Lecture Notes in Economics and Mathematical Systems

Managing Editors: M. Beckmann and H. P. Künzi

Control Theory

108

Frances Bauer · Paul Garabedian
David Korn · Antony Jameson

Supercritical Wing Sections II

Springer-Verlag
Berlin · Heidelberg · New York

Lecture Notes in Economics and Mathematical Systems

(Vol. 1–15: Lecture Notes in Operations Research and Mathematical Economics, Vol. 16–59: Lecture Notes in Operations Research and Mathematical Systems)

Vol. 1: H. Bühlmann, H. Loeffel, E. Nievergelt, Einführung in die Theorie und Praxis der Entscheidung bei Unsicherheit. 2. Auflage, IV, 125 Seiten. 1969. DM 16,–

Vol. 2: U. N. Bhat, A Study of the Queueing Systems M/G/1 and GI/M1. VIII, 78 pages. 1968. DM 16,–

Vol. 3: A. Strauss, An Introduction to Optimal Control Theory. VI, 153 pages. 1968. DM 16,–

Vol. 4: Branch and Bound: Eine Einführung. 2., geänderte Auflage. Herausgegeben von F. Weinberg. VII, 174 Seiten. 1972. DM 18,–

Vol. 5: Hyvärinen, Information Theory for Systems Engineers. VIII, 205 pages. 1968. DM 16,–

Vol. 6: H. P. Künzi, O. Müller, E. Nievergelt, Einführungskursus in die dynamische Programmierung. IV, 103 Seiten. 1968. DM 16,–

Vol. 7: W. Popp, Einführung in die Theorie der Lagerhaltung. VI, 173 Seiten. 1968. DM 16,–

Vol. 8: J. Teghem, J. Loris-Teghem, J. P. Lambotte, Modèles d'Attente M/G/1 et GI/M/1 à Arrivées et Services en Groupes. IV, 53 pages. 1969. DM 16,–

Vol. 9: E. Schultze, Einführung in die mathematischen Grundlagen der Informationstheorie. VI, 116 Seiten. 1969. DM 16,–

Vol. 10: D. Hochstädter, Stochastische Lagerhaltungsmodelle. VI, 269 Seiten. 1969. DM 18,–

Vol. 11/12: Mathematical Systems Theory and Economics. Edited by H. W. Kuhn and G. P. Szegö. VIII, IV, 486 pages. 1969. DM 34,–

Vol. 13: Heuristische Planungsmethoden. Herausgegeben von F. Weinberg und C. A. Zehnder. II, 93 Seiten. 1969. DM 16,–

Vol. 14: Computing Methods in Optimization Problems. Edited by A. V. Balakrishnan. V, 191 pages. 1969. DM 16,–

Vol. 15: Economic Models, Estimation and Risk Programming: Essays in Honor of Gerhard Tintner. Edited by K. A. Fox, G. V. L. Narasimham and J. K. Sengupta. VIII, 461 pages. 1969. DM 24,–

Vol. 16: H. P. Künzi und W. Oettli, Nichtlineare Optimierung: Neuere Verfahren, Bibliographie. IV, 180 Seiten. 1969. DM 16,–

Vol. 17: H. Bauer und K. Neumann, Berechnung optimaler Steuerungen, Maximumprinzip und dynamische Optimierung. VIII, 188 Seiten. 1969. DM 16,–

Vol. 18: M. Wolff, Optimale Instandhaltungspolitiken in einfachen Systemen. V, 143 Seiten. 1970. DM 16,–

Vol. 19: L. Hyvärinen Mathematical Modeling for Industrial Processes. VI, 122 pages. 1970. DM 16,–

Vol. 20: G. Uebe, Optimale Fahrpläne. IX, 161 Seiten. 1970. DM 16,–

Vol. 21: Th. Liebling, Graphentheorie in Planungs- und Tourenproblemen am Beispiel des städtischen Straßendienstes. IX, 118 Seiten. 1970. DM 16,–

Vol. 22: W. Eichhorn, Theorie der homogenen Produktionsfunktion. VIII, 119 Seiten. 1970. DM 16,–

Vol. 23: A. Ghosal, Some Aspects of Queueing and Storage Systems. IV, 93 pages. 1970. DM 16,–

Vol. 24: Feichtinger Lernprozesse in stochastischen Automaten. V, 66 Seiten. 1970. DM 16,–

Vol. 25: R. Henn und O. Opitz, Konsum- und Produktionstheorie. I. II, 124 Seiten. 1970. DM 16,–

Vol. 26: D. Hochstädter und G. Uebe, Ökonometrische Methoden. XII, 250 Seiten. 1970. DM 18,–

Vol. 27: I. H. Mufti, Computational Methods in Optimal Control Problems. IV, 45 pages. 1970. DM 16,–

Vol. 28: Theoretical Approaches to Non-Numerical Problem Solving. Edited by R. B. Banerji and M. D. Mesarovic. VI, 466 pages. 1970. DM 24,–

Vol. 29: S. E. Elmaghraby, Some Network Models in Management Science. III, 177 pages. 1970. DM 16,–

Vol. 30: H. Noltemeier, Sensitivitätsanalyse bei diskreten linearen Optimierungsproblemen. VI, 102 Seiten. 1970. DM 16,–

Vol. 31: M. Kühlmeyer, Die nichtzentrale t-Verteilung. II, 106 Seiten. 1970. DM 16,–

Vol. 32: F. Bartholomes und G. Hotz, Homomorphismen und Reduktionen linearer Sprachen. XII, 143 Seiten. 1970. DM 16,–

Vol. 33: K. Hinderer, Foundations of Non-stationary Dynamic Programming with Discrete Time Parameter. VI, 160 pages. 1970. DM 16,–

Vol. 34: H. Störmer, Semi-Markoff-Prozesse mit endlich vielen Zuständen. Theorie und Anwendungen. VII, 128 Seiten. 1970. DM 16,–

Vol. 35: F. Ferschl, Markovketten. VI, 168 Seiten. 1970. DM 16,–

Vol. 36: M. P. J. Magill, On a General Economic Theory of Motion. VI, 95 pages. 1970. DM 16,–

Vol. 37: H. Müller-Merbach, On Round-Off Errors in Linear Programming. VI, 48 pages. 1970. DM 16,–

Vol. 38: Statistische Methoden I. Herausgegeben von E. Walter. VIII, 338 Seiten. 1970. DM 22,–

Vol. 39: Statistische Methoden II. Herausgegeben von E. Walter. IV, 155 Seiten. 1970. DM 16,–

Vol. 40: H. Drygas, The Coordinate-Free Approach to Gauss-Markov Estimation. VIII, 113 pages. 1970. DM 16,–

Vol. 41: U. Ueing, Zwei Lösungsmethoden für nichtkonvexe Programmierungsprobleme. VI, 92 Seiten. 1971. DM 16,–

Vol. 42: A. V. Balakrishnan, Introduction to Optimization Theory in a Hilbert Space. IV, 153 pages. 1971. DM 16,–

Vol. 43: J. A. Morales, Bayesian Full Information Structural Analysis. VI, 154 pages. 1971. DM 16,–

Vol. 44: G. Feichtinger, Stochastische Modelle demographischer Prozesse. XIII, 404 Seiten. 1971. DM 28,–

Vol. 45: K. Wendler, Hauptaustauschschritte (Principal Pivoting). II, 64 Seiten. 1971. DM 16,–

Vol. 46: C. Boucher, Leçons sur la théorie des automates mathématiques. VIII, 193 pages. 1971. DM 18,–

Vol. 47: H. A. Nour Eldin, Optimierung linearer Regelsysteme mit quadratischer Zielfunktion. VIII, 163 Seiten. 1971. DM 16,–

Vol. 48: M. Constam, FORTRAN für Anfänger. 2. Auflage. VI, 148 Seiten. 1973. DM 16,–

Vol. 49: Ch. Schneeweiß, Regelungstechnische stochastische Optimierungsverfahren. XI, 254 Seiten. 1971. DM 22,–

Vol. 50: Unternehmensforschung Heute – Übersichtsvorträge der Züricher Tagung von SVOR und DGU, September 1970. Herausgegeben von M. Beckmann. VI, 133 Seiten. 1971. DM 16,–

Vol. 51: Digitale Simulation. Herausgegeben von K. Bauknecht und W. Nef. IV, 207 Seiten. 1971. DM 18,–

Vol. 52: Invariant Imbedding. Proceedings of the Summer Workshop on Invariant Imbedding Held at the University of Southern California, June–August 1970. Edited by R. E. Bellman and E. D. Denman. IV, 148 pages. 1971. DM 16,–

Vol. 53: J. Rosenmüller, Kooperative Spiele und Märkte. IV, 152 Seiten. 1971. DM 16,–

Vol. 54: C. C. von Weizsäcker, Steady State Capital Theory. III, 102 pages. 1971. DM 16,–

Vol. 55: P. A. V. B. Swamy, Statistical Inference in Random Coefficient Regression Models. VIII, 209 pages. 1971. DM 20,–

Vol. 56: Mohamed A. El-Hodiri, Constrained Extrema. Introduction to the Differentiable Case with Economic Applications. III, 130 pages. 1971. DM 16,–

Vol. 57: E. Freund, Zeitvariable Mehrgrößensysteme. VII, 160 Seiten. 1971. DM 18,–

Vol. 58: P. B. Hagelschuer, Theorie der linearen Dekomposition. VII, 191 Seiten. 1971. DM 18,–

continuation on page 298

Lecture Notes in Economics and Mathematical Systems

Control Theory

108

Frances Bauer · Paul Garabedian
David Korn · Antony Jameson

Supercritical Wing Sections II

A Handbook

Springer-Verlag
Berlin · Heidelberg · New York 1975

Managing Editors
Prof. Dr. M. Beckmann
Brown University
Providence, RI 02912/USA

Prof. Dr. H. P. Künzi
Universität Zürich
8090 Zürich/Schweiz

Authors
Dr. Frances Bauer · Prof. Paul Garabedian
Dr. David Korn · Prof. Antony Jameson
New York University
Courant Institute of Mathematical Sciences
251 Mercer Street
New York, N.Y. 10012/USA

Library of Congress Cataloging in Publication Data

Main entry under title:

Supercritical wing theory.

(Lecture notes in economics and mathematical systems ; 108 : Control theory)
Bibliography: p.
Includes index.
1. Aerodynamics, Supersonic--Computer programs. 2. Airplanes--Wings. 3. Boundary layer. I. Bauer, Frances. II. Series: Lecture notes in economics and mathematical systems ; 108. III. Series: Control theory (Berlin)
TL571.S82 629.134'32 74-34333

AMS Subject Classifications (1970): Primary: 76H05
Secondary: 65P05, 35M05

ISBN 3-540-07029-X Springer-Verlag Berlin · Heidelberg · New York
ISBN 0-387-07029-X Springer-Verlag New York · Heidelberg · Berlin

Offsetdruck: Julius Beltz, Hemsbach/Bergstr.

Preface

This handbook is a sequel to an earlier volume entitled "A Theory of Supercritical Wing Sections, with Computer Programs and Examples." Since the completion of the first volume, which we shall refer to as Volume I (cf. [1]), some effort has been made to improve our airfoil design program. A number of more desirable airfoils have been designed. In addition several of our wing sections have been tested in wind tunnels. We should like to make this material available here, since it is more convenient to use the design program in conjunction with data for a fairly broad range of examples. Moreover, we have developed new analysis programs that supersede our previous work.

Chapter I is devoted to a brief discussion of the mathematics involved in our additions and modifications. There is only a minimum emphasis on theory, since the representation of important physical phenomena such as boundary layer shock wave interaction and separation is partly empirical. It is our contention, however, that the computer programs provide a better simulation than might have been expected. Chapter II presents numerical results found by our new methods, as well as comparisons with experimental data. Chapter III contains a discussion of the use of the program together with Fortran listings.

We should like to acknowledge the support of this work by NASA under Grants NGR-33-016-167 and NGR-33-016-201 and by the AEC under Contract AT(11-1)-3077 with New York University. Many of the experimental results presented in Section 3 of Chapter II were made available to us by J. Kacprzynski of the National Aeronautical Establishment in Ottawa. Some of the test data shown are British Crown Copyright, and are reproduced by permission of the Controller, R & D Establishments and Research, Ministry of Defence (PE). The final example was prepared by John Dahlin from data obtained by the

McDonnell Douglas Corporation at the National Aeronautical Establishment in Ottawa. Figure 6 in Section 6 of Chapter II was given to us by Bill Evans of the Grumman Aerospace Corporation and is based on an airfoil designed by Don MacKenzie using our method. We are indebted to Ray Hicks, R. T. Jones, Jerry South and Richard Whitcomb of NASA for much encouragement and helpful advice. Dan Goodman and Steve Korn have assisted us in the preparation of technical data, and Connie Engle and Farntella Graham have typed the manuscript.

New York, N. Y.
November 1974

Work supported by NASA under Grants NGR-33-016-167 and NGR-33-016-201. Computations performed at the AEC Computing and Applied Mathematics Center, New York University, under Contract AT(11-1)-3077. Reproduction in whole or in part is permitted for any purpose of the United States Government.

Contents

I. THEORY

1. Introduction

In Volume I (cf. [1]) we have presented a mathematical theory for the design and analysis of supercritical wing sections, and we have included examples and computer programs showing how our methods work. By now several of the first shockless airfoils we designed have been tested with some success, and satisfactory agreement of the results of our analysis with experimental data has been established. General acceptance of supercritical wing technology by the aircraft industry encourages us to make available in this second volume an improved series of transonic airfoils as well as extensions of our analysis program that include three dimensional and boundary layer effects. We hope that the data we have compiled will be helpful in such projects as the development of a transonic transport with an oblique supercritical wing, which could operate economically at nearly sonic speeds.

The purpose of this book is to put our work on transonics in a more definitive form. For design we introduce a better model of the trailing edge which should eliminate a loss of fifteen or twenty percent in lift experienced with previous heavily aft loaded models, which we attribute to boundary layer separation. We also indicate how drag creep can be reduced at off-design conditions. A rotated finite difference scheme is presented that enables us to apply Murman's method of analysis (cf. [13]) in more or less arbitrary curvilinear coordinate systems (cf. [5]). This allows us to handle supersonic as well as subsonic free stream Mach numbers and to capture shock waves as far back on an airfoil as we please. Moreover, it leads to an effective three dimensional program for the computation of transonic flow past an oblique wing. In the case of two dimensional flow we extend the method to take into account the displacement thickness computed by a semi-empirical turbulent boundary layer correction. Extensive comparisons are made with experimental data that have become available to us in our design work. Excellent agreement is obtained even in situations where the theory is not on an entirely firm footing, for example when the shock waves are not defined sharply. Our contention is that the programs furnish a physically adequate computer simulation of the compressible flows that arise in practical problems of transonic aerodynamics.

In Chapter I we describe new theoretical contributions under the assumption that the reader has some familiarity with Volume I. In Chapter II we present a series of our latest supercritical wing sections together with a collection of comparisons between theoretical and experimental analysis data. Chapter III is devoted to listings of new computer programs as well as a brief manual for their operation and an update of the design program listed in Volume I. The emphasis of this handbook is more on the numerical data we have compiled than on the explanation of the relevant mathematics.

2. Models of Shock Structure

For the mathematical analysis of transonic flow past bodies in space of two or three dimensions it is interesting to consider models of shock structure based on an ordinary differential equation for a potential function ϕ depending on just one variable x. In this connection we ask for a solution of the equation

$$(\phi_x^2)_x = 0 ,$$

suggested by the transonic small disturbance equation (cf. [13]), that satisfies three boundary conditions of the form

$$\phi(a) = A\ , \quad \phi'(a) = C > 0\ , \quad \phi(b) = B$$

at the ends of some interval [a,b]. If we allow for a shock wave across which ϕ and ϕ_x^2 are conserved, but ϕ_x decreases, there exists a unique solution of this problem for values of the prescribed constants A, B and C in the range

$$|B - A| \leq C(b - a)\ .$$

The answer consists of two straight lines with the slopes C and -C which meet at the uniquely determined shock point

$$x_0 = \frac{a+b}{2} + \frac{B-A}{2C}\ .$$

(See Figure 1a.) The problem has an analogy with transonic aerodynamics if we think of the interval of positive ϕ_x as representing supersonic flow and the interval of negative ϕ_x as representing subsonic flow.

Our problem in ordinary differential equations can be used to test the validity of finite difference schemes for the numerical analysis of transonic flow. We shall exploit such a procedure to discuss the method of Murman and Cole [13]. Let equally spaced mesh points be laid down on the interval [a,b] and denote by ϕ_j the values of the potential ϕ at these points. We call the jth point subsonic when $\phi_{j+1} < \phi_{j-1}$ and supersonic when $\phi_{j+1} > \phi_{j-1}$. According to one version of the scheme of Murman and Cole our differential equation, which can be expressed in the quasilinear form

$$\phi_x\ \phi_{xx} = 0\ ,$$

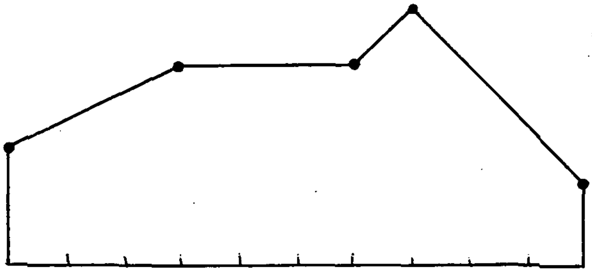
is approximated by the second order accurate centered relation

$$(\phi_{j+1}-\phi_{j-1})(\phi_{j+1}-2\phi_j+\phi_{j-1}) = 0$$

at subsonic points, but by the first order accurate retarded relation

(a) Exact solution.

(b) Forward shock solution.



(c) Smeared shock solution.

Figure 1. Solutions for one dimensional model.

$$(\phi_{j+1} - \phi_{j-1})(\phi_j - 2\phi_{j-1} + \phi_{j-2}) = 0$$

at supersonic points. The two relations are equivalent at the so-called shock points where $\phi_{j+1} = \phi_{j-1}$.

One can attempt to find a solution of our boundary value problem for the Murman-Cole difference equations iteratively by marching repeatedly from left to right solving successively at each mesh point for the unknown ϕ_j . Such an iterative scheme can be seen to converge monotonically from above when an initial guess of ϕ_j is made that is big enough and is concave. However, the answer is not unique because the shock condition has been lost through failure to use the conservation form of the equations. Any two straight lines satisfying our three boundary conditions and meeting at a mesh point define an admissible solution if the shock inequalities

$$\phi_j - \phi_{j-2} > \phi_j - \phi_{j+2} > 0$$

hold at that mesh point. (See Figure 1b.) Moreover, there are valid solutions containing a segment of shock points on which ϕ_j remains constant. These smeared shock waves terminate with one higher value and then a downturn leading to a supersonic point and a shock point followed by subsonic points. (See Figure 1c.) They need not fulfill any shock relations whatever, and they seem to occur in the applications.

One way to remedy the situation we have just described would be to replace the scheme of Murman and Cole by a finite difference analogue of the ordinary differential equation

$$(\phi_x^2)_x = h\,\phi_{xxx} ,$$

which is in conservation form and has been provided with an artificial viscosity term on the right. The small positive factor h should be of the same order of magnitude as the mesh size. The general solution of this equation is

$$\phi = - h \log \cosh\left(\frac{x-x_0}{y_0}\right) + \phi_0 ,$$

where x_0, y_0 and ϕ_0 are constants of integration that can be chosen to satisfy our three boundary conditions. As $h \to 0$ the solution approaches the two straight lines determined by the original shock structure problem. However, the truncation error of the artificial viscosity method tends to be larger than that of the Murman-Cole scheme, which is comparable in the present context to a finite difference approximation of the equation

$$\phi_x \phi_{xx} = h \varepsilon \phi_x \phi_{xxx} ,$$

where $\varepsilon = 0$ if $\phi_x < 0$ but $\varepsilon = 1$ if $\phi_x > 0$. This is not a conservation law because the variable factor ε is not differentiated.

An advantageous compromise would seem to be to develop an intermediate scheme suggested by the conservation law

$$(\phi_x^2)_x = h \frac{\partial}{\partial x} (\varepsilon \frac{\partial}{\partial x} \phi_x^2) ,$$

in which ε is now differentiated. The appearance of ε in the last equation means that the solutions should include a shock point x_0 to the left of which $\varepsilon = 1$ and to the right of which $\varepsilon = 0$. The derivative ϕ_x should approach zero from the left at x_0, but may be negative to the right. On the other hand, the conservation form of the equation implies that $\phi_x^2 - h\varepsilon(\phi_x^2)_x$ as well as ϕ should remain continuous. Applying our boundary conditions, we conclude that

$$\phi_x^2 = C^2 \frac{1 - \varepsilon \exp(x-x_0)/h}{1 - \exp(a-x_0)/h}$$

and that the location of the shock point x_0 is defined by the nonlinear relation

$$\frac{B-A}{C} = \int_a^b \left[\frac{1-\varepsilon \exp(x-x_0)/h}{1-\exp(a-x_0)/h}\right]^{1/2} \frac{dx}{2\varepsilon-1} .$$

In the limit as $h \to 0$ this reduces to our earlier formula for x_0.

To implement the above idea as a difference scheme we use central difference formulas to represent the differential equation on the left-hand side, but retarded differences to represent the artificial viscosity on the right. Taking h as the mesh width, we obtain

$$(\phi_{j+1}-\phi_j)^2 - (\phi_j-\phi_{j-1})^2 = p_j - p_{j-1},$$

where

$$p_j = \max\{0,(\phi_{j+1}-\phi_{j-1})\}\ (\phi_{j+1}-2\phi_j+\phi_{j-1}) .$$

Here p_j reduces to the left-hand side at supersonic points for which $\phi_{j+1} > \phi_{j-1}$, so the scheme is effectively retarded in the supersonic zone. At the shock point, however, $p_j = 0$ and $p_{j-1} \neq 0$ so that the sum of the central and backward difference operators is obtained, corresponding to the shock point operator introduced by Murman (cf. [12]). It can be verified that this difference scheme admits a unique solution which satisfies the correct shock jump condition.

Now consider the problem of calculating the transonic flow past a body in space of two or three dimensions. The solution satisfies a variational principle which asserts that the integral of the pressure p over the flow region is stationary with respect to perturbations of the velocity potential ϕ. A discrete version of this principle leads to second order accurate finite difference equations in conservation form, and it is especially helpful in treating the natural boundary condition on ϕ and the free surface condition at a vortex sheet (cf. [3]). For transonic flow the principal part of the Euler equation coming from the variational principle should be left as it stands. Instead of directly retarding the difference scheme for the differential equation in the manner of Murman and Cole, a suitable artificial viscosity should be added in conservation form. By using retarded difference expressions to represent the viscosity we then arrive at an effectively retarded scheme in conservation form.

To handle shock waves according to the theory outlined above it is suggestive to look for appropriate weak solutions of a partial differential equation for ϕ of the invariant divergence form

$$\nabla(\rho \nabla\phi) = h \nabla[\frac{\varepsilon}{q} \nabla(\rho q)] ,$$

where $q = |\nabla\phi|$ is the speed, ρ is the density defined by Bernoulli's law, h is an artificial viscosity coefficient, and ε vanishes when the flow is subsonic but is positive when the flow is supersonic. The introduction of the one sided term ε is motivated by the decision process of Murman and Cole, while the highest order derivatives appearing in the artificial viscosity are equivalent to a derivative of the Laplacian $\nabla^2\phi$ in the direction of the flow. In the next section we shall construct a convergent iterative scheme to solve the resulting difference equations by introducing additional terms that involve an artificial time parameter. Experience shows that the term on the left can be replaced by a quasilinear differential operator not in conservation form without entirely losing the shock condition, provided that the operator is represented by a suitably centered finite difference expression and a conservation form is retained for the artificial viscosity. The mean value theorem can be applied to expressions of the form $f_{j+1/2} - f_{j-1/2}$ appearing in the difference equations for the conservation form, where f is a function of the velocity components. It can then be deduced that in substituting the quasilinear form for the differential operator the shock jump condition would be retained to second order in the shock strength if ρ/c^2 were constant, where c is the local speed of sound. This is the case for a ratio of specific heats $\gamma = 2$, as in the shallow water equations. We shall subsequently refer to schemes of this type in which the differential equation is represented in quasilinear form, with artificial viscosity added in conservation form, as quasiconservative, while we shall refer to schemes retaining conservation form for both the differential equation and the artificial viscosity as fully conservative.

The shock condition that is lost in the original Murman-Cole scheme turns out to be the conservation of mass. Since ρq is stationary at Mach number $M = 1$, the scheme remains valid anyway up to errors of the second order in the shock strength $M^2 - 1$. Moreover, in considering the differential equation for two dimensional flow past an airfoil with a single valued stream function ψ, global considerations show that the total mass flux $\int d\psi$ is actually conserved across the shocks even when they do not satisfy the exact shock condition. Thus the method of Murman and Cole provides a good approximation to the flow at nearly sonic speeds.

Denoting by c_* the critical speed and using the subscript ∞ to indicate free stream quantities, we introduce the integral

$$C_D = \frac{2}{\rho_\infty q_\infty^2} \int \left[(\phi_x - c_*)\,d\psi + p\,dy\right] .$$

This integral for the wave drag coefficient is independent of path. The jump of the integrand across a shock wave is of the third order in the shock strength $M^2 - 1$, and the formula makes sense even though we have neglected changes in the entropy. It reduces to an obvious pressure integral over the profile that we use in practice.

In our computer programs we have used a version of the scheme of Murman and Cole that tends to yield shock waves behind which the speed drops barely below the speed of sound through a jump roughly one half that to be expected from exact theory. This is consistent with the existence of the forward and smeared shock solutions we made reference to at the beginning of the section for a one dimensional model. Such behavior is, however, also typical of the interaction of weak shock waves with a turbulent boundary layer. We have had excellent success with the method when we included a boundary layer correction, and it leads to remarkably stable results. More recently we have modified the programs to try out both quasiconservation and full conservation forms of the equations of motion like those that have been described above. For the most part the modified programs give pressure profiles quite similar to the ones obtained the old way. Some examples appear where the exact shock condition has resulted in better agreement with experimental data. Comparisons with exact hodograph solutions show that the additional terms introduced by representing the artificial viscosity in conservation form lead to larger truncation errors in supersonic regions where smooth recompression of the flow occurs (cf. Chapter II, Section 2). Where the flow is expanding in the supersonic region, comparisons of solutions on coarse and fine grids suggest that the truncation error remains quite small, on the other hand. Our conclusion is that the original procedure is generally satisfactory in practice, but we do include in the handbook a listing of an option for a quasiconservative scheme for purposes of comparison. Finally, we mention that our programs seem to give a reliable estimation of drag creep, but predict drag rise for Mach numbers that are about 0.02 smaller than those observed in wind tunnels. The discrepancy may be due to wall effect.

3. Iterative Schemes for Three Dimensional Analysis

Since the appearance of Volume I substantial progress has been made in developing methods for the computation of transonic flows. In this section we shall develop a rotated finite difference scheme to treat flows at both subsonic and supersonic free stream speeds, and we shall develop an iterative procedure to solve the resulting difference equations. The rotated scheme is invariant under a transformation of coordinates, so that any curvilinear system can be introduced that is appropriate for the geometry of a specific problem. The method has been applied both in two dimensional calculations of the flow over an airfoil with a correction for the boundary layer, and in three dimensional calculations of the flow past an isolated yawed wing of finite aspect ratio. In selecting the latter problem to demonstrate the feasibility of three dimensional calculations we are motivated by R. T. Jones' concept of an asymmetric airplane with an oblique wing and by our access to his experimental data for comparison with the theory [6].

To be specific we consider the three dimensional case. Ignoring changes in the entropy and using rectangular coordinates x, y, z, we have the partial differential equation

$$(c^2-u^2)\phi_{xx} + (c^2-v^2)\phi_{yy} + (c^2-w^2)\phi_{zz} - 2uv\phi_{xy} - 2vw\phi_{yz} - 2uw\phi_{xz} = 0$$

for the velocity potential ϕ, where c is the speed of sound defined by Bernoulli's law

$$\frac{q^2}{2} + \frac{c^2}{\gamma-1} = \text{const.} , \qquad q^2 = u^2 + v^2 + w^2 ,$$

and u, v, w are the velocity components. We look for weak solutions ϕ that satisfy an entropy inequality asserting that the speed decreases across any shock wave, and we use the standard approximations of linearized theory to specify what happens on the vortex sheet behind an obstacle.

The numerical method employed incorporates two basic features. First, in common with previous successful schemes for treating transonic flows, it uses retarded differencing in the supersonic zone to introduce artificial viscosity and to reproduce the proper upstream region of dependence. Second, it uses an iterative procedure which can be viewed as an embedding of the steady state equation in a suitably constructed artificially time dependent equation.

The difference scheme described in Volume I is based on the assumption that the flow is more or less aligned with one coordinate direction. To allow more flexibility this assumption has been removed from the new scheme. Instead the equation of motion is rearranged as if it were expressed locally in a coordinate system aligned with the flow. Let s denote the stream direction. Then the equation can be written in the canonical form

$$(c^2-q^2)\phi_{ss} + c^2(\Delta\phi-\phi_{ss}) = 0 ,$$

where $\Delta\phi$ denotes the Laplacian of ϕ. Since the direction cosines of the stream direction are u/q, v/q, and w/q, the streamwise second derivative can be expressed in the form

$$\phi_{ss} = \frac{1}{q^2} (u^2\phi_{xx}+ v^2\phi_{yy}+ w^2\phi_{zz}+ 2uv\phi_{xy}+ 2vw\phi_{yz}+ 2uw\phi_{xz}) .$$

At supersonic points retarded difference formulas are used to represent all contributions to ϕ_{ss} , while central difference formulas are used to represent all contributions to $\Delta\phi-\phi_{ss}$. At subsonic points all terms are represented by central difference formulas in the conventional manner. The result is a coordinate invariant difference scheme which is correctly oriented with the flow. The artificial viscosity induced in the supersonic zone ensures the proper entropy inequality, so that compression shocks are admitted while expansion shocks are excluded. By using the rotational invariance of the Laplacian the need to calculate explicit directional derivatives normal to the streamlines is avoided.

The difference equations are highly implicit, containing downstream points even in the supersonic zone. In order to devise a convergent iterative scheme to solve them, it is convenient to regard the iterations as steps in an artificial time coordinate. Let Δt be the time step, and let the superscript $^+$ denote updated values. Then a typical central difference formula at the mesh point $(i\Delta x,\ j\Delta y,\ k\Delta z)$ is

$$\frac{\phi_{i+1,j,k} - (1+r\ \Delta x)\phi^{+}_{i,j,k} - (1-r\ \Delta x)\phi_{i,j,k} + \phi^{+}_{i-1,j,k}}{(\Delta x)^2} ,$$

which may be regarded as a finite difference approximation of

$$\phi_{xx} - \frac{\Delta t}{\Delta x}(\phi_{xt} + r\ \phi_t) ,$$

where r is a parameter determined by the overrelaxation factor. Thus we must consider a time dependent equation which contains mixed space and time derivatives.

If we divide the equation of motion through by c^2 and neglect lower order terms, its principal part will have the form

$$(M^2-1)\phi_{ss} - \phi_{mm} - \phi_{nn} - 2\alpha_1\phi_{st} - 2\alpha_2\phi_{mt} - 2\alpha_3\phi_{nt} = 0 ,$$

where M is the local Mach number q/c, m and n denote directions normal to s, and the coefficients α_1, α_2 and α_3 depend on the split between new and old values in the difference scheme. Introducing a new time coordinate

$$T = t + \frac{\alpha_1 s}{M^2-1} - \alpha_2 m - \alpha_3 n ,$$

we obtain the time dependent equation

$$(M^2-1)\phi_{ss} - \phi_{mm} - \phi_{nn} - \left\{\frac{\alpha_1^2}{M^2-1} - \alpha_2^2 - \alpha_3^2\right\}\phi_{TT} = 0 \ .$$

In order for this equation to remain hyperbolic with s as the time-like direction, it is necessary to satisfy the compatibility condition

$$\text{(A)} \qquad \alpha_1 > \sqrt{(M^2-1)(\alpha_2^2+\alpha_3^2)} \ , \quad M > 1 \ .$$

This indicates the need to augment the term in ϕ_{st} to compensate for the terms in ϕ_{mt} and ϕ_{nt} produced by the central difference formulas. For that purpose ϕ_{ss} is evaluated using retarded difference formulas of the form

$$\frac{2\phi^+_{i,j,k} - \phi_{i,j,k} - 2\phi^+_{i-1,j,k} + \phi_{i-2,j,k}}{(\Delta x)^2} \ ,$$

which can be interpreted as approximating

$$\phi_{xx} + 2\,\frac{\Delta t}{\Delta x}\,\phi_{xt} \ .$$

The compatibility condition (A) may still be violated near the sonic boundary, where the coefficient of ϕ_{ss} vanishes. Therefore the term

$$\frac{\beta}{\max\left(|u|,|v|,|w|\right)}\,(u\phi_{xt} + v\phi_{yt} + w\phi_{zt})$$

should be added, where β is a damping parameter chosen by the user. In this term ϕ_{xt} is represented as

$$\frac{\phi^+_{i,j,k} - \phi_{i,j,k} - \phi^+_{i-1,j,k} + \phi_{i-1,j,k}}{\Delta x\,\Delta t} \ ,$$

with similar formulas for ϕ_{yt} and ϕ_{zt}. In some calculations it proves possible to set $\beta = 0$.

The three dimensional analysis program, called Program J, has been implemented in this form, using mixtures of new and old values to represent the spatial derivatives. Alternatively we can regard the iterative scheme as being derived directly by the addition of time dependent terms

$$\sum \alpha_i \phi_{x_i t} + r\phi_t$$

to the steady state equation. Then all spatial derivatives would be evaluated using old values, and the time dependent terms would be explicitly added to produce an artificially time dependent equation whose solution converges to the steady state solution. This approach proves more fruitful when one wishes to devise an iterative scheme for the equation in conservation or quasiconservation form, since it can be carried over unaltered. A conventional relaxation scheme, on the other hand, would require the densities at the midpoints of each mesh interval to be calculated twice, first with old and then with new values.

To derive a quasiconservation form of the rotated scheme we start from our invariant partial differential equation

$$\nabla(\rho\ \nabla\phi) = h\ \nabla[\frac{\varepsilon}{q}\ \nabla(\rho q)]$$

for the velocity potential ϕ, in which central differences will be used on the left and retarded differences will be used in the evaluation of the artificial viscosity on the right. Working with rectangular coordinates to simplify matters, we substitute a quasilinear form on the left to obtain

$$c^2\nabla^2\phi - \sum \phi_{x_i}\phi_{x_j}\phi_{x_i x_j} = \sum h_i \frac{\partial}{\partial x_i}\ \varepsilon\ \frac{c^2-q^2}{q}\ \frac{\partial q}{\partial x_i}\ .$$

This differs from the original equation by a factor c^2/ρ , where c is the local speed of sound, and by the use of anisotropic viscosity coefficients h_i which are different in the different coordinate directions. For these we take

$$h_i = |\phi_{x_i}| \, \Delta x_i \, ,$$

where the Δx_i represent prospective mesh sizes. Neglecting partial derivatives of lower order on the right, we arrive at the result

$$c^2 \nabla^2 \phi - \sum \phi_{x_i} \phi_{x_j} \phi_{x_i x_j} = \sum \Delta x_i \frac{\partial}{\partial x_i} \varepsilon \frac{c^2 - q^2}{q} |\phi_{x_i}| q_{x_i} \, .$$

To derive the rotated scheme from this, all that is necessary is to write down a similar equation in a more general orthogonal coordinate system and to replace the partial derivatives by suitable finite difference approximations, with the divergence terms on the right retarded or advanced according as the corresponding coefficients ϕ_{x_i} are positive or negative. We do not go into further details here because the rotated scheme has already been established on other grounds. The main advantage of the present approach is that it applies just as easily to the true conservation *form of the equation for ϕ as it does to the simpler quasi*-conservation form.

We summarize our ideas in the following

PROPOSITION. Transonic flow past a body in two or three dimensions can be calculated by means of a finite difference approximation of a partial differential equation for the velocity potential ϕ that consists of a central finite difference representation of the usual differential operator on ϕ plus artificial viscosity and artificial time terms that are defined by a formula such as

$$\sum \frac{\partial}{\partial x_i} \rho \frac{\partial \phi}{\partial x_i} = \sum \frac{\partial}{\partial x_i} \frac{h_i \varepsilon}{q} \frac{\partial}{\partial x_i} \rho q + \sum \alpha_i \phi_{x_i t} + r \, \phi_t \, ,$$

where the h_i stand for anisotropic artificial viscosity coefficients, the α_i comprise a vector governing the characteristics of an iterative scheme that involves the artificial time t, and r is a relaxation factor.

The proposition has the advantage that it breaks up into separate blocks of terms the contributions from the fundamental equation of motion, from the addition of artificial viscosity, and from the insertion of mixed partial derivatives with respect to artificial time that specify the iterative scheme we use. The more general point of view should be helpful in applying the method to other flow problems. It has been implemented in the quasiconservation option for the two dimensional program with boundary layer correction, Program H (cf. Chapter III, Sections 5 and 7).

4. Choice of Coordinates and Conformal Mapping

The rotated finite difference scheme which we have presented in Section 3 makes it possible to treat transonic flow problems in a variety of coordinate systems. The choice of coordinates can be quite important in a specific application. It is desirable that the coordinates follow the surface in regions of high curvature such as the leading edge. This can be achieved by conformal mapping. In three dimensional calculations, however, we wish to avoid the extra terms in the equations that would result from the use of different mappings at different spanwise stations. For calculation of the flow over a yawed wing we have therefore used a square root transformation independent of the spanwise direction z to unfold the wing about a singular line just inside the leading edge, which is assumed to be straight. In the plane of each wing section we thus obtain parabolic coordinates X and Y which are related to the physical coordinates x and y by the conformal transformation

$$x + iy = (X + iY)^2 .$$

The wing profile emerges as a shallow bump above the line $Y = 0$, so we use a second shearing transformation to obtain slightly nonorthogonal coordinates which coincide with the wing surface.

For the calculation of two dimensional flow past an airfoil a better distribution of mesh points is obtained by mapping the exterior of the airfoil conformally onto the interior of the unit circle. In particular, for the inclusion of a boundary layer correction based on iterating the map function, it is desirable to have a fast and accurate method of doing the conformal mapping. The purpose of this section is to describe such a method, based on the fast Fourier transform, which has been found to stand up well in practice.

The calculations are performed in the interior of the unit circle using polar coordinates r and ω. The modulus h of the mapping derivative becomes asymptotic to $1/r^2$ as r tends to zero. To avoid introducing large truncation errors that come from finite difference expressions for $\partial h/\partial\omega$ and $\partial h/\partial r$ it is convenient to introduce the mapping to the exterior of the circle and to use an explicit inversion.

Because we have in mind the extension of the boundary layer as a wake behind the airfoil, we wish to map the exterior of a profile with an open trailing edge in the z-plane onto the exterior of a circle in the σ-plane so that the wake is reduced to a slit. The well known method of Theodorsen and Garrick [16], in which the mapping of a star shaped contour in the z-plane onto a circle in the σ-plane is expressed in terms of $\log(z/\sigma)$, does not allow for an open trailing edge. For this reason it is preferable to express the mapping in terms of its derivative

$$\frac{dz}{d\sigma} = f(\sigma) \ .$$

Since the point at infinity is to be preserved, the Laurent series for $f(\sigma)$ must contain only inverse powers of σ. If the coefficient of $1/\sigma$ is $\tilde{c}$, then according to the Cauchy integral theorem, integration of the map function around any circle exterior to the unit circle in the σ-plane results in a gap

$$z_2 - z_1 = \oint \frac{dz}{d\sigma}\, d\sigma = 2\pi i \tilde{c} \ .$$

Thus the mapping represents the wake as a gap with a constant thickness determined by the residue $\tilde{c}$.

In order to devise a simple iterative process for calculating the mapping function it is convenient to write

$$\frac{dz}{d\sigma} = \exp \left[\sum_{n=0}^{N} \frac{c_n}{\sigma^n} \right] .$$

If α and s are the tangent angle and arc length of the contour in the z-plane, then

$$\log \frac{ds}{d\omega} + i(\alpha-\omega) = \sum_{n=0}^{N} c_n \, e^{-in\omega} .$$

Separating the real and imaginary parts, we obtain

$$\log \frac{ds}{d\omega} = \sum_{n=0}^{N} (a_n \cos n\omega + b_n \sin n\omega) ,$$

$$\alpha - \omega = \sum_{n=0}^{N} (b_n \cos n\omega - a_n \sin n\omega) ,$$

where

$$c_n = a_n + ib_n .$$

Now the tangent angle α is known as a function of the arc length s from the definition of the contour. Therefore if we have an estimate $s = s(\omega)$ of the arc length as a function of the angle ω in the circle plane, we can calculate the Fourier coefficients of the series for $\alpha - \omega$. Then by reversing the sine and cosine coefficients we can construct the conjugate Fourier series for log (ds/dω). The expression for ds/dω can be integrated in turn to provide an improved estimate of s(ω), and the process can be iterated until the corrections to s(ω) become negligible.

The Fourier series is not suitable for representing a jump. In order to apply this method to the mapping of an airfoil it is therefore desirable to modify the representation of $\frac{dz}{d\sigma}$ by including a Schwarz-Christoffel term to allow for a corner or cusp at the trailing edge. Thus we set

$$\frac{dz}{d\sigma} = \left(1 - \frac{1}{\sigma}\right)^{1-E/\pi} \exp\left[\sum_{n=0}^{N} \frac{c_n}{\sigma^n}\right] ,$$

where E is the included angle at the trailing edge. The gap becomes

$$2\pi i\tilde{c} = 2\pi i\left(\frac{E}{\pi} - 1 + c_1\right) .$$

The same iterative procedure is then used. Provided that c_1 is fixed by the gap condition, it converges rather rapidly for reasonably smooth airfoils. It is generally sufficient to use the flat plate relationship of s to ω for the starting guess, and the maximum correction to s(ω) usually reduces to the order of 10^{-9} in about 10 iterations.

To obtain good accuracy it is important to use a sufficiently large number of terms in the Fourier series. If the mapping function is to be calculated at 2K equally spaced mesh points $\omega_k = k\pi/K$ around the circle it is best to take N = K terms and to replace the Fourier series by trigonometric interpolation formulas for the corresponding values α_k of the angle α. This is equivalent to evaluating the Fourier coefficients by the trapezoid rule. It has been shown by Snider [15] that for a function with ℓ continuous derivatives the maximum error in the trigonometric interpolation formulas is of the order $(1/K)^{\ell-1}$.

The trigonometric interpolation formulas have the advantage that they can be evaluated with the aid of the fast Fourier transform. Thus we can reduce the number of computer operations at each iteration from $O(K^2)$ to $O(K \log K)$. In fact we can avoid the explicit evaluation of the coefficients a_n and b_n altogether and obtain the conjugate function $\log(ds/d\omega)$ directly from $\alpha - \omega$ with the aid of back-to-back fast Fourier transforms as follows: First let the angle function $\alpha - \omega$ at the mesh points $2k$ and $2k+1$ be regarded as the real and imaginary parts of a complex function

$$u_k = \alpha_{2k} - \omega_{2k} + i\left(\alpha_{2k+1} - \omega_{2k+1}\right)$$

defined for $0 \leq k \leq K-1$. Let U_k be the complex Fourier transform of u_k, and let

$$V_0 = 0 ,$$

$$V_k = U_k e^{-i\omega_k} , \qquad k > 0 .$$

Then the real and imaginary parts of the Fourier transform v_k of V_k yield $\log(ds/d\omega)$ at the shifted mesh points $2k+1$ and $2k+2$,

$$\log \left.\frac{ds}{d\omega}\right|_{2k+2} - i \log \left.\frac{ds}{d\omega}\right|_{2k+1} = v_k .$$

Unfortunately the contour is usually not defined by an explicit formula, but only by a table of coordinates. Thus we are obliged to use an interpolation procedure to estimate the tangent angle $\alpha(s)$ at the values s_k corresponding to equally spaced points in the circle plane. Most airfoils have continuous slope and curvature, but it is unwise to assume continuity of derivatives of order higher than the second. Accordingly, it is appropriate to use cubic splines for interpolation. Since neither x nor y is monotone around the contour it is not possible to use splines to represent one coordinate as a function of the other. Instead x and y are represented separately by splines as functions $x(\mu)$ and $y(\mu)$ of a monotone parameter μ. We can use the estimated arc length s itself as the parameter. With this choice the derivatives of the functions we encounter may become infinite at the trailing edge. It is better to remove this singularity by using as a parameter the stretched arc length

$$\mu = \cos^{-1} \frac{2s-s_0}{s_0} ,$$

where s_0 is the total arc length. This reduces the sensitivity to errors in the coordinates near the trailing edge.

The combination of the derivative representation of the mapping with trigonometric interpolation by fast Fourier transforms and with splines to represent the contour has been found in practice to provide a rapid and robust numerical algorithm which is not critically dependent on a high degree of smoothness in the data. Thus it is well suited to the treatment of a boundary layer correction, which can lead to rather irregular shapes, particularly in the earlier iterations.

5. Two Dimensional Analysis with a Turbulent Boundary Layer Correction

We turn our attention to the problem of adding a turbulent boundary layer correction to the two dimensional program for analysis of transonic flow past a supercritical wing section. Our approach is to calculate the displacement thickness $\delta = H\theta$ by means of von Karman's equation

$$\frac{d\theta}{ds} + (H + 2 - M^2)\,\frac{\theta}{q}\,\frac{dq}{ds} = \tau$$

for the momentum thickness θ, where M is the local Mach number and the shape factor H and the skin friction τ are determined from semi-empirical formulas of Nash and Macdonald [14]. We ignore the laminar boundary layer because it is so thin, and we initialize θ at a transition point that can be set arbitrarily. First we run a certain number of cycles of the flow computation using a two dimensional version of the new rotated finite difference scheme described in Section 3. Then we alter the shape of the airfoil by adding on a current estimate of the displacement thickness δ. After that we update the map function in the unit circle by the fast Fourier transform procedure outlined in Section 4, and finally we return to the flow calculation and repeat the whole process. Various smoothings of δ are introduced to overcome instabilities caused by the dependence of the boundary condition on the tangential pressure gradient dq/ds. However, the most serious difficulty encountered, which we shall discuss in more detail, stems from the inaccuracy and rapid variation of the Nash-Macdonald formulas for the shape factor H near the point where the boundary layer separates.

According to the turbulent boundary layer method of Nash and Macdonald [14], separation is predicted when the adverse pressure gradient becomes so big that

$$\mathrm{SEP} = -\frac{\theta}{q}\frac{dq}{ds} \geq .004 \;.$$

Beyond this threshold their semi-empirical formulas are less accurate and we have felt free to modify them. Thus over most of the airfoil, and in particular through any shock wave, we replace the parameter SEP by .004 if the calculation shows it to exceed that value. A reasonable simulation of the effects of turbulent boundary layer shock wave interaction seems to result for weak shocks. Because the flow outside the boundary layer cannot withstand arbitrarily large adverse pressure gradients, and because experimental data indicate that the pressure coefficient C_p tends to become linear or even flatten out after separation, we allow for an option that alters the computed values of C_p for insertion in the von Karman equation after the final point of separation by extrapolating them linearly to a base value. Since the adverse pressure gradient at the trailing edge ought to remain finite, we iterate to determine the base value of the pressure coefficient until the computed distribution of C_p just ceases to be monotonic over some prescribed interval near the trailing edge. Our idea is to thicken the displacement δ beyond final separation of the boundary layer until the pressure coefficient C_p begins to turn around and flatten out at the trailing edge as we know it does in wind tunnel tests. It is our experience that this procedure yields a quite reliable estimate of the distribution of lift at the rear of a heavily aft loaded airfoil.

Extensive comparisons with test data have been used to adjust the parameters at our disposal in arriving at a scheme of this type so as to achieve a good computer simulation of the physical flow. The details are best studied by examination of the full listing of our computer program in Section 5 of Chapter III. We mention that certain monotonicity properties which the final displacement thickness δ ought to have are imposed as part of the smoothing process. Both δ and the base pressure coefficient are underrelaxed to obtain convergence; the change in the latter at each iteration is made proportional to the smallest increment of C_p across any pair of adjacent mesh points in a prescribed interval at the rear of the profile.

It has been found best to integrate the von Karman equation over a mesh of 81 points equally spaced on the circumference of the unit circle, even when the flow is computed at a mesh twice as fine, because this leads to the right thickening of the boundary layer through a shock. Satisfactory agreement with the experimental data that is available to us seems to have been achieved (cf. Section 3 of Chapter II). Better resolution would require either an improvement in the semi-empirical description of the turbulent boundary layer we have drawn from the paper of Nash and Macdonald [14] or a more penetrating theory of the near wake in transonic flow past a heavily aft loaded airfoil. We note that Bavitz [2] has also developed an iterative procedure to include a boundary layer correction, for which he reports good agreement with experimental data.

6. Design in the Hodograph Plane: A New Model of the Trailing Edge

We turn our attention to the problem of design of shockless airfoils by the method of complex characteristics described in Volume I (cf. [1]). This transforms an analytic function depending on many arbitrary parameters into a solution of the partial differential equations of gas dynamics. The main difficulty lies in the choice of parameters to obtain desired properties of the flow in the physical plane. New insight has been gained by experience and as a result of wind tunnel tests. In particular, it has been found essential to improve on our old model of the trailing edge.

Several of our airfoils have been tested in wind tunnels achieving high enough Reynolds numbers so the boundary layer becomes turbulent throughout the transonic zone (cf. [7,8,9]). The agreement between theoretical and experimental pressure distributions turned out to be better when there was little aft loading and no boundary layer correction than it was in heavily aft loaded cases with a boundary layer correction, for which the observed lift was fifteen or twenty percent less than its predicted value. The loss of lift for the corrected cases seems to be due to boundary layer separation over the last three to five percent of chord on the upper surface of the profile (cf. Chapter II, Section 6, Figure 6). Since, as we indicated in the previous section, large adverse pressure gradients in the exterior flow cannot be sustained by the boundary layer, the design pressure gradient obtained near the trailing edge by the hodograph method ought to remain bounded on the upper surface. Heavy aft loading can still be achieved by allowing the favorable pressure gradient on the lower surface to become infinite (cf. Section 1 of Chapter II). The purpose of the present section is to describe a refinement of the Kutta-Joukowski model of the tail in the hodograph plane that enables us to generate such pressure distributions, which are like those observed experimentally (cf. [8]) and should, therefore, give rise to much less loss of lift in practice.

The method of complex characteristics constructs a flow from initial data defined by an analytic function g of the complex variable η specified in a plane that is analogous to the hodograph plane, but is simpler because a substitution has been made so the mapping to the physical plane becomes one-to-one. Since we deal primarily with cusped tails, the Kutta-Joukowski condition implies that the image of the tail in the η-plane lies at a critical point of the stream function ψ identified with some finite speed q (cf. the figures in Section 1 of Chapter II). Corresponding to the airfoil there is a profile $\psi = 0$ in the η-plane which must enclose no singularities of the input function $g(\eta)$ other than one at $\eta = 0$ associated with the point at infinity in the physical plane. In Volume I we allowed the stream function ψ to have a period about the origin in order to obtain a thickness at the trailing edge from which a boundary layer correction could be subtracted. However, we now ask that ψ remain single valued and introduce a period in the physical coordinate y instead. This has the advantage of making the values of the pressure coefficient C_p match up across the two edges of the trailing streamlines $\psi = 0$ that proceed from the tail out to infinity and in effect delineate the boundary layer wake. The new model of the trailing edge thus obtained agrees with the one we have been using all along in our analysis programs.

The requirement that the adverse pressure gradient remain finite on the upper surface of the airfoil near the tail means that in the η-plane the corresponding arc of the profile must become tangent to the level curve of the speed q through the tail. There are two different ways this can happen. First, we can impose a simple critical point of ψ at the tail, with q stationary on the profile $\psi = 0$ and with the angle of the flow monotonically increasing as we pass from the upper surface to the lower surface. Both surfaces are concave at such a tail, which has an appreciable base pressure coefficient and does not generate excessive aft loading (cf. Airfoil 79-03-12 in Section 1 of Chapter II). Second, there can be a multiple critical point of ψ at the tail, with q stationary only on the upper surface but exhibiting an unbounded favorable gradient on the lower surface, and with the flow angles above and below turning downward to form a hook at the tail (cf. Airfoil 72-06-16 in Section 1 of Chapter II). This is the case of a heavily aft loaded airfoil, and its success depends on the pressure coefficient C_p being nearly zero at the tail. Thus the speed at the tail is almost the same as that at infinity and the flow angles are sizeable, resulting in significant aft camber. When our design program is used to implement the two configuratios we have described, the new input parameter NCR specifying the number of constraints, which controls the order of the critical point of ψ at the tail, must be set equal to five and seven, respectively.

7. Design in the Hodograph Plane: Choice of Parameters

The purpose of this section is to describe improvements in our design method that have been introduced since Volume I appeared. Some minor additions and corrections to the basic computer programs have been made, and they are listed in Section 8 of Chapter III. We believe that the better model of the trailing edge which has been presented in Section 6 should be used in designing any future shockless airfoils. We have also worked out a number of new examples (cf. Section 1 of Chapter II), both before and after the discovery of the more desirable treatment of the trailing edge problem, and they furnish perhaps the best guide available to those interested in the design method, which has turned out to be harder for the uninitiated user to implement than we had hoped. Here we supplement the examples with a brief account of the improved techniques that enabled us to arrive at them.

In order to design a transonic airfoil by the method of complex characteristics, we pick a desirable location, i.e. desirable speed and slope, for the tail and lay down automation paths through which the profile ought to pass in the subsonic part of the complex η-plane, which plays the role of a hodograph plane. Then we place logarithmic singularities of the input function $g(\eta)$, whose coefficients are to be found automatically, at appropriate points surrounding the profile. We distribute more of them near the tail if a multiple critical point of the stream function ψ

is imposed there and if separation is to be avoided by fitting the profile to a level curve of the speed q. To achieve shockless flow few constraints should be set on the supersonic arc of the airfoil. However, the problem is overdetermined not only because of its transonic character, but also because we tend to impose too many interpolation conditions in the subsonic domain. Thus the most important consideration is to choose the branch point B of the transformation from the η-plane to the true hodograph plane, the location of the tail, and the more significant parameters defining the analytic function $g(\eta)$ so as to arrive at a compatible configuration. A good general principle to follow is that as few constraints as possible should be introduced and as few logarithms as possible should be used. Moreover, the coefficients of those terms that are required should be made as small as possible. The objective then becomes to obtain a smooth, closed profile $\psi = 0$ with as many desirable physical properties as the various trade-offs of the configuration at hand allow.

As we have indicated, the first shockless airfoils we developed that had heavy aft loading failed to come up to their design specifications in wind tunnel tests because we did not shape the profile in the η-plane closely enough to the level curve of q at the tail to eliminate significant boundary layer separation. Our present belief is that this fit should be carried far enough to ensure that the inequality

$$\mathrm{SEP} = -\frac{\theta}{q}\frac{dq}{ds} \leq .004 ,$$

which we use as a criterion on the momentum thickness θ for no separation to take place, holds in the flow calculated by the hodograph method, which occurs outside the boundary layer. Airfoils conforming to the new criterion have more camber near the tail than corresponding examples designed before (cf. Airfoils 70-10-13 and 70-11-12), which helps explain why the earlier models experienced a loss of lift. Runs of the analysis program we described in Section 5, which seems to simulate test data well, do suggest that five new airfoils we designed theoretically to have no separation ought to meet our specifications in practice (cf. Airfoils 79-03-12, 72-06-16, 71-08-14, 70-10-13 and 65-14-08). For a more satisfactory verification of the theory we look forward to seeing the experimental results from a test of one of these airfoils now being planned at the National Aeronautical Establishment in Ottawa.

Usually a new airfoil takes between 25 and 100 trial runs of the computer program to design, with most of the runs using about five minutes of CDC 6600 machine time at mesh parameter MRP = 2. However, John Dahlin of the McDonnell Douglas Corporation was able to design Airfoil 71-08-14 in only twelve runs starting from a combination of the input data for Airfoils 72-06-16 and 70-10-13. Full automation to prescribe the location of the arc of the profile inside the sonic locus of the η-plane is recommended. For a case with specifications close to those of one that has already been finished, 25 runs should suffice. On the other hand, when we tried out the concept of eliminating separation by fitting the profile $\psi = 0$ to the level curve of q through the tail in the η-plane, both our first example, the heavily aft loaded Airfoil 70-10-13 based on a multiple critical point with NCR = 7, and our second example, the low lift Airfoil 79-03-12 based on a simple critical point with NCR = 5, required about 100 runs to perfect. The difficulties encountered were to meet a large collection of interpolation conditions near the tail. The problem of achieving smooth nose curvatures, which caused a lot of trouble in preparing the examples for Volume I, is now made significantly easier by locating only one or two logarithms in the left half-plane, by cutting off the automation paths well short of the nose, and by choosing the parameters XU and XV that control the slope and curvature at the stagnation point so that they are more compatible with the automation paths.

One of the most subtle aspects of the inverse method of designing transonic airfoils is the control of limiting lines that result from overlap in the transformation from the hodograph plane. It is as important to control the limiting line that tends to appear at the front of the superonic zone, where there is a pressure peak, as it is to eliminate sharp gradients at the rear, where shock waves will appear at off-design conditions. In our method, problems of interpolation and analytic continuation play a significant role in the location of logarithmic singularities of the initial function $g(\eta)$. Experience shows that the limiting lines are very sensitive to logarithms situated in the transonic region of the η-plane just below the supersonic paths of integration (cf. the figures in Section 1 of Chapter II). We have found that a logarithm with a pure imaginary automated coefficient should be placed near the negative imaginary axis in this region. The position of a second fully automated logarithm near the point $\eta = -.1 - .4i$ then exercises strong control over the pressure peak at the front of the supersonic zone, which is also favorably influenced by a heavily weighted automation path making the profile cross the sonic locus early, say for $\text{Re}\{\eta\} < -.6$. A secondary peak appears in front of the primary one when this logarithm is moved toward the sonic locus. However, by careful adjustment the secondary peak can be merged into the primary one so as to form an unusually well rounded pressure distribution with supersonic speeds attained within five percent of chord from the leading edge (cf. Airfoil 78-06-10). Such a distribution can be expected to reduce the drag creep that tends to occur just below the shockless design condition. Experience has shown that designing airfoils near the limit of feasible specifications leads to poor performance at off-design conditions. It is preferable to reduce the size of the supersonic zone by subtracting, say, .01 from the maximum possible design Mach number. This also tends to suppress drag creep.

8. Bibliography

1. F. Bauer, P. Garabedian, and D. Korn, Supercritical Wing Sections, Lecture Notes in Economics and Mathematical Systems, vol. 66, Springer-Verlag, New York, 1972.
2. P. Bavitz, "Analysis Method for Two Dimensional Transonic Viscous Flow," NASA TND 7718, 1974.
3. O. Betancourt, "Three Dimensional Computation of Magnetohydrodynamic Equilibrium of Toroidal Plasma without Axial Symmetry," AEC Research and Development Report MF-67 and COO-3077-49, Courant Institute of Mathematical Sciences, New York University, 1974.
4. L. A. Graham, R. T. Jones, and F. W. Boltz, "An Experimental Investigation of Three Oblique-Wing and Body Combinations at Mach Numbers between 0.60 and 1.40," NASA TM X-62,256, Ames Research Center, 1973.
5. A. Jameson, "Iterative Solution of Transonic Flows over Airfoils and Wings," Comm. Pure Appl. Math., vol. 27 (1974).
6. R. T. Jones, "New Design Goals and a New Shape for the SST," Astronautics and Aeronautics, vol. 10 (1972), pp. 66-70.
7. J. J. Kacprzynski, "A Second Series of Wind Tunnel Tests of the Shockless Lifting Airfoil No. 1," Project Report 5x5/0062, National Research Council of Canada, Ottawa, 1972.
8. J. J. Kacprzynski; "Wind Tunnel Test of a Shockless Lifting Airfoil No. 2," Laboratory Technical Report LTR-HA-5x5/0067, National Research Council of Canada, Ottawa, 1973.
9. J. J. Kacprzynski, L. H. Ohman, P. R. Garabedian, and D. G. Korn, "Analysis of the Flow Past a Shockless Lifting Airfoil in Design and Off-Design Conditions," Aeronautics Report LR-554, National Research Council of Canada, Ottawa, 1971.

10. D.G. Korn, "Numerical Design of Transonic Cascades", to appear.

11. E. McIntyre, "Design of Transonic Cascades by Conformal Transformation of the Complex Characteristics," Thesis, New York University, to appear.

12. E. M. Murman, "Analysis of Embedded Shock Waves Calculated by Relaxation Methods," A.I.A.A. Computational Fluid Dynamics Conference, Palm Springs, California, 1973.

13. E. M. Murman and J. D. Cole, "Calculation of Plane Steady Transonic Flows," A.I.A.A.J., vol. 9 (1971), pp. 114-121.

14. J. F. Nash and A. G. J. Macdonald, "The Calculation of Momentum Thickness in a Turbulent Boundary Layer at Mach Numbers up to Unity," Aeronautical Research Council C. P. No. 963, London, 1967.

15. A. D. Snider, "An Improved Estimate of the Accuracy of Trigonometric Interpolation," S.I.A.M.J. Numerical Analysis, vol. 9 (1972), pp. 505-508.

16. T. Theodorsen and I. E. Garrick, "General Potential Theory of Arbitrary Wing Sections," NACA Technical Report 452, 1933.

17. R. T. Whitcomb, "Review of NASA Supercritical Airfoils," Ninth International Congress on Aeronautical Sciences, Haifa, Israel, 1974.

II. DATA

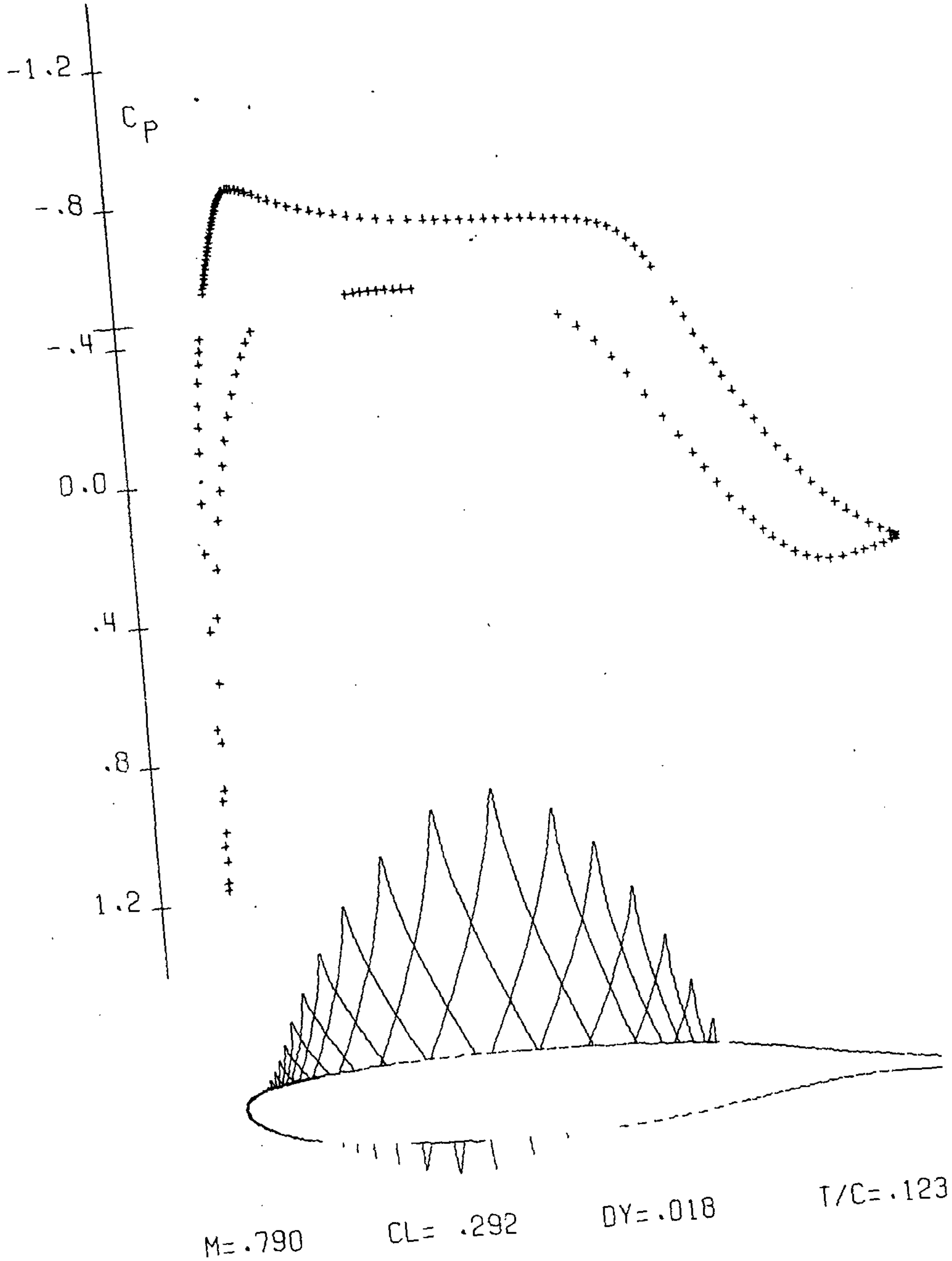
1. Catalog of Transonic Airfoils

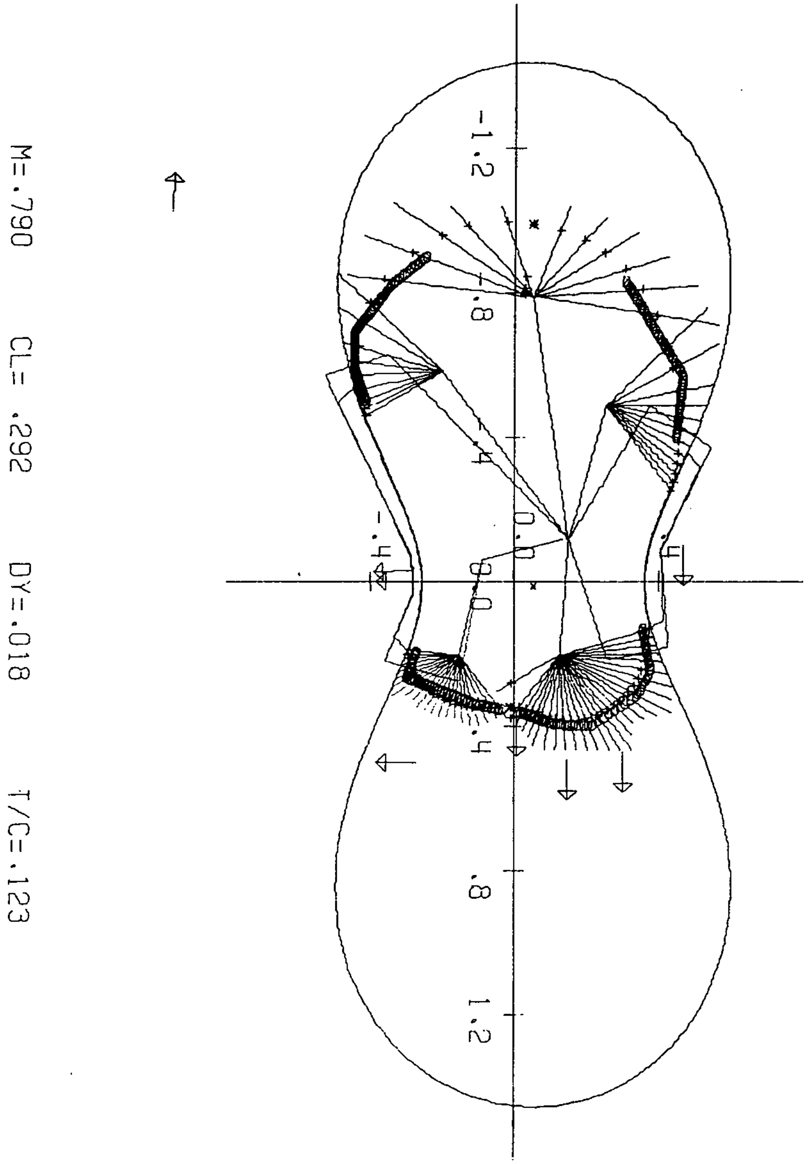
In this section we present some of the more promising airfoils which we have been able to design. These are labelled with six digit numbers composed of successive pairs indicating the free stream Mach number M, the lift coefficient C_L, and the thickness chord ratio T/C. For every example there is a plot of the airfoil geometry and the Mach lines together with the design pressure distribution. There is also a plot of the η-plane, related to the hodograph plane, which shows the location of the logarithms and automation paths (cf. Volume I) plus the remainder of the integration paths from Tape 6. Listings of Tape 7 and the automation paths from Tape 6 have been included. This should enable the reader to run the examples through Programs B and D and to use them as starting points for new designs. For our newer and better airfoils we have listed x, y coordinates also, so that it is not necessary to run the programs to obtain a definition of their geometry.

The newer airfoils are given first. The best are 79-03-12, 72-06-16, 71-08-14, 70-10-13 and 65-14-08, which incorporate the new model of the tail designed to eliminate boundary layer separation. Airfoil 79-03-12 uses NCR = 5 (see pages 27-28) and has a low lift coefficient in the range suitable for executive jets. Airfoil 78-06-10 is notable for its very smooth pressure distribution, obtained by controlling the limiting line at the front of the supersonic zone (see pages 31-32). Airfoil 72-06-16 is the closest we have come to simulating the supercritical wing of the T2-C. Airfoil 70-10-13 was designed especially for R. T. Jones to be used in his plans for a transonic transport with an oblique wing. It was designed to maximize the product M^2C_L while having a thickness ratio of twelve percent and meeting constraints imposed by the need to avoid drag creep and separation. It is expected to give an optimal three dimensional lift drag ratio at moderate supersonic speeds (cf. Section 5). Airfoil 65-14-08 resulted from applying the same criterion. Airfoils 70-11-12 and 65-15-10 are included largely for purposes of comparison; they have cusped trailing edges for which separation cannot be avoided. Airfoil 60-13-10 is an example of a subcritical design.

Airfoils 75-06-12, 75-07-15 and 82-06-09 are from an older series, and are included, not because they represent the best that can currently be achieved, but because they have been tested (cf. [7, 8,9]). The Grumman Aerospace Corporation used Airfoil 70-07-20 as a starting point to develop an airfoil by our design method for a series of tests in their transonic wind tunnel (cf. Section 6, Figure 6). A version of Airfoil 78-06-10 has been tested by Whitcomb at the NASA Langley Research Center. There are also plans for a two dimensional test of Airfoil 79-03-12 in the high Reynolds number wind tunnel of the National Aeronautical Establishment in Ottawa, for a three dimensional test of an oblique wing based on Airfoil 70-10-13 at the NASA Ames Research Center, and for a two dimensional test of a modified version of Airfoil 65-14-08 at the Grumman Aerospace Corporation.

Our final example is a compressor blade which was designed in collaboration with E. McIntyre (cf. [11]). This was obtained using a new program which permits the design of two dimensional cascades of airfoils and will be published elsewhere [10]. Additional transformations of the η-plane allowing for additional branch points enable one to design highly cambered blades suitable for turbines.


-1.2
C$_P$
-.8
-.4
0.0
.4
.8
1.2
M=.790
CL= .292
DY=.018
T/C=.123


-1.2
-.8
-.4
0.0
.4
.8
1.2
-.4
0.0
.4
M=.790
CL= .292
DY=.018
T/C=.123

```
                                                        07/23/74




                        RUN= -109

           CIRCULATORY FLOW ABOUT A TRANSONIC AIRFOIL

           M= .790    CL= .292    DY= .018    T/C= .123



                        TAPE 6, PATH 0

                      2   0
                       -.800    0.000   2
                      -1.000    0.000   2

                      2   0
                        .300     .050   2
                        .340    -.062   2



                          TAPE 7

-6-109   4  -.12  .15  .08  1.40  .790 -.009  -.052  .120 1.50   5
22   1   2   5   6  10  13  14  17  18  33  34  37  38  42  49
50  53  54  57  58  61  62

-.056  -.051   .520   .150        -.069  -.020   .500   .300
0.000  -.226  -.070   .470        -.169  -.094   .400   .010

-.102  -.022 -1.050  -.950        0.000  0.000 -2.000  0.000
0.000  0.000 -2.000  0.000        0.000  0.000 -2.000  0.000

-.169   .139  -.030  -.310         .039   .185   .500  -.300
 .100   .070  0.000  -.300        0.000  0.000  0.000  -.900

 .209  -.065   .460   .100        -.044   .074   .500   .050
 .119  -.027   .095   .032         .038  -.033  4.000  1.000
```

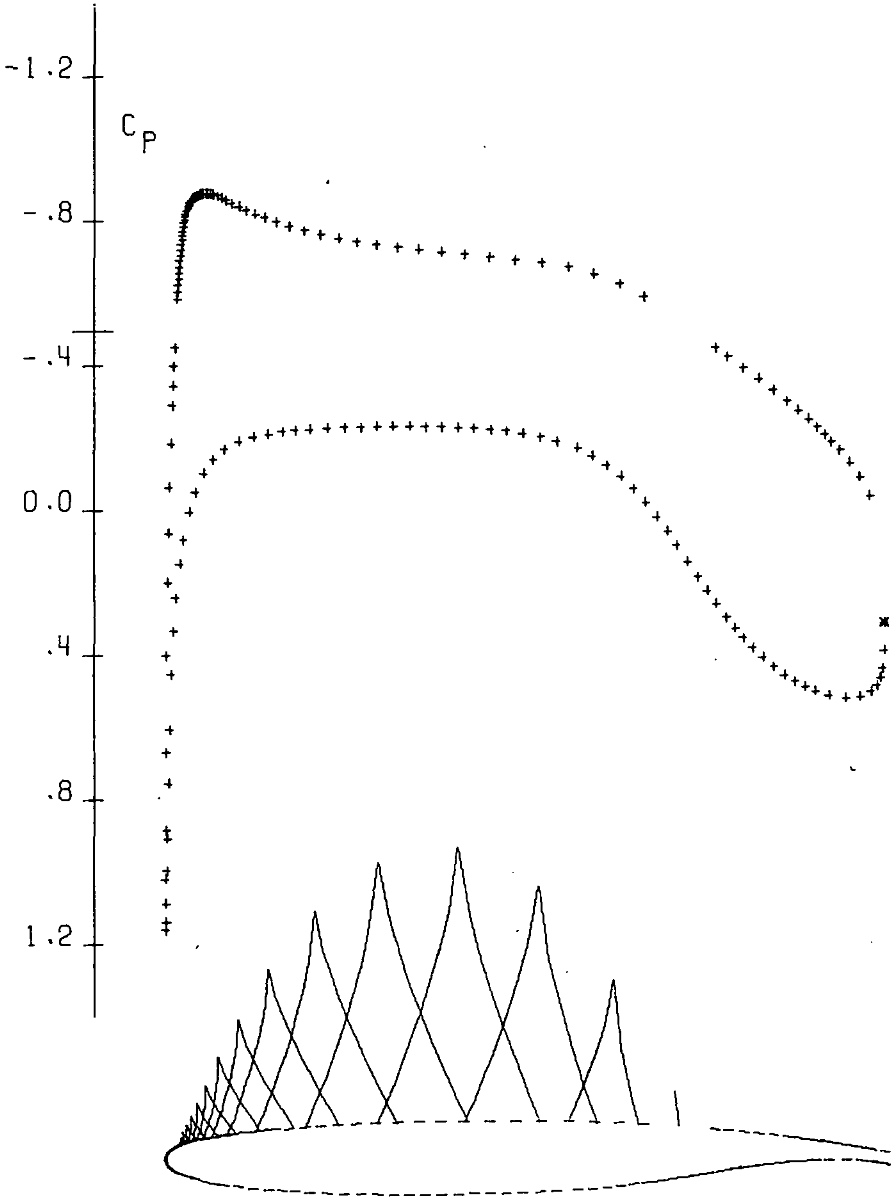
```
          AUTOMATION PATHS

     5   0
      -.070     -.130    1
        .190    -.325   -1
        .260    -.350    3
        .335    -.160    2
        .350    -.090    2

     3   0
      -.910     -.300   -1
      -.830     -.400    2
      -.700     -.500    2

     3   0
      -.700     -.500   -1
      -.600     -.500    4
      -.510     -.470    4

     3   0
      -.410      .390   -1
      -.580      .410    2
      -.840      .250    2

     4   0
        .120     .295   -1
        .240     .320    2
        .310     .280    1
        .390     .160    1

     3   0
        .390     .160   -1
        .390     .060    1
        .355    -.045    2
```

LISTING OF COORDINATES FOR AIRFOIL 79-03-12 RN=20.0 MILLION

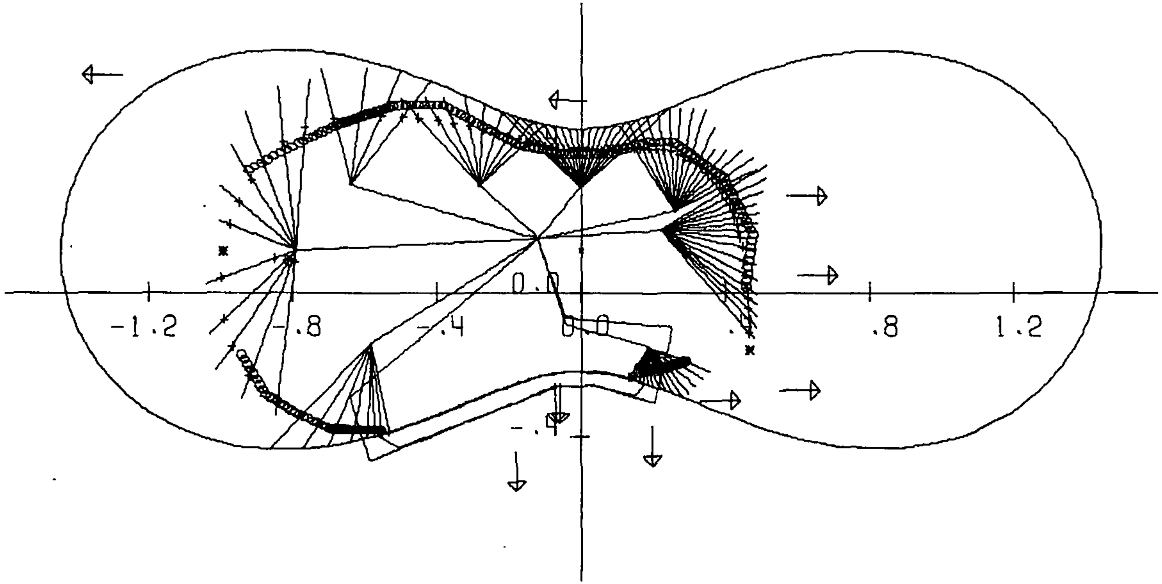
X	Y	YS	ANG	KAPPA	CP	THETA	SEP
1.00000	0.00000	.00348	-2.68	-19.62	.3010	.00285	-.00457
.99953	.00002	.00351	-2.38	-7.18	.3022	.00285	-.00434
.99812	.00007	.00360	-2.07	-2.62	.3052	.00285	-.00372
.99578	.00015	.00371	-1.77	-1.92	.3090	.00287	-.00287
.99249	.00024	.00384	-1.46	-1.41	.3127	.00288	-.00203
.98827	.00034	.00397	-1.16	-1.15	.3159	.00290	-.00147
.98312	.00043	.00411	-.84	-.99	.3192	.00291	-.00125
.97705	.00050	.00422	-.52	-.89	.3229	.00292	-.00117
.97005	.00054	.00430	-.17	-.84	.3271	.00293	-.00113
.96214	.00053	.00434	.20	-.81	.3317	.00295	-.00109
.95332	.00047	.00433	.61	-.81	.3364	.00297	-.00102
.94361	.00033	.00426	1.06	-.83	.3413	.00298	-.00093
.93302	.00008	.00410	1.58	-.87	.3460	.00300	-.00077
.92156	-.00030	.00384	2.17	-.93	.3501	.00301	-.00052
.90925	-.00084	.00345	2.85	-.99	.3526	.00301	-.00014
.89615	-.00158	.00288	3.62	-1.04	.3524	.00300	.00036
.88229	-.00256	.00208	4.46	-1.06	.3481	.00296	.00094
.86774	-.00381	.00099	5.34	-1.03	.3386	.00289	.00154
.85257	-.00535	-.00044	6.20	-.95	.3231	.00278	.00209
.83688	-.00717	-.00224	7.01	-.82	.3013	.00264	.00250
.82074	-.00925	-.00441	7.70	-.67	.2734	.00247	.00276
.80423	-.01157	-.00591	8.27	-.51	.2402	.00229	.00285
.78741	-.01408	-.00970	8.69	-.35	.2024	.00210	.00281
.77033	-.01673	-.01270	8.96	-.20	.1608	.00192	.00269
.75304	-.01947	-.01581	9.09	-.07	.1160	.00176	.00251
.73556	-.02227	-.01898	9.10	.06	.0685	.00160	.00231
.71793	-.02506	-.02213	8.97	.18	.0187	.00146	.00210
.70017	-.02784	-.02520	8.74	.28	-.0327	.00133	.00188
.68228	-.03054	-.02817	8.39	.38	-.0849	.00122	.00167
.66429	-.03312	-.03099	7.95	.46	-.1371	.00112	.00146
.64620	-.03557	-.03363	7.44	.52	-.1883	.00103	.00126
.62800	-.03785	-.03609	6.86	.57	-.2374	.00095	.00107
.60971	-.03995	-.03834	6.25	.59	-.2830	.00088	.00089
.59132	-.04186	-.04038	5.62	.60	-.3240	.00082	.00072
.57284	-.04357	-.04220	4.99	.58	-.3595	.00077	.00057
.55426	-.04510	-.04381	4.38	.55	-.3892	.00073	.00044
.53561	-.04643	-.04523	3.81	.52	-.4138	.00069	.00034
.51688	-.04759	-.04645	3.27	.48	-.4342	.00065	.00026
.49811	-.04858	-.04750	2.77	.45	-.4510	.00062	.00020
.47931	-.04941	-.04839	2.30	.42	-.4647	.00059	.00016
.46051	-.05010	-.04912	1.86	.40	-.4758	.00056	.00012
.44174	-.05064	-.04971	1.43	.39	-.4846	.00054	.00009
.42302	-.05104	-.05016	1.03	.38	-.4917	.00051	.00007
.40439	-.05131	-.05047	.63	.37	-.4972	.00049	.00005
.38587	-.05145	-.05065	.24	.37	-.5016	.00046	.00004
.36749	-.05146	-.05070	-.15	.37	-.5052	.00044	.00003
.34929	-.05136	-.05064	-.53	.37	-.5082	.00042	.00003
.33128	-.05113	-.05045	-.91	.38	-.5109	.00039	.00002
.31351	-.05079	-.05015	-1.30	.39	-.5133	.00037	.00001
.29599	-.05033	-.04973	-1.69	.40	-.5146	.00035	.00001
.27876	-.04976	-.04919	-2.10	.42	-.5149	.00033	-.00000

X	Y	YS	ANG	KAPPA	CP	THETA	SEP
.26185	-.04908	-.04855	-2.52	.44	-.5143	.00031	-.00000
.24527	-.04829	-.04779	-2.95	.47	-.5129	.00028	-.00000
.22906	-.04739	-.04693	-3.40	.47	-.5120	.00026	.00001
.21324	-.04639	-.04597	-3.82	.48	-.5143	.00024	.00001
.19784	-.04530	-.04492	-4.27	.54	-.5176	.00022	.00001
.18290	-.04412	-.04377	-4.77	.63	-.5201	.00020	.00000
.16845	-.04285	-.04253	-5.34	.76	-.5200	.00018	-.00001
.15452	-.04146	-.04118	-6.02	.92	-.5160	.00016	-.00003
.14113	-.03996	-.03971	-6.80	1.10	-.5073	.00014	-.00005
.12829	-.03833	-.03811	-7.69	1.31	-.4931	.00012	-.00006
.11603	-.03657	-.03638	-8.70	1.53	-.4730	.00010	-.00007
.10433	-.03466	-.03451	-9.82	1.77	-.4469	.00009	-.00008
.09323	-.03262	-.03250	-11.04	2.03	-.4146	.00007	-.00009
.08271	-.03044	-.03036	-12.39	2.37	-.3746	.00005	-.00008
.07279	-.02813	-.02807	-13.88	2.75	-.3263	.00002	-.00004
.06347	-.02569	-.02566	-15.52	3.19	-.2698		
.05476	-.02313	-.02312	-17.31	3.71	-.2057	TRANSITION .	
.04667	-.02046	-.02045	-19.27	4.35	-.1345		
.03919	-.01769	-.01769	-21.44	5.18	-.0562		
.03233	-.01484	-.01484	-23.87	6.29	.0297		
.02611	-.01191	-.01191	-26.63	7.82	.1240		
.02054	-.00893	-.00893	-29.83	10.06	.2279		
.01562	-.00590	-.00590	-33.67	13.35	.3437		
.01138	-.00282	-.00282	-38.39	18.61	.4738		
.00783	.00029	.00029	-44.43	26.69	.6239		
.00499	.00346	.00346	-52.13	36.49	.7957		
.00282	.00671	.00671	-61.00	40.74	.9712		
.00129	.01004	.01004	-69.81	44.46	1.1053		
.00035	.01341	.01341	-79.21	47.57	1.1647	STAGNATION	
0.00000	.01674	.01674	-88.91	56.93	1.1273		
.00027	.01997	.01997	-100.65	64.30	.9756		
.00120	.02312	.02312	-112.14	56.43	.7330		
.00279	.02617	.02617	-122.54	50.52	.4330		
.00504	.02907	.02907	-132.80	45.10	.1132		
.00803	.03180	.03180	-141.81	31.21	-.1013		
.01180	.03443	.03443	-148.08	18.48	-.2264		
.01632	.03698	.03698	-152.66	12.80	-.3235		
.02156	.03948	.03948	-156.27	9.29	-.4087		
.02749	.04189	.04189	-159.26	7.21	-.4892		
.03410	.04422	.04422	-161.86	5.89	-.5648		
.04138	.04643	.04643	-164.22	4.99	-.6390		
.04932	.04851	.04850	-166.40	4.31	-.7124		
.05795	.05043	.05041	-168.40	3.49	-.7793	TRANSITION	
.06729	.05221	.05217	-170.01	2.44	-.8219		
.07736	.05387	.05380	-171.18	1.66	-.8366	.00002	.00000
.08812	.05545	.05535	-172.07	1.25	-.8360	.00006	.00000
.09954	.05697	.05682	-172.81	1.01	-.8287	.00008	.00002
.11160	.05842	.05823	-173.45	.84	-.8185	.00010	.00003
.12427	.05981	.05958	-174.02	.73	-.8071	.00012	.00004
.13752	.06113	.06087	-174.54	.64	-.7955	.00014	.00004
.15133	.06239	.06209	-175.02	.57	-.7842	.00016	.00004
.16566	.06359	.06325	-175.47	.52	-.7736	.00018	.00004
.18050	.06471	.06433	-175.89	.48	-.7640	.00020	.00004
.19581	.06575	.06533	-176.29	.44	-.7555	.00022	.00004
.21158	.06672	.06626	-176.68	.41	-.7479	.00025	.00004

