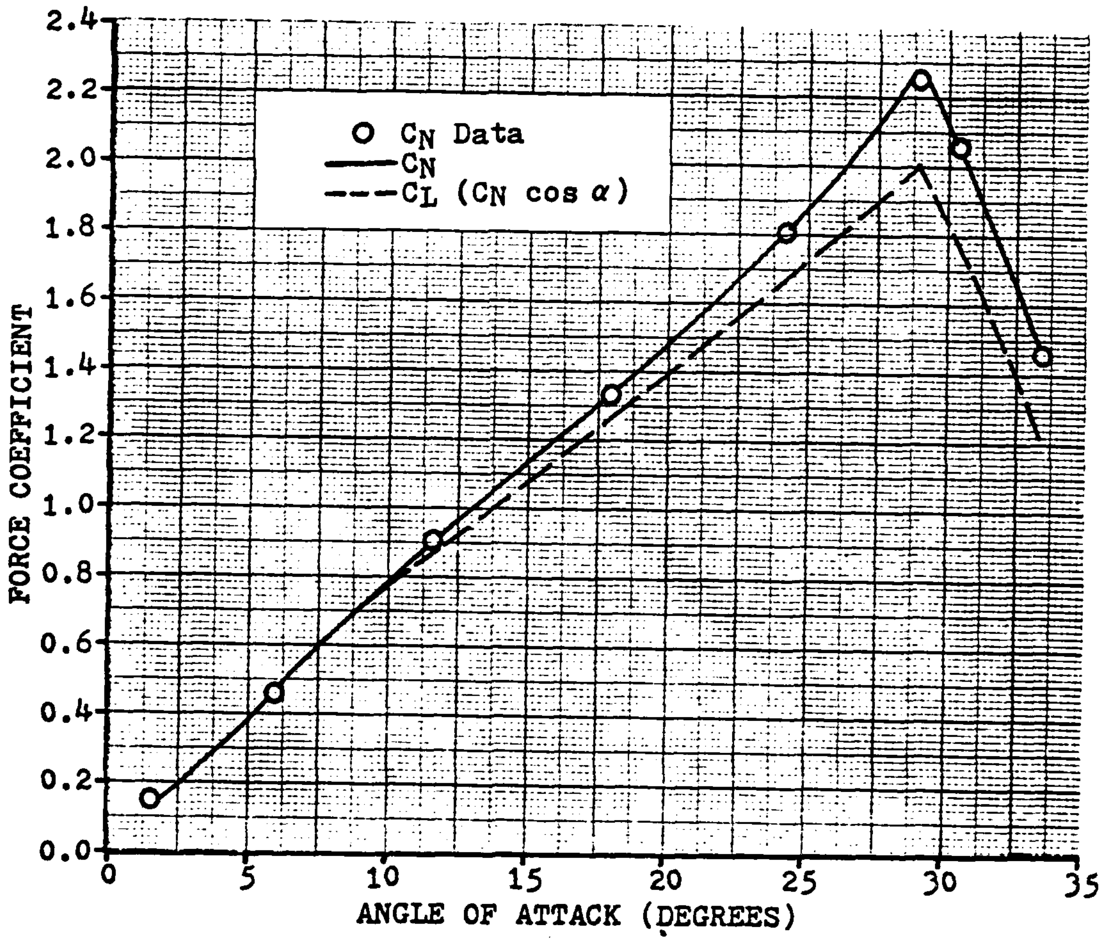


Figure 13. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=96.3 Deg/Sec and V_∞=29.4 Ft/Sec ($\dot{\alpha}_{ND}$=0.029)

Airfoil Rotation Initiated at α=0 Degrees

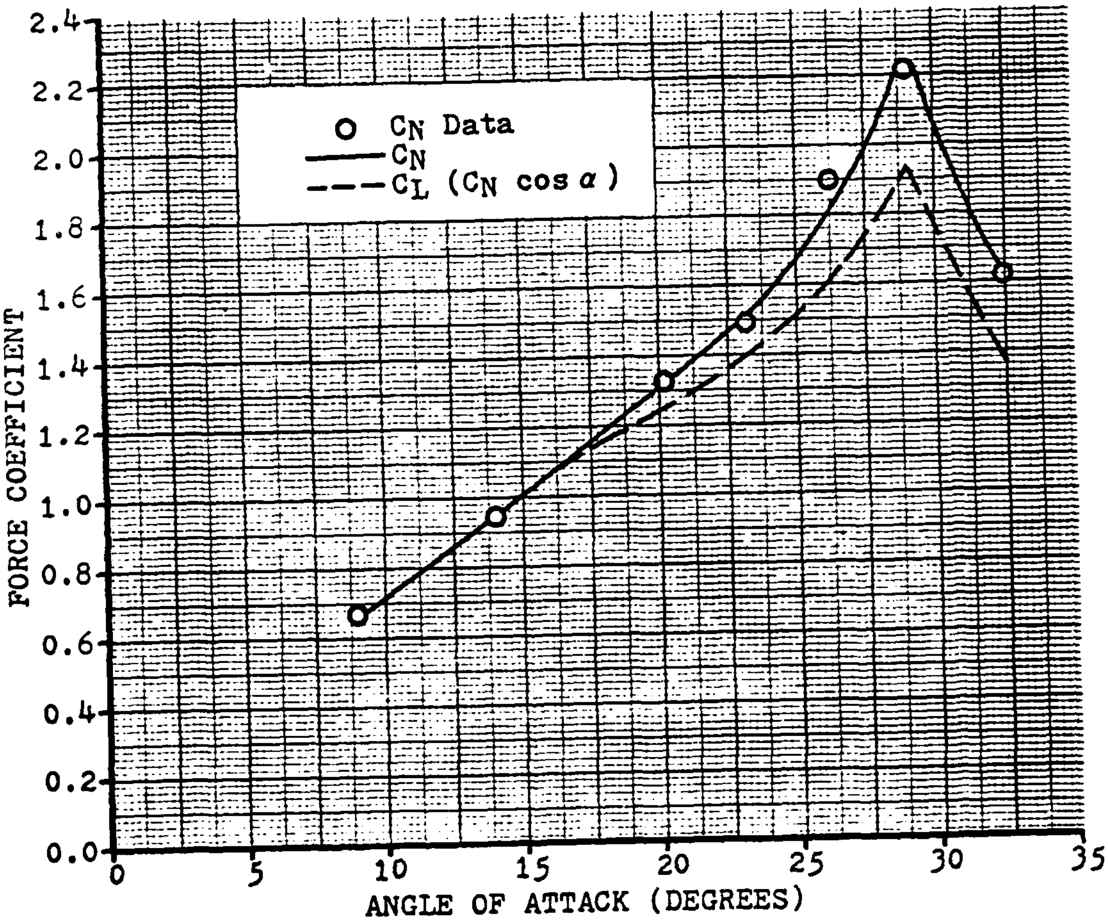


Figure 14. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=96.1 Deg/Sec and V_∞=29.4 Ft/Sec ($\dot{\alpha}_{ND}$=0.029)
Airfoil Rotation Initiated at α=5 Degrees

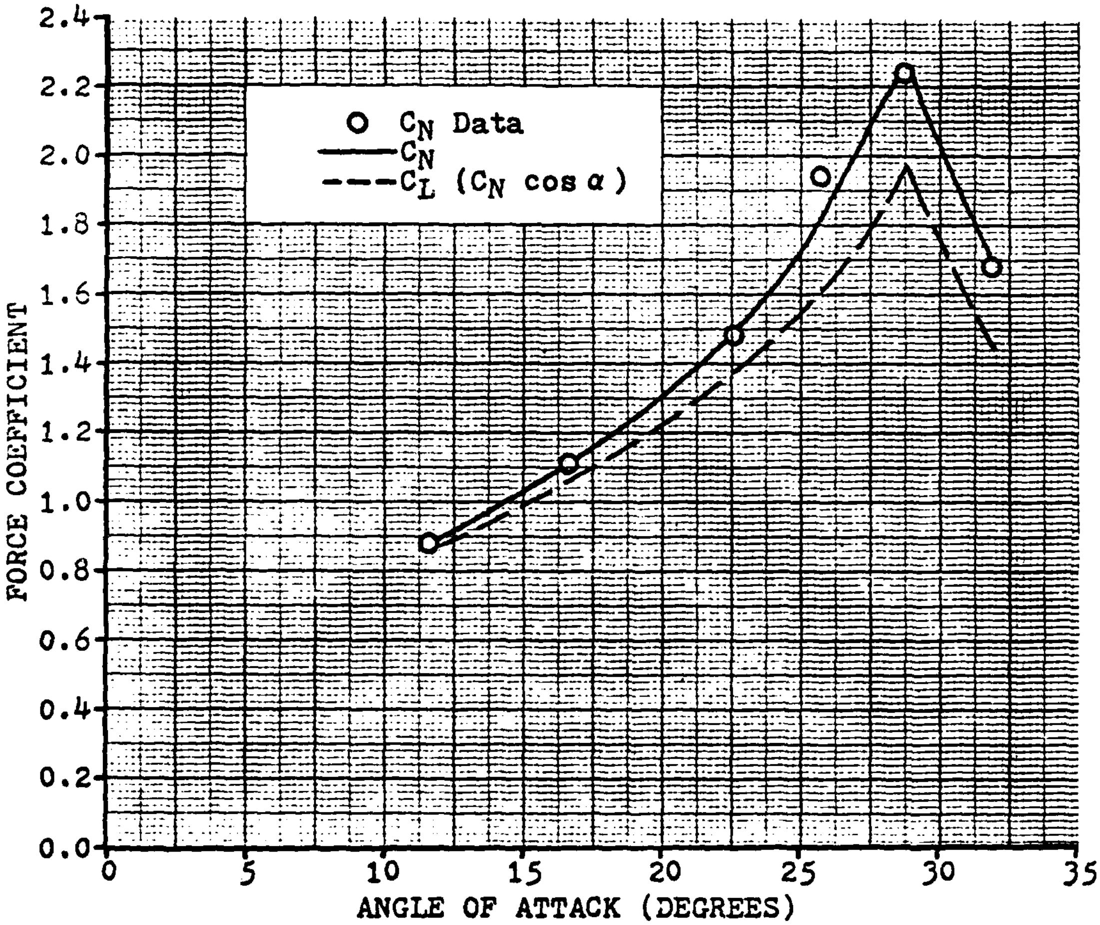


Figure 15. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=97.1 Deg/Sec and V_∞=29.4 Ft/Sec ($\dot{\alpha}_{ND}$=0.029)

Airfoil Rotation Initiated at α=10 Degrees

rence, which predict that the stall point is a function only of nondimensional rotation parameter. No clear trend is readily perceptible with respect to dynamic lift curve slope, however.

VI. Conclusions and Recommendations

Conclusions

The present investigation of dynamic stall has met all of the initial objectives. The following conclusions can be drawn from the results of this study.

The microcomputer-based data acquisition system constitutes an accurate and reliable system for gathering time, angle of attack, and pressure information associated with the physical phenomenon of dynamic stall. In addition, the software package developed to automate the reduction of this information is both efficient and dependable in accomplishing its purpose.

Based on the data obtained for the NACA 0015 airfoil, the following conclusions can be made concerning dynamic stall for nondimensional angular rate parameter between .006 and .032. First, there is a definite relationship between the delay of stall and the nondimensional angular rate parameter, $\dot{\alpha}_{ND}$. For $\dot{\alpha}_{ND} = .032$, stall was delayed by as much as 14 degrees beyond the static stall angle of attack at the same Reynolds number. The corresponding aerodynamic loads for this case reached $C_p = -5.1$ and $C_l = 1.94$. Second, neither the delay of stall nor the corresponding aerodynamic loads appear to be influenced by the angle at which airfoil rotation is initiated.

Recommendations

The results of the present study show that there is considerable potential for future dynamic stall work using the system developed for this experiment. There are, however, some possible improvements, the implementation of which would add to the value of future results.

First, the stack and rake now used in the smoke tunnel for smoke trail injection could be removed. This might reduce any flow irregularities now present in the test section, possibly improving the quality of the flow therein.

Second, an airfoil drive motor capable of higher angular rate and proportionately greater torque output should be installed. This would broaden the scope of the experiment by expanding the range of the nondimensional angular rate.

Third, supplying a higher excitation voltage to the transducers would improve the signal-to-noise ratio as well as increase the percent full-scale deflection at the output of the AIM-12 board. This modification would necessitate the recalibration of all transducers at the new excitation voltage.

Fourth, the present means of calibration is accurate, but is extremely time-consuming. By constructing an airtight case to enclose the airfoil, the calibration pressure could be applied to all 16 transducers simultaneously. Coupling with the proper software package would allow automation of the calibration process, resulting in considerable time being saved for each calibration run. This, in turn, would enable more frequent calibration of the transducers, thereby increasing experimental accuracy.

Finally, expanded RAM space in the computer would allow a more sophisticated software package to be implemented. Augmented RAM would also considerably decrease the time required for data reduction by reducing the time presently required for floppy disk input/output.

Bibliography

1. Kramer, Von M., "Die Zunahme des Maximalauftriebes von Tragflugeln bei plotzlicher Anstellwinkel-vergro Berung (Boeneffekt)," Zeitschrift fur Flugtechnik und Motor-luftschiffahrt, 7, 14 April 1932, pp. 185-189.

2. Docken, R. G. Jr., Jumper, E. J., and Hitchcock, J. E. "Theoretical Gust Response Prediction of a Joukowski Airfoil," Proceedings of the Ninth Annual AIAA (Dayton-Cincinnati) Mini-Symposium, AIAA(DAY/CIN)83-1 3-2-1 - 3-2-3 (February 1983).

3. Deekens, A. C., and Kuebler, W. R. "A Smoke Tunnel Investigation of Dynamic Separation," Air Force Academy Aeronautics Digest - Fall 1978, 2-16, USAFA-TR-79-1 (February 1979).

4. Francis, M. S. Unpublished work on dynamic stall characteristics.

5. Scheubel, N. "Some Tests on the Increase of the Maximum Lift of Aerofoils Whose Angle of Incident Changes at Constant Angular Velocity, "Mitt deut Akad Luftfahrt-Forsch 1, 39-45 (1942).

6. Jacobs, Eastman N.; and Sherman, Albert: Airfoil Section Characteristics as Affected by Variations of the Reynolds Number. NACA Report 586, 1937.

7. Williams, James C. III. "Incompressible Boundary-Layer Separation," Annual Review of Fluid Mechanics, 9: 113-144 (1977).

8. Carr, Lawrence W.; McAlister, Kenneth W.; and McCroskey, William J.: "Analysis of the Development of Dynamic Stall Based on Oscillating Airfoil Experiments," NASA Technical Note D-8382, 1977.

9. McAlister, Kenneth W.; Carr, Lawrence W.; and McCroskey, William J.: "Dynamic Stall Experiments on the NACA 0012 Airfoil," NASA Technical Paper 1100, 1978.

10. Daley, Daniel C. "Experimental Investigation of Dynamic Stall," Master's Thesis, AFIT/GAE/AA/82D-6 (1983).

11. Sisson, F. E. II "Completion of the AFIT Smoke Tunnel," Master's Thesis, AFIT/GAE-12 (1957).

12. Baldner, J. L. "Completion of the Development of the AFIT Smoke Tunnel," Master's Thesis, AFIT/GAE-2 (1959).

13. Endevco Corporation. Series 8507 Miniature Piezoresistive Pressure Transducers. San Juan Capistrano, California: Endevco Corp.

14. Mueller, T. J. and Batill, S. M. "Experimental Studies of the Laminar Separation Bubble on a Two-Dimensional Airfoil at Low Reynolds Numbers," Paper 80-1440, American Institute of Aeronautics, and Astronautics, 13th Fluid and Plasma Dynamics Conference, Snowmass, Colorado, 14-16 June 1980.

15. Lawrence, John S. "Investigation of Effects Contributing to Dynamic Stall Using a Momentum-Integral Method," Master's Thesis, AFIT/GAE/AA/83D-12 (1983).

16. Tupper, Kenneth W. "The Effect of Trailing Vortices on the Production of Lift on an Airfoil Undergoing a Constant Rate of Change of Angle of Attack," Master's Thesis, AFIT/GAE/AA/83D-26 (1983).

17. Pankhurst, R. C. and Holder, D. W. Wind Tunnel Technique. London: Sir Isaac Pittman and Sons, Ltd., 1954.

18. Schlichting, H. Boundary-Layer Theory, 7th ed. New York: McGraw-Hill, 1979.

19. Conversation with E. J. Jumper.

20. Chow, Chuen-Yen and Chiu, Chyn-Shan "Unsteady Aerodynamic Loading on an Airfoil Due to Vortices Released Intermittently from its Upper Surface," Proceedings of the Workshop on Unsteady Flow, United States Air Force Academy, Colorado, 10-11 August 1983.

APPENDICES

Appendix A

Tunnel Interference Effects

Even if the usual Reynolds number comparison criterion is met, the flow around an airfoil in a wind tunnel will generally differ from that which would be observed in free air, where the fluid is of unlimited extent. The factors which produce these discrepancies are collectively referred to as tunnel interference effects. Corrections for these interference effects can be applied to the parameters measured in the tunnel to determine the corresponding values that would be obtained for measurement of the same parameters in free air.

In view of the discussion presented by Pankhurst and Holder (Ref 17:327-388), three interference effects were considered significant for the case of an uncambered airfoil spanning the test section in incompressible, two-dimensional flow enclosed by a test section having rigid walls. These three effects are two-dimensional solid blockage, two-dimensional wake blockage, and two-dimensional streamline curvature.

Two-dimensional solid blockage causes an increase in axial velocity past the airfoil, owing to its partially blocking the flow in the presence of the wall constraint. This velocity increment is a function of the chordlength, thickness, and thickness distribution of the airfoil and is independent of camber or angle of attack while flow remains attached over the airfoil. The velocity increment due to solid blockage, however, is much less than that which would be caused by direct area reduction. The reason for this is that the streamlines most affected are those at the tunnel boundary, which are the greatest distance away from the airfoil.

Two-dimensional wake blockage also causes an increased axial velocity past the airfoil, but does so for a different reason. The wake behind the airfoil has a lower mean velocity than the freestream because of viscous interaction with the body. Since the wake has a lower mean velocity than the freestream, the average velocity of the freestream must increase in order to satisfy continuity in the test section.

Two-dimensional streamline curvature corrections must be applied because the presence of the test section floor and ceiling prevent the streamline curvature that would normally occur in the vicinity of an airfoil in free air. Relative to the flow straightened by the test section floor and ceiling, the airfoil appears to be cambered, and therefore delivers more lift than it would at the same angle of attack in free air.

These three blockage effects are quantified mathematically in Pankhurst and Holder as follows. The solid blockage coefficient, ε_{SB}, is defined:

$$\varepsilon_{SB} = \frac{\pi^2 \lambda}{12} \left(\frac{t}{h}\right)^2 \tag{10}$$

where λ is taken from Fig. 239 in Pankhurst and Holder, and t and h are the maximum thickness of the airfoil and the height of the tunnel (in the same units), respectively. This gives the result:

$$\varepsilon_{SB} = 0.006$$

The wake blockage coefficient, ε_{WB}, is defined:

$$\varepsilon_{WB} = \frac{c}{4h} C_D \tag{11}$$

where c is the model chordlength, h is the tunnel height, and C_D is the drag coefficient of the airfoil. Using a conservative estimate for C_D, the result is obtained:

$$\varepsilon_{WB} = 0.002$$

Finally, the streamline curvature correction factor which gives the change in effective angle of attack is:

$$\Delta\alpha_{SC} = \frac{\pi}{96} \left(\frac{c}{h}\right)^2 C_1 \tag{12}$$

where c is the airfoil chordlength, h is the test section height, and C_1 is the lift coefficient at which the correction is to be applied. With the maximum dynamic stall lift coefficient observed being $C_1 = 1.94$, the maximum $\Delta\alpha_{SC}$ is .35 degrees.

Using the correction coefficients obtained above, the corrected test section velocity and angle of attack can be found:

$$V_{\infty CORR} = V_{\infty}(1 + \varepsilon_{SB} + \varepsilon_{WB}) \tag{13}$$

$$\alpha_{CORR} = \alpha + \Delta\alpha_{SC} \tag{14}$$

These corrections assume steady-state test conditions and were arrived at using conservative quantities. Tunnel wall interference effects for the transient case are virtually impossible to compute accurately, but can logically be assumed smaller than those for the static case. Therefore, the computed correction factors for velocity and angle of attack define conservative upper limits on the error which would be introduced by omitting these corrections. Due to the virtual impossibility of computing tunnel wall interference corrections for the transient case, and the insignificance of the error introduced by not doing so, corrections for tunnel wall interference effects were not applied to the dynamic stall data gathered in this study.

Appendix B

Test Section Flow Characterization

The tunnel used in this study was originally intended for smoke-trace flow visualization, and the test section is only 2.75 inches wide. As such, there was initially some concern that the boundary layers on the side walls were so thick as to render the pressure data of little value. A preliminary investigation was therefore undertaken to determine the velocity profile of the transverse boundary layers in the test section.

Apparatus

Instrumentation used to measure the velocities consisted of the same Pitot-static probe and Meriam A-937 water micromanometer normally used to measure test section velocity in the smoke tunnel. The probe was connected to the manometer with two eight-foot lengths of Tygon tubing having an outside diameter of .125 inch.

To facilitate placement of the probe at the desired locations in the test section, a bracket was constructed from a wooden dowel .75 inch in diameter. The dowel was cut to a length of 2.625 inches so that when both ends were capped with a layer of .125 inch thick double-sided adhesive foam rubber, the bracket would be clamped tightly between the walls of the closed test section.

Three holes, each .187 inch in diameter and parallel to each other, were bored through the diameter of the dowel to accommodate the stem of the Pitot-static probe. One hole was centered at the midpoint of the dowel, and the remaining two were centered at locations .5 and 1.0 inches from the midpoint. Each hole was equipped with a set screw to hold the stem of the probe securely in the bracket. Thus, each boundary layer profile was defined by four points, including the no-slip condition at the wall station.