X	Y	YS	ANG	KAPPA	CP	THETA	SEP
.22778	.06761	.06711	-177.04	.37	-.7410	.00027	.00004
.24438	.06842	.06788	-177.38	.34	-.7344	.00029	.00004
.26137	.06915	.06857	-177.70	.32	-.7282	.00032	.00004
.27870	.06979	.06918	-178.01	.30	-.7224	.00034	.00004
.29636	.07036	.06970	-178.30	.28	-.7171	.00036	.00004
.31431	.07085	.07015	-178.58	.27	-.7125	.00038	.00003
.33252	.07126	.07052	-178.85	.25	-.7086	.00041	.00003
.35098	.07158	.07081	-179.12	.25	-.7054	.00043	.00002
.36963	.07183	.07101	-179.38	.24	-.7029	.00045	.00002
.38846	.07199	.07113	-179.63	.23	-.7010	.00048	.00002
.40743	.07207	.07118	-179.88	.23	-.6994	.00050	.00001
.42651	.07207	.07113	-180.14	.23	-.6980	.00052	.00002
.44566	.07198	.07101	-180.39	.24	-.6966	.00054	.00002
.46487	.07180	.07079	-180.66	.24	-.6949	.00057	.00002
.48408	.07154	.07048	-180.93	.25	-.6927	.00059	.00003
.50328	.07118	.07009	-181.21	.26	-.6901	.00061	.00003
.52242	.07072	.06959	-181.51	.27	-.6872	.00064	.00004
.54149	.07017	.06899	-181.82	.29	-.6838	.00066	.00005
.56044	.06952	.06829	-182.14	.31	-.6797	.00068	.00007
.57925	.06876	.06748	-182.49	.34	-.6741	.00071	.00011
.59788	.06768	.06655	-182.88	.39	-.6658	.00073	.00018
.61629	.06689	.06548	-183.32	.46	-.6529	.00076	.00029
.63447	.06575	.06424	-183.85	.56	-.6324	.00079	.00047
.65237	.06445	.06282	-184.49	.68	-.6003	.00083	.00072
.67000	.06295	.06118	-185.23	.77	-.5544	.00089	.00100
.68736	.06124	.05929	-186.02	.80	-.4970	.00096	.00128
.70449	.05932	.05717	-186.79	.75	-.4333	.00105	.00154
.72138	.05720	.05482	-187.48	.67	-.3694	.00115	.00175
.73804	.05493	.05229	-188.08	.57	-.3062	.00125	.00194
.75448	.05252	.04960	-188.58	.47	-.2454	.00137	.00212
.77068	.05002	.04679	-188.96	.36	-.1879	.00150	.00228
.78661	.04747	.04390	-189.25	.25	-.1341	.00163	.00243
.80226	.04489	.04097	-189.43	.15	-.0843	.00178	.00256
.81760	.04233	.03803	-189.52	.06	-.0383	.00192	.00269
.83258	.03982	.03512	-189.53	-.03	.0039	.00208	.00280
.84718	.03738	.03227	-189.47	-.12	.0424	.00223	.00291
.86135	.03503	.02951	-189.33	-.21	.0776	.00239	.00300
.87506	.03280	.02689	-189.13	-.29	.1095	.00256	.00308
.88826	.03071	.02444	-188.87	-.38	.1383	.00272	.00313
.90091	.02877	.02219	-168.56	-.46	.1642	.00287	.00315
.91296	.02699	.02018	-188.21	-.55	.1871	.00302	.00313
.92438	.02538	.01841	-187.82	-.63	.2071	.00316	.00307
.93512	.02395	.01688	-187.40	-.72	.2244	.00329	.00296
.94514	.02268	.01558	-186.96	-.80	.2391	.00340	.00283
.95441	.02158	.01450	-186.52	-.88	.2514	.00350	.00267
.96289	.02065	.01360	-186.07	-.96	.2616	.00358	.00250
.97055	.01986	.01278	-185.63	-1.04	.2699	.00364	.00234
.97737	.01921	.01198	-185.20	-1.14	.2767	.00370	.00223
.98332	.01869	.01118	-184.79	-1.26	.2821	.00375	.00223
.98838	.01828	.01030	-184.40	-1.45	.2867	.00379	.00240
.99255	.01798	.00913	-184.02	-1.72	.2908	.00382	.00279
.99580	.01776	.00782	-183.66	-2.23	.2947	.00386	.00333
.99813	.01761	.00681	-183.32	-2.93	.2979	.00389	.00384
.99953	.01753	.00624	-182.99	-7.46	.3002	.00392	.00420
1.00000	.01751	.00606	-182.68	-19.86	.3010	.00393	.00433



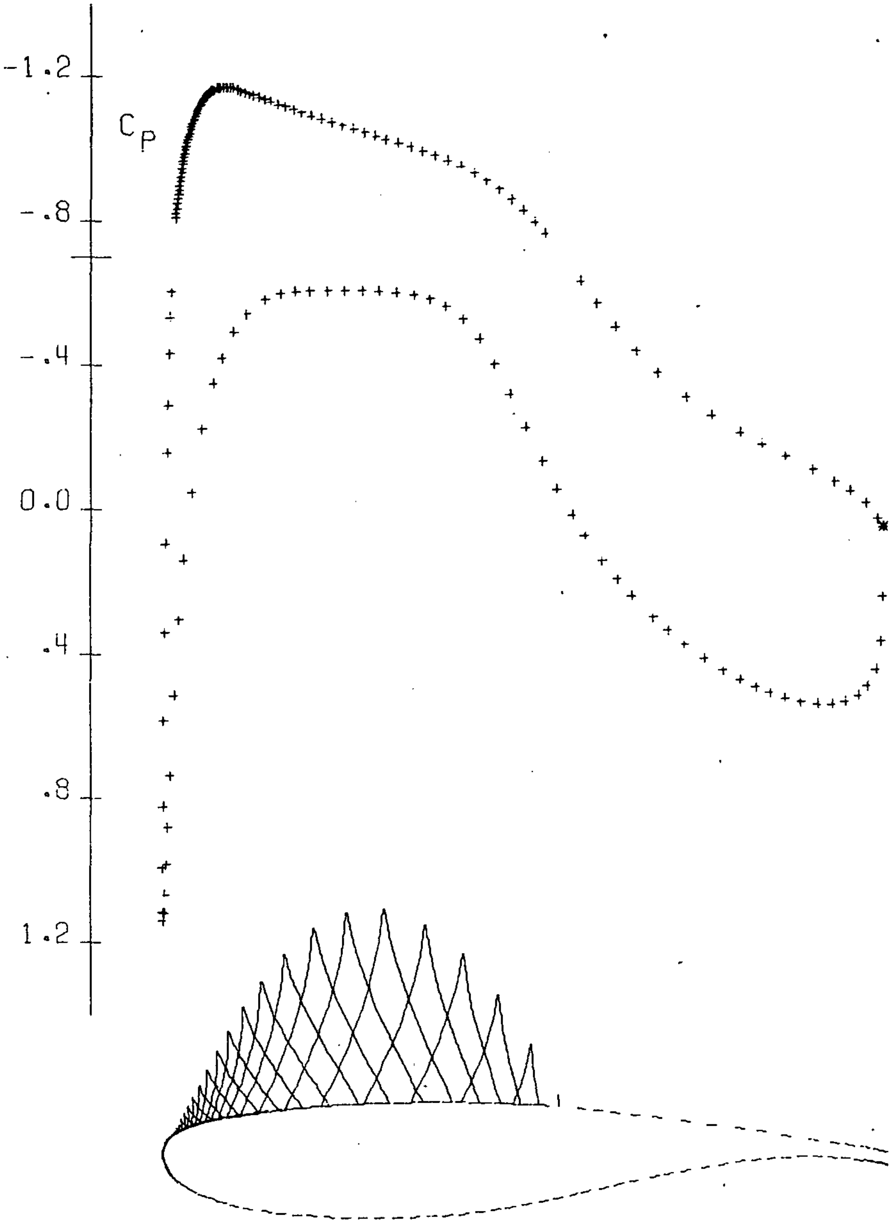
M=.780 CL= .591 DY=.016 T/C=.102



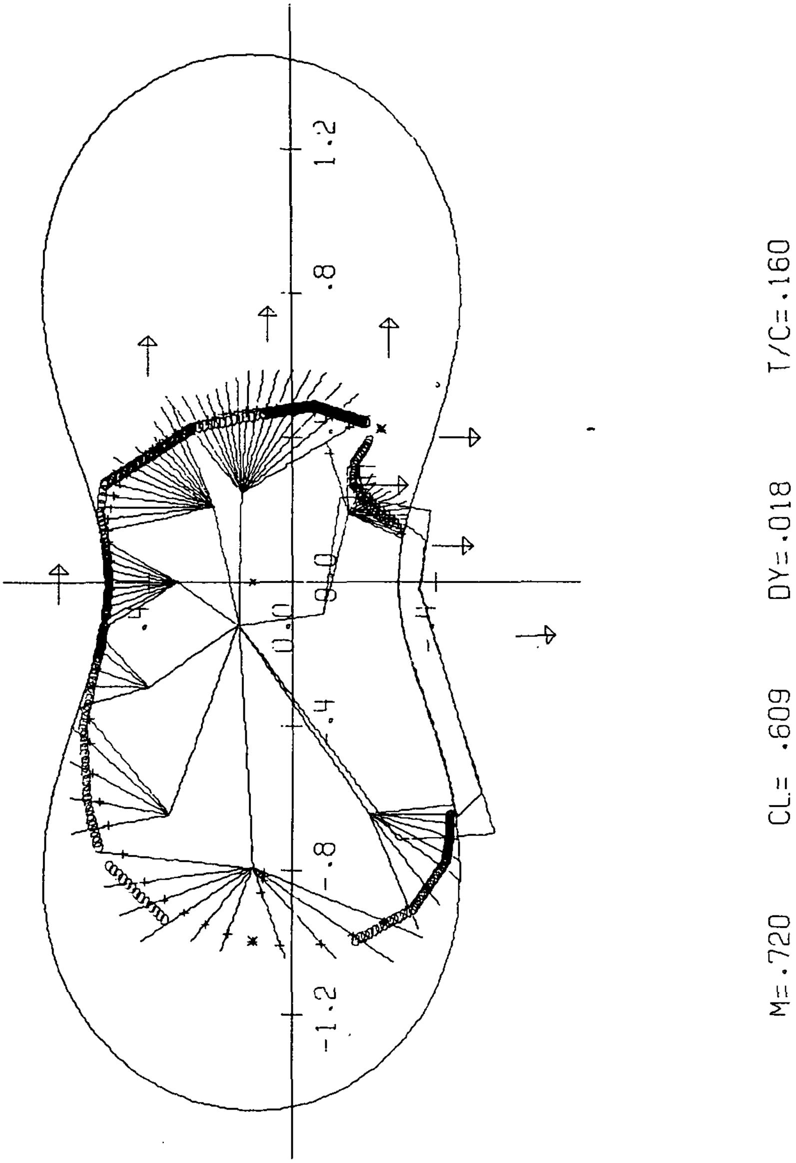
M=.780 CL= .591 DY=.016 T/C=.102

```
                                                   08/21/73



                          RUN=  -87

            CIRCULATORY FLOW ABOUT A TRANSONIC AIRFOIL

            M= .780    CL= .591    DY= .016    T/C= .102


                         TAPE 6, PATH 0

                      2   0
                       -.800    0.000    2
                      -1.000    0.000    2

                      2   0
                        .300     .050    2
                        .475    -.270    2


                         TAPE 7

-8 -87   4  -.12  .15  .08  1.40  .780 0.000  -.116  .055 1.50
25   1   2   5   6   9   10  13  14  17  18  22  33  34  37  38
41  42  49  50  53  54  57  58  61  62

-.256  -.131   .630   .050        .002  -.130   .600   .270
-.148  -.207   .360  -.300       -.023  -.167   .580  -.270

 .011  -.040 -1.300   .600       -.030  -.503  -.010   .530
0.000  0.000 -2.000  0.000       0.000  0.000 -2.000  0.000

-.090   .325   .200  -.400        .200  -.013  -.180  -.470
 .507   .057  -.056  -.280       -.320   .200  -.070  -.280

 .211   .032   .460   .100       -.063   .070   .500   .050
 .115   .044   .074   .029        .040  -.012  1.000  1.000
```

```
      AUTOMATION PATHS

4   0
 -.950      -.290   -2
 -.880      -.400    1
 -.800      -.450    1
 -.700      -.495    1

2   0
 -.700      -.495   -1
 -.560      -.495    5

4   0
 -.100      -.200    1
  .245      -.220    1
  .300      -.300   -1
  .245      -.318   15

4   0
 -.100      -.200    1
  .245      -.220    1
  .245      -.318   -1
  .170      -.310   15

5   0
 -.075       .240   -1
 -.180       .270    1
 -.270       .320    1
 -.390       .390    1
 -.530       .390    1

4   0
 -.530       .390   -1
 -.685       .340    2
 -.790       .290    1
 -.930       .220    1

4   0
 -.075       .240   -1
  .080       .240    1
  .200       .280    1
  .270       .280    1

4   0
  .270       .280   -1
  .410       .190    1
  .490       .040    1
  .470      -.100    1
```



M=.720 CL= .609 DY=.018 T/C=.160


1.2
.8
0.0
-.4
-.8
-1.2
M=.720
CL= .609
DY=.018
T/C=.160

```
                                                                07/23/74




                                  RUN=  -20

                    CIRCULATORY FLOW ABOUT A TRANSONIC AIRFOIL

                    M= .720    CL= .609    DY= .018    T/C= .160



                                 TAPE 6, PATH 0


                               2  0
                                -.800    0.000    2
                               -1.000    0.000    2

                               2  0
                                 .300    0.000    2
                                 .430    -.350    2



                                   TAPE 7

-8-020   4  -.12  .15  .08  1.40  .720  .001  -.109  .060 1.50    7
23   1   2   5   6   9  10  13  14  33  34  38  41  42  45  46
49  50  53  54  57  58  61  62


-.252  -.170   .680   .070         .118  -.247   .600   .400
 .041   .179   .650  -.270        -.129  -.132  -.030   .650

0.000  0.000 -2.000  0.000        0.000  0.000 -2.000  0.000
0.000  0.000 -2.000  0.000        0.000  0.000 -2.000  0.000

-.122   .361   .268  -.241        -.200   .138   .100  -.420
 .149  -.141  -.150  -.650        -.191   .218   .400  -.440

-.061   .071   .200   .010        -.174   .217  -.100   .050
 .157  -.008   .167   .077         .075  -.012  1.000  1.000
```

AUTOMATION PATHS

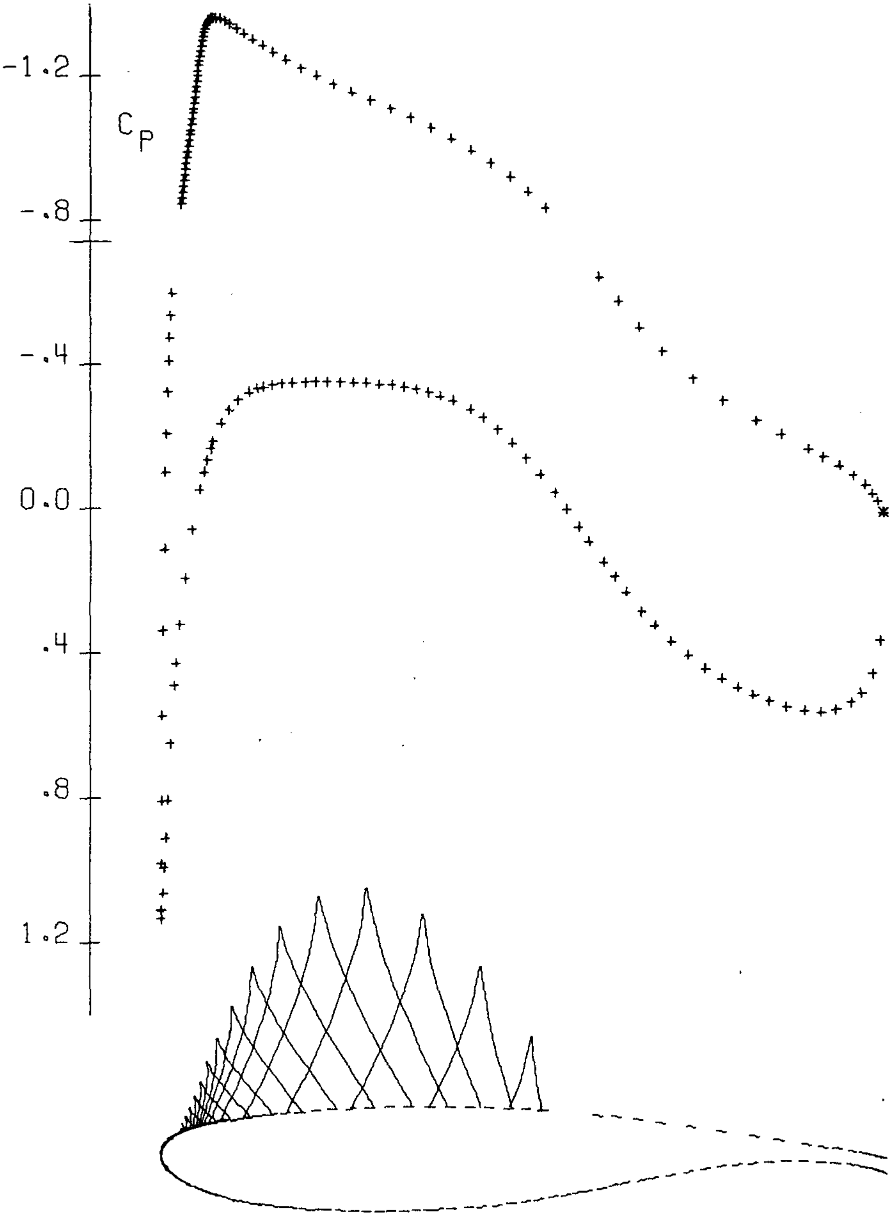
```
5  0
-1.000    -.290  -1
 -.910    -.450   1
 -.780    -.540   2
 -.710    -.550   5
 -.650    -.550   5

2  0
 -.940     .240  -1
 -.790     .390   1

5  0
 -.128     .394  -1
 -.210     .416   2
 -.410     .460   1
 -.620     .450   1
 -.740     .420   1

5  0
 -.098     .390  -1
 0.000     .382   2
  .102     .392   2
  .200     .410   1
  .280     .400   1

3  0
  .280     .400  -1
  .440     .160   3
  .480    -.030   1

3  0
  .480    -.030  -1
  .500    -.170   4
  .450    -.310   4

4  0
 -.100    -.170   1
  .170    -.370  -1
  .220    -.310   2
  .280    -.270   2

4  0
 -.100    -.170   1
  .280    -.270  -1
  .340    -.280   2
  .400    -.320   2
```

LISTING OF COORDINATES FOR AIRFOIL 72-06-16 RN=20.0 MILLION

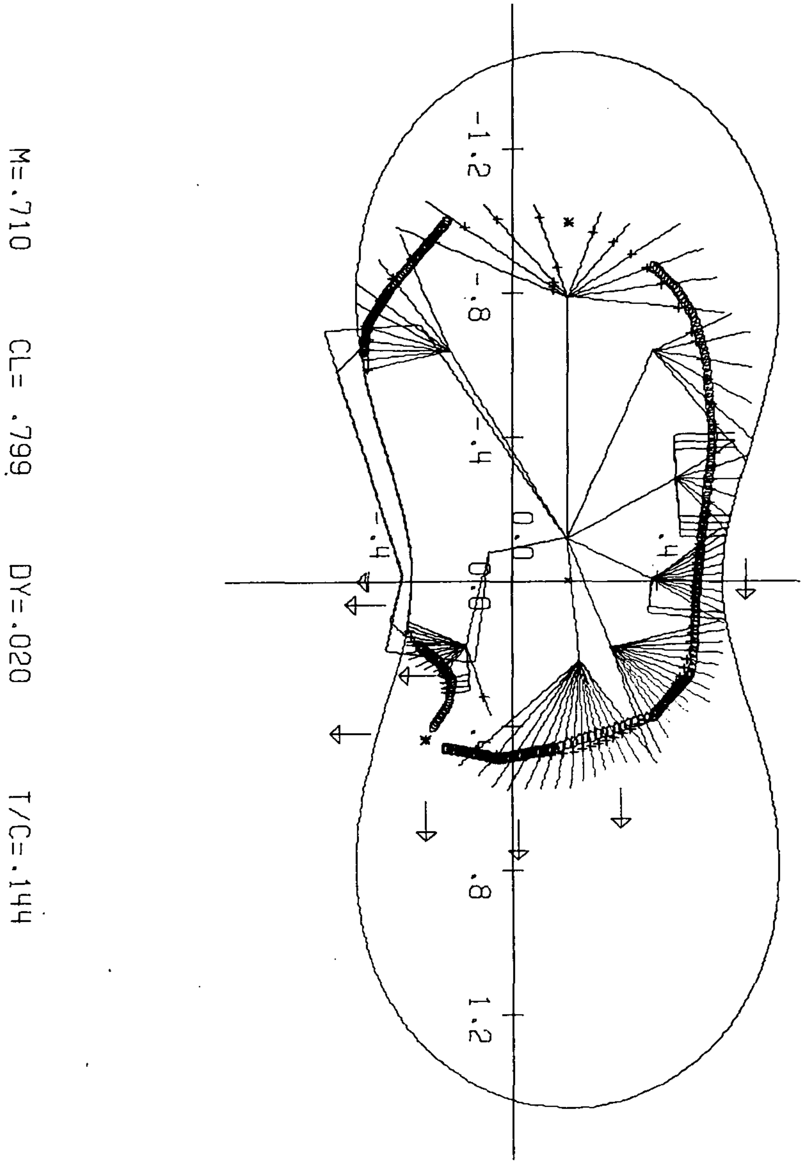
X	Y	YS	ANG	KAPPA	CP	THETA	SEP
1.00000	0.00000	.00421	-17.75	-287.89	.0431	.00330	-.34135
.99952	.00012	.00428	-13.93	-57.72	.3347	.00333	-.30462
.99809	.00047	.00471	-13.61	6.40	.3855	.00339	-.20995
.99566	.00105	.00541	-13.57	-4.05	.3582	.00345	-.09612
.99218	.00185	.00632	-12.45	-5.84	.4031	.00351	-.01238
.98758	.00280	.00742	-11.06	-4.33	.4559	.00376	.01056
.98187	.00385	.00866	-9.87	-3.13	.4784	.00414	-.01532
.97504	.00496	.00996	-8.69	-2.90	.4917	.00436	-.02270
.96708	.00608	.01126	-7.40	-2.67	.5077	.00442	-.00563
.95799	.00715	.01252	-6.09	-2.35	.5197	.00450	.00146
.94780	.00811	.01370	-4.78	-2.12	.5273	.00461	-.00275
.93653	.00892	.01477	-3.47	-1.94	.5324	.00467	-.00353
.92423	.00952	.01568	-2.16	-1.78	.5350	.00468	-.00053
.91093	.00987	.01636	-.86	-1.64	.5349	.00466	.00097
.89669	.00991	.01675	.43	-1.52	.5322	.00462	.00099
.88158	.00963	.01679	1.69	-1.41	.5270	.00454	.00137
.86566	.00898	.01640	2.93	-1.30	.5192	.00443	.00208
.84902	.00795	.01554	4.13	-1.20	.5086	.00428	.00255
.83172	.00653	.01418	5.27	-1.10	.4952	.00410	.00278
.81386	.00471	.01229	6.34	-1.00	.4791	.00390	.00297
.79552	.00251	.00990	7.34	-.90	.4602	.00368	.00311
.77676	-.00006	.00702	8.26	-.80	.4385	.00344	.00315
.75767	-.00298	.00370	9.10	-.71	.4142	.00320	.00312
.73831	-.00621	.00000	9.84	-.62	.3873	.00297	.00305
.71876	-.00972	-.00400	10.51	-.55	.3577	.00274	.00296
.69906	-.01348	-.00826	11.09	-.47	.3253	.00252	.00286
.67928	-.01745	-.01271	11.59	-.39	.2897	.00230	.00275
.65947	-.02158	-.01731	11.99	-.31	.2507	.00210	.00264
.63968	-.02584	-.02200	12.31	-.23	.2079	.00191	.00253
.61997	-.03018	-.02674	12.52	-.14	.1607	.00172	.00241
.60037	-.03455	-.03148	12.61	-.03	.1088	.00155	.00230
.58093	-.03889	-.03617	12.58	.09	.0516	.00138	.00217
.56167	-.04316	-.04075	12.40	.23	-.0109	.00123	.00203
.54262	-.04729	-.04516	12.06	.38	-.0786	.00109	.00187
.52379	-.05123	-.04936	11.56	.55	-.1511	.00097	.00169
.50519	-.05492	-.05328	10.87	.72	-.2269	.00086	.00149
.48680	-.05831	-.05686	10.02	.88	-.3037	.00076	.00127
.46861	-.06136	-.06008	9.02	1.01	-.3776	.00068	.00103
.45059	-.06404	-.06291	7.92	1.08	-.4433	.00061	.00079
.43270	-.06636	-.06534	6.80	1.08	-.4968	.00056	.00057
.41492	-.06831	-.06738	5.73	1.02	-.5363	.00052	.00039
.39723	-.06993	-.06908	4.72	.95	-.5634	.00049	.00025
.37965	-.07124	-.07045	3.80	.88	-.5813	.00046	.00016
.36217	-.07227	-.07153	2.94	.82	-.5929	.00043	.00010
.34483	-.07304	-.07235	2.15	.78	-.6003	.00041	.00006
.32765	-.07357	-.07292	1.40	.75	-.6050	.00038	.00004
.31065	-.07388	-.07327	.68	.73	-.6081	.00036	.00002
.29387	-.07398	-.07340	-.02	.72	-.6099	.00034	.00001
.27733	-.07387	-.07333	-.71	.73	-.6110	.00032	.00001
.26106	-.07357	-.07307	-1.40	.74	-.6116	.00030	.00000
.24509	-.07309	-.07262	-2.09	.77	-.6119	.00028	.00000

```
      X          Y          YS        ANG      KAPPA       CP        THETA      SEP

   .22946    -.07242    -.07199     -2.80       .81     -.6118     .00026   -.00000
   .21418    -.07158    -.07118     -3.53       .87     -.6113     .00024   -.00001
   .19929    -.07056    -.07019     -4.30       .94     -.6101     .00022   -.00001
   .18481    -.06937    -.06903     -5.12      1.03     -.6081     .00020   -.00002
   .17077    -.06800    -.06770     -6.00      1.15     -.6047     .00018   -.00002
   .15719    -.06646    -.06619     -6.95      1.28     -.5984     .00016   -.00004
   .14410    -.06475    -.06451     -7.98      1.45     -.5881     .00014   -.00005
   .13151    -.06286    -.06265     -9.11      1.65     -.5731     .00012   -.00006
   .11945    -.06079    -.06061    -10.35      1.90     -.5524     .00010   -.00007
   .10792    -.05855    -.05840    -11.72      2.18     -.5249     .00009   -.00009
   .09694    -.05612    -.05601    -13.23      2.51     -.4895     .00007   -.00010
   .08653    -.05352    -.05343    -14.89      2.90     -.4456     .00005   -.00008
   .07669    -.05074    -.05068    -16.72      3.38     -.3943     .00002   -.00004
   .06743    -.04778    -.04775    -18.75      3.91     -.3340
   .05876    -.04466    -.04464    -20.97      4.54     -.2630     TRANSITION
   .05068    -.04137    -.04136    -23.43      5.35     -.1819
   .04321    -.03792    -.03792    -26.18      6.35     -.0903
   .03634    -.03432    -.03432    -29.27      7.60      .0131
   .03009    -.03057    -.03057    -32.75      9.15      .1295
   .02446    -.02667    -.02667    -36.69     10.93      .2596
   .01943    -.02263    -.02263    -41.07     12.83      .4031
   .01502    -.01845    -.01845    -45.88     14.75      .5559
   .01119    -.01414    -.01414    -51.03     16.37      .7102
   .00794    -.00973    -.00973    -56.40     17.71      .8550
   .00524    -.00522    -.00522    -61.93     19.03      .9793
   .00310    -.00067    -.00067    -67.66     20.93     1.0732
   .00150     .00389     .00389    -73.83     23.69     1.1273
   .00046     .00842     .00842    -80.54     26.57     1.1313     STAGNATION
  0.00000     .01290     .01290    -87.75     29.16     1.0746
   .00012     .01731     .01731    -95.36     30.68      .9509
   .00082     .02162     .02162   -103.11     30.89      .7604
   .00210     .02581     .02581   -110.89     31.06      .5068
   .00397     .02983     .02983   -118.95     32.61      .1929
   .00647     .03362     .03362   -127.62     33.20     -.1595
   .00966     .03716     .03716   -136.18     28.14     -.4671
   .01361     .04049     .04049   -143.27     20.01     -.6753
   .01833     .04366     .04366   -148.75     14.31     -.8177
   .02379     .04667     .04667   -153.20     10.71     -.9180
   .02999     .04955     .04955   -156.83      8.03     -.9949
   .03692     .05230     .05230   -159.84      6.18    -1.0537
   .04457     .05491     .05491   -162.36      4.80    -1.0981
   .05293     .05739     .05738   -164.49      3.77    -1.1309     TRANSITION
   .06199     .05975     .05973   -166.28      2.97    -1.1535
   .07174     .06199     .06195   -167.80      2.34    -1.1666    0.00000   0.00000
   .08217     .06412     .06405   -169.07      1.86    -1.1715     .00002    .00000
   .09327     .06615     .06604   -170.15      1.51    -1.1702     .00006    .00001
   .10499     .06809     .06794   -171.09      1.25    -1.1648     .00008    .00001
   .11733     .06993     .06974   -171.91      1.06    -1.1568     .00010    .00002
   .13025     .07168     .07145   -172.65       .92    -1.1474     .00012    .00003
   .14374     .07334     .07307   -173.32       .81    -1.1371     .00014    .00003
   .15775     .07490     .07459   -173.93       .72    -1.1265     .00016    .00004
   .17228     .07637     .07603   -174.50       .65    -1.1158     .00018    .00004
   .18729     .07775     .07736   -175.03       .59    -1.1051     .00020    .00004
   .20276     .07902     .07860   -175.54       .54    -1.0944     .00022    .00005
   .21866     .08019     .07973   -176.02       .51    -1.0839     .00025    .00005
```

X	Y	YS	ANG	KAPPA	CP	THETA	SEP
.23495	.08126	.08076	-176.48	.48	-1.0735	.00027	.00005
.25163	.08222	.08168	-176.92	.45	-1.0632	.00029	.00006
.26865	.08308	.08249	-177.35	.43	-1.0528	.00032	.00006
.28598	.08381	.08319	-177.77	.41	-1.0424	.00034	.00006
.30361	.08444	.08377	-178.18	.40	-1.0318	.00036	.00007
.32150	.08494	.08423	-178.59	.39	-1.0209	.00039	.00008
.33961	.08532	.08457	-179.00	.39	-1.0095	.00041	.00009
.35793	.08558	.08478	-179.41	.39	-.9974	.00044	.00010
.37642	.08570	.08486	-179.82	.39	-.9842	.00046	.00011
.39505	.08569	.08480	-180.24	.40	-.9697	.00049	.00013
.41380	.08554	.08460	-180.68	.42	-.9532	.00052	.00016
.43262	.08524	.08424	-181.14	.44	-.9343	.00055	.00020
.45150	.08478	.08373	-181.63	.46	-.9119	.00058	.00025
.47042	.08416	.08304	-182.15	.49	-.8853	.00061	.00032
.48934	.08336	.08217	-182.70	.52	-.8537	.00065	.00039
.50825	.08237	.08111	-183.27	.54	-.8170	.00069	.00048
.52714	.08119	.07985	-183.86	.55	-.7755	.00073	.00058
.54599	.07982	.07838	-184.46	.55	-.7296	.00078	.00068
.56481	.07826	.07672	-185.04	.52	-.6802	.00084	.00078
.58359	.07651	.07487	-185.58	.47	-.6286	.00090	.00087
.60232	.07461	.07285	-186.05	.40	-.5769	.00097	.00093
.62098	.07256	.07070	-186.45	.34	-.5273	.00103	.00097
.63956	.07041	.06844	-186.77	.28	-.4812	.00110	.00098
.65802	.06818	.06610	-187.04	.22	-.4389	.00117	.00097
.67634	.06588	.06371	-187.25	.18	-.4009	.00124	.00095
.69447	.06355	.06128	-187.41	.14	-.3667	.00131	.00093
.71239	.06120	.05884	-187.54	.11	-.3361	.00137	.00090
.73005	.05885	.05640	-187.64	.09	-.3085	.00143	.00087
.74742	.05650	.05397	-187.73	.08	-.2837	.00149	.00085
.76446	.05418	.05157	-187.80	.07	-.2613	.00155	.00083
.78113	.05188	.04920	-187.87	.07	-.2408	.00160	.00081
.79740	.04963	.04686	-187.93	.07	-.2220	.00165	.00080
.81323	.04741	.04458	-187.99	.07	-.2046	.00171	.00079
.82860	.04525	.04233	-188.05	.07	-.1884	.00176	.00079
.84346	.04313	.04015	-188.12	.08	-.1734	.00180	.00080
.85780	.04108	.03802	-188.19	.09	-.1592	.00185	.00081
.87158	.03909	.03595	-188.26	.10	-.1458	.00190	.00083
.88478	.03716	.03394	-188.35	.12	-.1329	.00194	.00087
.89736	.03530	.03200	-188.44	.14	-.1205	.00198	.00091
.90932	.03352	.03013	-188.54	.16	-.1084	.00203	.00097
.92061	.03181	.02832	-188.66	.19	-.0966	.00207	.00104
.93123	.03018	.02657	-188.79	.22	-.0849	.00211	.00114
.94116	.02864	.02488	-188.93	.27	-.0732	.00216	.00124
.95036	.02718	.02323	-189.09	.35	-.0616	.00220	.00138
.95882	.02581	.02156	-189.29	.46	-.0503	.00225	.00158
.96654	.02453	.01984	-189.52	.55	-.0388	.00229	.00189
.97349	.02335	.01808	-189.75	.56	-.0267	.00234	.00233
.97967	.02226	.01633	-189.95	.64	-.0133	.00239	.00285
.98504	.02133	.01467	-190.21	1.08	.0007	.00246	.00331
.98960	.02049	.01320	-190.62	2.26	.0139	.00254	.00360
.99334	.01977	.01216	-191.33	4.43	.0250	.00261	.00369
.99624	.01916	.01150	-192.38	8.72	.0332	.00266	.00361
.99832	.01868	.01100	-193.79	14.56	.0388	.00270	.00349
.99958	.01836	.01062	-195.58	52.41	.0421	.00272	.00338
1.00000	.01824	.01041	-197.75	155.89	.0431	.00272	.00335



M=.710 CL= .799 DY=.020 T/C=.144


M=.710
CL= .799
DY=.020
T/C=.144
-1.2
-.8
-.4
0.0
.4
.8
1.2
-.4
0.0
.4

```
                                                       07/23/74



                               RUN=  -12

              CIRCULATORY FLOW ABOUT A TRANSONIC AIRFOIL

              M= .710   CL= .799   UY= .020   T/C= .144


                             TAPE 6, PATH 0

                           2  0
                            -.800   0.000   2
                           -1.000   0.000   2

                           2  0
                             .300   -.050   2
                             .455   -.380   2


                                TAPE 7

-5-012   4  -.12  .15  .08  1.40  .710  .004  -.152  .050 1.50   7
22   1   2   5   6   9  10  14  33  34  38  41  42  45  46  49
50  53  54  57  58  61  62

-.185  -.045   .690   .015        .034  -.220   .600   .300
 .024   .094   .635  -.240       0.000  -.184  -.030   .650

0.000  0.000 -2.000  0.000       0.000  0.000 -2.000  0.000
0.000  0.000 -2.000  0.000       0.000  0.000 -2.000  0.000

-.118   .347   .261  -.233       0.000   .109   .065  -.380
 .051   .104  0.000  -.350       -.103   .067   .420  -.425

 .085   .020   .330   .055       -.033   .239   .200   .050
 .117  -.050   .116   .066        .045  -.020  2.000  1.000
```

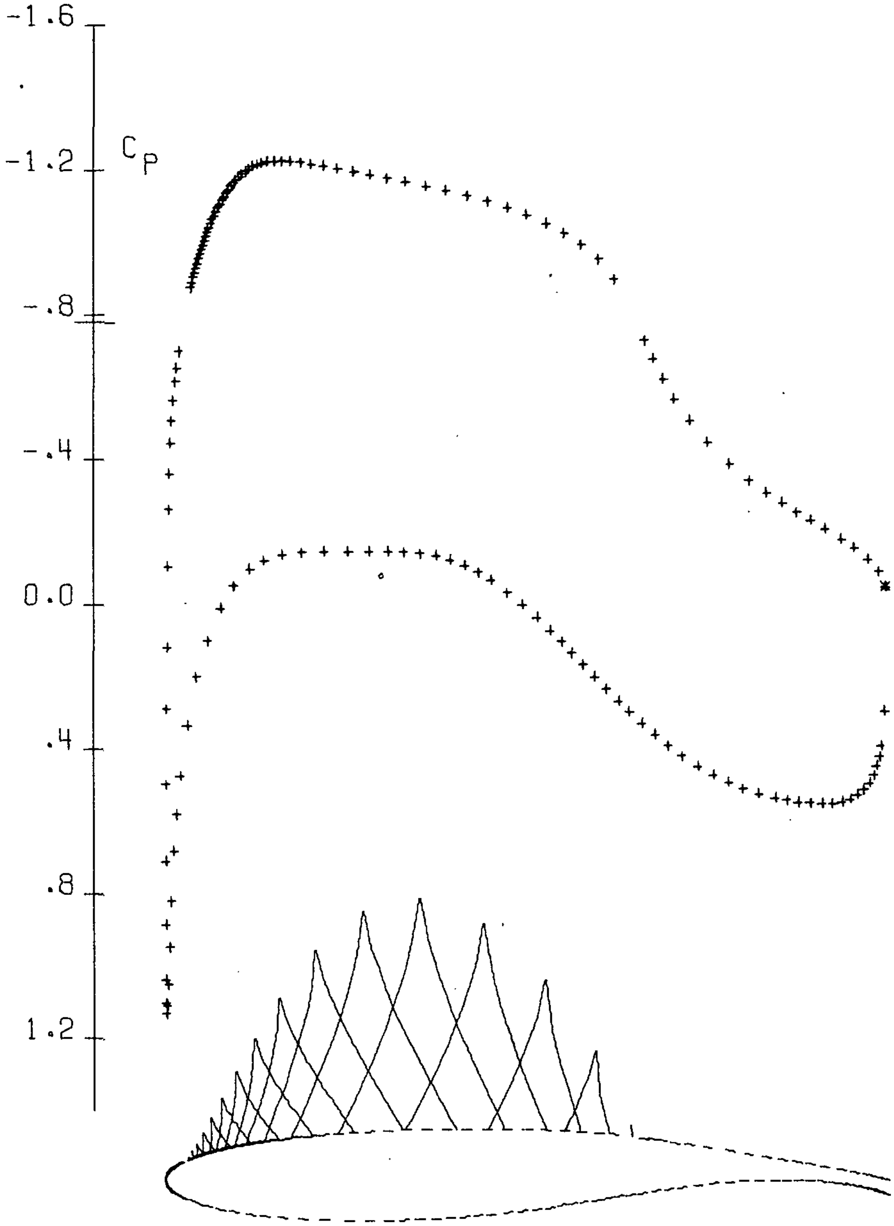
```
      AUTOMATION PATHS

5  0
-1.005     -.335  -1
 -.920     -.405   2
 -.815     -.492   2
 -.715     -.555   3
 -.644     -.563   4

7  0
 -.128      .339  -1
 -.210      .355   2
 -.410      .385   1
 -.620      .373   1
 -.740      .341   1
 -.820      .293   1
 -.880      .230   1

5  0
 -.098      .335  -1
 0.000      .324   2
  .102      .322   2
  .200      .325   1
  .280      .320   1

5  0
  .280      .320  -1
  .400      .224   3
  .490     -.035   1
  .515     -.185   3
  .480     -.330   3

6  0
 -.100     -.225   1
  .190     -.390  -1
  .242     -.335   2
  .295     -.300   2
  .360     -.315   2
  .420     -.360   2
```

LISTING OF COORDINATES FOR AIRFOIL 71-08-14 RN=20.0 MILLION

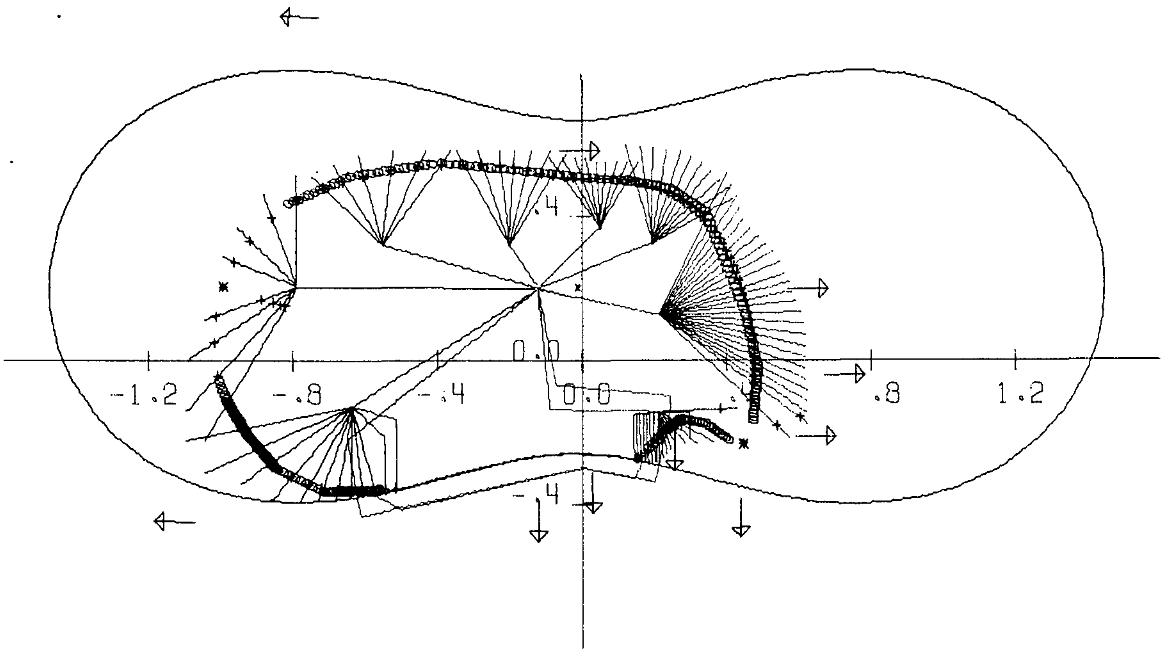
X	Y	YS	ANG	KAPPA	CP	THETA	SEP
1.00000	0.00000	.00233	-20.19	-96.66	.0025	.00177	-.10007
.99955	.00015	.00260	-18.71	-33.88	.0614	.00179	-.09611
.99818	.00059	.00338	-17.34	-10.69	.1971	.00192	-.08420
.99581	.00130	.00443	-16.10	-7.06	.3404	.00231	-.06473
.99242	.00224	.00557	-14.97	-4.62	.4187	.00293	-.04097
.98796	.00338	.00693	-13.89	-3.75	.4422	.00324	-.02085
.98240	.00469	.00851	-12.73	-3.39	.4587	.00318	-.00938
.97571	.00611	.01015	-11.46	-3.09	.4856	.00329	-.00610
.96789	.00760	.01178	-10.13	-2.77	.5101	.00358	-.00748
.95891	.00908	.01342	-8.75	-2.52	.5259	.00376	-.00660
.94881	.01051	.01503	-7.34	-2.33	.5377	.00381	-.00353
.93759	.01180	.01654	-5.89	-2.16	.5470	.00387	-.00176
.92530	.01290	.01788	-4.41	-2.00	.5526	.00393	-.00123
.91198	.01375	.01900	-2.94	-1.85	.5547	.00394	-.00044
.89770	.01430	.01981	-1.48	-1.71	.5534	.00391	.00058
.88252	.01450	.02024	-.06	-1.56	.5489	.00385	.00130
.86653	.01432	.02023	1.30	-1.42	.5413	.00375	.00175
.84981	.01374	.01975	2.60	-1.29	.5309	.00362	.00211
.83245	.01277	.01877	3.83	-1.17	.5178	.00347	.00238
.81451	.01138	.01730	4.98	-1.06	.5021	.00330	.00256
.79610	.00960	.01534	6.06	-.97	.4838	.00312	.00266
.77728	.00744	.01294	7.06	-.88	.4629	.00292	.00272
.75813	.00491	.01012	7.98	-.79	.4389	.00272	.00276
.73875	.00204	.00692	8.82	-.70	.4116	.00252	.00277
.71919	-.00112	.00338	9.57	-.60	.3805	.00231	.00275
.69954	-.00455	-.00043	10.20	-.49	.3452	.00210	.00269
.67987	-.00817	-.00445	10.69	-.36	.3056	.00190	.00259
.66022	-.01194	-.00860	11.02	-.22	.2616	.00171	.00245
.64066	-.01578	-.01281	11.18	-.07	.2139	.00153	.00227
.62120	-.01962	-.01700	11.17	.09	.1631	.00137	.00206
.60186	-.02341	-.02110	10.99	.23	.1103	.00123	.00183
.58265	-.02708	-.02504	10.66	.36	.0567	.00111	.00160
.56358	-.03059	-.02878	10.19	.48	.0035	.00100	.00138
.54463	-.03390	-.03229	9.61	.57	-.0481	.00091	.00116
.52580	-.03698	-.03554	8.95	.64	-.0969	.00083	.00097
.50707	-.03981	-.03850	8.23	.69	-.1419	.00076	.00079
.48845	-.04236	-.04119	7.46	.72	-.1823	.00070	.00064
.46992	-.04467	-.04359	6.68	.73	-.2176	.00065	.00051
.45148	-.04671	-.04570	5.90	.73	-.2476	.00061	.00040
.43314	-.04848	-.04755	5.14	.72	-.2726	.00057	.00030
.41491	-.05000	-.04913	4.39	.70	-.2927	.00053	.00023
.39679	-.05126	-.05046	3.67	.68	-.3088	.00050	.00017
.37880	-.05233	-.05156	2.99	.66	-.3215	.00047	.00013
.36096	-.05315	-.05243	2.32	.64	-.3313	.00045	.00009
.34329	-.05377	-.05309	1.68	.63	-.3386	.00042	.00007
.32581	-.05419	-.05355	1.06	.62	-.3441	.00040	.00005
.30855	-.05442	-.05382	.45	.61	-.3482	.00037	.00003
.29153	-.05447	-.05390	-.14	.61	-.3510	.00035	.00002
.27478	-.05434	-.05381	-.73	.62	-.3530	.00033	.00001
.25833	-.05404	-.05355	-1.33	.64	-.3543	.00031	.00001
.24221	-.05359	-.05313	-1.93	.67	-.3551	.00029	.00000

X	Y	YS	ANG	KAPPA	CP	THETA	SEP
.22643	-.05297	-.05254	-2.55	.71	-.3555	.00026	.00000
.21104	-.05220	-.05180	-3.20	.76	-.3553	.00024	-.00000
.19605	-.05127	-.05091	-3.88	.82	-.3547	.00022	-.00001
.18150	-.05020	-.04987	-4.59	.90	-.3529	.00020	-.00001
.16740	-.04897	-.04867	-5.37	1.01	-.3506	.00018	-.00002
.15378	-.04759	-.04732	-6.21	1.14	-.3470	.00016	-.00002
.14066	-.04606	-.04582	-7.14	1.31	-.3414	.00014	-.00003
.12807	-.04437	-.04416	-8.17	1.52	-.3329	.00012	-.00004
.11603	-.04252	-.04234	-9.31	1.77	-.3202	.00010	-.00005
.10454	-.04051	-.04036	-10.59	2.07	-.3018	.00009	-.00007
.09361	-.03833	-.03821	-12.01	2.41	-.2763	.00007	-.00008
.08327	-.03598	-.03589	-13.60	2.81	-.2422	.00005	-.00008
.07351	-.03346	-.03341	-15.35	3.26	-.1986	.00002	-.00004
.06434	-.03079	-.03075	-17.27	3.77	-.1446		
.05576	-.02795	-.02793	-19.38	4.43	-.0808	TRANSITION	
.04778	-.02496	-.02496	-21.72	5.13	-.0072		
.04039	-.02183	-.02183	-24.26	5.99	.0779		
.03360	-.01858	-.01858	-27.08	7.14	.1728		
.02743	-.01521	-.01521	-30.23	8.57	.2780		
.02187	-.01175	-.01175	-33.79	10.49	.3932		
.01692	-.00819	-.00819	-37.86	12.98	.5182		
.01261	-.00455	-.00455	-42.55	16.21	.6517		
.00893	-.00085	-.00085	-48.00	20.37	.7897		
.00589	.00291	.00291	-54.29	24.99	.9239		
.00347	.00673	.00673	-61.37	29.86	1.0392		
.00169	.01058	.01058	-69.25	34.63	1.1147		
.00054	.01446	.01446	-77.75	38.24	1.1293	STAGNATION	
0.00000	.01835	.01835	-86.60	39.85	1.0670		
.00007	.02223	.02223	-95.50	39.90	.9230		
.00074	.02606	.02606	-104.37	39.41	.7023		
.00202	.02980	.02980	-113.29	39.24	.4182		
.00394	.03341	.03341	-122.37	37.33	.1028		
.00653	.03688	.03688	-130.93	30.70	-.1788		
.00985	.04023	.04023	-138.04	22.26	-.3888		
.01387	.04349	.04349	-143.65	16.11	-.5459		
.01858	.04667	.04667	-148.20	12.09	-.6748		
.02396	.04975	.04976	-151.98	9.42	-.7887		
.03000	.05274	.05274	-155.28	7.80	-.8923		
.03666	.05559	.05559	-158.27	6.70	-.9944		
.04396	.05828	.05828	-161.13	6.31	-1.1033		
.05190	.06076	.06074	-164.10	5.97	-1.2318		
.06056	.06301	.06297	-166.73	3.95	-1.3356	TRANSITION	
.07000	.06509	.06503	-168.27	2.06	-1.3594		
.08018	.06710	.06700	-169.36	1.62	-1.3568	.00005	.00001
.09104	.06905	.06891	-170.27	1.29	-1.3455	.00007	.00002
.10255	.07094	.07076	-171.07	1.10	-1.3307	.00009	.00003
.11466	.07277	.07254	-171.78	.96	-1.3144	.00011	.00004
.12736	.07452	.07426	-172.44	.85	-1.2975	.00013	.00005
.14061	.07621	.07591	-173.06	.76	-1.2805	.00015	.00006
.15438	.07781	.07748	-173.63	.69	-1.2637	.00017	.00006
.16866	.07934	.07896	-174.17	.63	-1.2472	.00019	.00006
.18342	.08078	.08036	-174.68	.58	-1.2310	.00021	.00007
.19862	.08213	.08167	-175.18	.54	-1.2152	.00024	.00007
.21426	.08338	.08288	-175.65	.51	-1.1998	.00026	.00008
.23028	.08454	.08400	-176.10	.48	-1.1847	.00028	.00008

X	Y	YS	ANG	KAPPA	CP	THETA	SEP
.24669	.08559	.08501	-176.55	.46	-1.1698	.00031	.00008
.26343	.08653	.08591	-176.98	.44	-1.1550	.00033	.00009
.28049	.08737	.08671	-177.41	.43	-1.1403	.00035	.00009
.29785	.08809	.08739	-177.83	.42	-1.1255	.00038	.00010
.31546	.08869	.08794	-178.25	.42	-1.1103	.00040	.00011
.33331	.08917	.08838	-178.68	.42	-1.0946	.00043	.00012
.35136	.08952	.08868	-179.11	.42	-1.0781	.00045	.00013
.36959	.08973	.08885	-179.55	.43	-1.0604	.00048	.00015
.38797	.08980	.08887	-180.00	.43	-1.0412	.00051	.00017
.40647	.08973	.08874	-180.47	.44	-1.0206	.00054	.00019
.42507	.08950	.08846	-180.94	.45	-.9985	.00057	.00022
.44376	.08911	.08802	-181.43	.45	-.9753	.00060	.00024
.46249	.08856	.08742	-181.92	.46	-.9510	.00063	.00028
.48125	.08785	.08664	-182.42	.47	-.9246	.00066	.00033
.50002	.08698	.08570	-182.94	.50	-.8950	.00070	.00040
.51877	.08592	.08457	-183.49	.52	-.8611	.00074	.00048
.53749	.08469	.08325	-184.05	.53	-.8220	.00078	.00058
.55616	.08328	.08174	-184.62	.53	-.7779	.00083	.00069
.57477	.08168	.08005	-185.18	.51	-.7302	.00089	.00078
.59333	.07991	.07818	-185.71	.47	-.6807	.00095	.00087
.61181	.07799	.07615	-186.18	.41	-.6312	.00101	.00093
.63020	.07593	.07398	-186.58	.35	-.5835	.00108	.00097
.64849	.07377	.07171	-186.93	.30	-.5386	.00115	.00099
.66665	.07151	.06935	-187.22	.25	-.4969	.00121	.00099
.68464	.06920	.06693	-187.46	.21	-.4588	.00128	.00098
.70243	.06684	.06448	-187.66	.18	-.4242	.00135	.00096
.72000	.06445	.06200	-187.82	.15	-.3928	.00141	.00094
.73731	.06205	.05951	-187.96	.13	-.3644	.00148	.00092
.75431	.05966	.05703	-188.07	.11	-.3386	.00154	.00090
.77098	.05728	.05457	-188.18	.10	-.3151	.00159	.00088
.78729	.05492	.05213	-188.27	.10	-.2937	.00165	.00086
.80319	.05260	.04973	-188.36	.09	-.2740	.00170	.00085
.81865	.05031	.04736	-188.44	.10	-.2558	.00176	.00085
.83366	.04808	.04505	-188.53	.10	-.2388	.00181	.00085
.84816	.04589	.04279	-188.62	.11	-.2228	.00186	.00086
.86215	.04376	.04058	-188.71	.12	-.2079	.00190	.00087
.87558	.04169	.03843	-188.81	.13	-.1938	.00195	.00089
.88843	.03968	.03634	-188.91	.15	-.1804	.00200	.00092
.90069	.03775	.03432	-189.03	.17	-.1676	.00204	.00096
.91232	.03589	.03237	-189.15	.20	-.1551	.00208	.00102
.92330	.03411	.03048	-189.28	.23	-.1430	.00213	.00109
.93362	.03241	.02866	-189.43	.27	-.1311	.00217	.00118
.94325	.03079	.02691	-189.60	.33	-.1192	.00222	.00130
.95218	.02927	.02519	-189.79	.39	-.1074	.00226	.00146
.96039	.02784	.02347	-189.99	.46	-.0956	.00230	.00164
.96786	.02651	.02172	-190.21	.58	-.0838	.00235	.00186
.97458	.02528	.01992	-190.48	.79	-.0720	.00240	.00224
.98053	.02417	.01809	-190.79	.99	-.0600	.00244	.00280
.98571	.02317	.01630	-191.11	1.09	-.0471	.00250	.00342
.99010	.02229	.01466	-191.47	2.14	-.0335	.00258	.00402
.99368	.02154	.01326	-192.19	5.09	-.0211	.00266	.00450
.99645	.02092	.01214	-193.41	11.00	-.0111	.00272	.00488
.99842	.02042	.01126	-195.14	18.95	-.0037	.00277	.00516
.99961	.02009	.01062	-197.40	70.69	.0009	.00280	.00534
1.00000	.01996	.01027	-200.19	212.44	.0025	.00281	.00540



M=.700 CL= .998 DY=.020 T/C=.127



M=.700 CL= .998 DY=.020 T/C=.127

```
                                                                07/23/74



                                RUN= -138

                  CIRCULATORY FLOW ABOUT A TRANSONIC AIRFOIL

                  M= .700    CL= .998    DY= .020    T/C= .127


                               TAPE 6, PATH 0

                            2  0
                             -.800    0.000    2
                            -1.000    0.000    2

                            2  0
                              .300    -.100    2
                              .480    -.410    2


                                TAPE 7

-7-138    4   -.12   .20   .08   1.40   .700   .012   -.202   .040 1.50    7
25    1    2    5    6   10   13   14   18   21   22   33   34   37   38   42
45   46   49   50   53   54   57   58   61   62


-.115    .040    .700   -.040          -.157   -.211    .600    .200
0.000   -.132   -.030    .580           .028    .025    .620   -.210

0.000    .024  -1.100   -.450           .096    .097   -.750    .950
0.000   0.000  -2.000   0.000          0.000   0.000  -2.000   0.000

-.014   -.005    .440   -.410          -.059    .316    .255   -.225
0.000    .168    .030   -.340           .019    .157   -.120   -.420

 .168    .069    .460    .100           .033    .220    .500    .050
 .087   -.035    .065    .055           .027    .013   4.000   1.000
```

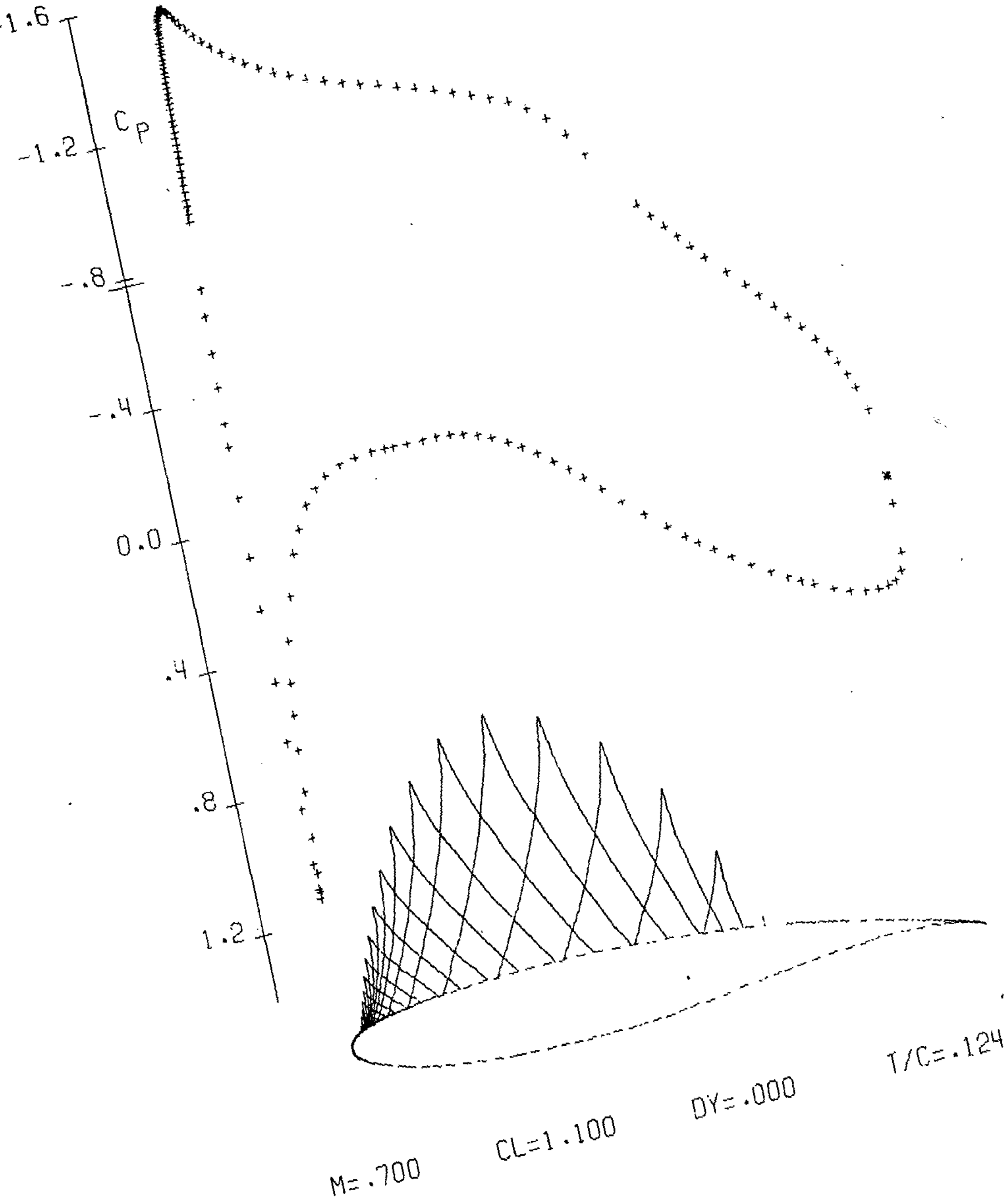
```
        AUTOMATION PATHS

6   0
 -.100      -.280    1
  .210      -.410   -1
  .265      -.360    1
  .310      -.330    2
  .380      -.350    2
  .440      -.400    2

3   0
 -.850      -.500   -1
 -.720      -.560    1
 -.570      -.550    4

4   0
 -.850      -.500   -1
 -.930      -.410    3
 -.990      -.310    3
-1.010      -.260    1

3   0
 -.430       .310   -1
 -.650       .290    1
 -.820       .220    1

2   0
  .200       .240   -1
 -.400       .310    1

5   0
  .200       .240   -1
  .300       .240    1
  .400       .190    1
  .500      -.040    1
  .530      -.200    1

3   0
  .300      -.100    1
  .530      -.200   -1
  .510      -.350    2
```

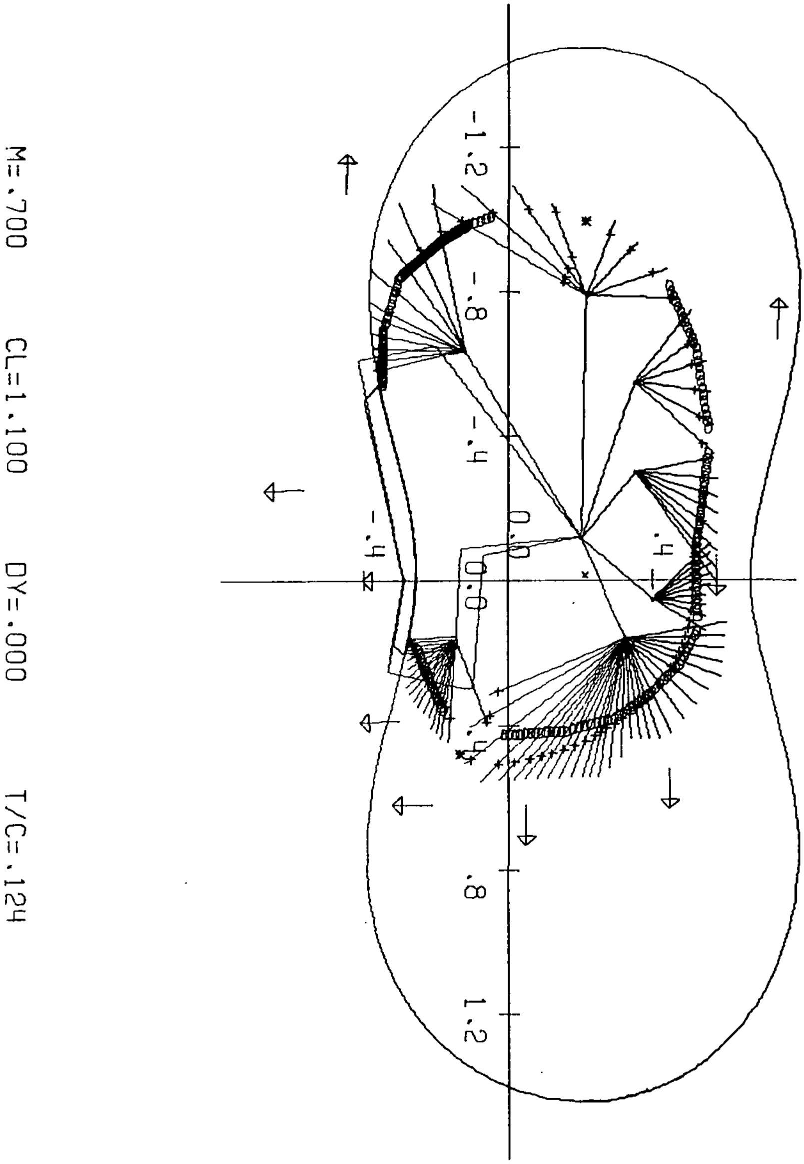
LISTING OF COORDINATES FOR AIRFOIL 70-10-13 RN=20.0 MILLION

X	Y	YS	ANG	KAPPA	CP	THETA	SEP
1.00000	0.00000	.00221	-22.73	-119.58	-.0486	.00166	-.16865
.99957	.00017	.00241	-20.96	-39.71	.0806	.00168	-.15341
.99824	.00065	.00304	-19.54	-10.16	.2962	.00175	-.11253
.99595	.00143	.00399	-18.39	-6.76	.3776	.00191	-.06117
.99265	.00249	.00516	-17.25	-5.18	.4034	.00216	-.02021
.98829	.00378	.00659	-16.00	-4.46	.4431	.00237	-.00405
.98285	.00527	.00825	-14.68	-3.80	.4729	.00251	-.00993
.97630	.00689	.01003	-13.30	-3.34	.4968	.00264	-.01147
.96863	.00860	.01186	-11.89	-2.96	.5155	.00276	-.00502
.95984	.01033	.01372	-10.46	-2.63	.5296	.00285	-.00177
.94995	.01202	.01554	-9.03	-2.34	.5397	.00291	-.00229
.93898	.01362	.01727	-7.63	-2.09	.5463	.00294	-.00198
.92698	.01508	.01886	-6.26	-1.87	.5501	.00296	-.00067
.91398	.01634	.02025	-4.93	-1.68	.5515	.00296	.00006
.90003	.01738	.02139	-3.65	-1.53	.5506	.00294	.00026
.88519	.01816	.02224	-2.40	-1.40	.5479	.00290	.00051
.86952	.01865	.02277	-1.19	-1.30	.5435	.00285	.00084
.85308	.01882	.02295	-.00	-1.22	.5372	.00279	.00108
.83593	.01864	.02275	1.16	-1.14	.5289	.00271	.00126
.81815	.01810	.02216	2.28	-1.07	.5185	.00261	.00144
.79981	.01719	.02117	3.37	-1.00	.5056	.00250	.00162
.78101	.01591	.01977	4.42	-.93	.4898	.00238	.00177
.76181	.01426	.01797	5.40	-.84	.4711	.00225	.00189
.74232	.01227	.01580	6.29	-.75	.4490	.00210	.00196
.72261	.00996	.01328	7.08	-.64	.4235	.00195	.00199
.70276	.00737	.01047	7.75	-.52	.3947	.00180	.00197
.68283	.00456	.00743	8.29	-.40	.3628	.00166	.00191
.66289	.00159	.00421	8.68	-.28	.3281	.00151	.00181
.64297	-.00149	.00089	8.93	-.16	.2911	.00138	.00169
.62311	-.00464	-.00247	9.05	-.04	.2524	.00126	.00154
.60334	-.00778	-.00583	9.02	.08	.2123	.00114	.00139
.58367	-.01088	-.00911	8.87	.18	.1717	.00104	.00124
.56412	-.01388	-.01228	8.61	.28	.1313	.00095	.00109
.54469	-.01676	-.01531	8.24	.36	.0917	.00087	.00094
.52539	-.01948	-.01816	7.80	.44	.0535	.00080	.00081
.50622	-.02202	-.02081	7.28	.50	.0174	.00074	.00068
.48718	-.02435	-.02325	6.70	.55	-.0159	.00068	.00057
.46827	-.02647	-.02545	6.08	.59	-.0459	.00063	.00046
.44949	-.02836	-.02742	5.43	.61	-.0722	.00059	.00037
.43086	-.03003	-.02916	4.77	.63	-.0944	.00055	.00028
.41236	-.03146	-.03065	4.10	.63	-.1124	.00051	.00021
.39401	-.03267	-.03191	3.44	.62	-.1262	.00048	.00015
.37582	-.03366	-.03295	2.80	.60	-.1362	.00045	.00010
.35779	-.03445	-.03378	2.18	.58	-.1427	.00043	.00006
.33995	-.03504	-.03441	1.60	.56	-.1465	.00040	.00003
.32231	-.03545	-.03486	1.05	.53	-.1483	.00038	.00002
.30490	-.03569	-.03514	.53	.51	-.1490	.00035	.00001
.28775	-.03577	-.03526	.03	.51	-.1491	.00033	.00000
.27088	-.03571	-.03523	-.46	.51	-.1491	.00031	.00000
.25432	-.03551	-.03506	-.95	.53	-.1491	.00029	.00000
.23811	-.03517	-.03476	-1.45	.56	-.1492	.00026	-.00000

```
     X        Y        YS       ANG     KAPPA      CP      THETA      SEP

 .22228  -.03470  -.03432    -1.98      .60   -.1491    .00024  -.00000
 .20684  -.03409  -.03375    -2.54      .67   -.1488    .00022  -.00001
 .19184  -.03335  -.03304    -3.14      .74   -.1478    .00020  -.00001
 .17729  -.03247  -.03219    -3.80      .84   -.1456    .00018  -.00002
 .16321  -.03144  -.03119    -4.53      .95   -.1414    .00016  -.00003
 .14962  -.03028  -.03006    -5.32     1.08   -.1347    .00014  -.00004
 .13655  -.02896  -.02877    -6.18     1.22   -.1247    .00012  -.00005
 .12399  -.02750  -.02735    -7.12     1.38   -.1109    .00010  -.00006
 .11198  -.02589  -.02577    -8.14     1.57   -.0925    .00008  -.00007
 .10051  -.02414  -.02405    -9.25     1.77   -.0692    .00006  -.00006
 .08961  -.02225  -.02219   -10.45     2.01   -.0403    .00003  -.00003
 .07927  -.02023  -.02019   -11.74     2.28   -.0055   0.00000  0.00000
 .06951  -.01808  -.01806   -13.13     2.60    .0352
 .06033  -.01581  -.01581   -14.64     2.99    .0821    TRANSITION
 .05175  -.01344  -.01344   -16.28     3.47    .1353
 .04378  -.01098  -.01098   -18.08     4.10    .1950
 .03643  -.00845  -.00845   -20.10     4.99    .2613
 .02971  -.00584  -.00584   -22.40     6.25    .3348
 .02365  -.00318  -.00318   -25.10     8.22    .4165
 .01826  -.00047  -.00047   -28.43    11.30    .5088
 .01357   .00229   .00229   -32.67    16.38    .6148
 .00958   .00510   .00510   -38.19    23.45    .7391
 .00631   .00800   .00800   -45.06    31.51    .8786
 .00371   .01097   .01097   -53.14    39.95   1.0127
 .00179   .01400   .01400   -62.58    53.58   1.1074
 .00056   .01707   .01707   -74.01    63.84   1.1215    STAGNATION
 .00000   .02019   .02019   -85.75    63.80   1.0200
 .00008   .02334   .02334   -97.22    63.50    .7989
 .00081   .02647   .02647  -108.75    58.59    .4872
 .00220   .02957   .02957  -119.15    49.10    .1513
 .00425   .03259   .03259  -128.77    42.62   -.1997
 .00701   .03552   .03552  -137.49    31.13   -.4403
 .01053   .03839   .03839  -143.72    18.69   -.5644
 .01479   .04124   .04124  -148.29    12.94   -.6512
 .01974   .04408   .04408  -151.90     9.48   -.7245
 .02538   .04690   .04690  -154.89     7.17   -.7931
 .03168   .04968   .04968  -157.37     5.61   -.8561
 .03862   .05241   .05241  -159.57     4.70   -.9133
 .04620   .05508   .05508  -161.54     3.91   -.9664
 .05442   .05768   .05767  -163.32     3.31  -1.0154    TRANSITION
 .06325   .06019   .06017  -164.92     2.82  -1.0603
 .07270   .06260   .06256  -166.38     2.42  -1.1005   0.00000  0.00000
 .08275   .06491   .06485  -167.71     2.08  -1.1359    .00002  -.00002
 .09342   .06712   .06702  -168.91     1.78  -1.1661    .00005  -.00003
 .10467   .06921   .06907  -169.98     1.50  -1.1902    .00008  -.00004
 .11652   .07120   .07102  -170.93     1.26  -1.2077    .00009  -.00003
 .12894   .07309   .07288  -171.76     1.05  -1.2187    .00011  -.00002
 .14192   .07488   .07463  -172.49      .90  -1.2243    .00013  -.00001
 .15543   .07658   .07630  -173.14      .78  -1.2261    .00015  -.00000
 .16946   .07820   .07788  -173.73      .68  -1.2252    .00017   .00001
 .18397   .07972   .07936  -174.27      .61  -1.2225    .00019   .00001
 .19894   .08116   .08076  -174.77      .56  -1.2186    .00021   .00002
 .21435   .08251   .08207  -175.24      .51  -1.2138    .00023   .00002
 .23017   .08376   .08329  -175.69      .48  -1.2084    .00025   .00003
 .24637   .08492   .08441  -176.12      .45  -1.2025    .00027   .00003
```

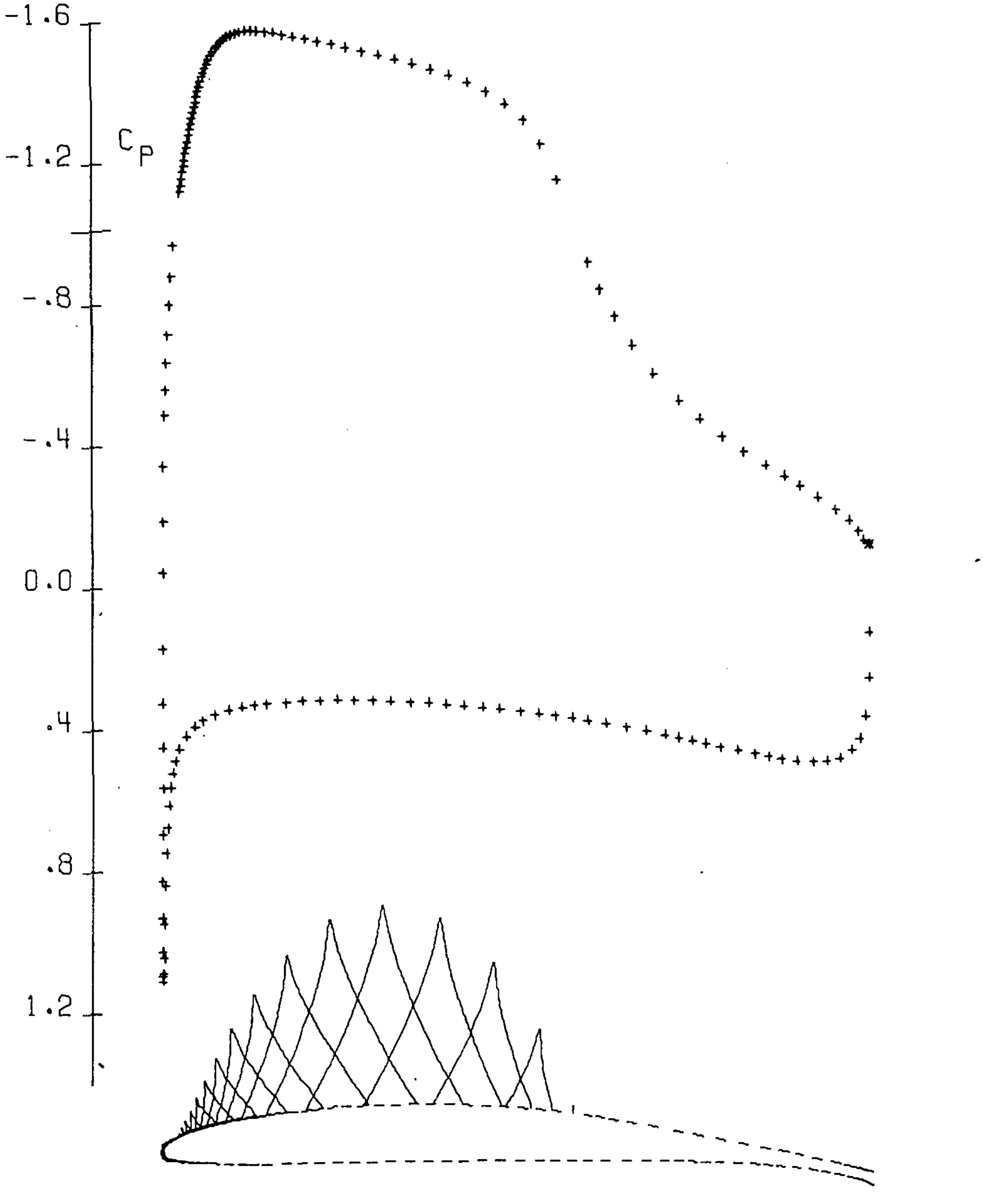
X	Y	YS	ANG	KAPPA	CP	THETA	SEP
.26292	.08598	.08543	-176.53	.42	-1.1964	.00029	.00003
.27981	.08695	.08636	-176.93	.40	-1.1900	.00032	.00004
.29699	.08781	.08718	-177.32	.39	-1.1833	.00034	.00004
.31445	.08857	.08790	-177.70	.38	-1.1764	.00036	.00004
.33215	.08922	.08851	-178.08	.37	-1.1693	.00038	.00005
.35007	.08976	.08901	-178.45	.36	-1.1619	.00041	.00005
.36818	.09019	.08940	-178.83	.36	-1.1542	.00043	.00006
.38644	.09051	.08967	-179.21	.36	-1.1461	.00045	.00006
.40483	.09070	.08982	-179.59	.37	-1.1374	.00048	.00007
.42332	.09077	.08985	-179.98	.37	-1.1281	.00050	.00008
.44189	.09071	.08975	-180.38	.38	-1.1179	.00053	.00009
.46049	.09052	.08951	-180.79	.39	-1.1067	.00055	.00011
.47911	.09019	.08914	-181.22	.41	-1.0941	.00058	.00013
.49772	.08972	.08862	-181.67	.43	-1.0800	.00060	.00015
.51629	.08910	.08795	-182.14	.45	-1.0641	.00063	.00018
.53480	.08833	.08713	-182.63	.47	-1.0463	.00066	.00021
.55322	.08741	.08614	-183.14	.50	-1.0264	.00069	.00026
.57153	.08632	.08498	-183.67	.53	-1.0033	.00072	.00034
.58970	.08506	.08363	-184.25	.58	-.9752	.00076	.00048
.60771	.08362	.08208	-184.90	.66	-.9379	.00080	.00068
.62555	.08198	.08032	-185.62	.73	-.8864	.00085	.00093
.64322	.08012	.07832	-186.38	.74	-.8193	.00092	.00117
.66074	.07805	.07610	-187.09	.65	-.7446	.00100	.00135
.67813	.07580	.07370	-187.68	.51	-.6726	.00109	.00144
.69538	.07340	.07117	-188.13	.39	-.6078	.00118	.00144
.71247	.07091	.06855	-188.46	.29	-.5519	.00127	.00139
.72938	.06836	.06588	-188.71	.22	-.5036	.00135	.00132
.74606	.06577	.06319	-188.90	.17	-.4623	.00143	.00124
.76248	.06318	.06050	-189.05	.14	-.4266	.00150	.00117
.77859	.06060	.05783	-189.17	.12	-.3957	.00157	.00111
.79435	.05804	.05518	-189.28	.11	-.3686	.00163	.00106
.80974	.05551	.05258	-189.38	.11	-.3446	.00169	.00102
.82472	.05302	.05001	-189.47	.11	-.3226	.00175	.00100
.83925	.05059	.04750	-189.56	.11	-.3025	.00180	.00098
.85330	.04821	.04504	-189.66	.13	-.2844	.00186	.00097
.86685	.04589	.04264	-189.76	.14	-.2675	.00191	.00097
.87986	.04364	.04031	-189.87	.15	-.2517	.00196	.00098
.89231	.04146	.03805	-189.99	.17	-.2370	.00200	.00101
.90418	.03936	.03586	-190.12	.20	-.2229	.00205	.00104
.91544	.03733	.03375	-190.26	.23	-.2094	.00209	.00108
.92607	.03539	.03171	-190.42	.27	-.1965	.00214	.00113
.93605	.03354	.02975	-190.59	.32	-.1842	.00218	.00118
.94536	.03179	.02786	-190.78	.38	-.1726	.00222	.00129
.95399	.03013	.02602	-191.00	.49	-.1613	.00226	.00144
.96192	.02857	.02419	-191.25	.62	-.1496	.00231	.00164
.96913	.02712	.02233	-191.54	.70	-.1374	.00236	.00186
.97561	.02576	.02042	-191.80	.69	-.1253	.00240	.00216
.98135	.02457	.01850	-192.05	.89	-.1136	.00244	.00256
.98633	.02349	.01664	-192.42	1.79	-.1020	.00250	.00312
.99054	.02254	.01491	-193.07	3.79	-.0903	.00256	.00387
.99396	.02172	.01342	-194.13	7.03	-.0783	.00263	.00489
.99660	.02102	.01219	-195.62	13.03	-.0671	.00270	.00605
.99849	.02047	.01121	-197.53	21.21	-.0576	.00276	.00711
.99962	.02009	.01049	-199.90	74.32	-.0511	.00280	.00784
1.00000	.01995	.01009	-202.73	220.07	-.0486	.00281	.00811


-1.6
-1.2
-.8
-.4
0.0
.4
.8
1.2
C_P
M=.700
CL=1.100
DY=.000
T/C=.124

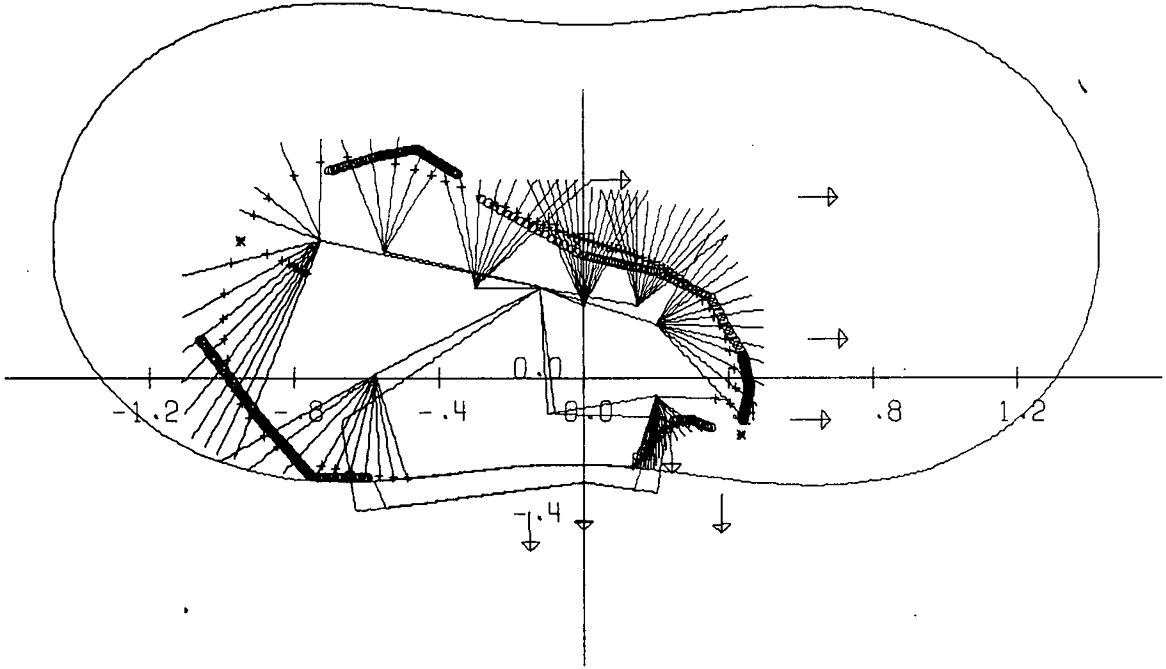

M=.700
CL=1.100
DY=.000
T/C=.124
-1.2
-.8
-.4
0.0
.4
.8
1.2
-.4
0.0
.4
-.4

```
                                                                07/23/74



                               RUN=  -85

                CIRCULATORY FLOW ABOUT A TRANSONIC AIRFOIL

                M= .700    CL=1.100    DY= .000    T/C= .124



                              TAPE 6, PATH 0

                          2  0
                           -.800     0.000    2
                          -1.000     0.000    2

                          2  0
                            .300      .050    2
                            .520     -.320    2



                                TAPE 7

-6-085   4  -.12  .20  .08  1.40  .700  .017  -.210 0.000 1.50    7
23   1   2   5   6  10  18  21  22  33  34  37  38  42  45  46
49  50  53  54  57  58  61  62

-.115   .148   .650   .050         .064  -.273   .550   .450
0.000  -.087  -.040   .580        0.000  0.000   .900  0.000

0.000   .002 -1.100  -.450         .059   .050  -.700   .750
0.000  0.000 -2.000  0.000        0.000  0.000 -2.000  0.000

-.020   .249   .620  -.240        -.612   .258   .390  -.330
0.000   .114  0.000  -.320         .108  -.256  -.250  -.600

 .237  -.106   .460   .100         .137   .049   .500   .050
 .098  -.016   .058   .062         .002   .016  2.000  1.000
```

```
          AUTOMATION PATHS

   3   0
    -.840     -.510   -1
    -.700     -.550    1
    -.550     -.545    4

   4   0
    -.840     -.510   -1
    -.940     -.400    3
    -.990     -.320    3
  -1.010      -.260    1

  -3   0
     .100      .520   -1
    -.100      .520    1
    -.350      .550    1

   3   0
    -.430      .310   -1
    -.650      .290    1
    -.820      .220    1

   6   0
     .200      .260   -1
     .300      .230    1
     .390      .160    1
     .450      .080    1
     .480     -.040    1
     .480     -.200    1

   5   0
    -.100     -.300    1
     .300     -.300    1
     .220     -.420   -1
     .320     -.390    1
     .400     -.360    1
```