Procedure

To characterize the development of the transverse boundary layer in the test section, two bracket locations were consistently used in this investigation. Location A placed the tip of the probe 10.0 inches above the floor of the test section and 4.0 inches downstream of where the test section begins. Location B placed the tip of the probe the same distance above the floor as location A, but 40.0 inches downstream of where the test section begins. At each bracket location, the flow velocity was measured with the stem of the probe positioned successively in each of the three holes in the bracket.

It was, of course, necessary to open the test section each time the bracket location or probe stem position was to be changed. The tunnel motor controls, however, were set at the beginning of the run and were not adjusted until the velocities at both bracket locations and all three stem positions were measured. Due to the unusually long tubes connecting the probe to the manometer, care was taken to allow the manometer ample time to reach equilibrium before recording the reading. It was also important to insure that the head of the probe was at zero angle of attack in order to obtain accurate velocity measurements. This was accomplished by positioning the probe such that its stem was parallel to a plumb line suspended from the front wall of the test section.

Results

The results of this investigation are presented in Figs. 16 through 19. There are two sets of velocity profiles, one for each test section velocity. Note that the no-slip condition has been assumed for all wall points. From Schlichting (Ref 18:42), the approximate thickness of a turbulent boundary layer developing on a flat plate in the absence of a pressure gradient is:

$$\frac{\delta}{l} = 0.37 \left(\frac{V_\infty l}{\nu}\right)^{-1/5} \tag{15}$$

If V_∞ is taken as the velocity at the centerline of the test section and the distance down the test section is used for l, the predicted thickness of the boundary layer along the wall of the test section at each of the bracket locations is as depicted in each of the Figs. As shown, the predicted and measured boundary layer thicknesses agree quite well.

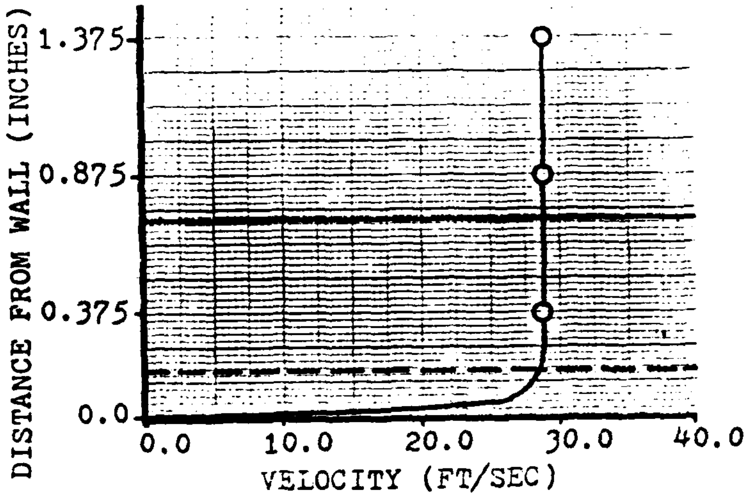


Figure 16. Boundary Layer Profile at Location A

O Boundary Layer Data
O——O Boundary Layer Profile
– – – Predicted Boundary Layer Thickness
—— Position of Transducer Closest to Wall

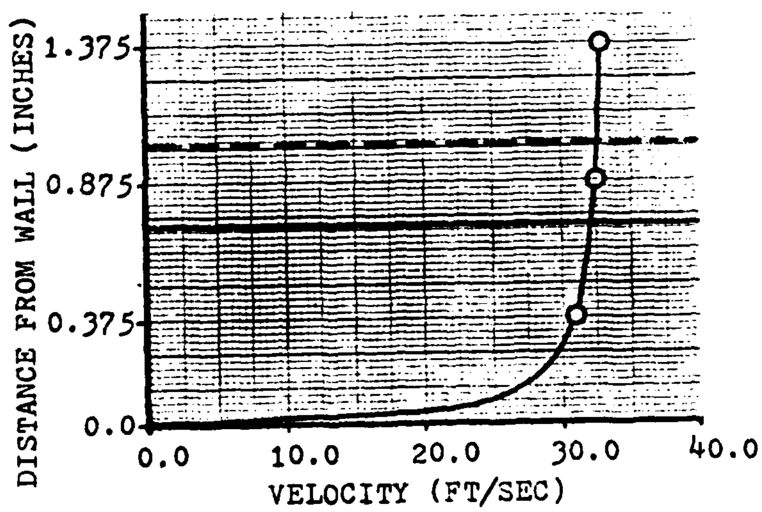


Figure 17. Boundary Layer Profile at Location B

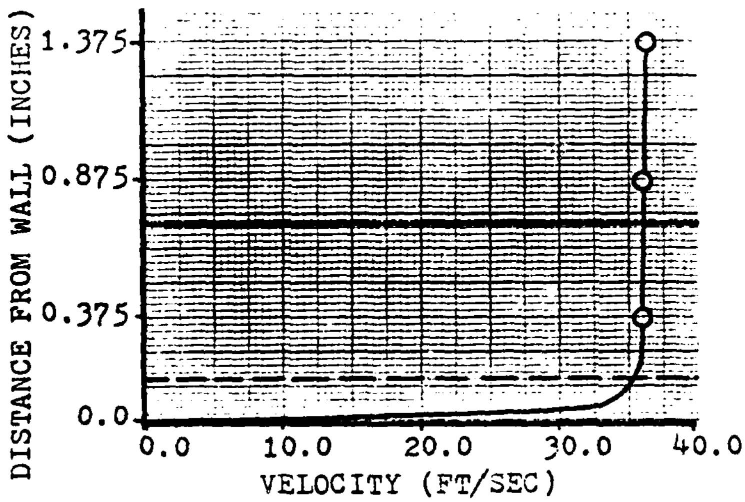


Figure 18. Boundary Layer Profile at Location A

O Boundary Layer Data
O——O Boundary Layer Profile
— — — Predicted Boundary Layer Thickness
——— Position of Transducer Closest to Wall

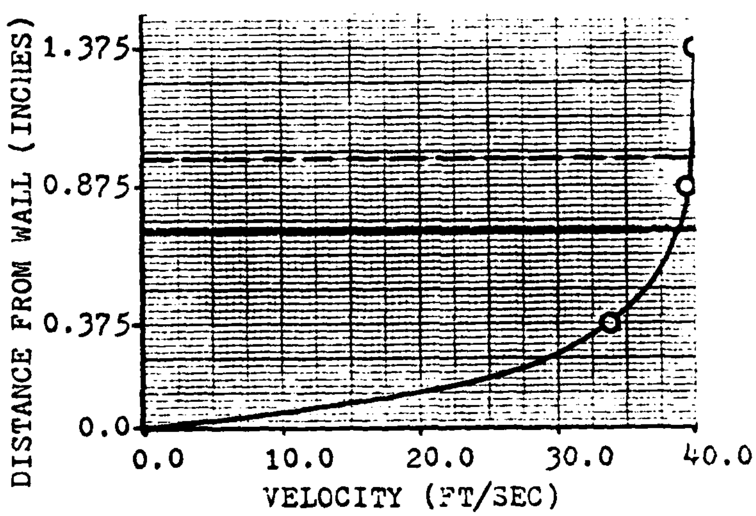


Figure 19. Boundary Layer Profile at Location B

Conclusions

Based on the results of this preliminary investigation, the effect of the wall boundary layers is not as severe as had been previously supposed. Even at the lower test section velocity and at the location downstream of the airfoil (conditions which would give the thickest boundary layer), the transducer closest to the wall was still in flow having 95 percent of the velocity at the centerline of the test section.

Appendix C

Transducer Calibration

Introduction

Even though each transducer came from Endevco complete with its own factory calibration, all 16 of the transducers were recalibrated following their installation in the airfoil. As Table II shows, some transducer sensitivities changed drastically between the time the transducer was factory calibrated and the time it was first calibrated after mounting in the airfoil. Calibration of the transducers was subsequently repeated every three weeks until the experimental portion of this study was completed. After the initial calibration mentioned above, later calibrations showed no transducer undergoing a sensitivity change of greater than approximately two percent.

Apparatus

The calibration was somewhat complicated by the fact that the transducers were embedded in the airfoil, and the airfoil could not be removed from the test section for calibration. Calibration pressure was individually applied to each of the transducers using a small rubber cup sealed to the airfoil with silicone vacuum grease. However, one of the central problems in calibration was to develop a method for holding the rubber cup in place on the airfoil. This

TABLE II

Transducer Sensitivities

Transducer	Endevco Calibration $\frac{mV}{psi}$	Initial Calibration $\frac{mV}{psi}$
1	173.5	196.2
2	154.6	168.4
3	175.6	173.5
4	189.5	226.4
5	187.8	203.8
6	178.7	200.1
7	195.7	229.9
8	174.3	208.8
9	160.2	170.9
10	112.0	113.9
11	119.6	119.3
12	112.7	112.3
13	131.0	139.2
14	142.5	167.6
15	195.6	217.4
16	185.7	217 2

problem was solved using the device pictured in Fig. 20. The two suction cups were of the standard variety used to attach cargo racks to the roofs of cars, and had a nut molded into them to accept the bolts shown. The tube was sealed into the rubber cup using silicone rubber adhesive sealant. With the airfoil in place in the test section, the suction cups were stuck to the rear wall of the test section, thus holding the rubber pressure cup securely in place on the airfoil.

The transducers were calibrated in the same pressure range in which they operated, the region of low magnitude, negative pressures. The Meriam A-937 water micromanometer provided a means of simultaneously generating and measuring a low magnitude, negative pressure for the calibration pressure input to the transducer. This was accomplished by applying a positive pressure to the left leg of the micromanometer such that the water column in the right leg rose approximately four inches. After the tube leading to the pressure cup was attached to the right leg of the manometer, the pressure to the left leg of the manometer was released, resulting in a negative pressure of approximately four inches of water being applied to the transducer. To control the magnitude of the pressure applied to the transducer, ambient air was bled into the system through a tee fitting midway in the line between the micromanometer and

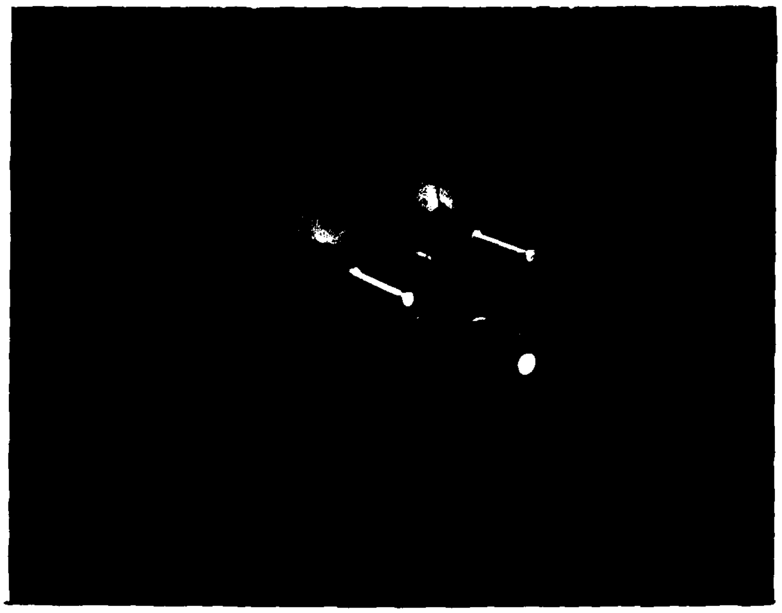


Figure 20. Calibration Device in Place on the Airfoil

the pressure cup, thereby lowering the level of the water column in the micromanometer.

Calibration Procedure

The computer, both power supplies, and all three voltmeters were powered up and allowed to warm up for a minimum of four hours before the actual calibration was begun. When calibration was ready to begin, a light coating of silicone vacuum grease was spread on the lip of the pressure cup to improve the seal between it and the airfoil surface, and the pressure cup was applied to the transducer to be calibrated. Next, the calibration program TEST, a copy of which is included in Appendix D, was executed. With TEST running, the voltage output by the transducer was continuously updated and displayed at the computer terminal in digital counts.

Pressure was then applied to the transducer as previously described, and both the micromanometer reading and the transducer digital output were then recorded. With the calibration system leakproofed, any fault in the transducer-airfoil seal would be shown by the failure of the meniscus to reach an equilibrium position. Sufficient air was then bled into the calibration system to lower the water column approximately one inch, and the previous procedure was then repeated.

A total of five successively lower water column heights were used in calibrating each transducer. These heights were between four and zero inches, spaced at approximately one-inch intervals. The five calibration points thus obtained were then plotted on a graph having the pressure input to the transducer in inches of water on its horizontal axis and the digital pressure reading from the transducer in counts on its vertical axis. In all cases, this calibration curve was found to be linear, as advertised by Endevco. The slope of this curve, then, was the sensitivity of the transducer in digital counts per inch of water. The sensitivity in millivolts per psi (pounds per square inch) was calculated using the following conversion.

$$\frac{\text{mV}}{\text{psi}} = \frac{\text{digital counts}}{\text{inches of water}} \; \frac{50 \text{ mV}}{2048 \text{ digital counts}} \; \frac{27.68 \text{ inches of water}}{\text{psi}} \qquad (16)$$

These transducer sensitivities in millivolts per psi were then typed into the appropriate arrays in programs RED and DEN.

Appendix D

Software Package

```
          PROGRAM BIG
C ----    To gather and store data for further processing
C ----    Link:  UNO,STCLK,GETTIM,ADIO,FORLIB/S,UNO/N/E
C
          IMPLICIT INTEGER (A-Z)
          REAL AVSTAT(16),STATIC(16),BAROM,TEMP,MANOM1,MANOM2,TUNVEL
          REAL MOTVOL,P90,P0,RHO,DTIM,DPOSV,DPOSD,ROTRAT,VPD
          REAL PORTU(10),PORTL(10),SENS(16),CPU(10),CPL(10)
          REAL IDATAT(16),NORMCO,PRESS,STICKY
          REAL CP(16),AREAUT,AREALT,LNGTHU,LNGTHL,AREAU,AREAL,INTU,INTL
          INTEGER IDATA(5050),HOUR,CHECK,CHAN,DAY,MONTH,YEAR,XX
          INTEGER VALUE,CHEK,NS,N,A,DI,K,J,B,AA,L,C,KOUNT,S,T,U,DD,EE,ZZ
          INTEGER DIFANG,INK,RUNS,XXX,YYY,RRR,ZERANG,SNAP,SELECT
          INTEGER CHECK,CHEK,CHAN,VALUE,KOUNT,Z,W,S,CCC
          INTEGER II,JJ,KK,WW,DD,X,V,Y,TT,ZZZ
C
C ---- Load transducer sensitivities (millivolts/psi)
        DATA SENS/206.5,174.2,174.9,237.5,207.0,205.2,243.1,
     +221.2,172.8,113.1,118.3,111.5,138.7,177.5,221.9,225.2/
C
C ---- Load transducer locations on upper surface (percent chord)
        DATA PORTU/0.0,0.0242,0.0484,0.0969,0.129,0.194,0.323,0.605,
     +0.888,1.000/
C
C ---- Load transducer locations on lower surface (percent chord)
        DATA PORTL/0.0,0.0161,0.0319,0.0484,0.0969,0.194,0.323,
     +0.686,1.000/
C
C ----    Initialize count of passes to zero.
C
 10       KOUNT=0
C
C ----    Input date, time, barometer, and room temperature
C ----    for experimental records.
C
          WRITE (1,15)
 15       FORMAT (' ENTER DAY, MONTH, YEAR SEPERATED BY COMMAS',/)
          READ (1,20)DAY,MONTH,YEAR
 20       FORMAT (I3,I3,I3)
          WRITE (1,25)
 25       FORMAT (' ENTER TIME (MILITARY: XXXX HOURS)',/)
          READ (1,30)HOUR
 30       FORMAT (I5)
          WRITE (1,35)
 35       FORMAT (' ENTER BAROMETER (INCHES OF MERCURY)',/)
          READ (1,40)BAROM
 40       FORMAT (F7.2)
          WRITE (1,45)
 45       FORMAT (' ENTER ROOM TEMPERATURE (DEGREES FAHRENHEIT)',/)
          READ (1,50)TEMP
 50       FORMAT (F6.1)
C
```

Following

PAGE'S

```
C ----  Echo date, time, barometer, and room temperature for
C ----  verification.  Offer option to correct faulty input.
C
        WRITE (1,55)DAY,MONTH,YEAR
 55     FORMAT (' DAY:',I3,'    MONTH:',I3,'    YEAR:',I3)
        WRITE (1,60)HOUR
 60     FORMAT (' TIME:',I5)
        WRITE (1,65)BAROM
 65     FORMAT (' BAROMETER:',F7.2,' INCHES OF MERCURY')
        WRITE (1,70)TEMP
 70     FORMAT (' ROOM TEMPERATURE:',F6.1,' DEGREES FAHRENHEIT')
        WRITE (1,75)
 75     FORMAT (///,'                    ARE THE INPUTS, ECHOED ABOVE,')
        WRITE (1,80)
 80     FORMAT ('                    CORRECT?  IF SO, ENTER A 1',/)
        READ (1,85)CHECK
 85     FORMAT (I1)
        IF (CHECK.NE.1) GO TO 10
C
C ----  Following part of program calculates an average zero-input
C ----  reading for each transducer.  Average is obtained from 10
C ----  readings of each transducer.
C
        WRITE (1,90)
 90     FORMAT (///,' THIS PART OF THE PROGRAM OBTAINS AVERAGE')
        WRITE (1,95)
 95     FORMAT (' TRANSDUCER ZERO-INPUT READINGS.  WHEN TEST-')
        WRITE (1,100)
 100    FORMAT (' SECTION VELOCITY IS ZERO, HIT RETURN KEY')
        WRITE (1,102)
 102    FORMAT (' IN RESPONSE TO "PAUSE"',///)
        PAUSE
C
C       Initialize all array elements to zero.
C
 110    CONTINUE
        DO 120 Z=1,16
        AVSTAT(Z)=0.0
 120    CONTINUE
C
C ----  Take 10 readings from each transducer, average them as shown
C ----  below, then write these averages to terminal. Also offer the
C ----  option to retake the average zero-input readings.
C
        DO 150 S=1,10
        DO 160 T=1,16
        CHAN=T-1
        CALL AD(VALUE,CHAN,80)
        AVSTAT(T)=AVSTAT(T)+(VALUE/10.0)
 160    CONTINUE
 150    CONTINUE
C
```

```
C
      WRITE (1,155)
 155  FORMAT (' AVERAGE ZERO-INPUT READINGS FOLLOW',/)
C
      DO 180 W=1,16
      WRITE (1,165)W,AVSTAT(W)
 165  FORMAT (' TRANSDUCER',I3,' AVERAGE STATIC READING:',F6.0)
 180  CONTINUE
      WRITE (1,177)
 177  FORMAT (///,' TO PROCEED WITH THE PROGRAM, ENTER A 1',/)
      READ (1,178)XX
 178  FORMAT (I2)
      IF (XX.NE.1) GO TO 110
C
C
C ----  Enter manometer reading, motor voltage, and 90 and 0
C ----  degree angle of attack voltages for experimental records.
C ----  Test-section velocity is also computed as shown below.
C
C
      WRITE (1,185)
 185  FORMAT (//////,'****************************NOW TURN ON THE
     + TUNNEL****************************',//////)
 187  WRITE (1,190)
 190  FORMAT (' ENTER ROOM PRESS. MINUS TUNNEL STAT. PRESS.
     + (INCHES OF WATER)',/)
      READ (1,195)MANOM1
 195  FORMAT (F8.4)
      WRITE (1,200)
 200  FORMAT (' ENTER TUNNEL TOTAL PRESS. MINUS TUNNEL STATIC PRESS.
     + (INCHES OF WATER)',/)
      READ (1,195)MANOM2
 205  FORMAT (F8.4)
      WRITE (1,210)
 210  FORMAT (' ENTER MOTOR VOLTAGE (VOLTS)',/)
      READ (1,215)MOTVOL
 215  FORMAT (F6.2)
      WRITE (1,220)
 220  FORMAT (' ENTER 90 AND 0 DEGREE VOLTAGES, RESPECTIVELY',/)
      READ (1,225)P90,P0
  225 FORMAT (2F7.4)
      RHO=(BAROM*70.45)/(1716.0*(460.0+TEMP))
      TUNVEL=SQRT((2.0*(5.204*MANOM2))/RHO)
C
C ----  Echo manometer readings, tunnel velocity, motor voltage and
C ----  90 and 0 degree angle of attack voltages for verification.
C ----  offer option to correct faulty input.
C
      WRITE (1,230)MANOM1
 230  FORMAT (' MANOMETER ONE: ',F8.4,' INCHES OF WATER')
      WRITE (1,233)MANOM2
 233  FORMAT (' MANOMETER TWO: ',F8.4,' INCHES OF WATER')
```

```
      WRITE (1,235)TUNVEL
 235  FORMAT (' TUNNEL VELOCITY: ',F7.2,' FT/SEC')
      WRITE (1,240)MOTVOL
 240  FORMAT (' MOTOR VOLTAGE: ',F6.2,' VOLTS')
      WRITE (1,245)P90,P0
 245  FORMAT (' P90: ',F7.4,' VOLTS     P0: ',F7.4,' VOLTS')
      WRITE (1,75)
      WRITE (1,80)
      READ (1,85)CHEK
      IF (CHEK.NE.1) GO TO 187
C
C ----  Initialize number or runs to zero, and then increment this
C ----  number by one each run thereafter.
C
      RUNS=0
 250  CONTINUE
      RUNS=RUNS+1
 255  CONTINUE
C
      WRITE (1,257)RUNS
 257  FORMAT (////,' ********RETURN AIRFOIL TO ZERO ANGLE OF
    + ATTACK IN PREPARATION FOR RUN',I2,'********',////)
      NS=0
      KOUNT=0
      WRITE (1,260)
 260  FORMAT (' ENTER NUMBER OF SAMPLES (MULTIPLE OF 18,
    + 5040 MAXIMUM)',/)
      READ (1,265)NS
 265  FORMAT (I5)
      WRITE (1,270)NS
 270  FORMAT (//,' ',25X,'NS:',I5,//)
C
C ----  In the next segement, the operator is given the choice
C ----  between manual and automatic trigger.
C
      WRITE (1,273)
 273  FORMAT (' DO YOU WANT MANUAL OR AUTOMATIC TRIGGER?
    + (1=AUTO, 2=MANUAL)',/)
      READ (1,277)SELECT
 277  FORMAT (I2)
      IF (SELECT.NE.1) GO TO 282
C
C ----  The program segement below is the automatic trigger.
C ----  The program stays in the 280 loop below until ZERANG
C ----  and VALUE differ by 2 or more digital counts.
C ----  When this occurs, due to rotation of the airfoil, the
C ----  program continues on to line number 285.
C
      CALL AD(VALUE,0,84)
      ZERANG=VALUE
 280  CALL AD(VALUE,0,84)
      SNAP=IABS(VALUE-ZERANG)
```

```
        IF (SNAP.LE.1) GO TO 280
        IF (SELECT.EQ.1) GO TO 285
  282   CONTINUE
        PAUSE
C
C ----  STCLK, below, will count up to 32,768 time clicks, each click
C ----  being .0010046 seconds long. Therefore, STCLK can only time
C ----  an event that lasts for no more than about 32 seconds.
C
  285   CALL STCLK
C
C ----  The following part of the program reads and stores the time
C ----  obtained from subroutine GETTIM, as well as position and
C ----  pressure information obtained from the potentiometer and
C ----  pressure transducers, respectively.  This position and pressur
C ----  information is obtained through subroutine ADIO.
C
        WRITE(1,290)
  290   FORMAT(///,' ',20X,'STARTING TO TAKE DATA',///)
        DO 320 J=1,NS,18
        KOUNT=KOUNT+1
        CALL GETTIM(TIME)
        IDATA(J)=TIME
        CHAN=0
        CALL AD(VALUE,CHAN,84)
        IDATA(J+1)=VALUE
        DO 300 K=1,16
        CHAN=K-1
        CALL AD(VALUE,CHAN,80)
        DI=K+J+1
        IDATA(DI)=VALUE
C
C ----  The do 305 loop below is a simple delay loop and can be used to
C ----  govern the number of time clicks that will elapse between pass
C ----  One click = .0010046 seconds and one pass is defined as one
C ----  complete cycle around the 18 data channels.  To find what ZZ
C ----  should approximately be, use the following equation:
C
C                      CLICKS/CYCLE=(0.46)*ZZ + 4.0
C
C ----  CAUTION: Making ZZ too large will result in data points that
C ----  are too far apart, thereby invalidating the linear interpo-
C ----  lation that is done in the data reduction program, RED.FOR
C
C       DO 305 ZZ=1,40
C305    CONTINUE
  300   CONTINUE
  320   CONTINUE
        WRITE (1,330)RUNS
  330   FORMAT (' ',15X,'DATA GATHERING COMPLETE FOR RUN',I2,//)
        WRITE (1,340)KOUNT
  340   FORMAT (' NUMBER OF PASSES = ',I6,//)
```

```
      N=KOUNT*18
      WRITE (1,343)N
 343  FORMAT (' NUMBER OF IDATA ELEMENTS= ',I6,//  )
      VPD=(P90-P0)/90.0
      DTIM=(IDATA(1801)-IDATA(343))*(0.0010046)
      DPOSV=((IDATA(1802)-IDATA(344))/4096.0)*10.0
      DPOSD=DPOSV/VPD
      ROTRAT=DPOSD/DTIM
      WRITE (1,410)ROTRAT
 410  FORMAT (' AIRFOIL AVERAGE ROTATION RATE:',F5.2,' DEG/SEC',////)
C
C ---- Options are now offered to list the IDATA array at the
C ---- terminal, to write this array to disk, and to repeat the
C ---- data run.
C
      WRITE(1,345)
 345  FORMAT(' DO YOU WANT TO LIST THE IDATA ARRAY?(Y=1)',//)
      READ(1,347)AA
 347  FORMAT (I2)
      IF (AA.NE.1)GO TO 350
      DO 420 XXX=180,N,180
      YYY=XXX-179
      WRITE (1,360)(IDATA(L),L=YYY,XXX)
 360  FORMAT (9I7)
      PAUSE
 420  CONTINUE
      WRITE (1,351)
 351  FORMAT (//)
 350  WRITE(1,355)
 355  FORMAT(' DO YOU WANT TO WRITE TO DISK?(Y=1)',//  )
      READ (1,347)B
      IF (B.EQ.1) GO TO 390
      WRITE (1,375)RUNS
 375  FORMAT (' DO YOU WANT TO REPEAT RUN',I2,'? (Y=1)',//)
      READ (1,380)C
 380  FORMAT (I2)
      IF (C.EQ.1) GO TO 255
      IF (C.NE.1) GO TO 4800
 390  CONTINUE
C
C ---- The following part of the program writes pertinent
C ---- information to file RAWDATA0DAT on disk.
C
      IF (RUNS.NE.1) GO TO 705
      CALL OPEN (3,'RAWDATA0DAT',2)
      WRITE (3,500)
 500  FORMAT (' DAY',10X,'MONTH',9X,'YEAR',9X,'TIME')
      WRITE (3,510)DAY,MONTH,YEAR,HOUR
 510  FORMAT (I3,11X,I3,11X,I3,9X,I5,/)
      WRITE (3,520)
 520  FORMAT (' TEMPERATURE',14X,'BAROMETER')
      WRITE (3,530)TEMP,BAROM
```