M=.650 CL=1.409 DY=.016 T/C=.083



M=.650 CL=1.409 DY=.016 T/C=.083

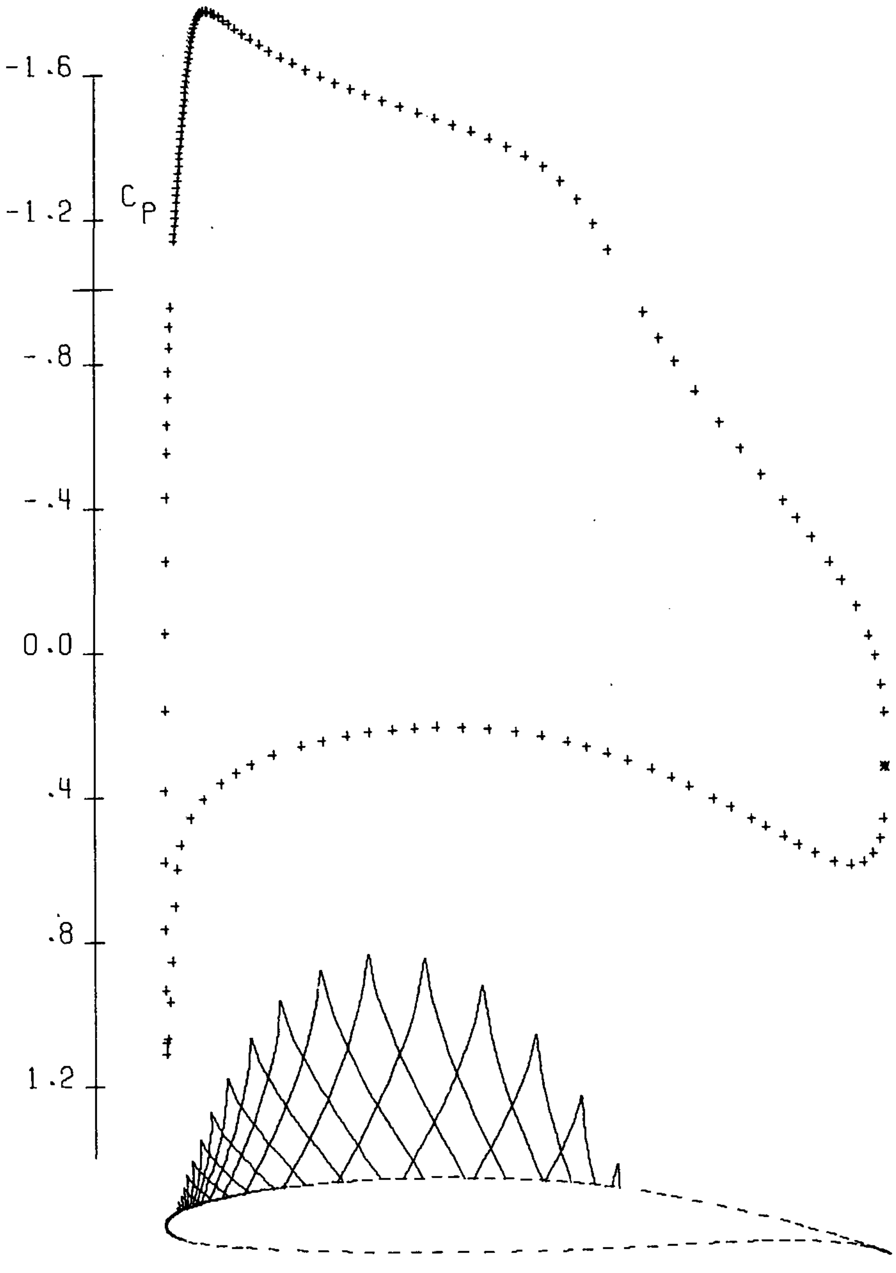
```
                                                        07/24/74




                        RUN=  -41

        CIRCULATORY FLOW ABOUT A TRANSONIC AIRFOIL

        M= .650   CL=1.409   DY= .015   T/C= .083



                     TAPE 6, PATH 0

                    2  0
                     -.800    0.000    2
                    -1.000    0.000    2

                    2  0
                      .400    -.200    2
                      .515    -.450    2



                       TAPE 7

-8-041   4  -.12  .25  .08  1.40  .650  .018  -.371  .023 1.50   7
25   1   2   5   6   9  10  13  14  17  18  33  34  37  38  42
45  46  49  50  53  54  57  58  61  62


-.095   .254   .650   .110        -.246  -.104   .620   .500
 .547  -.844   .050   .550         .025   .001   .600  -.115

 .011   .008  -.900  1.800        0.000  0.000 -2.000  0.000
0.000  0.000 -2.000  0.000        0.000  0.000 -2.000  0.000

-.094   .459   .242  -.180        -.070   .013   .380  -.350
0.000   .137  0.000  -.340         .102   .124  -.150  -.400

 .348  -.164   .460   .100         .116   .021   .500   .050
 .133  -.063   .037   .046         .048  -.004  2.000  1.000
```

```
        AUTOMATION PATHS

  2  0
   -.790    -.610  -1
   -.640    -.600   4

  2  0
   -.790    -.610  -1
   -.940    -.460   4

  2  0
   -.940    -.460  -1
  -1.100    -.260   3

 -4  0
   -.350     .560  -1
   -.460     .630   3
   -.570     .610   2
   -.700     .570   2

 -3  0
    .200     .290  -1
   0.000     .335   2
   -.280     .490   1

 -3  0
    .200     .290  -1
    .350     .220   1
    .440     .060   1

 -3  0
    .440     .060  -1
    .460    -.030   5
    .440    -.120   5

 -5  0
   -.100    -.120   1
    .150    -.220  -1
    .250    -.120   1
    .300    -.120   2
    .350    -.140   2
```

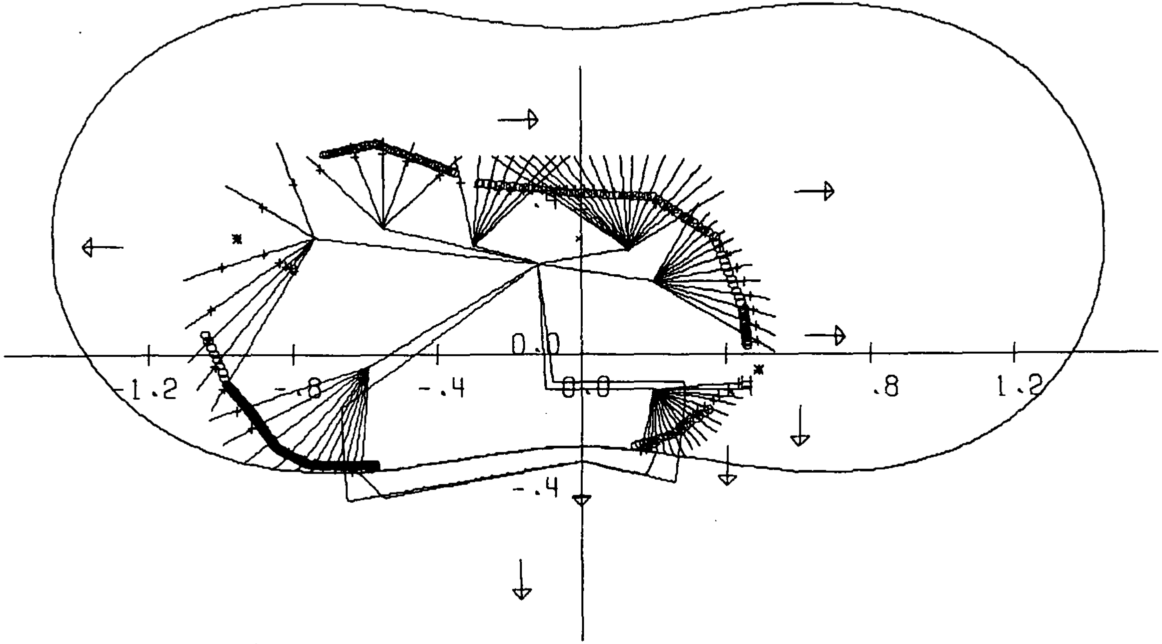
```
LISTING OF COORDINATES FOR  AIRFOIL 65-14-08    RN=20.0 MILLION

      X          Y          YS        ANG      KAPPA        CP       THETA       SEP

1.00000    0.00000     .00118   -26.25  -495.22     -.1263     .00087    -.08100
 .99958     .00016     .00136   -20.91   -61.96      .3460     .00087    -.07545
 .99833     .00066     .00202   -22.43    13.81      .2028     .00092    -.06030
 .99616     .00151     .00300   -20.71   -17.55      .2982     .00109    -.03893
 .99301     .00263     .00419   -18.65    -6.52      .4074     .00137    -.01816
 .98889     .00397     .00561   -17.53    -3.41      .4257     .00148    -.00561
 .98378     .00553     .00727   -16.54    -3.32      .4264     .00141    -.00261
 .97767     .00727     .00909   -15.30    -3.29      .4467     .00142    -.00242
 .97053     .00913     .01101   -14.04    -2.67      .4656     .00152    -.00237
 .96239     .01107     .01301   -12.88    -2.23      .4751     .00157    -.00175
 .95327     .01306     .01506   -11.76    -1.98      .4811     .00155    -.00083
 .94319     .01506     .01710   -10.66    -1.76      .4863     .00155    -.00036
 .93217     .01702     .01910    -9.60    -1.55      .4891     .00156    -.00024
 .92026     .01892     .02102    -8.59    -1.36      .4896     .00155    -.00007
 .90749     .02074     .02285    -7.65    -1.21      .4884     .00153     .00017
 .89391     .02245     .02455    -6.76    -1.07      .4856     .00150     .00031
 .87957     .02404     .02611    -5.93     -.94      .4816     .00147     .00037
 .86451     .02550     .02754    -5.16     -.83      .4763     .00144     .00042
 .84877     .02683     .02882    -4.46     -.73      .4702     .00140     .00046
 .83242     .02801     .02995    -3.83     -.64      .4634     .00136     .00047
 .81549     .02906     .03093    -3.25     -.56      .4559     .00131     .00047
 .79802     .02997     .03178    -2.72     -.49      .4482     .00127     .00045
 .78007     .03075     .03250    -2.26     -.42      .4400     .00122     .00043
 .76167     .03140     .03309    -1.84     -.37      .4318     .00118     .00041
 .74287     .03195     .03357    -1.47     -.32      .4235     .00114     .00038
 .72369     .03238     .03395    -1.14     -.28      .4152     .00109     .00035
 .70419     .03272     .03423     -.85     -.24      .4071     .00105     .00032
 .68440     .03297     .03442     -.60     -.21      .3992     .00101     .00030
 .66435     .03314     .03453     -.38     -.18      .3916     .00097     .00027
 .64407     .03324     .03457     -.18     -.16      .3842     .00093     .00024
 .62361     .03328     .03455     -.01     -.13      .3773     .00089     .00022
 .60299     .03325     .03448      .13     -.12      .3707     .00086     .00019
 .58226     .03318     .03435      .26     -.10      .3644     .00082     .00017
 .56143     .03307     .03419      .37     -.09      .3586     .00078     .00015
 .54055     .03291     .03398      .47     -.08      .3531     .00075     .00013
 .51964     .03273     .03375      .55     -.06      .3481     .00072     .00012
 .49875     .03251     .03349      .63     -.06      .3434     .00068     .00010
 .47789     .03227     .03320      .69     -.05      .3391     .00065     .00009
 .45711     .03201     .03290      .74     -.04      .3352     .00062     .00008
 .43644     .03174     .03258      .78     -.03      .3317     .00059     .00007
 .41591     .03145     .03225      .82     -.03      .3286     .00056     .00006
 .39554     .03116     .03192      .85     -.02      .3258     .00053     .00005
 .37538     .03085     .03157      .87     -.02      .3233     .00050     .00004
 .35545     .03055     .03123      .88     -.01      .3212     .00047     .00003
 .33579     .03024     .03088      .89     -.00      .3195     .00045     .00002
 .31641     .02994     .03054      .89      .00      .3181     .00042     .00002
 .29737     .02964     .03021      .89      .01      .3170     .00039     .00001
 .27868     .02936     .02989      .87      .02      .3163     .00037     .00001
 .26038     .02908     .02957      .85      .03      .3160     .00034     .00000
 .24250     .02882     .02928      .82      .04      .3160     .00031    -.00000
 .22506     .02858     .02900      .78      .05      .3164     .00029    -.00001
```

```
      X          Y         YS        ANG     KAPPA        CP      THETA       SEP

  .20809     .02835     .02874       .72       .06     .3172    .00027   -.00001
  .19162     .02816     .02851       .66       .08     .3185    .00024   -.00002
  .17568     .02799     .02830       .57       .10     .3202    .00022   -.00002
  .16029     .02785     .02813       .47       .13     .3225    .00019   -.00002
  .14549     .02774     .02799       .34       .17     .3253    .00017   -.00003
  .13128     .02767     .02790       .19       .21     .3288    .00015   -.00003
  .11770     .02765     .02784       .01       .27     .3330    .00013   -.00003
  .10477     .02767     .02783      -.22       .34     .3381    .00011   -.00003
  .09251     .02775     .02787      -.49       .44     .3441    .00009   -.00003
  .08094     .02788     .02797      -.81       .57     .3513    .00006   -.00003
  .07008     .02807     .02813     -1.22       .74     .3600    .00003   -.00001
  .05995     .02833     .02836     -1.72      1.00     .3704
  .05057     .02866     .02867     -2.34      1.35     .3830    TRANSITION
  .04195     .02907     .02907     -3.14      1.90     .3983
  .03412     .02956     .02956     -4.16      2.71     .4174
  .02707     .03015     .03015     -5.50      4.06     .4411
  .02084     .03085     .03085     -7.32      6.27     .4717
  .01544     .03166     .03166     -9.90     11.02     .5114
  .01088     .03260     .03260    -13.81     18.89     .5658
  .00721     .03371     .03371    -20.41     47.98     .6465
  .00442     .03507     .03507    -33.10     96.61     .7963
  .00243     .03680     .03680    -49.62    107.12    1.0132
  .00106     .03887     .03987    -63.85     96.73    1.1099    STAGNATION
  .00026     .04117     .04117    -77.48     95.20    1.0261
 0.00000     .04361     .04361    -90.37     86.85     .7646
  .00030     .04611     .04611   -102.93     90.43     .3432
  .00121     .04859     .04859   -116.89     84.63    -.1657
  .00283     .05107     .05107   -128.56     52.07    -.4556
  .00516     .05362     .05362   -136.03     28.11    -.6125
  .00817     .05624     .05624   -141.37     19.21    -.7359
  .01182     .05893     .05893   -145.55     13.66    -.8457
  .01608     .06166     .06166   -149.03     10.52    -.9475
  .02095     .06441     .06441   -152.01      8.26   -1.0413
  .02643     .06715     .06715   -154.63      6.81   -1.1282
  .03249     .06987     .06987   -157.02      5.78   -1.2113
  .03915     .07253     .07253   -159.21      4.91   -1.2890
  .04640     .07514     .07514   -161.21      4.18   -1.3594
  .05424     .07767     .07766   -163.02      3.52   -1.4200    TRANSITION
  .06269     .08011     .08009   -164.65      2.95   -1.4693
  .07173     .08247     .08243   -166.09      2.45   -1.5074   0.00000   0.00000
  .08136     .08474     .08467   -167.36      2.05   -1.5353    .00002   -.00001
  .09158     .08692     .08682   -168.48      1.71   -1.5544    .00005   -.00002
  .10237     .08902     .08889   -169.47      1.44   -1.5661    .00007   -.00002
  .11372     .09104     .09087   -170.35      1.24   -1.5722    .00009   -.00001
  .12561     .09298     .09277   -171.14      1.08   -1.5742    .00011   -.00000
  .13802     .09483     .09458   -171.87       .95   -1.5733    .00012    .00000
  .15092     .09659     .09631   -172.55       .86   -1.5704    .00014    .00001
  .16431     .09827     .09795   -173.18       .78   -1.5662    .00016    .00001
  .17815     .09985     .09950   -173.77       .71   -1.5611    .00018    .00002
  .19243     .10134     .10095   -174.33       .66   -1.5553    .00020    .00002
  .20712     .10272     .10230   -174.88       .62   -1.5491    .00022    .00002
  .22220     .10401     .10354   -175.40       .59   -1.5426    .00024    .00003
  .23764     .10518     .10468   -175.90       .56   -1.5359    .00026    .00003
  .25343     .10624     .10570   -176.40       .53   -1.5289    .00028    .00003
  .26953     .10719     .10661   -176.88       .51   -1.5217    .00030    .00004
```

X	Y	YS	ANG	KAPPA	CP	THETA	SEP
.28592	.10801	.10740	-177.36	.50	-1.5142	.00033	.00004
.30259	.10871	.10806	-177.83	.49	-1.5065	.00035	.00004
.31949	.10928	.10859	-178.30	.48	-1.4984	.00037	.00005
.33661	.10972	.10899	-178.77	.47	-1.4899	.00039	.00005
.35392	.11002	.10925	-179.24	.47	-1.4807	.00041	.00006
.37139	.11018	.10937	-179.71	.47	-1.4709	.00044	.00007
.38899	.11019	.10934	-180.19	.48	-1.4601	.00046	.00008
.40671	.11006	.10917	-180.68	.48	-1.4481	.00048	.00010
.42451	.10977	.10883	-181.18	.50	-1.4346	.00051	.00012
.44237	.10933	.10834	-181.69	.51	-1.4191	.00053	.00014
.46025	.10871	.10767	-182.23	.54	-1.4009	.00056	.00018
.47814	.10793	.10683	-182.81	.58	-1.3789	.00058	.00023
.49599	.10695	.10579	-183.42	.63	-1.3513	.00061	.00033
.51380	.10578	.10453	-184.10	.70	-1.3154	.00065	.00048
.53152	.10439	.10304	-184.87	.82	-1.2660	.00069	.00069
.54917	.10275	.10128	-185.77	.92	-1.1956	.00074	.00094
.56674	.10083	.09922	-186.69	.89	-1.1091	.00081	.00117
.58427	.09864	.09689	-187.54	.79	-1.0163	.00089	.00132
.60179	.09621	.09432	-188.27	.63	-.9267	.00098	.00139
.61930	.09358	.09155	-188.83	.49	-.8490	.00107	.00138
.63678	.09079	.08863	-189.27	.39	-.7806	.00115	.00134
.65420	.08790	.08561	-189.63	.31	-.7216	.00124	.00129
.67153	.08491	.08251	-189.90	.25	-.6704	.00131	.00124
.68874	.08187	.07937	-190.13	.21	-.6247	.00139	.00119
.70578	.07880	.07619	-190.33	.18	-.5843	.00147	.00114
.72262	.07570	.07300	-190.49	.15	-.5486	.00154	.00110
.73924	.07261	.06981	-190.63	.14	-.5165	.00160	.00106
.75558	.06952	.06663	-190.76	.13	-.4874	.00167	.00103
.77161	.06646	.06348	-190.88	.13	-.4612	.00173	.00101
.78731	.06343	.06036	-190.99	.13	-.4374	.00179	.00099
.80264	.06043	.05728	-191.11	.13	-.4154	.00185	.00097
.81757	.05749	.05425	-191.23	.14	-.3950	.00191	.00097
.83206	.05459	.05127	-191.35	.15	-.3760	.00196	.00097
.84611	.05176	.04835	-191.47	.16	-.3581	.00201	.00098
.85967	.04899	.04550	-191.60	.17	-.3413	.00207	.00099
.87272	.04630	.04271	-191.74	.19	-.3254	.00212	.00102
.88524	.04368	.04000	-191.89	.21	-.3103	.00217	.00105
.89720	.04114	.03737	-192.05	.24	-.2958	.00221	.00109
.90859	.03869	.03482	-192.22	.28	-.2818	.00226	.00114
.91938	.03634	.03236	-192.41	.32	-.2682	.00231	.00120
.92956	.03408	.02998	-192.62	.37	-.2551	.00236	.00128
.93911	.03192	.02768	-192.84	.42	-.2424	.00240	.00137
.94801	.02988	.02546	-193.08	.49	-.2299	.00245	.00148
.95624	.02794	.02328	-193.33	.58	-.2178	.00250	.00163
.96380	.02613	.02111	-193.62	.73	-.2057	.00254	.00183
.97067	.02445	.01894	-193.96	.95	-.1938	.00259	.00213
.97683	.02290	.01687	-194.35	1.22	-.1817	.00264	.00261
.98228	.02148	.01500	-194.78	1.38	-.1688	.00269	.00318
.98702	.02022	.01344	-195.20	1.75	-.1547	.00277	.00352
.99102	.01911	.01230	-195.77	3.50	-.1412	.00285	.00339
.99428	.01816	.01177	-196.77	7.24	-.1319	.00290	.00258
.99679	.01737	.01167	-198.31	14.43	-.1272	.00293	.00131
.99857	.01675	.01160	-200.39	24.29	-.1257	.00293	.00008
.99965	.01634	.01145	-203.05	87.84	-.1259	.00293	-.00079
1.00000	.01619	.01129	-206.25	260.57	-.1263	.00293	-.00110



M=.650 CL=1.472 DY=.000 T/C=.104



M=.650 CL=1.472 DY=.000 T/C=.104

```
                                                               08/24/73




                                  RUN= -114

                    CIRCULATORY FLOW ABOUT A TRANSONIC AIRFOIL

                    M= .650   CL=1.472   DY= .000   T/C= .105



                              TAPE 6, PATH 0

                           2  0
                            -.800    0.000    2
                           -1.000    0.000    2

                           2  0
                             .300     .050    2
                             .563    -.320    2



                                 TAPE 7

-7-114   4  -.12  .25  .08  1.40  .650  .006  -.320 0.000 1.50    7
23   1   2   5   6   9  10  17  18  33  34  37  38  42  45  46
49  50  53  54  57  58  61  52


-.102   .061   .650   .050        -.198  -.018   .620   .450
 .434  -.315  -.200   .650        0.000  0.000   .900  0.000

-.008  -.010 -1.300   .300        0.000  0.000 -2.000  0.000
0.000  0.000 -2.000  0.000        0.000  0.000 -2.000  0.000

 .038  -.084   .605  -.175        -.348   .338   .405  -.285
0.000   .120  0.000  -.340         .211  -.116  -.170  -.600

 .353  -.351   .460   .100         .334  -.032   .500   .050
 .081  -.095   .034   .068         .001  -.010  2.000  1.000
```

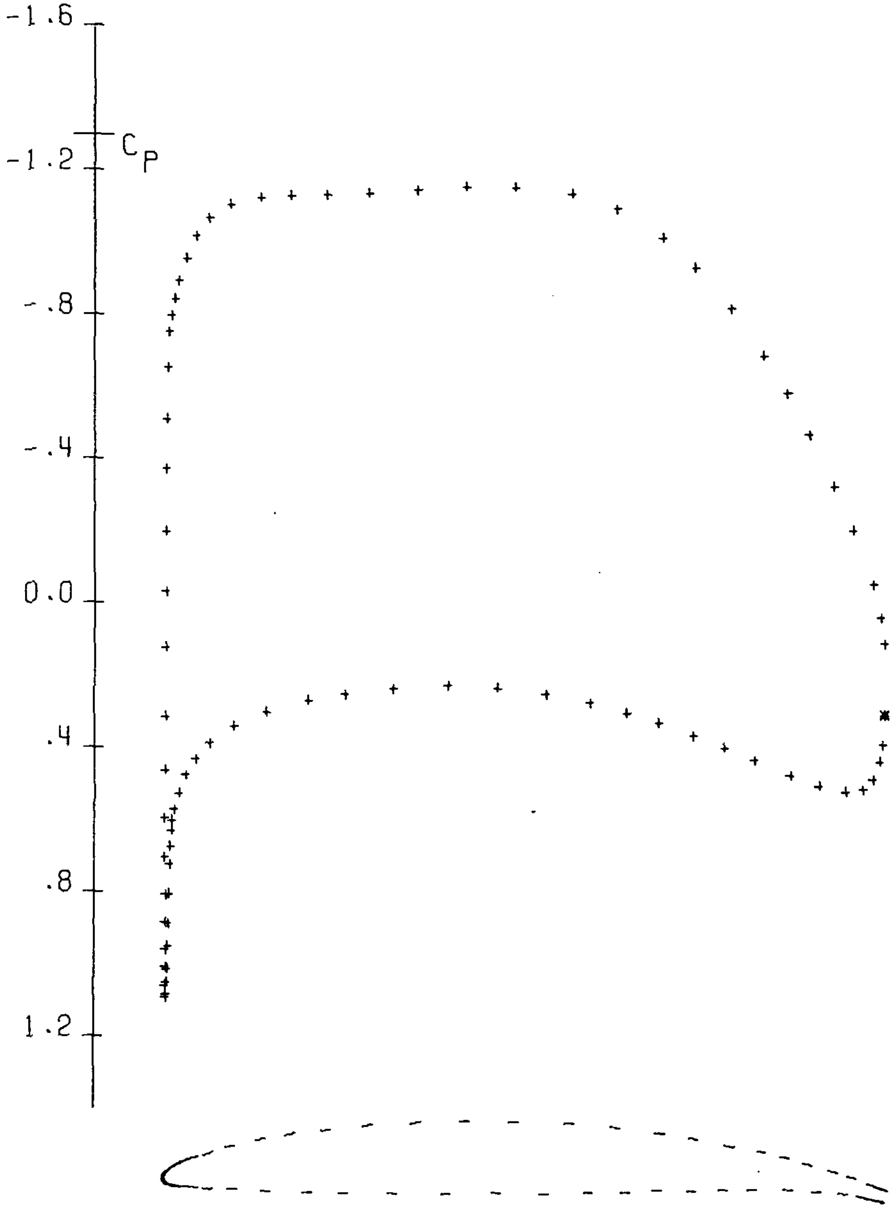
```
        AUTOMATION PATHS

 2   0
  -.790     -.600   -1
  -.610     -.590    4

 3   0
  -.790     -.600   -1
  -.880     -.560    3
  -.950     -.470    3

 3   0
  -.950     -.470   -1
 -1.020     -.390    3
 -1.080     -.260    1

-3   0
   .200      .430   -1
  0.000      .440    1
  -.280      .470    1

-3   0
  -.350      .500   -1
  -.570      .580    2
  -.710      .550    2

-4   0
   .200      .430   -1
   .370      .320    1
   .450      .120    1
   .460      .020    2

-4   0
  -.100     -.100    1
   .350     -.160   -1
   .250     -.230    1
   .150     -.260    1
```

LISTING OF COORDINATES FOR AIRFOIL 65-15-10

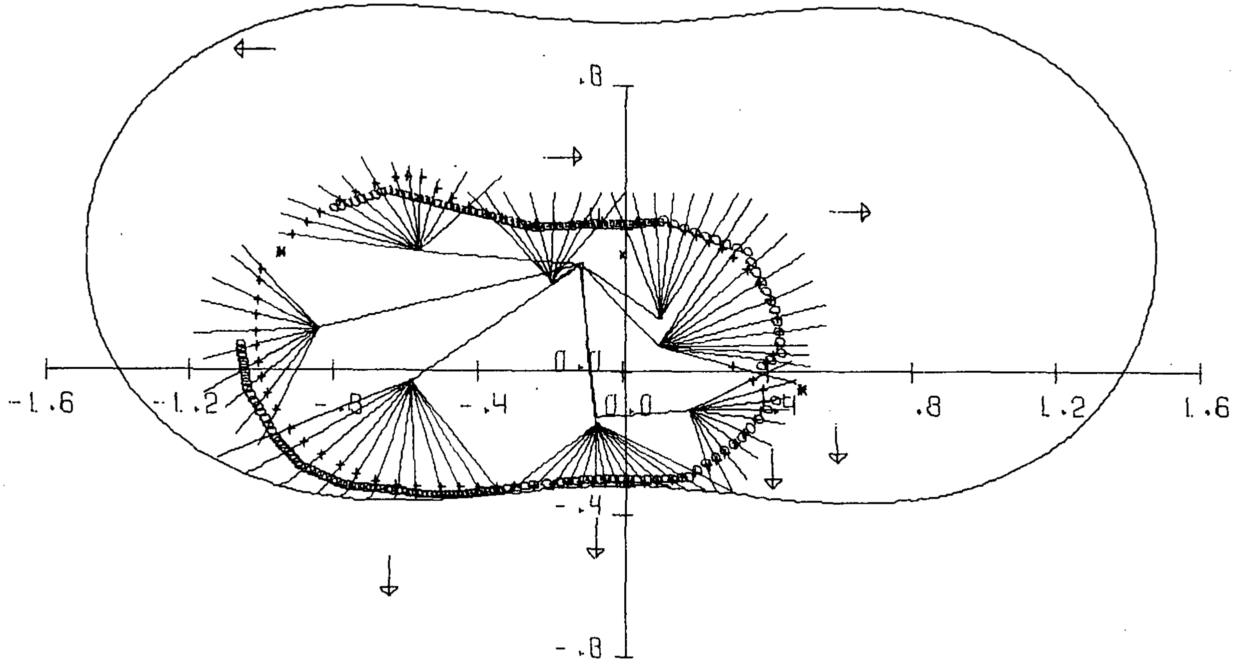
L	X	Y	ANG	KAPPA	MACH	CP
1	1.00000	0.00000	-24.61	-114.55	.5354	.3139
2	.99954	.00019	-22.76	-39.91	.5183	.3584
3	.99812	.00075	-21.08	-12.30	.4851	.4423
4	.99569	.00165	-19.56	-8.34	.4640	.4941
5	.99219	.00283	-18.10	-6.07	.4555	.5145
6	.98759	.00426	-16.56	-5.19	.4464	.5360
7	.98185	.00587	-14.91	-4.49	.4371	.5576
8	.97495	.00759	-13.20	-3.96	.4307	.5724
9	.96691	.00934	-11.44	-3.53	.4268	.5813
10	.95775	.01103	-9.66	-3.12	.4248	.5860
11	.94750	.01261	-7.93	-2.72	.4248	.5861
12	.93625	.01401	-6.29	-2.33	.4267	.5816
13	.92407	.01518	-4.79	-1.97	.4304	.5733
14	.91103	.01612	-3.44	-1.65	.4352	.5620
15	.89721	.01680	-2.25	-1.37	.4411	.5485
16	.88268	.01724	-1.21	-1.13	.4476	.5331
17	.86751	.01744	-.32	-.93	.4547	.5164
18	.85176	.01742	.44	-.76	.4621	.4987
19	.83548	.01720	1.07	-.61	.4696	.4803
20	.81873	.01681	1.60	-.49	.4773	.4615
21	.80154	.01627	2.03	-.38	.4850	.4425
22	.78396	.01559	2.37	-.29	.4927	.4235
23	.76602	.01481	2.63	-.22	.5002	.4045
24	.74777	.01394	2.82	-.14	.5076	.3859
25	.72923	.01301	2.94	-.08	.5147	.3677
26	.71043	.01203	3.00	-.03	.5216	.3500
27	.69140	.01103	3.01	.01	.5281	.3331
28	.67217	.01003	2.98	.05	.5343	.3169
29	.65277	.00903	2.91	.08	.5401	.3016
30	.63321	.00805	2.80	.11	.5455	.2873
31	.61353	.00711	2.67	.13	.5505	.2741
32	.59375	.00622	2.51	.15	.5551	.2619
33	.57389	.00538	2.33	.17	.5592	.2509
34	.55397	.00460	2.14	.18	.5629	.2411
35	.53402	.00389	1.93	.19	.5661	.2324
36	.51406	.00326	1.71	.20	.5689	.2249
37	.49412	.00271	1.48	.20	.5712	.2186
38	.47422	.00223	1.24	.21	.5731	.2135
39	.45438	.00184	1.01	.21	.5746	.2095
40	.43463	.00154	.77	.21	.5757	.2065
41	.41499	.00131	.53	.21	.5764	.2047
42	.39549	.00117	.29	.21	.5767	.2038
43	.37616	.00111	.06	.21	.5766	.2040
44	.35702	.00114	-.18	.21	.5763	.2050
45	.33810	.00123	-.41	.21	.5755	.2069
46	.31942	.00140	-.64	.22	.5745	.2097
47	.30101	.00165	-.87	.22	.5732	.2133
48	.28291	.00196	-1.09	.22	.5716	.2176
49	.26513	.00233	-1.32	.22	.5697	.2227
50	.24770	.00277	-1.55	.23	.5676	.2284
51	.23066	.00326	-1.77	.24	.5652	.2349
52	.21403	.00381	-2.01	.25	.5625	.2421
53	.19783	.00441	-2.24	.26	.5596	.2500

```
  L      X          Y        ANG     KAPPA     MACH       CP

 54   .18210     .00506    -2.49      .28     .5563     .2586
 55   .16687     .00575    -2.74      .31     .5529     .2679
 56   .15215     .00649    -3.02      .34     .5490     .2780
 57   .13797     .00727    -3.31      .38     .5449     .2890
 58   .12437     .00810    -3.62      .43     .5404     .3008
 59   .11136     .00896    -3.97      .50     .5356     .3135
 60   .09898     .00986    -4.36      .62     .5303     .3274
 61   .08724     .01080    -4.82      .74     .5244     .3426
 62   .07618     .01178    -5.33      .85     .5182     .3588
 63   .06581     .01280    -5.90     1.14     .5113     .3763
 64   .05617     .01386    -6.69     1.74     .5035     .3962
 65   .04728     .01499    -7.76     2.45     .4944     .4191
 66   .03915     .01618    -9.01     2.70     .4847     .4435
 67   .03181     .01743   -10.37     4.24     .4747     .4680
 68   .02532     .01875   -12.83     9.51     .4613     .5004
 69   .01972     .02025   -17.35    18.46     .4407     .5493
 70   .01494     .02202   -23.72    23.51     .3970     .6479
 71   .01087     .02407   -30.08    25.82     .3272     .7891
 72   .00742     .02632   -36.49    27.65     .2459     .9252
 73   .00461     .02868   -44.01    52.01     .1489    1.0411
 74   .00251     .03115   -55.85    64.25     .0288    1.1075
 75   .00105     .03378   -66.44    64.11     .1159    1.0682
 76   .00019     .03646   -78.55    87.21     .2845     .8646
 77   .00000     .03915   -93.29    95.08     .4674     .4857
 78   .00050     .04187  -107.31    80.64     .6377     .0347
 79   .00170     .04460  -119.50    59.41     .7713    -.3509
 80   .00361     .04738  -128.69    38.10     .8626    -.6174
 81   .00618     .05020  -135.60    26.23     .9322    -.8179
 82   .00940     .05305  -141.10    19.02     .9921    -.9867
 83   .01324     .05590  -145.65    14.69    1.0491   -1.1427
 84   .01771     .05871  -149.68    12.07    1.1015   -1.2815
 85   .02279     .06146  -153.31    10.03    1.1530   -1.4135
 86   .02850     .06412  -156.66     8.61    1.2044   -1.5397
 87   .03486     .06665  -159.76     7.06    1.2538   -1.6560
 88   .04191     .06907  -162.35     5.02    1.2901   -1.7382
 89   .04968     .07139  -164.24     3.34    1.3056   -1.7726
 90   .05813     .07366  -165.67     2.47    1.3085   -1.7789
 91   .06725     .07589  -166.85     1.96    1.3057   -1.7727
 92   .07701     .07807  -167.88     1.63    1.3002   -1.7607
 93   .08738     .08021  -168.79     1.39    1.2935   -1.7459
 94   .09833     .08230  -169.62     1.22    1.2864   -1.7299
 95   .10986     .08432  -170.39     1.08    1.2790   -1.7135
 96   .12193     .08629  -171.10      .96    1.2717   -1.6971
 97   .13452     .08819  -171.77      .87    1.2646   -1.6809
 98   .14762     .09000  -172.40      .80    1.2577   -1.6651
 99   .16120     .09174  -173.00      .74    1.2511   -1.6498
100   .17524     .09339  -173.58      .69    1.2447   -1.6350
101   .18972     .09495  -174.13      .64    1.2385   -1.6206
102   .20462     .09641  -174.67      .61    1.2326   -1.6068
103   .21991     .09777  -175.19      .58    1.2269   -1.5934
104   .23557     .09901  -175.70      .55    1.2215   -1.5805
105   .25157     .10015  -176.19      .53    1.2162   -1.5679
106   .26790     .10116  -176.68      .51    1.2110   -1.5556
107   .28452     .10206  -177.16      .50    1.2060   -1.5436
108   .30141     .10282  -177.64      .49    1.2011   -1.5318
```

```
  L       X          Y        ANG     KAPPA     MACH       CP

 109    .31855    .10346 -178.11       .48    1.1963   -1.5200
 110    .33591    .10396 -178.59       .47    1.1914   -1.5082
 111    .35347    .10432 -179.06       .47    1.1865   -1.4963
 112    .37119    .10454 -179.53       .47    1.1815   -1.4841
 113    .38906    .10461 -180.01       .46    1.1764   -1.4715
 114    .40705    .10453 -180.49       .47    1.1711   -1.4585
 115    .42513    .10430 -180.97       .47    1.1656   -1.4449
 116    .44328    .10391 -181.46       .47    1.1599   -1.4306
 117    .46146    .10337 -181.96       .48    1.1538   -1.4154
 118    .47966    .10267 -182.46       .49    1.1473   -1.3989
 119    .49785    .10181 -182.98       .51    1.1401   -1.3807
 120    .51599    .10078 -183.52       .53    1.1318   -1.3597
 121    .53406    .09957 -184.10       .57    1.1219   -1.3344
 122    .55203    .09819 -184.72       .63    1.1097   -1.3029
 123    .56989    .09661 -185.39       .69    1.0943   -1.2629
 124    .58761    .09482 -186.12       .74    1.0753   -1.2128
 125    .60519    .09282 -186.89       .75    1.0533   -1.1539
 126    .62264    .09060 -187.63       .72    1.0301   -1.0912
 127    .63995    .08817 -188.33       .67    1.0076   -1.0296
 128    .65712    .08556 -188.97       .62     .9862    -.9702
 129    .67413    .08278 -189.57       .58     .9661    -.9139
 130    .69096    .07986 -190.12       .54     .9474    -.8612
 131    .70759    .07682 -190.62       .51     .9301    -.8118
 132    .72401    .07367 -191.09       .48     .9137    -.7651
 133    .74018    .07044 -191.54       .46     .8983    -.7208
 134    .75607    .06713 -191.96       .45     .8837    -.6786
 135    .77168    .06377 -192.37       .43     .8698    -.6383
 136    .78696    .06036 -192.75       .42     .8565    -.5996
 137    .80190    .05693 -193.12       .42     .8437    -.5624
 138    .81647    .05349 -193.47       .41     .8314    -.5266
 139    .83064    .05005 -193.82       .41     .8196    -.4920
 140    .84440    .04662 -194.15       .42     .8081    -.4584
 141    .85771    .04323 -194.48       .42     .7969    -.4257
 142    .87056    .03987 -194.81       .43     .7860    -.3939
 143    .88293    .03656 -195.13       .45     .7754    -.3628
 144    .89478    .03332 -195.45       .46     .7650    -.3323
 145    .90610    .03016 -195.77       .49     .7547    -.3023
 146    .91686    .02709 -196.09       .52     .7445    -.2727
 147    .92706    .02412 -196.42       .56     .7345    -.2433
 148    .93666    .02126 -196.76       .61     .7244    -.2141
 149    .94565    .01852 -197.10       .67     .7143    -.1848
 150    .95402    .01592 -197.46       .75     .7041    -.1551
 151    .96174    .01347 -197.83       .86     .6937    -.1252
 152    .96879    .01117 -198.22      1.01     .6832    -.0948
 153    .97517    .00905 -198.64      1.17     .6724    -.0640
 154    .98085    .00711 -199.08      1.39     .6614    -.0324
 155    .98583    .00536 -199.55      1.74     .6495     .0015
 156    .99009    .00383 -200.06      2.32     .6361     .0394
 157    .99360    .00253 -200.64      3.12     .6228     .0767
 158    .99637    .00147 -201.29      5.05     .6111     .1092
 159    .99837    .00068 -202.11      8.34     .5922     .1615
 160    .99959    .00018 -203.20     32.83     .5571     .2565
 161   1.00000    .00000 -204.61    100.23     .5354     .3139
```



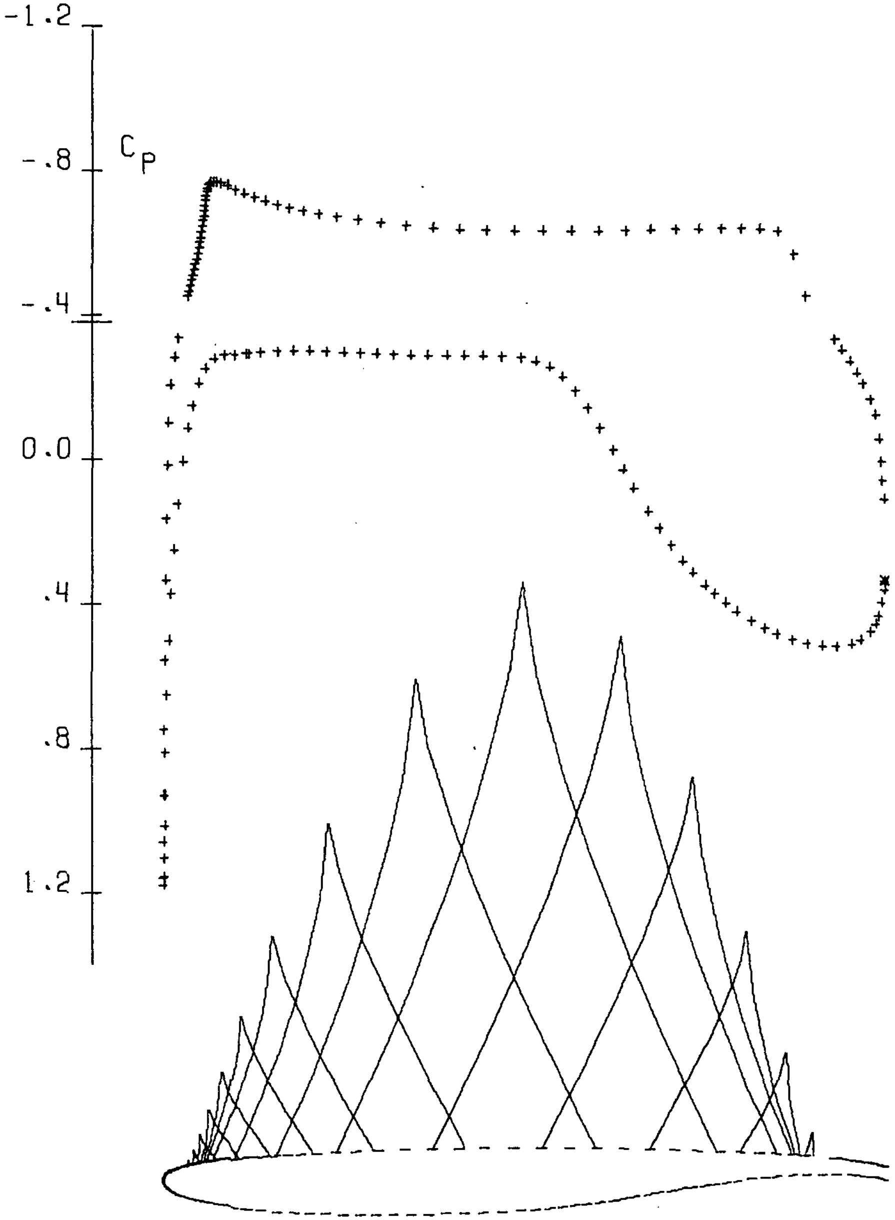
M=.600 CL=1.291 DY=.016 T/C=.100



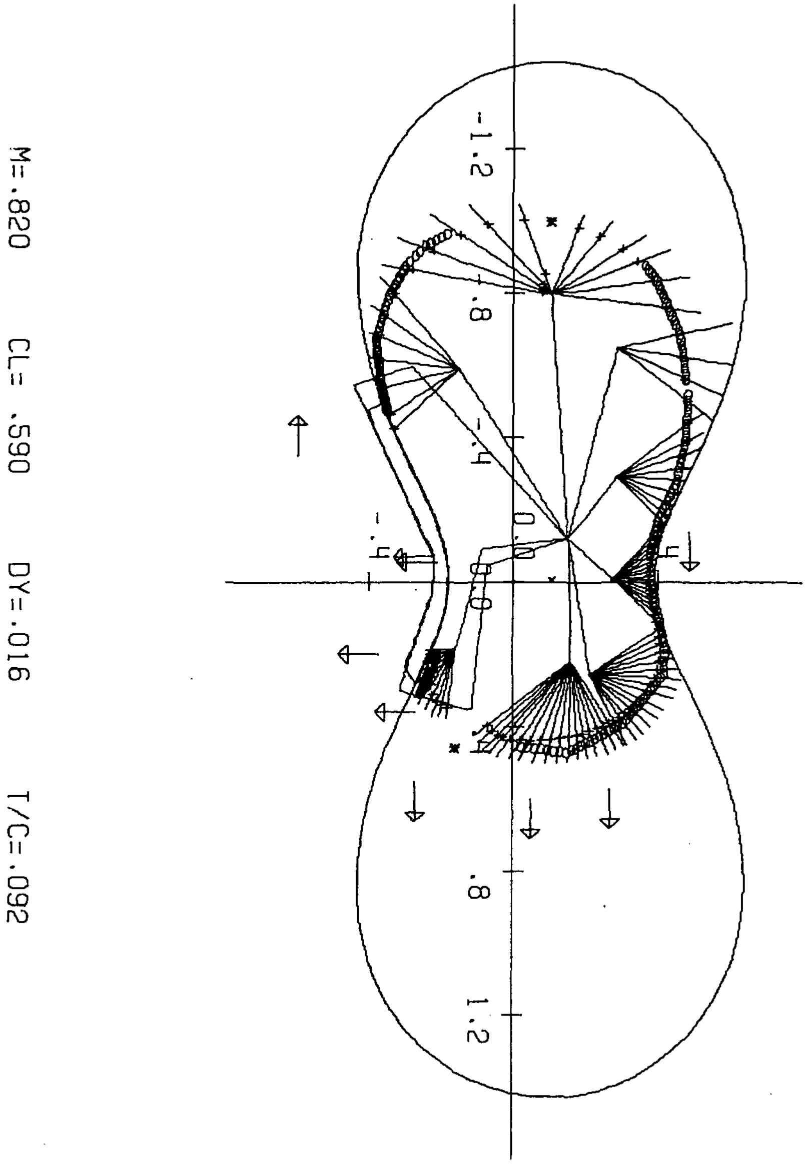
M=.600 CL=1.291 DY=.016 T/C=.100

```
                                                              07/18/74



                                  RUN= -274

                  CIRCULATORY FLOW ABOUT A TRANSONIC AIRFOIL

                  M= .600    CL=1.291    OY= .016    T/C= .100



                                  TAPE 6, PATH 0

                                  2  0
                                   -.800    0.000    2
                                  -1.000    0.000    2

                                  2  0
                                    .330     .050    2
                                    .570    -.325    2



                                    TAPE 7

-8-274   2  -.12  .30  .08  1.40  .600  .007  -.330  .025  .50    7
21   1   2   5   6  17  18  33  34  37  38  41  42  45  49  50
53  54  57  58  61  62

 .498  -.394  -.200   .600        -.256  -.220   .600   .450
0.000  0.000   .900  0.000        0.000  0.000   .900  0.000

 .011  -.009 -1.000   .900        0.000  0.000 -2.000  0.000
0.000  0.000 -2.000  0.000        0.000  0.000 -2.000  0.000

 .040  -.220   .590  -.170         .009   .119   .410  -.240
 .229   .588  -.080  -.440         .010  -.060  -.650  -.550

 .583  -.350   .460   .100         .580  -.130   .500   .050
 .120  -.114   .015   .040         .004  -.029  1.000  1.000
```

```
      AUTOMATION PATHS

 4   0
   -.800    0.000   1
 -1.100    -.250   -1
 -1.080    -.370    2
 -1.000    -.500    2

 5   0
   -.800    0.000   1
 -1.000    -.500   -1
   -.930    -.575   2
   -.814    -.620   2
   -.691    -.625   2

 6   0
   -.691    -.625  -1
   -.580    -.630   2
   -.480    -.620   2
   -.380    -.600   2
   -.280    -.575   1
   -.200    -.560   1

 5   0
   -.200    -.560  -1
   -.100    -.545   1
   -.054    -.540   1
    .046    -.540   1
    .110    -.545   1

-3   0
   -.250     .400  -1
   -.650     .500   2
   -.800     .450   2

-2   0
   -.250     .400  -1
    .100     .415   2

 5   0
    .340     .038  -1
    .480     .005   1
    .530    -.125   1
    .530    -.230   1
    .475    -.255   1

 6   0
   -.100    -.385   1
    .300    -.385   1
    .495    -.350  -1
    .385    -.440   1
    .210    -.540   1
    .110    -.545   1
```



M=.820 CL= .590 DY=.016 T/C=.092

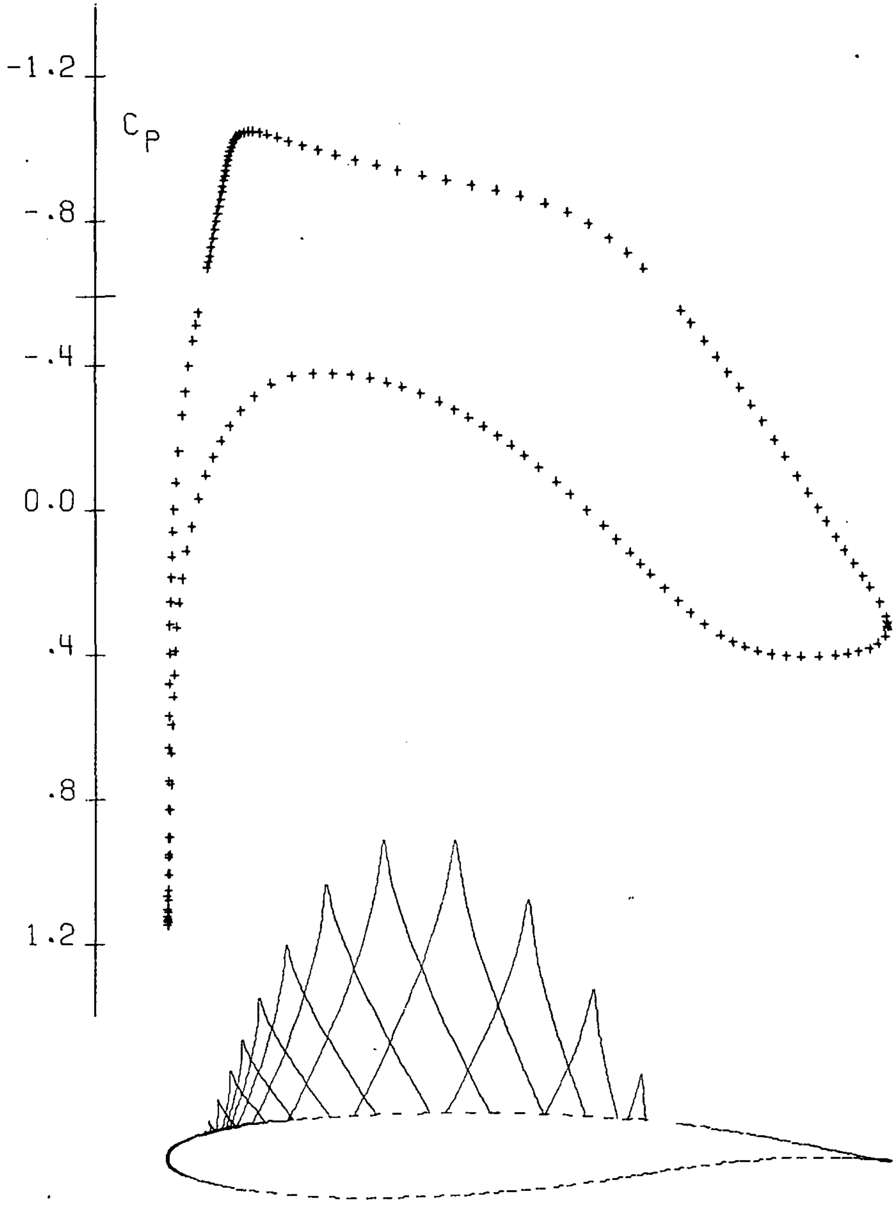

M=.820
CL= .590
DY=.016
T/C=.092
-1.2
-.8
-.4
0.0
.4
.8
1.2
-.4
0.0
.4

```
                                                        07/18/74



                          RUN= -255

         CIRCULATORY FLOW ABOUT A TRANSONIC AIRFOIL

         M= .820    CL= .590    DY= .016    T/C= .092


                        TAPE 6, PATH 0

                      2  0
                       -.800    0.000   2
                      -1.000    0.000   2

                      2  0
                        .300     .050   2
                        .470    -.260   2


                           TAPE 7

-8-255   4  -.12  .15  .08  1.40  .820  .005  -.105  .055  .50
23   1   2   5   6  10  13  14  17  18  33  34  37  38  41  42
49  50  53  54  57  58  61  62


-.278  -.198   .630   .050        .024  -.192   .600   .270
0.000  -.232  -.110   .490        .147  -.186   .580  -.270

 .104  -.034  -.380  -.500       0.000  0.000 -2.000  0.000
0.000  0.000 -2.000  0.000       0.000  0.000 -2.000  0.000

-.194   .165   .200  -.400        .102  -.079   .360  -.300
 .108   .231  -.056  -.240       0.000   .200  -.070  -.250

 .162   .120   .460   .100        .042   .208   .500   .050
 .058   .002   .062   .018        .029  -.023  1.000  1.000
```

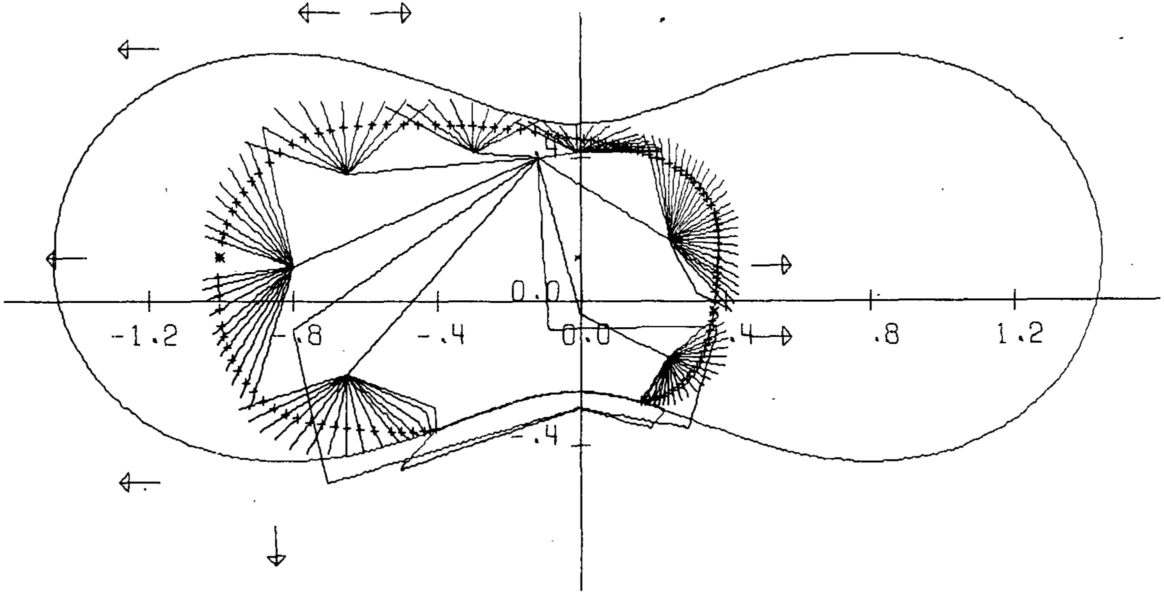
```
        AUTOMATION PATHS

5   0
 -.964     -.290   -1
 -.930     -.350    1
 -.874     -.405    1
 -.794     -.450    1
 -.675     -.485    1

3   0
 -.675     -.485   -1
 -.550     -.470    2
 -.460     -.450    5

5   0
 -.800     0.000    2
 -.875      .255   -1
 -.832      .285    1
 -.685      .350    1
 -.560      .365    1

5   0
 -.075      .250   -1
 -.180      .280    1
 -.270      .330    1
 -.390      .360    1
 -.520      .370    1

4   0
 -.020      .250   -1
  .080      .267    1
  .200      .300    1
  .270      .300    1

4   0
  .270      .300   -1
  .410      .190    1
  .490      .040    1
  .470     -.100    1

4   0
 -.100     -.200    1
  .245     -.220    1
  .320     -.355   -1
  .268     -.330   15

4   0
 -.100     -.200    1
  .245     -.220    1
  .268     -.330   -1
  .215     -.305   15
```

LISTING OF MEASURED COORDINATES FOR AIRFOIL 82-06-09

X	Y	X	Y	X	Y
1.00000	.01179	.24295	-.04044	.26430	.04430
.99940	.01198	.22222	-.03997	.28622	.04507
.99759	.01241	.20215	-.03936	.30866	.04577
.99459	.01335	.18280	-.03861	.33155	.04638
.99039	.01457	.16422	-.03772	.35486	.04691
.98502	.01592	.14645	-.03669	.37851	.04736
.97847	.01728	.12952	-.03553	.40245	.04772
.97077	.01862	.11349	-.03427	.42663	.04802
.96194	.01985	.09840	-.03290	.45099	.04823
.95200	.02095	.08426	-.03133	.47547	.04838
.94096	.02186	.07114	-.02954	.50000	.04844
.92886	.02254	.05904	-.02755	.52453	.04843
.91574	.02292	.04801	-.02535	.54901	.04835
.90160	.02295	.03806	-.02292	.57336	.04820
.88650	.02257	.02923	-.02027	.59755	.04798
.87048	.02174	.02153	-.01743	.62149	.04769
.85355	.02041	.01498	-.01452	.64514	.04733
.83578	.01856	.00961	-.01159	.66844	.04691
.81720	.01617	.00541	-.00859	.69134	.04643
.79785	.01329	.00241	-.00543	.71378	.04589
.77779	.00993	.00060	-.00201	.73570	.04530
.75705	.00614	0.00000	.00156	.75705	.04464
.73570	.00199	.00060	.00514	.77779	.04394
.71378	-.00245	.00241	.00847	.79785	.04317
.69134	-.00705	.00541	.01151	.81720	.04236
.66844	-.01167	.00961	.01427	.83578	.04150
.64514	-.01619	.01498	.01689	.85355	.04058
.62149	-.02045	.02153	.01942	.87048	.03940
.59755	-.02431	.02923	.02190	.88650	.03790
.57336	-.02770	.03806	.02431	.90160	.03618
.54901	-.03053	.04801	.02665	.91574	.03445
.52453	-.03286	.05904	.02871	.92886	.03272
.50000	-.03479	.07114	.03044	.94096	.03101
.47547	-.03637	.08426	.03203	.95200	.02933
.45099	-.03766	.09840	.03357	.96194	.02769
.42663	-.03869	.11349	.03506	.97077	.02608
.40245	-.03950	.12952	.03649	.97847	.02450
.37851	-.04013	.14645	.03785	.98502	.02297
.35486	-.04057	.16422	.03914	.99039	.02148
.33155	-.04085	.18280	.04034	.99459	.02010
.30866	-.04097	.20215	.04146	.99759	.01890
.28622	-.04093	.22222	.04249	.99940	.01804
.26430	-.04076	.24295	.04343	1.00000	.01774



M= .750 CL= .629 DY= .000 T/C= .117



M=.750 CL= .629 DY=.000 T/C=.117

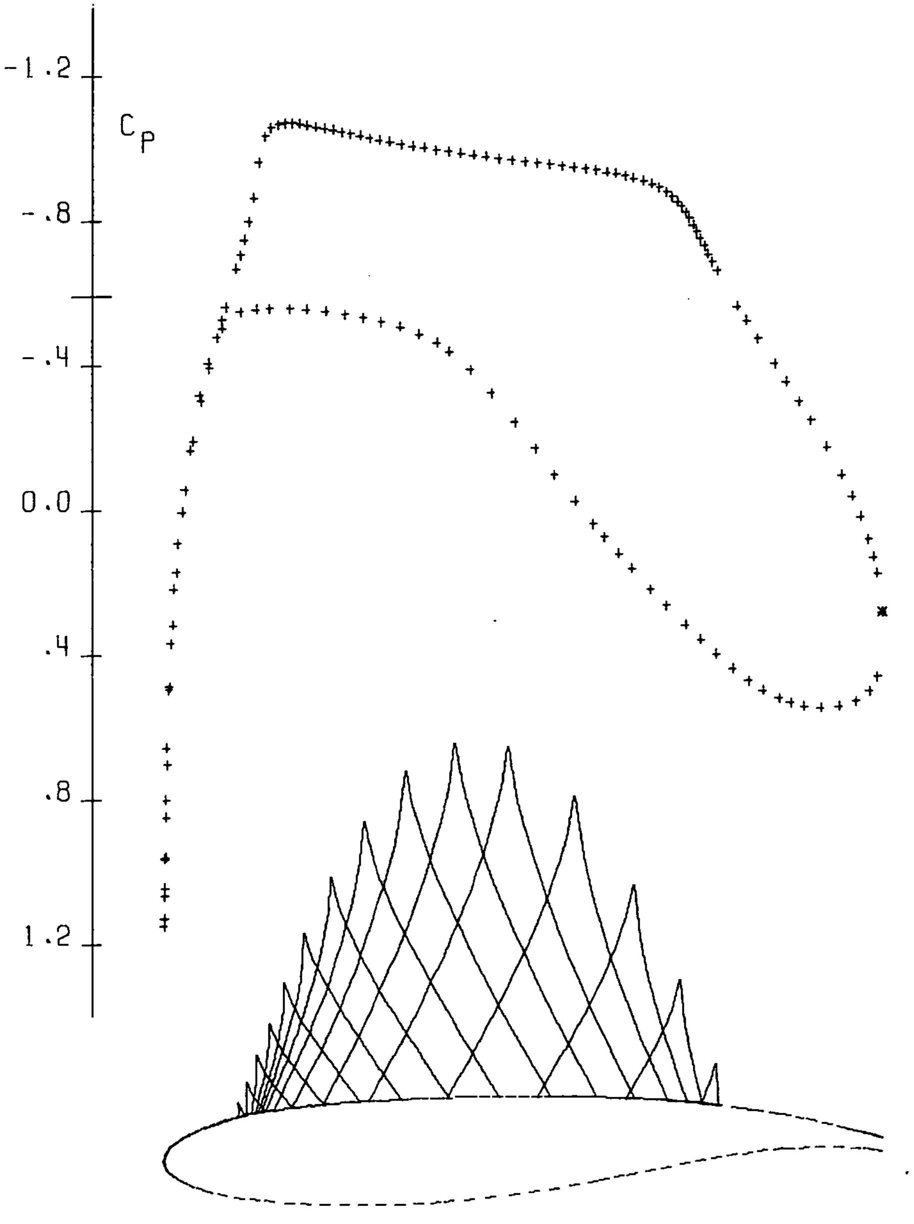
```
                                                              08/20/73



                                      RUN= -131

                      CIRCULATORY FLOW ABOUT A TRANSONIC AIRFOIL

                      M= .750     CL= .629     DY= .000     T/C= .117


                                   TAPE 6, PATH 0

                                  2   0
                                   -.800     0.000    1
                                  -1.000     0.000    1

                                  2   0
                                    .100      .200    1
                                    .390     -.142    1


                                       TAPE 7

99-131    4   -.12   .40   .08   1.40   .750   .010   -.120 0.000 0.00
 6    1    2    5    6   17   18

-.020   -.304    .500    .100        -.141    .237    .500   -.100
0.000    .060   -.550    .800        0.000   0.000   0.000    .900

-.039    .037  -1.400    .120        -.080   0.000  -1.200    .700
-.050    .050  -1.200   -.500        0.000    .030   -.700    .800

0.000    .040   -.850   -.650        0.000   0.000   0.000   -.900
0.000   0.000   0.000   -.900        0.000   0.000   0.000   -.900

1.700   0.000    .500   0.000        -.300    .200    .200   0.000
0.000   0.000    .080    .016        0.000   0.000   5.000   1.000
```

LISTING OF COORDINATES FOR AIRFOIL 75-06-12

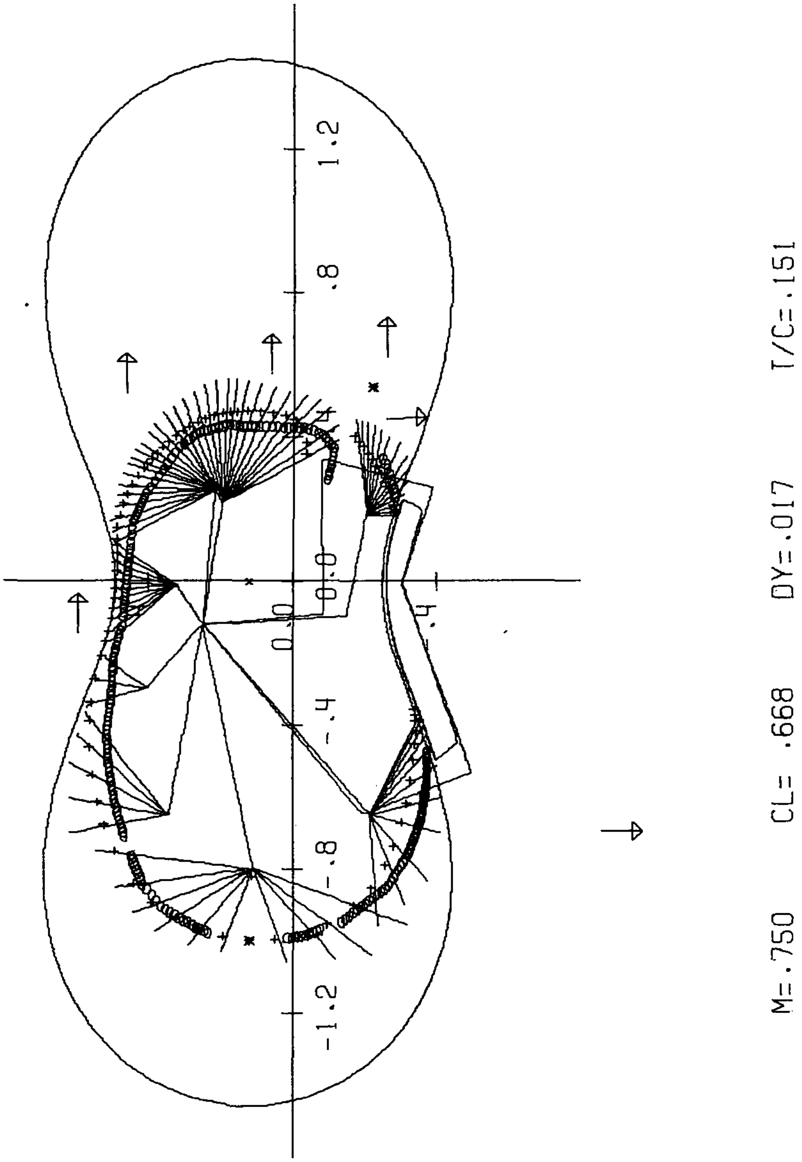
L	X	Y	ANG	KAPPA	MACH	CP
1	1.00000	0.00000	-7.21	-26.94	.6145	.3190
2	.99953	.00006	-6.79	-9.99	.6122	.3243
3	.99810	.00022	-6.35	-3.78	.6059	.3363
4	.99572	.00047	-5.90	-2.82	.6014	.3488
5	.99237	.00080	-5.43	-2.12	.5974	.3579
6	.98806	.00119	-4.96	-1.75	.5944	.3647
7	.98278	.00163	-4.47	-1.50	.5916	.3711
8	.97654	.00208	-3.97	-1.32	.5887	.3774
9	.96934	.00255	-3.45	-1.18	.5862	.3830
10	.96120	.00300	-2.92	-1.08	.5841	.3878
11	.95212	.00342	-2.38	-1.00	.5822	.3921
12	.94212	.00378	-1.83	-.94	.5805	.3958
13	.93122	.00407	-1.26	-.89	.5792	.3988
14	.91943	.00427	-.67	-.86	.5781	.4011
15	.90679	.00435	-.05	-.84	.5774	.4028
16	.89331	.00428	.59	-.83	.5770	.4037
17	.87902	.00405	1.27	-.83	.5770	.4036
18	.86398	.00362	1.98	-.83	.5777	.4022
19	.84821	.00296	2.74	-.84	.5790	.3991
20	.83177	.00206	3.53	-.84	.5814	.3938
21	.81471	.00088	4.36	-.84	.5852	.3854
22	.79713	-.00059	5.19	-.81	.5906	.3732
23	.77909	-.00236	6.01	-.76	.5981	.3563
24	.76068	-.00442	6.77	-.66	.6077	.3344
25	.74198	-.00675	7.42	-.54	.6195	.3075
26	.72309	-.00930	7.94	-.41	.6332	.2760
27	.70405	-.01202	8.31	-.26	.6483	.2410
28	.68491	-.01485	8.53	-.13	.6644	.2034
29	.66571	-.01774	8.61	-.01	.6810	.1643
30	.64646	-.02065	8.56	.08	.6978	.1246
31	.62720	-.02352	8.42	.17	.7144	.0851
32	.60792	-.02634	8.20	.23	.7307	.0462
33	.58864	-.02907	7.91	.29	.7465	.0084
34	.56937	-.03168	7.57	.33	.7617	-.0282
35	.55012	-.03418	7.18	.36	.7763	-.0632
36	.53091	-.03653	6.77	.39	.7903	-.0967
37	.51174	-.03873	6.33	.41	.8035	-.1285
38	.49263	-.04077	5.87	.42	.8160	-.1585
39	.47360	-.04265	5.40	.44	.8277	-.1866
40	.45467	-.04436	4.91	.45	.8387	-.2128
41	.43584	-.04589	4.42	.46	.8488	-.2371
42	.41714	-.04726	3.92	.47	.8581	-.2593
43	.39859	-.04844	3.41	.48	.8666	-.2796
44	.38020	-.04946	2.90	.49	.8742	-.2978
45	.36200	-.05030	2.39	.49	.8810	-.3140
46	.34401	-.05097	1.88	.50	.8870	-.3282
47	.32624	-.05147	1.36	.51	.8923	-.3406
48	.30873	-.05181	.84	.52	.8967	-.3511
49	.29149	-.05198	.32	.54	.9004	-.3599
50	.27455	-.05200	-.22	.56	.9035	-.3671
51	.25793	-.05186	-.76	.58	.9059	-.3728
52	.24165	-.05156	-1.32	.62	.9076	-.3769
53	.22574	-.05112	-1.90	.66	.9086	-.3793

```
  L       X          Y         ANG     KAPPA     MACH       CP

 54    .21022    -.05052     -2.50      .70     .9089    -.3799
 55    .19512    -.04978     -3.13      .76     .9082    -.3783
 56    .18044    -.04889     -3.80      .82     .9055    -.3742
 57    .16622    -.04786     -4.50      .90     .9036    -.3674
 58    .15247    -.04669     -5.25      .99     .8994    -.3574
 59    .13920    -.04538     -6.04     1.09     .8936    -.3438
 60    .12644    -.04394     -6.89     1.20     .8862    -.3263
 61    .11420    -.04237     -7.78     1.33     .8772    -.3049
 62    .10250    -.04067     -8.73     1.49     .8666    -.2797
 63    .09134    -.03886     -9.75     1.66     .8544    -.2504
 64    .08075    -.03693    -10.84     1.88     .8405    -.2173
 65    .07074    -.03491    -12.01     2.14     .8251    -.1804
 66    .06131    -.03280    -13.29     2.49     .8082    -.1397
 67    .05250    -.03061    -14.69     2.93     .7896    -.0950
 68    .04431    -.02834    -16.26     3.53     .7692    -.0461
 69    .03676    -.02602    -18.03     4.37     .7468     .0076
 70    .02987    -.02364    -20.10     5.61     .7222     .0666
 71    .02366    -.02122    -22.58     7.54     .6945     .1325
 72    .01814    -.01877    -25.68    10.71     .6623     .2082
 73    .01336    -.01626    -29.75    16.17     .6232     .2990
 74    .00932    -.01370    -35.37    25.93     .5711     .4169
 75    .00605    -.01105    -43.14    38.82     .4944     .5812
 76    .00351    -.00827    -52.45    44.89     .3854     .7895
 77    .00165    -.00538    -62.46    62.39     .2581     .9817
 78    .00049    -.00240    -75.49    70.59     .1098    1.1177
 79   0.00000     .00074    -86.59    49.28     .0423    1.1440
 80    .00003     .00399    -94.60    43.43     .1878    1.0590
 81    .00056     .00717   -104.40    64.46     .3552     .8406
 82    .00171     .01015   -117.71    74.24     .5297     .5071
 83    .00365     .01296   -130.57    51.62     .6490     .2393
 84    .00640     .01573   -138.64    25.91     .7141     .0859
 85    .00993     .01852   -144.14    17.39     .7661    -.0386
 86    .01415     .02133   -148.32    12.04     .8118    -.1484
 87    .01906     .02415   -151.72     9.11     .8532    -.2476
 88    .02461     .02696   -154.58     7.11     .8918    -.3394
 89    .03080     .02973   -157.07     5.81     .9284    -.4258
 90    .03761     .03245   -159.30     4.87     .9635    -.5075
 91    .04503     .03510   -161.32     4.13     .9979    -.5865
 92    .05304     .03766   -163.20     3.74    1.0331    -.6659
 93    .06163     .04009   -165.07     3.54    1.0709    -.7494
 94    .07080     .04237   -166.98     3.63    1.1143    -.8428
 95    .08057     .04444   -169.05     3.37    1.1659    -.9501
 96    .09101     .04629   -170.74     2.15    1.2015   -1.0218
 97    .10215     .04800   -171.78     1.27    1.2122   -1.0427
 98    .11398     .04963   -172.56     1.01    1.2138   -1.0460
 99    .12642     .05118   -173.21      .82    1.2120   -1.0424
100    .13946     .05266   -173.79      .71    1.2086   -1.0357
101    .15306     .05408   -174.31      .63    1.2044   -1.0274
102    .16720     .05543   -174.79      .56    1.1999   -1.0185
103    .18183     .05670   -175.24      .51    1.1952   -1.0092
104    .19694     .05791   -175.67      .47    1.1906    -.9999
105    .21251     .05903   -176.07      .43    1.1860    -.9907
106    .22850     .06007   -176.45      .40    1.1815    -.9817
107    .24488     .06104   -176.82      .38    1.1772    -.9730
108    .26164     .06192   -177.17      .36    1.1730    -.9646
```

```
  L       X          Y       ANG      KAPPA      MACH       CP

 109    .27873     .06271 -177.51       .34     1.1690    -.9564
 110    .29615     .06341 -177.84       .32     1.1651    -.9486
 111    .31385     .06403 -178.17       .31     1.1614    -.9409
 112    .33181     .06455 -178.48       .30     1.1578    -.9335
 113    .35000     .06499 -178.80       .29     1.1543    -.9262
 114    .36839     .06532 -179.10       .29     1.1508    -.9191
 115    .38695     .06556 -179.41       .29     1.1473    -.9120
 116    .40565     .06571 -179.71       .28     1.1439    -.9048
 117    .42446     .06575 -180.02       .29     1.1404    -.8975
 118    .44335     .06569 -180.33       .29     1.1367    -.8898
 119    .46229     .06553 -180.65       .30     1.1328    -.8818
 120    .48125     .06526 -180.97       .30     1.1286    -.8730
 121    .50019     .06489 -181.31       .32     1.1240    -.8632
 122    .51910     .06439 -181.66       .34     1.1187    -.8521
 123    .53793     .06379 -182.04       .36     1.1126    -.8391
 124    .55666     .06305 -182.44       .39     1.1053    -.8235
 125    .57526     .06219 -182.88       .43     1.0964    -.8046
 126    .59371     .06118 -183.35       .47     1.0858    -.7817
 127    .61198     .06003 -183.86       .50     1.0734    -.7548
 128    .63006     .05873 -184.39       .52     1.0595    -.7244
 129    .64794     .05727 -184.94       .54     1.0444    -.6909
 130    .66559     .05566 -185.50       .55     1.0282    -.6549
 131    .68302     .05390 -186.05       .56     1.0115    -.6172
 132    .70019     .05200 -186.60       .56      .9942    -.5779
 133    .71710     .04996 -187.15       .55      .9765    -.5374
 134    .73373     .04780 -187.67       .54      .9586    -.4961
 135    .75007     .04552 -188.18       .53      .9406    -.4544
 136    .76609     .04315 -188.67       .52      .9226    -.4121
 137    .78179     .04069 -189.14       .50      .9044    -.3693
 138    .79714     .03816 -189.57       .47      .8863    -.3265
 139    .81214     .03558 -189.98       .45      .8682    -.2834
 140    .82675     .03296 -190.35       .41      .8501    -.2403
 141    .84096     .03032 -190.67       .37      .8322    -.1974
 142    .85475     .02769 -190.95       .31      .8145    -.1550
 143    .86810     .02508 -191.17       .25      .7972    -.1133
 144    .88099     .02252 -191.33       .18      .7803    -.0727
 145    .89338     .02003 -191.43       .09      .7640    -.0336
 146    .90526     .01762 -191.46       .00      .7495     .0037
 147    .91660     .01533 -191.43      -.10      .7338     .0388
 148    .92735     .01316 -191.34      -.20      .7201     .0716
 149    .93749     .01114 -191.19      -.30      .7073     .1019
 150    .94699     .00928 -191.00      -.40      .6956     .1298
 151    .95580     .00759 -190.76      -.51      .6848     .1554
 152    .96390     .00607 -190.50      -.62      .6749     .1788
 153    .97125     .00472 -190.21      -.75      .6657     .2002
 154    .97783     .00355 -189.89      -.91      .6573     .2199
 155    .98360     .00256 -189.56     -1.08      .6496     .2380
 156    .98854     .00175 -189.22     -1.27      .6424     .2548
 157    .99262     .00109 -188.89     -1.56      .6355     .2706
 158    .99582     .00060 -188.55     -2.34      .6289     .2859
 159    .99813     .00026 -188.16     -3.51      .6224     .3009
 160    .99953     .00006 -187.72    -11.36      .6168     .3136
 161   1.00000     .00000 -187.21    -32.82      .6145     .3190
```



M=.750 CL= .668 DY=.017 T/C=.151

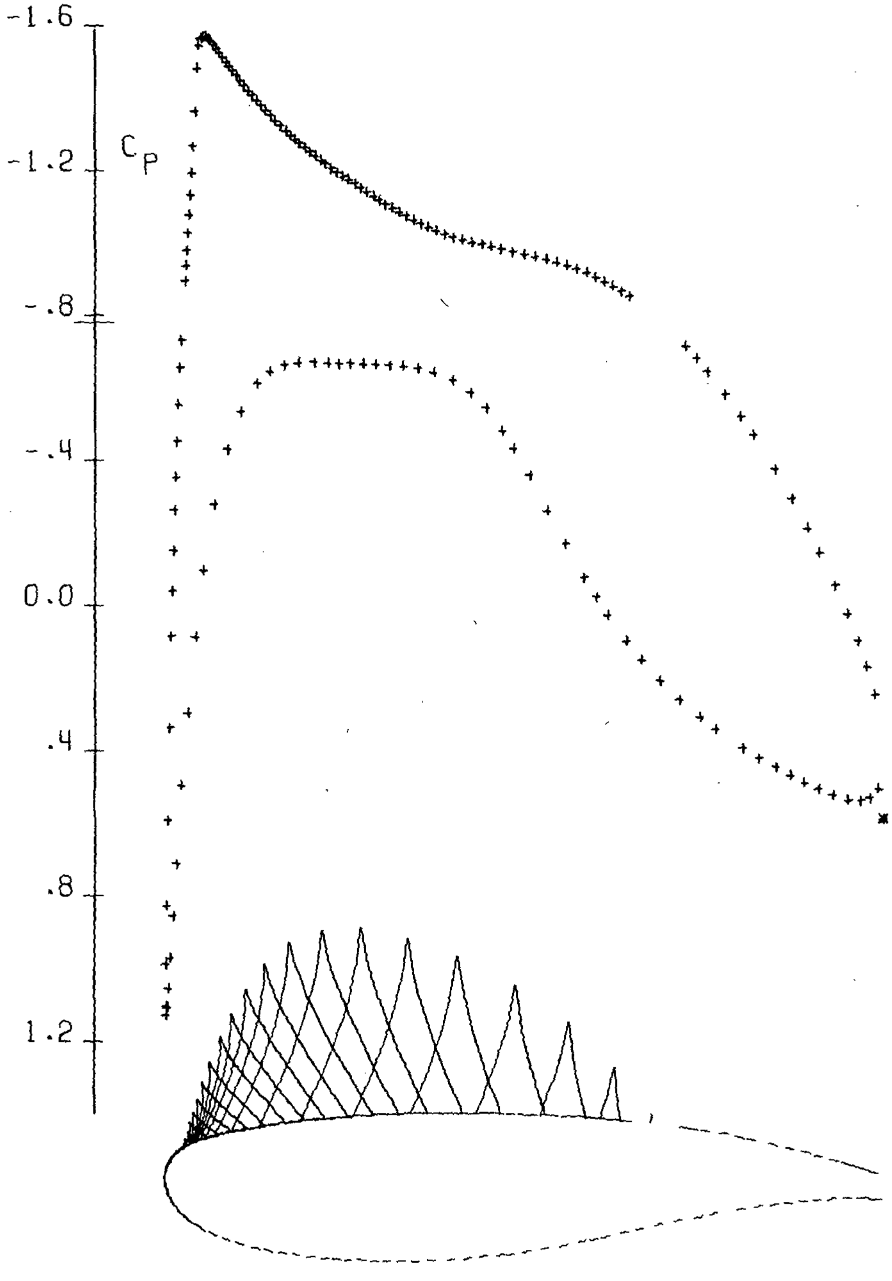

1.2
.8
.4
0.0
0.0
-.4
-.8
-1.2
M=.750
CL= .668
DY=.017
T/C=.151

```
                                                             08/24/73



                                 RUN= -242
                  CIRCULATORY FLOW ABOUT A TRANSONIC AIRFOIL

                  M= .750    CL= .668    DY= .017    T/C= .151



                               TAPE 6, PATH 0

                             2  0
                              -.800    0.000   2
                             -1.000    0.000   1

                             2  0
                               .300    0.000   1
                               .550    -.340   1



                                  TAPE 7

-6-242   4  -.12  .25  .08  1.40  .750  .007  -.110  .050  .50
17   1   2   5   6   9  10  14  33  34  37  38  49  50  53  54
57  58

-.150  -.134   .600   .050       -.048  -.174   .550   .450
 .013  -.081   .650  -.270       0.000  -.067  -.120   .580

0.000  0.000 -2.000  0.000       0.000  0.000 -2.000  0.000
0.000  0.000 -2.000  0.000       0.000  0.000 -2.000  0.000

-.112  -.017   .450  -.300        .146  -.228  -.700  -.900
0.000  0.000  0.000  -.900       0.000  0.000  0.000  -.900

 .706   .118   .460   .100        .305   .053   .500   .050
 .072   .092   .095   .010       0.000  0.000   .500  1.000
```

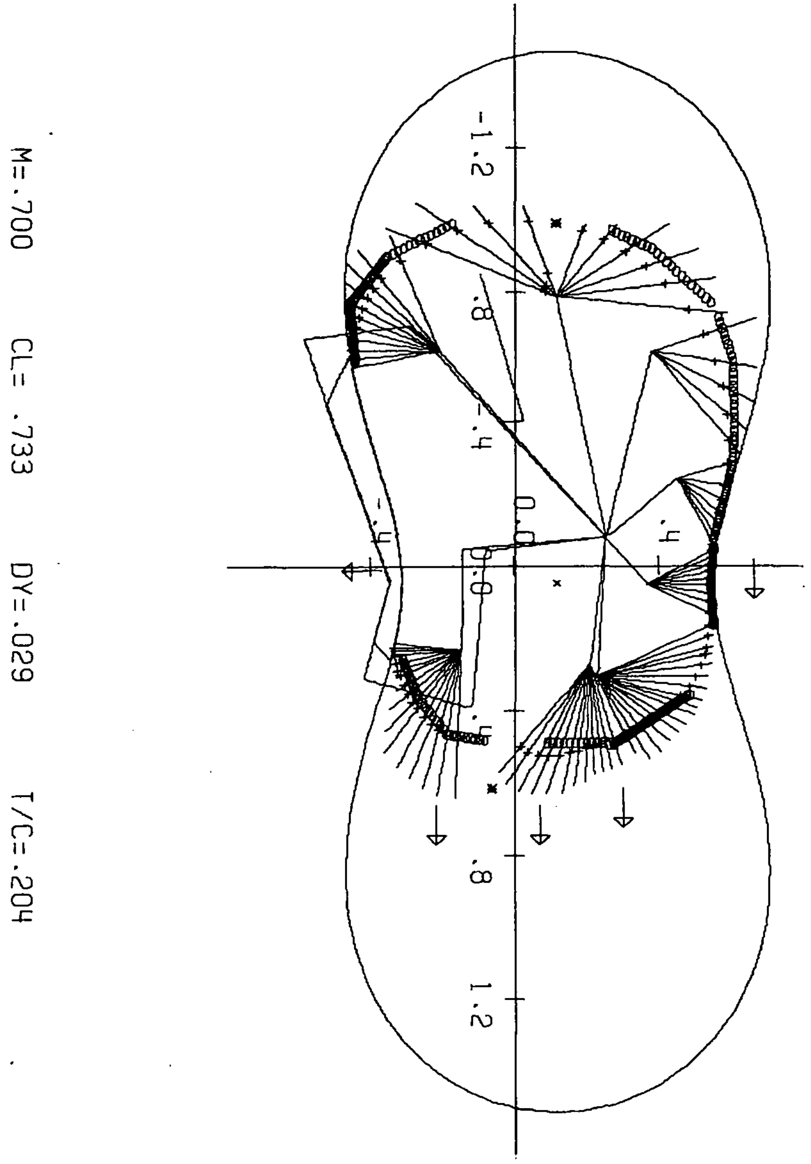
```
        AUTOMATION PATHS

 4   0
  -.800     0.000    2
  -.995     -.110   -1
  -.985     -.160    1
  -.970     -.200    1

 6   0
  -.948     -.260   -1
  -.848     -.390    1
  -.763     -.445    1
  -.663     -.480    2
  -.563     -.495    2
  -.478     -.495    2

 4   0
  -.093     -.210    1
   .242     -.400   -1
   .292     -.390    1
   .353     -.365    1

15   0
  -.098      .325   -1
  -.003      .315    1
   .102      .305    1
   .182      .300    1
   .261      .270    1
   .390      .183    1
   .427      .140    1
   .450      .060    1
   .447     -.030    1
   .443     -.095    1
   .442     -.150    1
   .420     -.200    1
   .382     -.230    1
   .337     -.220    1
   .297     -.208    1

 7   0
  -.128      .330   -1
  -.213      .355    1
  -.323      .365    1
  -.418      .380    1
  -.523      .375    1
  -.623      .360    1
  -.713      .335    1

 6   0
  -.800     0.000    2
  -.980      .110   -1
  -.935      .200    1
  -.900      .250    1
  -.840      .300    1
  -.760      .330    1
```

LISTING OF MEASURED COORDINATES FOR AIRFOIL 75-07-15

X	Y	X	Y	X	Y
23.0017	.1900	13.0017	0.0000	17.1994	.8852
22.7544	.2276	13.0483	.1479	17.3999	.8889
22.5046	.2519	13.0948	.2003	17.5949	.8913
22.0142	.2676	13.1962	.2798	17.7991	.8929
21.5407	.2461	13.2997	.3408	18.0006	.8934
21.0028	.1725	13.3997	.3919	18.2000	.8932
20.4717	.0691	13.4987	.4358	18.4000	.8914
20.0035	-.0287	13.5987	.4763	18.5993	.8888
19.5187	-.1316	13.6987	.5127	18.7994	.8851
19.0034	-.2385	13.7979	.5460	18.9999	.8798
18.4407	-.3454	13.8985	.5764	19.2825	.8711
18.0029	-.4186	13.9992	.6042	19.4705	.8634
17.5497	-.4827	14.1996	.6523	19.6595	.8547
17.0029	-.5410	14.3998	.6886	19.9998	.8345
16.5437	-.5738	14.5991	.7169	20.3325	.8077
16.0030	-.5944	14.7981	.7407	20.5000	.7906
15.5157	-.5982	14.9986	.7616	20.7275	.7640
15.0035	-.5855	15.1986	.7811	20.9985	.7266
14.5022	-.5539	15.3981	.7981	21.2986	.6791
14.0038	-.4948	15.5991	.8132	21.4991	.6437
13.7538	-.4485	15.7987	.8271	21.6925	.6068
13.5027	-.3820	15.9991	.8392	22.0012	.5422
13.4023	-.3470	16.1983	.8503	22.2235	.4890
13.3039	-.3056	16.3991	.8594	22.4965	.4074
13.2017	-.2531	16.5985	.8675	22.7482	.3141
13.0996	-.1820	16.7993	.8747	22.8835	.2559
13.0529	-.1331	16.9985	.8804	23.0017	.1990

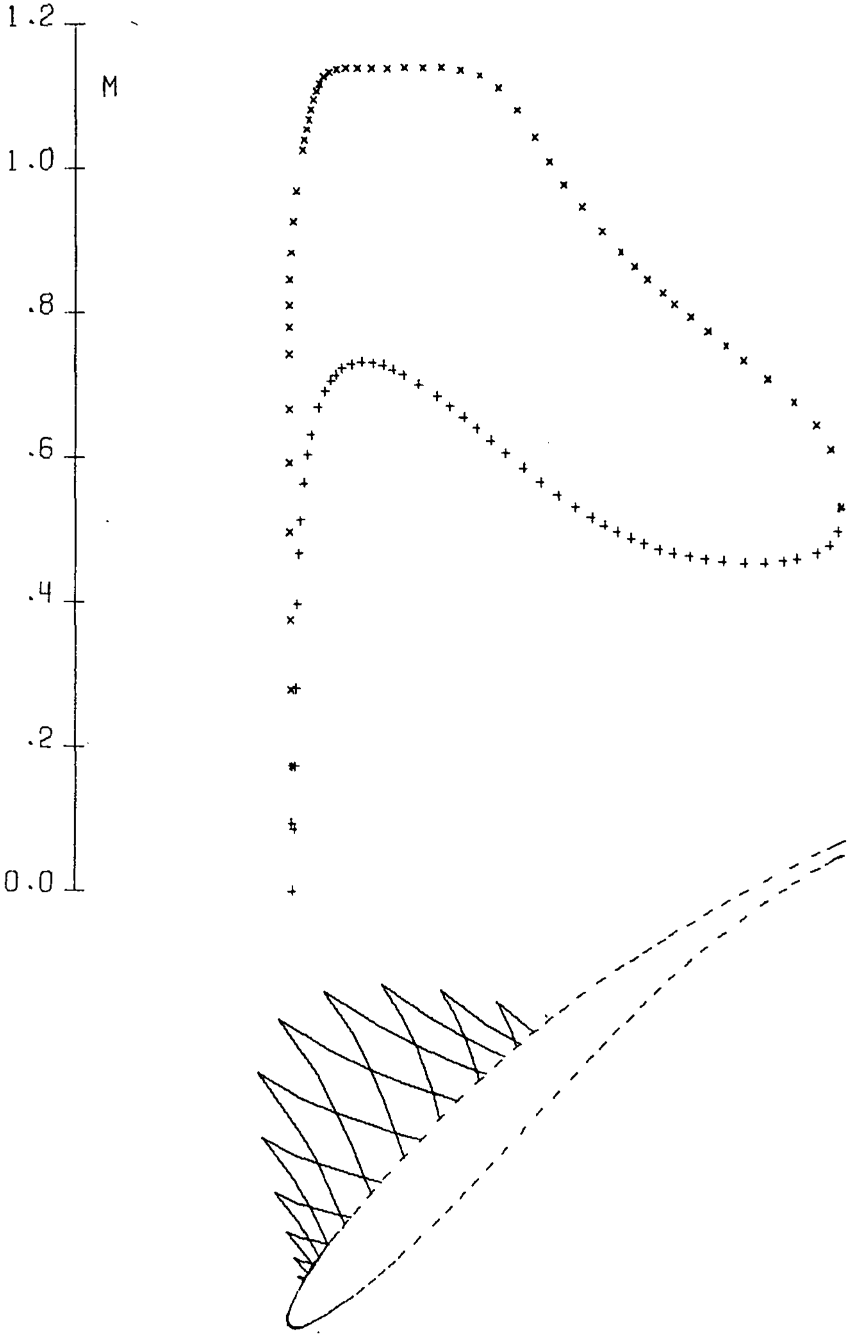


M=.700 CL= .733 DY=.029 T/C=.204

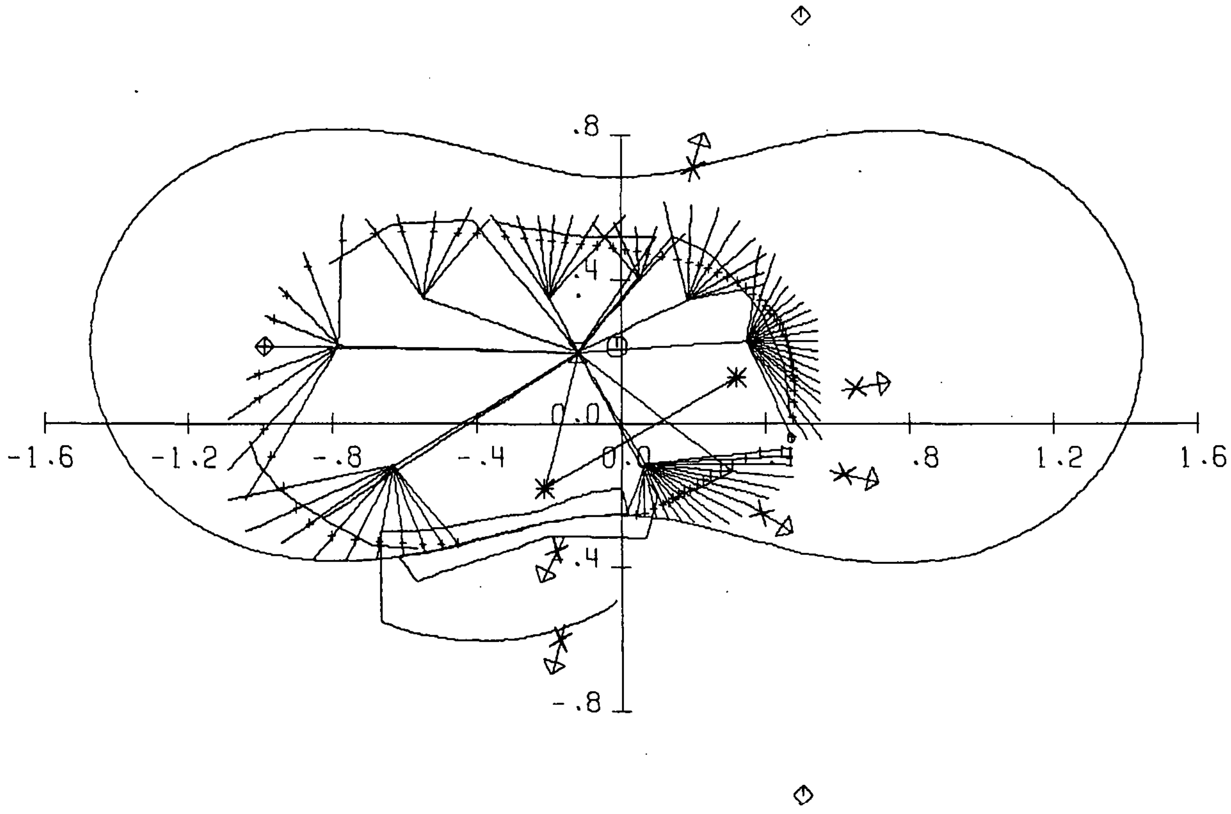

-1.2
-.8
-.4
0.0
.4
.8
1.2
.4
-.4
0.0
M=.700
CL= .733
DY=.029
T/C=.204

```
                                                         08/24/73



                              RUN=  -79

              CIRCULATORY FLOW ABOUT A TRANSONIC AIRFOIL

              M= .700   CL= .733   DY= .029   T/C= .204



                                 TAPE 6, PATH 0

                              2  0
                               -.800    0.000    2
                              -1.000    0.000    1

                              2  0
                                .300    0.000    1
                                .580    -.180    1


                                   TAPE 7

-7 -79   4  -.12  .25  .08  1.40  .700 -.005  -.115  .080  .50   7
18   1   2   5    6    9   10   13   14   33   34   49   50   53   54   57
58  61  62

-.107   .033   .650   .070         -.061  -.316   .600   .300
-.109  -.078   .650  -.220         -.078  -.132  -.030   .660

0.000  0.000 -2.000  0.000         0.000  0.000 -2.000  0.000
0.000  0.000 -2.000  0.000         0.000  0.000 -2.000  0.000

 .156   .190  -.030  -.400         0.000  0.000  0.000  -.900
0.000  0.000  0.000  -.900         0.000  0.000  0.000  -.900

 .113   .041   .200   .010          .265   .178  -.100   .050
 .214   .153   .200   .080          .063   .051  1.000  1.000
```

```
        AUTOMATION PATHS

3   0
-1.000    -.290   -1
 -.910    -.470    1
 -.780    -.575    2

2   0
 -.780    -.575   -1
 -.620    -.560    5

4   0
 -.800    0.000    2
 -.980     .150   -1
 -.930     .260    1
 -.780     .420    1

5   0
 -.128     .415   -1
 -.320     .470    2
 -.410     .480    1
 -.620     .480    1
 -.740     .440    1

3   0
 -.098     .415   -1
 0.000     .405    4
  .120     .415    4

3   0
  .320     .350   -1
  .460     .150    3
  .460    -.030    1

5   0
 -.093    -.210    1
  .220    -.420   -1
  .330    -.380    2
  .430    -.300    1
  .450    -.200    1
```



M1=.800 M2=.561 DEL TH= 12.03 G/C=1.45



M1=.800 M2=.561 DEL TH= 12.03 G/C=1.45

2. Evaluation of Analysis Methods

In this section we present a collection of computer generated plots comparing results from different calculations of identical flows which demonstrate the correctness and reliability of the computer programs which we have developed.

First we compare the design pressure distribution of Airfoil 78-06-10 with an inviscid analysis of the same flow at a very fine mesh size using the old Murman finite difference scheme listed in Section 5 of Chapter III, and with an analysis using the new quasiconservative option listed in Section 7 of Chapter III. Then we compare the analysis of the flow past an NACA 0012 airfoil by these two schemes, and also by a fully conservative scheme which we have not listed. The quasiconservative and fully conservative schemes can be seen to give essentially the same shock jump, but the fully conservative scheme requires more computer time. The nonconservative scheme does not give the full shock jump but agrees better with the design calculation for a shock free flow.

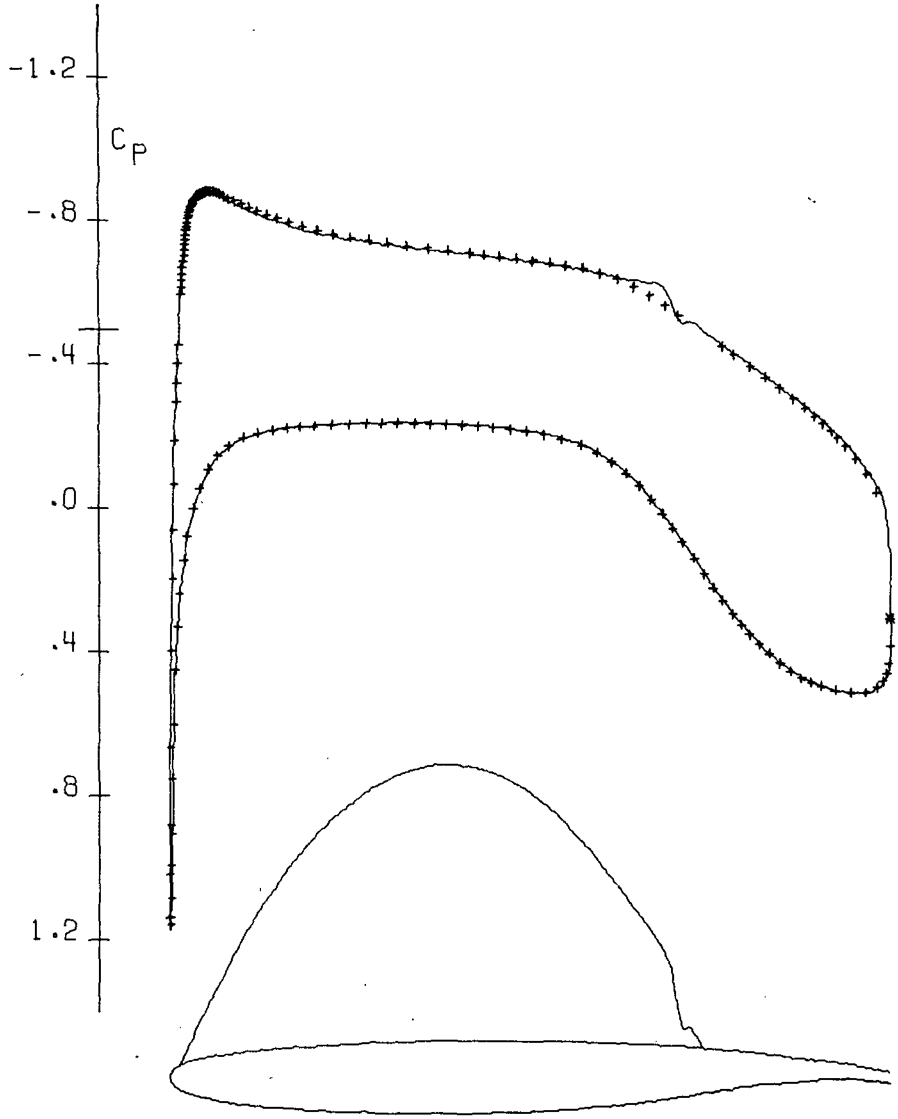
For Airfoil 70-10-13 we have subtracted the calculated boundary layer displacement thickness from the design profile. Then we have compared the design pressure distribution with the result of an analysis using Program H to add a boundary layer correction which should restore the original shape if there is no separation. The good agreement provides evidence that our new model of the tail should eliminate the loss of lift which was experienced with the airfoils from Volume I. The calculated displacement thickness of the boundary layer is also shown.

Next we compare the results of calculations on crude and fine grids for Airfoil 75-07-15 using Program H. Two shocks appear on the fine grid, but the flow is almost shock free on the crude grid. This illustrates that on a crude grid the artificial viscosity introduced by the retarded difference scheme can occasionally suppress a weak shock.

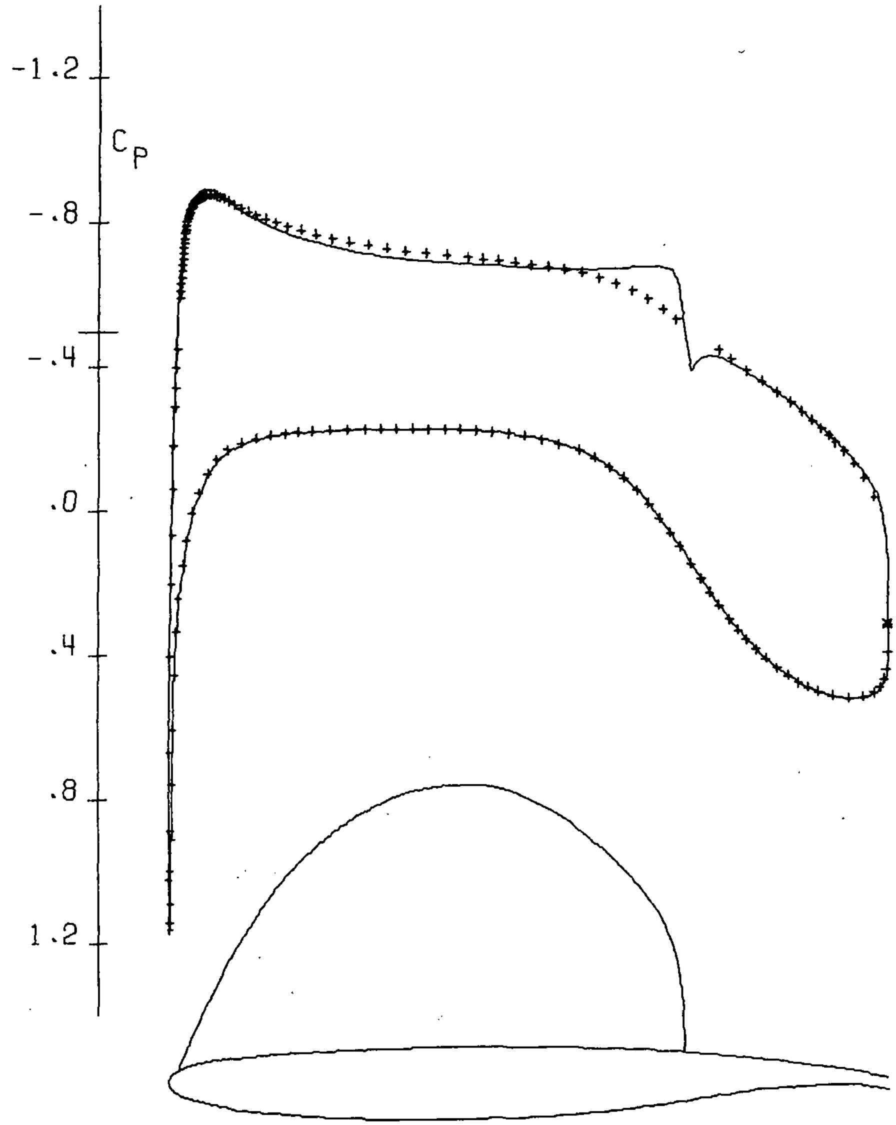
There follows a series of inviscid subsonic and supersonic two dimensional calculations for Airfoil 65-15-10 performed with an unlisted program which uses parabolic coordinates. The shock free flow at Mach .65 is in good agreement with the design calculation. Thus it appears that satisfactory accuracy can be obtained with the parabolic coordinate system, which we have used in three dimensional calculations.

For the same airfoil we present results from another unlisted program which calculates the flow over an infinite yawed wing. The program uses the full three dimensional difference scheme although the flow is effectively two dimensional. The purpose of these calculations is to check the effectiveness of the rotated difference scheme in preserving invariance of the flow at corresponding Mach numbers and yaw angles. First we compare crude and fine grid calculations for an unyawed wing. This is a typical case where the airfoil is operating below its design point and two shocks appear. The second shock is eliminated, however, on the crude grid. Next we compare corresponding yawed and unyawed conditions on the fine grid. In the yawed condition most of the flow is treated by the supersonic difference scheme, and the resulting extra artificial viscosity is sufficient to eliminate the second shock, as on a coarse grid in the unyawed condition. Away from the shock waves, however, the two calculations remain in remarkable agreement.

Finally we show a pressure distribution on a wing of low aspect ratio with a 79-03-12 section calculated by the three dimensional analysis Program J listed in Section 6 of Chapter III. Although the section was designed for Mach .79, drag rise is only just beginning at Mach .83, illustrating the Mach relief due to three dimensional effects.



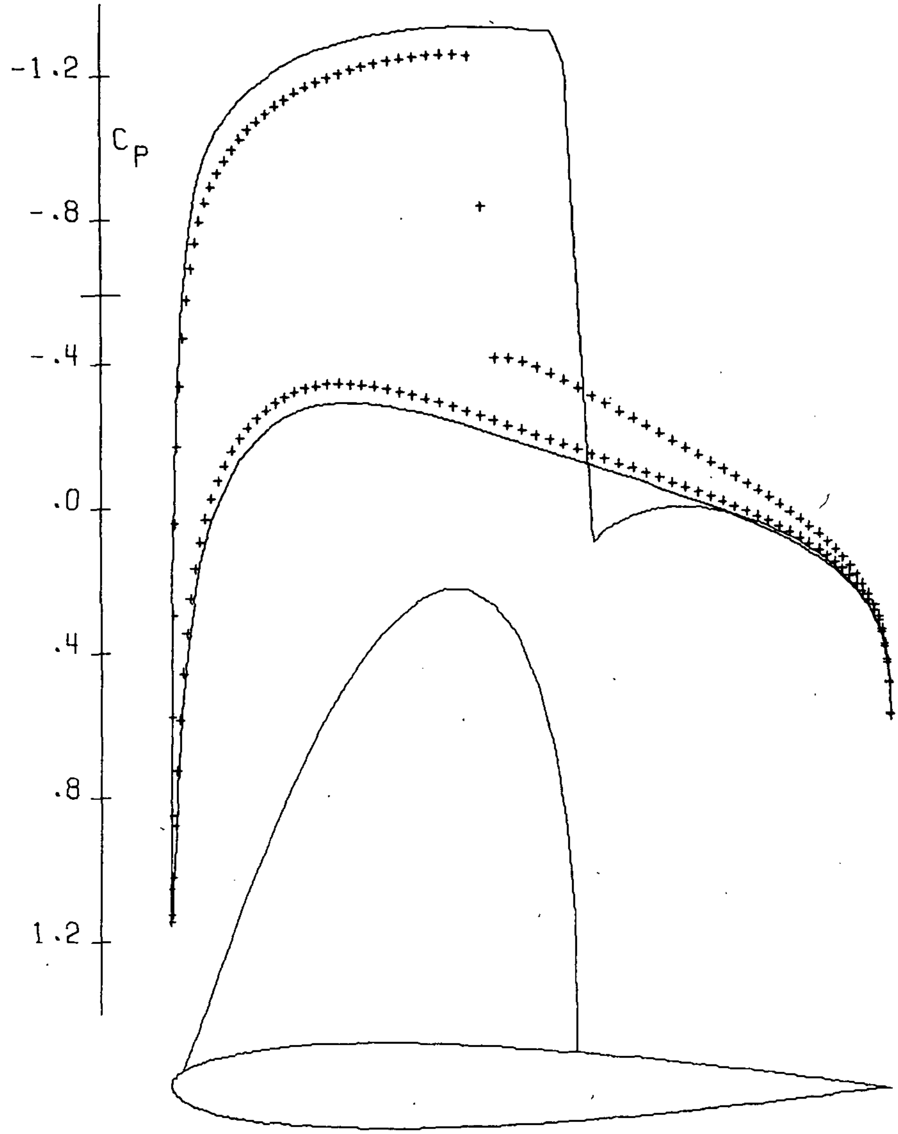
AIRFOIL 78-06-10   M*N=320*60   NCY= 400   NO VISCOSITY
— ANALYSIS   M=.780   ALP= 0.00   CL= .591   CD=.0005
+ DESIGN   M=.780   ALP= 0.00   CL= .591   CD=.0000



AIRFOIL 78-06-10 M*N=320*60 NCY= 400 NO VISCOSITY

— QUASI CON M=.780 ALP= 0.00 CL= .593 CD=.0002

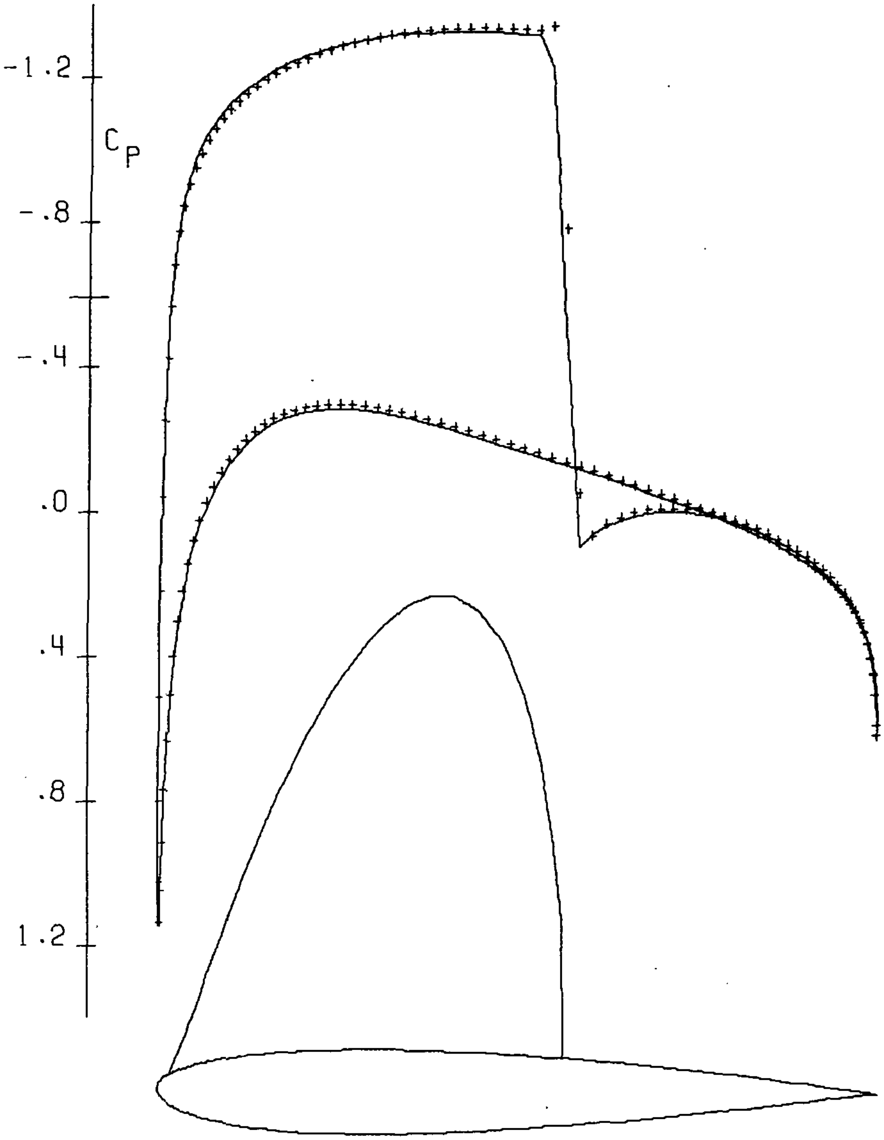
\+ DESIGN M=.780 ALP= 0.00 CL= .591 CD=.0000



NACA 0012 M*N=160*30 NCY= 800 NO VISCOSITY

— QUASI CON M=.750 ALP= 2.00 CL= .580 CD=.0156

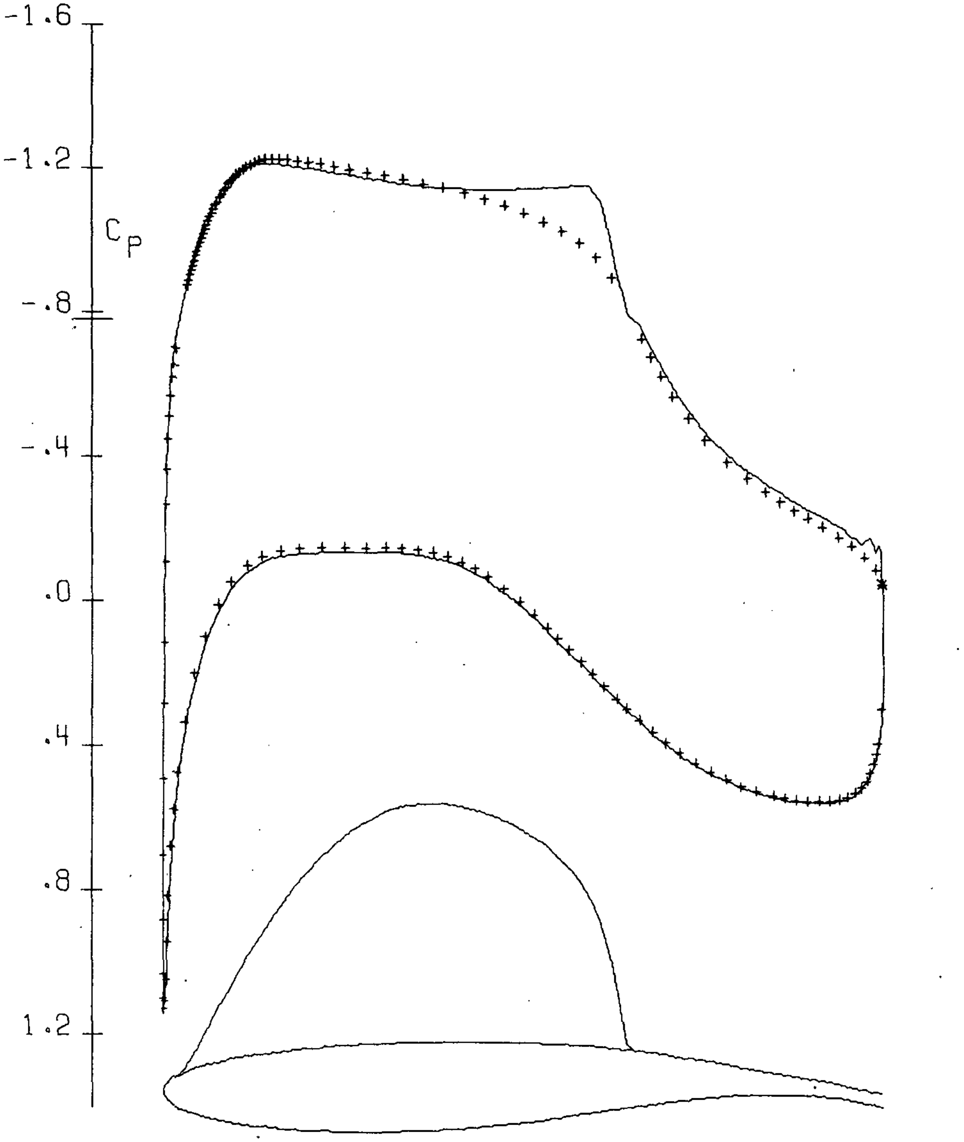
\+ OLD MURMAN M=.750 ALP= 2.00 CL= .444 CD=.0139



NACA 0012 M*N=160*30 NCY= 800 NO VISCOSITY

— QUASI CON M=.750 ALP= 2.00 CL= .580 CD=.0156

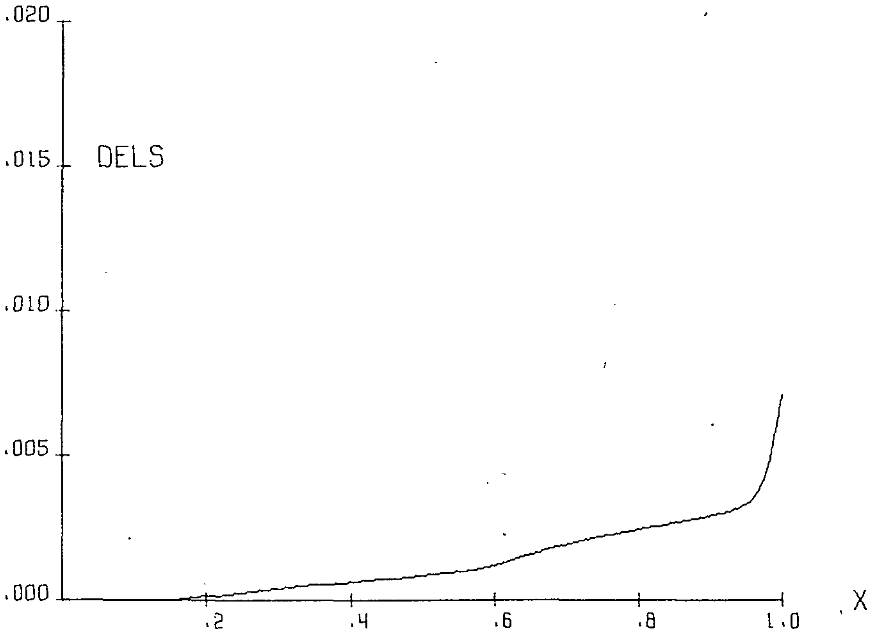
+ FULLY CON M=.750 ALP= 2.00 CL= .581 CD=.0176



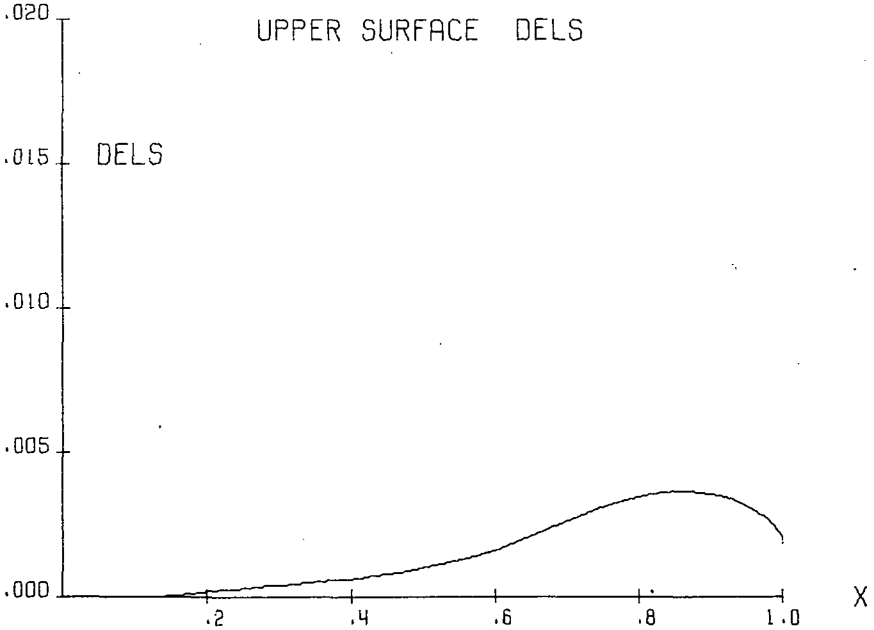
AIRFOIL 70-10-13 M*N=160*30 NCY= 400 R=20 MILLION

— THEORY M=.700 ALP= 0.00 CL=1.034 CD=.0082

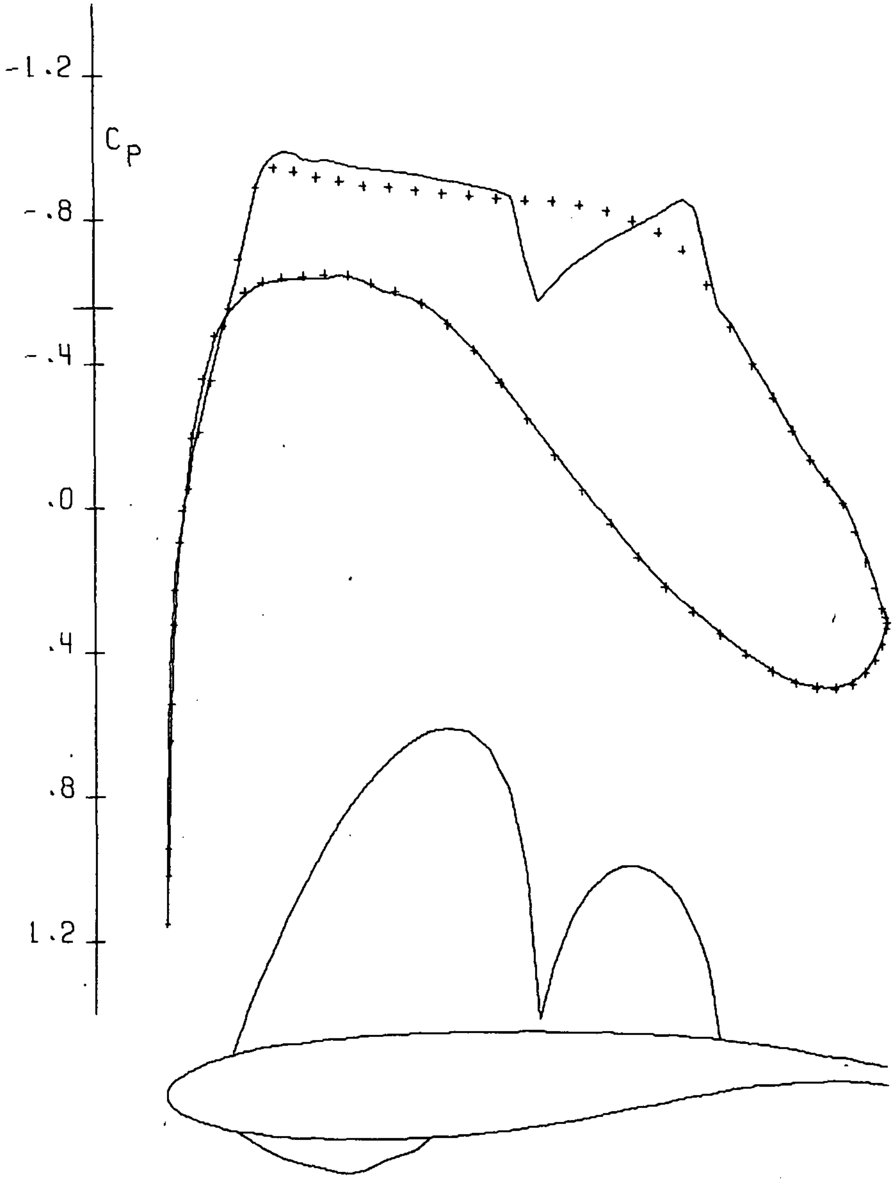
\+ DESIGN M=.700 ALP= 0.00 CL= .998 CD=.0000



UPPER SURFACE DELS



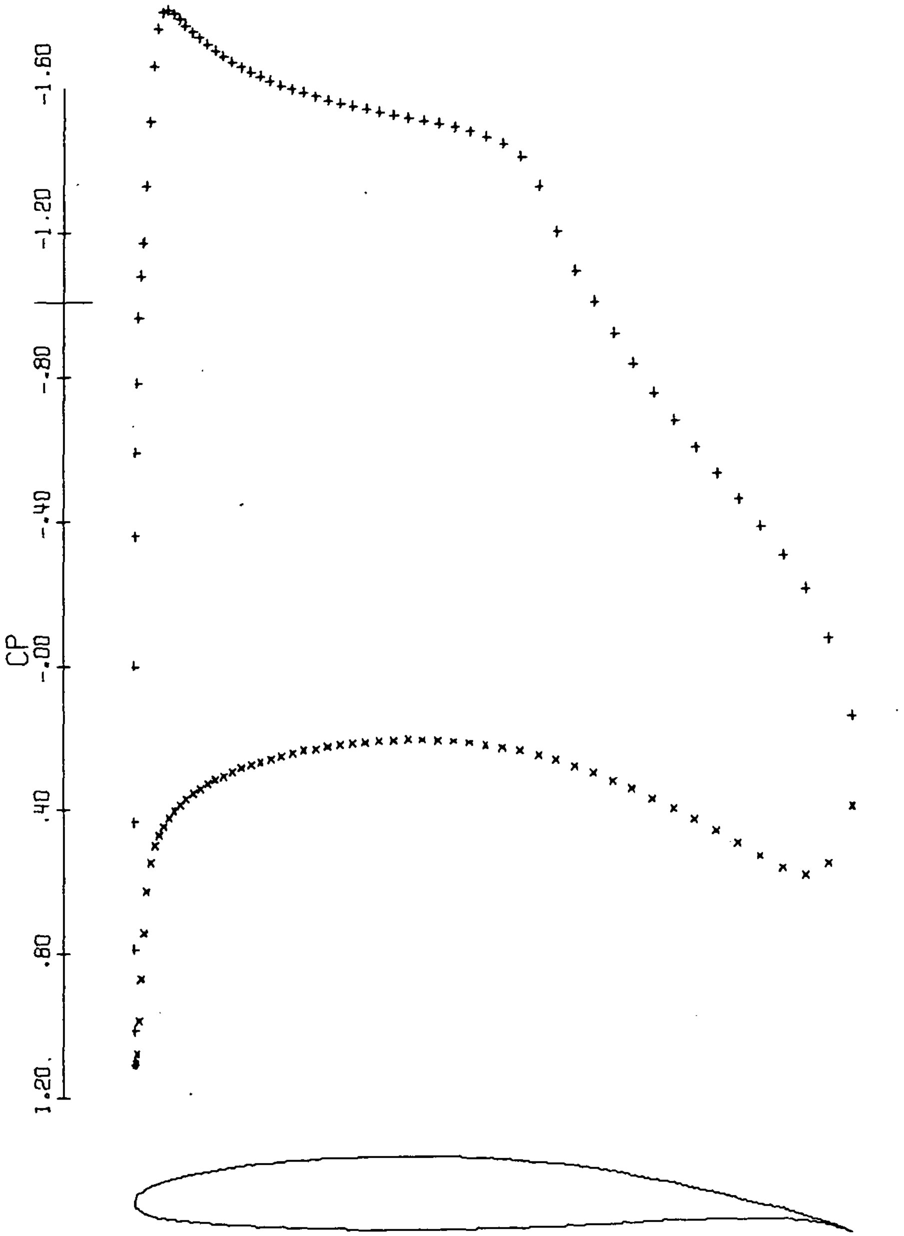
LØWER SURFACE DELS



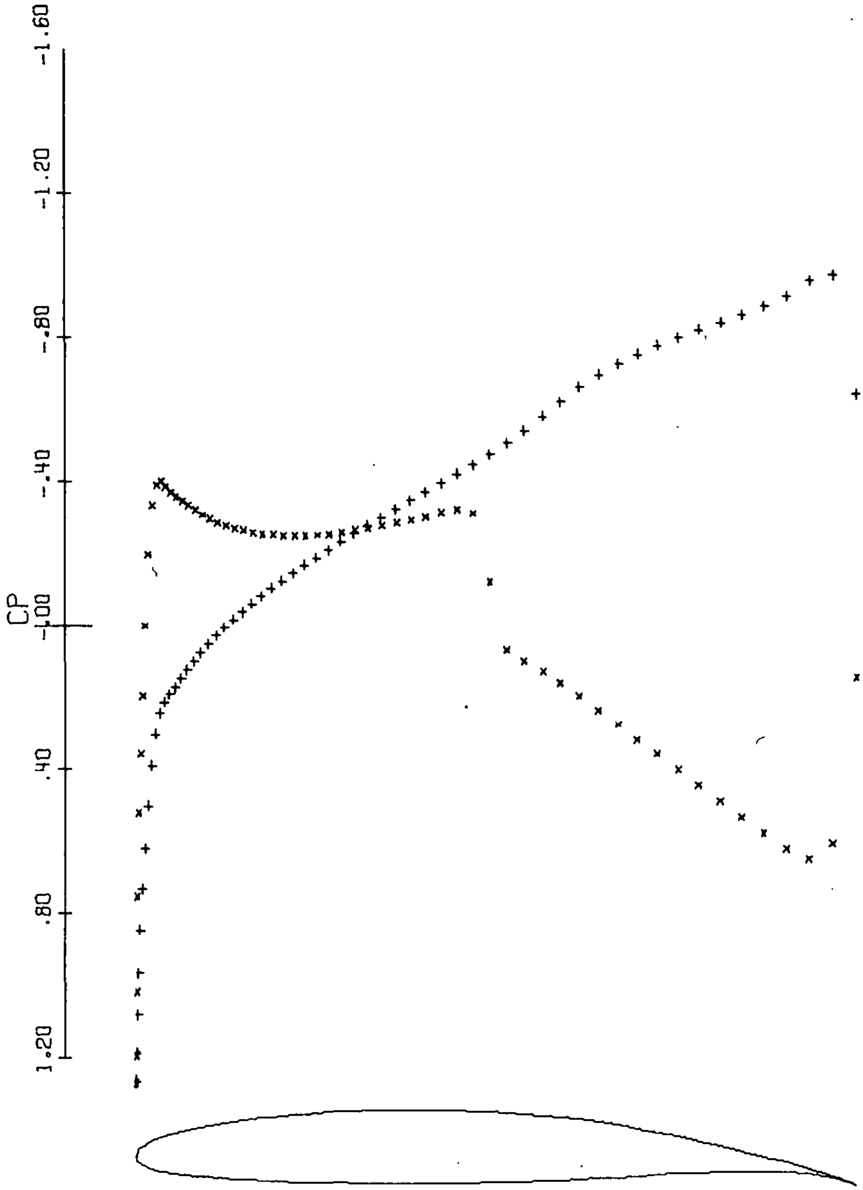
AIRFOIL 75-07-15 M=.760 CL= .499 R=20 MILLION

— FINE GRID M*N=160*30 ALP= .58 CD=.0150

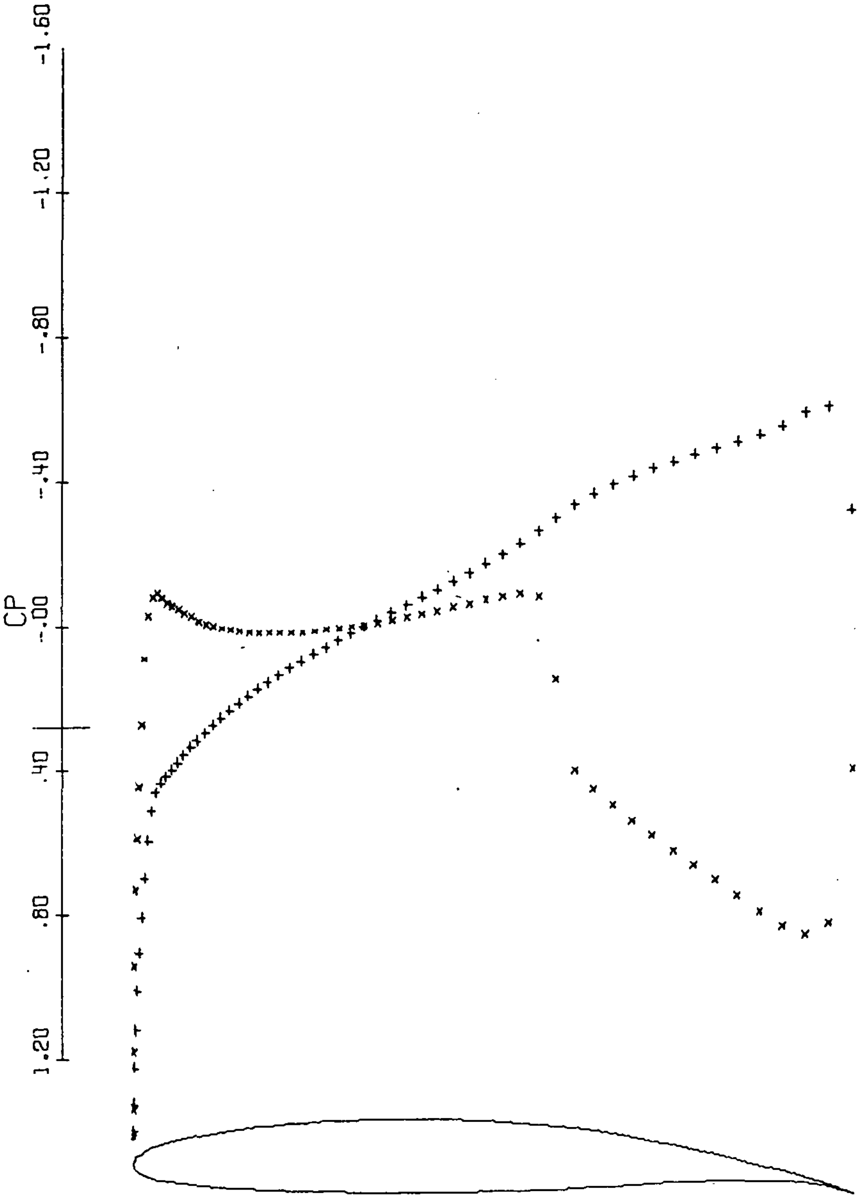
+ CRUDE GRID M*N= 80*15 ALP= .47 CD=.0138



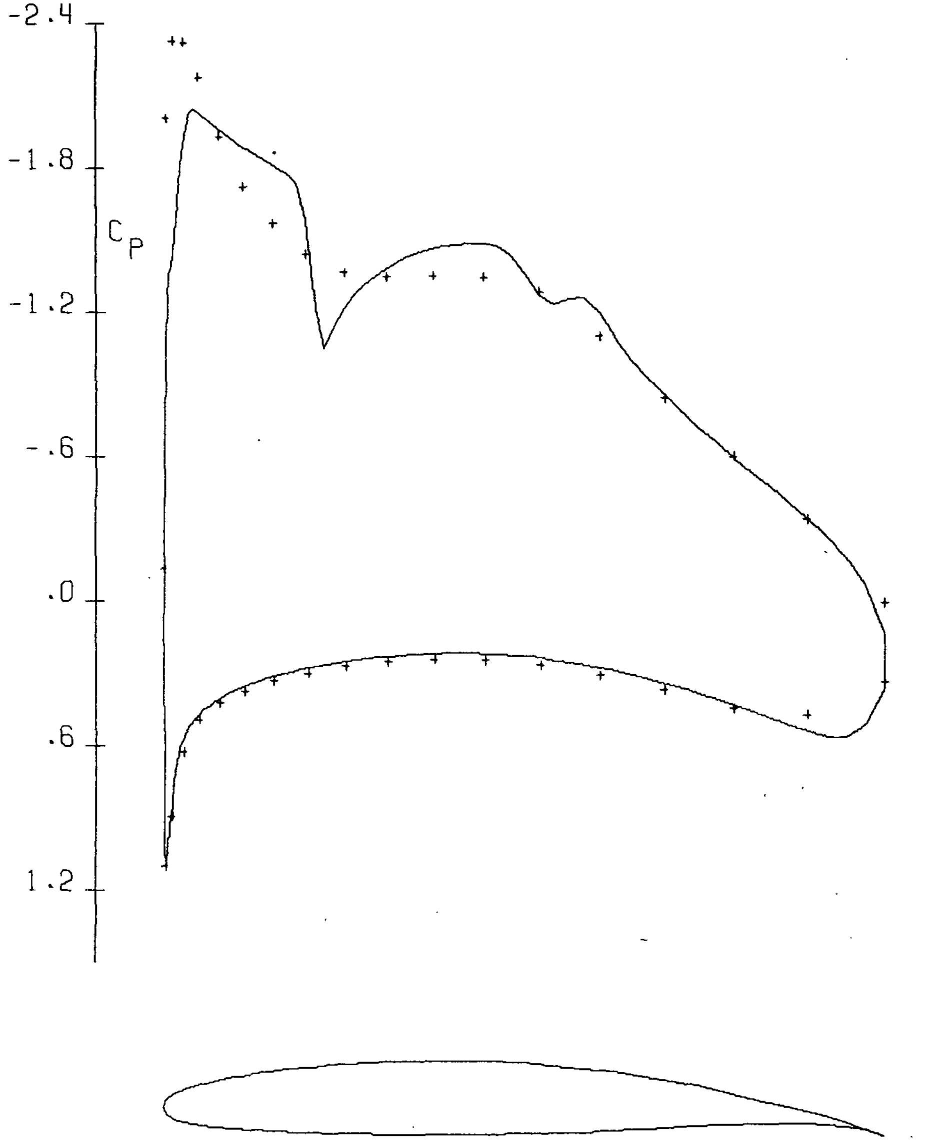
AIRFOIL 65-15-10
M = .650 ALP = 0.000
CL = 1.4866 CD = -.0001 CM = -.2474



AIRFØIL 65-15-10
M = 1.000   ALP = 0.000
CL = .5048   CD = .1154   CM = -.3198



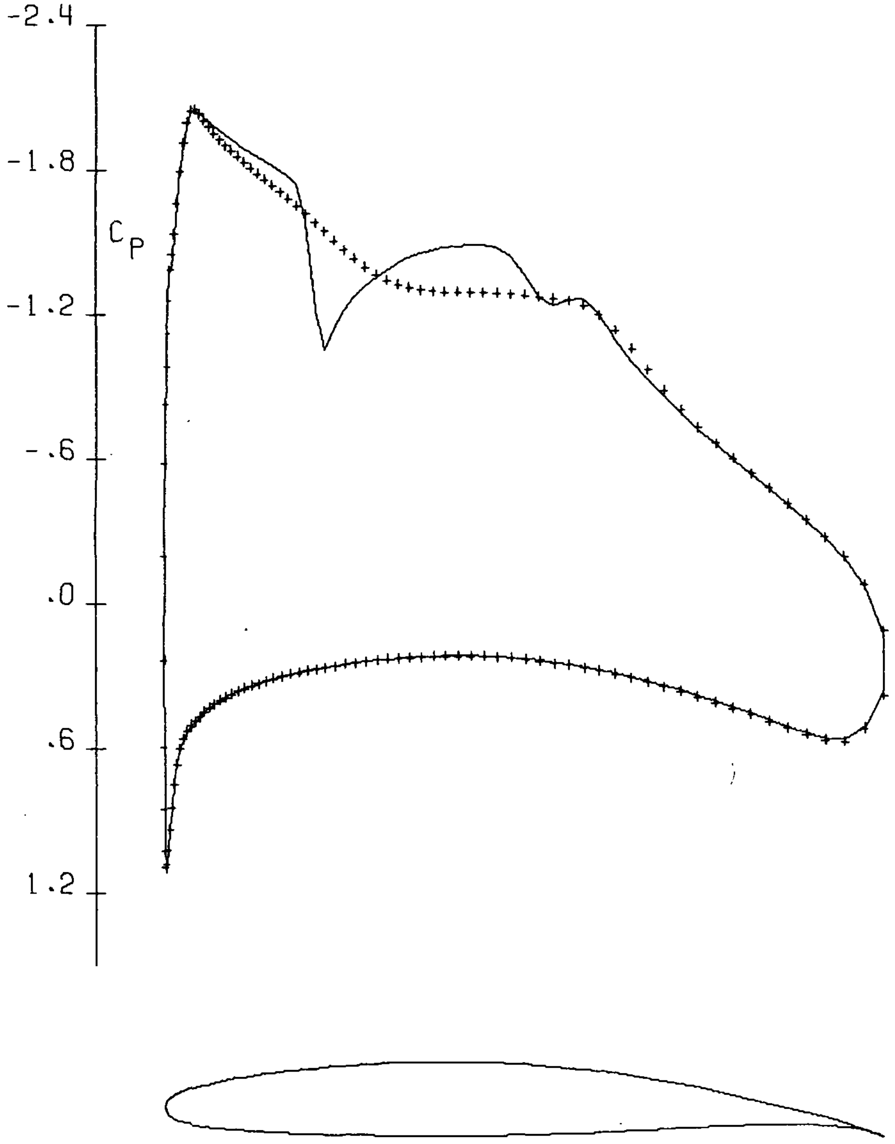
AIRFØIL 65-15-10
M = 1.200 ALP = 0.000
CL = .4074 CD = .1048 CM = -.2738



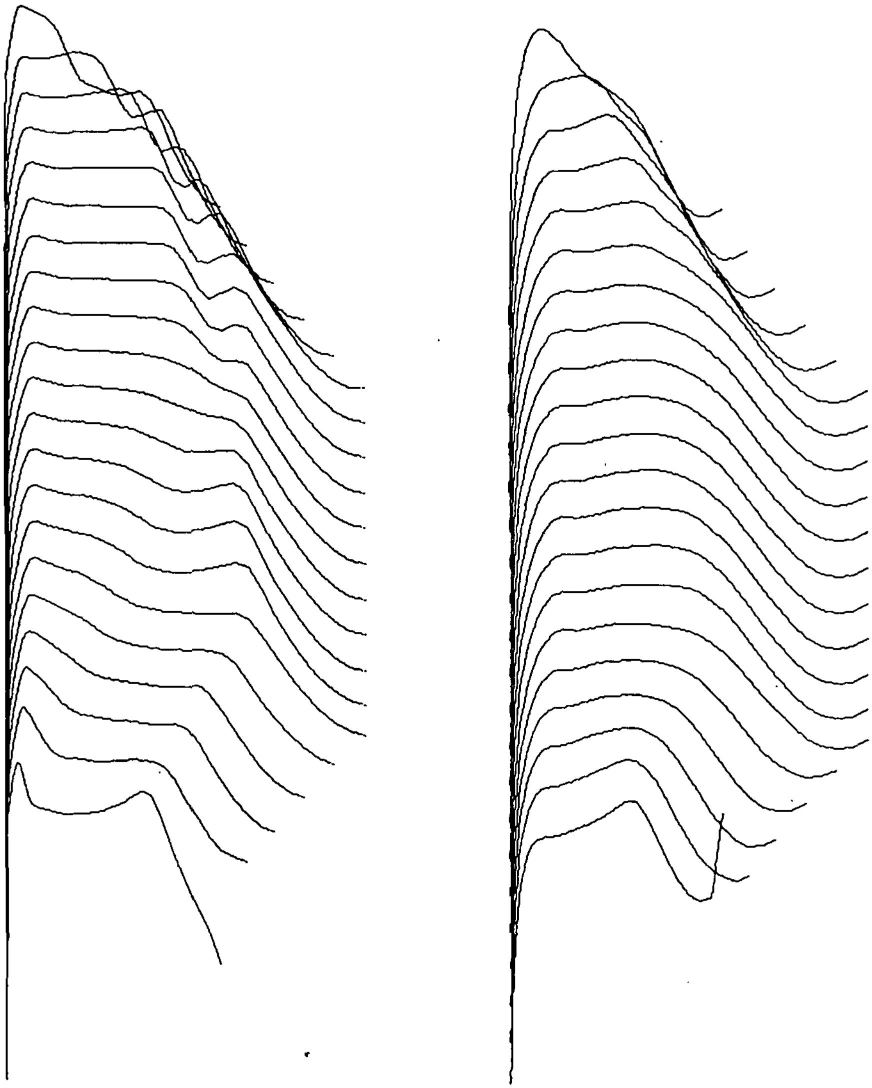
AIRFOIL 65-15-10 YAW= 0.00 M=.630 INFINITE YAWED WING

— FINE GRID L*M*N=240*32*6 CL=1.486 CD= .0041

\+ CRUDE GRID L*M*N= 60* 8*6 CL=1.481 CD=-.0185



```
AIRFOIL 65-15-10    L*M*N= 240*32*6       INFINITE YAWED WING
—  STRAIGHT      YAW=  0.00      M= .630    CL=1.486    CD=.0041
+  YAWED         YAW= 51.86      M=1.020    CL=1.480    CD=.0035
```

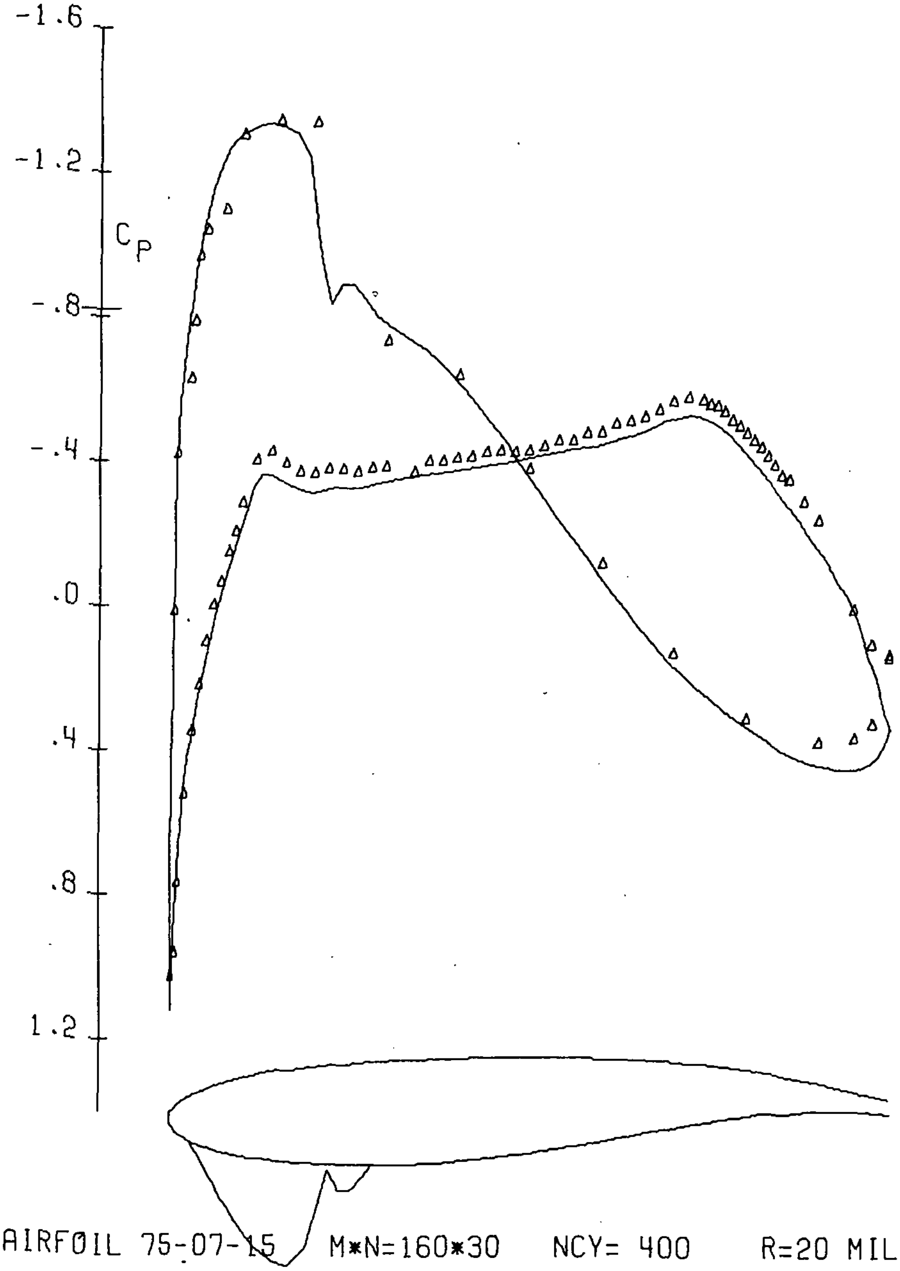


UPPER SURFACE PRESSURE LOWER SURFACE PRESSURE

79-03-12 SECTION AR= 6.0 TWIST 2 DEG
M = .830 YAW = 15.00 ALP = 1.20
L/D = 17.28 CL = .2562 CD = .0148

3. Comparison of Experimental Data with the Boundary Layer Correction

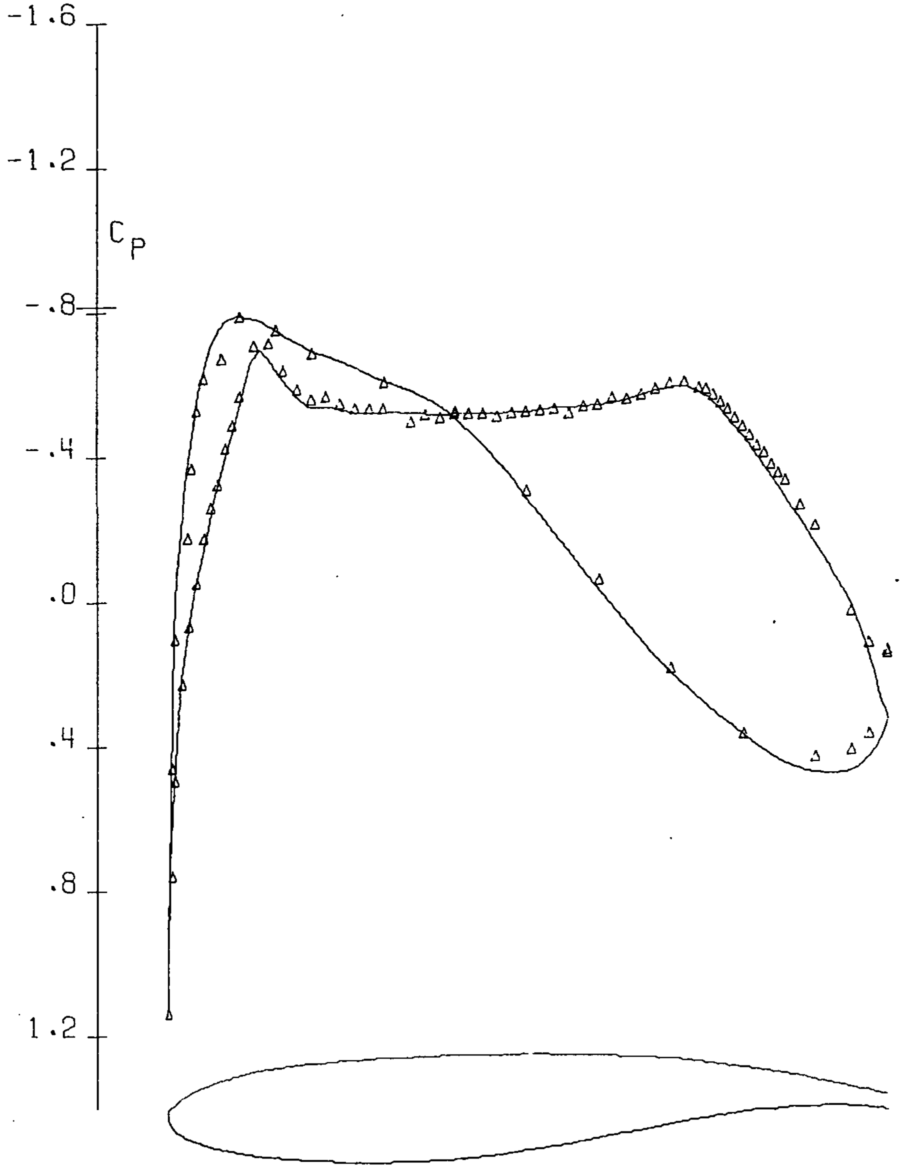
We present experimental data from the National Aeronautical Establishment in Ottawa on Airfoils 75-07-15 and 75-06-12 (cf. [7,8, 9]) and on an airfoil designed by John Dahlin at the McDonnell Douglas Corporation. We also present experimental data on Airfoil 82-06-09 obtained at Aircraft Research Associates, Ltd., in England that are British Crown Copyright, and are reproduced by permission of the Controller, R & D Establishments and Research, Ministry of Defence (PE). The experimental data is compared with numerical calculations using Program H (Chapter III, Section 5), which includes the effect of a boundary layer correction. The results for Airfoils 75-07-15 and 75-06-12, as well as for the Douglas airfoil, were found with transition set at PCH = .07, and with LSEP = 161. Those for Airfoil 82-06-09 were found with transition set at PCH = .07, but with LSEP = 153. The experimental data on pages 147 and 148 are from two different series of tests of Airfoil 75-06-12 with different tunnel porosities; they represent the closest to shock free flow that could be obtained. In most cases the agreement between theory and experiment is excellent.



AIRFOIL 75-07-15 M*N=160*30 NCY= 400 R=20 MILLION

— THEORY M=.690 ALP=-2.28 CL=-.059 CD=.0118

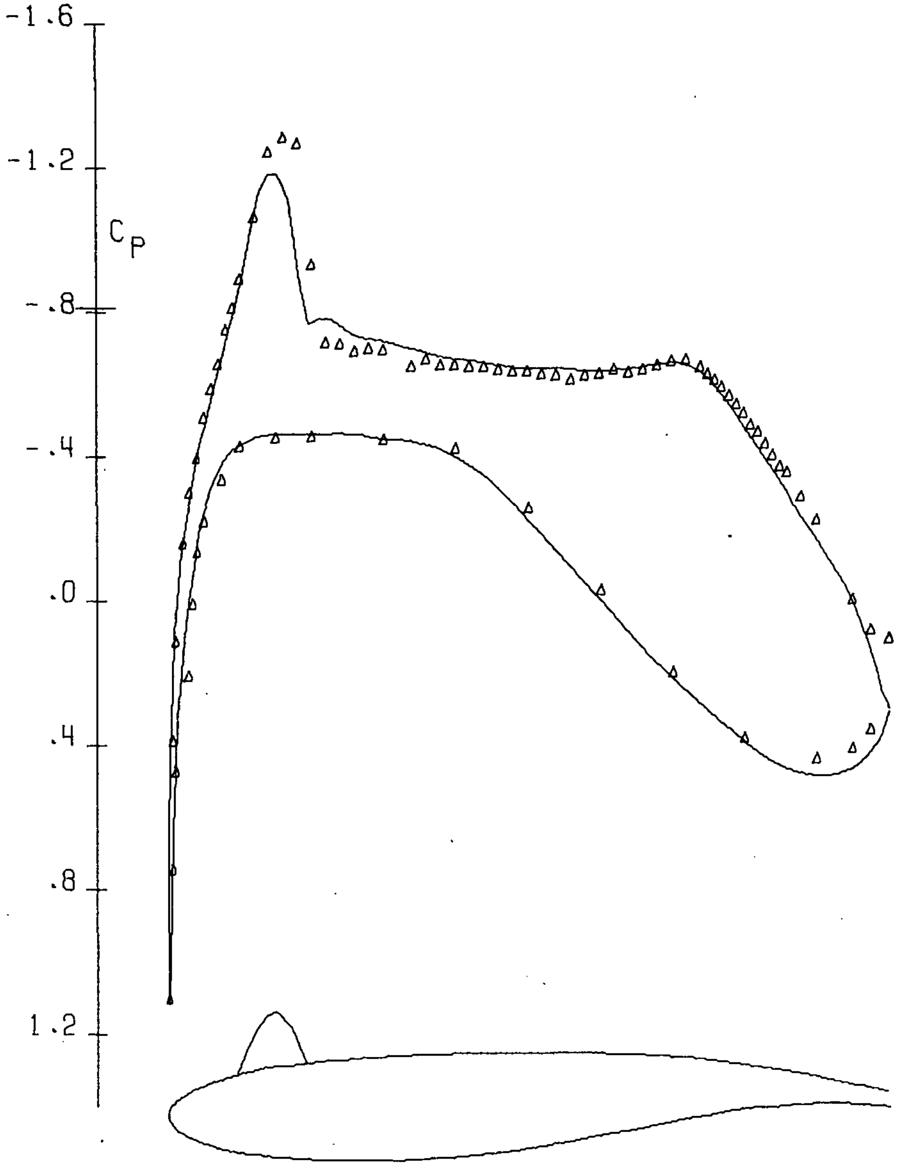
Δ EXPERIMENT M=.690 ALP=-2.07 CL=-.059 CD=.0128



AIRFOIL 75-07-15 M*N=160*30 NCY= 400 R=20 MILLION

— THEORY M=.691 ALP= -.66 CL= .237 CD=.0095

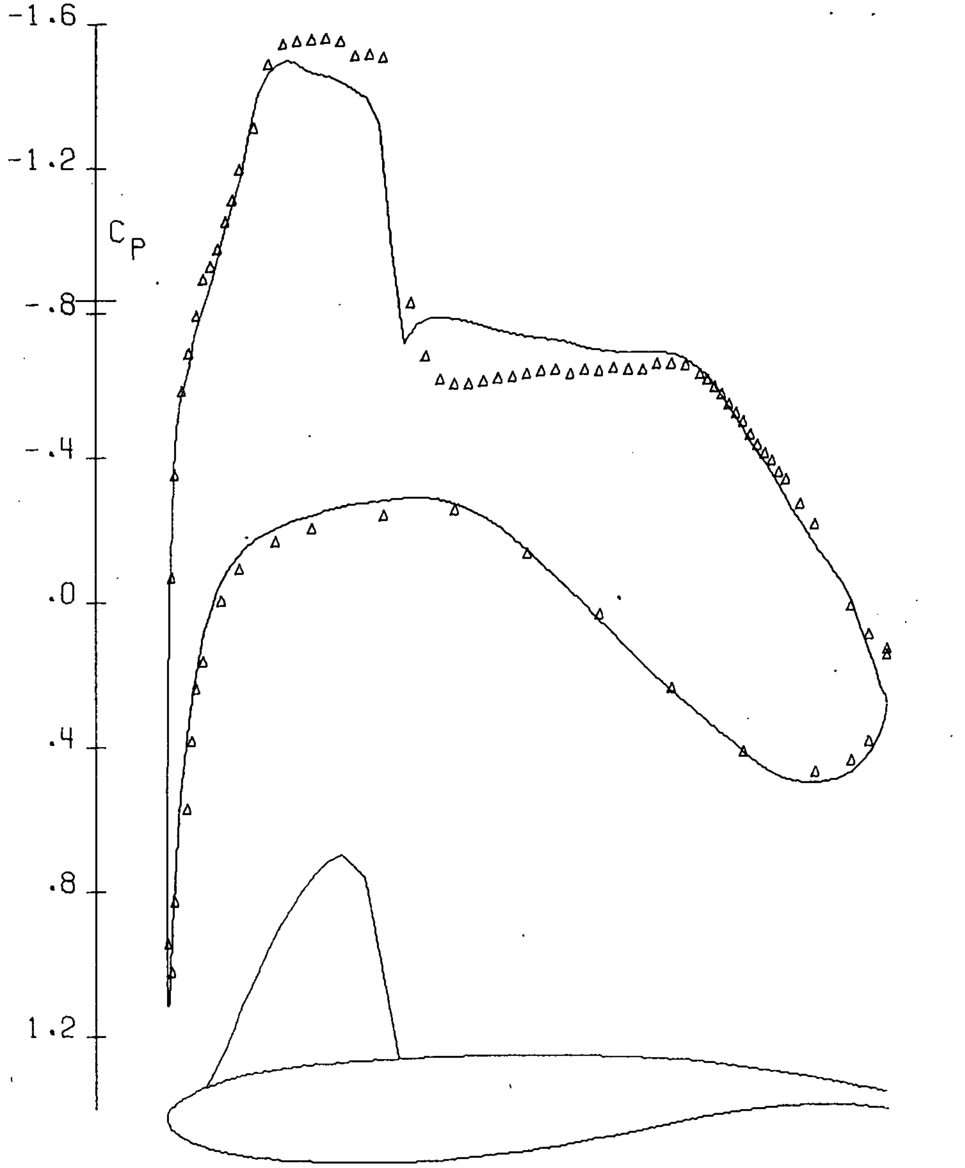
△ EXPERIMENT M=.691 ALP= -.02 CL= .237 CD=.0102



AIRFOIL 75-07-15 M*N=160*30 NCY= 400 R=20 MILLION

— THEORY M=.692 ALP= .87 CL= .511 CD=.0104

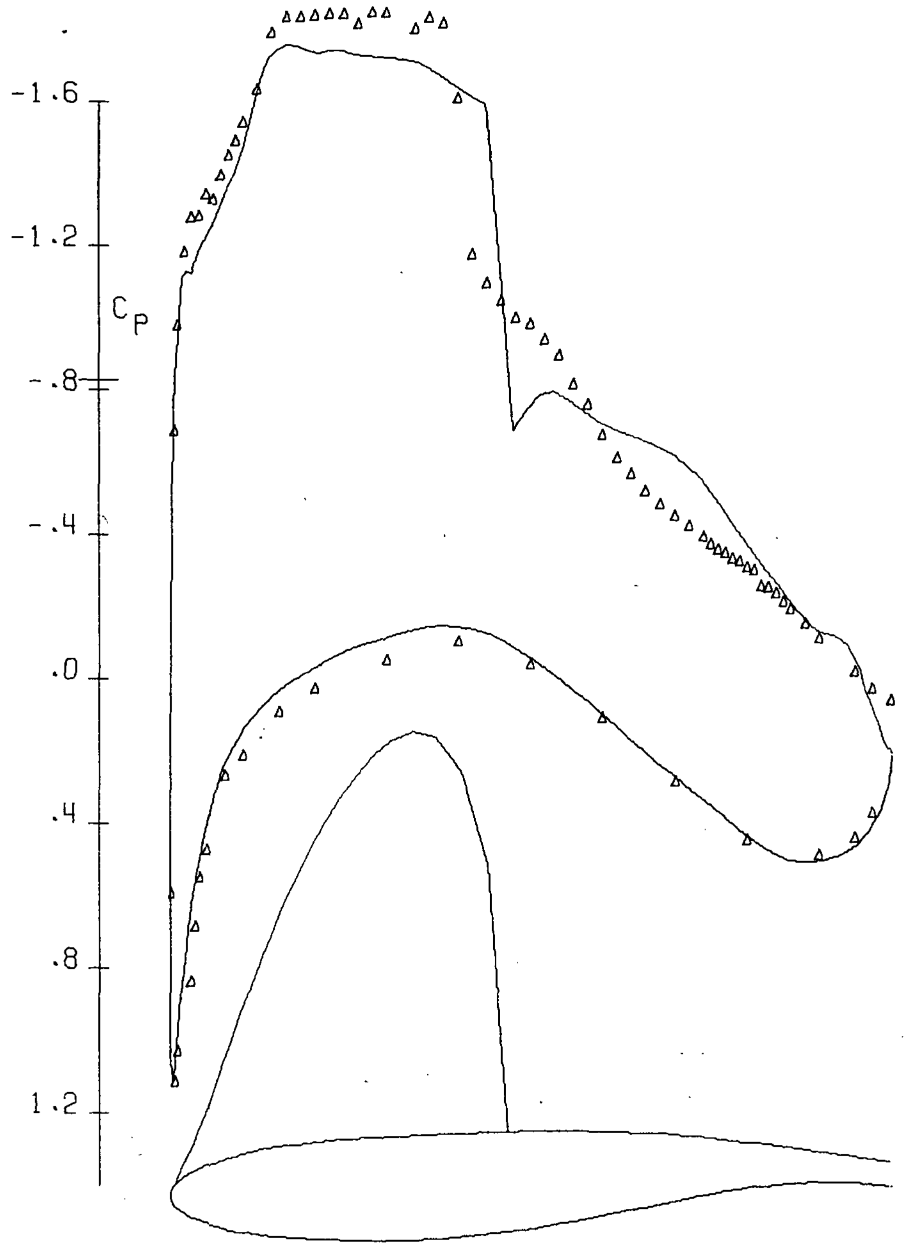
△ EXPERIMENT M=.692 ALP= 2.03 CL= .511 CD=.0102



AIRFOIL 75-07-15   M*N=160*30   NCY= 400   R=20 MILLION

— THEORY   M=.687   ALP= 2.61   CL= .809   CD=.0173

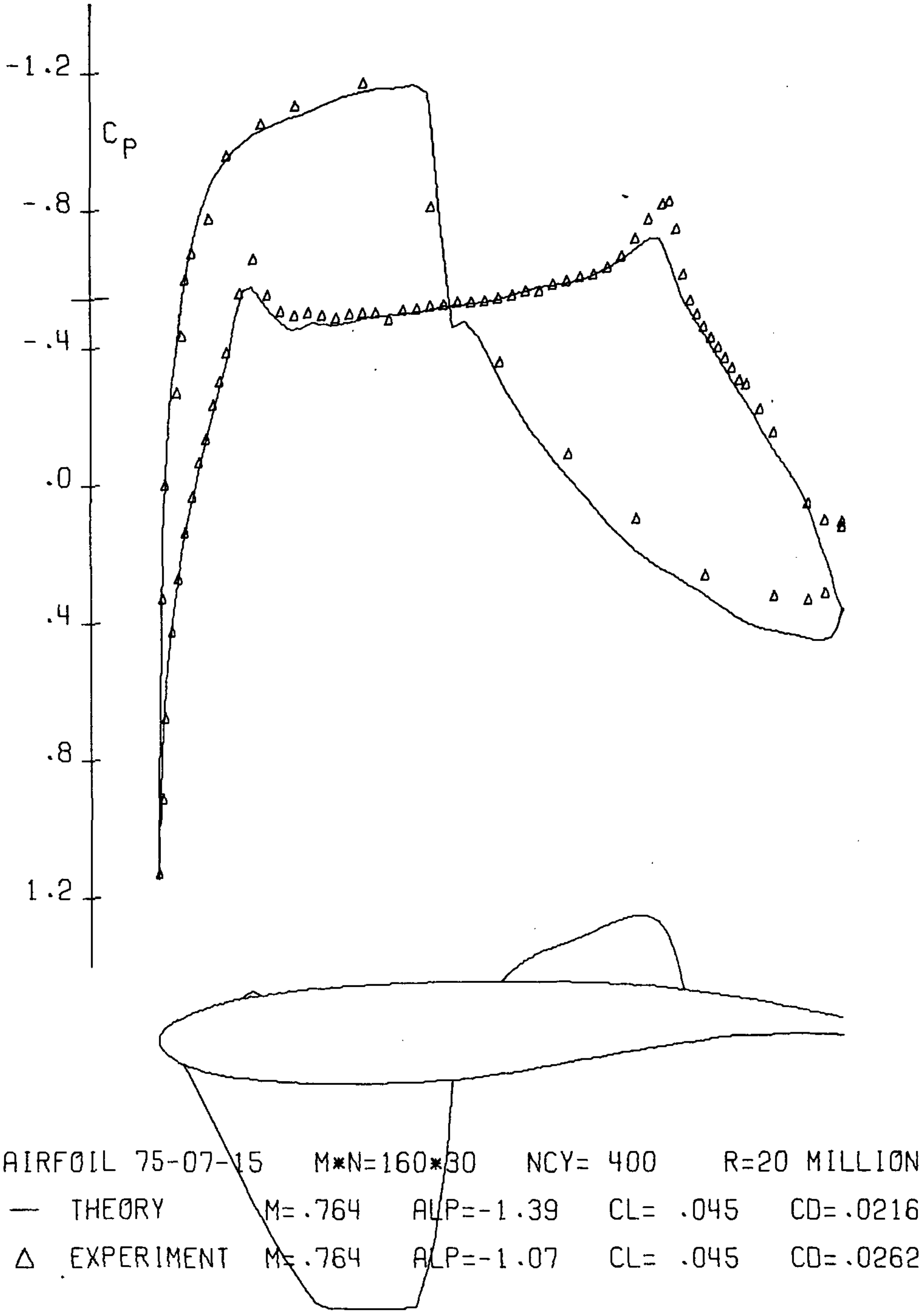
△ EXPERIMENT   M=.687   ALP= 4.09   CL= .809   CD=.0170