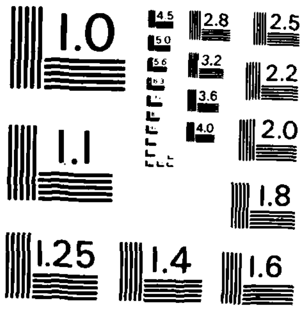
```
 530    FORMAT (2X,F6.1,18X,F7.2,/)
        WRITE (3,540)
 540    FORMAT (' MANOMETER 1',22X.'MANOMETER 2')
        WRITE (3,545)MANOM1,MANOM2
 545    FORMAT (2X,F8.4,25X,F8.4,/)
        WRITE (3,550)
 550    FORMAT (' TUNNEL VELOCITY',22X,'MOTOR VOLTAGE')
        WRITE (3,555)TUNVEL,MOTVOL
 555    FORMAT (4X,F7.2,31X,F6.2,/)
        WRITE (3,560)
 560    FORMAT (' 90 DEG. VOLTAGE',16X,'0 DEG. VOLTAGE')
        WRITE (3,570)P90,P0
 570    FORMAT (5X,F7.4,23X,F7.4,/)
        WRITE (3,580)
 580    FORMAT (' NUMBER OF PASSES',10X,'NUMBER OF IDATA ELEMENTS')
        WRITE (3,590)
 590    FORMAT (5X,'(KOUNT)',26X,'(N)')
        WRITE (3,600)KOUNT,N
 600    FORMAT (3X,I6,26X,I6,//)
        WRITE (3,610)
 610    FORMAT (' AVERAGE ZERO-INPUT READINGS GIVEN BELOW',/)
        WRITE (3,620)AVSTAT(1),AVSTAT(2),AVSTAT(3),AVSTAT(4)
        WRITE (3,620)AVSTAT(5),AVSTAT(6),AVSTAT(7),AVSTAT(8)
        WRITE (3,620)AVSTAT(9),AVSTAT(10),AVSTAT(11),AVSTAT(12)
        WRITE (3,620)AVSTAT(13),AVSTAT(14),AVSTAT(15),AVSTAT(16)
 620    FORMAT (F9.3,5X,F9.3,5X,F9.3,5X,F9.3)
        WRITE (3,660)
 660    FORMAT (///)
 705    CONTINUE
C
C ----  The part of the program below writes the collected data
C ----  to disk, in unformatted form, under the filename
C ----  RAWDATA1DAT, RAWDATA2DAT, . . . . , RAWDATA5DAT, depending
C ----  on the value of the variable RUNS. To view the data files
C ----  that are in unformatted form, use program LOOK.
C
        IF (RUNS.EQ.1) GO TO 710
        IF (RUNS.EQ.2) GO TO 720
        IF (RUNS.EQ.3) GO TO 730
        IF (RUNS.EQ.4) GO TO 740
        IF (RUNS.EQ.5) GO TO 750
C
 710    CONTINUE
        CALL OPEN (4,'RAWDATA1DAT',2)
        WRITE (4)(IDATA(L),L=1,N)
        GO TO 760
 720    CONTINUE
        CALL OPEN (5,'RAWDATA2DAT',2)
        WRITE (5)(IDATA(L),L=1,N)
        GO TO 760
 730    CONTINUE
        CALL OPEN (6,'RAWDATA3DAT',2)
```

```
      WRITE (6)(IDATA(L),L=1,N)
      GO TO 760
 740  CONTINUE
      CALL OPEN (7,'RAWDATA4DAT',2)
      WRITE (7)(IDATA(L),L=1,N)
      GO TO 760
 750  CONTINUE
      CALL OPEN (8,'RAWDATA5DAT',2)
      WRITE (8)(IDATA(L),L=1,N)
      GO TO 760
 760  CONTINUE
      IF (RUNS.NE.5) GO TO 250
      WRITE (1,2345)
 2345 FORMAT (//////,' FOLLOWING PART OF PROGRAM GIVES STATIC
     + NORMAL COEFF. FOR STATIC ALPHA',//////)
C
C ----  The remaining portion of the program takes and processes
C ----  data for static angle of attack lift-curves.
C
 2400 CONTINUE
      WRITE (1,2450)
 2450 FORMAT (' ENTER NS (MULTIPLE OF 18, LESS THAN OR
     + EQUAL TO 396)',/)
      READ (1,2150)NS
 2150 FORMAT (I4)
      KOUNT=0
      WRITE (1,2000)
 2000 FORMAT (////,' HIT RETURN TO START DATA COLLECTION',/)
      PAUSE
C
C ----  STCLK, below, will count up to 32,768 time clicks, each click
C ----  being .0010046 seconds long. Therefore, STCLK can only time
C ----  an event that lasts for no more than about 32 seconds.
C
      CALL STCLK
C
      WRITE(1,2100)
 2100 FORMAT(///,' ',20X,'STARTING TO TAKE DATA',///)
      DO 2200 J=1,NS,18
      KOUNT=KOUNT+1
      CALL GETTIM(TIME)
      IDATA(J)=TIME
      CHAN=0
      CALL AD(VALUE,CHAN,84)
      IDATA(J+1)=VALUE
      DO 2300 K=1,16
      CHAN=K-1
      CALL AD(VALUE,CHAN,80)
      DI=K+J+1
      IDATA(DI)=VALUE
 2300 CONTINUE
 2200 CONTINUE
```

AD-A136 820 CONTINUED EXPERIMENTAL INVESTIGATION OF DYNAMIC STALL (U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF ENGINEERING S J SCHRECK DEC 83 2/2

UNCLASSIFIED AFIT/GAE/AA/83D-21 F/G 20/4 NL

END
DATE
FILMED
2 84
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
              N=KOUNT*18
              WRITE (1,2500)N
       2500   FORMAT (' NUMBER OF IDATA ELEMENTS= ',I6,//)
              PAUSE
      C
      C ----  Time-average data
      C
              DO 2550 S=1,16
              IDATAT(S)=0.0
       2550   CONTINUE
              DO 2600 II=1,N,18
              DO 2700 JJ=3,18
              TT=II+JJ
              IDATAT(JJ-2)=((IDATA(TT-1))/KOUNT)+IDATAT(JJ-2)
       2700   CONTINUE
       2600   CONTINUE
      C
      C ----  Compute the pressure coefficients
      C
              DO 2800 KK=1,16
              STICKY=AVSTAT(KK)-2048.0
              PRESS=(((IDATAT(KK)-STICKY)-2048.0)/2048.0)*(50.0/SENS(KK))
              CP(KK)=(PRESS+(MANOM1/27.68))/(MANOM2/27.68)
       2800   CONTINUE
      C
      C ---- The next loop defines the pressure distribution on the upper
      C ---- surface of the airfoil, leading edge to trailing edge.
      C ---- Pressure coefficient is assumed to be zero at the trailing edge
      C
             WRITE (1,2900)
       2900  FORMAT (' UPPER SURFACE PRESSURE COEFFICIENTS,
           + L.E. TO T.E., ARE GIVEN BELOW',)
             DO 3000 V=1,9
             CPU(V)=CP(V)
       3000  CONTINUE
             CPU(10)=0.0
             DO 3100 V=1,10
             WRITE (1,3200)V,CPU(V)
       3200  FORMAT (' CPU',I3,'=',F8.4)
       3100  CONTINUE
      C
             WRITE (1,3300)
       3300  FORMAT (/,' LOWER SURFACE PRESSURE COEFFICIENTS,
           + L.E. TO T.E., ARE GIVEN BELOW')
             CPL(1)=CP(1)
             DO 3400 W=2,8
             DD=18-W
             CPL(W)=CP(DD)
       3400  CONTINUE
             CPL(9)=CPU(10)
             DO 3500 W=1,9
             WRITE (1,3600)W,CPL(W)
```

```
 3600   FORMAT (' CPL',I3,'=',F8.4)
 3500   CONTINUE
C
C ---- The following loop integrates the upper pressure
C ---- distribution using the trapezoidal rule.
C
        AREAUT=0.0
        DO 3700 X=1,9
        LNGTHU=PORTU(X+1)-PORTU(X)
        IF ((ABS(CPU(X+1)-CPU(X))).GT.(ABS((0.01)*CPU(X)))) GO TO 3800
        AREAU=(0.5)*(CPU(X+1)+CPU(X))*LNGTHU
 3800   IF ((ABS(CPU(X+1)-CPU(X))).LE.(ABS((0.01)*CPU(X)))) GO TO 4000
        INTU=(PORTU(X)-PORTU(X+1))*CPU(X)/(CPU(X+1)-CPU(X))
        IF (INTU.LT.LNGTHU) GO TO 3900
        AREAU=(.5)*(CPU(X+1)+CPU(X))*LNGTHU
        IF ((INTU).GE.(LNGTHU)) GO TO 4000
 3900   AREAU=((.5)*INTU*CPU(X))+
     +                ((.5)*(LNGTHU-INTU)*CPU(X+1))
 4000   AREAUT=AREAUT+AREAU
 3700   CONTINUE
C
C ---- The following loop integrates the lower pressure
C ---- distribution using the trapezoidal rule.
C
        AREALT=0.0
        DO 4100 Y=1,8
        LNGTHL=PORTL(Y+1)-PORTL(Y)
        IF ((ABS(CPL(Y+1)-CPL(Y))).GT.(ABS((0.01)*CPL(Y)))) GO TO 4200
        AREAL=(.5)*(CPL(Y+1)+CPL(Y))*LNGTHL
        IF ((ABS(CPL(Y+1)-CPL(Y))).LE.(ABS((0.01)*CPL(Y)))) GO TO 4400
 4200   INTL=(PORTL(Y)-PORTL(Y+1))*CPL(Y)/(CPL(Y+1)-CPL(Y) )
        IF ((INTL).LT.(LNGTHL)) GO TO 4300
        AREAL=(.5)*(CPL(Y+1)+CPL(Y))*LNGTHL
        IF ((INTL).GE.(LNGTHL)) GO TO 4400
 4300   AREAL=((.5)*INTL*CPL(Y))+
     +                ((.5)*(LNGTHL-INTL)*CPL(Y+1))
 4400   AREALT=AREALT+AREAL
 4100   CONTINUE
C
        NORMCO=AREALT-AREAUT
C
        WRITE (1,4500)NORMCO
 4500   FORMAT (/,' NORMAL FORCE COEFFICIENT=',F8.5,/)
        PAUSE
C
C ----  Option now offered to repeat the data run.
C
         WRITE (1,4600)
 4600    FORMAT (' DO YOU WANT TO REPEAT THE RUN?  (N=1)',/)
         READ (1,4700)CCC
 4700    FORMAT (I2)
         IF (CCC.NE.1) GO TO 2400
```

```
 4200   CONTINUE
        STOP
        END
```

```
      PROGRAM RED
      IMPLICIT INTEGER (A-Z)
      INTEGER IDATA(5050),IDATA1(30),IDATA2(300),PASS(30)
      INTEGER N,R,X,Y,V,W,S,I,J,L,AA,PP,QQ
      INTEGER RR,SS,TT,UU,VV,WW,XX,YY,ZZ,RUN,TRAP,PAZZ,DIV,NUMEL
      INTEGER ELEM1,ELEM2,DAY,MONTH,YEAR,HOUR,CHANG1,CHANG2
      INTEGER DD,EE,FF,HH,LL,NN
      REAL PORTU(10),PORTL(10),CP(16),CPU(10),CPL(10),SENS(16)
      REAL PRESS(16),REDAT(40),P90,P0,TEMP,BAROM,MANOM1,MANOM2
      REAL TUNVEL,MOTVOL,AVSTAT(16),AREAUT,AREALT,RE,RHO,MU
      REAL VPD,AOA,INTU,INTL,NORMCO,TUNQ,LNGTHU,LNGTHL
      REAL AREAU,AREAL,DTIM,DPOSD,DPOSV,ROTRAT,NDRATE
      REAL REDATC(40),IDATAT(1100)
C
C ---- Load transducer sensitivities (millivolts/psi)
      DATA SENS/206.5,174.2,174.9,237.5,207.0,205.2,243.1,
     +221.2,172.8,113.1,118.3,111.5,138.7,177.5,221.9,225.2/
C
C ---- Load transducer locations on upper surface (percent chord)
      DATA PORTU/0.0,0.0242,0.0484,0.0969,0.129,0.194,0.323,0.605,
     +0.888,1.000/
C
C ---- Load transducer locations on lower surface (percent chord)
      DATA PORTL/0.0,0.0161,0.0319,0.0484,0.0969,0.194,0.323,
     +0.686,1.000/
      WRITE (1,5)
    5 FORMAT (///,' *****THE DATA FILES TO BE REDUCED MUST BE ON
     + DISK DRIVE B AND MUST BE NAMED*****')
      WRITE (1,6)
    6 FORMAT (' *************RAWDATA0.DAT, RAWDATA1.DAT,. . . . . . .
     + RAWDATA5.DAT*************',///)
C
C ---- Read raw data from RAWDATA0DAT on drive B.
      CALL OPEN(3,'RAWDATA0DAT',2)
      READ (3,10)DAY,MONTH,YEAR,HOUR
   10 FORMAT (/,I3,11X,I3,11X,I3,9X,I5)
      READ (3,20)TEMP,BAROM
   20 FORMAT (//,2X,F6.1,18X,F7.2)
      READ (3,30)MANOM1,MANOM2
   30 FORMAT (//,2X,F8.4,25X,F8.4)
      READ (3,40)TUNVEL,MOTVOL
   40 FORMAT (//,4X,F7.2,31X,F6.2)
      READ (3,50)P90,P0
   50 FORMAT (//,5X,F7.4,23X,F7.4)
      READ (3,60)KOUNT,N
   60 FORMAT (///,3X,I6,26X,I6)
      READ (3,70)AVSTAT(1),AVSTAT(2),AVSTAT(3),AVSTAT(4)
      READ (3,75)AVSTAT(5),AVSTAT(6),AVSTAT(7),AVSTAT(8)
      READ (3,75)AVSTAT(9),AVSTAT(10),AVSTAT(11),AVSTAT(12)
      READ (3,75)AVSTAT(13),AVSTAT(14),AVSTAT(15),AVSTAT(16)
   70 FORMAT (////,F9.3,5X,F9.3,5X,F9.3,5X,F9.3)
   75 FORMAT (F9.3,5X,F9.3,5X,F9.3,5X,F9.3)
```

```
      VFD=(PPA-PU)*90.0
C
C ---- The following part of the program reads raw digital angle of
C ---- attack from data sets 1, 11, 21, 31, . . . . , etc., in the
C ---- IDATA array (where a data set is defined as the 18 data
C ---- elements read in one sampling pass). These data sets were
C ---- previously stored in the IDATA array in the first data file,
C ---- RAWDATA1DAT, by the data gathering program BIG. After these
C ---- angles of attack are read from the IDATA array in RAWDATADAT1.
C ---- they are stored sequentially in the array IDATA1.
C
      ELEM1=0
      CALL OPEN(4,'RAWDATA1DAT',2)
      READ(4)(IDATA(L),L=1,N)
      DO 500 PP=2,N,180
      ELEM1=ELEM1+1
      IDATA1(ELEM1)=IDATA(PP)
 500  CONTINUE
C
C ---- The DO 650 loop below begins by branching out to one of 4
C ---- program locations depending on the number of times the
C ---- computer has been through the 650 loop. The program segements
C ---- beginning at 510,525,535 and 545 all perform the same
C ---- operation, but do so on different data files, in turn.
C ---- The operation performed in all 4 segements is as follows.
C ---- The loop contained in each segement reads the raw digital
C ---- angle of attack from every data set in the array IDATA.
C ---- These data sets were previously stored in the second through
C ---- fifth data files, RAWDATA2DAT through RAWDATA5DAT by the data
C ---- gathering program UNO. After these angles of attack are read
C ---- from the IDATA array in the proper data file, they are stored
C ---- in the array IDATA2. Note that the contents of the IDATA2 array
C ---- are replaced each time the DO 650 loop is travelled.
C
      DO 650 RUN=2,5
      ELEM2=0
      IF (RUN.EQ.2) GO TO 510
      IF (RUN.EQ.3) GO TO 525
      IF (RUN.EQ.4) GO TO 535
      IF (RUN.EQ.5) GO TO 545
 510  CONTINUE
      CALL OPEN(5,'RAWDATA2DAT',2)
      READ(5)(IDATA(L),L=1,N)
      DO 520 QQ=2,N,18
      ELEM2=ELEM2+1
      IDATA2(ELEM2)=IDATA(QQ)
 520  CONTINUE
      GO TO 550
 525  CONTINUE
      CALL OPEN(6,'RAWDATA3DAT',2)
      READ(6)(IDATA(L),L=1,N)
      DO 530 QQ=2,N,18
```