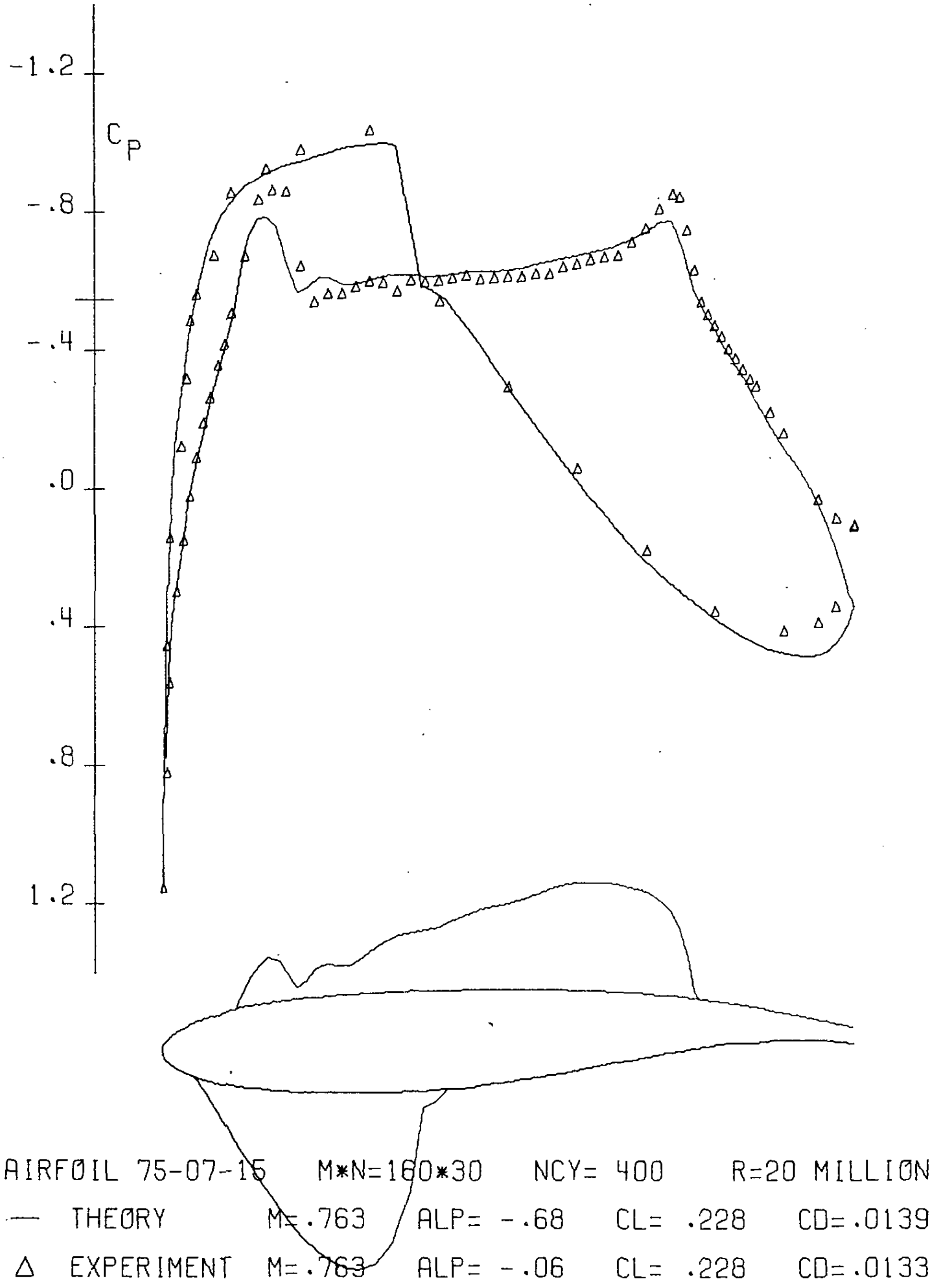


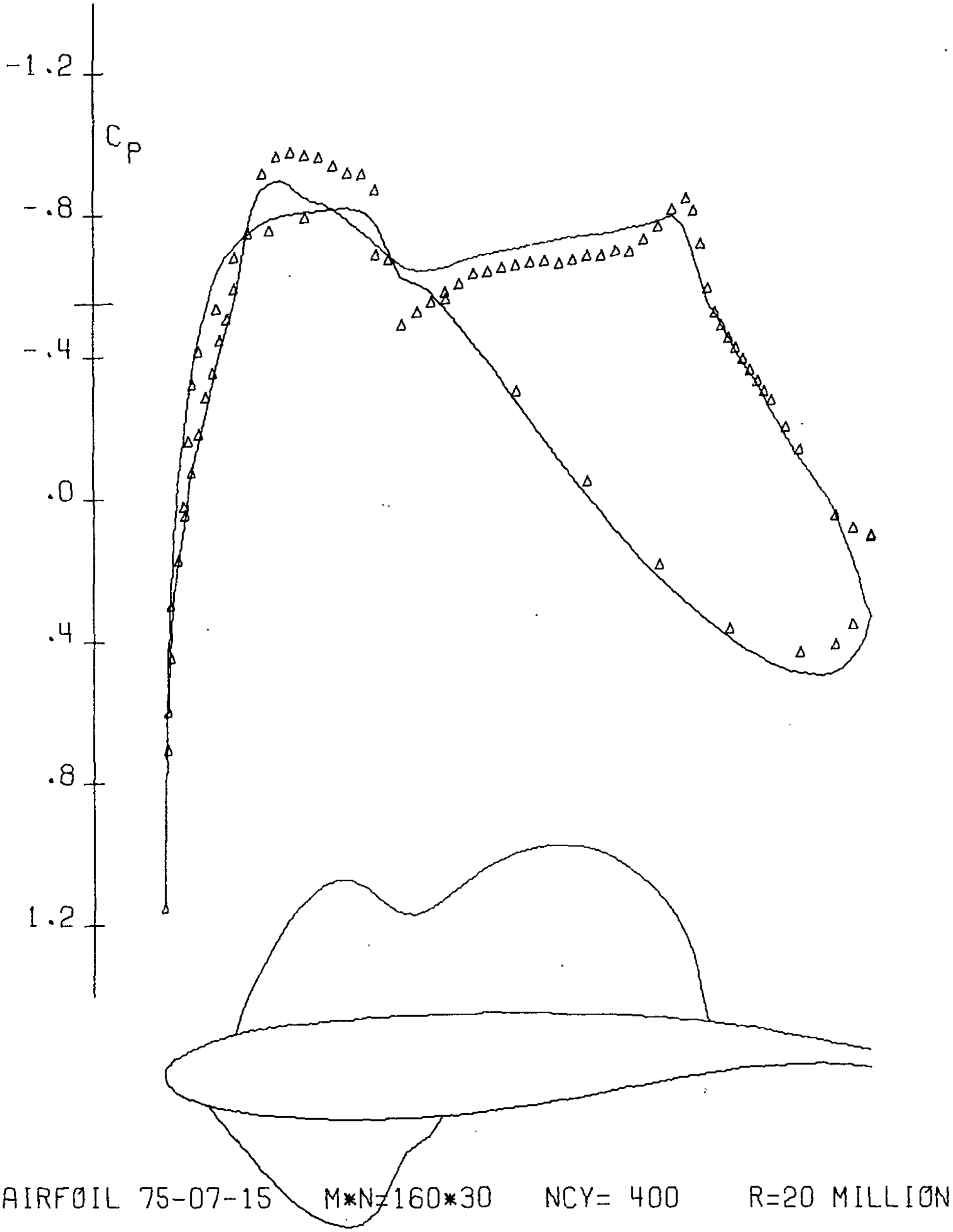
AIRFOIL 75-07-15 M*N=160*30 NCY= 400 R=20 MILLION

— THEORY M=.688 ALP= 5.01 CL=1.148 CD=.0486

△ EXPERIMENT M=.688 ALP= 7.17 CL=1.148 CD=.0502


-1.2
C_P
-.8
-.4
.0
.4
.8
1.2
AIRFOIL 75-07-15 M*N=160*30 NCY= 400 R=20 MILLION
— THEORY M=.764 ALP=-1.39 CL= .045 CD=.0216
△ EXPERIMENT M=.764 ALP=-1.07 CL= .045 CD=.0262

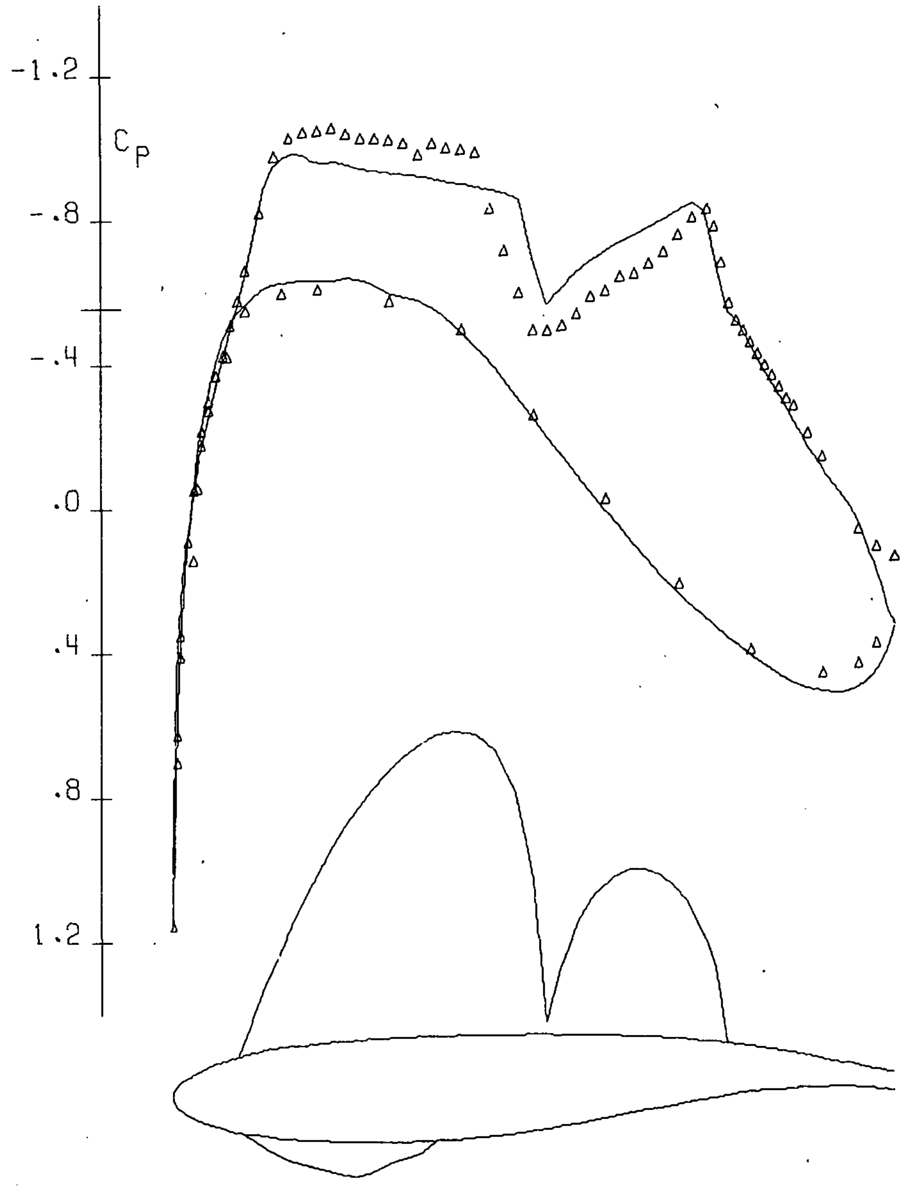

-1.2
C_P
-.8
-.4
.0
.4
.8
1.2
AIRFOIL 75-07-15 M*N=160*30 NCY= 400 R=20 MILLION
— THEORY M=.763 ALP= -.68 CL= .228 CD=.0139
Δ EXPERIMENT M=.763 ALP= -.06 CL= .228 CD=.0133



AIRFOIL 75-07-15 M*N=160*30 NCY= 400 R=20 MILLION

— THEORY M=.762 ALP= -.09 CL= .362 CD=.0122

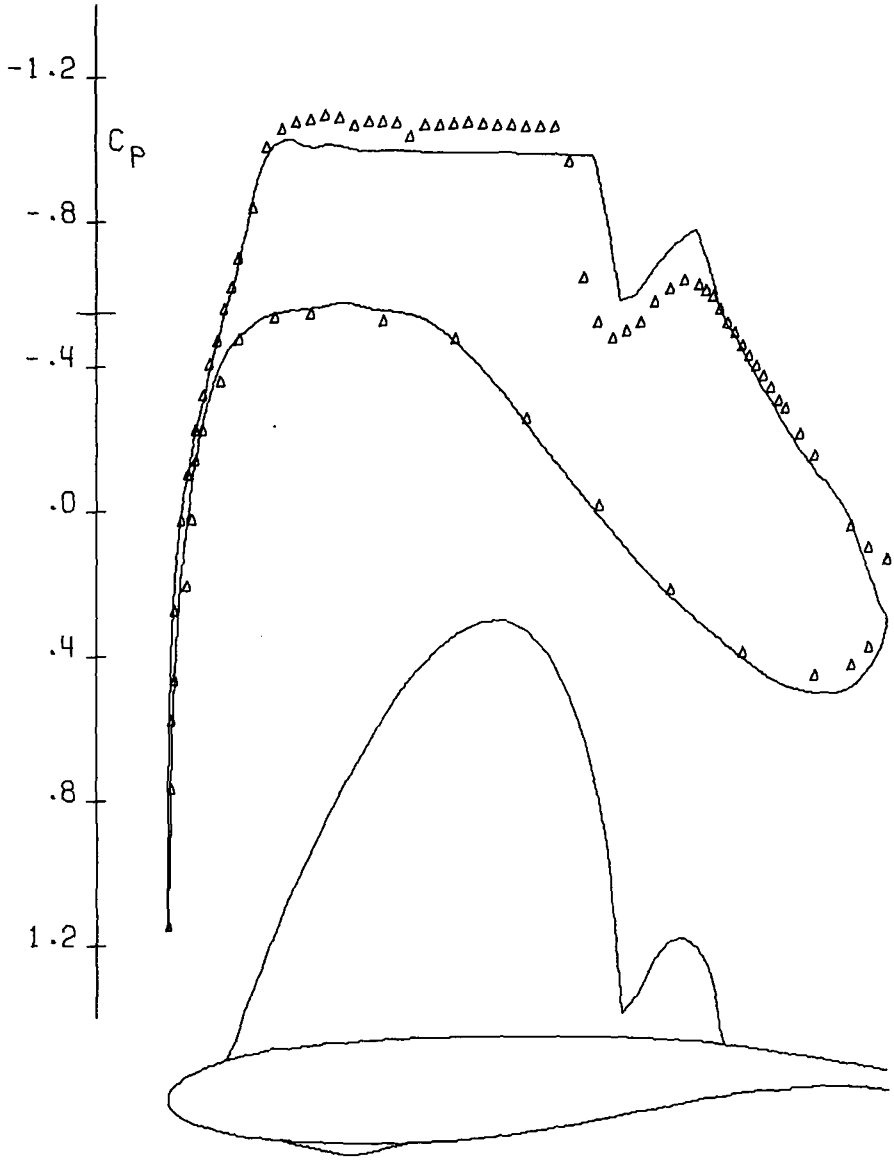
△ EXPERIMENT M=.762 ALP= .82 CL= .362 CD=.0121



AIRFOIL 75-07-15   M*N=160*30   NCY= 400   R=20 MILLION

— THEORY   M=.760   ALP= .58   CL= .499   CD=.0150

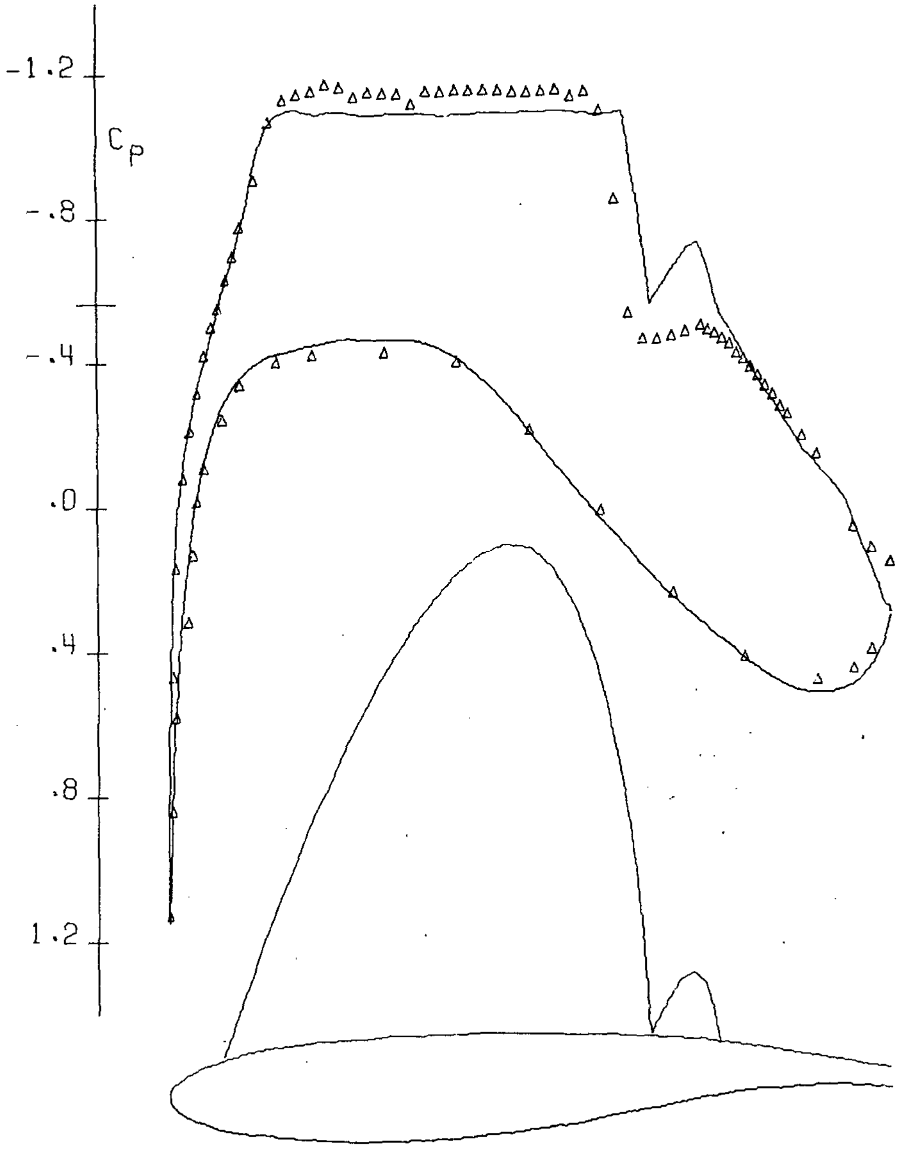
△ EXPERIMENT   M=.760   ALP= 1.55   CL= .499   CD=.0122



AIRFOIL 75-07-15 M*N=160*30 NCY= 400 R=20 MILLION

— THEORY M=.763 ALP= 1.03 CL= .584 CD=.0190

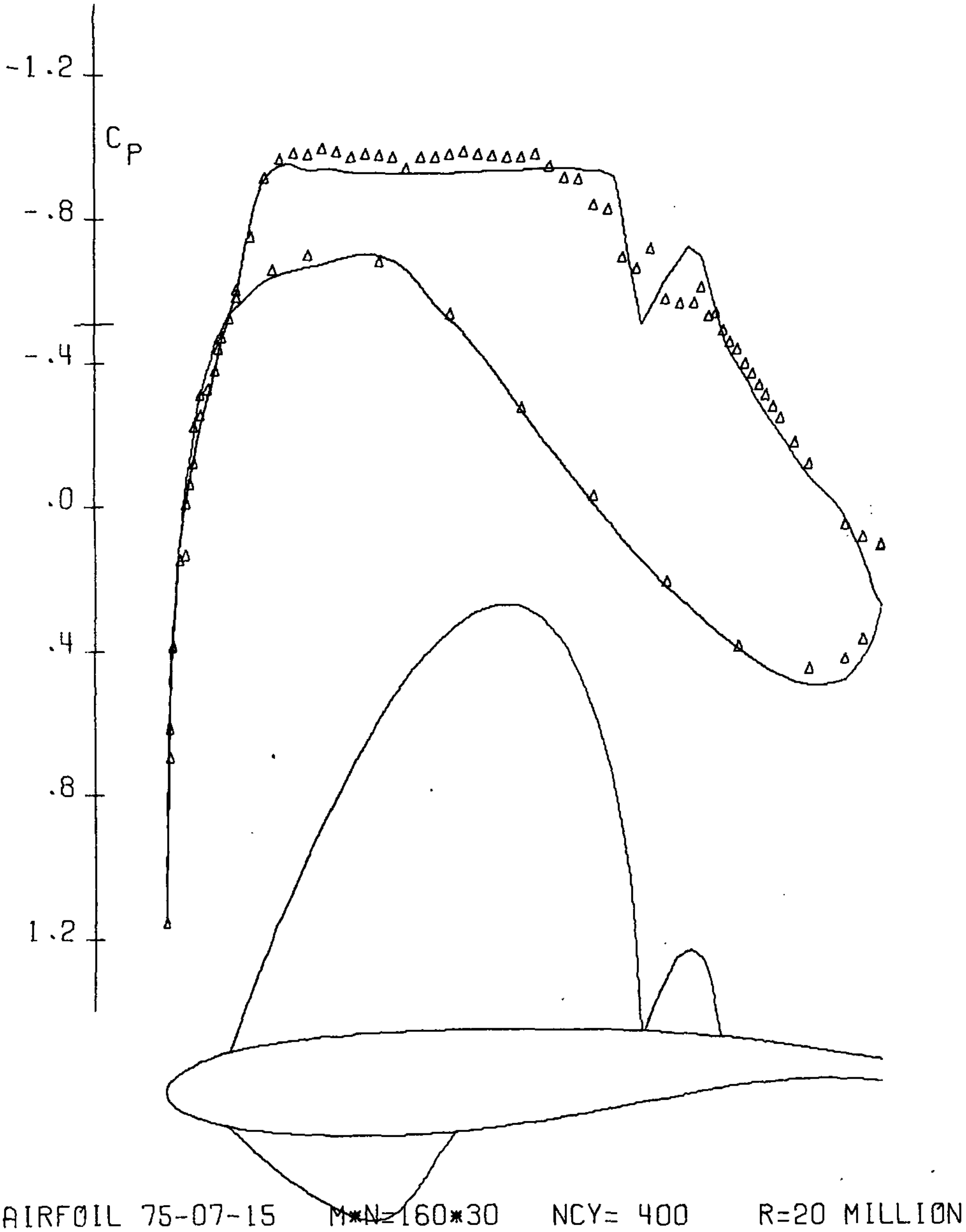
Δ EXPERIMENT M=.763 ALP= 2.01 CL= .584 CD=.0128



AIRFOIL 75-07-15 M*N=160*30 NCY= 400 R=20 MILLION

— THEORY M=.758 ALP= 1.63 CL= .706 CD=.0247

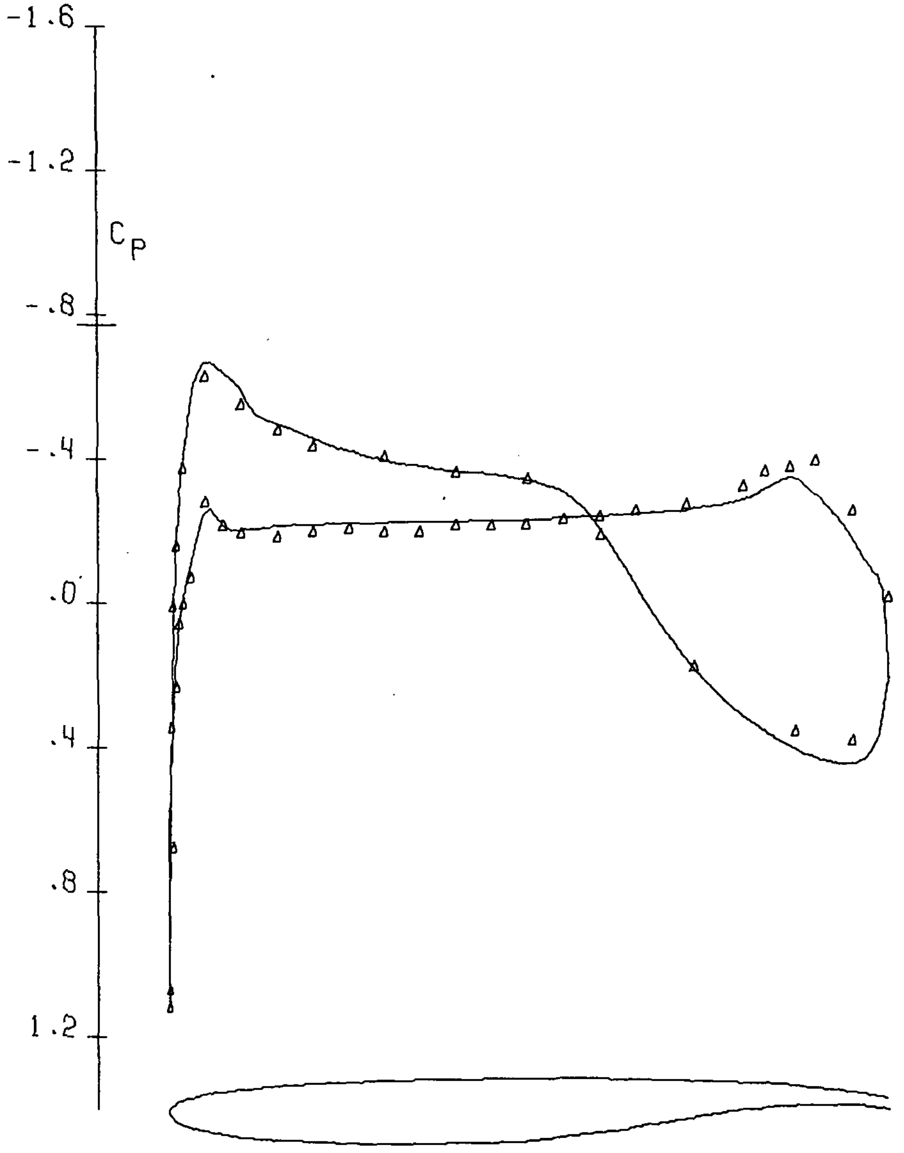
△ EXPERIMENT M=.758 ALP= 2.68 CL= .706 CD=.0163



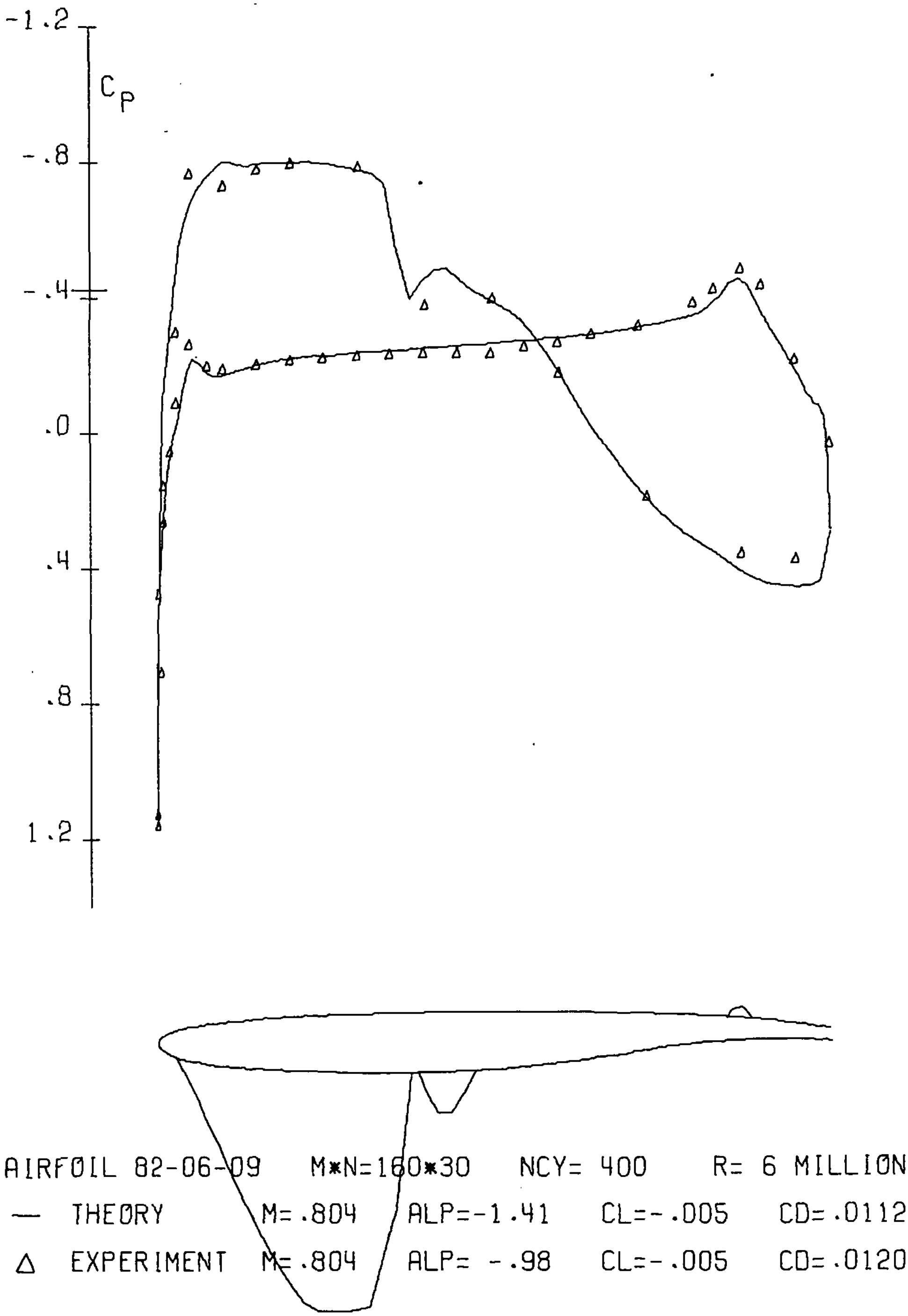
AIRFOIL 75-07-15 M*N=160*30 NCY= 400 R=20 MILLION

— THEORY M=.775 ALP= .85 CL= .494 CD=.0190

Δ EXPERIMENT M=.778 ALP= 1.44 CL= .494 CD=.0115



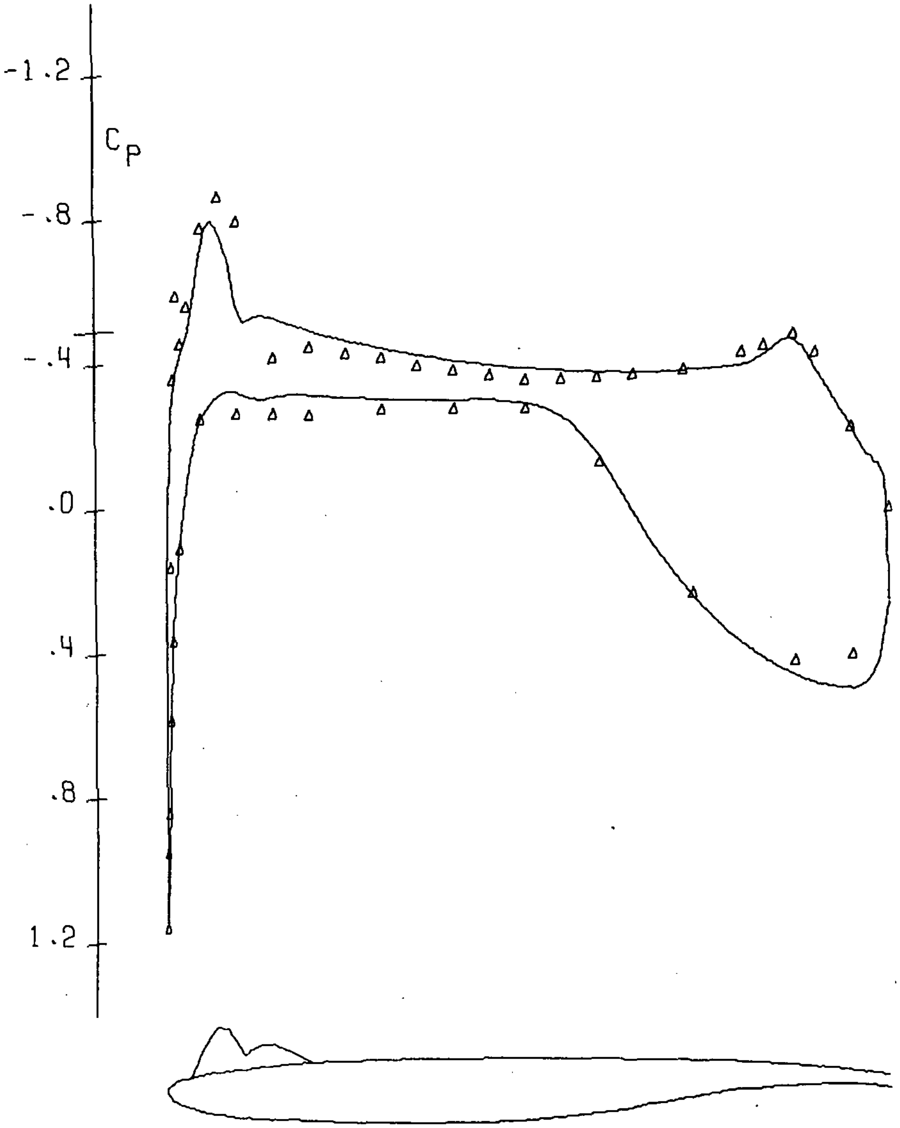
AIRFOIL 82-06-09   M*N=160*30   NCY= 400   R= 6 MILLION
— THEORY   M=.702   ALP=-1.15   CL= .070   CD=.0072
Δ EXPERIMENT   M=.702   ALP=-1.00   CL= .070   CD=.0097



AIRFOIL 82-06-09   M*N=160*30   NCY= 400   R= 6 MILLION

— THEORY   M=.804   ALP=-1.41   CL=-.005   CD=.0112

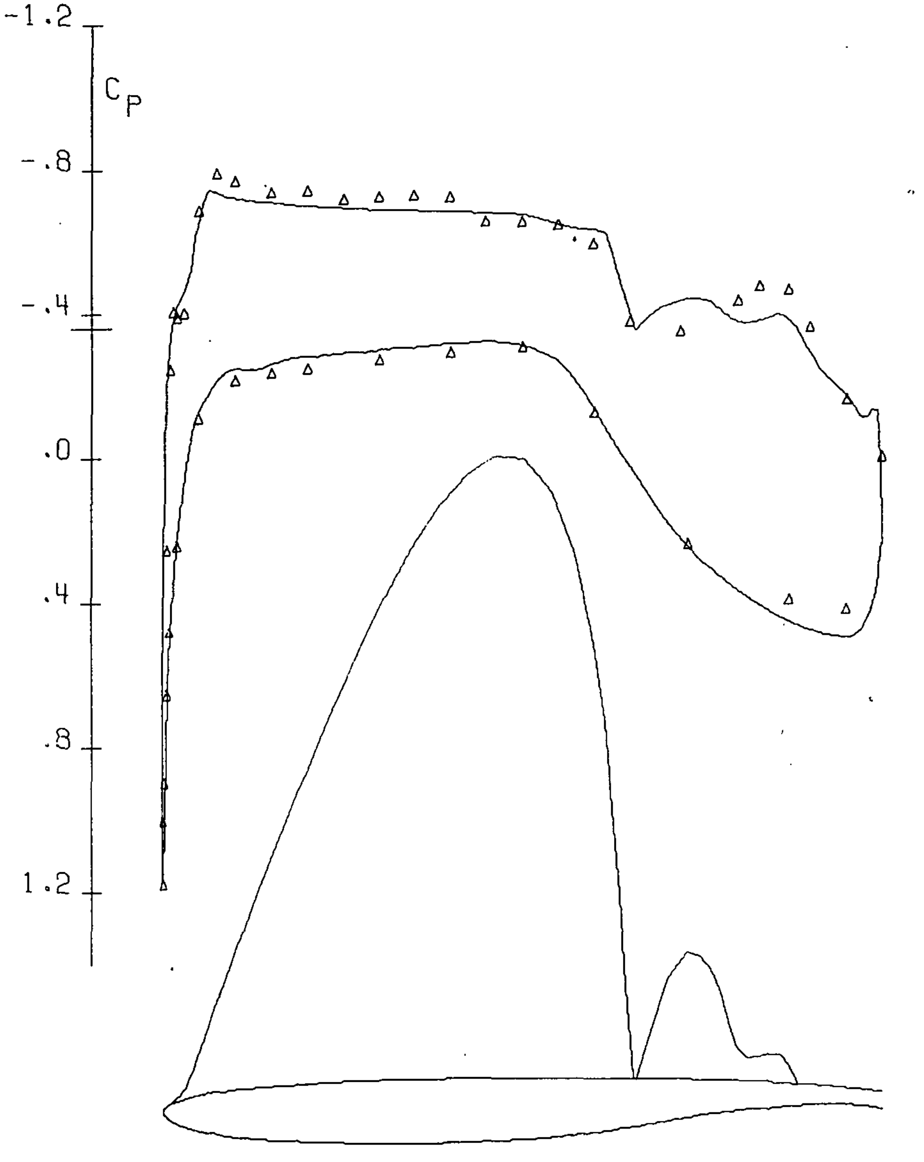
Δ EXPERIMENT   M=.804   ALP= -.98   CL=-.005   CD=.0120



AIRFOIL 82-06-09   M*N=160*30   NCY= 400   R= 6 MILLION

— THEORY   M=.781   ALP= .18   CL= .377   CD=.0073

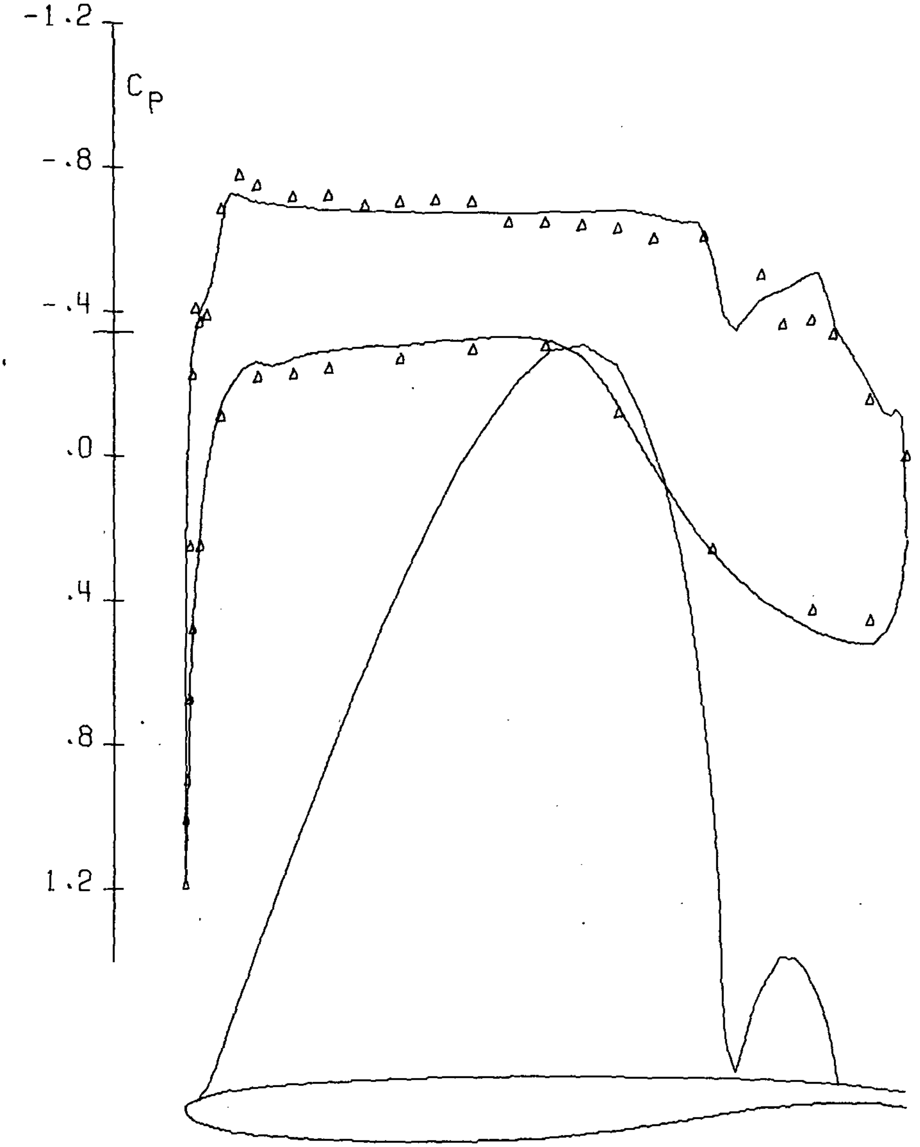
Δ EXPERIMENT   M=.781   ALP= 1.00   CL= .377   CD=.0104



AIRFOIL 82-06-09 M*N=160*30 NCY= 400 R= 3 MILLION

— THEORY M=.827 ALP= .92 CL= .515 CD=.0149

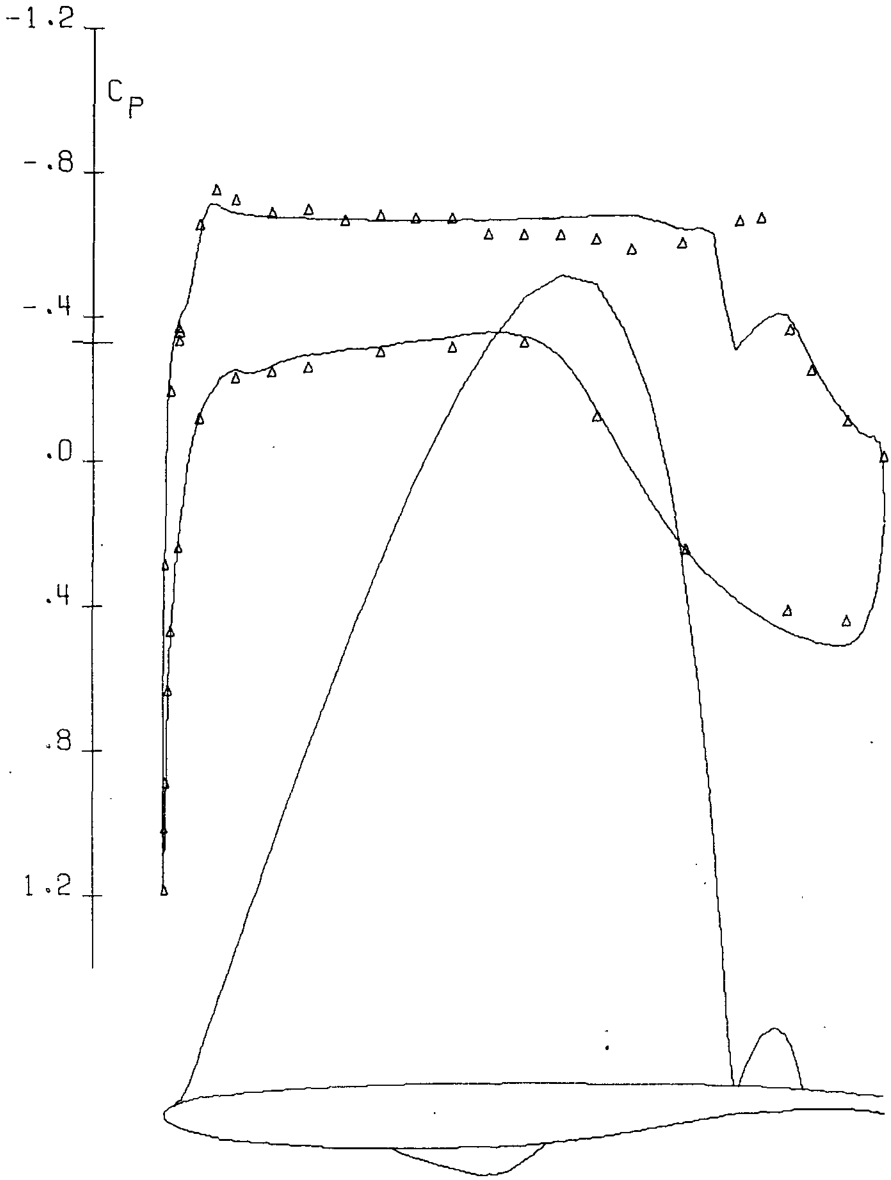
Δ EXPERIMENT M=.827 ALP= 1.51 CL= .515 CD=.0120



AIRFOIL 82-06-09 M*N=160*30 NCY= 400 R= 7 MILLION

— THEORY M=.833 ALP= .82 CL= .551 CD=.0150

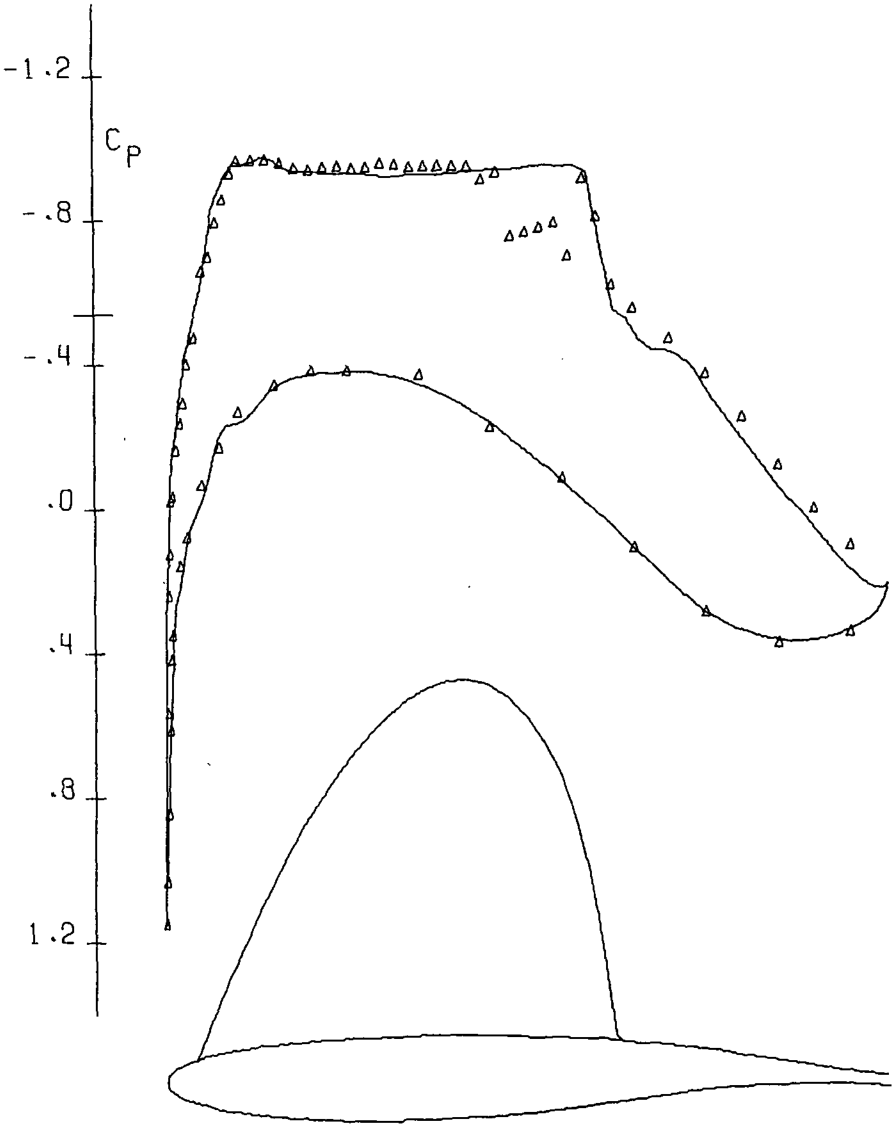
Δ EXPERIMENT M=.833 ALP= 1.51 CL= .551 CD=.0123



AIRFOIL 82-06-09 M*N=160*30 NCY= 400 R= 6 MILLION

— THEORY M=.840 ALP= 1.05 CL= .530 CD=.0184

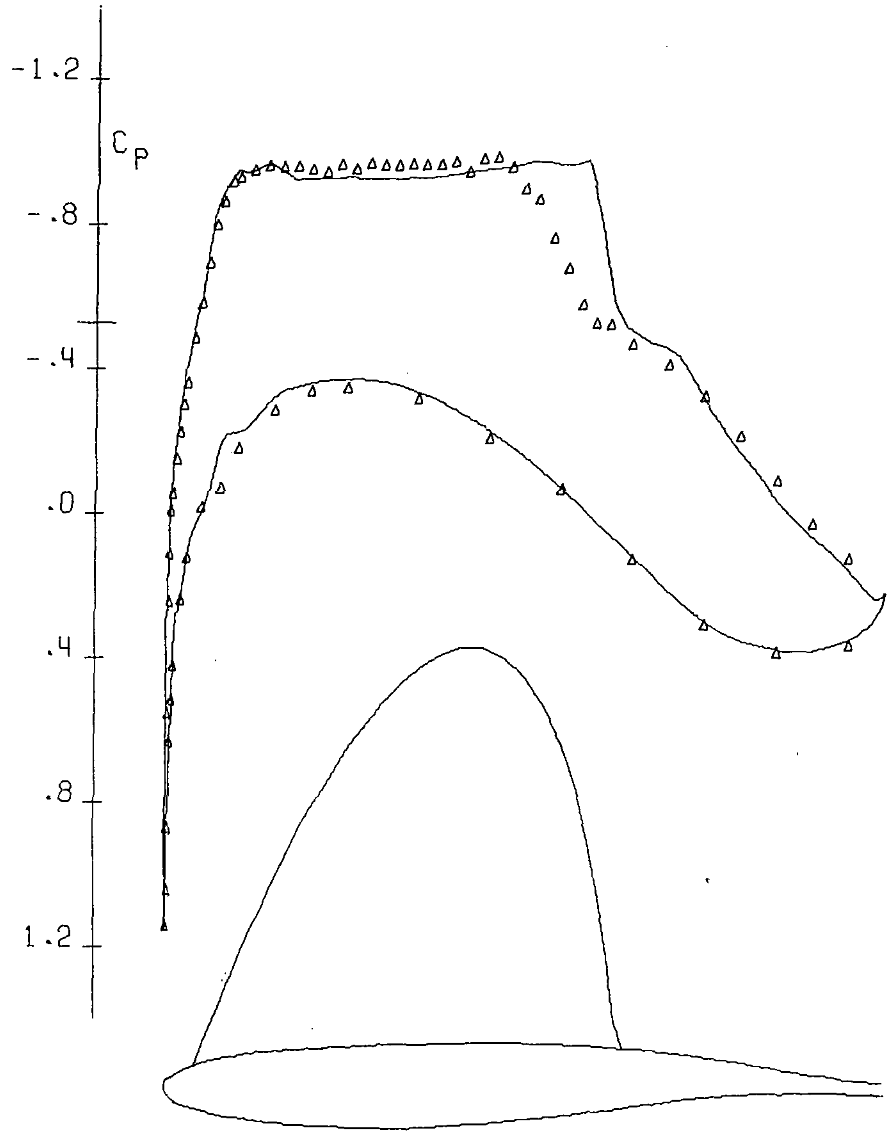
Δ EXPERIMENT M=.840 ALP= 1.50 CL= .530 CD=.0136



AIRFOIL 75-06-12 M*N=160*30 NCY= 400 R=20 MILLION

— THEORY M=.765 ALP= .72 CL= .576 CD=.0127

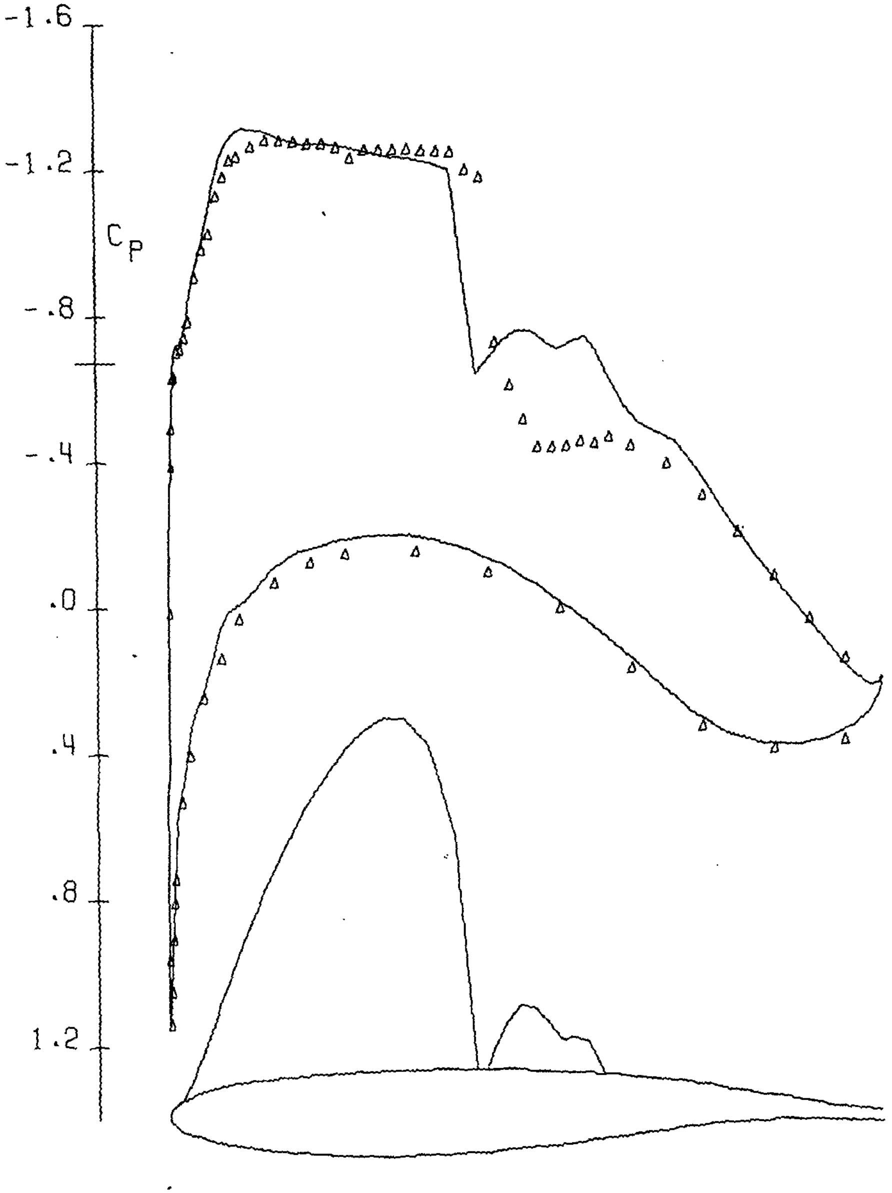
△ EXPERIMENT M=.765 ALP= .89 CL= .576 CD=.0110



AIRFOIL 75-06-12   M*N=160*30   NCY= 400   R=21 MILLION

— THEORY   M=.769   ALP= .81   CL= .588   CD=.0147

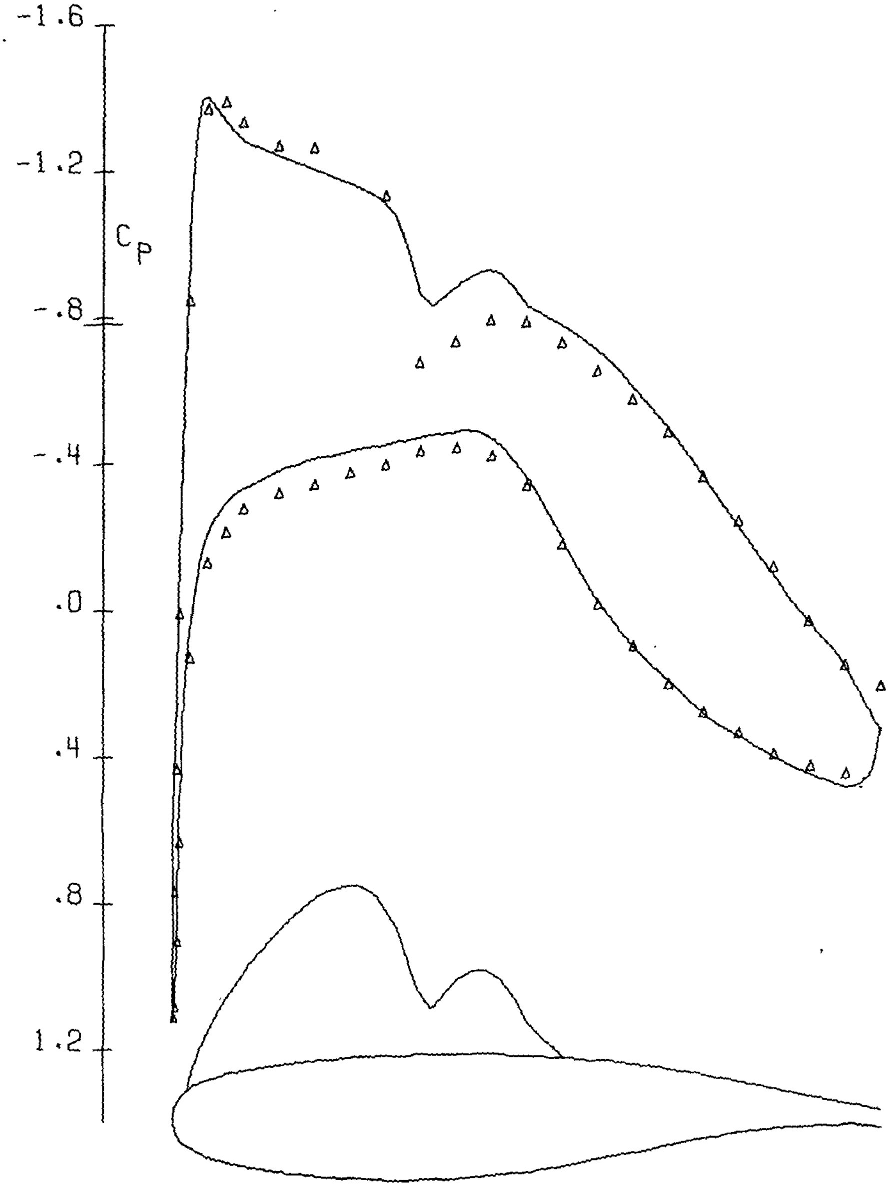
△ EXPERIMENT   M=.769   ALP= 1.65   CL= .588   CD=.0090



AIRFOIL 75-06-12 M*N=160*30 NCY= 400 R=20 MILLION

— THEORY M=.727 ALP= 2.16 CL= .787 CD=.0193

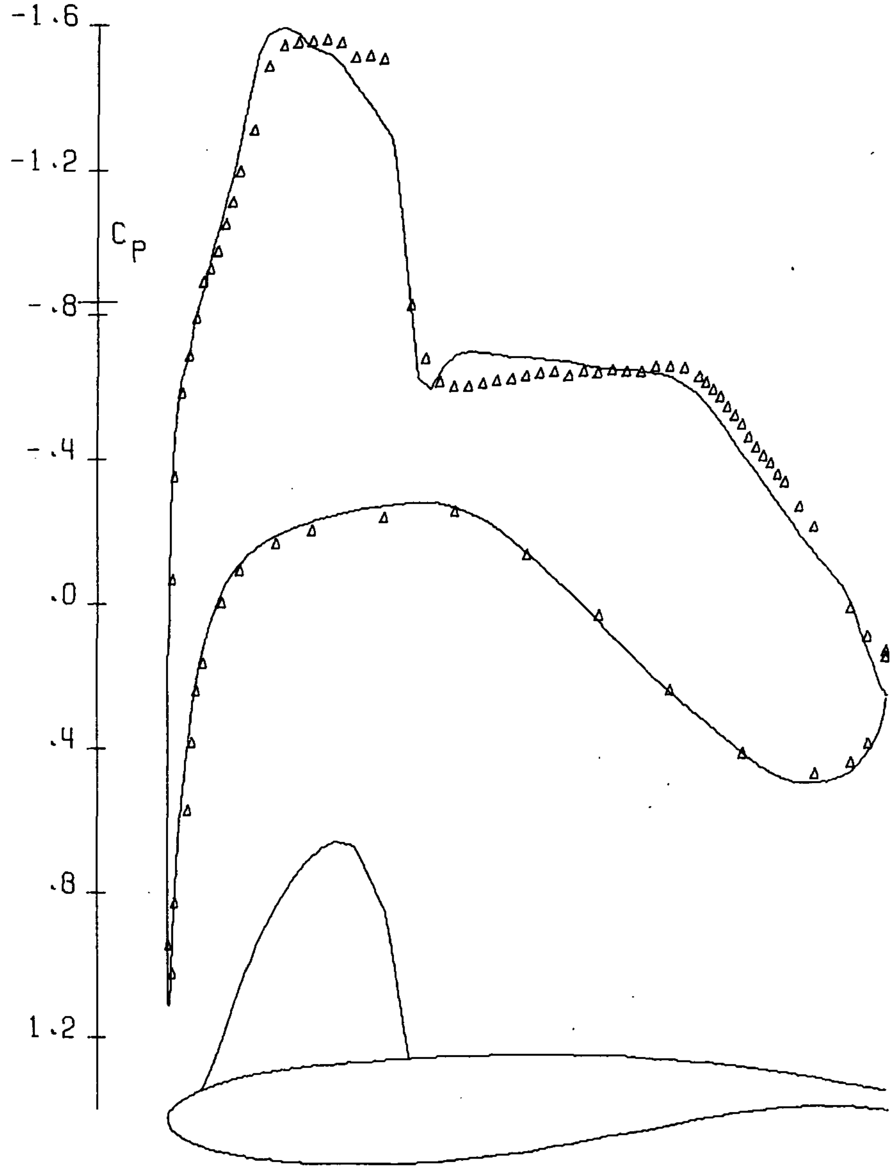
Δ EXPERIMENT M=.727 ALP= 3.49 CL= .787 CD=.0185



DOUGLAS AIRFOIL M*N=160*30 NCY= 400 R=14 MILLION
— THEORY M=.699 ALP= 1.22 CL= .615 CD=.0128
Δ EXPERIMENT M=.699 ALP= 1.38 CL= .615 CD=.0115

4. Comparison of Experimental Data with the Boundary Layer Correction Using the Quasiconservation Option

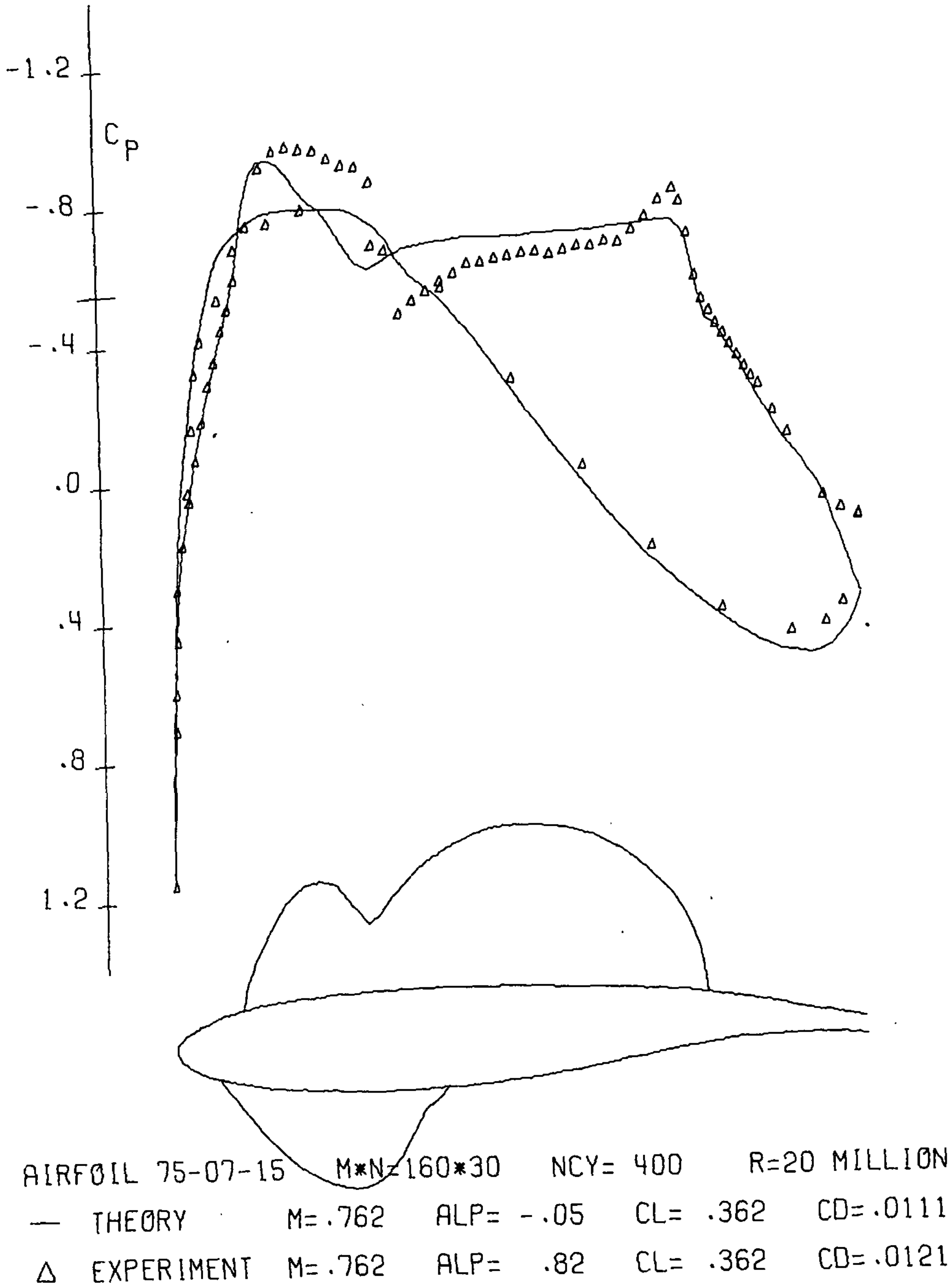
In this section we repeat some of the runs from Section 3 with the old Murman finite difference scheme replaced by the quasiconservation option (Chapter III, Section 7). The quasiconservation form gives better agreement with experiment when there is a strong shock wave well forward on the wing section where the boundary layer is relatively thin (see pages 132 and 152). It gives worse agreement in some cases with sizeable supersonic zones unless a Mach number correction is applied (see pages 138, 154 and 155). A full conservation form of the finite difference scheme, not listed, gives results virtually identical to those presented here (cf. Section 2).

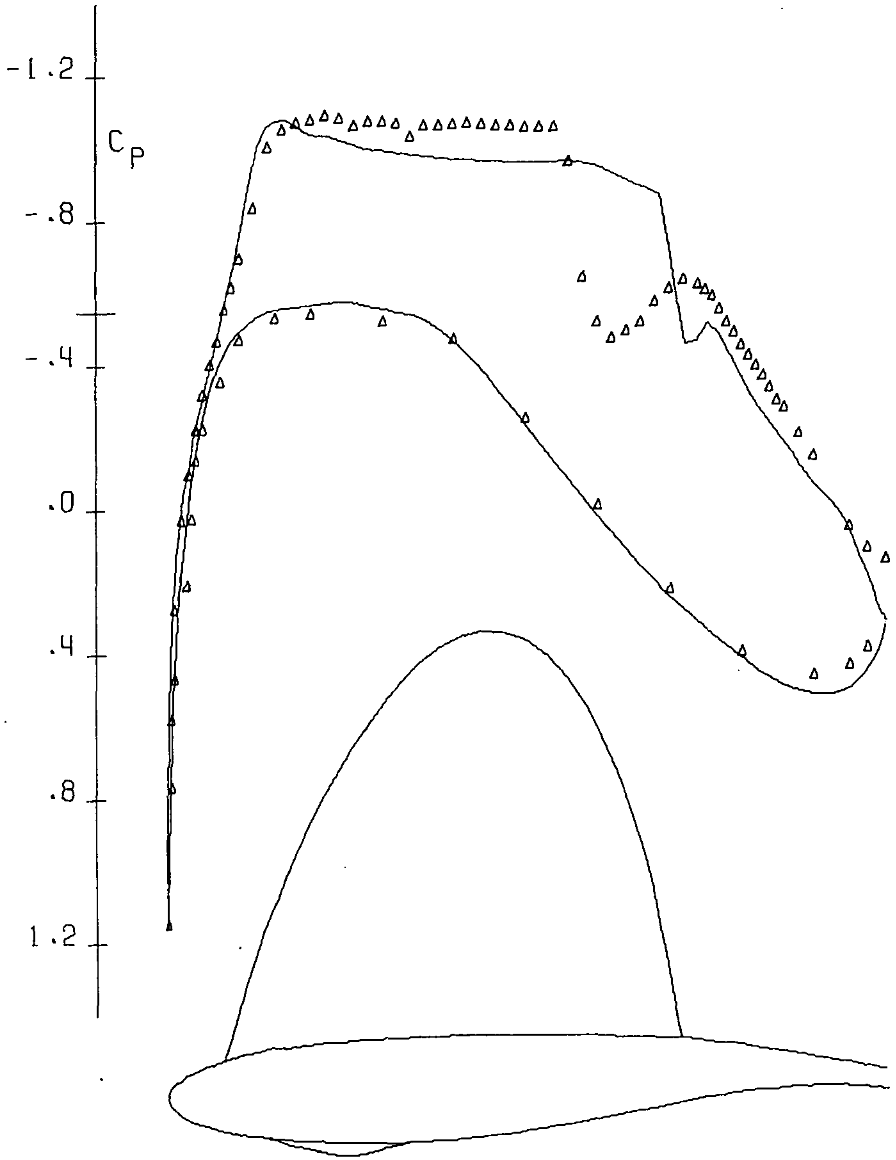


AIRFOIL 75-07-15 M*N=160*30 NCY= 400 R=20 MILLION

— THEORY M=.687 ALP= 2.80 CL= .809 CD=.0144

Δ EXPERIMENT M=.687 ALP= 4.09 CL= .809 CD=.0170

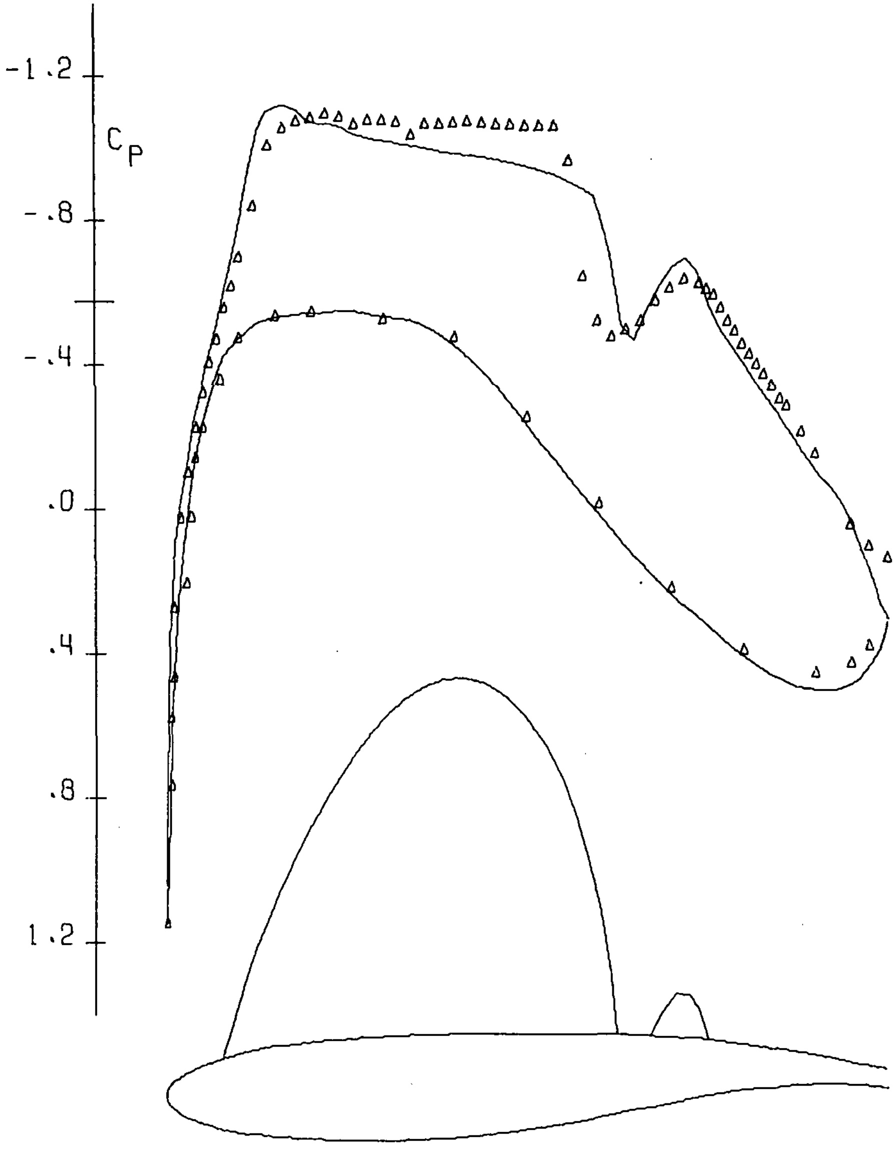

-1.2
C_P
-.8
-.4
.0
.4
.8
1.2
AIRFOIL 75-07-15 M*N=160*30 NCY= 400 R=20 MILLION
— THEORY M=.762 ALP= -.05 CL= .362 CD=.0111
Δ EXPERIMENT M=.762 ALP= .82 CL= .362 CD=.0121



AIRFOIL 75-07-15   M*N=160*30   NCY= 400   R=20 MILLION

— THEORY   M=.763   ALP= .86   CL= .584   CD=.0132

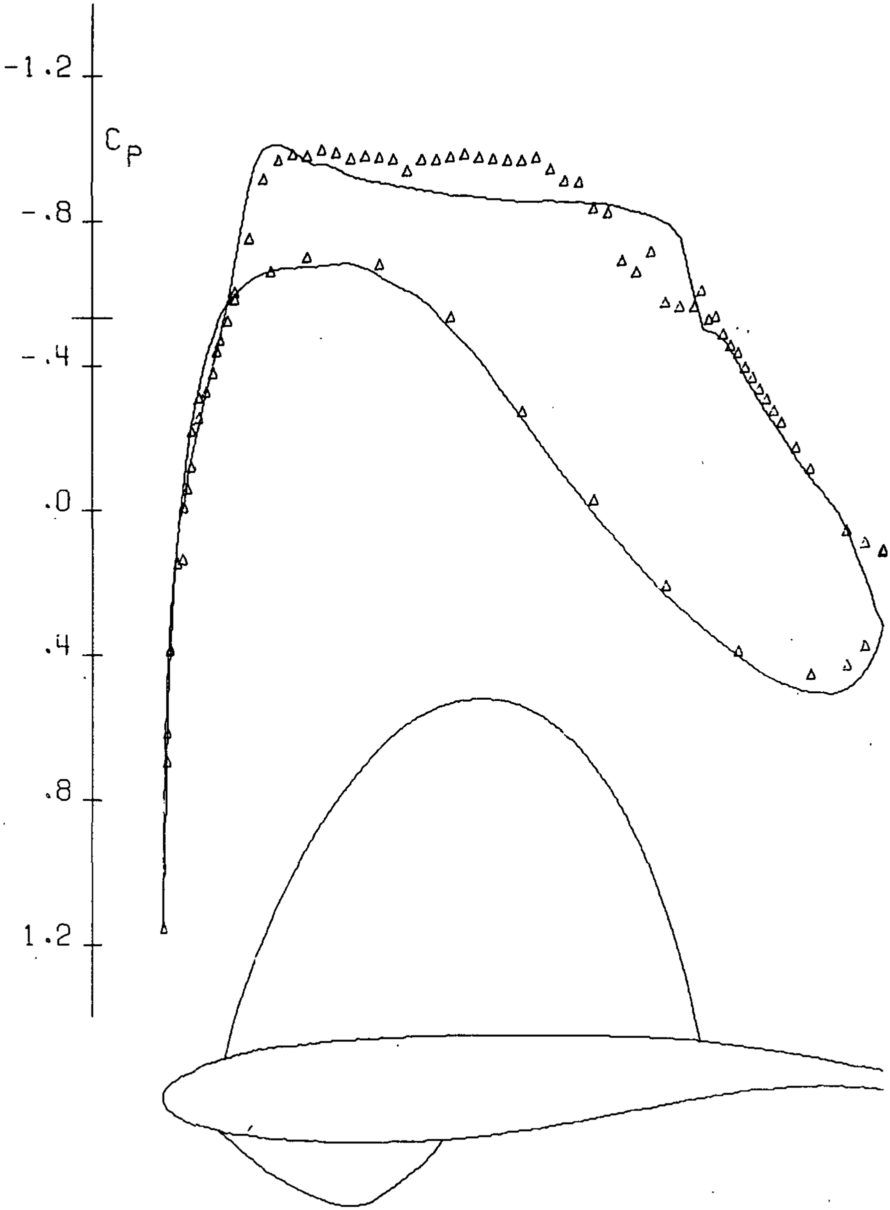
Δ EXPERIMENT   M=.763   ALP= 2.01   CL= .584   CD=.0128



AIRFOIL 75-07-15 M*N=160*30 NCY= 400 R=20 MILLION

— THEORY M=.755 ALP= .92 CL= .584 CD=.0123

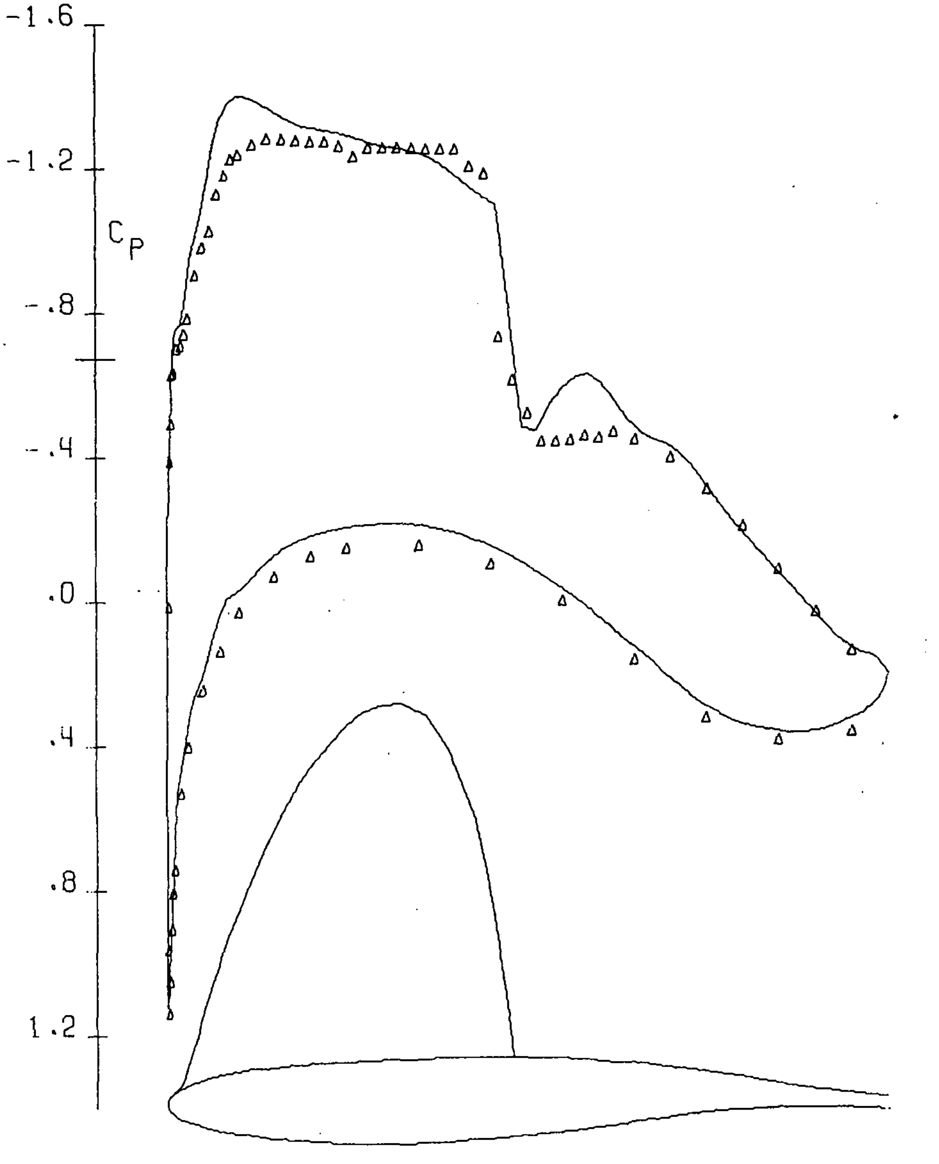
Δ EXPERIMENT M=.763 ALP= 2.01 CL= .584 CD=.0128



AIRFOIL 75-07-15 M*N=160*30 NCY= 400 R=20 MILLION

— THEORY M=.768 ALP= .45 CL= .494 CD=.0119

△ EXPERIMENT M=.778 ALP= 1.44 CL= .494 CD=.0115



AIRFOIL 75-06-12 M*N=160*30 NCY= 400 R=20 MILLION

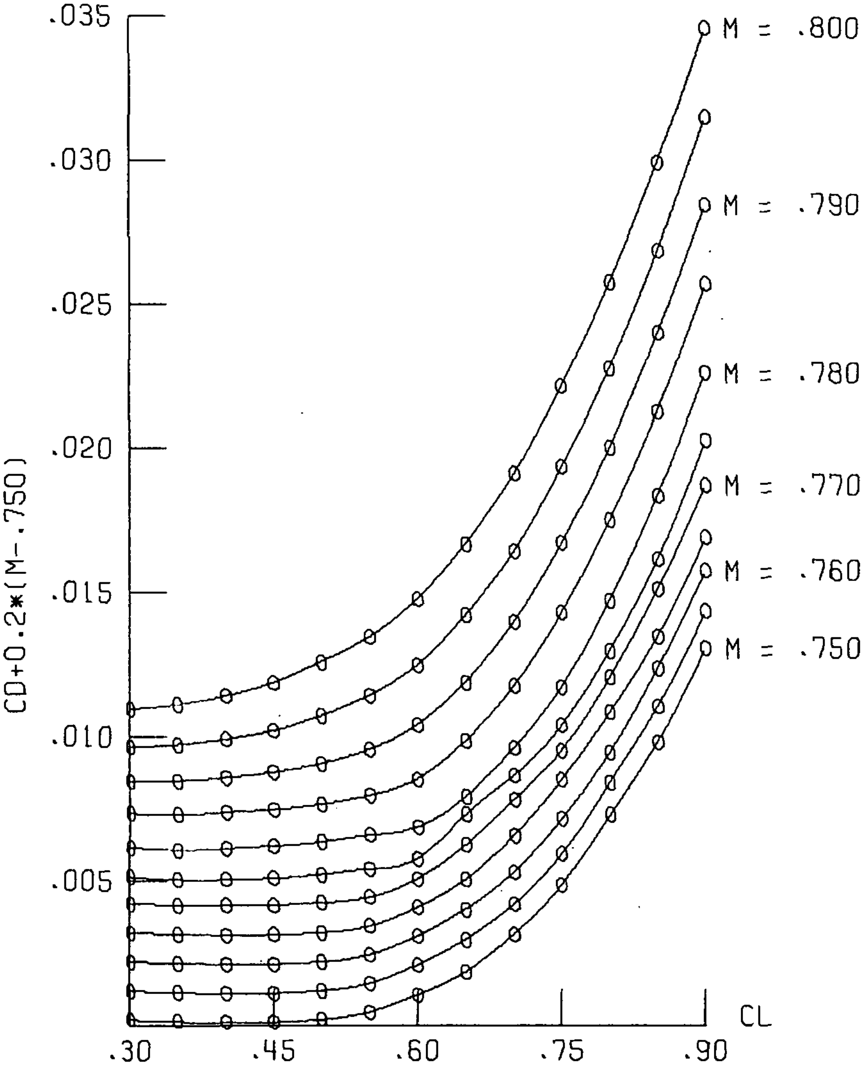
— THEORY M=.727 ALP= 1.97 CL= .787 CD=.0124

Δ EXPERIMENT M=.727 ALP= 3.49 CL= .787 CD=.0185

5. Drag Polars

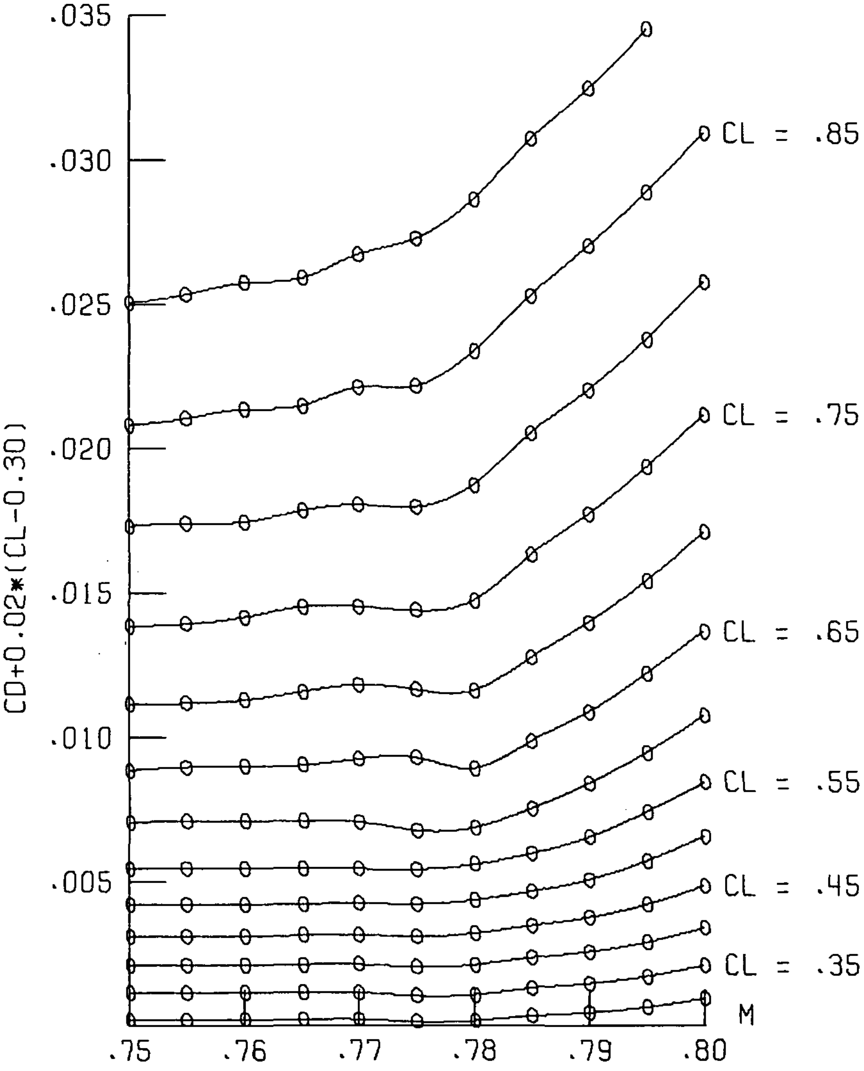
The first two figures presented in this section were obtained without any boundary layer correction, and they involve just the wave drag. The remaining two dimensional drag polars include comparisons with the experimental data described in Section 3. The agreement is good, except that drag rise may be predicted by the theory for Mach numbers that are as much as .02 less than those indicated by the experimental data. Discrepancies of this order of magnitude have been attributed to wall effect.