```
      ELEM2=ELEM2+1
      IDATA2(ELEM2)=IDATA(QQ)
 530  CONTINUE
      GO TO 550
 535  CONTINUE
      CALL OPEN(7,'RAWDATA4DAT',2)
      READ(7)(IDATA(L),L=1,N)
      DO 540 QQ=2,N,18
      ELEM2=ELEM2+1
      IDATA2(ELEM2)=IDATA(QQ)
 540  CONTINUE
      GO TO 550
 545  CONTINUE
      CALL OPEN(8,'RAWDATA5DAT',2)
      READ(8)(IDATA(L),L=1,N)
      DO 550 QQ=2,N,18
      ELEM2=ELEM2+1
      IDATA2(ELEM2)=IDATA(QQ)
 550  CONTINUE
C
C ---- We now have an array, IDATA1, containing every tenth angle of
C ---- attack element from disk file RAWDATA1DAT. We also have an
C ---- array, IDATA2, containing every angle of attack element from
C ---- one of the four remaining disk files. The program segement
C ---- below compares each element in IDATA1 with every element in
C ---- IDATA2, in turn. When the closest match is found, this infor
C ---- mation is stored in the PASS array for use in the next pro-
C ---- gram segement.
C
      RR=0
      DO 560 SS=1,ELEM1
      CHANG2=4096
      TRAP=0
      DO 570 TT=1,ELEM2
      RR=RR+1
      CHANG1=IABS(IDATA1(SS)-IDATA2(TT))
      IF (CHANG1.GE.CHANG2) GO TO 580
      CHANG2=CHANG1
      GO TO 570
 580  TRAP=TRAP+1
      IF (TRAP.NE.1) GO TO 570
      PASS(SS)=RR-1
 570  CONTINUE
 560  CONTINUE
C
C ---- The next part, up to and including line 610, reads
C ---- two data sets every ten data sets, ie, 1,2, 11,12, 21,22,
C ---- etc., from disk file RAWDATA1DAT, and then repacks each
C ---- data set sequentially in array IDATAT, ie, data sets
C ---- 1,2,11,12,21,22, . . . . . in IDATA become data sets
C ---- 1,2,3,4,5,6, . . . . . in IDATAT.
C
```

```
      IF (RUN.NE.2) GO TO 630
      REWIND 4
      READ(4)(IDATA(L),L=1,N)
      CALL OPEN(9,'AVRGDATADAT',2)
      PAZZ=0
      DO 610 UU=1,N,180
      PAZZ=PAZZ+1
      DO 620 VV=1,36
      DD=(UU+VV)-1
      WW=VV+(36*(PAZZ-1))
      IDATAT(WW)=IDATA(DD)
 620  CONTINUE
 610  CONTINUE
      NN=PAZZ*36
      WRITE(9)(IDATAT(LL),LL=1,NN)
 630  CONTINUE
C
C ---- The following program segements beginning on lines 640,665,
C ---- 690, and 720 perform the same operation, but do so on
C ---- different disk files, in turn. The operation performed is as
C ---- follows. The information previously obtained and stored in
C ---- the array PASS is now used to determine which data sets will
C ---- be read (from whichever disk file of the remaining four that
C ---- is then being worked on), then added, element by element, to
C ---- the element totals in the array IDATAT. Note that the contents
C ---- of the array IDATA changes each time the computer passes throug
C ---- this part of the program, but the contents of array IDATAT is
C ---- retained and added to each time the computer passes through thi
C ---- part of the program.
C
      IF (RUN.EQ.2) GO TO 640
      IF (RUN.EQ.3) GO TO 665
      IF (RUN.EQ.4) GO TO 690
      IF (RUN.EQ.5) GO TO 720
 640  CONTINUE
      REWIND 9
      REWIND 5
      READ(5)(IDATA(L),L=1,N)
      READ(9)(IDATAT(LL),LL=1,NN)
      DO 655 XX=1,ELEM1
      DO 660 YY=1,36
      ZZ=(XX-1)*36
      DD=((PASS(XX)-1)*18)+YY
      FF=YY+ZZ
      IDATAT(FF)=IDATAT(FF)+IDATA(DD)
 660  CONTINUE
 655  CONTINUE
      WRITE(9)(IDATAT(LL),LL=1,NN)
      GO TO 649
 665  CONTINUE
      REWIND 9
      REWIND 6
```

```
      READ(6)(IDATA(L),L=1,N)
      READ(9)(IDATAT(LL),LL=1,NN)
      DO 670 XX=1,ELEM1
      DO 680 YY=1,36
      ZZ=(XX-1)*36
      DD=((PASS(XX)-1)*18)+YY
      FF=YY+ZZ
      IDATAT(FF)=IDATAT(FF)+IDATA(DD)
680   CONTINUE
670   CONTINUE
      WRITE(9)(IDATAT(LL),LL=1,NN)
      GO TO 649
690   CONTINUE
      REWIND 9
      REWIND 7
      READ(7)(IDATA(L),L=1,N)
      READ(9)(IDATAT(L),LL=1,NN)
      DO 700 XX=1,ELEM1
      DO 710 YY=1,36
      ZZ=(XX-1)*36
      DD=((PASS(XX)-1)*18)+YY
      FF=YY+ZZ
      IDATAT(FF)=IDATAT(FF)+IDATA(DD)
710   CONTINUE
700   CONTINUE
      WRITE(9)(IDATAT(LL),LL=1,NN)
      GO TO 649
720   CONTINUE
      REWIND 9
      REWIND 8
      READ(8)(IDATA(L),L=1,N)
      READ(9)(IDATAT(LL),LL=1,NN)
      DO 730 XX=1,ELEM1
      DO 740 YY=1,36
      ZZ=(XX-1)*36
      DD=((PASS(XX)-1)*18)+YY
      FF=YY+ZZ
      IDATAT(FF)=IDATAT(FF)+IDATA(DD)
740   CONTINUE
730   CONTINUE
      WRITE(9)(IDATAT(LL),LL=1,NN)
649   CONTINUE
650   CONTINUE
C
C ---- We now have array IDATAT, having as its individual elements
C ---- the total of five individual IDATA elements from each of
C ---- five elements. To get the average value of these, we must
C ---- divide the total we have by 5. This division is done in the
C ---- program segement below. This average value is then stored
C ---- in the array IDATAT for further processing.
C
      REWIND 9
```

```
      READ(9)(IDATAT(L),L=1,NN)
      DO 750 DIV=1,NN
      IDATAT(DIV)=(IDATAT(DIV))/5.0
 750  CONTINUE
      WRITE(9)(IDATAT(L),L=1,NN)
C
C ---- The steps below compute airfoil rotation rate (deg/sec),
C ---- non-dimensional rotation rate, Reynolds number, tunnel "Q"
C ---- and volts per degree for the run.
C
      REWIND 9
      READ(9)(IDATAT(L),L=1,NN)
      DTIM=(IDATAT(451)-IDATAT(91))*(0.0010046)
      DPOSV=((IDATAT(452)-IDATAT(92))/4096.0)*10.0
      DPOSD=DPOSV/VPD
      ROTRAT=DPOSD/DTIM
      NDRATE=(ROTRAT*.017453*1.016)/(2.0*TUNVEL)
      RHO=(BAROM*70.45)/(1716.0*(460+TEMP))
      MU=(2.270*(10.0**(-8.0))*((460.0+TEMP)**1.5))/(460.0+TEMP+198.
      RE=(RHO*TUNVEL*1.016)/MU
      TUNQ=(0.5)*RHO*(TUNVEL**2)
C
C ---- The following writes pertinent information to disk file
C ---- REDUDATADAT as a heading.
C
      CALL OPEN(10,'REDUDATADAT',2)
      WRITE (10,800)
 800  FORMAT (' DAY',10X,'MONTH',9X,'YEAR',9X,'TIME')
      WRITE (10,810)DAY,MONTH,YEAR,HOUR
 810  FORMAT (I3,11X,I3,11X,I3,9X,I5,/)
      WRITE (10,820)
 820  FORMAT (' TEMPERATURE',14X,'BAROMETER')
      WRITE (10,830)TEMP,BAROM
 830  FORMAT (2X,F6.1,18X,F7.2,/)
      WRITE (10,840)
 840  FORMAT (' MANOMETER 1',22X,'MANOMETER 2')
      WRITE (10,845)MANOM1,MANOM2
 845  FORMAT (2X,F8.4,25X,F8.4,/)
      WRITE (10,850)
 850  FORMAT (' TUNNEL VELOCITY',22X,'MOTOR VOLTAGE')
      WRITE (10,855)TUNVEL,MOTVOL
 855  FORMAT (4X,F7.2,31X,F6.2,/)
      WRITE (10,860)
 860  FORMAT (' DEGREES/SECOND',15X,'NON-DIMENSIONAL RATE')
      WRITE (10,870)ROTRAT,NDRATE
 870  FORMAT(4X,F6.2,25X,F8.7,/)
      WRITE (10,880)
 880  FORMAT (' REYNOLDS NUMBER',25X,'TUNNEL "Q"')
      WRITE (10,890)RE,TUNQ
 890  FORMAT (4X,E11.4,30X,F6.3,/)
      DO 895 HH=1,16
      WRITE (10,897)HH,AVSTAT(HH)
```

```
 397   FORMAT (' AVERAGE ZERO-INPUT READING, TRANSDUCER',I3,' = ',F8.
 395   CONTINUE
C
C ---- One pass through the DO 100 J=1,N,18 loop computes one
C ---- point in the CN (normal force coefficient) versus ALPHA curve.
C
       REWIND 9
       READ(9)(IDATAT(L),L=1,NN)
       DO 100 J=1,NN,36
       REDAT(1)=IDATAT(J)
       REDAT(2)=IDATAT(J+1)
       REDAT(19)=IDATAT(J+18)
       REDAT(20)=IDATAT(J+19)
C
C ---- The loop below subtracts the average zero input readings
C ---- (AVSTAT) from each appropriate IDATAT element.
C
       DO 120 I=1,16
       AA=(J+I)+1
       REDAT(I+2)=IDATAT(AA)-((AVSTAT(I))-2048.0)
       REDAT(I+20)=IDATAT(AA+18)-((AVSTAT(I))-2048.0)
 120   CONTINUE
C
C ---- Operations in the following loop correct for the finite
C ---- time between samples using a linear interpolation. Time
C ---- between passes must be sufficiently small or the linear
C ---- interpolation will be invalid.
C
       DO 140 R=1,18
       REDATC(R)=REDAT(R+18)-((REDAT(R+18)-REDAT(R))*((R-1)/18.0))
 140   CONTINUE
C
C ---- The following loop converts digital quantities to degrees
C ---- (angle of attack) and psi (sensed differential pressure).
C
C ---- The AOA conversion below assumes the A/D board is strapped
C ---- for the 0-10 volt unipolar input range. The amp on the
C ---- board is set for a gain of 1, so any input to the board
C ---- greater than 10 volts will saturate the A/D conversion system.
C
       AOA=(((REDATC(2)/4096.0)*10.0)-P0)/VPD
       DO 160 S=1,16
       BB=(J+S)+1
C
C ---- The PRESS conversion below assumes the A/D board is strapped
C ---- for the (-5)-(+5) volt bipolar input range, where the input
C ---- (from the transducers) is first amplified through an
C ---- amplifier of gain 100. So any input greater than +/-50 milli-
C ---- volts will saturate the A/D conversion system.
C
       PRESS(S)=((REDATC(S+2)-2048.0)/2048.0)*50.0/SENS(S)
       CP(S)=(PRESS(S)+(MANOM1/27.68))/(MANOM2/27.68)
```

```
 160   CONTINUE
C
C ---- The next loop defines the pressure distribution on the upper
C ---- surface of the airfoil, leading edge to trailing edge.
C ---- Pressure coefficient is assumed to be zero at the trailing edge
C
       WRITE (10,170)REDAT(19)
 170   FORMAT (///,'TIME BACK TO WHICH DATA FOR THIS PASS HAS BEEN
     + CORRECTED:',F6.0,///)
       WRITE (10,185)
 185   FORMAT (' UPPER SURFACE PRESSURE COEFFICIENTS,
     + L.E. TO T.E., ARE GIVEN BELOW',/)
       DO 200 V=1,9
       CPU(V)=CP(V)
 200   CONTINUE
       CPU(10)=CPU(9)+(0.341*(CPU(9)-CPU(8)))
       DO 195 V=1,10
       WRITE (10,190)V,CPU(V)
 190   FORMAT (' CPU',I3,'=',F8.4)
 195   CONTINUE
C
C ---- The next loop defines the pressure distribution on the lower
C ---- surface of the airfoil, leading edge to trailing edge.
C ---- Pressure coefficient is assumed to be zero at the trailing edge
C
       WRITE (10,205)
 205   FORMAT (//,' LOWER SURFACE PRESSURE COEFFICIENTS,
     + L.E. TO T.E., ARE GIVEN BELOW',/)
       CPL(1)=CP(1)
       DO 220 W=2,8
       DD=18-W
       CPL(W)=CP(DD)
 220   CONTINUE
       CPL(9)=CPU(10)
       DO 215 W=1,9
       WRITE (10,225)W,CPL(W)
 225   FORMAT (' CPL',I3,'=',F8.4)
 215   CONTINUE
C
C ---- The following loop integrates the upper pressure
C ---- distribution using the trapezoidal rule.
C
       AREAUT=0.0
       DO 240 X=1,9
       LNGTHU=PORTU(X+1)-PORTU(X)
       IF ((ABS(CPU(X+1)-CPU(X))).GT.(ABS((0.01)*CPU(X)))) GO TO 245
       AREAU=(0.5)*(CPU(X+1)+CPU(X))*LNGTHU
 245   IF ((ABS(CPU(X+1)-CPU(X))).LE.(ABS((0.01)*CPU(X)))) GO TO 235
       INTU=(PORTU(X)-PORTU(X+1))*CPU(X)/(CPU(X+1)-CPU(X))
       IF (INTU.LT.LNGTHU) GO TO 260
```

```
      AREAU=(.5)*(CPU(X+1)+CPU(X))*LNGTHU
      IF ((INTU).GE.(LNGTHU)) GO TO 235
 260  AREAU=((.5)*INTU*CPU(X))+
     +                 ((.5)*(LNGTHU-INTU)*CPU(X+1))
 235  AREAUT=AREAUT+AREAU
 240  CONTINUE
C
C ---- The following loop integrates the lower pressure
C ---- distribution using the trapezoidal rule.
C
      AREALT=0.0
      DO 280 Y=1,8
      LNGTHL=PORTL(Y+1)-PORTL(Y)
      IF ((ABS(CPL(Y+1)-CPL(Y))).GT.(ABS((0.01)*CPL(Y)))) GO TO 270
      AREAL=(.5)*(CPL(Y+1)+CPL(Y))*LNGTHL
      IF ((ABS(CPL(Y+1)-CPL(Y))).LE.(ABS((0.01)*CPL(Y)))) GO TO 275
 270  INTL=(PORTL(Y)-PORTL(Y+1))*CPL(Y)/(CPL(Y+1)-CPL(Y))
      IF ((INTL).LT.(LNGTHL)) GO TO 300
      AREAL=(.5)*(CPL(Y+1)+CPL(Y))*LNGTHL
      IF ((INTL).GE.(LNGTHL)) GO TO 275
 300  AREAL=((.5)*INTL*CPL(Y))+
     +                 ((.5)*(LNGTHL-INTL)*CPL(Y+1))
 275  AREALT=AREALT+AREAL
 280  CONTINUE
C
      NORMCO=AREALT-AREAUT
C
      WRITE (10,900)AOA
 900  FORMAT (' ANGLE OF ATTACK=',F7.3,'  DEGREES')
      WRITE (10,910)NORMCO
 910  FORMAT (' NORMAL FORCE COEFFICIENT=',F8.5,/)
      WRITE (10,950)
 950  FORMAT (' ******************************************************
     +*****************',///)
 100  CONTINUE
      STOP
      END
```

```
      PROGRAM DEN
      IMPLICIT INTEGER (A-Z)
      INTEGER IDATA(5050),IDATA1(35),IDATA2(290),PASS(35)
      INTEGER N,R,X,Y,V,W,S,I,J,L,AA,PP,QQ
      INTEGER RR,SS,TT,UU,VV,WW,XX,YY,ZZ,RUN,TRAP,PAZZ,DIV,NUMEL
      INTEGER ELEM1,ELEM2,DAY,MONTH,YEAR,HOUR,CHANG1,CHANG2
      INTEGER DD,EE,FF,HH,LL,NN,BEG,START,END,FWD,HLT,NONO
      REAL PORTU(10),PORTL(10),CP(16),CPU(10),CPL(10),SENS(16)
      REAL IDATAT(1260),REDATC(36),REDAT(36),PRESS(16)
      REAL P90,P0,TEMP,BAROM,MANOM1,MANOM2
      REAL TUNVEL,MOTVOL,AVSTAT(16),AREAUT,AREALT,RE,RHO,MU
      REAL VPD,AOA,INTU,INTL,NORMCO,TUNQ,LNGTHU,LNGTHL
      REAL AREAU,AREAL,DTIM,DPOSD,DPOSV,ROTRAT,NDRATE
      REAL AOASEP,RAWSEP,HIGH,LOW,TAKE
C
C ---- Load transducer sensitivities (millivolts/psi)
      DATA SENS/206.5,174.2,174.9.237.5,207.0,205.2,243.1,
     +221.2,172.8,113.1,118.3,111.5,138.7,177.5,221.9.225.2/
C
C ---- Load transducer locations on upper surface (percent chord)
      DATA PORTU/0.0,0.0242,0.0484,0.0969,0.129,0.194,0.323,0.605,
     +0.888,1.000/
C
C ---- Load transducer locations on lower surface (percent chord)
      DATA PORTL/0.0,0.0161,0.0319,0.0484,0.0969,0.194,0.323,
     +0.686,1.000/
      WRITE (1,5)
    5 FORMAT (///,' *****THE DATA FILES TO BE REDUCED MUST BE ON
     + DISK DRIVE B AND MUST BE NAMED*****')
      WRITE (1,6)
    6 FORMAT (' *************RAWDATA0.DAT, RAWDATA1.DAT,. . . . . . .
     + RAWDATA5.DAT*************',///)
C
C ---- Read raw data from RAWDATA0DAT on drive B.
      CALL OPEN(3,'RAWDATA0DAT',2)
      READ (3,10)DAY,MONTH,YEAR,HOUR
   10 FORMAT (/,I3,11X,I3,11X,I3,9X,I5)
      READ (3,20)TEMP,BAROM
   20 FORMAT (//,2X,F6.1,18X,F7.2)
      READ (3,30)MANOM1,MANOM2
   30 FORMAT (//,2X,F8.4,25X,F8.4)
      READ (3,40)TUNVEL,MOTVOL
   40 FORMAT (//,4X,F7.2,31X,F6.2)
      READ (3,50)P90,P0
   50 FORMAT (//,5X,F7.4,23X,F7.4)
      READ (3,60)KOUNT,N
   60 FORMAT (///,3X,I6,26X,I6)
      READ (3,70)AVSTAT(1),AVSTAT(2),AVSTAT(3),AVSTAT(4)
      READ (3,75)AVSTAT(5),AVSTAT(6),AVSTAT(7),AVSTAT(8)
      READ (3,75)AVSTAT(9),AVSTAT(10),AVSTAT(11),AVSTAT(12)
      READ (3,75)AVSTAT(13),AVSTAT(14),AVSTAT(15),AVSTAT(16)
   70 FORMAT (////,F9.3,5X,F9.3,5X,F9.3,5X,F9.3)
```

```
  75   FORMAT (F9.3,5X,F9.3,5X,F9.3,5X,F9.3)
       VPD=(P90-P0)/90.0
C
C ---- The following part of the program first reads the predicted
C ---- seperation AOA , input from the terminal. The corresponding
C ---- raw digital seperation AOA is then computed and used as follows
C ---- LOW is computed as shown below, then used in the program
C ---- segement beginning with BEG=2, to determine the first AOA
C ---- data element that should be read from IDATA in RAWDATA1.DAT.
C ---- This AOA data element is designated START, and the AOA data
C ---- element 30 data sets later is designated END. With the first
C ---- and last AOA elements known, every AOA data element in between
C ---- read, and then stored in the IDATA1 array for matching later on
C
       WRITE (1,85)
  85   FORMAT (' ENTER PREDICTED SEPERATION AOA (DEGREES)',/)
       READ (1,95)AOASEP
  95   FORMAT (F5.2)
       RAWSEP=(((AOASEP*VPD)+P0)/10.0)*4096.0
       IF ((MOTVOL.GT.5.00).AND.(MOTVOL.LE.7.00))TAKE=90.0
       IF ((MOTVOL.GT.7.00).AND.(MOTVOL.LE.9.00))TAKE=140.0
       IF ((MOTVOL.GT.9.00).AND.(MOTVOL.LE.11.00))TAKE=200.0
       IF ((MOTVOL.GT.11.00).AND.(MOTVOL.LE.13.00))TAKE=250.0
       LOW=RAWSEP-TAKE
       CALL OPEN(4,'RAWDATA1DAT',2)
       READ(4)(IDATA(L),L=1,N)
C
       BEG=2
 400   CONTINUE
       IF(IDATA(BEG).LE.LOW)GO TO 420
       START=BEG
       GO TO 410
 420   BEG=BEG+18
       GO TO 400
 410   CONTINUE
       END=START+540
C
       ELEM1=0
       DO 500 PP=START,END,18
       ELEM1=ELEM1+1
       IDATA1(ELEM1)=IDATA(PP)
 500   CONTINUE
C
C ---- The DO 650 loop below begins by branching out to one of 4
C ---- program locations depending on the number of times the
C ---- computer has been through the 650 loop. The program segements
C ---- beginning at 510,525,535 and 545 all perform the same
C ---- operation, but do so on different data files, in turn.
C ---- The operation performed in all 4 segements is as follows.
C ---- The loop contained in each segement reads the raw digital
C ---- angle of attack from every data set in the array IDATA.
C ---- These data sets were previously stored in the second through
```

```
C ---- fifth data files, RAWDATA2DAT through RAWDATA5DAT by the data
C ---- gathering program BIG. After these angles of attack are read
C ---- from the IDATA array in the proper data file, they are stored
C ---- in the array IDATA2. Note that the contents of the IDATA2 array
C ---- are replaced each time the DO 650 loop is travelled.
C
      DO 650 RUN=2,5
      ELEM2=0
      IF (RUN.EQ.2) GO TO 510
      IF (RUN.EQ.3) GO TO 525
      IF (RUN.EQ.4) GO TO 535
      IF (RUN.EQ.5) GO TO 545
 510  CONTINUE
      CALL OPEN(5,'RAWDATA2DAT',2)
      READ(5)(IDATA(L),L=1,N)
      DO 520 QQ=2,N,18
      ELEM2=ELEM2+1
      IDATA2(ELEM2)=IDATA(QQ)
 520  CONTINUE
      GO TO 550
 525  CONTINUE
      CALL OPEN(6,'RAWDATA3DAT',2)
      READ(6)(IDATA(L),L=1,N)
      DO 530 QQ=2,N,18
      ELEM2=ELEM2+1
      IDATA2(ELEM2)=IDATA(QQ)
 530  CONTINUE
      GO TO 550
 535  CONTINUE
      CALL OPEN(7,'RAWDATA4DAT',2)
      READ(7)(IDATA(L),L=1,N)
      DO 540 QQ=2,N,18
      ELEM2=ELEM2+1
      IDATA2(ELEM2)=IDATA(QQ)
 540  CONTINUE
      GO TO 550
 545  CONTINUE
      CALL OPEN(8,'RAWDATA5DAT',2)
      READ(8)(IDATA(L),L=1,N)
      DO 550 QQ=2,N,18
      ELEM2=ELEM2+1
      IDATA2(ELEM2)=IDATA(QQ)
 550  CONTINUE
C
C ---- We now have an array, IDATA1, containing every AOA element
C ---- from within a specified interval in the array IDATA,
C ---- as read from disk file RAWDATA1.DAT.  We also have an
C ---- array, IDATA2, containing every angle of attack element
C ---- from one of the four remaining disk files. The program segement
C ---- below compares each element in IDATA1 with every element in
C ---- IDATA2, in turn. When the closest match is found, this infor-
C ---- mation is stored in the PASS array for use in the next pro-
```

```
C ---- gram segement.
C
      RR=0
      DO 560 SS=1,ELEM1
      CHANG2=4096
      TRAP=0
      DO 570 TT=1,ELEM2
      RR=RR+1
      CHANG1=IABS(IDATA1(SS)-IDATA2(TT))
      IF (CHANG1.GE.CHANG2) GO TO 580
      CHANG2=CHANG1
      GO TO 570
      580   TRAP=TRAP+1
      IF (TRAP.NE.1) GO TO 570
      PASS(SS)=RR-1
 570  CONTINUE
 560  CONTINUE
C
C ---- The next part, up to and including line 610, reads every data
C ---- set from IDATA, in disk file RAWDATA1.DAT, that corresponds to
C ---- an AOA element contained in array IDATA1. That is, every data
C ---- set within the previously defined interval is read from the
C ---- IDATA array in disk file RAWDATA1.DAT. These data sets are then
C ---- repacked sequentially in the array IDATAT.
C
      IF (RUN.NE.2) GO TO 630
      REWIND 4
      READ(4)(IDATA(L),L=1,N)
      CALL OPEN(9,'AVRGDATADAT',2)
      PAZZ=0
      FWD=START-1
      HLT=END-1
      DO 610 UU=FWD,HLT,36
      PAZZ=PAZZ+1
      DO 620 VV=1,36
      DD=(UU+VV)-1
      WW=VV+(36*(PAZZ-1))
      IDATAT(WW)=IDATA(DD)
 620  CONTINUE
 610  CONTINUE
      NN=PAZZ*36
      WRITE(9)(IDATAT(LL),LL=1,NN)
 630  CONTINUE
C
C ---- The following program segements beginning on lines 640,665,
C ---- 690, and 720 perform the same operation, but does so on
C ---- different disk files, in turn. The operation performed is as
C ---- follows. The information previously obtained and stored in
C ---- the array PASS is now used to determine which data sets will
C ---- be read (from whichever disk file of the remaining four that
C ---- is then being worked on), then added, element by element, to
C ---- the element totals in the array IDATAT. Note that the contents
```

```
C ---- of the array IDATA changes each time the computer passes [illegible]
C ---- this part of the program, but the contents of array IDATAT [illegible]
C ---- retained and added to each time the computer passes through [illegible]
C ---- part of the program.
C
        IF (RUN.EQ.2) GO TO 640
        IF (RUN.EQ.3) GO TO 665
        IF (RUN.EQ.4) GO TO 690
        IF (RUN.EQ.5) GO TO 720
        640   CONTINUE
        REWIND 9
        REWIND 5
        READ(5)(IDATA(L),L=1,N)
        READ(9)(IDATAT(LL),LL=1,NN)
        DO 655 XX=1,ELEM1
        DO 660 YY=1,36
        ZZ=(XX-1)*36
        DD=((PASS(XX)-1)*18)+YY
        FF=YY+ZZ
        IDATAT(FF)=IDATAT(FF)+IDATA(DD)
  660   CONTINUE
  655   CONTINUE
        WRITE(9)(IDATAT(LL),LL=1,NN)
        GO TO 649
  665   CONTINUE
        REWIND 9
        REWIND 6
        READ(6)(IDATA(L),L=1,N)
        READ(9)(IDATAT(LL),LL=1,NN)
        DO 670 XX=1,ELEM1
        DO 680 YY=1,36
        ZZ=(XX-1)*36
        DD=((PASS(XX)-1)*18)+YY
        FF=YY+ZZ
        IDATAT(FF)=IDATAT(FF)+IDATA(DD)
  680   CONTINUE
  670   CONTINUE
        WRITE(9)(IDATAT(LL),LL=1,NN)
        GO TO 649
  690   CONTINUE
        REWIND 9
        REWIND 7
        READ(7)(IDATA(L),L=1,N)
        READ(9)(IDATAT(L),LL=1,NN)
        DO 700 XX=1,ELEM1
        DO 710 YY=1,36
        ZZ=(XX-1)*36
        DD=((PASS(XX)-1)*18)+YY
        FF=YY+ZZ
        IDATAT(FF)=IDATAT(FF)+IDATA(DD)
  710   CONTINUE
  700   CONTINUE
```

```
      WRITE(9)(IDATAT(LL),LL=1,NN)
      GO TO 649
 720  CONTINUE
      REWIND 9
      REWIND 8
      READ(8)(IDATA(L),L=1,N)
      READ(9)(IDATAT(LL),LL=1,NN)
      DO 730 XX=1,ELEM1
      DO 740 YY=1,36
      ZZ=(XX-1)*36
      DD=((PASS(XX)-1)*18)+YY
      FF=YY+ZZ
      IDATAT(FF)=IDATAT(FF)+IDATA(DD)
 740  CONTINUE
 730  CONTINUE
      WRITE(9)(IDATAT(LL),LL=1,NN)
 649  CONTINUE
 650  CONTINUE
C
C ---- We now have array IDATAT, having as its individual elements
C ---- the total of five individual IDATA elements from each of
C ---- five elements. To get the average value of these, we must
C ---- divide the total we have by 5. This division is done in the
C ---- program segement below. This average value is then stored
C ---- in the array IDATAT for further processing.
C
      REWIND 9
      READ(9)(IDATAT(L),L=1,NN)
      DO 750 DIV=1,NN
      IDATAT(DIV)=(IDATAT(DIV))/5.0
 750  CONTINUE
      WRITE(9)(IDATAT(L),L=1,NN)
C
C ---- The steps below compute airfoil rotation rate (deg/sec),
C ---- non-dimensional rotation rate, Reynolds number, tunnel "Q"
C ---- and volts per degree for the run.
C
      REWIND 9
      READ(9)(IDATAT(L),L=1,NN)
      DTIM=(IDATAT(451)-IDATAT(91))*(0.0010046)
      DPOSV=((IDATAT(452)-IDATAT(92))/4096.0)*10.0
      DPOSD=DPOSV/VPD
      ROTRAT=DPOSD/DTIM
      NDRATE=(ROTRAT*.017453*1.016)/(2.0*TUNVEL)
      RHO=(BAROM*70.45)/(1716.0*(460+TEMP))
      MU=(2.270*(10.0**(-8.0))*((460.0+TEMP)**1.5))/(460.0+TEMP+198.6
      RE=(RHO*TUNVEL*1.016)/MU
      TUNQ=(0.5)*RHO*(TUNVEL**2)
C
C ---- The following writes pertinent information to disk file
C ---- REDUDATADAT as a heading.
C
```

```
      CALL OPEN(10,'REDUDATADA(',2
      WRITE (10,800)
800   FORMAT (' DAY',10X,'MONTH',9X,'YEAR',9X,'TIME'  )
      WRITE (10,810)DAY,MONTH,YEAR,HOUR
810   FORMAT (I3,11X,I3,11X,I3,9X,I5,/)
      WRITE (10,820)
820   FORMAT (' TEMPERATURE',14X,'BAROMETER' )
      WRITE (10,830)TEMP,BAROM
830   FORMAT (2X,F6.1,18X,F7.2,/)
      WRITE (10,840)
840   FORMAT (' MANOMETER 1',22X,'MANOMETER 2')
      WRITE (10,845)MANOM1,MANOM2
845   FORMAT (2X,F8.4,25X,F8.4,/)
      WRITE (10,850)
850   FORMAT (' TUNNEL VELOCITY',22X,'MOTOR VOLTAGE')
      WRITE (10,855)TUNVEL,MOTVOL
855   FORMAT (4X,F7.2,31X,F6.2,/)
      WRITE (10,860)
860   FORMAT (' DEGREES/SECOND',15X,'NON-DIMENSIONAL RATE')
      WRITE (10,870)ROTRAT,NDRATE
870   FORMAT(4X,F6.2,25X,F8.7,/)
      WRITE (10,880)
880   FORMAT (' REYNOLDS NUMBER',25X,'TUNNEL "Q"'   )
      WRITE (10,890)RE,TUNQ
890   FORMAT (4X,E11.4,30X,F6.3,/)
      DO 895 HH=1,16
      WRITE (10,897)HH,AVSTAT(HH)
897   FORMAT (' AVERAGE ZERO-INPUT READING, TRANSDUCER',I3,' =',F .0)
895   CONTINUE
C
C ---- One pass through the DO 100 J=1,NONO,18 loop computes one
C ---- point in the CN (normal force coefficient) versus ALPHA cur e.
C
      REWIND 9
      READ(9)(IDATAT(L),L=1,NN)
      NONO=NN-18
      DO 100 J=1,NONO,18
      REDAT(1)=IDATAT(J)
      REDAT(2)=IDATAT(J+1)
      REDAT(19)=IDATAT(J+18)
      REDAT(20)=IDATAT(J+19)
C
C ---- The loop below subtracts the average zero input readings
C ---- (AVSTAT) from each appropriate IDATAT element.
C
      DO 120 I=1,16
      AA=(J+I)+1
      REDAT(I+2)=IDATAT(AA)-((AVSTAT(I))-2048.0)
      REDAT(I+20)=IDATAT(AA+18)-((AVSTAT(I))-2048.0)
120   CONTINUE
C
C ---- Operations in the following loop correct for the finite
```

```
C ---- time between samples using a linear interpolation. Time
C ---- between passes must be sufficiently small or the linear
C ---- interpolation will be invalid.
C
      DO 140 R=1,18
      REDATC(R)=REDAT(R+18)-((REDAT(R+18)-REDAT(R))*((R-1)/18.0))
 140  CONTINUE
C
C ---- The following loop converts digital quantities to degrees
C ---- (angle of attack) and psi (sensed differential pressure).
C
C ---- The AOA conversion below assumes the A/D board is strapped
C ---- for the 0-10 volt unipolar input range. The amp on the
C ---- board is set for a gain of 1, so any input to the board
C ---- greater than 10 volts will saturate the A/D conversion system.
C
      AOA=(((REDATC(2)/4096.0)*10.0)-P0)/VPD
      DO 160 S=1,16
      BB=(J+S)+1
C
C ---- The PRESS conversion below assumes the A/D board is strapped
C ---- for the (-5)-(+5) volt bipolar input range, where the input
C ---- (from the transducers) is first amplified through an
C ---- amplifier of gain 100. So any input greater than +/-50 milli-
C ---- volts will saturate the A/D conversion system.
C
      PRESS(S)=((REDATC(S+2)-2048.0)/2048.0)*50.0/SENS(S)
      CP(S)=(PRESS(S)+(MANOM1/27.68))/(MANOM2/27.68)
 160  CONTINUE
C
C ---- The next loop defines the pressure distribution on the upper
C ---- surface of the airfoil, leading edge to trailing edge.
C ---- Pressure coefficient is assumed to be zero at the trailing edge.
C
      WRITE (10,170)REDAT(19)
 170  FORMAT (//,'TIME BACK TO WHICH DATA FOR THIS PASS HAS BEEN
     + CORRECTED:',F6.0,/)
      WRITE (10,185)
 185  FORMAT (' UPPER SURFACE PRESSURE COEFFICIENTS,
     + L.E. TO T.E., ARE GIVEN BELOW',/)
      DO 200 V=1,9
      CPU(V)=CP(V)
 200  CONTINUE
      CPU(10)=CPU(9)+(0.341*(CPU(9)-CPU(8)))
      DO 195 V=1,10
      WRITE (10,190)V,CPU(V)
 190  FORMAT (' CPU',I3,'=',F8.4)
 195  CONTINUE
C
C ---- The next loop defines the pressure distribution on the lower
C ---- surface of the airfoil, leading edge to trailing edge.
C ---- Pressure coefficient is assumed to be zero at the trailing edge.
```

```
C
      WRITE (10,205)
 205  FORMAT (//,' LOWER SURFACE PRESSURE COEFFICIENTS,
     + L.E. TO T.E., ARE GIVEN BELOW',/)
      CPL(1)=CP(1)
      DO 220 W=2,8
      DD=18-W
      CPL(W)=CP(DD)
 220  CONTINUE
      CPL(9)=CPU(10)
      DO 215 W=1,9
      WRITE (10,225)W,CPL(W)
 225  FORMAT (' CPL',I3,'=',F8.4)
 215  CONTINUE
C
C ---- The following loop integrates the upper pressure
C ---- distribution using the trapezoidal rule.
C
      AREAUT=0.0
      DO 240 X=1,9
      LNGTHU=PORTU(X+1)-PORTU(X)
      IF ((ABS(CPU(X+1)-CPU(X))).GT.(ABS((0.01)*CPU(X)))) GO TO 245
      AREAU=(0.5)*(CPU(X+1)+CPU(X))*LNGTHU
 245  IF ((ABS(CPU(X+1)-CPU(X))).LE.(ABS((0.01)*CPU(X)))) GO TO 235
      INTU=(PORTU(X)-PORTU(X+1))*CPU(X)/(CPU(X+1)-CPU(X))
      IF (INTU.LT.LNGTHU) GO TO 260
      AREAU=(.5)*(CPU(X+1)+CPU(X))*LNGTHU
      IF ((INTU).GE.(LNGTHU)) GO TO 235
 260  AREAU=((.5)*INTU*CPU(X))+
     +                 ((.5)*(LNGTHU-INTU)*CPU(X+1))
 235  AREAUT=AREAUT+AREAU
 240  CONTINUE
C
C ---- The following loop integrates the lower pressure
C ---- distribution using the trapezoidal rule.
C
      AREALT=0.0
      DO 280 Y=1,8
      LNGTHL=PORTL(Y+1)-PORTL(Y)
      IF ((ABS(CPL(Y+1)-CPL(Y))).GT.(ABS((0.01)*CPL(Y)))) GO TO  270
      AREAL=(.5)*(CPL(Y+1)+CPL(Y))*LNGTHL
      IF ((ABS(CPL(Y+1)-CPL(Y))).LE.(ABS((0.01)*CPL(Y)))) GO TO 275
 270  INTL=(PORTL(Y)-PORTL(Y+1))*CPL(Y)/(CPL(Y+1)-CPL(Y))
      IF ((INTL).LT.(LNGTHL)) GO TO 300
      AREAL=(.5)*(CPL(Y+1)+CPL(Y))*LNGTHL
      IF ((INTL).GE.(LNGTHL)) GO TO 275
 300  AREAL=((.5)*INTL*CPL(Y))+
     +                 ((.5)*(LNGTHL-INTL)*CPL(Y+1))
 275  AREALT=AREALT+AREAL
 280  CONTINUE
C
      NORMCO=AREALT-AREAUT
```

```
C
      WRITE (10,900)AOA
900   FORMAT (' ANGLE OF ATTACK=',F7.3,'  DEGREES')
      WRITE (10,910)NORMCO
910   FORMAT (' NORMAL FORCE COEFFICIENT=',F8.5,/)
      WRITE (10,950)
950   FORMAT (' *****************************************************
     +*****************',///)
100   CONTINUE
      STOP
      END
```

```
Page     1 -- TEST    .FOR

         PROGRAM TEST
C        FOR CALIBRATION OF THE AIM 12 IN BIPOLAR. 10 JUL 81
C        CHANNEL USED IS ALWAYS CHANNEL 1
C        LINK: TEST,ADIO,FORLIB/S,TEST/N/E
         REAL VALTOT
         IMPLICIT INTEGER (A-Z)
         WRITE (1,20)
 20      FORMAT (' ENTER CHANNEL YOU WISH TO SEE (0-15)',/)
         READ (1,30)CHAN
 30      FORMAT (I4)
 10      CONTINUE
         VALTOT=0.0
         DO 100 I=1,100
         CALL AD(VALUE,CHAN,80)
         VALTOT=VALTOT+(VALUE/100.0)
 100     CONTINUE
         WRITE (1,40)CHAN,VALTOT
 40      FORMAT ('+AVERAGE VALUE,CHANNEL',I4,'  = ',F8.2)
         GO TO 10
         STOP
         END
```

```
        .Z80
        ENTRY   AD
;       A/D SERVICE ROUTINE
;       FORTRAN CALLABLE
;       CALL AD(VALUE,CHAN,BASE)
;
;       GET ONE SAMPLE FROM THE CHAN'TH CHANNEL
;       ON THE A/D BOARD WITH BASE ADDRESS 'BASE'
;
AD:     LD      (VALUE),HL
        LD      (CHAN),DE
        LD      (BASE),BC
        EX      DE,HL   ;HL->CHAN
        LD      A,(HL)  ;GET CHAN NO.
        LD      HL,(BASE)
        LD      C,(HL)  ;GET BASE I/O ADDRESS TO C REG FOR OUTING
        OUT     (C),A   ;MODE 0 TO CHAN NO.
                        ;USES BASE ADDRESS IN C REG
        INC     C       ;POINT TO START CONVERSION PORT
        LD      A,0
        OUT     (C),A   ;START CONVERSION
        DEC     C       ;POINT TO BASE REGISTER
NRDY:   IN      A,(C)   ;GET STATUS
        AND     080H    ;BIT 7 IS STATUS, =1 IS BUSY
        JR      NZ,NRDY ;NOT ALL 0'S => BUSY
        INC     C       ;POINT TO BASE ADD+1
        INC     C       ;POINT TO DRL
        IN      A,(C)   ;LOW BYTE OF VALUE
        LD      E,A
        INC     C       ;POINT TO DRH
        IN      A,(C)   ;HIGH BYTE OF VALUE
        AND     0FH     ;MASK OUT HIGH NIBBLE
        LD      D,A     ;DE=VALUE
        LD      HL,(VALUE)      ;HL->WHERE TO PUT VALUE
        LD      (HL),E  ;PUT LOW BYTE OF VALUE
        INC     HL
        LD      (HL),D  ;THAT GIVES THE CALLER THE VALUE
;
        RET
;
VALUE:  DW      0       ;STORAGE FOR ADDRESS OF VALUE
CHAN:   DW      0       ;STORAGE FOR ADDRESS OF CHANNEL NO
BASE:   DW      0       ;STORAGE FOR ADDRESS OF BASE ADDRESS
;
        .Z80
        ENTRY DA
;       CALL DA(VAL,CHAN,BASE)
;       CHAN IS 0
;       BASE IS 72(BASE 10)
DA:     LD      A,(DE)          ;GET CHAN
        ADD     A,A             ;DOUBLE IT
        INC     A               ;ADD ONE
```

```
        PUSH    HL              ;SAVE VAL
        PUSH    BC              ;
        POP     HL              ;HL=:BASE
        LD      C,(HL)          ;C=LOW BYTE OF BASE
        ADD     A,C             ;
        LD      C,A             ;C=LOW BYTE VALUE OF PORT
        POP     HL              ;GET VAL
        LD      A,(HL)          ;GET LOW BYTE
        OUT     (C),A           ;PUT LOW BYTE
        DEC     C               ;C=HIGH BYTE PORT
        INC     HL              ;HL=>HI BYTE
        LD      A,(HL)          ;GET HI BYTE
        OUT     (C),A           ;PUT HI BYTE
        RET
        END
```

Page 1 -- STCLK .ONE

```
ENTRY STCLK
.Z80
STCLK:  LD A,017H   ; CHANNEL 1 CTRL WD=TIME/16
        OUT (079H).A
        LD A,09AH   ;TIME CONSTANT
        OUT (079H),A
        LD A,057H   ;CHANNEL 2 CNTRL WD=CTR
        OUT (07AH),A
        LD A,0FFH   ;TIME CONSTANT=256(BASE 10)
        OUT (07AH),A
        LD A,057H   ;CHANNEL 3 CNTRL WD=CTR
        OUT (07BH),A
        LD A,0FFH   ;TIME CONSTANT
        OUT (07BH),A
        JP 0        ;SYSTEM REENTRY POINT
        END STCLK
```

Page 1 -- GETTIM .MAC