The final drag polars are for three dimensional flows past oblique wings. It appears that our computation of the induced drag plus the wave drag, which is a relatively small number, is accurate enough to yield physically significant values of the lift drag ratio L/D. Using a semi-empirical value of the profile drag, we have found that our evaluation of L/D compares favorably with the test data obtained by R. T. Jones [6] in the transonic wind tunnel at the NASA Ames Research Center. Moreover, the theoretical and experimental predictions of the effects of angle of attack and twist on the distribution of lift agree fairly well. Figure 9 compares experimental curves of maximum L/D against Mach number at fixed yaw angles with the envelope obtained from calculations in which the yaw angle was optimized at each Mach number. Figure 10 compares the predicted optimal lift drag ratios for two different supercritical wing sections. The calculations indicate that near Mach 1 the optimal lift drag ratio does not change much with the design Mach number of the section if the angle of yaw is adjusted properly.



WAVE DRAG ON AIRFOIL 78-06-10 AT
M = .750, .755, . . . , .800 AND R = 20 MILLION

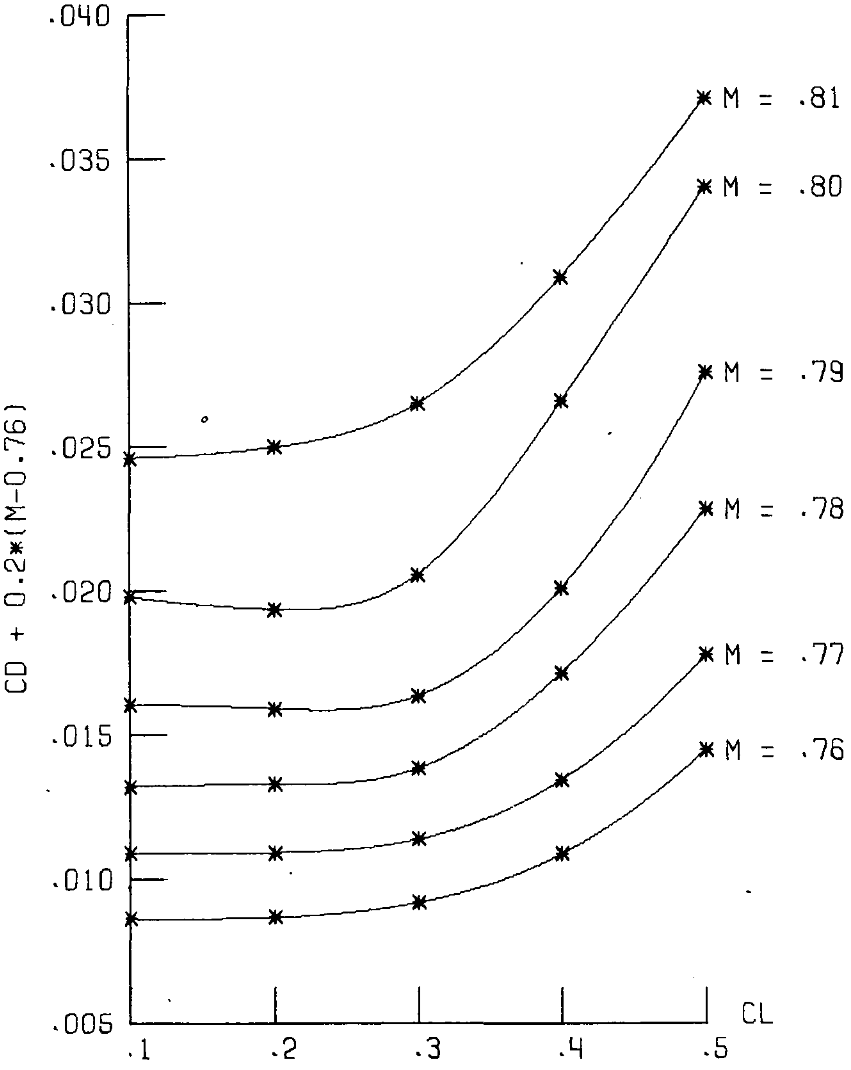
FIGURE 1



WAVE DRAG ON AIRFOIL 78-06-10 AT

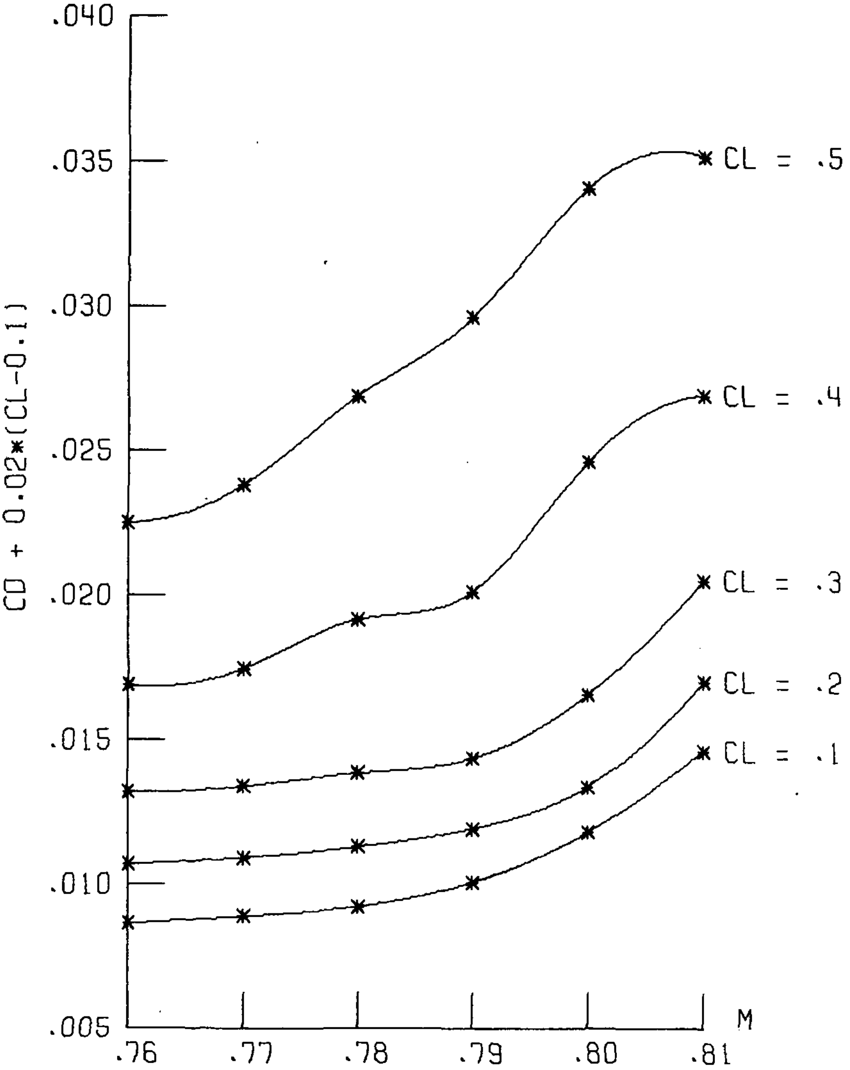
CL = .30, .35, . . . , .90 AND R = 20 MILLION

FIGURE 2



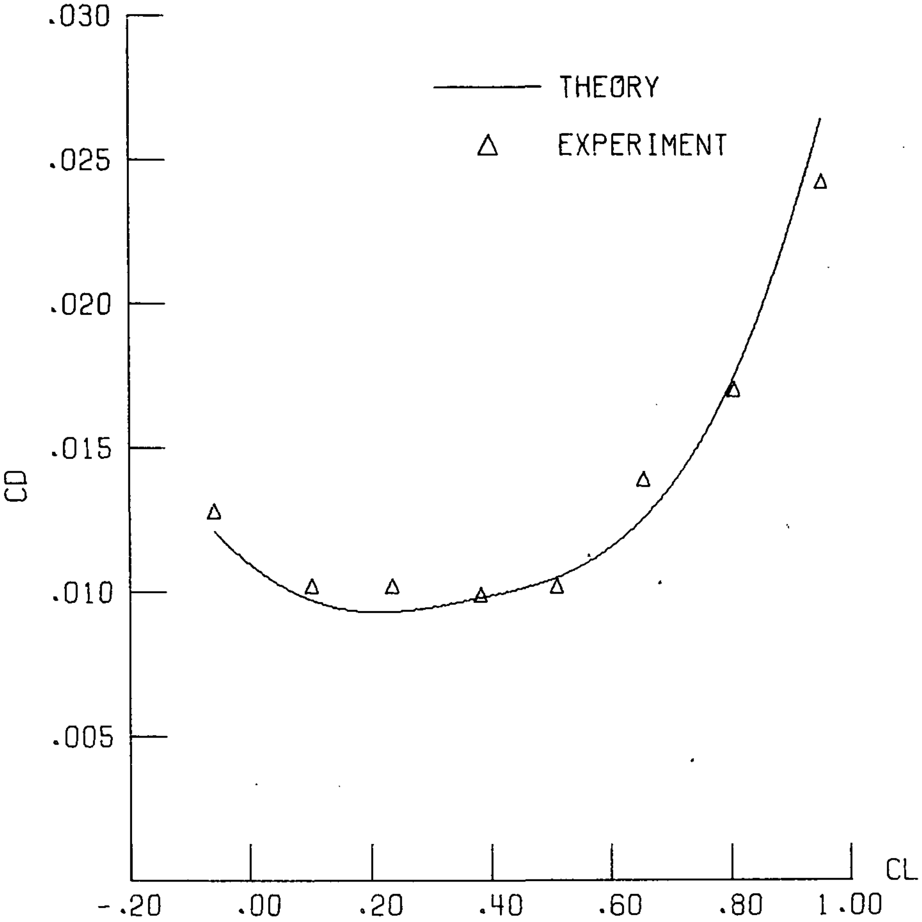
DRAG PØLAR FØR AIRFØIL 79-03-12 AT
M = .76, . . . , .81 AND R = 20 MILLIØN

FIGURE 3



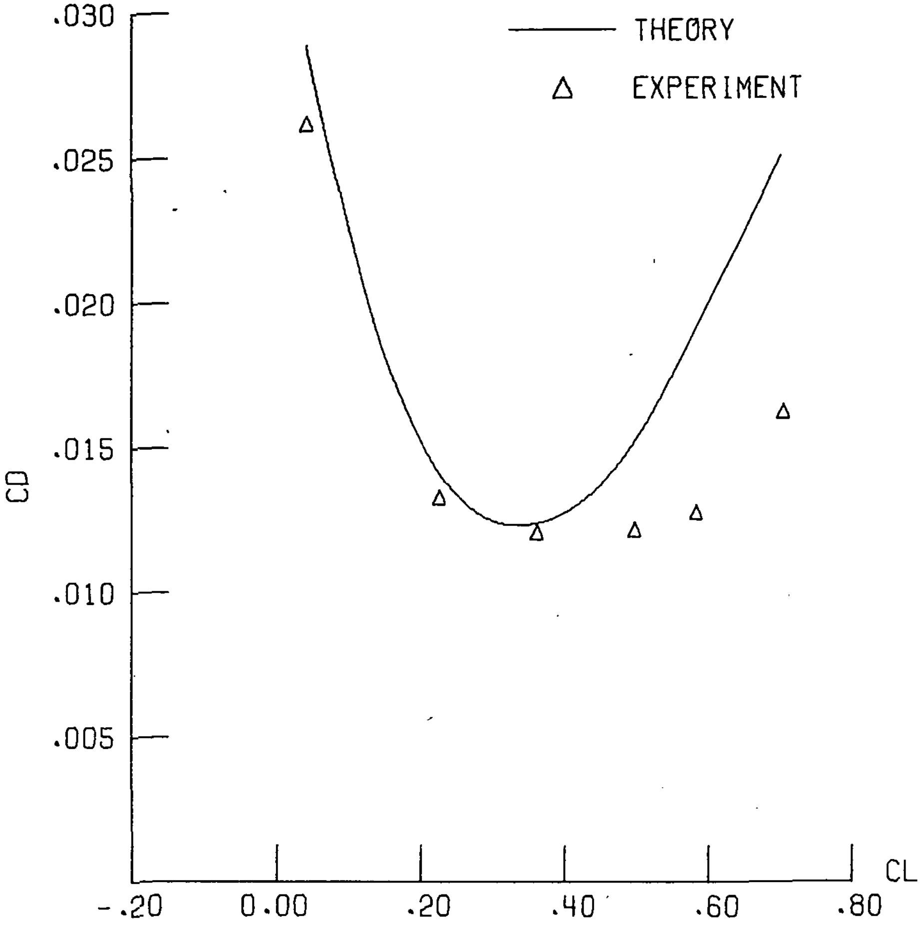
DRAG POLAR FOR AIRFOIL 79-03-12 AT
CL = .1, . . . , .5 AND R = 20 MILLION

FIGURE 4



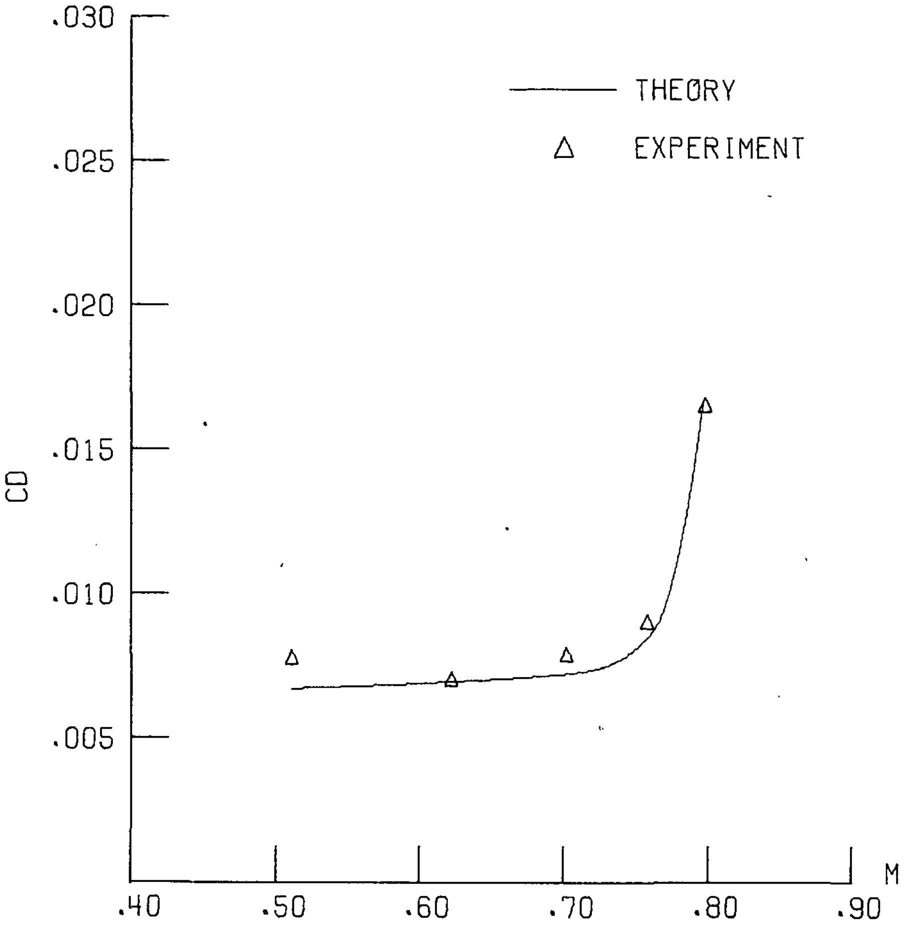
DRAG POLAR FOR AIRFOIL 75-07-15 COMPARING OTTAWA EXPERIMENTAL DATA WITH THEORETICAL ANALYSIS AT M = 0.69 AND R = 20 MILLION

FIGURE 5



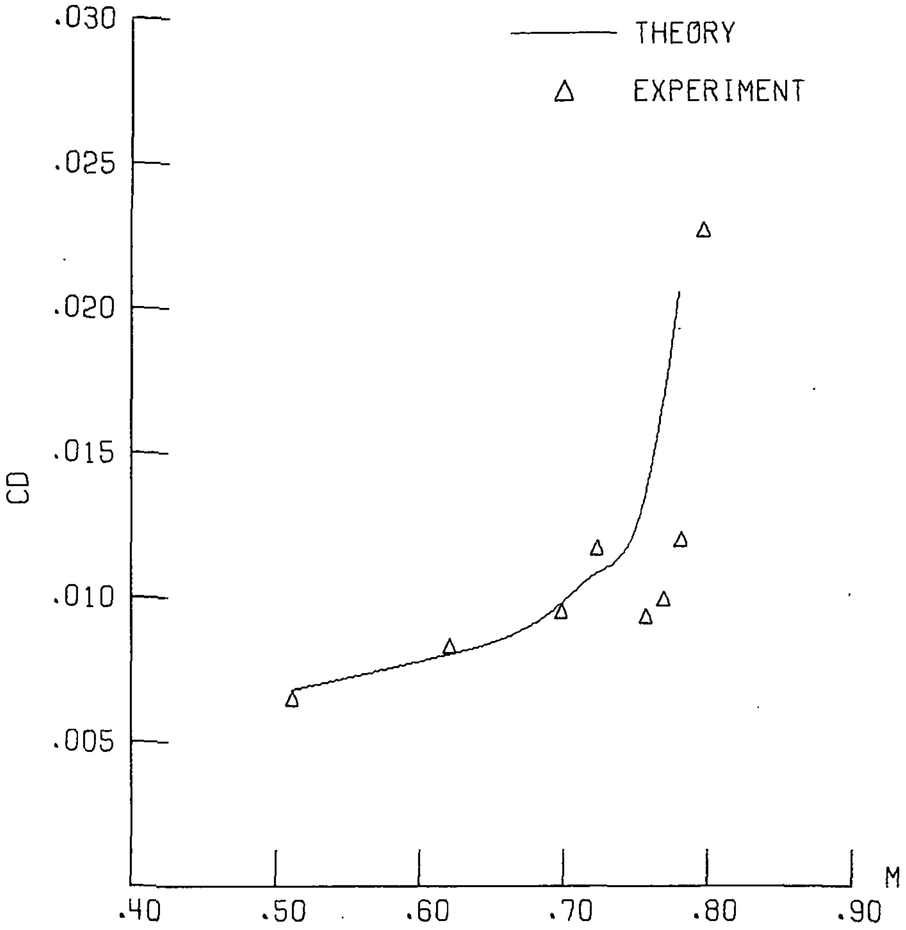
DRAG POLAR FOR AIRFOIL 75-07-15 COMPARING OTTAWA EXPERIMENTAL DATA WITH THEORETICAL ANALYSIS AT M $\doteq$ 0.76 AND R = 20 MILLION

FIGURE 6



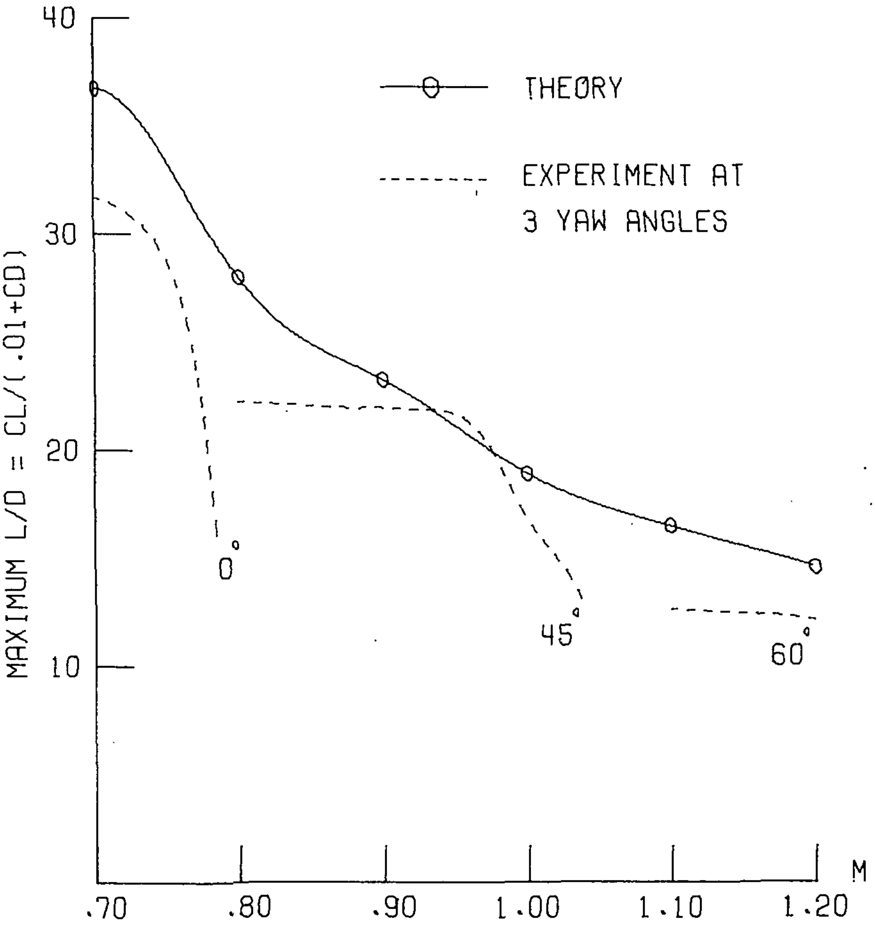
DRAG POLAR FOR AIRFOIL 75-06-12 COMPARING OTTAWA EXPERIMENTAL DATA WITH THEORETICAL ANALYSIS FOR CL = 0.4 AND R = 20 MILLION

FIGURE 7



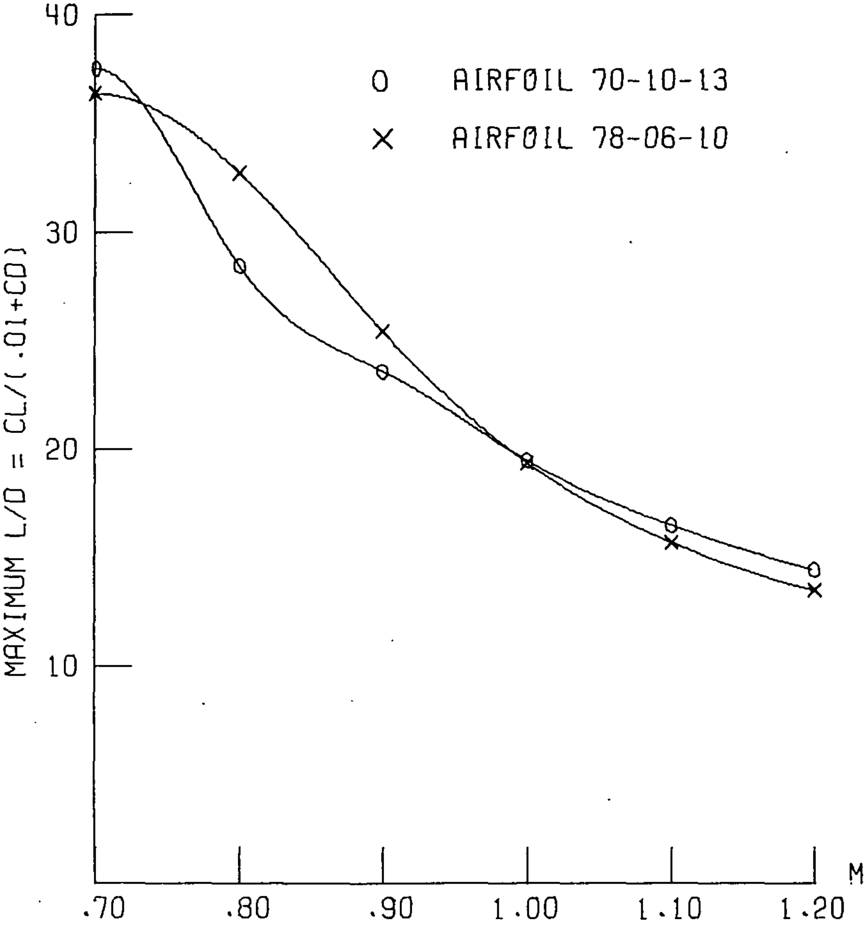
DRAG POLAR FOR AIRFOIL 75-06-12 COMPARING OTTAWA EXPERIMENTAL DATA WITH THEORETICAL ANALYSIS FOR CL = 0.6 AND R = 20 MILLION

FIGURE 8



DRAG PENALTY ON AN OBLIQUE WING WITH AR=12.7
COMPARISON BETWEEN THEORY AND EXPERIMENT FOR
JONES AIRFOIL TESTED AT AMES RESEARCH CENTER

FIGURE 9



DRAG PENALTY ON AN OBLIQUE WING WITH AR=12.7
THEORETICAL COMPARISON OF TWO DIFFERENT AIRFOILS

FIGURE 10

6. Schlieren Photographs

Figures 1, 2 and 3 are from a test at the NASA Langley Research Center of Airfoil 79-07-10 appearing in Volume I. Figure 1 shows the flow at a fairly high Mach number below the design lift, with two shocks on the upper surface and a shock on the lower surface. Figure 2 shows the flow slightly below the design point with two shocks quite far back, and Figure 3 shows the nearest approach to the design flow for this airfoil with one fairly weak shock. Figures 4 and 5 are from the Aircraft Research Associates test of Airfoil 82-06-09; they are British Crown Copyright, and are reproduced by permission of the Controller, R & D Establishments and Research, Ministry of Defence (PE). Figure 4 shows the flow somewhat below the design point, as in Figure 2, with two shocks. Figure 5 shows the flow above the design point with a single shock far back on the airfoil. Figure 6 is from a test performed at the Grumman Aerospace Corporation of an airfoil designed by Don MacKenzie and Bill Evans, using Airfoil 70-07-20 as a starting point. The design pressure gradient was too severe near the tail on the upper surface and the flow was strongly separated in the test.

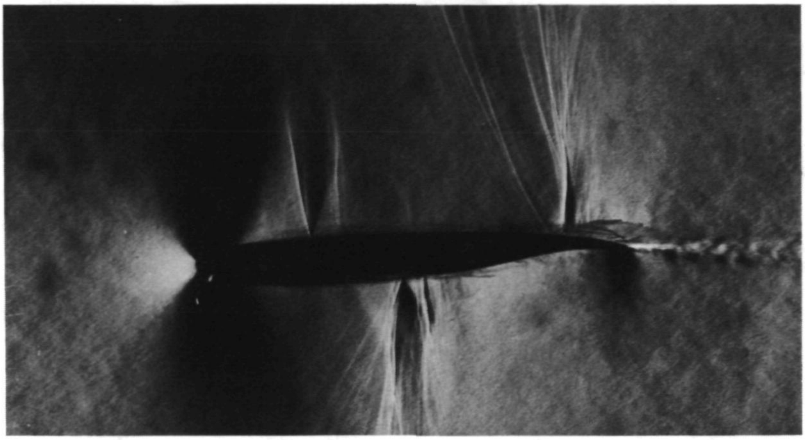


Fig. 1 AIRFOIL 79-07-10 AT M=.83 AND α=0°

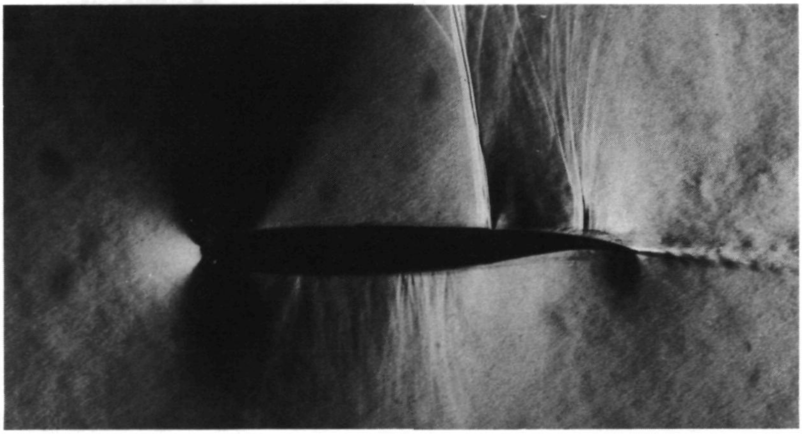


Fig. 2 AIRFOIL 79-07-10 AT M=.82 AND α =2°

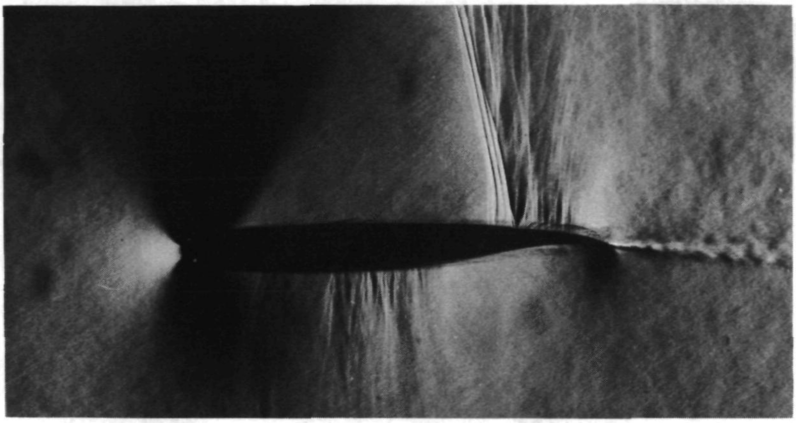


Fig. 3 AIRFOIL 79-07-10 AT M=.84 AND α=2°

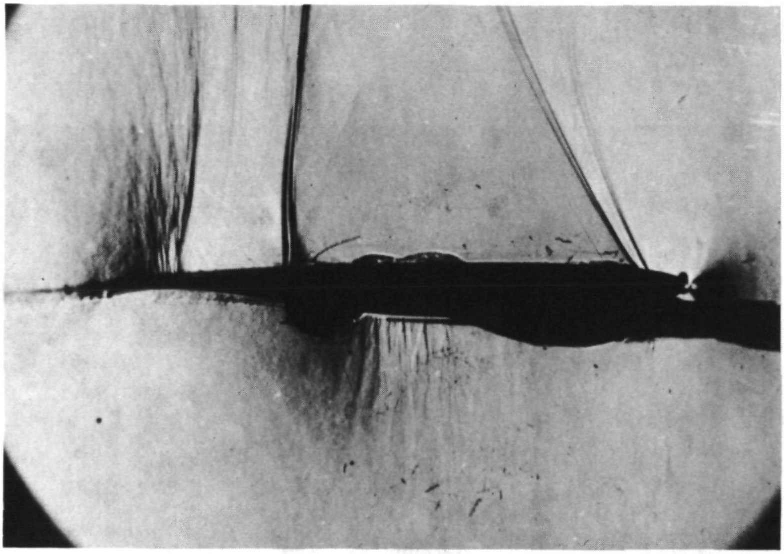


Fig. 4 AIRFOIL 82-06-09 AT M=.83 AND α=1.5°

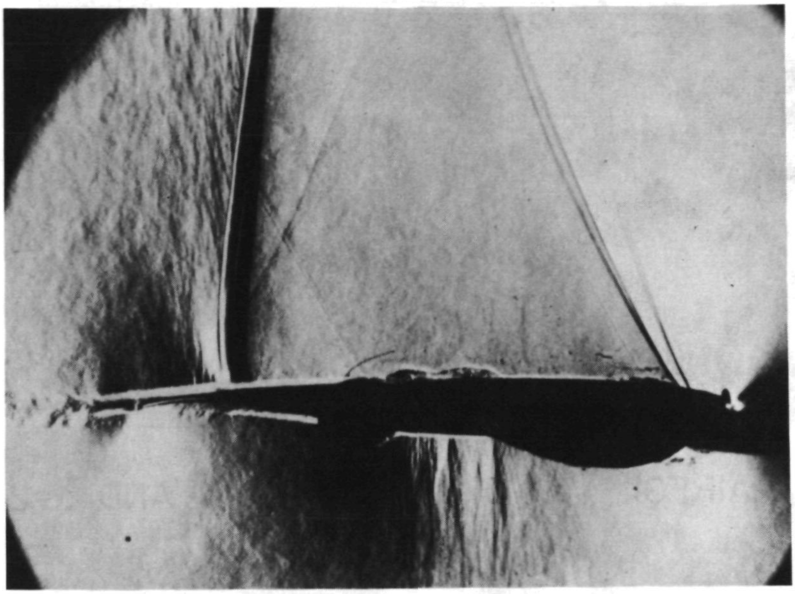


Fig. 5 AIRFOIL 82-06-09 AT M=.84 AND α=1.75°

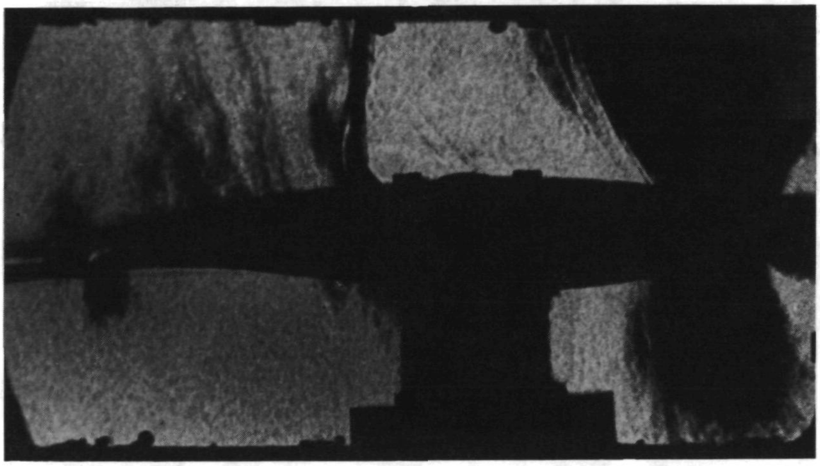


Fig. 6 GRUMMAN AIRFOIL AT M=.72 AND α=3°

III. FORTRAN PROGRAMS

1. Operation of the Turbulent Boundary Layer Correction Program H

The program which was written for the analysis of the flow past an airfoil (Program G) and described in Volume I has been rewritten and expanded to incorporate a turbulent boundary layer correction. Program F, which was discussed in Volume I and removes the turbulent boundary layer, has been superseded by our new program. The new Program H can now be used:

1. As F, in order to compute the turbulent boundary layer and remove it from an airfoil produced by our design programs explained in Volume I.

2. As G, to compute the flow around an airfoil without a boundary layer correction.

3. To add iteratively the boundary layer displacement to an airfoil in order to evaluate its performance including the effects of viscosity.

4. To obtain a redistribution of airfoil coordinates.

Program H has been written to operate in much the same way as Program G of Volume I. In fact, anyone familiar with Program G should have little difficulty in using Program H. Again Tape 3 is used for coordinate input and the format for it is prescribed by the FSYM value. The deck structure and data structure corresponding to the FSYM values appear in Table 1 of Section 2. In order to facilitate the comparison of theoretical results with experimental data or any other data we have provided the user with the option of plotting the comparison data. If Tape 4 contains test data and XP = 1, the comparison data, if it is test data, will be designated by triangles (Δ) on the Calcomp plots of the pressure distribution. If the comparison data is design data, it is designated by plus signs (+). The format of Tape 4 is shown in Table 2. The parameter SNX selects the plot symbol for the comparison data.

All input parameters which can be varied appear in the input glossary and they can be redefined by data cards prepared with standard namelist conventions. Many of the parameters are the same as those used previously in Program G. Additional parameters which will generally not need to be changed from case to case have been initialized by means of data statements in the various subroutines. They can be changed by updating the program.

The output consists of a printed copy of the numerical results, Calcomp plots of the results and a printed Mach number chart.

If the program is to run as old F, no namelist information is necessary; default values will be used for this entire program. The default values set transition at the maximum value of the pressure distribution CP and set the Reynolds number RN equal to 20.E06. If these default values are not those required for the case under consideration then it is sufficient to provide a data card with the correct values. During execution of the so-called F mode, the coordinates will be redistributed, angles for the airfoil computed, a mapping onto the circle performed and the boundary layer removed. The printout from this program is similar to the printout from Program F. Upon termination Tape 3 will contain the output data in the FSYM = 2. format and it can be used as input to the program when it is used to perform a flow calculation with the boundary layer displacement. The termination of the program produces an output listing of the original airfoil coordinates x, y, the corrected coordinates YS, the surface slope and curvature, the pressure distribution CP, the momentum thickness THETA and the separation parameter SEP.

As explained in Section 5 of Chapter I, when this code is used to add on the boundary layer correction, it is run as G for NS1 (see the Glossary) cycles and then the displacement thickness is computed by means of the Nash-Macdonald equations (von Karman equation) and added to the original airfoil. The resulting airfoil is then mapped onto the circle and the flow is computed for another NS1 cycles. This process is repeated every NS1 cycles until the total number of flow cycles is computed. The program runs until NCY = NS, where NCY is the running tally of flow cycles, or until the convergence tolerance ST is achieved. If NS1 $\geq$ NS or if RN =0, no boundary layer correction will be made and the program runs as old G.

If FSYM = 2. or FSYM = 4., surface slopes are not provided. The slopes are needed to perform the mapping and are obtained by passing splines through the x and y coordinates. If the parameter IS is non-negative the x and y coordinates are redistributed before computing the slopes. If IS > 0 these new x,y coordinates are smoothed IS times. The smoothing formula is weighted so that the most smoothing is applied at the tail. IS = 2 is the default. After the surface slopes are obtained the mapping to the circle is performed as discussed in Chapter I, Section 4, at NMP = 2*NFC intervals, where NFC is the number of terms in the Fourier series. The mapped airfoil coordinates are obtained at M+1 points; M = 160 for the coordinates after the first mapping. Of these M+1 mapped coordinates, 108 points (NT) are saved. The points are obtained by thinning out every other point of the upper and lower surfaces near the trailing edge. The flow calculation is generally started on a cruder grid, M×N = 80×15. After the flow, the ordinary differential equation given by Nash and Macdonald is integrated at NPTS points in the circle plane. NPTS = 81 is the default value. The input to this equation is the set of local Mach numbers at the NPTS points of the circle plane which are obtained from the flow calculation. The differential equation is solved for the momentum thickness THETA, from which the displacement thickness DELS is computed at the NPTS points. In solving the differential equation the quantity SEP is bounded by a limit SEPM in order to permit integration through a shock. For all x values greater than a prescribed value XSEP, SEP is set equal to its computed values even if SEP is greater than SEPM. XSEP = .93 and SEPM = .004 are the default values. An interpolation is performed to obtain the displacement thickness at the NT points of the original airfoil. The correspondence used for interpolation is through arc length. The original airfoil is modified by adding the displacement thickness to it along a vertical projection. In order to improve the convergence process the computed displacement thickness is subjected to monotonicity conditions on the upper and lower surfaces. It is also smoothed IS times and underrelaxed using a relaxation parameter RDEL. For NPTS = 81, we have chosen IS = 2 and RDEL = .125. If the number of points used to solve the differential equation is doubled it is reasonable to increase IS by a factor of 4. The new airfoil defined by the NT coordinates x,y is mapped onto the circle and then the flow calculations are resumed.

An additional feature of this program is the option of modifying the Mach number distribution after separation for input to the Nash-Macdonald equation. The pressure distribution on the upper surface is altered by a linear extrapolation of the pressure from some point along the upper surface to a base pressure BCP at the tail. LSEP is the index of the x array at which this extrapolation begins and should be placed after the point at which separation occurs. LSEP is obtained empirically. LSEP+1 is the index of the first point at which the pressure distribution is modified. The initial BCP value is an estimate. This value is iterated making use of the assumption that the pressure distribution is monotonic after separation. The search for monotonicity is made from x = XMON to the trailing edge. The straight line for the pressure distribution after separation is determined each time by the value at the point LSEP and the iterated BCP.

We have found that good results are generally obtained without making use of this option and have, therefore, set the LSEP default value to M+1. This means that the pressure distribution derived from the flow will not be modified on the upper surface. In difficult cases it may help to alter the pressure distribution in this manner.

There is no printout during each boundary layer calculation, but the boundary layer computed at the NPTS points on the upper surface and lower surface is plotted. In the plot the upper surface starts at the center and x = 1. is at the bottom of the graph paper. The lower surface starts at the center and x = 1. is at the top of the page. The quantities printed out after each KP cycles of flow are described in the output glossary. The first eight variables change during the flow calculation. The last five quantities are the result of computations in the boundary layer correction subroutines and remain constant for the NS1 flow cycles. The flow program has been modified so that the flow can be computed with a fixed CL or a fixed ALP; the parameter which varies is printed out after each KP cycles.

The computation proceeds for NS cycles or until both the maximum velocity potential correction and the maximum circulation correction are less than the tolerance ST. The ITYP parameter is used as in Volume I to select the type of output. If ITYP $\geq$ 3, the coefficients of lift, wave drag, form drag, total drag and pitching moment are printed. The Mach number diagram is also printed. For ITYP ≥ 3 the printout consists of the displaced airfoil geometry and the flow and boundary layer characteristics at each of the M+1 points. The description of each column of output appears in the Output Glossary. If ITYP ≥ 4 a Calcomp plot of the pressure distribution from computation, comparison test data if available, the airfoil and the sonic line are plotted. A plot of the final displacement thickness for the upper and lower surfaces appears on the last page. If ITYP < 4 no Calcomp plots are made. If ITYP = 5 the sonic line is not plotted. If ITYP = 1 there are no Calcomp plots, Mach number diagram or final printout.

An example of a set of control cards for the CDC 6600 Scope 3.2 operating system and data cards for the boundary layer program is given below. In order to use Program H control cards are needed which retrieve the program from the program library, store the airfoil geometry on Tape 3 and the test data, if available, on Tape 4.

For the CIMS CDC 6600 the control cards are:

ATTACH(T,AIRFOIL)	AIRFOIL contains airfoil geometry
REWIND(T,TAPE3)	
COPYBF(T,TAPE3)	Tape 3 is input to Program H
ATTACH(TAPE4,TESTDATA)	Tape 4 is input for H and TESTDATA is a file which contains experimental data
ATTACH(H,PROGRAM)	File PROGRAM contains compiled Program H
H.	Execution of Program H

See Table 1 for the format of data for Tape 3. See Table 2 for the format of data for Tape 4.

We assume that Tape 3 contains the model coordinates for one of our airfoils designed at Mach number EM = 0.75, CL = 0.667 and T/C = 0.151 (Airfoil 75-07-15) which was tested at the National Aeronautical Establishment, Ottawa, Canada. Since only x and y were measured, Tape 3 contains the coordinates in FSYM = 4. mode. Tape 4 is assumed to contain the experimental results of a test made at Mach number equal to 0.762, angle of attack equal to 0.82 degrees with coefficient of lift equal to 0.362, and Reynolds number equal to 20.0×10^6.

The first data card which may be used as input to the program is:

```
[ $P NS=1, FSYM=4., EM=0.762, CL=0.362, IS=2, PCH=0.07,
  XSEP=0.93, RN=20.0E06, BCP=.4, SEPM=.004, XP=1., RBCP=0.1,
  RDEL=0.125,NFC=80$ ]
```

However, for this case the input data can be simplified considerably, since most of the parameters are equal to their default values set by the program. Thus, the first input card we use is:

```
[ $P FSYM=4., EM=0.762, CL=0.362, XP=1.$ ]
```

After reading this card the program begins the mapping. The airfoil coordinates are redistributed and smoothed 2 (IS) times. The airfoil is mapped using 2*NFC intervals on the circle. The x,y coordinates are obtained at 161 (M+1) points equally spaced in the circle. From these coordinates 108 (NT) are saved: every other point in the first third of the original 161, every point in the second third (around the nose) and every other point in the last third, which includes the points near the upper surface trailing edge. This was done to maintain the resolution at the nose and to improve the convergence, since points at the trailing edge spaced too closely can lead to computational difficulties. The 108 points define the inner airfoil to which we add the boundary layer correction and obtain a new outer airfoil at each boundary layer correction cycle.

The second data card supplied to the program is:

```
[ $P NS = - 1, ITYP = 1$ ]
```

The machine response to this card is a change in mesh size to a cruder mesh. The mesh size is then $M \times N = 80 \times 15$ for the flow calculation. The original airfoil remains unchanged, defined at 108 points.

The third data card is:

[$P NS=400, NS1=20, LSEP=75, XMON=.95, IS=2, ITYP=4, KP=4$]

This card initiates the computation of the flow around the airfoil. 400 (NS) cycles of the flow on the crude grid size will be computed before termination. After each 20 (NS1) cycles of the flow a boundary layer correction will be made. The pressure distribution and Mach numbers resulting from flow cycle NS1 are computed. Since LSEP is not equal to its default (81 for the crude mesh, 161 for the fine mesh) the pressure distribution is modified on the upper surface from LSEP+1 to the trailing edge. Since the pressure BCP at the trailing edge was not read in on a data card, the default value BCP = 0.4 is used initially. At all boundary layer cycles after the first, BCP is iterated and underrelaxed using the monotonicity condition on the pressure distribution beyond separation. The search for monotonic behavior starts at $x = 0.95$ (XMON). The Nash-Macdonald equation is integrated at 81 points (NPTS) equally spaced in the circle plane. After the displacement thickess δ has been computed from the equation it is subjected to the requirement that on the upper surface it be monotonically increasing and that on the lower surface for $x \leq 0.6$ it be monotonically increasing and for larger x once it starts to decrease it should not increase again. Then δ is smoothed, and the number of smoothings is given by IS. This is the same parameter name used for the smoothings of the original airfoil, but that smoothing is not done for each new outer airfoil. The amount of δ to be added to the original airfoil is underrelaxed to achieve convergence. RDEL is the relaxation factor. After a spline fit at the NPTS points at which the equation was solved and an interpolation at the NT points of the inner airfoil, δ is added to the inner airfoil. The resulting airfoil defined at 108 (NT) points is then mapped onto the circle. The mapping is done at 161 (2 NFC + 1) points. The new mapped coordinates are obtained at 81(M+1) points. The program then returns to the flow cycles. When NS = 400 the crude mesh calculation is complete. ITYP = 4 gives the Mach number chart, a printout of the relevant variables, the Calcomp plot of the pressure distribution, airfoil and sonic line, and the plot of the last upper and lower δ. Since some default values are used, Card 3 can be shortened to:

```
[ $P NS=400, LSEP=75, ITYP=4, KP=4$ ]
```

The fourth data card is:

```
[ $P NS=1, ITYP= -1$ ]
```

The mesh is restored to the finer grid, M×N = 160×30. The inner airfoil is still defined at 108 points. However, the mapping is redefined on the fine grid and the new airfoil is obtained at 161 points on the circle. The value of LSEP is adjusted to the corresponding index for the fine mesh. All other required variables are interpolated.

The fifth data card is:

```
[ $P NS=400, ITYP=1$ ]
```

400 cycles of flow are done on the fine mesh with a boundary layer correction computed every NS1 cycles. The Nash-Macdonald equation is still integrated using 81 (NPTS) points.

The sixth data card is:

```
[ $P ITYP=0$ ]
```

The program terminates with printout of final results and Calcomp plots. On the CDC 6600 any namelist error is treated as an exit card like the sixth data card.

The time required to compute 400 cycles of the flow and obtain a new boundary layer correction each 20 cycles is approximately 110 seconds on the crude grid. Approximately 2.3 seconds are required to obtain each new outer airfoil and map it onto the circle. In total about 65 seconds are spent on the flow and 45 seconds on the outer airfoil. 307 seconds are required for the 400 cycles calculation at a fine mesh size. 231 seconds are needed for the flow and 76 seconds for mapping the 19 outer airfoils.

2. Glossaries and Tables for Program H

Glossary of Input Parameters

ALP	Real. Angle of attack in degrees relative to angle 0° at design. Default 0°. See CL.
BCP	Real. Starting value of the base pressure which is used when the pressure distribution is extrapolated linearly on the upper surface. Default 0.4.
BETA	Real. Damping coefficient for rotated difference scheme used to solve the flow equations. Default 0. BETA > 0 may help the convergence for Mach numbers near 1.0.
CL	Real. Coefficient of lift. The default is based on the ALP default since the program permits either ALP or CL to be prescribed. CL defaults to the design value for FSYM < 3.
EM	Real. The free stream Mach number. It must be less than 1. Default 0.75 or design Mach number if FSYM < 3.
FSYM	Real. Indicates format of original airfoil coordinates on Tape 3. See Table 1. Default 1.
GAMMA	Real. Gas constant γ. Default 1.4.
IS	Integer. Number of smoothings of original airfoil coordinates. Also the number of smoothings of the displacement thickness. Default 2.
ITYP	Integer. Used along with NS to indicate mode of operation. ITYP = 0 causes program to terminate. See Table 3. Default 1.
IZ	Integer. Width of output line control. Controls the number of characters on a line of output as well as the file to which output is written. In addition, if IZ = 120 the Fourier coefficients of the mapping are printed. Default 125.
KP	Integer. Print parameter. The output from each KPth flow cycle is printed. Default 1.
LL	Integer. Index of location on airfoil where the sweep through the upper and lower surfaces begins for the finite difference scheme. Default M/2+1. Smaller values of LL are used for high angles of attack.
LSEP	Integer. Index of x which gives the location at which the linear extrapolation for the

pressure distribution is begun on the upper surface. It should be placed at the point of separation, if used. The pressure distribution is modified from x at LSEP+1 to the trailing edge. If LSEP > M then the pressure distribution is not altered. Default M+1.

M	Integer. The number of mesh intervals in the angular direction in the circle plane at which the flow equations are solved. Default 160.
N	Integer. The number of mesh intervals in the radial direction in the circle plane. Default 30.
NFC	Integer. The number of Fourier coefficients used for the mapping. Default 80.
NPTS	Integer. The number of points at which the Nash-Macdonald boundary layer equation is solved. Default 81.
NRN	Integer. Run number. Default 1. If FSYM<3., NRN has the design value. If NRN > 1000, the Calcomp plots are done on blank paper on the CIMS CDC 6600.
NS	Integer. If positive and ITYP > 0 it is the total number of flow cycles to be computed before the next input card. Otherwise it is an indicator of the mode of operation. See Table 3. Default 1.
NS1	Integer. Number of flow cycles computed between boundary layer corrections. Default 20.
PCH	Real. Chord location at which the turbulent boundary layer calculation is begun. Transition is assumed to occur at this point. The program uses the x coordinate of the airfoil closest to PCH for transition. Default 0.07 unless in F mode, where the default is the peak pressure.
RBCP	Real. Relaxation parameter for iterating BCP. Default 0.1.
RCL	Real. Relaxation parameter for the circulation or the angle of attack. Default 1.
RDEL	Real. Relaxation parameter for the boundary layer displacement thickness. Default 0.125.
RFLO	Real. Relaxation parameter for the velocity potential in the flow calculation. Default 1.4.

RN	Real. Reynolds number. If RN is set to zero no boundary layer correction is made. Default 20.0E6.
SEPM	Real. Bound set on the separation parameter SEP. Default .004.
ST	Real. Convergence tolerance on the maximum velocity potential correction and the maximum circulation correction.
XMON	Real. x location where search for monotonicity of the pressure distribution is begun when modifying BCP for the pressure extrapolation. Default 0.95.
XP	Real. Indicator for test data. If XP > 0 then test data appear on Tape 4 and the points will appear on the Calcomp pressure distribution plots. If the program is used as F and XP < 0 then a plot of the airfoil is produced before and after the displacement thickness is subtracted. This plot is also obtained if the number of points defining the airfoil is greater than 140. If XP $\neq$ 0 and the program is in the F mode the redistributed coordinates are written on Tape 3 in FSYM = 1. format; if XP = 0 the displaced coordinates are written on Tape 3 in FSYM = 2. format.
XSEP	Real. $\|XSEP\|$ is the x location beyond which SEP assumes its calculated value even if SEP > SEPM. For all x < $\|XSEP\|$ the bound SEPM is imposed on the upper surface. If XSEP < 0 the upper and lower surfaces of the airfoil are both treated as upper surfaces. Default 0.93.

Glossary of Output Parameters

Printout of Original Airfoil Data

X,Y	Original airfoil coordinates smoothed IS times and redistributed if IS $\geq$ 0 and FSYM = 2. or 4.
ARC LENGTH	Arc length of airfoil defined by X,Y.
ANG	Surface angles of airfoil.
KAPPA	Curvature of airfoil.
KP	Second derivative of ANG with respect to arc length.
KPP	Third derivative of ANG with respect to arc length.

Printout from Mapping of Original Airfoil

ERR	Maximum correction in arc length for each iteration of the mapping. Convergence if $ERR < 0.4 \times 10^{-7}$.
DA,DB	Correction needed at each iteration to ensure closure.
A(NFC), B(NFC)	NFC Fourier coefficients. Printed out if IZ = 120.
EPSIL	Trailing edge angle divided by π.

Printout after Each Cycle of Flow

NCY	The running tally of flow cycles computed for a given grid size and Mach number.
DPHI	The maximum change in the velocity potential array at two consecutive flow cycles.
DCL	Change in lift necessary to satisfy the Kutta condition.
DDEL	Maximum increment in displacement thickness during each boundary layer calculation.
DBCP	Maximum residual in the base pressure BCP iteration.
IK, JK	The location of the maximum velocity potential correction. $1 < IK \leq M+1$, $1 \leq JK \leq N$.
NSP	The number of supersonic points in the flow calculation.
ALP	Angle of attack, which is printed if CL is held fixed.
CL	Coefficient of lift, which is printed if ALP is held fixed.
ANGO	Angle of zero lift. Computed after each mapping.
CPI	CP at LSEP . This is the first value used for the linear extrapolation if the pressure distribution is modified on the upper surface after separation. The pressure distribution is modified from LSEP+1 to the trailing edge.
BCP	Base pressure. The value to which the pressure is extrapolated at the trailing edge.
SL	Slope of the line through CPI and BCP.

Final Printout

XS	x coordinate of the last mapped outer airfoil at M+1 points.
YS	y coordinate of the last mapped outer airfoil at M+1 points.
ANG	Surface angles of the last mapped outer airfoil.
KAPPA	Local curvature of the last mapped airfoil.
MACH	Local Mach number resulting from the last flow cycle.
CP	Pressure distribution corresponding to the Mach number.
CP1	Pressure distribution used in the last boundary layer correction.
THETA	The momentum thickness obtained by solving the Nash-Macdonald equation in the last boundary layer correction cycle.
DELS	Displacement thickness obtained from the last boundary layer correction.
SEP	Quantity used as criterion for determining separation. If SEP > SEPM the boundary layer separates.
H	Shape factor.
DD	The last displacement thickness increment added to the inner airfoil.
CS	The location computed by the program at which SEP $\geq$ SEPM.
LM	The point at which the program starts looking for monotonicity in the pressure distribution.
LP	Gives the location of XSEP.
LS	Indicates the location of LSEP.
CDW	Wave drag coefficient.
CDF	Form drag coefficient.
CD	Total drag CD = CDW + CDF.
CM	Moment coefficient.

Parameters for Table 1

EPSIL	Real. Trailing edge angle divied by π.
FNU	Real. Number of points on upper surface defining airfoil.
FNL	Real. Number of points on lower surface defining airfoil.

CARDS \ COLS.	1 - 10	11 - 20	21 - 30	31 - 40
1	Title in Hollerith (Columns 2-17 will be printed on plot)			
2	FNU	FNL	EPSIL	
3	Blank			
4	Coordinates at nose			
⋮	Points on upper surface			
FNU + 3	Coordinates at trailing edge			
FNU + 4	Blank			
FNU + 5	Coordinates at nose			
⋮	Points on lower surface			
FNU+FNL+4	Coordinates at trailing edge			

Deck Structure

FSYM \ COLS.	1 - 10	11 - 20	21 - 30	31 - 40
3.0	u	v	x	y
4.0			x	y
5.0	x	y	θ°	

Data Structure

Table 1. Tape 3 Card Input for Program H.

COLS. CARDS	1 - 10	11-13	
1		NP	

COLS. CARDS	1 - 6	7 - 12	13 -18	19 - 25	26 - 34
2	EMX	ALPX	CLX	CDX	SNX

COLS. CARDS	1 - 10	11 - 20	
3	XL	CPX	
⋮	⋮	⋮	
NP + 2	XL	CPX	

Table 2. Deck and Data Structure for Tape 4.

NP	Integer. Number of comparison points.
EMX	Real. Free stream Mach number of comparison airfoil.
ALPX	Real. Angle of attack of comparison airfoil.
CLX	Real. Coefficient of lift for the comparison data.
CDX	Real. Coefficient of drag for the comparison data.
SNX	Real. Selects plotting symbol for comparison data. If positive, triangles (Δ) are used as for test data. If negative, plus signs (+) are used as for design data.
XL	Real. x coordinates of comparison data scaled from 0. to 1.
CPX	Real. Coefficient of pressure at corresponding XL values.

	ITYP < 0	ITYP=0	ITYP > 0
NS < 0	RETURN TO CONTROL MODE	TERMINATE PROGRAM	CRUDER GRID
NS = 0	STORE ON TAPE	TERMINATE PROGRAM	RETRIEVE FROM TAPE
NS > 0	FINER GRID	TERMINATE PROGRAM	FLOW COMPUTATION

Table 3. Control of Program H.

3. Operation of the Three Dimensional Analysis Program J

The three dimensional program was written specifically to treat the flow past a yawed wing as proposed by R. T. Jones. It will calculate the pressure distribution and force coefficients throughout the anticipated range of flight conditions up to Mach numbers of about 1.3 and yaw angles around 60°. At large Mach numbers and yaw angles, however, the artificial viscosity in the difference scheme causes the shock waves to become smeared.

The configuration is illustrated in Figure 1. To simplify the coordinate transformations the leading edge is assumed to be straight. The sheared parabolic coordinates described in Chapter I, Section 4, are then introduced in planes normal to the leading edge. The input parameters XSING and YSING determine the location of the singular line about which the square root transformation is made (see the Glossary, Section 4). It is important to choose these so that the unfolded profile does not have any sharp bumps. The mapped coordinates are printed so that this can be checked. The section can be varied in an arbitrary manner, and the planform can be tapered as desired by varying the location of the trailing edge. The trailing edge defined by the input is actually replaced by a piecewise straight line through the nearest mesh points in the computational lattice.

The geometry is defined by giving the cross section at successive span stations from the leading to the trailing tip of the yawed wing. Each section is defined by scaling and rotating a prescribed profile. The profile is given by a table of x,y coordinates. If the wing sections are all similar only the profile for the first span station is needed as input. The coordinates for the other stations are obtained by scaling the original profile to the

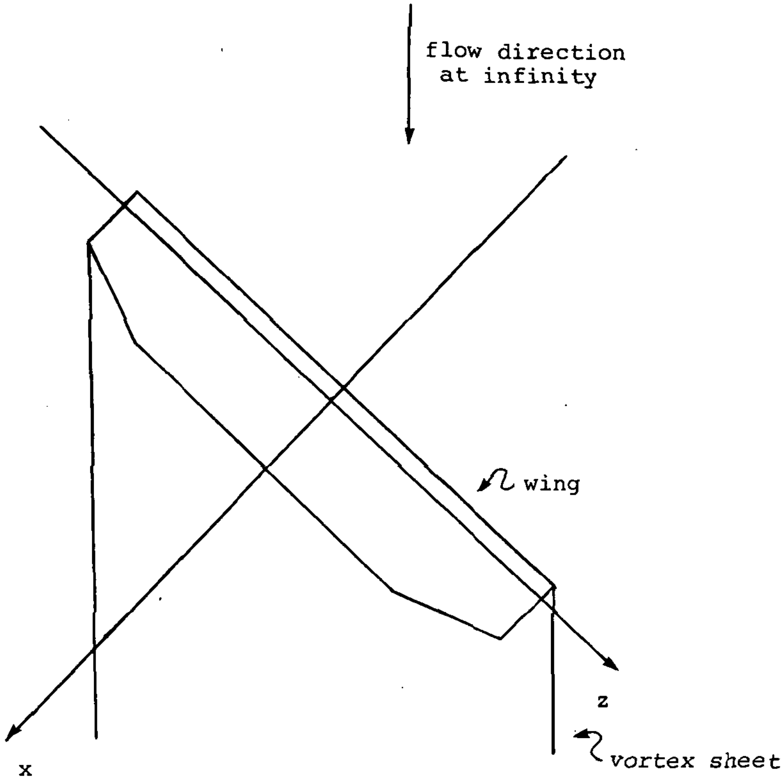


Figure 1. Configuration showing coordinate system relative to the body, with yaw angle introduced by rotating the flow at infinity.

proper chord, and rotating it to obtain the appropriate twist. If, on the other hand, the sections are not similar, the program permits profiles to be read in at each span station. The wing section between stations is generated by interpolation.

Another version of the program exists which allows for a curved leading edge. The parabolic coordinates are then introduced in planes parallel to the free stream, which leads to a skewed coordinate system. The resulting extra terms in the equations cause the computer time to be increased by about 30%. This version of the program has the advantage that it could be adapted to treat a swept back wing on a wall by the inclusion of a symmetry plane at the center line.

The difference scheme and iterative procedure conform closely to the description in Chapter I, Section 3. They are implemented as a line relaxation procedure in the x,y coordinate planes. These are updated in succession starting from the upstream side when the wing is yawed. In order to sweep in the general direction of the flow each x,y plane is divided into three strips. Then horizontal lines are relaxed in the middle strip, marching towards the body, and vertical lines are relaxed in each outer strip, marching outwards. The width of the center strip is determined by the parameter STRIP (see the Glossary, Section 4). Fastest convergence is usually obtained by using horizontal relaxation over the entire plane.

Normally calculations are first performed on a coarse mesh, and then on a fine mesh with twice as many intervals in each coordinate direction. The coarse mesh result is interpolated to provide the starting guess for the fine mesh. This procedure greatly reduces the computer time required for a fine mesh solution. Using the CDC 6600 it takes one second to sweep through about 4500 mesh points. The time for one iteration cycle on a mesh with 72×12×16 points is three seconds. A run usually consists of 200 cycles on such a coarse mesh, followed by 100 cycles on a fine mesh with 144×24×32 points. The total running time is about one hour.

The main input to the program is on Tape 5 and output is on Tape 6. Tapes 1, 2 and 3 are disc files used for internal storage in order to reduce the requirements for high speed memory. Tape 4 is a permanent storage device such as a magnetic tape on which intermediate results can be stored. The computation can be restarted and continued for more iterations using the data on Tape 4 as the new starting values. The disc instructions are specialized to the CDC 6600 using the FTN compiler. A version of the code which does not use disc storage is also available. This version should be readily adaptable to other computers, but requires a large amount of high speed memory.

The input data deck for a run is arranged to include title cards listing the required data items. The complete set of title cards provides a list of all the data which must be supplied, and can be used as a guide in setting up the data deck. Each title card is followed by one or more cards supplying the numerical values for the parameters listed. The input parameters are given in the Glossary, Section 4, in the order of their appearance on the data cards. All data items are read in as floating point numbers in fields of 10 columns, and values representing integer parameters are converted inside the program. The data deck for Airfoil 79-03-12 is shown in Table 1.

The output consists of printout and Calcomp plots. For convenience the section profile is printed at the first span station so that the input profile can be checked. If all the sections are similar only the chord and twist angle are printed at the remaining stations. If the sections are different the corresponding input profiles will be printed. The program next prints the mapped coordinates of the section at the wing center line, generated at the mesh points of the computational lattice. Parameters such as mesh size, Mach number, angle of yaw and angle of attack are also printed so that the case can easily be identified. Then for each iteration the program prints the iteration number, the maximum correction to the velocity potential and the maximum residual in satisfying the flow equation together with the coordinates of the points where these occur in the computational lattice, the circulation at the center section, the relaxation factors Rel Fct 1, Rel Fct 2 and Rel Fct 3 (see Glossary, Section 4), and the number of supersonic points.

After a maximum number of cycles has been completed or a convergence criterion has been satisfied the section lift, drag and moment coefficients are printed for each span station, starting with the leading tip; if desired, the section pressure distributions are also plotted. Finally the characteristics of the complete wing are printed. These include the coefficients of lift, form drag, friction drag and total drag, the ratios of lift to form drag and lift to total drag, and the pitching, rolling and yawing moments. In addition, charts are printed showing the Mach numbers at points in planes containing the upper and lower surfaces of the wing. A Calcomp plot is generated to show a view of the complete wing and the pressure distributions over the upper and lower surfaces separately, with the leading tip at the bottom of the picture. If the mesh is to be refined the program then repeats the same sequence of calculations and output on the new mesh.

4. Glossary and Table for Program J

Glossary of Input Parameters

The parameters are listed in the order of their occurrence on the data title cards (see Table 1).

TITLE CARD 1

NX	The number of mesh cells in the direction of the chord used at the start of the calculation. NX = 0 causes termination of the program.
NY	The number of mesh cells in the direction normal to the chord and span.
NZ	The number of mesh cells in the span direction.
FPLOT	Controls the generation of Calcomp plots. FPLOT = 0. for no plots. FPLOT = 1. for a three dimensional plot of the surface pressure distribution. FPLOT = 2. for a three dimensional plot and individual plots at each span station.
FCONT	Indicator which tells the manner of starting the program. FCONT = 0. indicates the calculation begins at iteration zero. FCONT = 1. indicates the computation is to be continued from a previous calculation. In this case the values of the velocity potential and the circulation are read from a magnetic tape where they were previously stored (Tape 4). It is still necessary to provide the complete data deck to redefine the geometry. The count of the iteration cycles is continued from the final count of the previous calculation so that the number of cycles NRELAX consists of the count of the previous calculation plus the number of iterations to be continued.

TITLE CARD 2

NRELAX	The maximum number of iteration cycles which will be computed.
RELAX TOL	The desired accuracy. If the maximum correction is less than RELAX TOL the calculation terminates or proceeds to a finer mesh, otherwise the number of cycles set by NRELAX are completed.
REL FCT 1	The subsonic relaxation factor for the velocity potential. It is between 1. and 2. and should be increased towards 2. as the mesh

	is refined.
REL FCT 2	The supersonic relaxation factor for the velocity potential. It is not greater than 1. and is normally set to 1.
REL FCT 3	The relaxation factor for the circulation. It is usually set to 1., but can be increased.
BETA	The damping parameter controlling the amount of added ϕ_{st} (see Chapter I, Section 3). It is normally set between 0. and 0.25.
STRIP	Determines the split between horizontal and vertical line relaxation and is the proportion of the total mesh in which horizontal line relaxation is used. Fastest convergence is usually obtained by setting STRIP = 1., where horizontal line relaxation is used for the entire mesh. If convergence difficulties are encountered STRIP may be reduced to some fraction between 0. and 1.
FHALF	Determines whether the mesh will be refined. FHALF = 0.: The computation terminates after completing the prescribed number of iteration cycles or after convergence for the input mesh size. FHALF $\neq$ 0.: The mesh spacing will be halved after NRELAX cycles have been run on the crude mesh size. An additional data card must be provided for the refined mesh giving the numerical values requested by Title Card 2. If FHALF < 0 the interpolated potential will be smoothed $\lvert \mathrm{FHALF} \rvert$ times.
TITLE CARD 3	(Aerodynamic Parameters)
FMACH	The free stream Mach number.
YAW	The yaw angle of the wing in degrees.
ALPHA	The angle of attack in degrees. When the wing is yawed, ALPHA is measured in the plane normal to the leading edge, not in the free stream direction.
CDO	The estimated parasite drag due to skin friction and separation. It is added to the pressure drag (sum of vortex drag plus wave drag) calculated by the program to give the total drag.
TITLE CARD 4	
NC	The number of span stations at which the wing section is defined on subsequent data cards from leading tip (smallest value of z) to trailing tip. If NC < 2 it is assumed that the wing geometry is the same as for the last

case calculated and the computation for new values of FMACH, YAW, ALPHA and CDO begins without any further data items being read.

TITLE CARD 5	(The Geometry at the First Span Station)
Z	Span location of the section.
CHORD	The local chord value by which the profile coordinates are scaled.
THICK	Modifies the section thickness. The Y coordinates are multiplied by THICK.
ALPHA	The angle through which the section is rotated to introduce twist. This angle is measured normal to the leading edge, not in the direction of the free stream.
NEWSEC	Indicates whether or not the geometry for a new profile is supplied. NEWSEC = 0.: The section is obtained by scaling the profile used at the previous span section according to the parameters CHORD, THICK, ALPHA. No further cards are read for this span station, and the next card should be the title card for the next span station, if any. NEWSEC = 1.: The coordinates for a new profile are read from the data cards which follow.
TITLE CARD 6	(Profile Geometry Supplied if NEWSEC = 1.)
ISYM	Indicates the type of profile. ISYM = 0. denotes a cambered profile. Coordinates are supplied for upper and lower surfaces, each ordered from nose to tail with the leading edge included in both surfaces. ISYM = 1. denotes a symmetric profile. A table of coordinates is read for the upper surface only.
NU	The number of upper surface coordinates.
NL	The number of lower surface coordinates. For ISYM = 1., NL = NU even though no lower surface coordinates are given.
TITLE CARD 7	(Additional Profile Geometry Supplied if NEWSEC = 1.)
TE ANGLE	The included angle at the trailing edge in degrees. The profile may be open, in which case it is the difference in angle between the upper and lower surfaces.
TE SLOPE	The slope of the mean camber line at the trailing edge. This is used to continue the coordinate surface, assumed to contain the

	vortex sheet, smoothly off the trailing edge. For heavily aft loaded airfoils, the lift is sensitive to the value of this parameter, which should be adjusted by comparing two dimensional calculations using parabolic coordinates with two dimensional calculations in the circle plane.
XSING, YSING	The coordinates of the singular point inside the nose about which the square root transformation is applied to generate parabolic coordinates. This point should be located as symmetrically as possible between the upper and lower surfaces at a distance from the nose roughly proportional to the leading edge radius. It can be seen whether the location has been correctly chosen by inspecting the coordinates of the mapped profile printed in the output. If the mapped profile has a bump at the center, the singular point should be moved closer to the leading edge. If the mapped profile is not symmetric near the center, with a step increase in y, say, as x increases through 0, the singular point should be moved closer to the upper surface. The coordinates of the singular point are chosen relative to the profile coordinates supplied on the cards which follow.
TITLE CARD 8	(Upper Surface Coordinates)
X,Y	The coordinates of the upper surface. These are read on the data cards which follow, one pair of coordinates per card in the first two fields of 10, from leading to trailing edge inclusive.
TITLE CARD 9	(Lower Surface Coordinates, Read if ISYM = 0.)
X,Y	The coordinates of the lower surface, read from leading edge to trailing edge. The leading edge point is the same as the upper surface leading edge point. The trailing edge point may be different if the profile has an open tail.
TITLE CARDS 10,11,...	(Geometry at the Other Span Stations)
	These title cards are the same as Title Card 5 (geometry for the first span station). The number of such cards depends on the number of input span stations NC. If the profiles are similar at each station except for scaling, thickness chord ratio and rotation to introduce twist, NEWSEC = 0. and no new profile coordinates are needed.

Cols. Title Card	1-10	11-20	21-30	31-40	41-50	51-60	61-70	71-80
	79-03-12 Section AR = 6.0 Twist 2 Deg (Columns 1-48 Copied on CALCOMP Plots)							
1	NX	NY	NZ	FPLOT	FCONT			
	72.	12.	16.	1.	0.			
2	NRELAX	RELAX TOL	REL FCT 1	REL FCT 2	REL FCT 3	BETA	STRIP	FHALF
	200.	1.E-6	1.6	1.	1.	.05	1.	1.
	100.	1.E-6	1.75	1.	1.	.05	1.	0.
3	FMACH	YAW	ALPHA	CDO				
	.830	15.	1.2	.010				
4	NC							
	4.							
5	Z	CHORD	THICK	ALPHA	NEWSEC			
	-270.	60.	1.	1.0	1.			
6	ISYM	NU	NL					
	0.	82.	80.					
7	TE ANGLE	TE SLOPE	XSING	YSING				
	0.	-.047	.0085	.0164				
8	X	Y	(Upper Surface Coordinates Nose to Tail)					
1. ⋮ NU								
9	X	Y	(Lower Surface Coordinates Nose to Tail)					
1. ⋮ NL								
10	Z	CHORD	THICK	ALPHA	NEWSEC			
	-135.0	100.	1.	0.33	0.			
11	Z	CHORD	THICK	ALPHA	NEWSEC			
	135.0	100.	1.	-0.33	0.			
12	Z	CHORD	THICK	ALPHA	NEWSEC			
	270.0	60.	1.	-1.0	0.			

Table 1. Data Deck for Three Dimensional Program.

```
      PROGRAM H(INPUT = 66,OUTPUT = 500,TAPE3 = 500,TAPE4 = 400,TAPE2 =
     1OUTPUT,TAPE5 = INPUT)
      COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
     1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CO(162)
     2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162),DSUM(162)
     3 ,ANGOLU(162),XOLD(162),YOLD(162),ARCOLD(162),UELOLU(162)
      COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP,TC,CHD,DPHI,CL,RCL,YR
     1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,UCN,DSN,RA4,EPSIL,QCRIT,C1,C2
     2 ,C4,C5,C6,C7,BET,BETA,FSYM,XSEP,SEPM,TTLE(4),M,N,MM,NN,NSP
     3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCY,NRN,NG,IDIM,N2,N3,N4,NT,IXX
     4 , NPTS,LL,I,LSEP,M4
      DIMENSION COMC(68),CLA(2),NAMERR(6)
      EQUIVALENCE (COMC(1),PI),(CLX,CLA(1)),(ALPX,CLA(2))
C     LSTERR IS THE SUBROUTINE TO PROCESS A NAMELIST ERROR
      EXTERNAL LSTERR
C     ***NON-ANSI***
      NAMELIST /P/ ALP,BETA,BCP,CL,EM,FSYM,GAMMA,IS,ITYP,IZ,KP,LL,LSEP,
     1   M,N,NFC,NPTS,NRN,NS,NS1,PCH,RBCP,RCL,RDEL,RFLO,RN,SEPM,ST,
     2  XMON,XP,XSEP
      DATA GAMMA/1.4/ , ST/0./ , XMON/.95/  , RBCP/.10/ , RFLO/1.4/ ,
     1                RDEL/.125/ , BCP/.4/ , NS1/20/ , NS/1/ , KP/1/
      DATA N5/5/ , NAMERR/6*0/ , D1,U2,SL/3*0./ , CP1/.4/ ,XPF/1./
C     THESE TWO CARDS TRANSMIT TO THE SYSTEM THE RECOVERY ADDRESS
      NAMERR(5) = LOCF(LSTERR)
      CALL SYSTEMC(66,NAMERR)
      M4 = N4
      REWIND N4
      WRITE (N2,180)
      READ (N5,P)
      IF (CL.NE.100.) MODE = 0
      IF (IZ.GE.80) N4 = N2
      IF (NS.EQ.0) GO TO 30
C     SET UP CONSTANTS AND DO CONFORMAL MAPPING
      CALL RESTRT
      CLX = CL
      ALPX= RAD*ALP
      GO TO 140
   10 WRITE (N2,180)
      ALP = 100.
      CL = 100.
C     ****NON-ANSI****
      READ (N5,P)
C     SELECT OUTPUT TAPE
      N4 = M4
      IF (IZ.GE.80) N4 = N2
      C2 = .5*(GAMMA-1.)
      C7 = GAMMA/(GAMMA-1.)
      IF (ALP.EQ.100.) GO TO 20
C     ALP HAS BEEN INPUTTED, KEEP IT FIXED
      NCY = 0
      MODE = 1
      ALPX = ALP
   20 ALP = ALPX/RAD
```

```
      IF (CL.EQ.100.) GO TO 25
C     CL HAS BEEN INPUTTED, KEEP IT FIXED
      NCY = 0
      MODE = 0
      YA = .5*CL/CHD-DPHI
      DO 114 L = 1,M
      DO 114 J = 1,NN
  114 PHI(L,J) = PHI(L,J)+YA*PHIR(L)
      DPHI = .5*CL/CHD
      CLX = CL
   25 CL = CLX
C     CHANGE PARAMETERS WHICH DEPEND ON THE MACH NUMBER
      EM  = AMAX1(EM,.1E-40)
      IF (EM.NE.EMX) NCY = 0
      C1 = C2+1./(EM *EM )
      C6 = C2*EM *EM
      C4 = 1.+C6
      C5 = 1./(C6*C7)
      QCRIT = (C1+C1)/(GAMMA+1.)
      BET = SQRT(1.-EM *EM )-1.
C     CHECK FOR TERMINATE,RETRIEVE, OR STORE INSTRUCTIONS
C     IK WILL BE -1 ONLY IF THERE IS A NAMELIST ERROR
      IF ((ITYP.EQ.0).OR.(IK.EQ.-1)) GO TO 170
      CALL COSI
      IF (NS.NE.0) GO TO 40
      REWIND N3
      IF (ITYP.GT.0) GO TO 30
      WRITE(N3) COMC,PHI,AA,BB,ARCOLD,ANGOLD,XOLD,YOLD,DELOLD,R,RS,RI
     1 ,DSUM,GAMMA,XMON,RBCP,RFLO,RDEL,BCP,NS1,KP,ST
      GO TO 140
   30 READ (N3) COMC,PHI,AA,BB,ARCOLD,ANGOLD,XOLD,YOLD,DELOLD,R,RS,RI
     1 ,DSUM,GAMMA,XMON,RBCP,RFLO,RDEL,BCP,NS1,KP,ST
      CALL MAP
      GO TO 140
   40 CONTINUE
      IF (NS.GT.0) GO TO 70
      NS = 0
C     GO TO CRUDE GRID IF ITYP.GT.0
      IF (ITYP.GT.0) CALL REMESH(-1)
      GO TO 140
   70 IF (ITYP.GT.0) GO TO 100
C     GO BACK TO FINER GRID
      CALL REMESH(1)
      GO TO 140
  100 XPHII = 0.
      IF ( RCL.NE.0.) XPHII = 2.*CHD/RCL
      XA = 1.-2./RFLO
      ANGO = -RAD*BB(1)
      TXT = 3H CL
      IF (MODE.EQ.0) TXT = 3HALP
C     NO BOUNDARY LAYER CORRECTIONS ARE MADE FOR RN.LE.0.
      IF (RN.LE.0.) NS1 = 1000000
      IXX = M+2
   80 IXX = IXX-1
      IF (XC(IXX-1).GT.XMON) GO TO 80
```

```
      LC = 0
C     DO AT MOST NS CYCLES
      DO 120 K = 1,NS
      IF (MOD(LC,56).NE.0) GO TO 105
      WRITE (N2,210) TXT
      LC = LC+1
  105 CALL SWEEP
C     KEEP TRACK OF TOTAL NUMBER OF CYCLES
      NCY = NCY+1
      ALPX = RAD*ALP
      CLX= 2.*DPHI*CHD
      YA = YA*XPHII
C     WRITE RESIDUALS ON N2 EVERY KP CYCLES
      IF (MOD(K,KP).NE.0) GO TO 110
      LC = LC+1
      WRITE (N2,190) NCY,YR,YA,D1,D2,IK,JK,NSP,CLA(2-MODE),ANGO,CP1,
     1 BCP,SL
C     DO A BOUNDRY LAYER CORRECTION EVERY NS1 CYCLES
  110 IF (MOD(K,NS1).NE.0) GO TO 125
      IF (K.EQ.NS) GO TO 140
      WRITE (N2,190)
      LC = LC+1
      FSYM = 6.
      CALL GTURB(D1,D2,CP1,BCP,SL,RDEL,RBCP)
      ANGO = -RAD*BB(1)
      IF (MODE.EQ.0) DPHI = .5*CLX/CHD
C     CHECK TO SEE IF WE HAVE SATISFIED CONVERGENCE CRITERIA
  125 IF (AMAX1(ABS(YR),ABS(YA)).LT.ST) GO TO 140
  120 CONTINUE
  140 ITYP = IABS(ITYP)
      CL = CLX
      LN = RN*1.E-6+.5
      XPF = XPF*AMIN0(1,IABS(M4-N4))
      XP = XP*XPF
      CALL SECOND(TIME)
      NTPE = N4
      TXT = 3HALP
      IF (MODE.EQ.0) TXT = 3H CL
  150 WRITE (NTPE,200) EM,TXT,CLA(MODE+1),LN,M,N,NS,TIME,RFLO,RCL,RDEL,
     1 RBCP,BETA,ST,PCH,SEPM,XSEP,NPTS,IS,LL,IZ
      IF (NTPE.EQ.N2) GO TO 160
      NTPE = N2
      GO TO 150
  160 IF (ITYP.GE.2) CALL GTURB(D1,D2,CP1,BCP,SL,RDEL,RBCP)
      EMX = EM
      ITYP=1
      GO TO 10
  170 ITYP = 4
      IF (IK.EQ.-1) WRITE (N4,220)
      CALL GTURB(D1,D2,CP1,BCP,SL,RDEL,RBCP)
C     TERMINATE PLOT
      CALL PLOT(0.,0.,999)
      CALL EXIT
  180 FORMAT (7H READ P/)
  190 FORMAT (5X,I4,4E12.3,I4,I3,I6,2F10.4,3F11.5)
```

```
200 FORMAT (4H0EM=F4.3,3XA3,1H=F5.2,3X3HRN=I2,2HE6,3X4HM*N=,I3,1H*,I2,
   1 3X3HNS=I4,3X5HTIME=F7.2/6H RFLO=F4.2.3X,4HRCL=F4.2,3X5HRDEL=F4.3
   2 ,3X5HRBCP=F3.2,3X5HBETA=F4.2,3X3HST=.E7.1/ 5H PCH=F4.2,
   3  3X5HSEPM=F5.4,3X5HXSEP=F4.2,3X5HNPTS=I3,3X3HIS=I2,3X3HLL=I3,
   4 3X3HIZ=I3//)
210 FORMAT(1H15X3HNCY6X4HDPHI8X3HDCL,8X,4HDDEL,8X,4HDBCP,5X,2HIK,
   1 2X,2HJK,2X3HNSP,5XA4,5X4HANG0,8X3HCPT,8X3HBCP,8X2HSL/)
220 FORMAT (21H0***NAMELIST ERROR***,10X,20HPROGRAM TO TERMINATE   )
    END




    SUBROUTINE LSTERR
    COMMON /A/ M(47),IK
    IK = -1
    RETURN
    END




    SUBROUTINE RESTRT
    COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
   1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CO(162)
   2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162),DSUM(162)
   3 ,ANGOLD(162),XOLD(162),YOLD(162),ARCOLD(162),DELOLD(162)
    COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP.TC,CHD,DPHI,CL,RCL,YR
   1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,DCN,DSN,RA4,EPSIL,QCRIT,C1,C2
   2 ,C4,C5,C6,C7,BET,BETA,FSYM,XSEP,SEPM.TTLE(4),M,N,MM,NN,NSP
   3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCY,NRN,NG,IDIM,N2,N3,N4,NT,IXX
   4 , NPTS,LL,I,LSEP,M4
C    SET UP CONSTANTS
    TP = PI+PI
    RAD = 180./PI
    ALP = ALP/RAD
    IF ((N+1).NE.NN.OR.(M+1).NE.MM) NCY = 0
    MM = M+1
    IF (LL.EQ.0) LL = M/2+1
    NN = N+1
    DR = -1./FLOAT(N)
    DT = TP/FLOAT(M)
    DCN = COS(DT)
    DSN = SIN(DT)
    DELR = .5/DR
    DELTH = .5/DT
    RA = DT/DR
    RA4 = DT*DT
    DO 10 K = 1,N
    R(K) = 1.+DR*FLOAT(K-1)
    RS(K) = (RA*R(K))*(RA*R(K))
    RI(K) = -.25*DT/R(K)
 10 CONTINUE
    R(NN) = 0.
    BET = SQRT(1.-EM*EM) -1.
```

```
C     DO MAPPING
      CALL AIRFOL
      IF (MODE.EQ.1) CL = 8.*PI*CHD*SI(1)/(1.+BET)
      DPHI = .5*CL/CHD
C     SELECT NT OF THE MM MAPPED COORDINATES
      MA = MM/3
      MB = MM-2*((MA+1)/2)
      IF((NT.GT.140).OR.(XP.LT.0.)) JK = -1
      J=1
      DO 40 L = 1,MM
      DELOLD(L) = 0.
      DSUM(J) = 0.
      ARCOLD(L)=ARCL(J)
      IF(J.GE.MM) GO TO 70
      IF((J.LT.MA ).OR.(J.GE.MB)) J=J+1
      DSUM(J) = 0.
      J=J+1
   40 CONTINUE
   70 NT = L
      WRITE (N4,100) NT
  100 FORMAT (1H0,I4,45H POINTS WILL BE USED TO DEFINE INNER AIRFOIL )
      CALL SPLIF(MM,ARCL,XC,PHI(1,3),PHI(1,5),PHI(1,7),3,0.,3,0.)
      CALL INTPL(NT,ARCOLD,  XOLD,ARCL,XC,PHI(1,3),PHI(1,5),PHI(1,7))
      CALL SPLIF(MM,ARCL,YC,PHI(1,3),PHI(1,5),PHI(1,7),3,0.,3,0.)
      CALL INTPL(NT,ARCOLD,  YOLD,ARCL,YC,PHI(1,3),PHI(1,5),PHI(1,7))
      CALL SPLIF(MM,ARCL,FM,PHI(1,3),PHI(1,5),PHI(1,7),3,0.,3,0.)
      CALL INTPL(NT,ARCOLD,ANGOLD,ARCL,FM,PHI(1,3),PHI(1,5),PHI(1,7))
      DO 60 L = 1,M
      DO 50 J = 1,NN
   50 PHI(L,J) = R(J)*CO(L)+DPHI*PHIR(L)
   60 CONTINUE
      FSYM = FSYM-12.
      IS = 2
      RETURN
      END




      SUBROUTINE COSI
C     SET THE SINES,COSINES, AND THE TERM AT INFINITY
      COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
     1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CO(162)
     2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162),DSUM(162)
     3 ,ANGOLD(162),XOLD(162),YOLD(162),ARCOLD(162),DELOLD(162)
      COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP,TC,CHD,DPHI,CL,RCL,YR
     1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,DCN,DSN,RA4,EPSIL,QCRIT,C1,C2
     2 ,C4,C5,C6,C7,BET,BETA,FSYM,XSEP,SEPM,TTLE(4),M,N,MM,NN,NSP
     3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCY,NRN,NG,IDIM,N2,N3,N4,NT,IXX
     4 , NPTS,LL,I,LSEP,M4
      TPI = 1./TP
      ANG = ALP+BB(1)
      SN = SIN(ANG)
      CN = SQRT (1.-SN*SN)
      DO 10 L = 1,M
```

```
      CO(L) = CN
      SI(L) = SN
      PHIR(L) =(ANG+ATAN((BET*SN*CN)/(1.+BET*SN*SN)))*TPI
      CN = CN*DCN-SN*DSN
      SN = CO(L)*DSN+SN*DCN
      ANG = ANG+DT
   10 CONTINUE
      CO(MM) = CN
      CO(MM+1) = CO(2)
      SI(MM) = SN
      SI(MM+1) = SI(2)
      RETURN
      END




      SUBROUTINE SWEEP
C     SWEEP THROUGH THE GRID ONE TIME
      COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
     1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CO(162)
     2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162),DSUM(162)
     3 ,ANGOLD(162),XOLD(162),YOLD(162),ARCOLD(162),DELOLD(162)
      COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP,TC,CHD,DPHI,CL,RCL,YR
     1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,DCN,DSN,RA4,EPSIL,QCRIT,C1,C2
     2 ,C4,C5,C6,C7,BET,BETA,FSYM,XSEP,SEPM,TTLE(4),M,N,MM,NN,NSP
     3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCY,NRN,NG,IDIM,N2,N3,N4,NT,IXX
     4 , NPTS,LL,I,LSEP,M4
      YR = 0.
      NSP = 0
      DO 10 J = 1,NN
      PHI(MM,J) = PHI(1,J)+DPHI
      PHI(MM+1,J) = PHI(2,J)+DPHI
      E(J) = 0.
   10 RPP(J) = 0.
C     SWEEP THROUGH THE GRID FROM NOSE TO TAIL ON UPPER SURFACE
      TE = -2.
      DO 30 I = LL,MM
      CALL MURMAN
      DO 30 J = 1,N
   30 PHI(I-1,J) = PHI(I-1,J)-RP(J)
C     UPDATE PHI AT THE TAIL FROM UPPER SURFACE
      DO 50 J = 1,N
      PHI(MM,J) = PHI(MM,J)-E(J)
      E(J) = 0.
      RPP(J) = 0.
   50 PHI(1,J) = PHI(MM,J)-DPHI
C     SWEEP THROUGH THE GRID FROM NOSE TO TAIL ON LOWER SURFACE
      TE = 2.
      I = LL
   80 I = I-1
      CALL MURMAN
      DO 60 J = 1,N
   60 PHI(I+1,J) = PHI(I+1,J)-RP(J)
      IF (I.GT.2) GO TO 80
```

```
      DO 70 J = 1,N
   70 PHI(2,J) = PHI(2,J)-E(J)
C     ADJUST CIRCULATION TO SATISFY THE KUTTA CONDITION
      IF (RCL .EQ.0.) GO TO 90
      YA = RCL*((PHI(M,1)-(PHI(2,1)+DPHI))*DELTH+SI(1))
      IF (MODE.EQ.1) GO TO 90
      ALP = ALP-.5*YA
      CALL COSI
      GO TO 95
   90 YA = TP*YA/(1.+BET)
      DPHI = DPHI+YA
   95 DO 97 L = 1,M
   97 PHI(L,NN) = DPHI*PHIR(L)
      IF(MODE.EQ.0) RETURN
      DO 100 J = 1,N
      DO 100 L = 1,M
  100 PHI(L,J) = PHI(L,J)+YA*PHIR(L)
      RETURN
      END



      SUBROUTINE MURMAN
C     SET UP COEFFICIENT ARRAYS FOR THE TRIDIAGONAL SYSTEM USED FOR LINE
C     RELAXATION AND COMPUTE THE UPDATED PHI ON THIS LINE
      COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
     1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CO(162)
     2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162),DSUM(162)
     3 ,ANGOLD(162),XOLD(162),YOLD(162),ARCOLD(162),DELOLD(162)
      COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP,TC,CHD,DPHI,CL,RCL,YR
     1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,DCN,DSN,RA4,EPSIL,QCRIT,C1,C2
     2 ,C4,C5,C6,C7,BET,BETA,FSYM,XSEP,SEPM,TTLE(4),M,N,MM,NN,NSP
     3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCY,NRN,NG,IDIM,N2,N3,N4,NT,IXX
     4 , NPTS,LL,I,LSEP,M4
C     DO THE BOUNDARY
      E(NN) = 0.
      FAC = -.5*TE
      IM = I-1
      IF (FAC.LT.0.) IM = I+1
      KK = 0
      PHIO = PHI(I,2)-2.*DR*CO(I)
      PHIYP= PHI(I,2)-PHI(I,1)
      PHIYY = PHIYP+PHIO-PHI(I,1)
      PHIXX = PHI(I+1,1)+PHI(I-1,1)-PHI(I,1)-PHI(I,1)
      PHIXM = PHI(I+1,1)-PHI(I-1,1)
      PHIXP = PHI(I+1,2)-PHI(I-1,2)
C     CHECK FOR THE TAIL POINT
      IF (I.NE.MM) GO TO 10
      C(1) = (C1+C1)*RS(1)
      A(1) = -C(1)+XA*C1-C1
      D(1) = C1*(PHIXX+RS(1)*PHIYY+RA4*CO(I)-E(1))
      GO TO 40
   10 U = PHIXM*DELTH-SI(I)
      BQ = U/FP(I,1)
```

```
      QS = U*BQ
      CS = C1-C2*QS
      BQ = BQ*QS*(FP(I-1,1)-FP(I+1,1))
      X = RA4*(CS+QS)*CO(I)
      C(1) = (CS+CS)*RS(1)
      D(1) = CS*RS(1)*PHIYY+RI(1)*BQ+X
      CMQS = CS-QS
      PHIXT = BETA*ABS(U)+ABS(CMQS)
      IF (QS.LE.QCRIT) GO TO 30
C     FLOW IS SUPERSONIC, BACKWARD DIFFERENCES
      KK = 1
      PHIXT = PHIXT-CMQS
      PHIXXM = RPP(1)
      A(1) = -(C(1)+PHIXT)
      D(1) = D(1)+CMQS*PHIXXM-PHIXT*E(1)
      GO TO 40
C     FLOW SUBCRITICAL, CENTRAL DIFFERENCES
   30 A(1) = XA*CMQS -C(1)-PHIXT
      D(1) = D(1)+CMQS*PHIXX-PHIXT*E(1)
C     DO NON-BOUNDARY POINTS
   40 RPP(1) = PHIXX
      DO 60 J = 2,N
      PHIXX = PHI(I+1,J)+PHI(I-1,J)-PHI(I,J)-PHI(I,J)
      DU = PHIXP
      PHIXP = PHI(I+1,J+1)-PHI(I-1,J+1)
      PHIXY = PHIXP-PHIXM+(E(J+1)-E(J-1))*FAC
      PHIXM = DU
      DU = DU*DELTH
      PHIYYM = PHIYY
      PHIYM = PHIYP
      PHIYP = PHI(I,J+1)-PHI(I,J)
      PHIYY = PHIYP-PHIYM
      U = R(J)*DU-SI(I)
      DV = R(J)*(PHI(I,J+1)-PHI(I,J-1))*DELR
      V = DV*R(J)-CO(I)
      RAV = R(J)*RA*V
      BQ = 1./FP(I,J)
      BQU = BQ*U
      US = BQU*U
      UV = (BQU+BQU)*V
      VS = BQ*V*V
      QS = US+VS
      CS = C1-C2*QS
      CMVS = CS-VS
      CMUS = CS-US
      PHIXT = BETA*ABS(U)
      PHIYT = BETA*ABS(RAV)
C     COMPUTE CONTRIBUTION OF RIGHT-HAND SIDE FROM LOW ORDER TERMS
      D(J) =RA4*((CMVS+US-VS)*DV-UV*DU)+RI(J)*QS*BQ*(U*(FP(I-1,J)-
     1 FP(I+1,J))+RAV*(FP(I,J-1)-FP(I,J+1)))
      UV = .5*BQU*RAV
      IF (QS.LE.QCRIT) GO TO 50
C     SUPERSONIC FLOW, USE BACKWARD DIFFERENCING
      KK = KK+1
      CMQS = CS-QS
```

```
      FQ = 1./QS
      AUU = US*FQ
      BUU = RS(J)*AUU
      BVV = VS*FQ
      AVV = RS(J)*BVV
      BUV = UV*FQ
      AUV = BQU*ABS(RAV)*FQ*TE
      PHINN = BVV*PHIXX-BUV*PHIXY+BUU*PHIYY
      B(J) =  CS*BUU
      PHIXT = PHIXT-CMQS*(AUU+AUU-AUV) +CS*AVV
      PHIYT = PHIYT -CMQS*(AVV+AVV-AUV)
      C(J) = B(J)+PHIYT
      PHIXXM = RPP(J)
      IF (V.LT.0) GO TO 45
      PHIYYM = PHI(I,J+2)-PHI(I,J+1)-PHIYP
      PHIXYM = PHIYP+PHI(IM,J)-PHI(IM,J+1)
      GO TO 46
   45 PHIXYM = PHI(IM,J)-PHI(IM,J-1)-PHIYM
      BQ = B(J)
      B(J) = C(J)
      C(J) = BQ
   46 PHISS = AUU*PHIXXM+AUV*PHIXYM+AVV*PHIYYM
      A(J) = -(B(J)+C(J)+PHIXT)
      D(J) = D(J)+CMQS*PHISS+CS*PHINN-E(J)*PHIXT
      GO TO 60
C     SUBSONIC FLOW, USE CENTRAL DIFFERENCES
   50 C(J) =  RS(J)*CMVS
      B(J) = C(J)+PHIYT
      PHIXT = PHIXT+CMUS
      A(J) =  XA*CMUS-B(J)-C(J)-PHIXT
      D(J) = D(J)+CMUS*PHIXX-UV*PHIXY+C(J)*PHIYY-PHIXT*E(J)
      IF (V.LT.0.) GO TO 60
      B(J) = C(J)
      C(J) = C(J)+PHIYT
   60 RPP(J) = PHIXX
      NSP = NSP+KK
C     SOLVE THE TRIDIAGONAL SYSTEM
      CALL TRIO
      RETURN
      END




      SUBROUTINE TRIO
C     SOLVE N DIMENSIONAL TRIDIAGONAL SYSTEM OF EQUATIONS
      COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
     1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CO(162)
     2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162),DSUM(162)
     3 ,ANGOLD(162),XOLD(162),YOLD(162),ARCOLD(162),DELOLD(162)
      COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP,TC,CHD,DPHI,CL,RCL,YR
     1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,DCN,DSN,RA4,EPSIL,QCRIT,C1,C2
     2 ,C4,C5,C6,C7,BET,BETA,FSYM,XSEP,SEPM,TTLE(4),M,N,MM,NN,NSP
     3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCY,NRN,NG,IDIM,N2,N3,N4,NT,IXX
     4 , NPTS,LL,I,LSEP,M4
```

```
      XX = 1./A(1)
      RP(1) = E(1)
      E(1) = XX*D(1)
C     DO ELIMINATION
      DO 10 J = 2,N
      C(J-1) = C(J-1)*XX
      XX = 1./(A(J)-B(J)*C(J-1))
      RP(J) = E(J)
   10 E(J) = (D(J)-B(J)*E(J-1))*XX
C     DO BACK SUBSTITUTION
      EMX = ABS(E(N))
      DO 20 J = 2,N
      L = NN-J
      E(L) = E(L)-C(L)*E(L+1)
   20 EMX = AMAX1(EMX,ABS(E(L)))
C     FIND THE LOCATION OF THE MAXIMUM RESIDUAL
      IF (EMX.LE.ABS(YR)) RETURN
      IK = I
      DO 70 J = 1,N
      IF (ABS(E(J)).EQ.EMX) GO TO 74
   70 CONTINUE
   74 JK = J
      YR = E(JK)
      RETURN
      END




      SUBROUTINE REMESH(LSIGN)
C     GO TO CRUDER GRID IF LSIGN IS -1
C     GO TO FINER GRID IF LSIGN IS +1
      COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
     1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CO(162)
     2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162),DSUM(162)
     3 ,ANGOLD(162),XOLD(162),YOLD(162),ARCOLD(162),DELOLD(162)
      COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP,TC,CHD,DPHI,CL,RCL,YR
     1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,DCN,DSN,RA4,EPSIL,QCRIT,C1,C2
     2 ,C4,C5,C6,C7,BET,BETA,FSYM,XSEP,SEPM,TTLE(4),M,N,MM,NN,NSP
     3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCY,NRN,NG,IDIM,N2,N3,N4,NT,IXX
     4 , NPTS,LL,I,LSEP,M4
      X = 2.**LSIGN
      NG = FLOAT(NG)/X+.2
      M = FLOAT(M)*X +.2
      N = FLOAT(N)*X+.2
      LL = FLOAT(LL-1)*X+1.2
      IF (LSIGN.GT.0) MM = M+1
      IF (LSIGN.GT.0) NN = N+1
      LSEP = FLOAT(LSEP-1)*X+1.2
      PF = 1./X
      DELR = X*DELR
      DELTH = X*DELTH
      DR = PF*DR
      DT = PF*DT
      DCN = COS(DT)
```

```
      DSN = SIN(DT)
      RA4 = PF*PF*RA4
      NCY = 0
      I = LSIGN
      MP = MM+1
      CALL PERMUT (R,NN,1)
      CALL PERMUT (RS,NN,1)
      DO 5 J = 1,N
    5 RI(J) = -.25*DT/R(J)
      CALL PERMUT (DSUM,MP,1)
      DO 20 L = 1,NN
   20 CALL PERMUT (PHI(1,L),MP,1)
      DO 30 L = 1,MP
   30 CALL PERMUT (PHI(L,1),NN,IDIM)
      MM = M+1
      NN = N+1
      IF (X.EQ..5) GO TO 80
      DO 40 L = 1,M,2
      DSUM(L+1) = .5*(DSUM(L)+DSUM(L+2))
      DO 40 J = 1,NN,2
   40 PHI(L+1,J) = .5*(PHI(L,J)+PHI(L+2,J))
      DO 50 J = 1,N,2
      DO 50 L = 1,MM
   50 PHI(L,J+1) = .5*(PHI(L,J)+PHI(L,J+2))
   80 CALL MAP
      RETURN
      END



      SUBROUTINE PERMUT (AX,NX,JX)
C     REORDERS POINTS WITHIN AN ARRAY
      COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
     1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CO(162)
     2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162),DSUM(162)
     3 ,ANGOLD(162),XOLD(162),YOLD(162),ARCOLD(162),DELOLD(162)
      COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP.TC,CHD,DPHI,CL,RCL,YR
     1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,DCN,DSN,RA4,EPSIL,QCRIT,C1,C2
     2 ,C4,C5,C6,C7,BET,BETA,FSYM,XSEP,SEPM.TTLE(4),M,N,MM,NN,NSP
     3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCY,NRN,NG,IDIM,N2,N3,N4,NT,IXX
     4 , NPTS,LL,I.LSEP,M4
      DIMENSION AX(1)
      L = 1
      JY = JX+JX
      NY = 2*((NX-1)/2)+1
      NZ = 2*(NX/2)
      IF(I.GT.0) GO TO 30
      NY = JX*(NY-1)+1
      NZ = JX*(NZ-1)
      DO 10 J = 1,NY,JY
      A(L) = AX(J)
   10 L = L+1
      DO 20 J = JX,NZ,JY
      A(L) = AX(J+1)
```

```
   20 L = L+1
      GO TO 60
   30 DO 40 J = 1,NY,2
      A(J) = AX(L)
   40 L = L+JX
      DO 50 J = 2,NZ,2
      A(J) = AX(L)
   50 L = L+JX
   60 L = 1
      DO 70 J = 1,NX
      AX(L) = A(J)
   70 L = L+JX
      RETURN
      END
```

```
      SUBROUTINE GETCP(CDF)
C     COMPUTE CP,CD, AND CM BY INTEGRATION AND OUTPUT MACH DIAGRAM
      COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
     1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CO(162)
     2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162),DSUM(162)
     3 ,ANGOLD(162),XOLD(162),YOLD(162),ARCOLD(162),DELOLD(162)
      COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP.TC,CHD,DPHI,CL,RCL,YR
     1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,DCN,DSN,RA4,EPSIL,QCRIT,C1,C2
     2 ,C4,C5,C6,C7,BET,BETA,FSYM,XSEP,SEPM.TTLE(4),M,N,MM,NN,NSP
     3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCY,NRN,NG,IDIM,N2,N3,N4,NT,IXX
     4 , NPTS,LL,I,LSEP,M4
      REAL MACHN,MACH
      COMPLEX CLCD,TMP
      DIMENSION MACHN(1),CPX(1),MN(1),IMACH(21)
      EQUIVALENCE (MACHN(1),A(1)),(CPX(1),PHIR(1)),(MN(1),FP(1,31))
      DATA IMACH/1HQ,1HR,1HS,1HT,1HU,1HV,1HW,1HX,1HY,1HZ,1H0,1H1,1H2,1H3
     1,1H4,1H5,1H6,1H7,1H8,1H9,1H+/
      DATA TX /4HCDF=/
      MACH(Q) = SQRT(Q/(C1-C2*Q))
      IMC(Q) = MIN0(21,IFIX(10.*Q)+1)
      CLCD = 0.
      CM = 0.
      IF ((XP.GT.0.).OR.(IZ.LE.80)) GO TO 10
      DY = YOLD(NT)-YOLD(1)
      REWIND M4
      WRITE (M4,120) EM,CL,DY,TC, NRN,MM
   10 DO 20 L = 1,MM
      CP = CPX(L)
C     COMPUTE CP*DZ
      TMP =  CP*SQRT(FP(L,1))*CMPLX(COS(FM(L)),SIN(FM(L)))
C     SUM UP CL,CD, AND CM
      CLCD = CLCD+TMP
      CM = CM+(XC(L)-.25)*REAL(TMP)-YC(L)*AIMAG(TMP)
C     WRITE PUNCH OUTPUT ON M4 IF XP=0 AND IZ.GT.80
      IF ((XP.GT.0.).OR.(IZ.LE.80)) GO TO 20
      Q = MACHN(L)*SQRT(C1/(1.+C2*MACHN(L)*MACHN(L)))
      V = Q*SIN(FM(L))
```

```
      U = Q*COS(FM(L))
      IF (XP.EQ.0) GO TO 15
      WRITE (M4,130) U,V,XC(L),YOLD(L),CP
      GO TO 20
   15 WRITE (M4,130) U,V,XC(L),YC(L),CP
   20 CONTINUE
C     CORRECT CL,CD FOR ANGLE OF ATTACK
      CLCD = -(D T*CHD)*CLCD*CMPLX(SIN(ALP),COS(ALP))
      CM = DT*CHD*CM
C     WRITE CD,CL,CM ONTO N4
      CDW = REAL(CLCD)
      CD = CDW+CDF
      CL2 = AIMAG(CLCD)
      IF (M4.EQ.N3) GO TO 85
      IF (CDF.EQ.0.) GO TO 70
      WRITE (N4,90) EM,CL2,CM,CDW,TX,CDF, CD
      GO TO 80
   70 WRITE (N4,90) EM,CL2,CM,CDW
C     CONSTRUCT MACH NUMBER DIAGRAM
      WRITE (N4,140)
   80 I = IMC(EM)
      I = IMACH(I)
C     USE PRINT WIDTH OF IZ FOR MACH NUMBER DIAGRAM
      MB = MM
      MC = MAX0(1,MB/IZ)
      MA = MC+MAX0(1,MB-IZ*MC)
C     WRITE OUT MACH NUMBERS AT INFINITY
      WRITE (N4,100) (I, L = MA,MB,MC)
C     DO MACH NUMBERS ONE LINE AT A TIME DOWN TO THE BODY
      J = NN-MC
   40 RSJ = R(J)*R(J)
      DO 50 L = MA,MB,MC
      U = (PHI(L+1,J)-PHI(L-1,J))*R(J)*DELTH-SI(L)
      V = (PHI(L,J+1)-PHI(L,J-1))*DELR*RSJ  -CO(L)
      Q = (U*U+V*V)/FP(L,J)
      I = IMC(MACH(Q))
      MN(L) = IMACH(I)
   50 CONTINUE
      WRITE (N4,100) (MN(L),L = MA,MB,MC)
      J = J-MC
      IF (J.GT.1) GO TO 40
C     DO THE LINE WHICH IS THE BODY
      DO 60 L = MA,MB,MC
      I = IMC(MACHN(L))
   60 MN(L) = IMACH(I)
      WRITE (N4,100) (MN(L),L = MA,MB,MC)
      IF (ITYP.GE.4) CALL GRAFIC(CD)
      RETURN
   85 RNX = .1*AINT(RN*1.E-5)
      WRITE (N4,150) EM,CL,TC,CM,RNX,CDF
      RETURN
   90 FORMAT (1H12X3HEM=F5.4,4X3HCL=F7.4,4X3HCM=F6.4,4X4HCDW=F7.5,4XA4
     1 ,F7.5,4X 3HCD=F7.5///)
  100 FORMAT (3X,130A1)
  120 FORMAT (3H M=,F4.3,5X,3HCL=,F5.3,5X,3HDY=,F4.3,6X,4HT/C=,
```

```
     1 F4.3,14X,2I5)
  130 FORMAT (4020)
  140 FORMAT (1H0//)
  150 FORMAT (1H0//7X3HEM=,F4.3,4X3HCL=,F6.4,4X4HT/C=,F4.3,4X3HCM=,
     1 F6.4,4X3HRN=,F4.1,4X4HCDF=,F6.4/)
      END


      SUBROUTINE GRAFIC(CD)
      COMPLEX ZP,ZQ,SFAC,SIG
      REAL MACHN
      COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
     1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CD(162)
     2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162),DSUM(162)
     3 ,ANGOLD(162),XOLD(162),YOLD(162),ARCOLD(162),DELOLD(162)
      COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP,TC,CHD,DPHI,CL,RCL,YR
     1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,DCN,DSN,RA4,EPSIL,QCRIT,C1,C2
     2 ,C4,C5,C6,C7,BET,BETA,FSYM,XSEP,SEPM,TTLE(4),M,N,MM,NN,NSP
     3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCY,NRN,NG,IDIM,N2,N3,N4,NT,IXX
     4 , NPTS,LL,I,LSEP,M4
      DIMENSION CPX(1),MACHN(1),T(6)
      EQUIVALENCE (CPX(1),PHIR(1)) ,(MACHN(1),A(1))
      DATA TOL/1.E-6/ , PF/-.4/ , SCF/5.0/,YOR/4.0/,SIZE/.14/,SCD/200./
C     MOVE THE ORIGIN TWO INCHES OVER AND TWO INCHES UP
      CALL PLOT(2.0,2.5,-3)
      YOR = AMAX1(3.5,.5*AINT(20.*EM-7.0))
C     PLOT CP CURVE AS A FUNCTION OF X
      CPF = 1./PF
      CCP = CPF*CPX(1)
      CALL PLOT(SCF*XC(1),YOR+CCP,3)
      DO 10 L = 2,MM
      CCP = AMIN1(8.5-YOR,CPF*CPX(L))
   10 CALL PLOT(SCF*XC(L),YOR+CCP,2)
C     DRAW AND LABEL THE CP-AXIS
      CALL CPAXIS(-.5,YOR,1.-1./PF,7.5-YOR,PF)
C     COMPUTE AND PLOT CRITICAL SPEED
      CALL SYMBOL (-.5,YOR+CPF*CPX(MM+1),2.*SIZE,15,0.,-1)
C     PLOT BODY
      CALL PLOT(SCF*XC(1),SCF*YC(1),3)
      DO 20 L = 2,MM
   20 CALL PLOT(SCF*XC(L),SCF*YC(L),2)
C     LABEL THE PLOT
      ALPX = RAD*ALP
      TXT=8HANALYSIS
      IF(FSYM.GE.6.) TXT=6HTHEORY
      XL=-.9
C     ****NON-ANSI - SEE VOLUME I, PAGE 209****
      IF(FSYM.GE.6.) GO TO 30
      ENCODE(60,191,T) TTLE,M,N,NCY
      GO TO 40
   30 LN=RN*1.E-6+.5
      ENCODE(60,190,T) TTLE,M,N,NCY,LN
   40 CALL SYMBOL(-1.14,-1.0,SIZE,T,0.,56)
```

```
C     ****NON-ANSI - SEE VOLUME I, PAGE 209****
      ENCODE (60,170,T) TXT,EM,ALPX,CL,CD
      CALL SYMBOL(XL,-1.35,SIZE,T,0.,60)
      CALL SYMBOL(XL-.10,-1.35+.5*SIZE,1.5*SIZE,15,0.,-1)
      CN=CO(1)
      SN=SI(1)
C     READ AND PLOT EXPERIMENTAL DATA IF XP IS NOT ZERO
      IF (XP.EQ.0.) GO TO 130
      REWIND M4
      READ (M4,140) NP
      IF (EOF(M4).NE.0) GO TO 130
      READ (M4,150) EMX,ALPX,CLX,CDX,SNX
      READ (M4,160) (CO(L),SI(L),L = 1,NP)
      TXT = 10HEXPERIMENT
      NC=59
      IF(SNX.GE.0.)GO TO 50
      TXT=6HDESIGN
      NC=3
C     ****NON-ANSI - SEE VOLUME I, PAGE 209****
   50 ENCODE (60,170,T) TXT,EMX,ALPX,CLX,CDX
      CALL SYMBOL(XL,-1.7,SIZE,T,0.,60)
      CALL SYMBOL(XL-.10,-1.7+.5*SIZE,SIZE,NC,0.,-1)
      DO 180 L = 1,NP
      CCP = YOR+CPF*SI(L)
      IF (CCP.GT.8.4) GO TO 180
      CALL SYMBOL(SCF*CO(L),CCP,.5*SIZE,NC,0.,-1)
  180 CONTINUE
  130 IF (ITYP.EQ.5) GO TO 122
C     PLOT THE SONIC LINE
      EX = 1.-EPSIL
C     SET SINES AND COSINES FOR USE IN FOURIER SERIES
      MX = M/2
      CO(1) = 1.
      SI(1) = 0.
      DO 60 L = 1,MX
      CO(L+1) = CO(L)*DCN-SI(L)*DSN
      CO(MM-L) = CO(L+1)
      SI(L+1) = CO(L)*DSN+SI(L)*DCN
   60 SI(MM-L) = -SI(L+1)
      DO 120 L = 2,M
C     LOOK FOR SONIC POINTS ON THE BODY
      IF (MACHN(L).LT.1.) GO TO 110
      IF (MACHN(L-1).GE.1.) GO TO 80
      IPEN = 3
C     COMPUTE Z AT SONIC LINE ON BODY
   70 R1 = (MACHN(L)-1.)/(MACHN(L)-MACHN(L-1))
      ZP = CMPLX(XC(L)+R1*(XC(L-1)-XC(L)),YC(L)+R1*(YC(L-1)-YC(L)))
      CALL PLOT(SCF*REAL(ZP),SCF*AIMAG(ZP),IPEN)
      IF (IPEN.EQ.2) GO TO 120
C     FIND THE SONIC LINE ALONG A RAY
   80 Q = MACHN(L)
      SX = SI(L)*CN+SN*CO(L)
      CX = CO(L)*CN-SN*SI(L)
      FAC = .5*DR
      ZQ = CMPLX(XC(L),YC(L))
```

```
      DO 90 J = 1,N
      ZP = SFAC
      RJ = R(J)
      QS = Q
      IF (J.EQ.1) GO TO 82
      U = (PHI(L+1,J)-PHI(L-1,J))*RJ*DELTH-SX
      V = (PHI(L,J+1)-PHI(L,J-1))*DELR*RJ*RJ-CX
      Q = (U*U+V*V)/FP(L,J)
      Q = SQRT(Q/(C1-C2*Q))
   82 SIG = CMPLX(RJ*CO(L),RJ*SI(L))
C     COMPUTE ((1-SIGMA)**(1-EPSIL))SIGMA
      SFAC = CEXP(EX*CLOG((1.,0.)-SIG))/SIG
C     SUM UP FOURIER SERIES TO OBTAIN CONJUGATE OF W
      S = -BB(1)
      DO 84 K = 1,NFC
      LT = MOD((L-1)*K,M)
      S = S+RJ*(AA(K+1)*SI(LT+1)-BB(K+1)*CO(LT+1))
      RJ = RJ*R(J)
      IF (RJ.LT.TOL) GO TO 86
   84 CONTINUE
C     COMPUTE THE ARGUMENT OF DZ/DR
   86 SFAC = -SFAC*CMPLX(COS(S),SIN(S))/CABS(SFAC)
C     MULTIPLY THE ARGUMENT BY THE MAGNITUDE TO OBTAIN DZ/DR
      SFAC = SFAC*(CHD*SQRT(FP(L,J)))/(R(J)*R(J))
C     PERFORM THE INTEGRATION
      ZQ = ZQ+FAC*SFAC
      FAC = DR
      IF (Q.LE.1.) GO TO 100
   90 CONTINUE
  100 ZQ = ZQ-.5*DR*SFAC
      ZP = ZQ-.5*DR*(SFAC+ZP)
      R1 = (Q-1.)/(Q-QS)
      ZP = ZQ+R1*(ZP-ZQ)
      CALL PLOT (SCF*REAL(ZP),AMAX1(-2.0,SCF*AIMAG(ZP)),2)
      GO TO 120
  110 IPEN = 2
      IF (MACHN(L-1).GE.1.) GO TO 70
  120 CONTINUE
C     POSITION PEN AT BEGINNING OF NEXT PAGE
  122 CALL PLOT (10.0,-2.5,-3)
      IF ((FSYM.NE.7.).OR.(ITYP.EQ.6)) RETURN
C     PLOT THE BOUNDARY LAYER DISPLACEMENT
      MX = INDEXR (0.,XC,M)
      CALL PLOT(2.,1.5,-3)
      CALL SYMBOL(1.36,-.65,SIZE,19HLOWER SURFACE  DELS ,0.,19)
      CALL CPAXIS (0.,0.,0.,4.,1./SCD)
C     PLOT LOWER SURFACE
      CALL PLOT (SCF*XC(1),SCD*DSUM(1),3)
      DO 132 L = 2,MX
  132 CALL PLOT (SCF*XC(L),SCD*DSUM(L),2)
      CALL PLOT(0.,4.5,-3)
      CALL SYMBOL(1.36,-.65,SIZE,19HUPPER SURFACE  DELS ,0.,19)
      CALL CPAXIS (0.,0.,0.,4.,1./SCD)
C     PLOT UPPER SURFACE
      CALL PLOT (SCF*XC(MX),SCD*DSUM(MX),3)
```

```
      DO 134 L = MX,M
  134 CALL PLOT (SCF*XC(L+1),SCD*DSUM(L+1),2)
      CALL PLOT(10.,-6.,-3)
      RETURN
  140 FORMAT (10X,I3)
  150 FORMAT (3F6.3,F7.5,E9.1)
  160 FORMAT (2F10.4)
  170 FORMAT (A12,4H  M=F4.3,3X4HALP=F5.2,3X3HCL=,F5.3,3X3HCD=,F5.4)
  190 FORMAT(4A4,3X4HM*N=I3,1H*I2,3X4HNCY=I4,4X2HR=I2,8H MILLION)
  191 FORMAT(4A4,3X4HM*N=I3,1H*I2,3X4HNCY=I4,4X12HNO VISCOSITY)
      END




      SUBROUTINE CPAXIS(XOR,YOR,BOT,TOP,SCF)
C     DRAWS AND LABELS THE CP AXIS
C     XOR,YOR IS THE LOCATION OF THE ORIGIN OF THE AXIS
C     BOT IS THE LENGTH OF THE AXIS BELOW THE ORIGIN
C     SCF IS A SCALE FACTOR USED FOR LABELING
C     SCF NEGATIVE FOR CP AXIS AND POSITIVE FOR DELS AXIS
      SIZE = .12-SIGN(.02,SCF)
C     DRAW THE VERTICAL AXIS
      CALL PLOT (XOR,YOR+TOP,3)
      CALL PLOT (XOR,YOR-BOT,2)
C     DRAW HATCH MARKS AND LABELS ONE INCH APART
      N = 1+INT(BOT)+INT(TOP)
      S = -AINT(BOT)*SCF +1.E-12
      XH = XOR-(3.*SIZE)/.7
      YH = YOR-AINT(BOT)
      DO 10 I = 1,N
      CALL SYMBOL (XOR,YH,SIZE,15,0.,-1)
C     ****NON-ANSI -  SEE VOLUME I, PAGE 209****
      IF (SCF.GT.0.) ENCODE (10,25,A) S
      IF (SCF.LE.0.) ENCODE (10,20,A) S
      S = S+SCF
      CALL SYMBOL (XH,YH,SIZE,A,0.,4)
   10 YH = YH+1.
      IF (SCF.GT.0.) GO TO 30
      CALL SYMBOL(XOR+.1,YOR+2.5,.14,1HC,0.,1)
      CALL SYMBOL(XOR+.25,YOR+2.38,.14,1HP,0.,1)
      RETURN
C     DRAW THE X-AXIS
   30 CALL PLOT (XOR,YOR-BOT,3)
      CALL PLOT (XOR+5.0,YOR-BOT,2)
      CALL SYMBOL (XOR+5.5,YOR-.07,.14,1HX,0.,1)
      YH = YOR-BOT-SIZE-SIZE
      DO 40 I = 1,5
      S = .2*FLOAT(I)
      ENCODE (10,20,A) S
      XH = YOR+FLOAT(I)-SIZE-SIZE
      CALL SYMBOL (XH,YH,SIZE,A,0.,4)
   40 CALL SYMBOL (XOR+FLOAT(I),YOR-BOT,SIZE,15,90.,-1)
      CALL SYMBOL (XOR+.25,YOR+3.0,.14,4HDELS,0.,4)
      RETURN
```

```
   25 FORMAT ( F4.3)
   20 FORMAT (F4.1)
      END




      SUBROUTINE GOPLOT (NRN)
C     INITIATE PLOT
C      ******************************************************************
C     THIS SUBROUTINE SHOULD BE REPLACED BY ANY ROUTINE WHICH INSTRUCTS
C     THE SYSTEM TO INITIATE A PLOT
C      ******************************************************************
      DIMENSION ID(6), LTAB(8), NAME(16)
      DATA MS,NU/77777777000000B,16/
      DATA NAME/10HGARABEDIAN,7H 109-01,10HDAVID KORN,7H 109-03,10H F. B
     1AUER ,7H 109-02,10HD. GOODMAN,7H 109-06,10HJ. DAHLIN ,7H 109-07,10
     2HDAVID KORN,7H 141-01,9HF. BAUER ,7H 143-07,10HA. JAMESON,7H 109-0
     34/
      DATA LTAB/34334434B,34334436B,34334435B,34334441B,34334442B,343734
     134B,34373642B,34334437B/
      ISHIFT(XXX,YYY) = SHIFT(XXX,YYY)
      N = MOD(IABS(NRN),1000)
      CALL READCP (ID,21B,1)
      ID(1) = ISHIFT(ID(2).AND.MS,-18)
      DO 10 L = 1,NU,2
      J = L/2+1
      IF (LTAB(J)-ID(1)) 10,20,10
   10 CONTINUE
      L = NU+1
   20 ENCODE (60,30,ID) NAME(L),NAME(L+1),N
      IF (NRN.GT.1000) GO TO 50
      CALL PLOTS (600,ID)
      RETURN
   50 CALL PLOTSBL (600,ID)
      RETURN
   30 FORMAT(A10,5H --- ,A7,11X,I3)
      END




      SUBROUTINE AIRFOL
C     READS IN DATA FOR AIRFOIL AND DETERMINES THE MAPPING
C     FUNCTION BY COMPUTING FOURIER COEFFICIENTS
C     IF ONLY X,Y COORDINATES ARE PRESCRIBED SLOPES ARE COMPUTED
      COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
     1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CO(162)
     2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162),DSUM(162)
     3 ,ANGOLD(162),XOLD(162),YOLD(162),ARCOLD(162),DELOLD(162)
      COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP,TC,CHD,DPHI,CL,RCL,YR
     1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,DCN,DSN,RA4,EPSIL,QCRIT,C1,C2
     2 ,C4,C5,C6,C7,BET,BETA,FSYM,XSEP,SEPM,TTLE(4),M,N,MM,NN,NSP
     3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCY,NRN,NG,IDIM,N2,N3,N4,NT,IXX
     4 , NPTS,LL,I,LSEP,M4
```

```
      DIMENSION XX(1),YY(1),U(1),V(1),W(1),SP(1),CIRC(1),TH(1),TT(1)
     1 ,DS(1),SS(1),CX(1),SX(1),QSR(1),TITLE(15),Z(1)
      EQUIVALENCE(XX(1),FP(1,3 )),(YY(1),FP(1,5)),(U(1),FP(1,1)),
     1 (V(1),FP(1,7)),(W(1),FP(1,9)),(SP(1),FP(1,11)),(CIRC(1),FP
     2 (1,13)),(TH(1),FP(1,15)),(TT(1),FP(1,17)),(DS(1),FP(1,19)),
     3 (SS(1),FP(1,21)),(CX(1),FP(1,23)),(SX(1),FP(1,25)),(QSR(1),
     4 FP(1,27)),(Z(1),FP(1,29))
      SQ(Q2) = Q2*Q2
      SMOOTH(Q1,Q2,Q3,Q4) = Q2+SQ(SQ(SQ(Q4)))*.25*(Q1-Q2-Q2+Q3)
      DIS(Q1) = (Q1-ERR)*((Q1-ERR)*(Q1-ERR)+CONST)
      DATA TOL,NT,ISYM,CONST,VAL/.4E-7,999,0,.2,4HRUN /
      DATA DXDS1,DXDS2,DYDS1,DYDS2/4*0./ , XT/-1./
C     NMP IS THE NUMBER OF POINTS IN CIRCLE PLANE FOR FOURIER SERIES
      LC = NFC
      NMP = 2*LC
      MC = NMP + 1
      PILC = PI/FLOAT(LC)
      IF (FSYM.GE.6.) GO TO 150
      WRITE (N4,470)
      REWIND N3
      READ (N3,410) TITLE
      IF (FSYM.GE.3.) GO TO 100
C     READ IN COORDINATES AS PRODUCED BY PROGRAMS D AND F
      EPSIL = 2.
      XX(1) = 0.
      NL = 2
      REWIND N3
      READ (N3,510) EM,CL,DY,TC,NRN
      IMC = MOD(INT(100.*EM+.5),100)
      ICL1 = MOD(INT(CL+.05),10)
      ICL2 = MOD(INT(10.*CL+.5),10)
      ITC1 = MOD(INT(10.*TC+.05),10)
      ITC2 = MOD(INT(100.*TC+.5),10)
      ENCODE (40,530,TTLE) IMC,ICL1,ICL2,ITC1,ITC2
      MODE = 0
      IF (NRN.LT.0) FSYM=2.
   10 READ (N3,500) U(2),V(2),XX(2),YY(2),FAC
      IF (XX(2).LT.1.) GO TO 20
C     SAVE TAIL POINT ON LOWER SURFACE
      U(1) = U(2)
      V(1) = V(2)
      XX(1) = XX(2)
      YY(1) = YY(2)
      GO TO 10
   20 DO 40 L = 3,999
      READ (N3,500) U(L),V(L),XX(L),YY(L),FAC
C     ****CHECK FOR END OF FILE****
      IF (EOF(N3).NE.0) GO TO 50
      IF (XX(L).EQ.1.) GO TO 70
      IF (XX(L).LT.XX(NL)) NL = L
   40 CONTINUE
C     AIRFOIL HAS BEEN EXTENDED IN PROGRAM D
   50 XT = 1.
   70 NT = L
      IF (XX(1).EQ.1.) GO TO 95
```

```
      IF (XT.LT.0.) XT = 1.+.6*DY
      NRN = IABS(NRN)
C     INTERPOLATE TO PUT THE TAIL AT X=XT
C     LOWER SURFACE INTERPOLATION
      I = 1
      L = 2
   80 R1 = (XT-XX(L+1))/(XX(L)-XX(L+1))
      R2 = 1.-R1
      YY(I) = R1*YY(L)+R2*YY(L+1)
      U(I) = R1*U(L)+R2*U(L+1)
      V(I) = R1*V(L)+R2*V(L+1)
      XX(I) = XT
      IF (I.EQ.NT) GO TO 150
C     UPPER SURFACE INTERPOLATION
      I = NT
      L = NT-2
      GO TO 80
C     READ IN AIRFOIL DATA FROM CARDS
  100 READ (N3,420) FNU,FNL,EPSIL
      READ (N3,470)
      NT = FNU+FNL-1.
      NL = FNL
      DO 110 I = NL,NT
  110 READ (N3,420) U(I),V(I),XX(I),YY(I)
      READ (N3,470)
      DO 120 I = 1,NL
      J = NL+1-I
  120 READ (N3,420) U(J),V(J),XX(J),YY(J)
      DO 130 J = 1,4
  130 TTLE(J) = TITLE(J)
      IF (FSYM.LE.4.) GO TO 150
      DO 140 L = 1,NT
      TH(L) = XX(L)/RAD
      XX(L) = U(L)
  140 YY(L) = V(L)
      GO TO 195
C     NO PERIOD IN THE STREAM FUNCTION
   95 EPSIL = 0.
C     DEFINE SLOPES SO THAT ARC LENGTHS CAN BE COMPUTED TO FIRST ORDER
  150 IF ((FSYM.EQ.1.).OR.(FSYM.EQ.3.)) GO TO 170
      DO 160 I = 1,NT
  160 TH(I) = 0.
      ISYM = 1
      GO TO 200
C     COMPUTE SLOPES FROM VELOCITIES
  170 TH(1) = ATAN(V(1)/U(1))
      QSR(1) = U(1)*U(1)+V(1)*V(1)
      DO 190 I = 2,NT
C     CHOOSE NEAREST BRANCH FOR THE ARCTANGENT
      DTH = ATAN((U(I-1)*V(I)-U(I)*V(I-1))/(U(I-1)*U(I)+V(I-1)*V(I)))
      TH(I) = TH(I-1)+DTH
  190 QSR(I) = U(I)*U(I)+V(I)*V(I)
  195 IF (EPSIL.GT.1.) EPSIL = (TH(1)-(PI+TH(NT)))/PI
      IF (FSYM.GT.5.) EPSIL = (TH(1)+TH(2)-TH(NT)-TH(NT-1))/TP-1.
C     COMPUTE ARC LENGTH TO FOURTH ORDER ACCURACY
```

```
  200 SP(1) = 0.
      DO 210 I = 2,NT
      DUM = AMAX1(.1E-20,.5*ABS(TH(I)-TH(I-1)))
      DX = XX(I)-XX(I-1)
      DY = YY(I)-YY(I-1)
  210 SP(I) = SP(I-1)+SQRT(DX*DX+DY*DY)*DUM/SIN(DUM)
      ARC = SP(NT)
      SN = 2./ARC
      SCALE = .25*ARC
      EE = .5*(1.-EPSIL)
      DO 220 L = 1,NT
  220 SS(L) = ACOS(1.-SN*SP(L))
      SS(NT) = PI
      IF (ISYM.NE.0) GO TO 350
      CALL SPLIF (NT,SS,TH,U,V,W,3,0.,3,0.)
      IF (FSYM.GT.5.) GO TO 232
      WRITE (N4,410) TITLE,VAL,NRN
      IF (N4.NE.N2) WRITE (N2,410) TITLE,VAL,NRN
C     PRINT OUT AIRFOIL DATA
      WRITE (N4,430)
      DO 230 L = 1,NT
      VAL = TH(L)*RAD
      SUM =-SN*U(L)/AMAX1(.1E-5,SIN(SS(L)))
      IF ((L.EQ.1).OR.(L.EQ.NT)) SUM = V(L)*SIGN(SN,FLOAT(L-2))
  230 WRITE (N4,480) XX(L),YY(L),SP(L),VAL,SUM,V(L),W(L)
      WRITE (N4,440)
C     MAKE INITIAL GUESS OF ARC LENGTH AS A FUNCTION OF CIRCLE ANGLE
  232 DX = (XX(NT)-XX(1))/TP
      DY = (YY(NT)-YY(1))/TP
      DO 240 I = 1,MC
      ANGL = FLOAT(I-1)*PILC
      CIRC(I) = ANGL
      CX(I) = COS(ANGL)
      SX(I) = SIN(ANGL)
      YY(I) = 1.
      IF (EE.NE.0.) YY(I) = (2.-2.*CX(I))**EE
      FAC = SIGN(1.+CX(I),FLOAT(LC-I))
  240 SP(I) = ACOS(.5*FAC)
      SP(MC) = PI
      CIRC(MC) = TP
      IF (FSYM.LT.6.) GO TO 244
      SCALE = ARC/ARCL(MM)
      DO 242 L = 1,MM
  242 Z(L) = FLOAT(L-1)*DT
      CALL SPLIF (MM,Z,ARCL,CO,SI,PHIR,3,0.,3,0.)
      CALL INTPL (NMP,CIRC,SP,Z,ARCL,CO,SI,PHIR)
  244 DO 245 L = 1,LC
      BB(L) = CX(2*L-1)
  245 AA(L) = -SX(2*L-1)
C     DO AT MOST 100 ITERATIONS TO FIND THE FOURIER COEFFICIENTS
      DO 320 K   = 1,100
      CALL INTPL(NMP,SP,TT,SS,TH,U,V,W)
      DO 250 I = 1,NMP
  250 TT(I) = TT(I)+.5*(CIRC(I)+EPSIL*(CIRC(I)-PI))
C     ENSURE CLOSURE
```

```
      DUM = 0.
      SUM = 0.
      FAC = 0.
      DO 260 L = 1,NMP
      DUM = DUM -TT(L)
      SUM = SUM-TT(L)*CX(L)
  260 FAC = FAC+TT(L)*SX(L)
      DUM = DUM/FLOAT(NMP)
      DA = 1.-EPSIL-(DX*SIN(DUM)+DY*COS(DUM))/SCALE-FAC/FLOAT(LC)
      DB = (DY*SIN(DUM)-DX*COS(DUM))/SCALE-SUM/FLOAT(LC)
      DO 270 L = 1,NMP
  270 TT(L) = TT(L)+DA*SX(L)-DB*CX(L)
C     FIND THE CONJUGATE FUNCTION DS
      CALL CONJ(NMP,TT,DS,XX,BB,AA)
      DO 290 I = 1,NMP
      SUM = DS(I)
  290 DS(I) = YY(I)*EXP(SUM)
      DS(MC) = DS(1)
      CALL SPLIF(MC,CIRC,DS,XX,XC,Z,-3,0.,3,0.)
      SCALE = ARC/Z(MC)
      ERR = 0.
      DO 310 I = 1,NMP
      VAL = ACOS(1.-2.*Z(I)/Z(MC))
      ERR = AMAX1(ERR,ABS(SP(I)-VAL))
  310 SP(I) = VAL
      IF (FSYM.LE.5.) WRITE (N4,490) ERR,DA,DB
      IF (ERR.LT.TOL) GO TO 330
  320 CONTINUE
      WRITE (N4,450)
  330 CALL FOUCF(NMP,TT,CX,BB,AA)
      AA(1) = ARC
      AA(2) = 1.-EPSIL-(DX*SIN(BB(1))+DY*COS(BB(1)))/SCALE
      BB(2) = (-DX*COS(BB(1))+DY*SIN(BB(1)))/SCALE
      IF (FSYM.GT.5.) GO TO 342
      WRITE (N4,460) EPSIL, NMP
      IF ((FSYM.NE.1.).AND.(FSYM.NE.3.)) GO TO 341
      DO 344 L = 1,MM
  344 Z(L) = FLOAT(L-1)*DT
      CALL SPLIF(MC,CIRC,SP,U,V,W,3,0.,3,0.)
      CALL INTPL(MM,Z,DS,CIRC,SP,U,V,W)
      CALL SPLIF (NT,SS,QSR,U,V,W,1,0.,1,0.)
      CALL INTPL(MM,DS,A,SS,QSR,U,V,W)
  341 IF (IZ.NE.120) GO TO 342
      WRITE (N4,540)
      DO 340 L = 1,NFC
  340 WRITE (N4,490) AA(L),BB(L)
  342 CALL MAP
      RETURN
  350 IF (FSYM.LE.5.) GO TO 355
      DXDS1 = (XX(2)-XX(1))/SS(2)
      DXDS2 = (XX(NT)-XX(NT-1))/(SS(NT)-SS(NT-1))
      DYDS1 = (YY(2)-YY(1))/SS(2)
      DYDS2 = (YY(NT)-YY(NT-1))/(SS(NT)-SS(NT-1))
  355 CALL SPLIF(NT,SS,XX,U,SP,W,1,DXDS1,1,DXDS2)
      CALL SPLIF(NT,SS,YY,V,TT,DS,1,DYDS1,1,DYDS2)
```

```
      IF (IS.LT.0) GO TO 397
      DC = PI/FLOAT(NMP)
      ERR = SS(NL)
      DUM = DIS(0.)
      FAC = PI/(DIS(PI)-DUM)
      DO 360 L = 1,MC
  360 CIRC(L) = FAC*(DIS(FLOAT(L-1)*DC)-DUM)
      CALL INTPL(NMP,CIRC,SX,SS,XX,U,SP,W)
      CALL INTPL(NMP,CIRC,CX,SS,YY,V,TT,DS)
      SX(MC) = XX(NT)
      CX(MC) = YY(NT)
      SFAC = 1./(XX(NT)-XX(NL))
      XXNL = XX(NL)
      DO 370 L = 1,MC
      CX(L) = SFAC*CX(L)
      SX(L) = SFAC*(SX(L)-XXNL)
      XX(L) = SX(L)
  370 YY(L) = CX(L)
      WRITE (N4,520) IS
      IF (N2.NE.N4) WRITE (N2,520) IS
      IF(IS.EQ.0) GO TO 395
C     DO  IS  SMOOTHING ITERATIONS
      DO 390 K = 1,IS
      DO 380 L = 2,NMP
      XX(L) = SMOOTH(SX(L-1),SX(L),SX(L+1),SX(L))
  380 YY(L) = SMOOTH(CX(L-1),CX(L),CX(L+1),SX(L))
      DO 390 L = 2,NMP
      SX(L) = XX(L)
  390 CX(L) = YY(L)
  395 NT = MC
      CALL SPLIF(NT,CIRC,XX,U,SP,W,1,0.,1,0.)
      CALL SPLIF(NT,CIRC,YY,V,TT,DS,1,0.,1,0.)
  397 ISYM = 0
      IF (FSYM.GT.5.) GO TO 170
      U(1) = SP(1)
      V(1) = TT(1)
      U(NT) = SP(NT)
      V(NT) = TT(NT)
      GO TO 170
  410 FORMAT (1X16A4,I4)
  420 FORMAT (5F10.7)
  430 FORMAT (35H0AIRFOIL COORDINATES AND CURVATURES/1H0,6X,1HX,14X1HY
     1 ,9X,10HARC LENGTH,7X3HANG,8X5HKAPPA,10X,2HKP,11X,3HKPP//)
  440 FORMAT (1H1,4X,3HERR,14X,2HDA,14X,2HDB//)
  450 FORMAT (32H FOURIER SERIES DID NOT CONVERGE)
  460 FORMAT (34H0MAPPING TO THE INSIDE OF A CIRCLE//3X11HDZ/DSIGMA =
     1 50H -(1/SIGMA**2)*(1-SIGMA)**(1-EPSIL)*(EXP(W(SIGMA))//3X,
     242HW(SIGMA) = SUM((A(N)-I*B(N))*SIGMA**(N-1))//3X,7HEPSIL =
     3 F5.3,20X,I4,25H POINTS AROUND THE CIRCLE )
  470 FORMAT (1H1)
  480 FORMAT (F12.6,2F14.6,F14.3,F14.4,2E14.3)
  490 FORMAT (3E15.6)
C     ****CHANGE (4020) TO (20A4) ON IBM 360****
  500 FORMAT (4020)
  510 FORMAT (3X,F4.3,8X,F5.3,8X,F4.3,10X,F4.3,14X,I5)
```

```
520 FORMAT (10HOTHERE ARE,I4,26H SMOOTHING ITERATIONS USED /)
530 FORMAT(4HAIRF,6X,3HOIL,7X,I2,1H-,I1,6X,I1,1H-,2I1)
540 FORMAT (//7X4HA(N),10X4HB(N)//)
    END




    SUBROUTINE MAP
C     SUM UP FOURIER SERIES TO OBTAIN MAPPING FUNCTION
      COMPLEX TT,TMP
      COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
     1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CO(162)
     2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162),DSUM(162)
     3 ,ANGOLD(162),XOLD(162),YOLD(162),ARCOLD(162),DELOLD(162)
      COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP,TC,CHD,DPHI,CL,RCL,YR
     1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,DCN,DSN,RA4,EPSIL,QCRIT,C1,C2
     2 ,C4,C5,C6,C7,BET,BETA,FSYM,XSEP,SEPM,TTLE(4),M,N,MM,NN,NSP
     3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCY,NRN,NG,IDIM,N2,N3,N4,NT,IXX
     4 , NPTS,LL,I,LSEP,M4
C     ****CHANGE TO 1.E-6 FOR SINGLE PRECISION IBM 360****
      DATA POW,TOL/-12.,10.E-12/
C     NOTE THAT THE SQUARE OF THE MAPPING MODULUS IS BEING COMPUTED
      MX = M/2
C     SET THE SINES AND COSINES
      CO(1) = 1.
      SI(1) = 0.
      DO 5 L = 1,MX
      CO(L+1) = CO(L)*DCN-SI(L)*DSN
      CO(MM-L) = CO(L+1)
      SI(L+1) = CO(L)*DSN+SI(L)*DCN
    5 SI(MM-L) = -SI(L+1)
C     SET MAPPING MODULUS FOR CUSP AT THE TAIL
      DO 10 J = 1,N
      FP (1,J) = 1.+R(J)*(R(J)-2.)
      DO 10 L = 1,MX
   10 FP(L+1,J) = 1.+R(J)*(R(J)-2.*CO(L+1))
      IF (EPSIL.EQ.0.) GO TO 30
C     ADJUST IF THERE IS AN ANGLE AT THE TAIL
      DO 20 J = 1,N
      FP(1,J) = FP(1,J)**(1.-EPSIL)
      DO 20 L = 1,MX
   20 FP(L+1,J) = FP(L+1,J)**(1.-EPSIL)
C     NOW COMPUTE CONTRIBUTION FROM FOURIER SERIES
   30 DO 50 J = 1,N
      NFCX = MIN0(NFC,1+INT(POW/ALOG10(R(J)-TOL)))
      RJ = 2.*R(J)
      K = NFCX
      S = AA(K+1)
   35 S = R(J)*S+AA(K)
      K = K-1
      IF (K.GT.1) GO TO 35
      FP(1,J) = FP(1,J)*EXP(S*RJ)
      DO 50 L = 1,MX
      K = NFCX
```

```
      LX = K*L
      LT = MOD(LX,M)
      S = AA(K+1)*CO(LT+1)
      Q = BB(K+1)*SI(LT+1)
   40 LX = LX-L
      LT = MOD(LX,M)
      S = R(J)*S+AA(K)*CO(LT+1)
      Q = R(J)*Q+BB(K)*SI(LT+1)
      K = K-1
      IF (K.GT.1) GO TO 40
      DUM = FP(L+1,J)
      FP(MM-L,J) = EXP(RJ*(S-Q))*DUM
   50 FP(L+1,J) = EXP(RJ*(S+Q))*DUM
      DO 65 L = 1,M
      S = PI-BB(1)
      DO 60 K = 1,NFC
      LT = MOD((L-1)*K,M)
   60 S = S+AA(K+1)*SI(LT+1)-BB(K+1)*CO(LT+1)
      ANG = FLOAT(L-1)*DT
      FP(L,NN) = 1.
   65 FM(L) = S-.5*(ANG+EPSIL*(ANG-PI))
      FM(MM) = FM(1)-(1.+EPSIL)*PI
      DO 70 J = 1,NN
      FP(MM,J) = FP(1,J)
   70 FP(MM+1,J) = FP(2,J)
C     COMPUTE ARC LENGTH AND BODY FROM THE MAPPING BY INTEGRATION
      XMIN = 0.
      YMIN = 0.
      YMAX = 0.
      S = -SQRT(FP(1,1))
      TMP = CMPLX(S*COS(FM(1)),S*SIN(FM(1)))
      DO 80 L = 1,MM
      Q = SQRT(FP(L,1))
      S = S+Q
      ARCL(L) = S
      S = S+Q
      TT = CMPLX(Q*COS(FM(L)),Q*SIN(FM(L)))
      TMP = TMP+TT
      XC(L) = REAL(TMP)
      YC(L) = AIMAG(TMP)
      XMIN = AMIN1(XMIN,REAL(TMP))
      YMIN = AMIN1(YMIN,AIMAG(TMP))
      YMAX = AMAX1(YMAX,AIMAG(TMP))
      TMP = TMP+TT
   80 CONTINUE
      CHD = -1./XMIN
      TC = (YMAX-YMIN)*CHD
      DO 90 L = 1,MM
      ARCL(L) = CHD*ARCL(L)
      XC(L) = 1.+CHD*XC(L)
   90 YC(L) = CHD*YC(L)
      CHD = CHD/(.5*DT)
      IF (ABS(FSYM).GT.5.) GO TO 100
      ANGO= -RAD*BB(1)
      WRITE (N4,120) TC,ANGO
```

```
      IF (N2.NE.N4) WRITE (N2,120) TC,ANGO
      IF (MODE.EQ.0) ALP = (1.+BET)*CL/(8.*PI*CHD)-BB(1)
  100 CALL COSI
      RETURN
  120 FORMAT (32H0THE THICKNESS TO CHORD RATIO IS ,F6.4//10H THE ANGLE
     1 17H OF ZERO LIFT IS ,F6.3,8H DEGREES)
      END




      SUBROUTINE SPLIF (N,S,F,FP,FPP,FPPP,KM,VM,KN,VN)
C     SPLINE FIT
C     GIVEN S AND F AT N CORRESPONDING POINTS,COMPUTE A CUBIC SPLINE
C     THROUGH THESE POINTS SATISFYING AN END CONDITION IMPOSED ON
C     EITHER END.  FP,FPP,FPPP WILL BE THE FIRST,SECOND AND THIRD
C     DERIVATIVE RESPECTIVELY AT EACH POINT ON THE SPLINE
C     KM IS THE DERIVATIVE IMPOSED AT THE START OF THE SPLINE
C     VM WILL BE THE VALUE OF THE DERIVATIVE THERE
C     KN IS THE DERIVATIVE IMPOSED AT THE END OF THE SPLINE
C     VN WILL BE THE VALUE OF THE DERIVATIVE THERE
C     KM,KN CAN TAKE VALUES 1,2, OR 3
C     S MUST BE MONOTONIC
      DIMENSION S(1), F(1), FP(1), FPP(1), FPPP(1)
      K = 1
      M = 1
      I = M
      J = M+K
      DS = S(J)-S(I)
      D = DS
      IF (DS.EQ.0.) CALL ABORT
      DF = (F(J)-F(I))/DS
      IF (IABS(KM)-2) 10,20,30
   10 U = .5
      V = 3.*(DF-VM)/DS
      GO TO 50
   20 U = 0.
      V = VM
      GO TO 50
   30 U = -1.
      V = -DS*VM
      GO TO 50
   40 I = J
      J = J+K
      DS = S(J)-S(I)
      IF (D*DS.LE.0.) CALL ABORT
      DF = (F(J)-F(I))/DS
      B = 1./(DS+DS+U)
      U = B*DS
      V = B*(6.*DF-V)
   50 FP(I) = U
      FPP(I) = V
      U = (2.-U)*DS
      V = 6.*DF+DS*V
      IF (J.NE.N) GO TO 40
```

```
      IF (KN-2) 60,70,80
   60 V = (6.*VN-V)/U
      GO TO 90
   70 V = VN
      GO TO 90
   80 V = (DS*VN+FPP(I))/(1.+FP(I))
   90 B = V
      D = DS
  100 DS = S(J)-S(I)
      U = FPP(I)-FP(I)*V
      FPPP(I) = (V-U)/DS
      FPP(I) = U
      FP(I) = (F(J)-F(I))/DS-DS*(V+U+U)/6.
      V = U
      J = I
      I = I-K
      IF (J.NE.M) GO TO 100
      FPPP(N) = FPPP(N-1)
      FPP(N) = B
      FP(N) = DF+D*(FPP(N-1)+B+B)/6.
      IF (KM.GT.0) RETURN
C     IF KM IS NEGATIVE COMPUTE THE INTEGRAL IN FPPP
      FPPP(J) = 0.
      V = FPP(J)
  105 I = J
      J = J+K
      DS = S(J)-S(I)
      U = FPP(J)
      FPPP(J) = FPPP(I)+.5*DS*(F(I)+F(J)-DS*DS*(U+V)/12.)
      V = U
      IF (J.NE.N) GO TO 105
      RETURN
      END



      SUBROUTINE INTPL (NX,SI,FI,S,F,FP,FPP,FPPP)
C     GIVEN S,F(S) AND THE FIRST THREE DERIVATIVES AT A SET OF POINTS
C     FIND FI(SI) AT THE NX VALUES OF SI BY EVALUATING THE TAYLOR SERIES
C     OBTAINED BY USING THE FIRST THREE DERIVATIVES
      DIMENSION SI(1), FI(1), S(1), F(1), FP(1), FPP(1), FPPP(1)
      DATA PT/.33333333333333/
      J = 0
      DO 30 I = 1,NX
      VAL = 0.
      SS = SI(I)
   10 J = J+1
      TT = S(J)-SS
      IF (FLOAT(J-1)*TT) 10,30,20
   20 J = J-1
      SS = SS-S(J)
      VAL = SS*(FP(J)+.5*SS*(FPP(J)+SS*PT*FPPP(J)))
   30 FI(I) = F(J)+VAL
      RETURN
      END
```

```
      SUBROUTINE CONJ (N,F,G,X,CN,SN)
C     CONJUGATION BY FAST FOURIER TRANSFORM
C     GIVEN THE REAL PART F OF AN ANALYTIC FUNCTION ON THE UNIT CIRCLE
C     THE IMAGINARY PART G IS CONSTRUCTED
      COMPLEX F,G,EIV,EIT
      DIMENSION   F(1),G(1),X(1),     CN(1),SN(1)
      DATA PI/3.14159265358979/
      L          = N/2
      DX = 1./FLOAT(L)
      EIV = CMPLX(COS(PI*DX),SIN(PI*DX))
      DO 2 I = 1,L
    2 G(I)       = F(I)
      CALL FFORM(L,G,X,CN,SN)
      G(1) = 0.
      I = 1
      DO 10 J = 1,L,2
      EIT = CMPLX(SN(I)*DX,CN(I)*DX)
      I = I+1
      G(J) = G(J)*EIT
   10 G(J+1) = G(J+1)*EIT*EIV
      DO 22 I=1,L
   22 SN(I)      = -SN(I)
      CALL FFORM(L,G,X,CN,SN)
      DO 32 I=1,L
   32 SN(I)      = -SN(I)
      EIV  = CMPLX(AIMAG(G(L)),REAL(G(1)))
      I = L
   40 G(I) = CMPLX(AIMAG(G(I-1)),REAL(G(I)))
      I = I-1
      IF (I.GT.1) GO TO 40
      G(1) = EIV
      RETURN
      END
```

```
      SUBROUTINE FOUCF(N,G,X,A,B)
C     FOURIER COEFFICIENTS BY FAST FOURIER TRANSFORM
      COMPLEX G,EIV,QP,X,GK
      DIMENSION        G(1),X(1),     A(1),B(1)
      DATA PI/3.14159265358979/
      L          = N/2
      V          = PI/L
      EIV = CMPLX(COS(V),SIN(V))
      ENI = 1./FLOAT(N)
      CALL FFORM(L,G,X,A,B)
      GK = 0.
      I = 1
      DO 5 J = 1,L,2
      X(J) = CMPLX(B(I),A(I))
      X(J+1) = X(J)*EIV
    5 I = I+1
      K = L
      DO 10 J = 1,L
```

```
      QP = GK-CONJG(G(J))
      GK = GK+CONJG(G(J))-QP*X(J)
      A(J) = -REAL(GK)*ENI
      B(J) = AIMAG(GK)*ENI
      GK = G(K)
   10 K = K-1
      A(L+1) = -B(1)
      B(1)      = 0.
      B(L+1)    = 0.
      RETURN
      END



      SUBROUTINE FFORM(N,F,X,CN,SN)
C     FAST FOURIER TRANSFORM
C     INPUT ARRAY F WITH REAL AND IMAGINARY PARTS IN ALTERNATE CELLS
C     REPLACED BY ITS FOURIER TRANSFORM
      COMPLEX F(1),X(1),W
      DIMENSION CN(1),SN(1)
      IF (N.LT.2) RETURN
      NS = 1
      NR = 2
      NQ = N
   11 DO 10 K = NR,N
      IF (MOD(NQ,K).EQ.0) GO TO 21
   10 CONTINUE
   21 ND=NQ/K
      NS = NS*K
      NR = K
      IQ = 0
      ID = 0
      DO 22 I = 1,NS
      DO 24 J = 1,ND
      L = MOD(IQ+J,N)
      W = F(L)
      M = 0
      DO 26 K = 2,NR
      L = L+ND
      M = MOD(M+ID,N)
   26 W = W+F(L)*CMPLX(CN(M+1),SN(M+1))
   24 X(ID+J) = W
      ID = ID+ND
   22 IQ = IQ+NQ
      NQ = ND
      IF (ND.GT.1) GO TO 61
      DO 32 K = 1,N
   32 F(K) = X(K)
      RETURN
   61 DO 60 K = NR,N
      IF (MOD(NQ,K).EQ.0) GO TO 71
   60 CONTINUE
   71 ND=NQ/K
      NS = NS*K
```

```
      NR = K
      IQ = 0
      ID = 0
      DO 72 I = 1,NS
      DO 74 J = 1,ND
      L = MOD(IQ+J,N)
      W = X(L)
      M = 0
      DO 76 K = 2,NR
      L = L+ND
      M = MOD(M+ID,N)
   76 W = W+X(L)*CMPLX(CN(M+1),SN(M+1))
   74 F(ID+J) = W
      ID = ID+ND
   72 IQ = IQ+NQ
      NQ = ND
      IF (ND.GT.1) GO TO 11
      RETURN
      END
```

```
      FUNCTION INDEXR(X,ARRAY,N)
      DIMENSION ARRAY(1)
      S = ABS(X-ARRAY(N))
      DO 10 L = 1,N
      IF (ABS(X-ARRAY(L)).GT.S) GO TO 10
      INDEXR = L
      S = ABS(X-ARRAY(L))
   10 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE GTURB(DELMAX,DELBP,CPO,BCP,SL,ROEL,RBCP)
      COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
     1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CO(162)
     2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162),DSUM(162)
     3 ,ANGOLD(162),XOLD(162),YOLD(162),ARCOLD(162),DELOLD(162)
      COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP.TC,CHD,DPHI,CL,RCL,YR
     1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,DCN,DSN,RA4,EPSIL,QCRIT,C1,C2
     2 ,C4,C5,C6,C7,BET,BETA,FSYM,XSEP,SEPM.TTLE(4),M,N,MM,NN,NSP
     3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCY,NRN,NG,IDIM,N2,N3,N4,NT,IXX
     4 , NPTS,LL,I,LSEP,M4
      REAL MACH,MACHN,NEW,MACHS
      DIMENSION HP(162),SEPP(162),CPP(162),THETAP(162),DELP(162)
     1  ,DELX(1),TD(1)
      DIMENSION H(1),THETA(1),DELS(1),XX(1),YY(1),MACHN(1)
     1 ,SEPR(1),CPX(1),DSDT(1),S(1),MACHS(1),ANGNEW(1)
      EQUIVALENCE (MACHN(1),A(1))   ,(H(1),FP(1,6)),(THETA(1),FP(1,8))
     1          ,(XX(1),FP(1,3 )),(YY(1),FP(1,5)),(DELS(1),FP(1,10))
     2,(ANGNEW(1),FP(1,24)),(SEPR(1),FP(1,14)),(CPX(1),PHIR(1))
     3 ,(S(1),FP(1,16)),(MACHS(1),FP(1,28)).(DSDT(1),FP(1,30))
     4 ,(DELX(1),FP(1,12)),(TD(1),FP(1,20))
```

```
      CP(Q) = C5*((C4/(1.+C2*Q*Q))**C7-1.)
      QSX(Q) = (C4-(1.+Q/C5)**(1./C7))/C6
      MACH(Q) = SQRT (Q/(C1-C2*Q))
      DATA ISW/0/,CDF/0./,XPLT/.5/,XFAC/100./
      DO 10 J = 1,NN
      PHI(MM,J) = PHI(1,J)+DPHI
   10 PHI(MM+1,J) = PHI(2,J)+DPHI
      IF (ISW.EQ.0) CALL GOPLOT(NRN)
C     COMPUTE AND STORE CP CRITICAL
      CPX(MM+1) = CP(1.)
C     ISX SET TO 1 FOR FSYM=1. AND FSYM=3 IF FLOW HAS NOT BEEN COMPUTED
      ISX = (NCY+1)*(ITYP-3)*ABS(FSYM+10.)+.2
      IF (ISX.NE.1) GO TO 30
      M4 = N3
      FSYM = 0.
      XC(MM) = 1.
      ALP = 0.
      XSEP = AMAX1(0.,XSEP-1.)
      QS = A(MM)
      DO 20 L = 1,MM
      XOLD(L) = XC(L)
      YOLD(L) = YC(L)
      MACHN(L) = MACH(A(L))
   20 CPX(L) = CP(MACHN(L))
      IF ((ABS(YC(MM)-YC(1)).LE.1.E-5).AND.(IABS(NRN).GT.999)) GO TO 50
      GO TO 110
   30 DO 40 L = 2,M
      U = (PHI(L+1,1)-PHI(L-1,1))*DELTH-SI(L)
      QS = (U*U)/FP(L,1)
      MACHN(L) = MACH(QS)
   40 CPX(L) = CP(MACHN(L))
      MACHN(MM) = .5*(MACHN(2)+MACHN(M))
      MACHN(1) = MACHN(MM)
      CPX(1) = CP(MACHN(1))
      CPX(MM) = CPX(1)
      QS=QSX(CPX(MM))
      IF (FSYM.EQ.6.) GO TO 60
      IF ((FSYM.LE.5.).OR.(ITYP.LE.2)) GO TO 50
C     ADVANCE PLOTTER PAPER TO THE NEXT BLANK PAGE
      IF(XPLT.GT..5) CALL PLOT(12.0*FLOAT(INT((20.2+XPLT)/12.)),0.,-3)
      XPLT = .5
   50 CALL GETCP(CDF)
      CALL GOPRIN (HP,THETAP,SEPP,CPP,DELP,XTRANS)
      IF (ISX.EQ.1) CALL EXIT
      ISW = 1
      RETURN
   60 DO 70 L = 1,MM
   70 CPP(L) = CPX(L)
      IF((ISW.EQ.0).OR.(FSYM.NE.6.)) GO TO 90
C     FIND THE BASE PRESSURE
      DELBP = 10.
      CP0 = CP(MACHN(IXX-1))
      DO 80 L = IXX,M
      CPN = CP(MACHN(L))
      DELBP = AMIN1(DELBP,CPN-CP0)
```

```
   80 CPO = CPN
      BCP = BCP+RBCP*DELBP
   90 ISW = 1
      PCH = ABS(PCH)
      IF (LSEP.GE.MM) GO TO 110
C     MODIFY THE MACH DISTRIBUTION
      CPO = CP(MACHN(LSEP))
      SEPX = XC(LSEP)
      SL = (BCP-CPO)/(XC(MM)-SEPX)
      DO 100 L = LSEP,MM
      CPP(L) = CPO+SL*(XC(L)-SEPX)
  100 MACHN(L) = MACH(QSX(CPP(L)))
  110 KQMIN = 1
      KQMAX = 1
      QMIN = MACHN(1)
      QMAX = QMIN
      DARC = TP/FLOAT(NPTS-1)
      DO 115 L = 1,NPTS
  115 H(L) = FLOAT(L-1)*DARC
      H(NPTS) = TP
      DO 116 L = 1,M
  116 YY(L) = FLOAT(L-1)*DT
      YY(MM) = TP
      CALL SPLIF (MM,YY,ARCL,DSDT,CO,TD,3,0.,3,0.)
      CALL INTPL (NPTS,H,S,YY,ARCL,DSDT,CO,TD)
      S(NPTS) = ARCL(MM)
      CALL SPLIF (MM,ARCL,MACHN,DSDT,CO,TD,3,0.,3,0.)
      CALL INTPL(NPTS,S    ,MACHS,ARCL,MACHN,DSDT,CO,TD)
      CALL SPLIF (MM,ARCL,XC,DSDT,CO,TD,3,0.,3,0.)
      CALL INTPL (NPTS,S,XX,ARCL,XC,DSDT,CO,TD)
      DO 120 L = 1,NPTS
      IF (MACHS(L).GT.QMAX) KQMAX = L
      IF (MACHS(L).LT.QMIN) KQMIN = L
      QMIN = AMIN1(MACHS(L),QMIN)
      QMAX = AMAX1(MACHS(L),QMAX)
      SEPR(L) = 0.
      H(L) = 0.
      DELS(L) = 0.
  120 THETA(L) = 0.
      IF (PCH.LT.0.) GO TO 140
      KQMAX = KQMIN+INDEXR(PCH,XX(KQMIN+1),NPTS-KQMIN)
      IF (KQMAX.GE.NPTS) CALL ABORT
  140 CALL NASHMC (KQMAX,NPTS)
      XTRANS = PCH
      IF (PCH.LT.0) XTRANS = XX(KQMAX)
      KQBOT = INDEXR(XTRANS,XX,KQMIN)
      IF (KQBOT.LE.1) CALL ABORT
      CALL NASHMC (KQBOT,1)
      FAC=S(4)/(S(4)-S(2))
      THETA(1)=FAC*THETA(2)+(1.-FAC)*THETA(4)
      H(1)=FAC*H(2)+(1.-FAC)*H(4)
      DELS(1)=H(1)*THETA(1)
C     COMPUTE THE SKIN FRICTION DRAG
      Q = SQRT(QS)
      RT = (C1-C2*QS)/(C1-C2)
```

```
      HBT = (H(NPTS)+1.)*(1.-C2*QS/C1)-1.
      HBB = (H(1)+1.)*(1.-C2*QS/C1)-1.
      COF = 2.*THETA(NPTS)*Q**(.5*(  HBT  +5.))*RT**3
      COF = COF+2.*THETA(1)*Q**(.5*( HBB+5.))*RT**3
      IF (ISX.EQ.1) GO TO 200
C     MAKE DISPLACEMENT MONOTONE INCREASING ON THE UPPER SURFACE
      DO 170 L = KQMAX,NPTS
      IF (DELS(L+1).LT.DELS(L)) DELS(L+1) = DELS(L)
  170 CONTINUE
C     LOWER SURFACE - FIND WHERE DELS STARTS DECREASING
C     TREAT THE LOWER SURFACE LIKE THE UPPER SURFACE IF XSEP.LT.0
      XPC = .60
      IF (XSEP.LT.0.) XPC = 2.
      J = KQBOT
  180 J = J-1
      IF (DELS(J-1).LT.DELS(J)) GO TO 185
      IF (J.GE.2) GO TO 180
      GO TO 200
  185 IF (XX(J).GT.XPC) GO TO 190
      DELS(J-1) = DELS(J)
      GO TO 180
C     DISPLACEMENT MUST STAY MONOTONE DECREASING
  190 J = J-1
      IF (DELS(J-1).GT.DELS(J)) DELS(J-1) = DELS(J)
      IF (J.GE.2) GO TO 190
C     SMOOTH DELS  IS  TIMES
  200 IF (IS.LE.0) GO TO 220
      DO 210  I = 1,IS
      OLD = DELS(1)
      DO 210 L = 3,NPTS
      NEW = DELS(L-1)
      DELS(L-1) = .25*(OLD+NEW+NEW+DELS(L))
  210 OLD = NEW
  220 XPLT = XPLT+.5
      FAC=(S(NPTS-1)-S(NPTS))/(S(NPTS-1)-S(NPTS-2))
      DELS(NPTS)=FAC*DELS(NPTS-2)+(1.-FAC)*DELS(NPTS-1)
      IF (ISX.EQ.1) GO TO 260
      YFAC = 10./S(NPTS)
      DH = (H(KQMAX+1)-H(KQBOT-1))/ FLOAT(2+KQMAX-KQMIN)
      FAC = ARCOLD(NT)/S(NPTS)
      IF(XPLT.LT.1.2) CALL SYMBOL(.35,8.74,.14,55HDISPLACEMENT THICKNESS
     1 AT EACH BOUNDARY LAYER ITERATION,270.,55)
      CALL PLOT (XPLT+XFAC*DELS(1),10.5,3)
      DO 230 L = 1,NPTS
      CALL PLOT(XPLT+XFAC*DELS(L),10.5-YFAC*S(L),2)
      IF ((L.GE.KQBOT).AND.(L.LE.KQMAX)) H(L) = H(L-1)+DH
  230 YY(L) = S(L)*FAC
      YY(NPTS) = ARCOLD(NT)
C     DELX WILL BE BOUNDARY LAYER DISPLACEMENT AT NT POINTS
      CALL SPLIF(NPTS,YY,DELS,DSDT,CO,TD,3,0.,3,0.)
      CALL INTPL(NT,ARCOLD,DELX,YY,DELS,DSDT,CO,TD)
C     THE FOLLOWING ARE BEING COMPUTED FOR FUTURE PRINT OUT
      CALL SPLIF(NPTS,S,DELS,DSDT,CO,TD,3,0.,3,0.)
      CALL INTPL(MM,ARCL,DELP,S,DELS,DSDT,CO,TD)
      CALL SPLIF (NPTS,S,H,DSDT,CO,TD,3,0.,3,0.)
```

```
      CALL INTPL(MM,ARCL,HP,S,H,DSDT,CO,TD)
      CALL SPLIF(NPTS,S,THETA,DSDT,CO,TD,3,0.,3,0.)
      CALL INTPL (MM,ARCL,THETAP,S,THETA,DSDT,CO,TD)
      CALL SPLIF(NPTS,S, SEPR,DSDT,CO,TD,3,0.,3,0.)
      CALL INTPL(MM,ARCL,SEPP,S,SEPR,DSDT,CO,TD)
C     GET THE SLOPES FOR THE OUTER AIRFOIL AT CORRESPONDING POINTS
      DO 240 L = 1,MM
      DDEL = RDEL*(DELP(L)-DSUM(L))
      DELP(L) = DDEL
      DSUM(L) = DSUM(L)+DDEL
  240 S(L) = FAC*ARCL(L)
      S(MM) = ARCOLD(NT)
      CALL SPLIF(MM,S,FM,DSDT,CO,TD,3,0.,3,0.)
      CALL INTPL(NT,ARCOLD,ANGNEW,S,FM,DSDT,CO,TD)
      DELMAX = 0.
      DO 250 L = 1,NT
      DDEL = DELX(L)-DELOLD(L)
      DELMAX = AMAX1(DELMAX,ABS(DDEL))
      DY = DELOLD(L)+RDEL*DDEL
      ANG = .5*(ANGOLD(L)+ANGNEW(L))
      XX(L)=XOLD(L)
      YY(L)=YOLD(L)+DY/COS(ANG)
  250 DELOLD(L) = DY
      ISS = IS
      IS = -1
      IF (ITYP.EQ.99) CALL GOPRIN (HP,THETAP,SEPP,CPP,DELP,XTRANS)
      CALL AIRFOL
      IS = ISS
      FSYM = 7.
      RETURN
  260 DO 270 L = 1,MM
      ARCOLD(L) = ARCL(L)
      CPP(L) = CPX(L)
  270 ANGOLD(L) = FM(L)
      CALL SPLIF(NPTS,S,DELS,DSDT,CO,TD,3,0.,3,0.)
      CALL INTPL(MM,ARCL,DSUM,S,DELS,DSDT,CO,TD)
      CALL SPLIF(NPTS,S,SEPR,DSDT,CO,TD,3,0.,3,0.)
      CALL INTPL (MM,ARCL,SEPP,S,SEPR,DSDT,CO,TD)
      CALL SPLIF (NPTS,S,THETA,DSDT,CO,TD,3.0.,3,0.)
      CALL INTPL (MM,ARCL,THETAP,S,THETA,DSDT,CO,TD)
      CALL GOPRIN (HP,THETAP,SEPP,CPP,DELP,XTRANS)
      NT = MM
      CALL GETCP(CDF)
      IF (JK.LE.-1 ) CALL PLOT (0.,0.,999)
      CALL EXIT
      END




      SUBROUTINE GOPRIN(H,THETA,SEP,CPP,DEL,XTR)
      REAL MACHN
      COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