```
ENTRY GETTIM
.Z80
GETTIM:
        PUSH    HL          ;SAVE DEST ADDRESS
        IN      A,(07AH)
        LD      E,A
        IN      A,(07BH)
        LD      D,A
        LD      HL,0FFFFH           ;MAX COUNT
        XOR     A           ;CLEAR CARRY
        SBC     HL,DE       ;SUBTR CURRENT COUNT FROM MAX COUNT
        EX      DE,HL       ;TIME TO DE
        POP     HL          ;GET ADDRESS
        LD      (HL),E
        INC     HL
        LD      (HL),D
        RET
        END
```

Appendix E

Remainder of Plotted Results

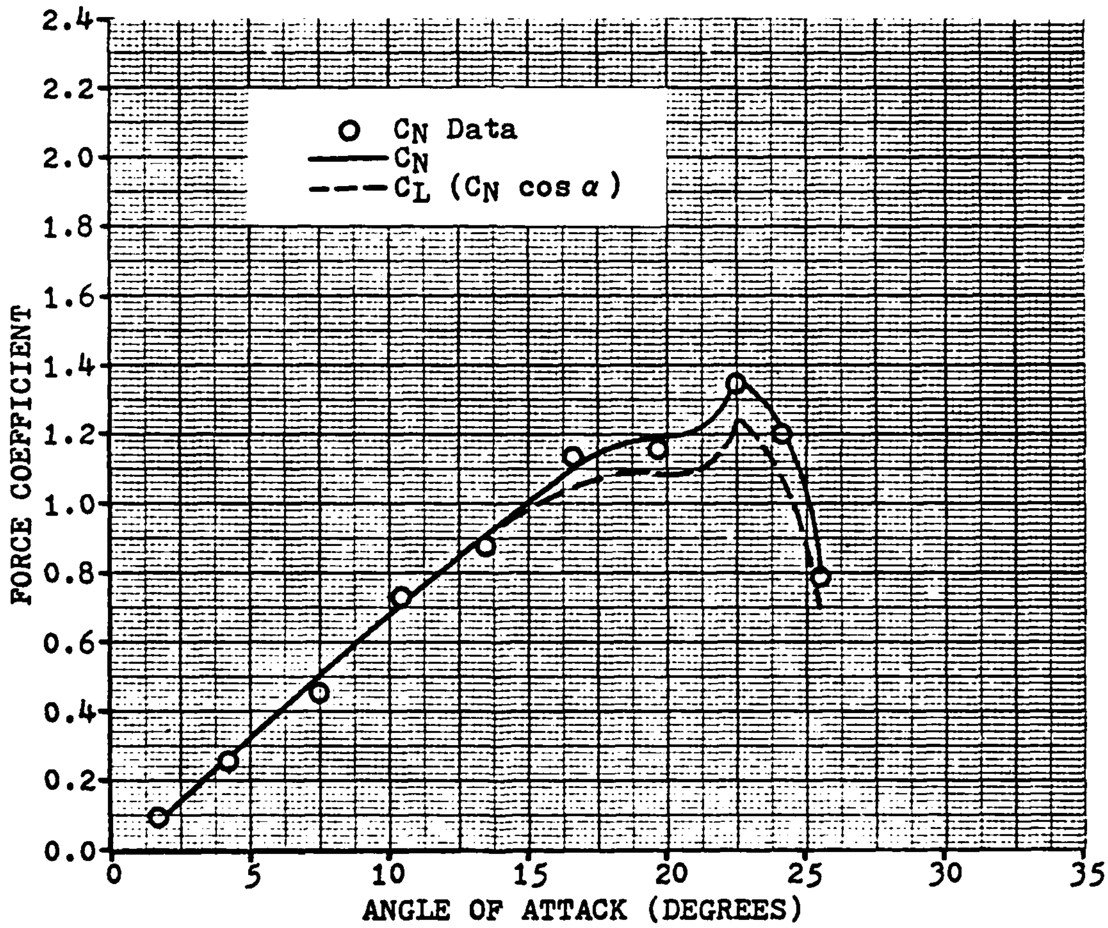


Figure 21. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=32.8 Deg/Sec and V_∞=26.7 Ft/Sec ($\dot{\alpha}_{ND}$=0.011)

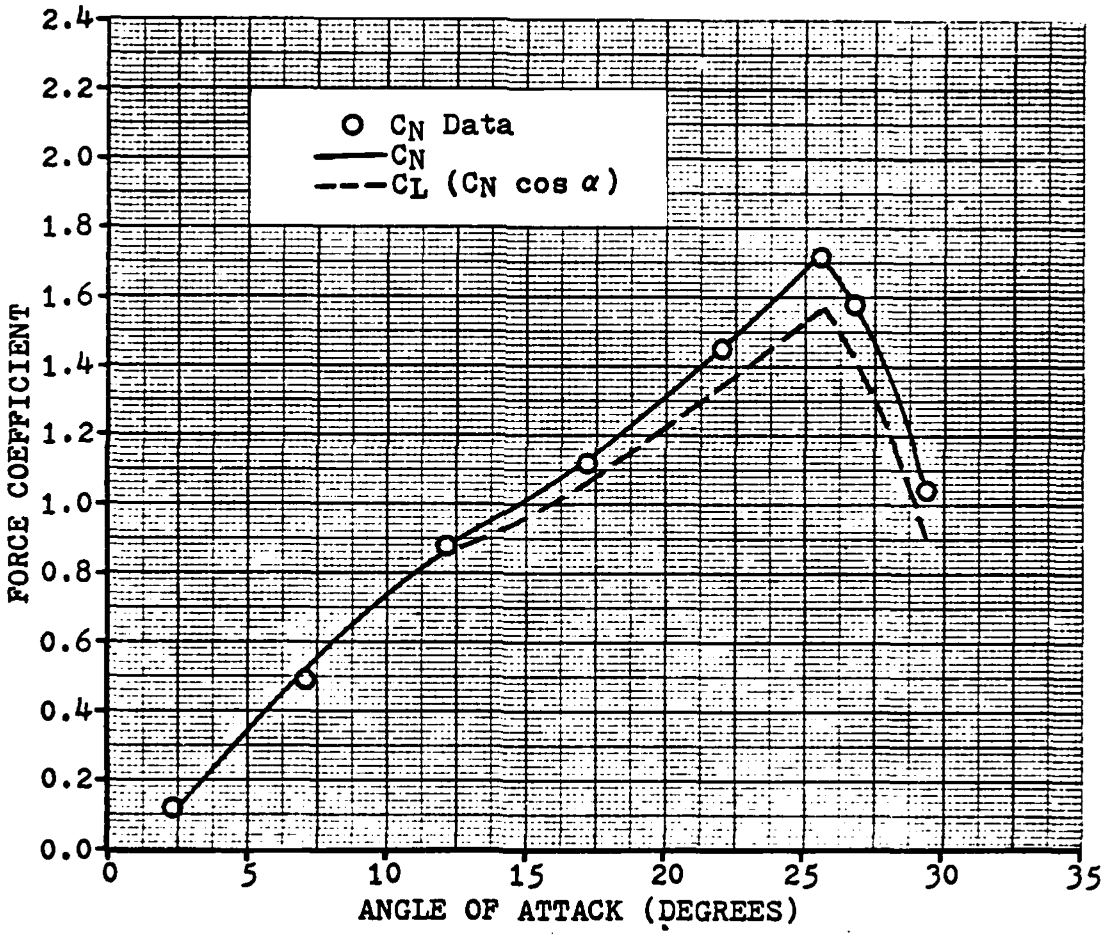


Figure 22. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=52.2 Deg/Sec and V_∞=26.7 Ft/Sec ($\dot{\alpha}_{ND}$=0.018)

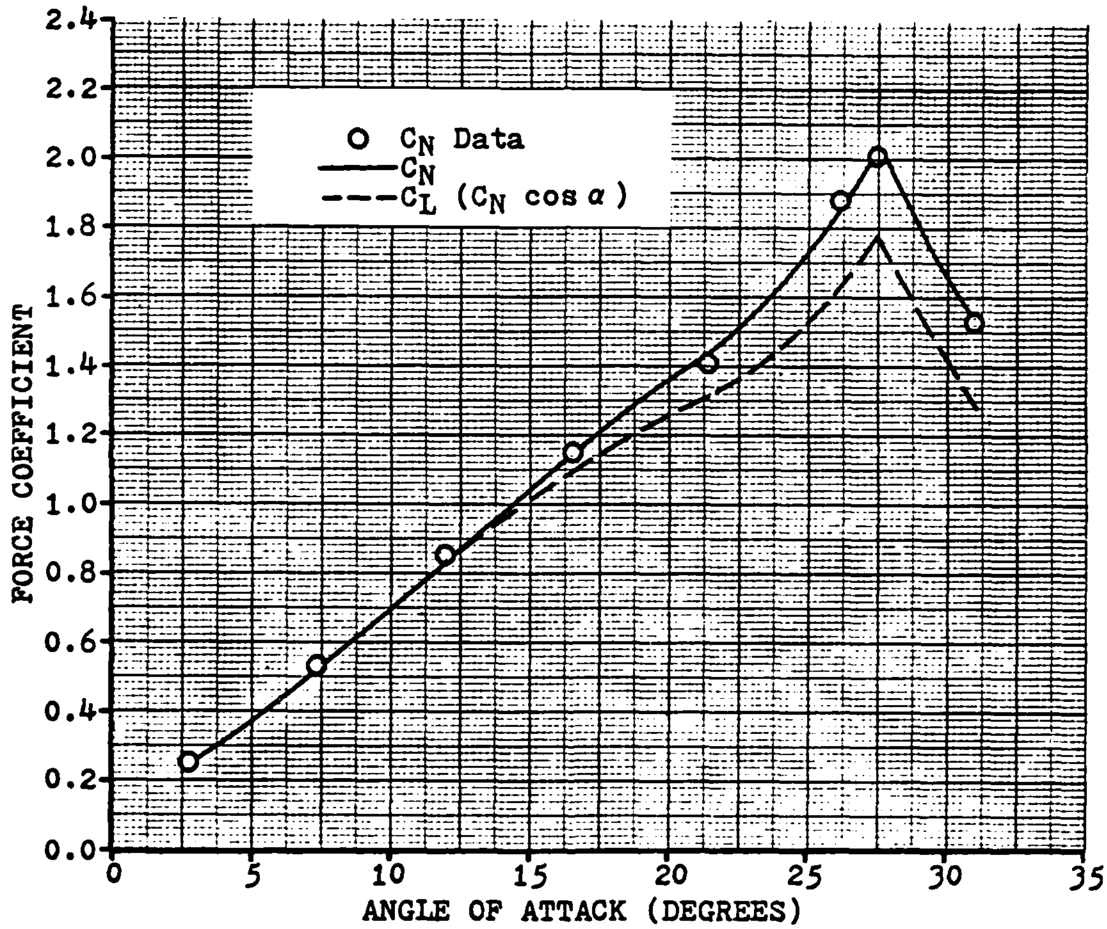


Figure 23. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=74.5 Deg/Sec and V_∞=26.7 Ft/Sec ($\dot{\alpha}_{ND}$=0.025)

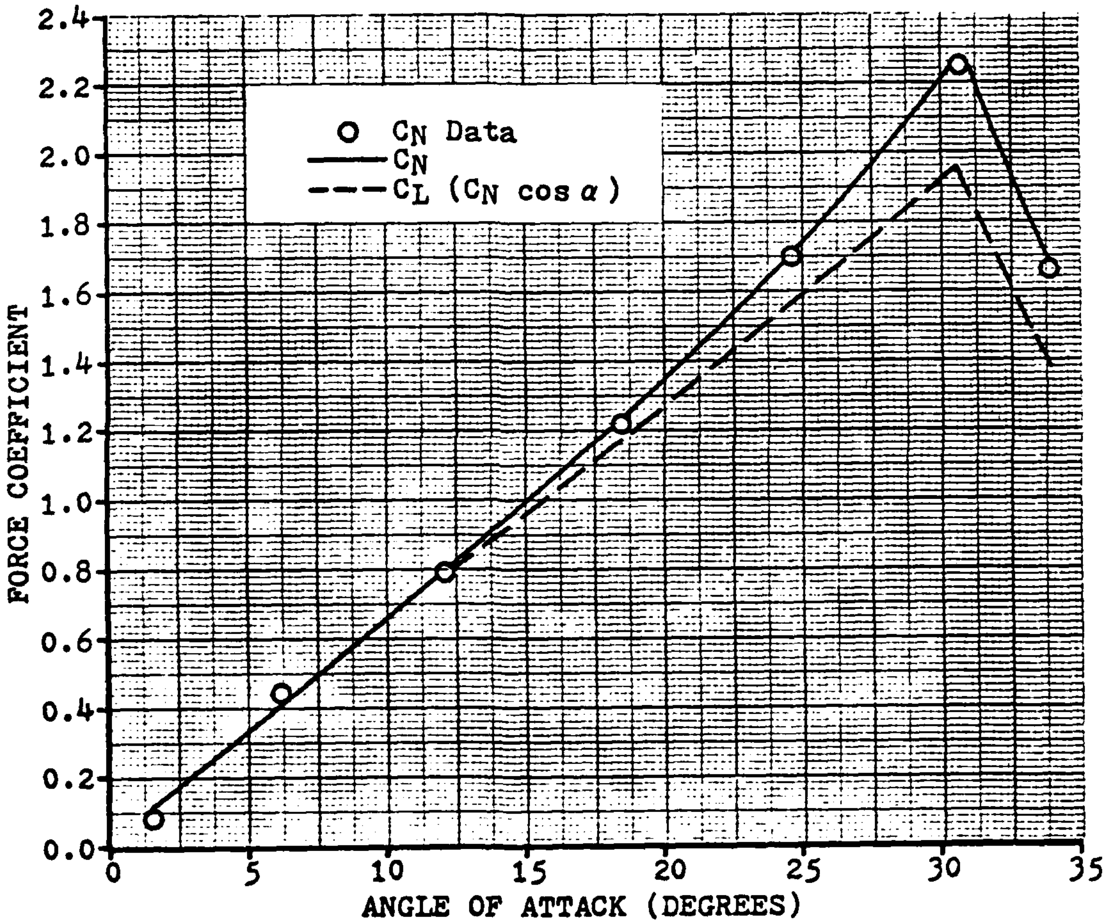


Figure 24. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=97.6 Deg/Sec and V_∞=26.7 Ft/Sec ($\dot{\alpha}_{ND}$=0.032)

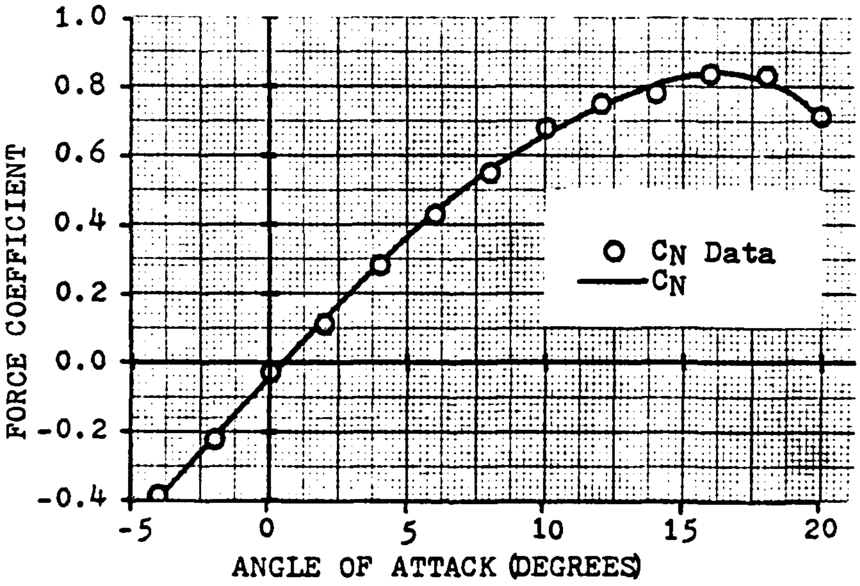


Figure 25. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=0.0 Deg/Sec and V_∞=26.7 Ft/Sec

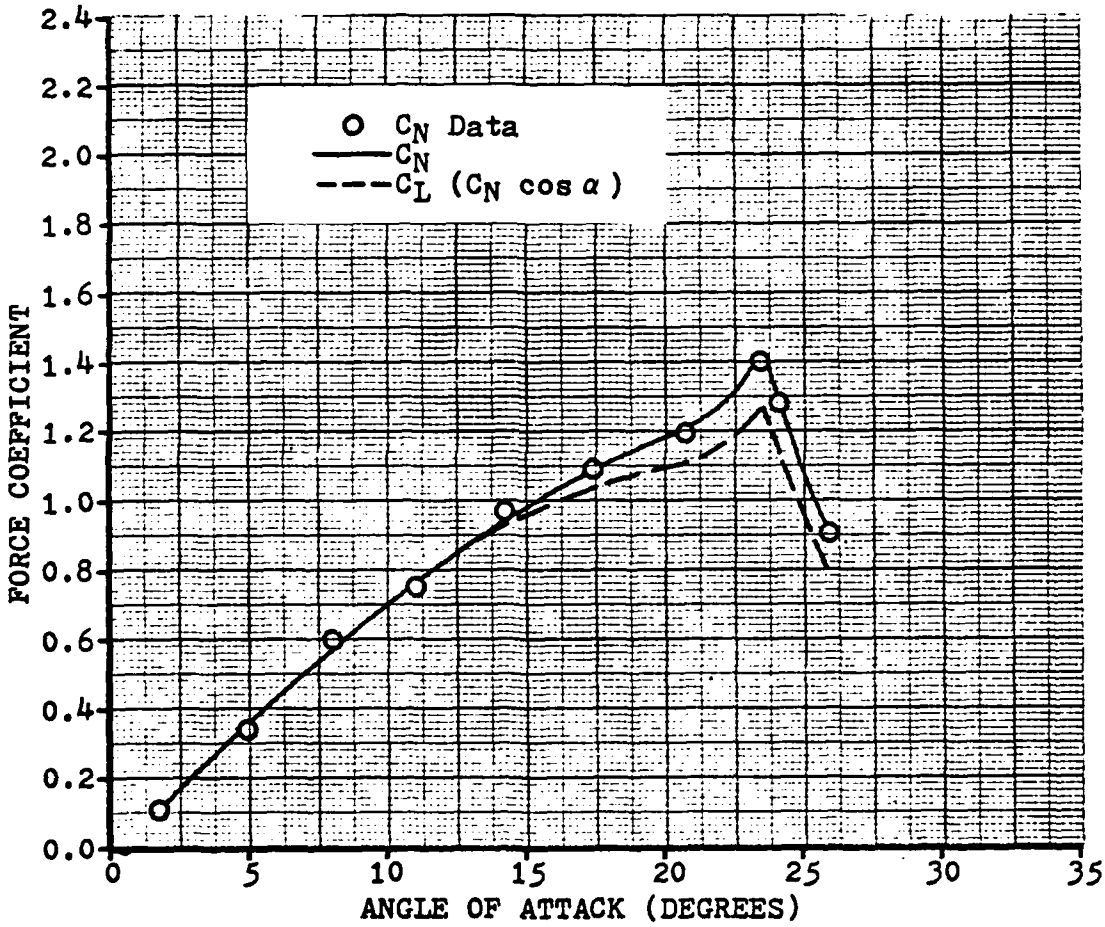


Figure 26. **Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=33.8 Deg/Sec and V_∞=30.1 Ft/Sec ($\dot{\alpha}_{ND}$=0.010)**

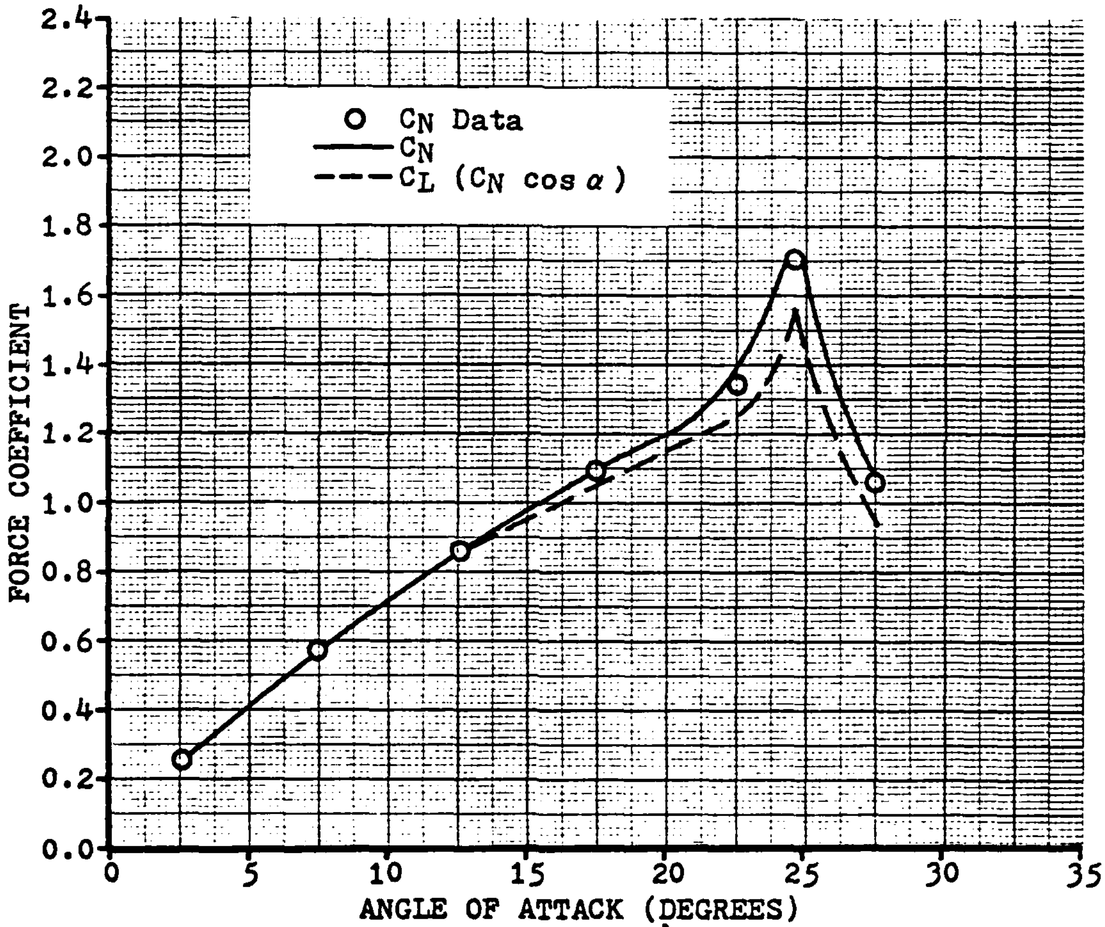


Figure 27. Force Coefficient vs. Angle of Attack for $\dot{\alpha} = 54.8$ Deg/Sec and $V_\infty = 30.1$ Ft/Sec ($\dot{\alpha}_{ND} = 0.016$)

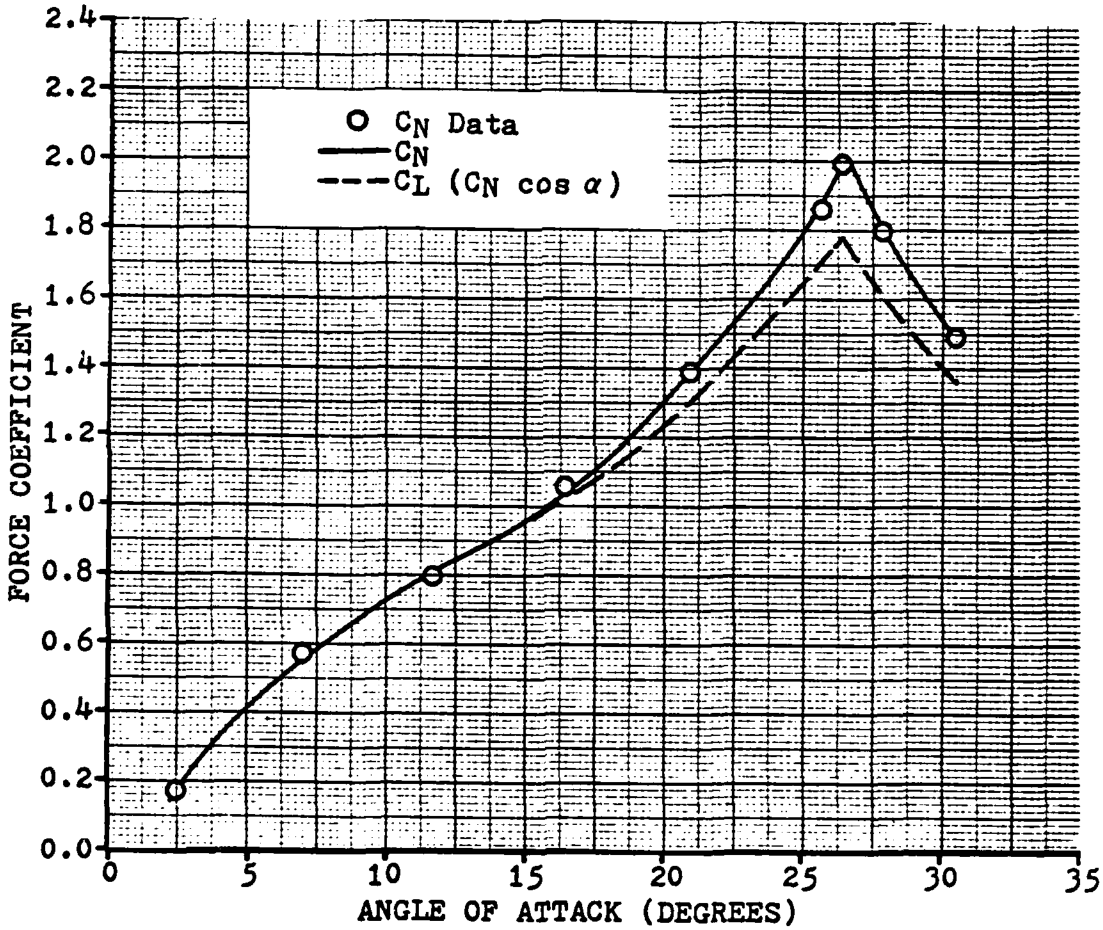


Figure 28. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=75.6 Deg/Sec and V_∞=30.1 Ft/Sec ($\dot{\alpha}_{ND}$=0.022)

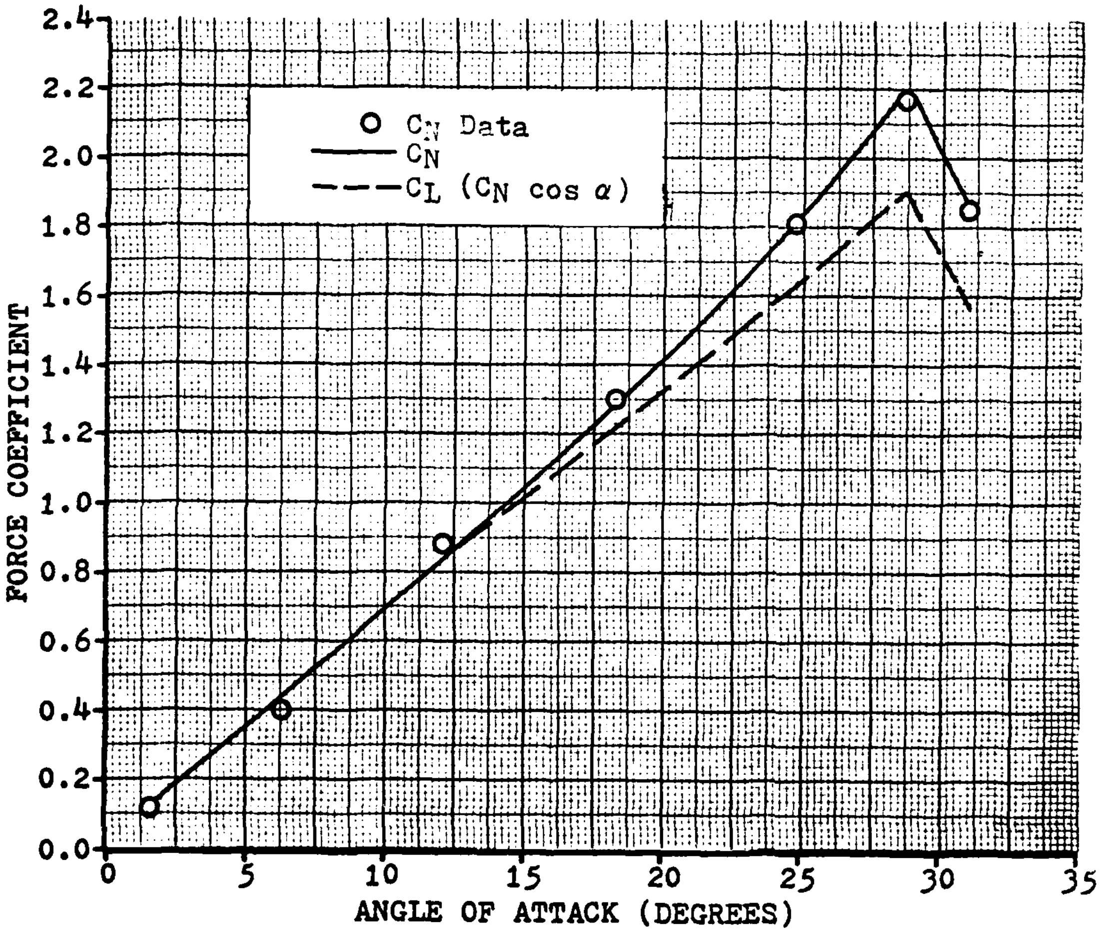


Figure 29 . Force Coefficient vs. Angle of Attack for $\dot{\alpha}$ =95.1 Deg/Sec and V_∞ =30.1 Ft/Sec ($\dot{\alpha}_{ND}$=0.029)

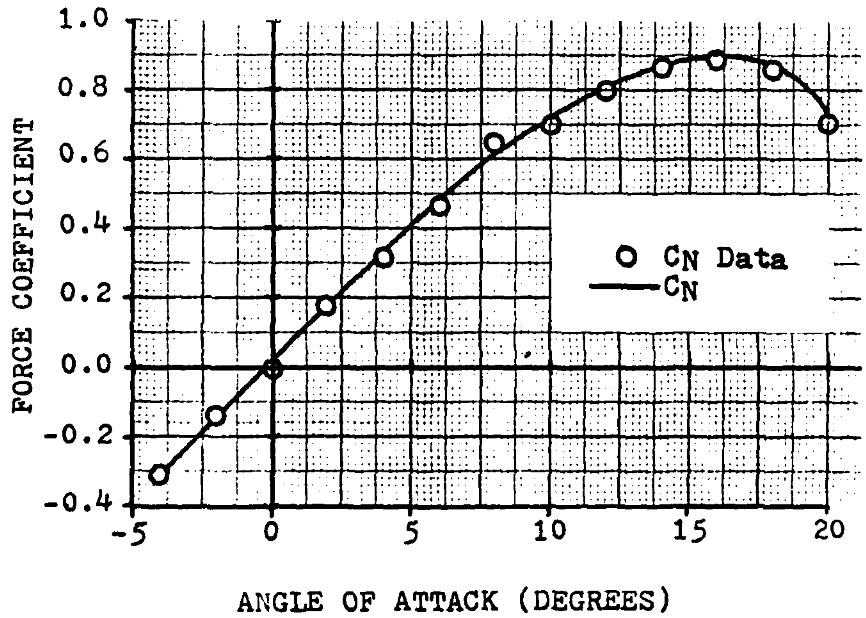


Figure 30. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=0.0 Deg/Sec and V =30.1 Ft/Sec

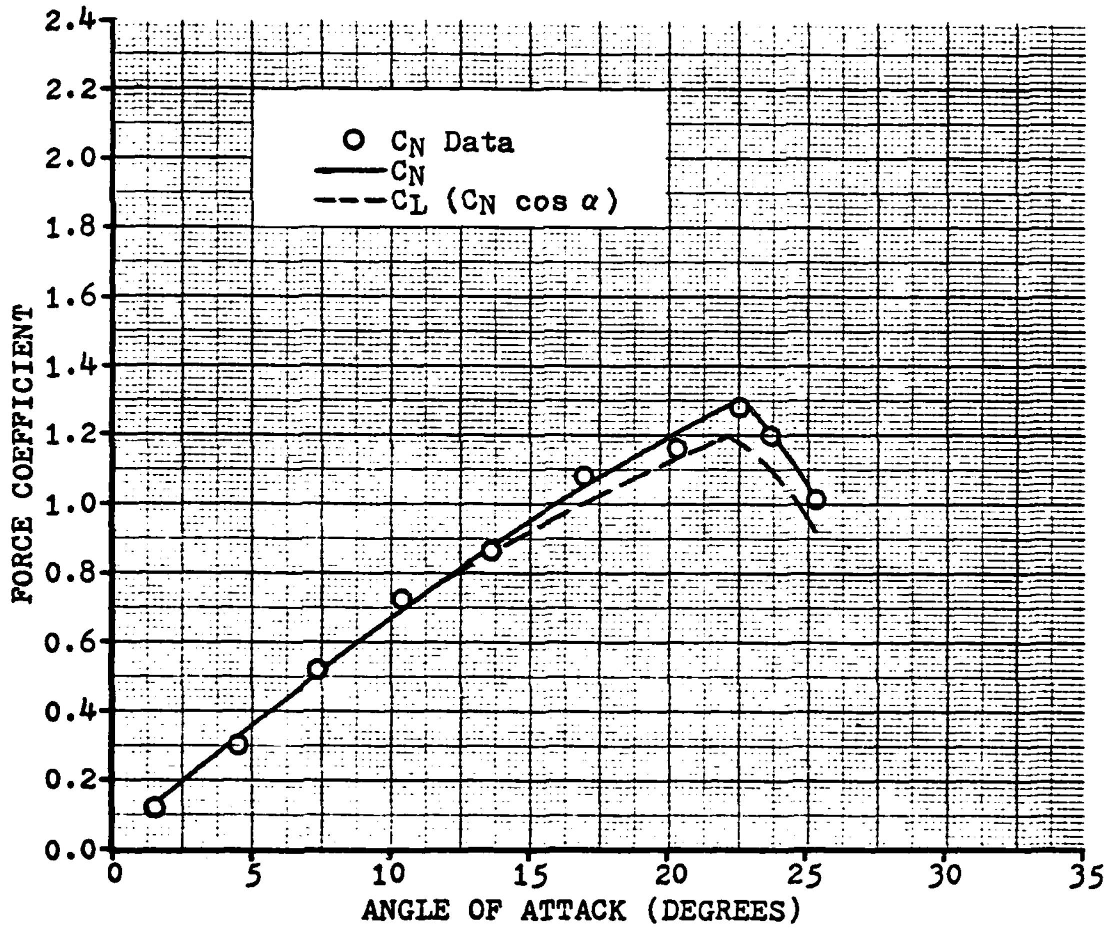


Figure 31. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=32.7 Deg/Sec and V_∞=39.9 Ft/Sec ($\dot{\alpha}_{ND}$=0.008)

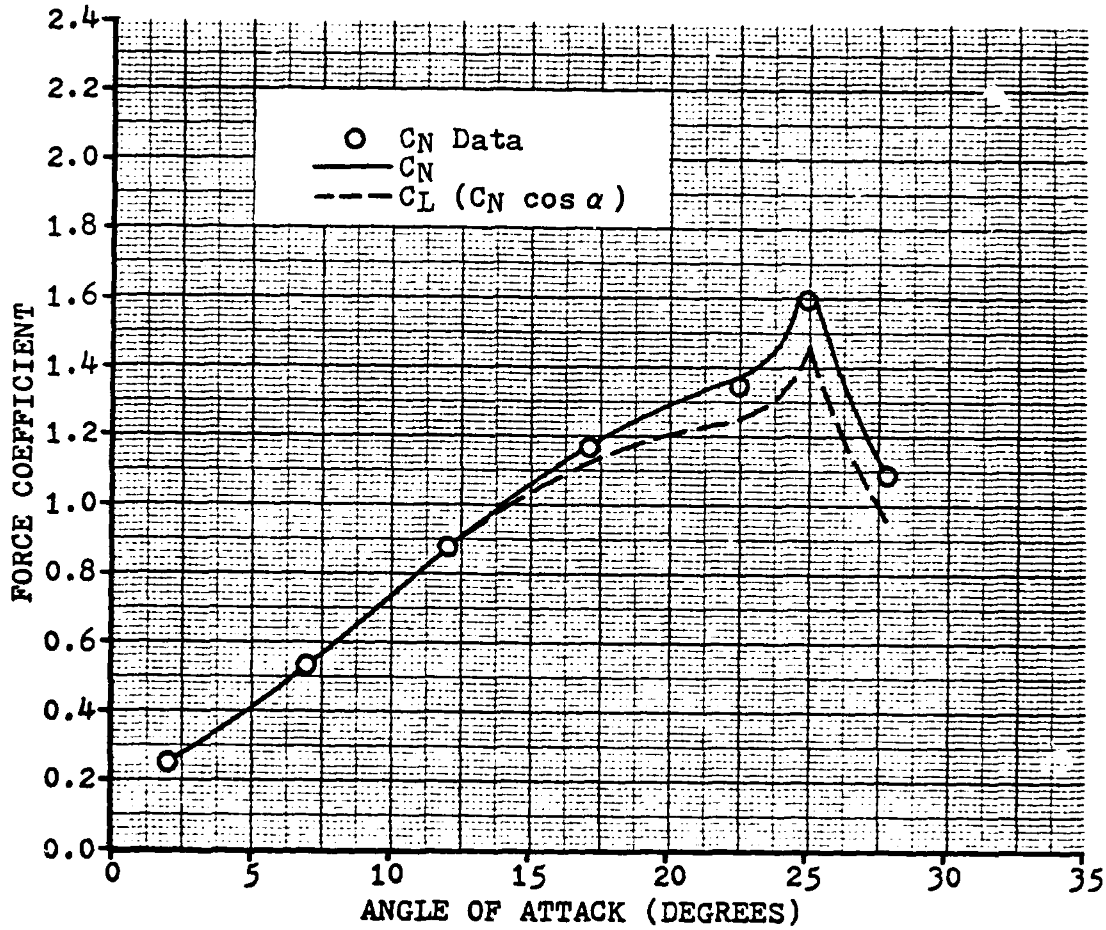


Figure 32. Force Coefficient vs. Angle of Attack for $\dot{\alpha}=55.8$ Deg/Sec and $V_\infty=39.9$ Ft/Sec ($\dot{\alpha}_{ND}=0.013$)

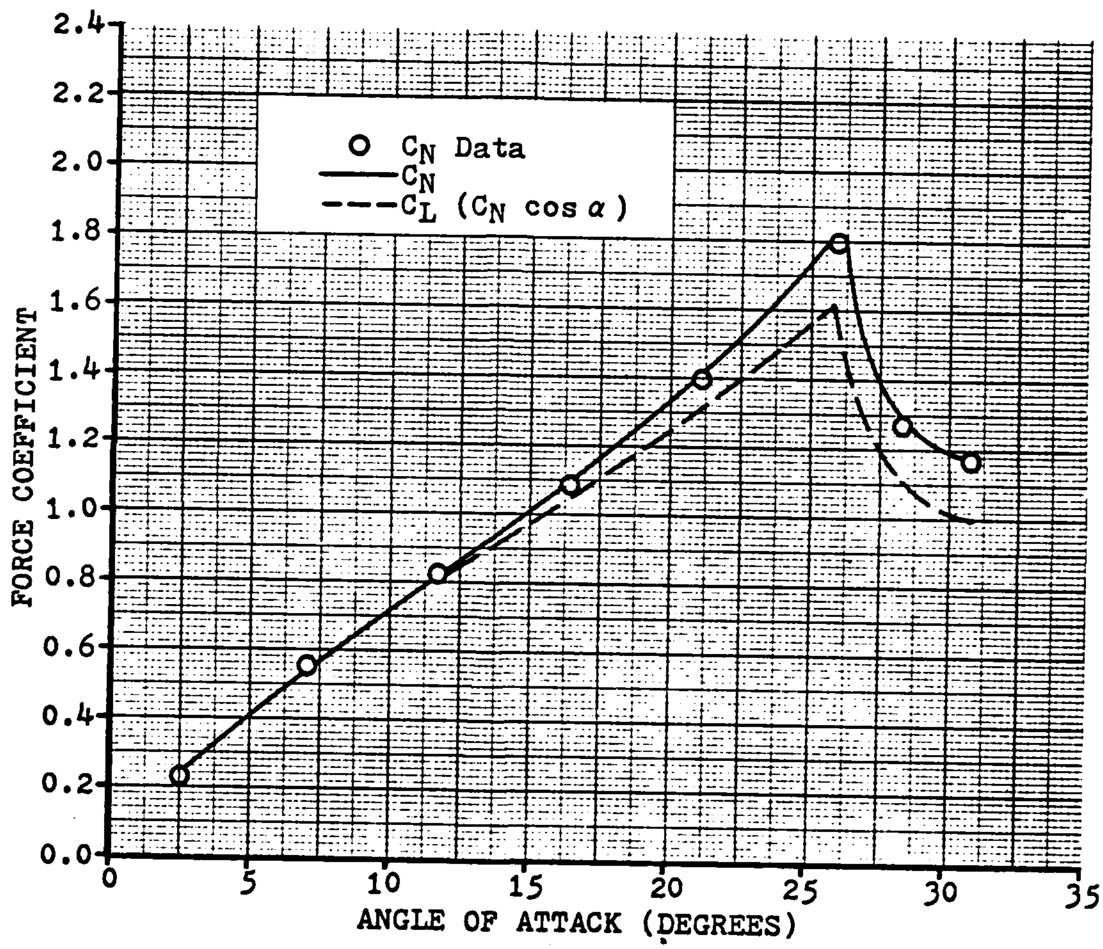


Figure 33. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=75.2 Deg/Sec and V_∞=39.9 Ft/Sec ($\dot{\alpha}_{ND}$=0.017)

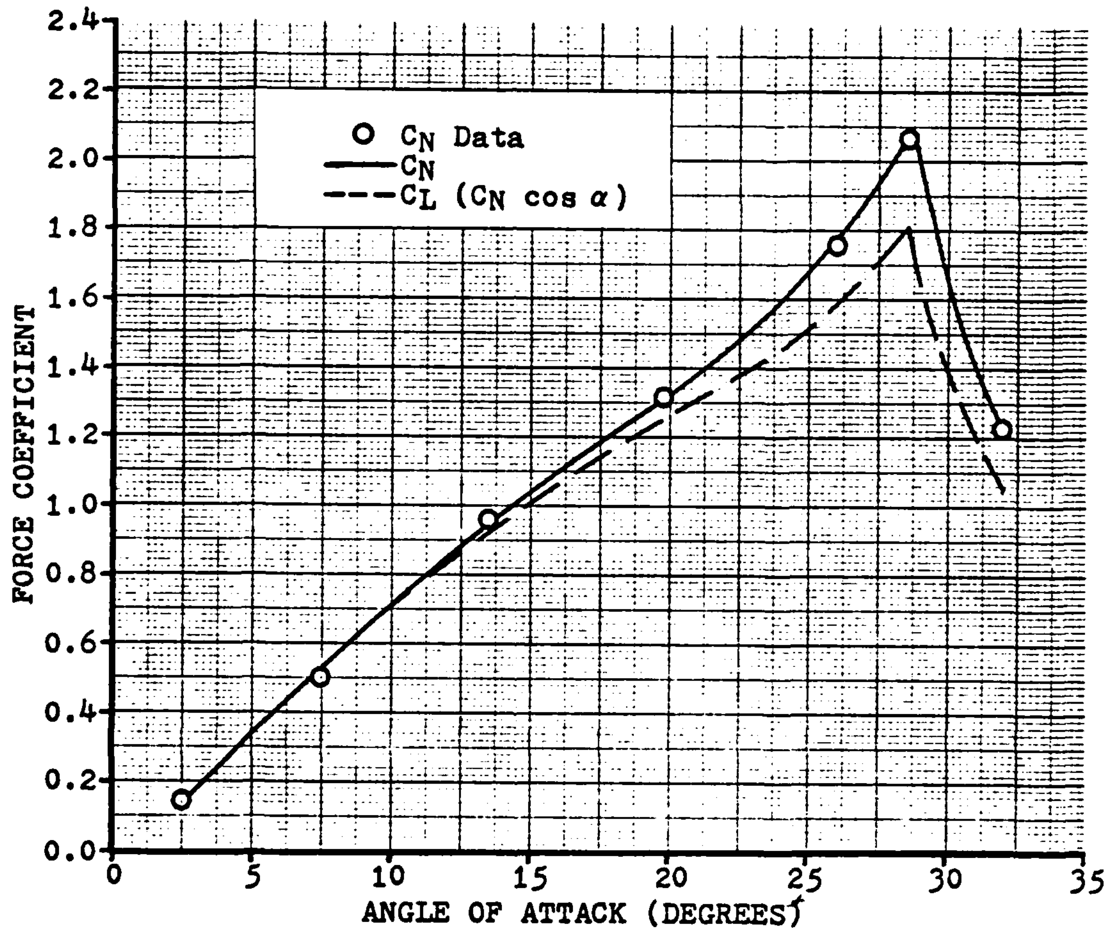


Figure 34. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=97.1 Deg/Sec and V_∞=39.9 Ft/Sec ($\dot{\alpha}_{ND}$=0.022)

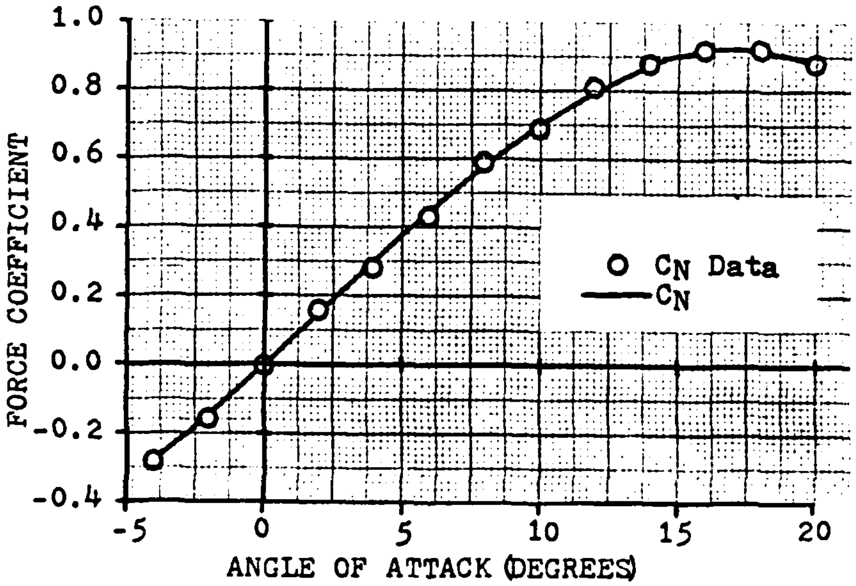


Figure 35. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=0.0 Deg/Sec and V_∞=39.9 Ft/Sec

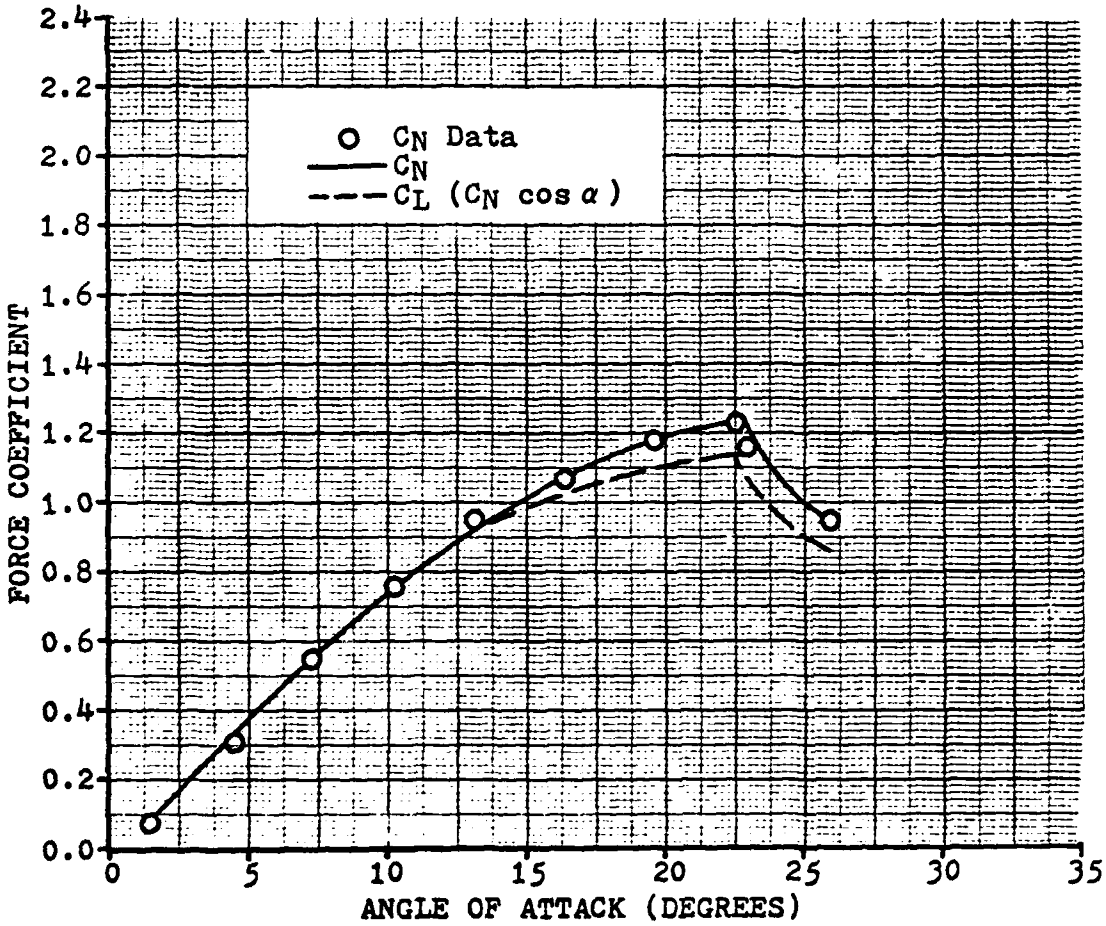


Figure 36. Force Coefficient vs. Angle of Attack for $\dot{\alpha}=32.7$ Deg/Sec and $V_\infty=47.8$ Ft/Sec ($\dot{\alpha}_{ND}=0.006$)

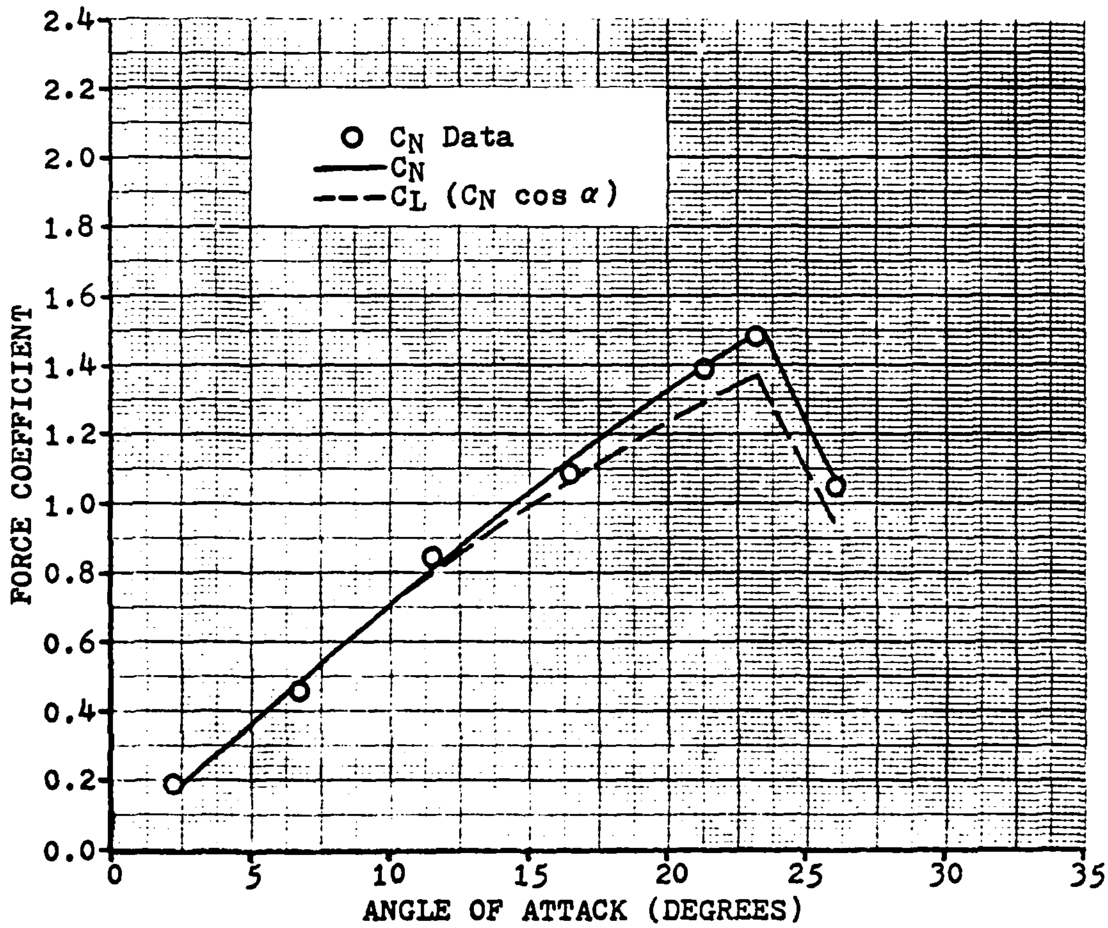


Figure 37. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=53.4 Deg/Sec and V_∞=47.8 Ft/Sec ($\dot{\alpha}_{ND}$=0.010)

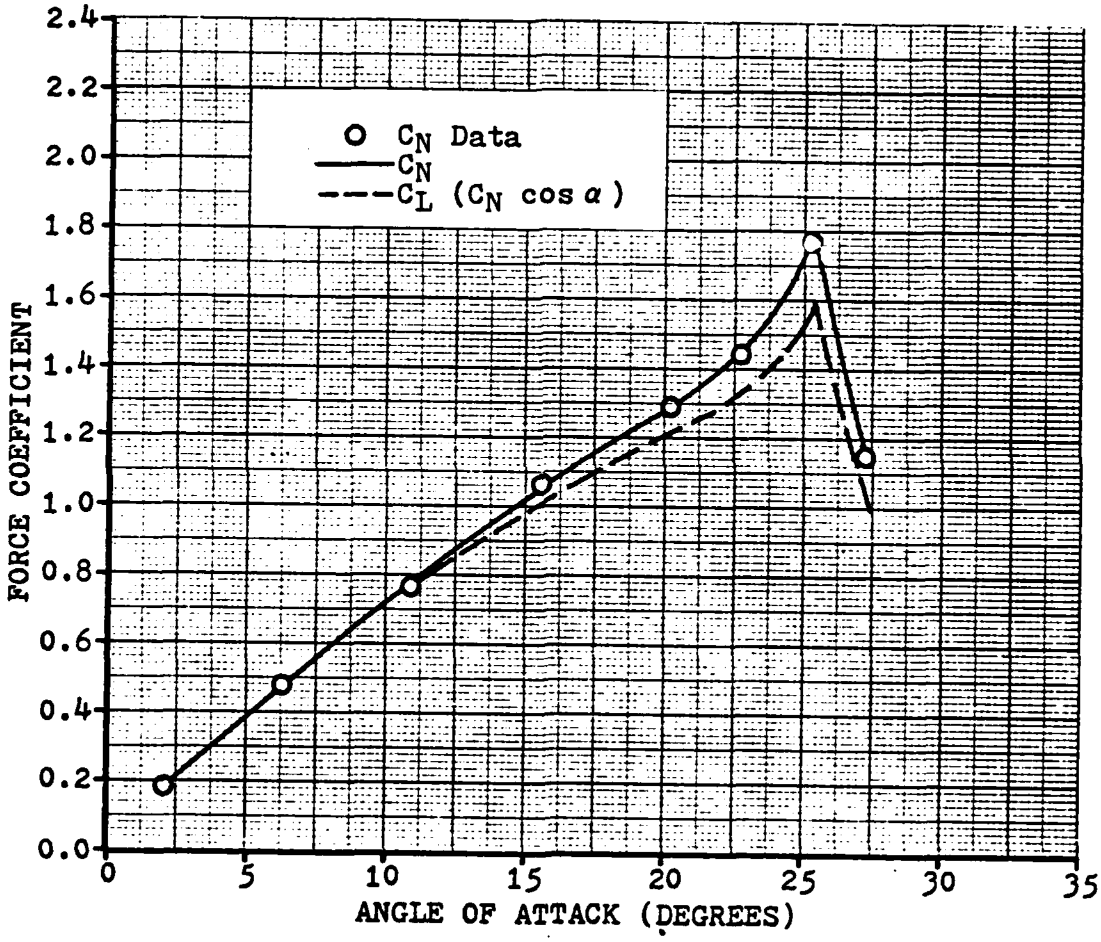


Figure 38. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=74.8 Deg/Sec and V_∞=47.8 Ft/Sec ($\dot{\alpha}_{ND}$=0.014)

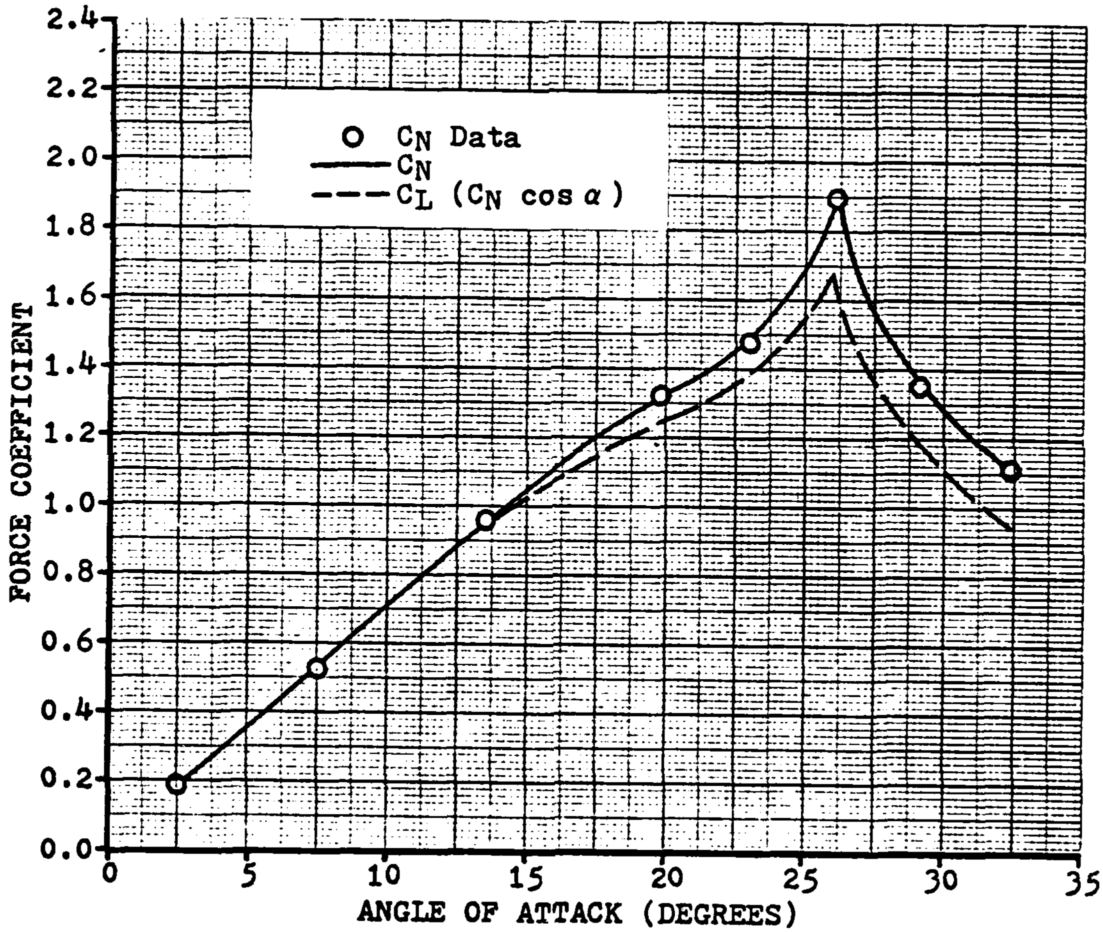


Figure 39. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$=97.1 Deg/Sec and V_∞=47.8 Ft/Sec ($\dot{\alpha}_{ND}$=0.018)

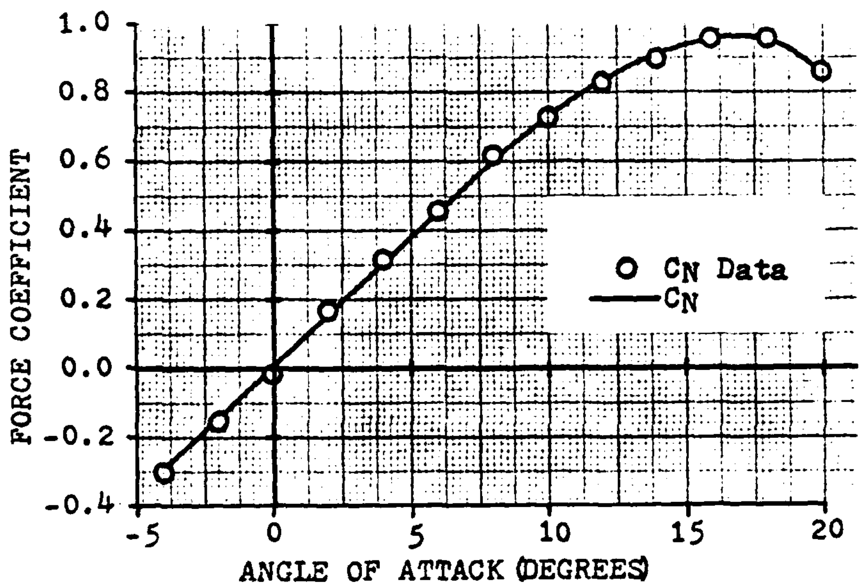


Figure 40. Force Coefficient vs. Angle of Attack for $\dot{\alpha}$ =0.0 Deg/Sec and V_∞=47.8 Ft/Sec

VITA

Scott J. Schreck was born on 11 March 1959 in Coon Rapids, Iowa. Upon graduation from Coon Rapids High School in 1977 he received an appointment to the United States Air Force Academy. He graduated from the Academy in 1981 with a Bachelor of Science Degree in Engineering Science, and remained there for a year as a graduate assistant football coach. He entered the School of Engineering, Air Force Institute of Technology, in June of 1982.

Permanent address: 623 Sixth Avenue
Coon Rapids, Iowa 50058

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

AD-A136 920

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GAE/AA/83D-21	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION School of Engineering 6b. OFFICE SYMBOL (If applicable) AFIT/ENG	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS (City, State and ZIP Code) AF Institute of Technology Wright-Patterson AFB, Ohio 45433	7b. ADDRESS (City, State and ZIP Code)
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State and ZIP Code)	10. SOURCE OF FUNDING NOS. PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT NO.
11. TITLE (Include Security Classification) See Box 19	

12. PERSONAL AUTHOR(S)
Scott J. Schreck, B.S., 1st Lt, USAF

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Yr., Mo., Day)	15. PAGE COUNT
MS Thesis	FROM ______ TO ______	1983 December	143

16. SUPPLEMENTARY NOTATION

Approved for public release: IAW AFR 190-17
Lynn Wolaver
LYNN E. W[illegible]
Dean for [illegible] Professional Development
Air Force [illegible] (ATC)
Wright-Patterson AFB OH [illegible]
13 JAN 1984

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB. GR.	Stall, Dynamic Stall, Increased Lift, Ramp Angle of Attack
20	04		

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Title: FURTHER EXPERIMENTAL INVESTIGATION OF DYNAMIC STALL

Thesis Advisor: Eric J. Jumper, Major, USAF

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS ☐	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL Eric J. Jumper, Major, USAF	22b. TELEPHONE NUMBER (Include Area Code) (513) 255-2998 22c. OFFICE SYMBOL AFIT/ENG

DD FORM 1473, 83 APR EDITION OF 1 JAN 73 IS OBSOLETE.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

The flow over an NACA 0015 airfoil undergoing a constant rate of change of angle of attack was experimentally studied over a range of tunnel speeds and rotation rates. Surface-pressure transducers coupled with a microcomputer-based data acquisition system were used to collect surface-pressure data at the rate of 4000 samples per second. Data reduction was also microcomputer-based. The data were reduced in two forms. First, C_ℓ versus α curves through stall were determined for each dynamic experimental configuration. This was accomplished by numerical integration of the pressure data at a number of angles through stall, each data point representing the average of five experiments at the same experimental conditions. These curves indicated a slight decrease in C_ℓ - α slope with increasing $\dot{\alpha}$. Secondly, the increase in stall angle of attack of the dynamic over the static case was also plotted against a nondimensional angular rate parameter (defined as the product of one-half the chord length and angular rotation rate, divided by the freestream velocity). This comparison gave rise to an apparently universal curve of nondimensional angular rotation rate versus increase in stall angle of attack. This curve was in agreement in some sense with previous experiments using stall indicators other than the actual stall. Data was collected in the range of nondimensional angular rates between .006 and .032.

END

DATE
FILMED
2-84

DTIC