     1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CO(162)
     2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162),DSUM(162)
```

```
     3 ,ANGOLD(162),XOLD(162),YOLD(162),ARCOLD(162),DELOLD(162)
      COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP,TC,CHD,DPHI,CL,RCL,YR
     1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,DCN,DSN,RA4,EPSIL,QCRIT,C1,C2
     2 ,C4,C5,C6,C7,BET,BETA,FSYM,XSEP,SEPM,TTLE(4),M,N,MM,NN,NSP
     3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCY,NRN,NG,IDIM,N2,N3,N4,NT,IXX
     4 , NPTS,LL,I,LSEP,M4
      DIMENSION DSDT(1),FPP(1),FPPP(1),H(1),SEP(1),THETA(1),CPP(1)
     1 ,MACHN(1),CP(1),DEL(1),BL(4)
      EQUIVALENCE (FPP(1),CO(1)),(FPPP(1),SI(1)),(DSDT(1),FP(1,31))
      EQUIVALENCE (MACHN(1),A(1)),(CP(1),PHIR(1))
      DATA ION,IOFF,Z,SEPMAX/1,0,0.,.004/
      SN = -2./ARCL(MM)
      QMIN=MACHN(1)
      DO 10 L = 1,M
      QMIN=AMIN1(MACHN(L),QMIN)
   10 ARCL(L) = ACOS(1.+SN*ARCL(L))
      ARCL(MM) = PI
      CALL SPLIF(MM,ARCL,FM,DSDT,FPP,FPPP,1.0,,1,0.)
      DSDT(1) = FPP(1)*1.E-5
      DSDT(MM) = -FPP(MM)*1.E-5
      DO 20 L = 1,MM
      FPP(L) = RAD*FM(L)-180.
   20 FPPP(L)= SN*DSDT(L)/AMAX1(1.E-5,SIN(ARCL(L)))
      IF (FSYM.GT.5.) GO TO 120
      IF (FSYM.EQ.0.) GO TO 60
      WRITE (N4,310)
   25 IF (FSYM.EQ.0) WRITE (N4,320) TTLE
      WRITE (N4,360) IOFF
      DO 30 L = 1,MM
      IF (MOD(L+1,55).EQ.0) WRITE (N4,360) ION
   30 WRITE (N4,260) L,XC(L),YC(L),FPP(L),FPPP(L),MACHN(L),CP(L)
C     RESTORE QUANTITIES TO VALUES THEY HAD UPON ENTERING THIS ROUTINE
   40 DO 50 L = 1,MM
      ARCL(L) = (COS(ARCL(L))-1.)/SN
   50 FP(L,NN) = 1.
      CALL COSI
      RETURN
   60 RNX = .1*AINT(RN*1.E-5)
      IF ((ABS(YC(MM)-YC(1)).LE.1.E-5).AND.(IABS(NRN).GT.999)) GO TO 25
      WRITE (N4,390) TTLE,RNX
      WRITE (N4,330) IOFF
      IF ( JK.GE.0 ) GO TO 80
      CALL PLOT (2.,0.,-3)
      ENCODE (30,370,TTLE) EM,CL,TC
      CALL SYMBOL (1.2,.7,.14,TTLE,0.,30)
      ENCODE (20,380,TTLE) RNX
      CALL SYMBOL (1.5,1.0,.14,TTLE,0.,20)
      CALL PLOT(50.*XC(1),5.0+50.*YC(1),3)
      DO  70 L = 2,MM
   70 CALL PLOT (50.*XC(L),5.0+50.*YC(L),2)
      IPEN = 3
   80 DO 100 L = 1,MM
      YS = YOLD(L)-DSUM(L)/COS(ANGOLD(L))
      YC(L) = YS
      IF (JK.LE.-1)  CALL PLOT(50.*XOLD(L),5.0+50.*YS,IPEN)
```

```
      IPEN = 2
      IF (MOD(L+3,55).EQ.0) WRITE (N4,330) ION
      IF (XOLD(L).GT.XTR) GO TO 90
      TRANS = 1H
      IF (MACHN(L).EQ.QMIN) TRANS = 10HSTAGNATION
      IF ((XOLD(L+1).GT.XTR).OR.(XOLD(L-1).GT.XTR)) GO TO 85
      IF ((XOLD(L+2).GT.XTR).OR.(XOLD(L-2).GT.XTR))TRANS= 10HTRANSITION
   85 WRITE (N4,340) XOLD(L),YOLD(L),YS,FPP(L),FPPP(L),CPP(L),TRANS
      GO TO 100
   90 WRITE (N4,350) XOLD(L),YOLD(L),YS,FPP(L),FPPP(L),CPP(L),THETA(L)
     1 ,SEP(L)
  100 CONTINUE
      IF (XP.EQ.0.) NRN = -IABS(NRN)
      XP = -ABS(XP)
      RETURN
  120 WRITE (N4,310)
      WRITE (N4,300) IOFF
      I = 1
      YSEP = ABS(XSEP)
      IF (XSEP.GT.0.) YSEP = 2.
      DO 150 L = 1,MM
      IF (MOD(L,55).EQ.0) WRITE (N4,300) ION
      IF (XC(L).GT.XTR) GO TO 130
      TRANS = 1H
      IF (MACHN(L).EQ.QMIN) TRANS = 10HSTAGNATION
      IF ((XC(L+1).GT.XTR).OR.(XC(L-1).GT.XTR)) GO TO 125
      I = -1
      YSEP = ABS(XSEP)
      IF ((XC(L+2).GT.XTR).OR.(XC(L-2).GT.XTR)) TRANS = 10HTRANSITION
  125 WRITE (N4,290)         L,XC(L),YC(L),FPP(L),FPPP(L),MACHN(L),
     1 CP(L),CPP(L),Z,Z,TRANS,L
      GO TO 150
  130 BL(1) = 1H
      BL(2) = 1H
      BL(3) = 1H
      BL(4)= 1H
      IF (L.EQ.LSEP)   BL(1) = 2HLS
      IF((SEP(L).GT.SEPMAX).AND.(SEP(L+I).LT.SEPMAX)) BL(2)= 2HCS
      IF (L.EQ.IXX) BL(3)= 2HLM
      IF((XC(L).GE.YSEP).AND.(XC(L+I).LT.YSEP)) BL(4) = 2HLP
      WRITE (N4,280) BL     ,L,XC(L),YC(L),FPP(L),FPPP(L),MACHN(L),
     1 CP(L),CPP(L),THETA(L),DSUM(L),SEP(L),H(L),DEL(L),L
  150 CONTINUE
      GO TO 40
  260 FORMAT(I14,2F9.5,2F8.2,2F9.4)
  280 FORMAT(3X,  4A2,I5,2F9.5,F9.2,F8.2,F8.4,2F9.4,F9.5,F9.5,F9.5,F7.2,
     1E9.2,I5)
  290 FORMAT (I16,2F9.5,F9.2,F8.2,F8.4,2F9.4,2F9.5,8X,A10,7X,I5)
  300 FORMAT(I1,14X1HL5X2HXS,7X,2HYS,7X,3HANG,4X,5HKAPPA,4X,4HMACH6X2HCP
     1 ,6X3HCP1,4X5HTHETA,5X4HDELS,6X3HSEP,6X1HH,6X2HDD,6X1HL/)
  310 FORMAT(1H1,15X,40HLOWER SURFACE TAIL TO UPPER SURFACE TAIL )
  320 FORMAT(1H1/  17X26HLISTING OF COORDINATES FOR,2X,4A4)
  330 FORMAT(I1 /11X1HX,8X,1HY,7X,2HYS,6X,3HANG,4X,5HKAPPA,6X,2HCP,5X,
     1 5HTHETA,5X,3HSEP/)
  340 FORMAT (F14.5,2F9.5,F8.2,F8.2,F9.4,4X,A10)
```

```
350 FORMAT (F14.5,2F9.5,F8.2,F8.2,F9.4,2F9.5)
360 FORMAT (I1/12X1HL,6X,1HX,8X,1HY,6X,3HANG,4X5HKAPPA4X4HMACH6X2HCP/)
370 FORMAT ( 2HM=,F4.3,4X,3HCL=,F5.3,4X,4HT/C=,F4.3)
380 FORMAT (4H RN=,F4.1,9H  MILLION )
390 FORMAT(1H1/   9X26HLISTING OF COORDINATES FOR,2X,4A4 ,4X,3HRN=,
   1 F4.1,8H MILLION )
    END




    SUBROUTINE NASHMC (K1,K2)
C   COMPUTE THE BOUNORY LAYER FROM POINT K1 TO K2
C   K3 WILL BE THE SEPARATION POINT
    COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
   1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CO(162)
   2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162),DSUM(162)
   3 ,ANGOLD(162),XOLD(162),YOLD(162),ARCOLD(162),DELOLD(162)
    COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP,TC,CHD,DPHI,CL,RCL,YR
   1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,DCN,DSN,RA4,EPSIL,QCRIT,C1,C2
   2 ,C4,C5,C6,C7,BET,BETA,FSYM,XSEP,SEPM,TTLE(4),M,N,MM,NN,NSP
   3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCT,NRN,NG,IDIM,N2,N3,N4,NT,IXX
   4 , NPTS,LL,I,LSEP,M4
    DIMENSION MACHS(1),H(1),THETA(1),SEPR(1),S(1),DELS(1),XX(1)
    EQUIVALENCE(MACHS(1),FP(1,28)),(H(1),FP(1,6)),(THETA(1),FP(1,8))
    EQUIVALENCE (SEPR(1),FP(1,14)),(DELS(1),FP(1,10)),(S(1),FP(1,16))
    EQUIVALENCE (XX(1),FP(1,3))
    REAL MH,MHSQ,NU,MACHS
    DATA TR,RTHO,TE1,TE2,SEPMAX,PIMIN,PIMAX /.3424,320.,5.E-3,5.E-5,
   1 .004,-1.5,1.E4/
    GAM1 = .5/C2
    CSIINF = C4
    INC = ISIGN(1,K2-K1)
    YSEP = ABS(XSEP)
    IF ((XSEP.GT.0.).AND.(INC.LT.0)) YSEP = 1.
    SEPMAX = SEPM
    GE = 6.5
    L = K1
    DS = ABS(S(L)-S(L-INC))
 10 LP = L+INC
    MH = .5*(MACHS(L)+MACHS(LP))
    MHSQ = MH*MH
    CSIH = 1.+C2*MHSQ
    DSOLD = DS
    DS = ABS(S(LP)-S(L))
    DQDS = (MACHS(LP)-MACHS(L))/(DS*MH*CSIH)
    T = CSIINF/CSIH
    RHOH = T**GAM1
    NU = T*(1.+TR)/(RHOH*(T+TR))
    RTH = RN*MH/(EM*NU)
    IF (L.NE.K1) GO TO 30
    THETAH= RTHO/RTH
    THT = THETAH
 30 FC = 1.0+.066*MHSQ-.008*MH*MHSQ
    FR = 1.-.134*MHSQ+.027*MHSQ*MH
```

```
C     DO AT MOST 500 ITERATIONS
      DO 140 J = 1,499
      RTAU= 1./(FC*(2.4711*ALOG(FR*RTH*THETAH)+4.75)+1.5*GE+1724./
     1  (GE*GE+200.)-16.87)
      TAU = RTAU*RTAU
      HB = 1./(1.-GE*RTAU)
      HH   = (HB+1.)*(1.+.178*MHSQ)-1.
      SEP = -THETAH*DQDS
      IF (SEP.LT.SEPMAX) GO TO 50
      IF (XX(L).LT.YSEP) SEP = SEPMAX
   50 PIE = HH*SEP/TAU
      PIE = AMAX1(PIMIN,AMIN1(PIMAX,PIE))
      G = 6.1*SQRT(PIE+1.81)-1.7
      T2 = ABS(G-GE)/GE
      GE = G
      DT2 = DT1
      DT1 = (HH+2.-MHSQ)*SEP+TAU
      IF (J.EQ.1) GO TO 110
      TI = ABS((DT1-DT2)/DT1)
      IF ((TI.LT.TE2).AND.(T2.LT.TE1)) GO TO 130
  110 THETAH = THT+.5*DT1*DS
  140 CONTINUE
  130 THETA(LP) = THT+DT1*DS
      SEP = -THETAH*DQDS
      THETAH = THETA(LP)
      THT = THETA(LP)
      SEPR(L) = (SEPR(L)*DS+SEP*DSOLD)/(DS+DSOLD)
      SEPR(LP) = SEP
      H(L) = (H(L)*DS+HH*DSOLD)/(DS+DSOLD)
      H(LP) = HH
      DELS(L) = H(L)*THETA(L)
      L = LP
      IF (L.NE.K2) GO TO 10
      H(K2)=H(K2-INC)+(DS/DSOLD)*(H(K2-INC)-H(K2-INC-INC))
      SEPR(K2) = 2.*SEPR(K2)-SEPR(K2-INC)
      DELS(K2) = H(K2)*THETA(K2)
      H(K1) = 0.
      SEPR(K1) = 0.
      RETURN
      END



      BLOCK DATA
      COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
     1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CO(162)
     2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162),DSUM(162)
     3 ,ANGOLD(162),XOLD(162),YOLD(162),ARCOLD(162),DELOLD(162)
      COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP,TC,CHD,DPHI,CL,RCL,YR
     1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,DCN,DSN,RA4,EPSIL,QCRIT,C1,C2
     2 ,C4,C5,C6,C7,BET,BETA,FSYM,XSEP,SEPM,TTLE(4),M,N,MM,NN,NSP
     3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCY,NRN,NG,IDIM,N2,N3,N4,NT,IXX
     4 , NPTS,LL,I,LSEP,M4
C     ****IDIM MUST BE SET TO THE FIRST DIMENSION OF PHI****
      DATA PI/3.14159265358979/ , EM/.75/ ,  ALP/0./ ,  CL/100./ ,
     1     PCH/.07/ , FSYM/1.0/ , RCL/1.0/ , BETA/0.0/, RN/20.E6/ ,
```

```
2 SEPM/.004/ , XSEP/.93/ , XP/0.0/ , M/160/ , N/30/ , NRN/1/ ,
3 NFC/80/ , NPTS/81/ , LL/0/ , NG/1/ , IS/2/ , IDIM/162/ , MODE /1/
4  , JK/0/ , N2/2/ , N3/3/ , N4/4/ , LSEP/161/ , IZ/125/ , ITYP/1/
 END
```

```
      PROGRAM FLO17(INPUT,OUTPUT,TAPE1,TAPE2,TAPE3,TAPE4,
     1                 TAPE5=INPUT,TAPE6=OUTPUT)
C     THREE DIMENSIONAL WING ANALYSIS IN TRANSONIC FLOW
C     USING SHEARED PARABOLIC COORDINATES
C     WITH STORAGE ON THE DISC
C     TAPES 1,2,3 ARE DISC FILES USED IN ROTATION TO STORE
C     THE THREE DIMENSIONAL POTENTIAL ARRAY DURING THE CALCULATION
C     TAPE 4 STORES ENOUGH INFORMATION TO CONTINUE THE CALCULATION
C     WITH ANOTHER COMPUTER RUN, IF THIS IS DESIRED
C     IT SHOULD THEN BE SPECIFIED AS A MAGNETIC TAPE
      COMMON        G(193,26,4),SEP1,
     1              A0(193),SEP2,A1(193),SEP3,A2(193),SEP4,A3(193),SEP5,
     2              B0(26),SEP6,B1(26),SEP7,B2(26),SEP8,B3(26),SEP9,
     3              C0(33),SEP10,C1(33),SEP11,C2(33),SEP12,C3(33),SEP13,
     4              S0(193,33),SEP14,E0(129),SEP15,Z0(129),SEP16,
     5              IV(193,33),SEP17,ITE1(33),SEP18,ITE2(33),SEP19,
     6              NX,NY,NZ,KTE1,KTE2,KSYM,SCAL,SCALZ,
     7              YAW,CYAW,SYAW,ALPHA,CA,SA,FMACH,N1,N2,N3,IO
      COMMON/FLO/ GK1(193,26),BUF1,GK2(193,26),BUF2,
     1              SXX(193),BUF3,SXZ(193),BUF4,SZZ(193),BUF5,
     2              SX(193),BUF6,SZ(193),BUF7,R0(193),BUF8,R1(193),BUF9,
     3              C(193),BUF10,D(193),BUF11,G1(193),BUF12,G2(193),
     4              STRIP,P1,P2,P3,BETA,FR,IR,JR,KR,DG,IG,JG,KG,NS
      DIMENSION     XS(241,11),YS(241,11),ZS(11),SLOPT(11),TRAIL(11),
     1              XP(241),YP(241),D1(241),D2(241),D3(241),
     2              X(193),Y(193),SV(193),SM(193),CP(193),
     3              CHORD(33),SCL(33),SCD(33),SCM(33),TITLE(20),
     4              FIT(3),COV0(3),P10(3),P20(3),P30(3),BETA0(3),
     5              STRIP0(3),FHALF(3),NP(11)
C     G IS REDUCED VELOCITY POTENTIAL
      ND         = 241
      NE         = 193
      IREAD      = 5
      IWRIT      = 6
      KPLOT      = 0
      IPLOT      = -1
      ISTOP      = 2
      N1         = 1
      N2         = 2
      N3         = 3
      REWIND 1
      REWIND 2
      REWIND 3
      REWIND 4
      JO         = 0
      RAD        = 57.2957795130823
    1 WRITE (IWRIT,600)
      WRITE (IWRIT,2)
    2 FORMAT(14H0PROGRAM FLO17,70X,32HANTONY JAMESON,COURANT INSTITUTE/
     1        50H0THREE DIMENSIONAL WING ANALYSIS IN TRANSONIC FLOW,
     2        28H USING PARABOLIC COORDINATES)
C     READ NEW DATA
C     PROGRAM STOPS ON READING THREE BLANK CARDS
```

```
      READ  (IREAD,530) TITLE
      WRITE (IWRIT,630) TITLE
      READ  (IREAD,500)
      READ  (IREAD,510)  FNX,FNY,FNZ,FPLOT,FCONT
      NX          = FNX
      NY          = FNY
      NZ          = FNZ
C     IF NX = 0 STOP
      IF (NX.LT.1) GO TO 301
      KPLOT       = FPLOT
C     NX,NY,NZ ARE NUMBERS OF CELLS IN FIRST GRID
C     KPLOT = 0 GIVES NO CALCOMP PLOTS
C     KPLOT = 1 GIVES THREE DIMENSIONAL CALCOMP PLOT
C     KPLOT = 2 GIVES CALCOMP PLOTS AT SEPARATE SPAN STATIONS
C     FCONT = 1. INDICATES CONTINUATION OF PREVIOUS RUN
      L           = 5*NX/16
      XMAX        = 2.*L/NX
      L           = 5*NZ/16
      ZMAX        = 2.*L/NZ
C     XMAX AND ZMAX ARE MAXIMUM EXTENT OF WING IN COMPUTATIONAL SPACE
      READ  (IREAD,500)
      NM          = 0
C     READ RELAXATION PARAMETERS FOR EACH MESH
   11 NM          = NM  +1
      READ  (IREAD,510) FIT(NM),COV0(NM),P10(NM),P20(NM),P30(NM),
     1                  BETA0(NM),STRIP0(NM),FHALF(NM)
C     FIT0 IS MAXIMUM NUMBER OF ITERATIONS
C     COV0 IS TOLERANCE FOR CONVERGENCE
C     P10 IS SUBSONIC RELAXATION FACTOR
C     P20 IS SUPERSONIC RELAXATION FACTOR
C     P30 IS RELAXATION FACTOR FOR CIRCULATION
C     BETA0 DETERMINES ADDED GST
C     STRIP0 IS WIDTH OF REGION FOR HORIZONTAL LINE RELAXATION
C     FHALF NE 0 INDICATES THAT A MESH REFINEMENT SHOULD BE PERFORMED
C     IF FHALF LT 0 INTERPOLATED POTENTIAL WILL BE SMOOTHED
C     ABS(FHALF) TIMES AFTER THE MESH REFINEMENT
      IF (FHALF(NM).NE.0..AND.NM.LT.3) GO TO 11
      FHALF(3)    = 0.
C     READ AERODYNAMIC PARAMETERS
      READ  (IREAD,500)
      READ  (IREAD,510) FMACH,YA,AL,CD0
      YAW         = YA/RAD
      ALPHA       = AL/RAD
C     FMACH IS FREE STREAM MACH NUMBER
C     YAW IS YAW ANGLE IN DEGREES
C     ALPHA IS ANGLE OF ATTACK NORMAL TO LEADING EDGE IN DEGREES
C     CD0 IS ADDED PARASITE DRAG COEFFICIENT
C     READ GEOMETRIC DATA
C     SQUARE ROOT TRANSFORMATION REQUIRES STRAIGHT LEADING EDGE
C     PLANFORM AND SECTION VARIATION ARE OTHERWISE UNRESTRICTED
C     XS AND YS ARE COORDINATES OF WING SURFACE
      CALL GEOM  (ND,NC,NP,ZS,XS,YS,SLOPT,TRAIL,XP,YP,
     1            XTE0,CHORD0,ZTIP,ISYM)
      KSYM        = ISYM
      IF (ALPHA.NE.0.) KSYM = 0
```

```
C     KSYM = 1 INDICATES SYMMETRIC NONLIFTING FLOW
      CYAW       = COS(YAW)
      SYAW       = SIN(YAW)
      CA         = CYAW*COS(ALPHA)
      SA         = CYAW*SIN(ALPHA)
      IF (FCONT.LT.1.) GO TO 101
C     READ PARAMETERS FOR CONTINUATION OF PREVIOUS CALCULATION
      READ  (4)    NX,NY,NZ,NM,K1,K2,NIT
      MX         = NX  +1
      MY         = NY  +2
      MZ         = NZ  +1
C     READ CURRENT VALUES OF POTENTIAL
      DO 62 K=1,MZ
      READ  (4)    ((G(I,J,1),I=1,MX),J=1,MY)
      BUFFER OUT(N3,1) (G(1,1,1),G(MX,MY,1))
      IF (UNIT(N3).GT.0.) GO TO 1
      BUFFER OUT(N1,1) (G(1,1,1),G(MX,MY,1))
C     GIVE UP IF VALUES ARE NOT PROPERLY STORED IN DISC FILES
      IF (UNIT(N1).GT.0.) GO TO 1
   62 CONTINUE
      READ  (4)    (E0(K),K=K1,K2)
      REWIND N3
      REWIND N1
      REWIND 4
C     CALCULATE MESH POINTS OF STRETCHED COORDINATES
C     A0,B0,C0 ARE MESH LOCATIONS
C     A1,B1,C1 ARE MULTIPLIERS FOR FIRST DERIVATIVES
C     A2,B2,C2 AND A3,B3,C3 ARE MULTIPLIERS FOR SECOND DERIVATIVES
  101 CALL COORD (NX,NY,NZ,XTE0,ZTIP,XMAX,ZMAX,SY,SCAL,SCALZ,
     1            AX,AY,AZ,A0,A1,A2,A3,B0,B1,B2,B3,C0,C1,C2,C3)
C     INTERPOLATE UNWRAPPED SURFACE AT MESH POINTS
C     S0 IS COORDINATE SURFACE CONTAINING WING SURFACE AND VORTEX SHEET
C     IV = 2 INDICATES POINTS ON WING SURFACE
C     IV = 1 INDICATES POINTS ON VORTEX SHEET
C     IV = 0 INDICATES POINTS ON THE SINGULAR LINE
C     OF THE SQUARE ROOT TRANSFORMATION
C     IV = -1 INDICATES POINTS JUST BEYOND EDGE OF WING OR VORTEX SHEET
C     IV = -2 INDICATES POINTS IN THE FREE STREAM ON THE CUT
C     IN THE SQUARE ROOT PLANES
      CALL SURF  (ND,NE,NC,NX,NZ,KSYM,NP,KTE1,KTE2,ITE1,ITE2,IV,
     1            YAW,SCAL,SCALZ,ZS,XS,YS,SLOPT,TRAIL,
     2            S0,Z0,A0,C0,XP,YP,D1,D2,D3,X,Y,IND)
C     IND = 0 INDICATES SPLINE FAILURE DUE TO BAD DATA,GIVE UP
      IF (IND.EQ.0) GO TO 291
      IF (FCONT.GE.1.) GO TO 111
      NM         = 1
      NIT        = 0
C     GENERATE STARTING GUESS FOR NEW CALCULATION
      CALL ESTIM
C     IO = 0 INDICATES DISC FAILURE,GIVE UP
      IF (IO.EQ.0) GO TO 1
      REWIND N3
      REWIND N1
  111 WRITE (IWRIT,600)
      FCONT      = 0.
```

```
      MX        = NX  +1
      MY        = NY  +2
      MZ        = NZ  +1
      MIT       = FIT(NM)
      KIT       = MIT  +2  -NM
      IF (NM.EQ.3) KIT = 10
      JIT       = 0
      COV       = COVO(NM)
      STRIP     = STRIPO(NM)
      BETA      = BETAO(NM)
      WRITE (IWRIT,112)
  112 FORMAT(49H0CHORDWISE CELL DISTRIBUTION IN SQUARE ROOT PLANE,
     1      46H AND MAPPED SURFACE COORDINATES AT CENTER LINE/
     2      15H0     X        ,15H SURFACE HEIGHT)
      LZ        = NZ/2  +1
      DO 114 I=2,NX
  114 WRITE (IWRIT,610) AO(I),SO(I,LZ)
      WRITE (IWRIT,116)
  116 FORMAT(15H0  TE LOCATION ,15H    POWER LAW  )
      WRITE (IWRIT,610) XMAX,AX
      WRITE (IWRIT,600)
      WRITE (IWRIT,118)
  118 FORMAT(46H0NORMAL CELL DISTRIBUTION IN SQUARE ROOT PLANE/
     1      15H0     Y        )
      KY        = NY  +1
      DO 120 J=2,KY
  120 WRITE (IWRIT,610) BO(J)
      WRITE (IWRIT,122)
  122 FORMAT(15H0  SCALE FACTOR,15H    POWER LAW  )
      WRITE (IWRIT,610) SY,AY
      WRITE (IWRIT,600)
      WRITE (IWRIT,124)
  124 FORMAT(27H0SPANWISE CELL DISTRIBUTION/
     1      15H0     Z        )
      DO 126 K=2,NZ
  126 WRITE (IWRIT,610) CO(K)
      WRITE (IWRIT,128)
  128 FORMAT(15H0  TIP LOCATION,15H    POWER LAW  )
      WRITE (IWRIT,610) ZMAX,AZ
      WRITE (IWRIT,600)
      WRITE (IWRIT,132)
  132 FORMAT(19H0ITERATIVE SOLUTION/
     1      43H0STRIP WIDTH FOR HORIZONTAL LINE RELAXATION)
      WRITE (IWRIT,610) STRIP
      WRITE (IWRIT,134)
  134 FORMAT(15H0      NX     ,15H          NY  ,15H          NZ   )
      WRITE (IWRIT,640) NX,NY,NZ
      CALL SECOND(T)
      WRITE (IWRIT,660) T
      WRITE (IWRIT,136)
  136 FORMAT(15H0    MACH NO  ,15H       YAW    ,15H  ANG OF ATTACK)
      WRITE (IWRIT,610) FMACH,YA,AL
      WRITE (IWRIT,138)
  138 FORMAT(10H0ITERATION,15H     CORRECTION ,4H  I ,4H  J ,4H  K ,
     1                     15H     RESIDUAL  ,4H  I ,4H  J ,4H  K ,
```

```
     2       10H CIRCULATN,10H REL FCT 1,10H REL FCT 2,10H REL FCT 3,
     3       10H    BETA   ,10H SONIC PTS)
  141 NIT        = NIT  +1
      JIT        = JIT  +1
      P1         = P10(NM)
      P2         = P20(NM)
      P3         = P30(NM)
      IF (NIT.LE.10) P1 = 1.
      IF (NIT.LE.10) P3 = 1.
C     UPDATE POTENTIAL BY RELAXATION
C     EACH ITERATION IS ONE STEP IN ARTIFICIAL TIME
C     EQUIVALENT TIME DEPENDENT EQUATION IS
C     (1.  -M**2)*GSS  +GMM  +GNN  +TERMS IN GST,GMT,GNT AND GT
      CALL MIXFLO
C     IO = 0 INDICATES DISC FAILURE,RETURN TO PREVIOUS ITERATION
      IF (IO.EQ.0) GO TO 151
      JO         = 0
      REWIND N1
      REWIND N2
C     UPDATED VALUES ARE STORED IN DISC FILES 1,2,3 IN ROTATION
C     SET FILE NUMBERS FOR NEXT ITERATION
      N          = N1
      N1         = N2
      N2         = N3
      N3         = N
C     WRITE NUMBER OF ITERATIONS NIT,
C     LARGEST CORRECTION DG AND ITS LOCATION IG,JG,KG,
C     LARGEST RESIDUAL FR AND ITS LOCATION IR,JR,KR,
C     CIRCULATION EO,RELAXATION PARAMETERS P1,P2,P3 AND BETA,
C     AND NUMBER OF SUPERSONIC POINTS NS
      WRITE (IWRIT,650) NIT,DG,IG,JG,KG,FR,IR,JR,KR,EO(LZ),
     1                  P1,P2,P3,BETA,NS
C     EVERY KIT CYCLES SAVE CURRENT VALUES ON TAPE 4
C     TO ALLOW RESTART IN CASE OF MACHINE FAILURE
      IF (JIT.EQ.KIT) GO TO 251
      IF (NIT.LT.MIT.AND.ABS(DG).GT.COV.AND.ABS(DG).LT.10.) GO TO 141
C     STOP ON ITERATION COUNT OR IF ERROR MEETS TOLERANCE
C     OR IF ITERATIONS DIVERGE.
      GO TO 161
C     JO = 1 INDICATES SUCCESSIVE DISC FAILURES,GIVE UP
  151 IF (JO.EQ.1) GO TO 1
      REWIND N1
      REWIND N2
      JO         = 1
C     RESET FILE NUMBERS FOR PREVIOUS ITERATION
      N          = N3
      N3         = N2
      N2         = N1
      N1         = N
      GO TO 141
C     GENERATE AND WRITE AERODYNAMIC PARAMETERS FOR EACH SPAN STATION
C     READ FROM THE DISC AND PROCESS SLICES OF THE G ARRAY FOR FIXED Z,
C     REPRESENTING VALUES OF POTENTIAL ON X-Y PLANES
C     CONTAINING SUCCESSIVE WING SECTIONS
  161 LX         = NX/2  +1
```

```
      CALL SECOND(T)
      WRITE (IWRIT,660) T
      WRITE (IWRIT,600)
C     READ FIRST THREE SLICES OF POTENTIAL ARRAY FROM DISC FILE
      DO 162 L=1,3
      BUFFER IN (N1,1) (G(1,1,L),G(MX,MY,L))
C     RETURN TO PREVIOUS ITERATION IN EVENT OF DISC FAILURE
      IF (UNIT(N1).GT.0.) GO TO 151
  162 CONTINUE
      K             = 2
C     INCREMENT Z
  171 K             = K   +1
      IF (K.EQ.MZ) GO TO 191
C     SHIFT SLICES OF POTENTIAL ARRAY
      DO 172 J=1,MY
      DO 172 I=1,MX
      G(I,J,1)   = G(I,J,2)
  172 G(I,J,2)   = G(I,J,3)
C     READ SLICE OF POTENTIAL ARRAY FROM DISC FILE
      BUFFER IN (N1,1) (G(1,1,3),G(MX,MY,3))
C     RETURN TO PREVIOUS ITERATION IN EVENT OF DISC FAILURE
      IF (UNIT(N1).GT.0.) GO TO 151
      IF (K.LT.KTE1.OR.K.GT.KTE2) GO TO 171
      Z             = SCALZ*CO(K)
      I1            = ITE1(K)
      I2            = ITE2(K)
C     CALCULATE SURFACE SPEED SV,MACH NUMBER SM,PRESSURE COEFFICIENT CP
C     AND COORDINATES X,Y OF WING SECTION
      CALL VELO (K,2,SV,SM,CP,X,Y)
      CHORD(K)   = X(I1)   -X(LX)
C     CALCULATE SECTION LIFT,DRAG AND MOMENT COEFFICIENTS
      CALL FORCF (I1,I2,X,Y,CP,AL,CHORD(K),0.,SCL(K),SCD(K),SCM(K))
      IF (KPLOT.GT.0.AND.K.GT.KTE1) GO TO 185
      WRITE (IWRIT,600)
      WRITE (IWRIT,182)
  182 FORMAT(24H0SECTION CHARACTARISTICS/
     1       15H0     MACH NO     ,15H        YAW     ,15H  ANG OF ATTACK)
      WRITE (IWRIT,610) FMACH,YA,AL
      WRITE (IWRIT,184)
  184 FORMAT(15H0   SPAN STATION,15H         CL      ,15H         CD      ,
     1       15H         CM      )
  185 WRITE (IWRIT,610) Z,SCL(K),SCD(K),SCM(K)
C     IF KPLOT = 0 LIST AND PRINT-PLOT CP
      IF (KPLOT.EQ.0) CALL CPLOT (I1,I2,FMACH,X,Y,CP)
      IF (KPLOT.NE.2) GO TO 171
C     IF KPLOT = 2 GENERATE CALCOMP PLOT OF SECTION CP
      CALL GRAPH (IPLOT,I1,I2,X,Y,CP,TITLE,FMACH,YA,AL,
     1            Z,SCL(K),SCD(K),CHORD0)
      IPLOT         = 0
      GO TO 171
C     CALCULATE TOTAL LIFT ,DRAG AND MOMENT COEFFICIENTS
  191 CALL TOTFOR(KTE1,KTE2,CHORD,SCL,SCD,SCM,CO,SCALZ,.25,
     1            CL,CD1,CMP,CMR,CMY)
      CD1           = CYAW*CD1
      CD            = CD0   +CD1
```

```
      VLD1       = 0.
      IF (ABS(CD1).GT.1.E-6) VLD1 = CL/CD1
      VLD        = 0.
      IF (ABS(CD).GT.1.E-6) VLD = CL/CD
      WRITE (IWRIT,600)
      REWIND N1
      CALL CHARTY
      WRITE (IWRIT,600)
      WRITE (IWRIT,192)
  192 FORMAT(21H0WING CHARACTARISTICS/
     1      15H0     MACH NO    ,15H        YAW     ,15H  ANG OF ATTACK)
      WRITE (IWRIT,610) FMACH,YA,AL
      WRITE (IWRIT,194)
  194 FORMAT(15H0         CL     ,15H    CD FORM    ,15H  CD FRICTION ,
     1      15H          CD     ,15H    L/D FORM   ,15H      L/D      )
      WRITE (IWRIT,610) CL,CD1,CD0,CD,VLD1,VLD
      WRITE (IWRIT,196)
  196 FORMAT(15H0    CM PITCH   ,15H    CM ROLL    ,15H      CM YAW   )
      WRITE (IWRIT,610) CMP,CMR,CMY
      REWIND N1
      IF (KPLOT.LT.1) GO TO 201
C     IF KPLOT GT 0 GENERATE THREE DIMENSIONAL CALCOMP PLOT
      CALL THREED(IPLOT,SV,SM,CP,X,Y,TITLE,YA,AL,VLD,CL,CD,CHORD0)
      IPLOT      = 0
C     IO = 0 INDICATES DISC FAILURE,RETURN TO PREVIOUS ITERATION
      IF (IO.EQ.0) GO TO 151
C     STOP ON OPERATOR COMMAND
  201 IF (ISTOP.EQ.1) GO TO 301
      IF (FHALF(NM).EQ.0.) GO TO 1
C     REFINE GRID IF FHALF NE 0
      NX         = NX  +NX
      NY         = NY  +NY
      NZ         = NZ  +NZ
C     RECALCULATE MESH LOCATIONS ON REFINED GRID
      CALL COORD (NX,NY,NZ,XTE0,ZTIP,XMAX,ZMAX,SY,SCAL,SCALZ,
     1            AX,AY,AZ,A0,A1,A2,A3,B0,B1,B2,B3,C0,C1,C2,C3)
C     INTERPOLATE UNWRAPPED SURFACE ON REFINED GRID
      CALL SURF  (ND,NE,NC,NX,NZ,KSYM,NP,KTE1,KTE2,ITE1,ITE2,IV,
     1            YAW,SCAL,SCALZ,ZS,XS,YS,SLOPT,TRAIL,
     2            S0,Z0,A0,C0,XP,YP,D1,D2,D3,X,Y,IND)
C     IND = 0 INDICATES SPLINE FAILURE DUE TO BAD DATA,GIVE UP
      IF (IND.EQ.0) GO TO 291
C     INTERPOLATE POTENTIAL ON REFINED GRID
      CALL REFIN
C     IO = 0 INDICATES DISC FAILURE,RETURN TO PREVIOUS GRID
      IF (IO.EQ.0) GO TO 221
      REWIND N1
      REWIND N2
      NSMOO      = -FHALF(NM)
C     IF FHALF LT 0 SMOOTH INTERPOLATED POTENTIAL ABS(FHALF) TIMES
      IF (NSMOO.LT.1) GO TO 211
      DO 202 N=1,NSMOO
      CALL SMOO
C     IO = 0 INDICATES DISC FAILURE,RETURN TO PREVIOUS GRID
      IF (IO.EQ.0) GO TO 221
```

```
      REWIND N1
  202 REWIND N2
C     RESET FILE NUMBERS
  211 N          = N1
      N1         = N2
      N2         = N3
      N3         = N
C     INCREMENT NUMBER OF MESH
      NM         = NM  +1
      NIT        = 0
      GO TO 111
C     RESTORE PREVIOUS GRID
  221 NX         = NX/2
      NY         = NY/2
      NZ         = NZ/2
C     RECALCULATE MESH LOCATIONS ON PREVIOUS GRID
      CALL COORD (NX,NY,NZ,XTE0,ZTIP,XMAX,ZMAX,SY,SCAL,SCALZ,
     1           AX,AY,AZ,A0,A1,A2,A3,B0,B1,B2,B3,C0,C1,C2,C3)
C     INTERPOLATE UNWRAPPED SURFACE ON PREVIOUS GRID
      CALL SURF  (ND,NE,NC,NX,NZ,KSYM,NP,KTE1,KTE2,ITE1,ITE2,IV,
     1           YAW,SCAL,SCALZ,ZS,XS,YS,SLOPT,TRAIL,
     2           S0,Z0,A0,C0,XP,YP,D1,D2,D3,X,Y,IND)
C     IND = 0 INDICATES SPLINE FAILURE DUE TO BAD DATA,GIVE UP
      IF (IND.EQ.0) GO TO 291
      GO TO 151
C     WRITE THREE COPIES OF INFORMATION NEEDED TO RESTART ON TAPE 4
  251 K1         = KTE1  -1
      K2         = KTE2  +ITE2(KTE2)  -NX/2
      DO 252 M=1,3
      WRITE (4)    NX,NY,NZ,NM,K1,K2,NIT
      DO 262 K=1,MZ
      BUFFER IN (N1,1) (G(1,1,1),G(MX,MY,1))
C     RETURN TO PREVIOUS ITERATION IN EVENT OF DISC FAILURE
      IF (UNIT(N1).GT.0.) GO TO 281
  262 WRITE (4)    ((G(I,J,1),I=1,MX),J=1,MY)
      REWIND N1
      WRITE (4)    (E0(K),K=K1,K2)
      ENDFILE 4
  252 CONTINUE
      REWIND 4
C     ALLOW OPERATOR TO STOP CALCULATION
      CALL SSWTCH(1,ISTOP)
      IF (ISTOP.EQ.1) GO TO 161
      JIT        = 0
      IF (NIT.LT.MIT.AND.ABS(DG).GT.COV.AND.ABS(DG).LT.10.) GO TO 141
      GO TO 161
  281 REWIND 4
      GO TO 151
  291 WRITE (IWRIT,600)
      WRITE (IWRIT,292)
  292 FORMAT(24H0BAD DATA,SPLINE FAILURE)
      GO TO 1
C     TERMINATE CALCOMP FILE
  301 IF (KPLOT.GT.0) CALL PLOT(0.,0.,999)
      STOP
```

```
  500 FORMAT(1X)
  510 FORMAT(8E10.7)
  530 FORMAT(20A4)
  600 FORMAT(1H1)
  610 FORMAT(F12.4,7F15.4)
  620 FORMAT(8E15.5)
  630 FORMAT(1H0,20A4)
  640 FORMAT(I8,7I15)
  650 FORMAT(I10,E15.5,3I4,E15.5,3I4,5F10.5,I10)
  660 FORMAT(15H0COMPUTING TIME,F10.3,10H   SECONDS)
      END



      SUBROUTINE  GEOM  (ND,NC,NP,ZS,XS,YS,SLOPT,TRAIL,XP,YP,
     1                   XTE0,CHORD0,ZTIP,ISYM)
C     GEOMETRIC DEFINITION OF WING
C     XS AND YS ARE COORDINATES OF WING SURFACE
C     THE SECTIONS AT DIFFERENT SPAN STATIONS ARE ALIGNED
C     SO THAT THEIR SINGULAR POINTS AS DEFINED BY THE DATA
C     LIE ON A STRAIGHT LINE
C     THE WING IS UNWRAPPED ABOUT THIS LINE
C     BY A SQUARE ROOT TRANSFORMATION TO PARABOLIC COORDINATES
      DIMENSION   XS(ND,1),YS(ND,1),ZS(1),SLOPT(1),TRAIL(1),
     1            XP(1),YP(1),NP(1)
      IREAD      = 5
      IWRIT      = 6
      RAD        = 57.2957795130823
      READ  (IREAD,500)
      READ  (IREAD,510) FNC
C     NC IS NUMBER OF SPAN STATIONS AT WHICH THE SECTION IS DEFINED
C     IF NC LT 2 THE GEOMETRY IS ASSSUMED TO BE UNCHANGED
C     FROM THE PREVIOUS CASE
      IF (FNC.LT.2.) RETURN
      NC         = FNC
      ISYM       = 1
      XTE0       = 0.
      CHORD0     = 0.
      K          = 1
   11 READ  (IREAU,500)
      READ  (IREAD,510) ZS(K),CHORD,THICK,AL,FSEC
      ALPHA      = AL/RAD
C     ZS IS SPAN STATION
C     PROFILE IS SCALED TO A LENGTH EQUAL TO CHORD
C     AND ROTATED THROUGH THE TWIST ANGLE AL
C     MEASURED NORMAL TO THE LEADING EDGE IN DEGREES
C     ITS THICKNESS CHORD RATIO IS REDUCED BY THE FACTOR THICK
C     FSEC = 1 INDICATES THAT A NEW PROFILE IS DEFINED
C     BY A TABLE OF COORDINATES
C     FSEC = 0 INDICATES THAT THE PROFILE IS DERIVED
C     FROM THE EXISTING TABLE OF COORDINATES
      IF (K.GT.1.AND.FSEC.EQ.0.) GO TO 31
      READ  (IREAD,500)
      READ  (IREAD,510) FSYM,FNU,FNL
```

```
      NU          = FNU
      NL          = FNL
      N           = NU   +NL   -1
C     FSYM = 1 INDICATES SYMMETRIC PROFILE
C     FOR WHICH ONLY THE UPPER SURFACE IS READ
C     NU AND NL ARE NUMBERS OF UPPER AND LOWER SURFACE POINTS
      READ   (IREAD,500)
C     READ TRAILING EDGE INCLUDED ANGLE AND SLOPE,
C     AND COORDINATES OF SINGULAR POINT
      READ   (IREAD,510) TRL,SLT,XSING,YSING
      READ   (IREAD,500)
C     READ UPPER SURFACE COORDINATES
      DO 12 I=NL,N
   12 READ   (IREAD,510) XP(I),YP(I)
      L           = NL   +1
      IF (FSYM.GT.0.) GO TO 15
      READ   (IREAD,500)
C     READ LOWER SURFACE COORDINATES
      DO 14 I=1,NL
      READ   (IREAD,510) VAL,DUM
      J           = L   -I
      XP(J)       = VAL
   14 YP(J)       = DUM
      GO TO 21
   15 J           = L
      DO 16 I=NL,N
      J           = J   -1
      XP(J)       = XP(I)
   16 YP(J)       = -YP(I)
   21 WRITE (IWRIT,600)
      WRITE (IWRIT,22) ZS(K)
   22 FORMAT(16H0PROFILE AT Z = ,F10.5/
     1         15H0     TE ANGLE   ,15H      TE SLOPE   ,15H         X SING    ,
     2         15H       Y SING    )
      WRITE (IWRIT,610) TRL,SLT,XSING,YSING
      WRITE (IWRIT,24)
   24 FORMAT(15H0          X      ,15H          Y      )
      DO 26 I=1,N
   26 WRITE (IWRIT,610) XP(I),YP(I)
C     SCALE AND ROTATE PROFILE
   31 SCALE       = CHORD/(XP(1)   -XP(NL))
      XX          = XP(NL)   +(XSING   -XP(NL))*THICK
      YY          = YP(NL)   +(YSING   -YP(NL))*THICK
      CA          = COS(ALPHA)
      SA          = SIN(ALPHA)
      DO 32 I=1,N
      XS(I,K)     = SCALE*((XP(I)   -XX)*CA   +THICK*(YP(I)   -YY)*SA)
   32 YS(I,K)     = SCALE*(THICK*(YP(I)   -YY)*CA   -(XP(I)   -XX)*SA)
      SLOPT(K)    = THICK*SLT   -TAN(ALPHA)
      TRAIL(K)    = THICK*TRL/RAD
      NP(K)       = N
C     NP IS NUMBER OF POINTS DEFINING PROFILE
      CHORD0      = AMAX1(CHORD0,CHORD)
      XTE0        = AMAX1(XTE0,XS(1,K))
C     CHORD0 AND XTE0 ARE MAXIMUM CHORD AND REARMOST EXTENT OF WING
```

```
      IF (FSYM.LE.0..OR.ALPHA.NE.0.) ISYM = 0
C     ISYM = 1 INDICATES SYMMETRIC WING
      WRITE (IWRIT,52) ZS(K)
   52 FORMAT(27H0SECTION DEFINITION AT Z = ,F10.5/
     1        15H0       CHORD     ,15HTHICKNESS RATIO,15H       ALPHA    )
      WRITE (IWRIT,610) CHORD,THICK,AL
      K          = K  +1
      IF (K.LE.NC) GO TO 11
      Z0         = .5*(ZS(1)  +ZS(NC))
      DO 62 K=1,NC
   62 ZS(K)      = ZS(K)  -Z0
      ZTIP       = ZS(NC)
C     ZTIP IS TIP LOCATION AFTER WING HAS BEEN CENTERED AT Z = 0.
      RETURN
  500 FORMAT(1X)
  510 FORMAT(8E10.7)
  600 FORMAT(1H1)
  610 FORMAT(F12.4,7F15.4)
      END



      SUBROUTINE  COORD (NX,NY,NZ,XTE0,ZTIP,XMAX,ZMAX,SY,SCAL,SCALZ,
     1                  AX,AY,AZ,A0,A1,A2,A3,B0,B1,B2,B3,C0,C1,C2,C3)
C     SETS UP STRETCHED PARABOLIC AND SPANWISE COORDINATES
C     STRETCHING LAW HAS FORM X = XX,XX LT C,
C     X = C  +(XX  -C)/(1.  -((XX  -C)/(1.  -C))**2)**AX,XX GT C
C     WHERE AX DETERMINES POWER LAW
C     IN COMPUTATIONAL SPACE XX RANGES FROM -1. TO 1.,
C     YY RANGES FROM 0. TO 1.,ZZ RANGES FROM -1. TO 1.
C     A0,B0,C0 ARE MESH LOCATIONS
C     A1,B1,C1 ARE MULTIPLIERS FOR FIRST DERIVATIVES
C     A2,B2,C2 AND A3,B3,C3 ARE MULTIPLIERS FOR SECOND DERIVATIVES
C     IF DGI AND DGII ARE FIRST AND SECOND DIFFERENCES
C     GX = A1*DGI AND GXX = A2*(DGII  +A3*DGI)
      DIMENSION   A0(1),A1(1),A2(1),A3(1),B0(1),B1(1),B2(1),B3(1),
     1            C0(1),C1(1),C2(1),C3(1)
      KY         = NY  +1
      DX         = 2./NX
      DY         = 1./NY
      DZ         = 2./NZ
C     SELECT POWER LAWS
      AX         = .5
      AY         = .5
      AZ         = .5
      SY         = .5
C     SY SCALES Y SPACING RELATIVE TO X SPACING
      SCAL       = XTE0/(.50001*XMAX*XMAX)
      SCALZ      = ZTIP/(1.000001*ZMAX)
      V2         = (DX/DY)**2
      W1         = SCAL/SCALZ
      W2         = (W1*DX/DZ)**2
C     GENERATE X MESH
      DO 12 I=2,NX
```

```
      DD          = (I   -1)*DX   -1.
      B           = 1.
      IF (ABS(DD).GT.XMAX) GO TO 13
      D0          = DD
      D1          = 1.
      D2          = 0.
      GO TO 14
   13 IF (DD.LT.0.) B = -1.
      A           = 1.   -((DD   -B*XMAX)/(1.   -XMAX))**2
      C           = A**AX
      D           = (AX   +AX   -1.)*(1.   -A)
      D0          = B*XMAX   +(DD   -B*XMAX)/C
      D1          = A*C/(1.   +D)
      D2          = -(AX   +AX)*(DD   -B*XMAX)
     1                       *(3.   +D)/((1.   +D)*A*(1.   -XMAX)**2)
   14 A0(I)       = D0
      A1(I)       = .5*D1/DX
      A2(I)       = D1*D1
   12 A3(I)       = .5*DX*D2
C     GENERATE Y MESH
      DO 22 J=2,KY
      DD          = (KY   -J)*DY
      A           = 1.   -DD*DD
      C           = A**AY
      D           = (AY   +AY   -1.)*(1.   -A)
      D1          = A*C/((1.   +D)*SY)
      B0(J)       = SY*DD/C
      B1(J)       = .5*D1/DY
      B2(J)       = D1*D1*V2
   22 B3(J)       = -AY*DD*DY*(3.   +D)/((1.   +D)*A)
C     GENERATE Z MESH
      DO 32 K=2,NZ
      DD          = (K   -1)*DZ   -1.
      B           = 1.
      IF (ABS(DD).GT.ZMAX) GO TO 33
      D0          = DD
      D1          = 1.
      D2          = 0.
      GO TO 34
   33 IF (DD.LT.0.) B = -1.
      A           = 1.   -((DD   -B*ZMAX)/(1.   -ZMAX))**2
      C           = A**AZ
      D           = (AZ   +AZ   -1.)*(1.   -A)
      D0          = B*ZMAX   +(DD   -B*ZMAX)/C
      D1          = A*C/(1.   +D)
      D2          = -(AZ   +AZ)*(DD   -B*ZMAX)
     1                       *(3.   +D)/((1.   +D)*A*(1.   -ZMAX)**2)
   34 C0(K)       = D0
      C1(K)       = .5*D1*W1/DZ
      C2(K)       = D1*D1*W2
   32 C3(K)       = .5*DZ*D2
      RETURN
      END
```

```
      SUBROUTINE  SURF  (ND,NE,NC,NX,NZ,KSYM,NP,KTE1,KTE2,ITE1,ITE2,IV,
     1                   YAW,SCAL,SCALZ,ZS,XS,YS,SLOPT,TRAIL,
     2                   S0,Z0,A0,C0,XP,YP,D1,D2,D3,X,Y,IND)
C     INTERPOLATES MAPPED WING SURFACE AT MESH POINTS
C     INTERPOLATION IS LINEAR IN PHYSICAL PLANE
C     AND QUADRATIC IN TRANSFORMED PLANE Y = 0,
C     XS AND YS ARE WING COORDINATES IN PHYSICAL SPACE
C     AT SPAN STATIONS ZS
C     S0 IS COORDINATE SURFACE CONTAINING WING SURFACE AND VORTEX SHEET
C     IN TRANSFORMED SPACE
C     Z0 IS STREAMWISE PROJECTION ON SINGULAR LINE
C     OF TRAILING EDGE AND DOWNSTREAM SIDE EDGE
C     USED IN DETERMINATION OF STRENGTH OF VORTEX SHEET
C     IV = 2 INDICATES POINTS ON WING SURFACE
C     IV = 1 INDICATES POINTS ON VORTEX SHEET
C     IV = 0 INDICATES POINTS ON THE SINGULAR LINE
C     OF THE SQUARE ROOT TRANSFORMATION
C     IV = -1 INDICATES POINTS JUST BEYOND EDGE OF WING OR VORTEX SHEET
C     IV = -2 INDICATES POINTS IN THE FREE STREAM ON THE CUT
C     IN THE SQUARE ROOT PLANES
C     KTE1 AND KTE2 ARE K INDICES AT WING TIPS
C     ITE1 AND ITE2 ARE I INDICES AT LOWER AND UPPER TRAILING EDGE
C     INTERPOLATION IS LINEAR IN PHYSICAL PLANE
      DIMENSION   S0(NE,1),XS(ND,1),YS(ND,1),ZS(1),SLOPT(1),TRAIL(1),
     1            A0(1),C0(1),Z0(1),XP(1),YP(1),D1(1),D2(1),D3(1),
     2            X(1),Y(1),IV(NE,1),NP(1),ITE1(1),ITE2(1)
      PI          = 3.14159265358979
      TYAW        = .5*SCAL*TAN(YAW)
      DX          = 2./NX
      LX          = NX/2  +1
      MX          = NX  +1
      MZ          = NZ  +1
C     VORTEX AND EDGE POINTS ARE REPRESENTED BY SETTING IV TO IV0 OR IV1
C     IF WING IS SYMMETRIC VORTEX AND EDGE POINTS DO NOT EXIST
C     AND ALL POINTS OFF WING SURFACE ARE TREATED AS FREE STREAM POINTS
C     BY SETTING IV0 AND IV1 TO -2
      IV0         = 1  -KSYM  -KSYM  -KSYM
      IV1         = -1  -KSYM
C     INITIALIZE IV FOR POINTS OUTSIDE WING AND VORTEX SHEET
C     AND POINTS ON THE SINGULAR LINE
      DO 2 K=1,MZ
      ITE1(K)     = MX
      ITE2(K)     = MX
      DO 4 I=1,MX
      IV(I,K)     = -2
    4 S0(I,K)     = 0.
    2 IV(LX,K)    = 0
      K           = 1
      K2          = 2
   11 K           = K  +1
      IF (K.EQ.MZ) GO TO 91
      Z           = SCALZ*C0(K)
      IF (Z.GE.ZS(1)) GO TO 13
C     Z IS SHORT OF FIRST SPAN STATION
      KTE1        = K  +1
```

```
C     TRY NEXT VALUE OF Z
      GO TO 11
   13 IF (Z.GT.ZS(NC)) GO TO 81
      K2          = K2  -1
C     Z IS ON WING
C     INTERPOLATE PROFILE BETWEEN ADJACENT SPAN STATIONS
   21 K2          = K2  +1
      K1          = K2  -1
      R2          = 1.
      IF (ZS(K2)  -Z) 21,25,23
   23 R2          = (Z  -ZS(K1))/(ZS(K2)  -ZS(K1))
   25 R1          = 1.  -R2
      C           = R1*XS(1,K1)  +R2*XS(1,K2)
      CC          = SQRT((C  +C)/SCAL)
C     C IS INTERPOLATED CHORD
C     CC IS CHORD IN SQUARE ROOT PLANE
C     DETERMINE I INDICES AT TRAILING EDGE
      DO 32 I=2,NX
      IF ((A0(I)  +.5*DX).LT.-CC) I1 = I  +1
      IF ((A0(I)  -.5*DX).LT.CC) I2 = I
   32 CONTINUE
      ITE1(K)     = I1
      ITE2(K)     = I2
C     SCALE CHORD SO THAT TRAILING EDGE COINCIDES
C     WITH NEAREST MESH LOCATION
      CC          = A0(I2)/CC
C     PROJECT TRAILING EDGE POINT ON SINGULAR LINE
      Z0(K)       = Z  -TYAW*A0(I2)*A0(I2)
C     GENERATE TRANSFORMED PROFILE AT SPAN STATIONS K1 AND K2
C     AND CORRESPONDING PROFILE AT INTERPOLATED SPAN STATION K
C     SET KK TO INDEX OF FIRST SPAN STATION
      KK          = K1
      P           = R1
   41 N           = NP(KK)
      Q           = SQRT(XS(1,KK)/C)/CC
C     SCALE MESH LOCATIONS FOR INTERPOLATION OF PROFILE
      DO 42 I=2,NX
   42 X(I)        = Q*A0(I)
C     APPLY SQUARE ROOT TRANSFORMATION TO PROFILE
C     USING CONTINUITY TO OBTAIN CORRECT BRANCH
      ANGL        = PI  +PI
      U           = 1.
      V           = 0.
      DO 44 I=1,N
      ANGL        = ANGL  +ATAN((U*YS(I,KK)  -V*XS(I,KK))
     1                        /(U*XS(I,KK)  +V*YS(I,KK)))
      R           = SQRT(XS(I,KK)**2  +YS(I,KK)**2)
      U           = XS(I,KK)
      V           = YS(I,KK)
      R           = SQRT((R  +R)/SCAL)
      XP(I)       = R*COS(.5*ANGL)
   44 YP(I)       = R*SIN(.5*ANGL)
C     DETERMINE SLOPES T1 AND T2 OF LOWER AND UPPER SURFACE
C     AT TRAILING EDGE TO PROVIDE END CONDITIONS FOR SPLINE
      ANGL        = ATAN(SLOPT(KK))
```

```
      ANGL1       = ATAN(YS(1,KK)/XS(1,KK))
      ANGL2       = ATAN(YS(N,KK)/XS(N,KK))
      ANGL1       = ANGL  -.5*(ANGL1  -TRAIL(KK))
      ANGL2       = ANGL  +.5*(ANGL2  +TRAIL(KK))
      T1          = TAN(ANGL1)
      T2          = TAN(ANGL2)
C     FIT SPLINE TO UNWRAPPED PROFILE
      CALL SPLIF (1,N,XP,YP,D1,D2,D3,1,T1,1,T2,0,0.,IND)
C     INTERPOLATE SPLINE AT MESH LOCATIONS
      CALL INTPL (I1,I2,X,Y,1,N,XP,YP,D1,D2,D3,0)
C     CONTINUE UNWRAPPED PROFILE BEYOND LOWER TRAILING EDGE
      X1          = .25*XS(1,KK)
      A           = SLOPT(KK)*(XS(1,KK)  -X1)
      B           = 1./(XS(1,KK)  -X1)
      ANGL        = PI  +PI
      U           = 1.
      V           = 0.
      M           = I1  -1
      DO 52 I=2,M
      XX          = .5*SCAL*X(I)**2
      D           = B*(XX  -X1)
      YY          = YS(1,KK)  +A*ALOG(D)/D
      ANGL        = ANGL  +ATAN((U*YY  -V*XX)/(U*XX  +V*YY))
      R           = SQRT(XX**2  +YY**2)
      U           = XX
      V           = YY
      R           = SQRT((R  +R)/SCAL)
   52 Y(I)        = R*SIN(.5*ANGL)
C     CONTINUE UNWRAPPED PROFILE BEYOND UPPER TRAILING EDGE
      A           = SLOPT(KK)*(XS(N,KK)  -X1)
      B           = 1./(XS(N,KK)  -X1)
      ANGL        = 0.
      U           = 1.
      V           = 0.
      M           = I2  +1
      DO 54 I=M,NX
      XX          = .5*SCAL*X(I)**2
      D           = B*(XX  -X1)
      YY          = YS(N,KK)  +A*ALOG(D)/D
      ANGL        = ANGL  +ATAN((U*YY  -V*XX)/(U*XX  +V*YY))
      R           = SQRT(XX**2  +YY**2)
      U           = XX
      V           = YY
      R           = SQRT((R  +R)/SCAL)
   54 Y(I)        = R*SIN(.5*ANGL)
C     ADD CONTRIBUTION TO PROFILE AT INTERPOLATED SPAN STATION
      Q           = P*Q*CC*CC
      DO 62 I=2,NX
   62 SO(I,K)     = SO(I,K)  +Q*Y(I)
      IF (KK.EQ.K2) GO TO 71
C     SET KK TO INDEX OF SECOND SPAN STATION
      KK          = K2
      P           = R2
      GO TO 41
C     SET IV TO INDICATE SURFACE POINT
```

```
   71 DO 72 I=I1,I2
   72 IV(I,K)    = 2
C     SEARCH FOR POINTS ON VORTEX SHEET AT I INDICES OFF WING SURFACE
      M          = I1  -1
      DO 74 I=2,M
C     DETERMINE STREAMWISE PROJECTION ON SINGULAR LINE
      ZZ         = Z   -TYAW*A0(I)*A0(I)
C     SET IV TO INDICATE VORTEX POINT
C     IF PROJECTION IS BEYOND PROJECTION OF UPSTREAM TIP
      IF (ZZ.GE.Z0(KTE1)) IV(I,K) = IV0
   74 CONTINUE
      M          = I2  +1
      DO 76 I=M,NX
C     DETERMINE STREAMWISE PROJECTION ON SINGULAR LINE
      ZZ         = Z   -TYAW*A0(I)*A0(I)
C     SET IV TO INDICATE VORTEX POINT
C     IF PROJECTION IS BEYOND PROJECTION OF UPSTREAM TIP
      IF (ZZ.GE.Z0(KTE1)) IV(I,K) = IV0
   76 CONTINUE
      KTE2       = K
      GO TO 11
C     Z IS BEYOND LAST SPAN STATION
C     SEARCH FOR POINTS ON VORTEX SHEET
   81 DO 82 I=2,NX
C     DETERMINE STREAMWISE PROJECTION ON SINGULAR LINE
      ZZ         = Z   -TYAW*A0(I)*A0(I)
C     SET IV TO INDICATE VORTEX POINT
C     IF PROJECTION IS WITHIN PROJECTION OF DOWNSTREAM TIP
      IF (ZZ.LE.ZS(NC).AND.ZZ.GE.Z0(KTE1)) IV(I,K) = IV0
   82 CONTINUE
      GO TO 11
   91 N          = KTE2
      IF (YAW.LE.0.) GO TO 93
C     PROJECT DOWNSTREAM SIDE EDGE POINTS ON SINGULAR LINE
      I0         = ITE1(KTE2)   +1
      DO 92 I=I0,LX
      N          = N   +1
   92 Z0(N)      = SCALZ*C0(KTE2)   -TYAW*A0(I)*A0(I)
   93 I          = ITE1(KTE1)
      Z0(KTE1-1)   = SCALZ*C0(KTE1-1)   -TYAW*A0(I)*A0(I)
      Z0(N+1)      = SCALZ*C0(KTE2+1)
C     LOCATE POINTS JUST BEYOND EDGE OF WING OR VORTEX SHEET
      DO 102 K=2,NZ
      DO 102 I=2,NX
      IF (IV(I,K).GT.0) GO TO 102
      IF (IV(I+1,K+1).GT.0.OR.IV(I-1,K+1).GT.0) IV(I,K) = IV1
      IF (IV(I+1,K-1).GT.0.OR.IV(I-1,K-1).GT.0) IV(I,K) = IV1
  102 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  ESTIM
C     GENERATES INITIAL ESTIMATE OF POTENTIAL
C     SUCCESSIVE SLICES OF THE G ARRAY ,REPRESENTING VALUES OF POTENTIAL
C     ON X-Y PLANES AT SUCCESSIVE VALUES OF Z,ARE GENERATED
C     AND STORED ON TWO DISC FILES TO PROVIDE BACK UP
C     IN EVENT OF SUBSEQUENT DISC FAILURE
      COMMON      G(193,26,4),SEP1,
     1            A0(193),SEP2,A1(193),SEP3,A2(193),SEP4,A3(193),SEP5,
     2            B0(26),SEP6,B1(26),SEP7,B2(26),SEP8,B3(26),SEP9,
     3            C0(33),SEP10,C1(33),SEP11,C2(33),SEP12,C3(33),SEP13,
     4            S0(193,33),SEP14,E0(129),SEP15,Z0(129),SEP16,
     5            IV(193,33),SEP17,ITE1(33),SEP18,ITE2(33),SEP19,
     6            NX,NY,NZ,KTE1,KTE2,KSYM,SCAL,SCALZ,
     7            YAW,CYAW,SYAW,ALPHA,CA,SA,FMACH,N1,N2,N3,IO
      MX        = NX  +1
      KY        = NY  +1
      MY        = NY  +2
      MZ        = NZ  +1
C     SET THE G ARRAY TO ZERO
      DO 12 I=1,193
      DO 12 J=1,26
      DO 12 K=1,4
   12 G(I,J,K)  = 0.
      K         = 1
C     SET VALUES OF POTENTIAL AT DUMMY POINTS BEHIND BOUNDARY
   21 DO 22 I=2,NX
      G(I,KY+1,1) = 0.
      IF (IV(I,K).LT.2) GO TO 22
C     IV = 2 INDICATES POINT ON WING SURFACE
C     SET POTENTIAL BELOW SURFACE TO SATISFY BOUNDARY CONDITION
      DSI       = S0(I+1,K)  -S0(I-1,K)
      DSK       = S0(I,K+1)  -S0(I,K-1)
      SX        = A1(I)*DSI
      SZ        = C1(K)*DSK
      U         = CA*A0(I)  +SA*S0(I,K)
      W         = SYAW
      FH        = A0(I)*A0(I)  +S0(I,K)*S0(I,K)
      V         = B1(KY)*(1.  +SX*SX  +FH*SZ*SZ)
      G(I,KY+1,1) = G(I,KY-1,1)
     1              -(CA*S0(I,K)  -SA*A0(I)  +U*SX  +FH*W*SZ)/V
   22 CONTINUE
C     WRITE SLICE OF POTENTIAL ARRAY ON TWO DISC FILES
      BUFFER OUT(N3,1) (G(1,1,1),G(MX,MY,1))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N3).GT.0.) GO TO 41
      BUFFER OUT(N1,1) (G(1,1,1),G(MX,MY,1))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N1).GT.0.) GO TO 41
C     INCREMENT Z
      K         = K  +1
      IF (K.LE.MZ) GO TO 21
C     SET TRAILING JUMP E0 IN POTENTIAL TO ZERO
      K1        = KTE1  -1
      K2        = KTE2  +ITE2(KTE2)  -NX/2
      DO 32 K=K1,K2
```

```
   32 E0(K)      = 0.
C     SET IO TO INDICATE SUCCESSFUL COMPLETION
      IO         = 1
      RETURN
C     SET IO TO INDICATE DISC FAILURE
   41 IO         = 0
      RETURN
      END




      SUBROUTINE  MIXFLO
C     UPDATES POTENTIAL BY RELAXATION USING ROTATED DIFFERENCE SCHEME
C     EQUIVALENT TIME DEPENDENT EQUATION IS
C     (1.  -M**2)*GSS  +GMM  +GNN  +TERMS IN GST,GMT,GNT AND GT
C     SUCCESSIVE SLICES OF THE G ARRAY ,REPRESENTING VALUES OF POTENTIAL
C     ON X-Y PLANES AT SUCCESSIVE VALUES OF Z,ARE READ
C     FROM ONE DISC FILE,UPDATED,AND WRITTEN ON A SECOND DISC FILE
C     THREE SLICES ARE REQUIRED FOR COMPUTATION
C     A FOURTH SLICE IS USED AS A BUFFER FOR DISC OPERATIONS
C     INPUT AND OUTPUT BY BUFFER IN AND BUFFER OUT PROCEED IN PARALLEL
C     WITH COMPUTATION
C     IF THE BUFFER OPERATION IS NOT YET FINISHED,
C     THE IF UNIT TEST DOES NOT RETURN CONTROL TO THE CENTRAL PROCESSOR
C     UNTIL ITS COMPLETION,PREVENTING PREMATURE PROCESSING
      COMMON        G(193,26,4),SEP1,
     1              A0(193),SEP2,A1(193),SEP3.A2(193),SEP4,A3(193),SEP5,
     2              B0(26),SEP6,B1(26),SEP7,B2(26),SEP8,B3(26),SEP9,
     3              C0(33),SEP10,C1(33),SEP11,C2(33),SEP12,C3(33),SEP13,
     4              S0(193,33),SEP14,E0(129),SEP15,Z0(129),SEP16,
     5              IV(193,33),SEP17,ITE1(33).SEP18,ITE2(33),SEP19,
     6              NX,NY,NZ,KTE1,KTE2,KSYM,SCAL,SCALZ,
     7              YAW,CYAW,SYAW,ALPHA,CA,SA.FMACH,N1,N2,N3,IO
      COMMON/FLO/ GK1(193,26),BUF1,GK2(193,26),BUF2,
     1              SXX(193),BUF3,SXZ(193),BUF4,SZZ(193),BUF5,
     2              SX(193),BUF6,SZ(193),BUF7.R0(193),BUF8,R1(193),BUF9,
     3              C(193),BUF10,D(193),BUF11.G1(193),BUF12,G2(193),
     4              STRIP,P1,P2,P3,BETA,FR,IR,JR,KR,DG,IG,JG,KG,NS
      COMMON/SWP/ G10(26),SPA1,G20(26),SPA2.G30(26),SPA3,G40(26),
     1              I1,I2,K,L,N0,LX,MX,KY,MY,T1,AA0,Q1,Q2,Z,TYAW
      LX         = NX/2  +1
      MX         = NX  +1
      KY         = NY  +1
      MY         = NY  +2
      TYAW       = .5*SCAL*SYAW/CYAW
      DX         = 2./NX
      T1         = DX*DX
      AA0        = 1./FMACH**2  +.2
      Q1         = 2./P1
      Q2         = 1./P2
      FR         = 0.
      IR         = 0
      JR         = 0
      KR         = 0
```

```
      DG        = 0.
      IG        = 0
      JG        = 0
      KG        = 0
      NS        = 0
C     FR,IR,JR AND KR ARE VALUE AND LOCATION OF LARGEST RESIDUAL
C     DG,IG,JG AND KG ARE VALUE AND LOCATION OF LARGEST CORRECTION
C     NS IS NUMBER OF SUPERSONIC POINTS
C     START AT THIRD ROW IF FLOW IS SUPERSONIC AT INFINITY,
C     REQUIRING CAUCHY DATA
      K1        = 2
      IF (FMACH.GE.1.) K1 = 3
C     DEFINE CENTRAL STRIP OF X-Y PLANE FOR HORIZONTAL LINE RELAXATION
C     EXTENDING FROM I = I1 TO I = I2 WITH WIDTH DEFINED BY STRIP
C     STRIP = 0. ELIMINATES THE CENTRAL STRIP
C     STRIP = 1. ELIMINATES THE OUTER STRIPS
      F         = ABS(.5*STRIP*NX)
      L         = F
      IF (L.EQ.NX/2) L = L  -1
      I1        = LX  -L
      I2        = LX  +L
      IF (L.EQ.0) I2 = LX  -1
C     READ FIRST THREE SLICES OF POTENTIAL ARRAY FROM FIRST DISC FILE
      DO 2 L=1,3
      BUFFER IN (N1,1) (G(1,1,L),G(MX,MY,L))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N1).GT.0.) GO TO 101
    2 CONTINUE
C     SAVE OLD VALUES OF POTENTIAL AT UPSTREAM Z STATIONS
C     TO GENERATE CORRECT MIXED SPACE-TIME DERIVATIVES
      DO 4 J=1,MY
      DO 4 I=1,MX
      G(I,J,4)  = G(I,J,1)
      GK1(I,J)  = G(I,J,1)
    4 GK2(I,J)  = G(I,J,1)
      K         = 2
      L         = 2
      NO        = KTE1  -1
      IF (K.EQ.K1) GO TO 11
C     ADVANCE AN EXTRA SLICE IF THE FLOW IS SUPERSONIC AT INFINITY
C     WRITE FIRST SLICE OF UPDATED POTENTIAL ARRAY ON SECOND DISC FILE
      BUFFER OUT(N2,1) (G(1,1,4),G(MX,MY,4))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N2).GT.0.) GO TO 101
C     READ FOURTH SLICE OF POTENTIAL ARRAY FROM FIRST DISC FILE
      BUFFER IN (N1,1) (G(1,1,4),G(MX,MY,4))
      GO TO 51
C     WRITE SLICE OF UPDATED POTENTIAL ARRAY ON SECOND DISC FILE
   11 BUFFER OUT(N2,1) (G(1,1,4),G(MX,MY,4))
      Z         = SCALZ*CO(K)
      DO 12 J=1,MY
      G10(J)    = G(I2,J,2)
      G20(J)    = G(I2-1,J,2)
      G30(J)    = G(I1,J,2)
   12 G40(J)    = G(I1+1,J,2)
```

```
C     DETERMINE FIRST AND SECOND DERIVATIVES OF SURFACE SLOPE
C     FOR USE IN RELAXATION SUBROUTINES YSWEEP AND XSWEEP
      DO 22 I=2,NX
      DSI         = SO(I+1,K)  -SO(I-1,K)
      DSK         = SO(I,K+1)  -SO(I,K-1)
      DSII        = SO(I+1,K)  -SO(I,K)  -SO(I,K)  +SO(I-1,K)
     1              +A3(I)*DSI
      DSKK        = SO(I,K+1)  -SO(I,K)  -SO(I,K)  +SO(I,K-1)
     1              +C3(K)*DSK
      DSIK        = SO(I+1,K+1)  -SO(I-1,K+1)  -SO(I+1,K-1)  +SO(I-1,K-1)
      SX(I)       = A1(I)*DSI
      SZ(I)       = C1(K)*DSK
      SXX(I)      = A2(I)*DSII
      SZZ(I)      = C2(K)*DSKK
   22 SXZ(I)      = T1*A1(I)*C1(K)*DSIK
C     UPDATE THE CENTRAL STRIP BY HORIZONTAL LINE RELAXATION
      IF (I2.GT.I1) CALL YSWEEP
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N2).GT.0.) GO TO 101
C     READ SLICE OF POTENTIAL ARRAY FROM FIRST DISC FILE
      IF (K.LT.NZ) BUFFER IN (N1,1) (G(1,1,4),G(MX,MY,4))
C     UPDATE THE OUTER STRIPS BY VERTICAL LINE RELAXATION
      IF (I1.GT.2) CALL XSWEEP
      IF (K.NE.KTE2.OR.YAW.LE.0.) GO TO 51
C     DETERMINE NEW JUMP E0 IN POTENTIAL ALONG SIDE EDGE
C     OF DOWNSTREAM TIP
      IO          = ITE1(K)  +1
      DO 42 I=IO,LX
      M           = NX  +2  -I
      E           = G(M,KY,2)  -G(I,KY,2)
      NO          = NO  +1
   42 E0(NO)      = E0(NO)  +P3*(E  -E0(NO))
C     GIVE UP IN EVENT OF DISC FAILURE
   51 IF (UNIT(N1).GT.0.) GO TO 101
      IF (K.EQ.NZ) GO TO 61
C     SHIFT SLICES OF POTENTIAL ARRAY
      DO 52 J=1,MY
      DO 52 I=1,MX
      G(I,J,1)    = G(I,J,2)
      G(I,J,2)    = G(I,J,3)
      G(I,J,3)    = G(I,J,4)
   52 G(I,J,4)    = G(I,J,1)
C     INCREMENT Z
      K           = K  +1
      GO TO 11
C     WRITE LAST TWO SLICES OF POTENTIAL ARRAY ON SECOND DISC FILE
   61 DO 62 L=2,3
      BUFFER OUT(N2,1) (G(1,1,L),G(MX,MY,L))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N2).GT.0.) GO TO 101
   62 CONTINUE
      FR          = 1.2*FR/AA0
C     SET IO TO INDICATE SUCCESSFUL COMPLETION
      IO          = 1
      RETURN
```

```
C     SET IO TO INDICATE DISC FAILURE
  101 IO          = 0
      RETURN
      END




      SUBROUTINE  YSWEEP
C     ROW RELAXATION
      COMMON        G(193,26,4),SEP1,
     1              A0(193),SEP2,A1(193),SEP3,A2(193),SEP4,A3(193),SEP5,
     2              B0(26),SEP6,B1(26),SEP7,B2(26),SEP8,B3(26),SEP9,
     3              C0(33),SEP10,C1(33),SEP11,C2(33),SEP12,C3(33),SEP13,
     4              S0(193,33),SEP14,E0(129),SEP15,Z0(129),SEP16,
     5              IV(193,33),SEP17,ITE1(33),SEP18,ITE2(33),SEP19,
     6              NX,NY,NZ,KTE1,KTE2,KSYM,SCAL,SCALZ,
     7              YAW,CYAW,SYAW,ALPHA,CA,SA,FMACH,N1,N2,N3,IO
      COMMON/FLO/ GK1(193,26),BUF1,GK2(193,26),BUF2,
     1              SXX(193),BUF3,SXZ(193),BUF4,SZZ(193),BUF5,
     2              SX(193),BUF6,SZ(193),BUF7,R0(193),BUF8,R1(193),BUF9,
     3              C(193),BUF10,D(193),BUF11,G1(193),BUF12,G2(193),
     4              STRIP,P1,P2,P3,BETA,FR,IR,JR,KR,DG,IG,JG,KG,NS
      COMMON/SWP/ G10(26),SPA1,G20(26),SPA2,G30(26),SPA3,G40(26),
     1              I1,I2,K,L,N0,LX,MX,KY,MY,T1,AA0,Q1,Q2,Z,TYAW
      J1          = 2
      IF (FMACH.GE.1.) J1 = 3
      C(I1-1)     = 0.
      D(I1-1)     = 0.
      DO 12 I=I1,I2
      R0(I)       = 1.
      R1(I)       = 1.
      G1(I)       = G(I,J1-1,L)
   12 G2(I)       = G(I,1,L)
      J           = J1
      I3          = I2
   31 BC          = -T1*B1(J)*C1(K)
      DO 32 I=I1,I3
      AB          = -T1*A1(I)*B1(J)
      AC          = T1*A1(I)*C1(K)
      YP          = S0(I,K)   +B0(J)
      A           = 1.   -R0(I)   +A0(I)*A0(I)   +YP*YP
      H           = R0(I)/A
      FH          = R0(I)*A
      DGI         = G(I+1,J,L)   -G(I-1,J,L)
      DGJ         = G(I,J+1,L)   -G1(I)
      DGK         = G(I,J,L+1)   -GK1(I,J)
      DGII        = G(I+1,J,L)   -G(I,J,L)   -G(I,J,L)   +G(I-1,J,L)
     1              +A3(I)*DGI
      DGJJ        = G(I,J+1,L)   -G(I,J,L)   -G(I,J,L)   +G(I,J-1,L)
     1              -B3(J)*DGJ
      DGKK        = G(I,J,L+1)   -G(I,J,L)   -G(I,J,L)   +G(I,J,L-1)
     1              +C3(K)*DGK
      DGIJ        = G(I+1,J+1,L)   -G(I-1,J+1,L)
     1              -G(I+1,J-1,L)   +G(I-1,J-1,L)
```

```
      DGIK        = G(I+1,J,L+1)  -G(I+1,J,L-1)
     1              -G(I-1,J,L+1)  +G(I-1,J,L-1)
      DGJK        = G(I,J+1,L+1)  -G(I,J-1,L+1)
     1              -G(I,J+1,L-1)  +G(I,J-1,L-1)
      GY          = -B1(J)*DGJ
      U           = A1(I)*DGI  -SX(I)*GY  +CA*A0(I)  +SA*YP
      V           = GY  +SA*A0(I)  -CA*YP
      W           = R0(I)*(C1(K)*DGK  -SZ(I)*GY  +SYAW)
      QXY         = H*(U*U  +V*V)
      QQ          = QXY  +W*W
      AA          = DIM(AA0,.2*QQ)
      HZ          = FH*SZ(I)
      F           = 1.  +SX(I)*SX(I)  +SZ(I)*HZ
      AV          = V  -U*SX(I)  -W*HZ
      UU          = H*U*U
      VV          = H*AV*AV
      WW          = FH*W*W
      UV          = H*U*AV
      VW          = AV*W
      UW          = U*W
      AXX         = R1(I)*(AA  -UU)
      AZZ         = FH*AA  -WW
      R           = -(AXX*SXX(I)  +AZZ*SZZ(I)  -(UW  +UW)*SXZ(I))*GY
     1              -T1*H*(CA*(U*U  -V*V)  +(SA  +SA)*U*V
     2                    -QXY*(U*A0(I)  +V*YP))
      AXT         = ABS(U*A1(I))
      AYT         = ABS(AV*B1(J))
      AZT         = ABS(FH*W*C1(K))
      A           = R0(I)*BETA*AA/AMAX1(AXT,AYT,AZT,(1.  -R0(I)))
      AXT         = A*AXT
      AYT         = A*AYT
      AZT         = A*AZT
      IF (QQ.GE.AA) GO TO 33
      AXX         = AXX*A2(I)
      AYY         = (F*AA  -VV)*B2(J)
      AZZ         = AZZ*C2(K)
      AXY         = -R1(I)*(AA*SX(I)  +UV)*(AB  +AB)
      AYZ         = -(AA*HZ  +VW)*(BC  +BC)
      AXZ         = -UW*(AC  +AC)
      BP          = AXX
      BM          = AXX
      B           = -AXX  -AXX  -Q1*(AYY  +AZZ)
      R           = AXX*DGII  +AYY*DGJJ  +AZZ*DGKK
     1              +AXY*DGIJ  +AYZ*DGJK  +AXZ*DGIK  +R
      GO TO 35
   33 NS          = NS  +1
      S           = SIGN(1.,U)
      II          = S
      IM          = I  -II
      IMM         = IM  -II
      AXX         = UU*A2(I)
      AYY         = VV*B2(J)
      AZZ         = WW*C2(K)
      AXY         = 8.*S*UV*AB
      AYZ         = 8.*VW*BC
```

```
      AXZ           = 8.*S*UW*AC
      BXX           = (QQ   -UU)*A2(I)
      BYY           = (F*QQ   -VV)*B2(J)
      BZZ           = (FH*QQ   -WW)*C2(K)
      BXY           = -(QQ*SX(I)   +UV)*(AB   +AB)
      BYZ           = -(QQ*HZ   +VW)*(BC   +BC)
      BXZ           = -UW*(AC   +AC)
      AQ            = AA/QQ
      DELTAG        = BXX*DGII   +BYY*DGJJ   +BZZ*DGKK
     1                +BXY*DGIJ   +BYZ*DGJK   +BXZ*DGIK
      DGII          = G(I,J,L)   -G(IM,J,L)   -G(IM,J,L)   +G(IMM,J,L)
     1                +A3(I)*DGI
      DGJJ          = G(I,J,L)   -G(I,J-1,L)   -G(I,J-1,L)   +G2(I)
     1                -B3(J)*DGJ
      DGKK          = G(I,J,L)   -G(I,J,L-1)   -G(I,J,L-1)   +GK2(I,J)
     1                +C3(K)*DGK
      DGIJ          = G(I,J,L)   -G(IM,J,L)
     1                -G(I,J-1,L)   +G(IM,J-1,L)
      DGIK          = G(I,J,L)   -G(I,J,L-1)
     1                -G(IM,J,L)   +G(IM,J,L-1)
      DGJK          = G(I,J,L)   -G(I,J,L-1)
     1                -G(I,J-1,L)   +G(I,J-1,L-1)
      GSS           = AXX*DGII   +AYY*DGJJ   +AZZ*DGKK
     1                +AXY*DGIJ   +AYZ*DGJK   +AXZ*DGIK
      B             = .5*(AQ   -1.)*(AXX   +AXX   +AXY   +AXZ)
      BP            = AQ*BXX   -(1.   -S)*B
      BM            = AQ*BXX   -(1.   +S)*B
      B             = -AQ*(BXX   +BXX   +Q2*(BYY   +BZZ))
     1                +(AQ   -1.)*(2.*(AXX   +AYY   +AZZ)   +AXY   +AYZ   +AXZ)
      R             = (AQ   -1.)*GSS   +AQ*DELTAG   +R
   35 IF (ABS(R).LE.ABS(FR)) GO TO 37
      FR            = R
      IR            = I
      JR            = J
      KR            = K
   37 R             = R   -AYT*(G1(I)   -G(I,J-1,L))
     1                -AZT*(GK1(I,J)   -G(I,J,L-1))
      B             = B   -AXT   -AYT   -AZT
      BM            = BM   +AXT
      B             = 1./(B   -BM*C(I-1))
      C(I)          = B*BP
   32 D(I)          = B*(R   -BM*D(I-1))
      CG            = 0.
      I             = I3
      DO 42 M=I1,I3
      CG            = D(I)   -C(I)*CG
      IF (ABS(CG).LE.ABS(DG)) GO TO 43
      DG            = CG
      IG            = I
      JG            = J
      KG            = K
   43 G2(I)         = G1(I)
      G1(I)         = G(I,J,L)
      GK2(I,J)      = GK1(I,J)
      GK1(I,J)      = G(I,J,L)
```

```
      G(I,J,L)   = G(I,J,L)   -CG
   42 I          = I  -1
      J          = J  +1
      IF (J  -KY) 31,51,61
   51 IF (I2.GT.ITE2(K)) I3 = ITE2(K)
      IF (ITE2(K).EQ.MX) I3 = LX
      DO 52 I=I1,I3
      LV         = IABS(1  -IABS(IV(I,K)))
      R0(I)      = AMIN0(LV,IABS(IV(I,K)))
   52 R1(I)      = LV
      GO TO 31
   61 N          = N0
      I          = LX  +1
      IF (K.LT.KTE1.OR.K.GT.KTE2) GO TO 71
      I0         = NX  +2  -I3
      DO 62 I=I0,I3
      DGI        = G(I+1,KY,L)  -G(I-1,KY,L)
      DGK        = G(I,KY,L+1)  -GK2(I,KY)
      U          = A1(I)*DGI  +CA*A0(I)  +SA*S0(I,K)
      W          = C1(K)*DGK  +SYAW
      FH         = A0(I)*A0(I)  +S0(I,K)*S0(I,K)
      V          = B1(KY)*(1.  +SX(I)*SX(I)  +FH*SZ(I)*SZ(I))
   62 G(I,KY+1,L) = G(I,KY-1,L)
     1              -(CA*S0(I,K)  -SA*A0(I)  +U*SX(I)  +FH*W*SZ(I))/V
      I          = I0
      IF (I0.NE.ITE1(K)) GO TO 71
      E          = G(I3,KY,L)  -G(I0,KY,L)
      N0         = N0  +1
      E0(N0)     = E0(N0)  +P3*(E  -E0(N0))
      N          = N0
   71 IF (I.LE.I1) RETURN
      I          = I  -1
      E          = 0.
      IF (IV(I,K).NE.1) GO TO 77
      ZZ         = Z  -TYAW*A0(I)*A0(I)
   73 IF (ZZ.GE.Z0(N-1)) GO TO 75
      N          = N  -1
      GO TO 73
   75 R          = (ZZ  -Z0(N-1))/(Z0(N)  -Z0(N-1))
      E          = R*E0(N)  +(1.  -R)*E0(N-1)
   77 M          = NX  +2  -I
      G(I,KY+1,L) = G(M,KY-1,L)  -E
      G(M,KY+1,L) = G(I,KY-1,L)  +E
      GK2(M,KY)   = GK1(M,KY)
      GK1(M,KY)   = G(M,KY,L)
      G(M,KY,L)   = G(I,KY,L)  +E
      GO TO 71
      END


      SUBROUTINE  XSWEEP
C     COLUMN RELAXATION
      COMMON      G(193,26,4),SEP1,
```

```
     1          A0(193),SEP2,A1(193),SEP3,A2(193),SEP4,A3(193),SEP5,
     2          B0(26),SEP6,B1(26),SEP7,B2(26),SEP8,B3(26),SEP9,
     3          C0(33),SEP10,C1(33),SEP11,C2(33),SEP12,C3(33),SEP13,
     4          S0(193,33),SEP14,E0(129),SEP15,Z0(129),SEP16,
     5          IV(193,33),SEP17,ITE1(33),SEP18,ITE2(33),SEP19,
     6          NX,NY,NZ,KTE1,KTE2,KSYM,SCAL,SCALZ,
     7          YAW,CYAW,SYAW,ALPHA,CA,SA,FMACH,N1,N2,N3,IO
      COMMON/FLO/ GK1(193,26),BUF1,GK2(193,26),BUF2,
     1          SXX(193),BUF3,SXZ(193),BUF4,SZZ(193),BUF5,
     2          SX(193),BUF6,SZ(193),BUF7,R0(193),BUF8,R1(193),BUF9,
     3          C(193),BUF10,D(193),BUF11,G1(193),BUF12,G2(193),
     4          STRIP,P1,P2,P3,BETA,FR,IR,JR,KR,DG,IG,JG,KG,NS
      COMMON/SWP/ G10(26),SPA1,G20(26),SPA2,G30(26),SPA3,G40(26),
     1          I1,I2,K,L,N0,LX,MX,KY,MY,T1,AA0,Q1,Q2,Z,TYAW
      N         = N0
      J1        = 2
      IF (FMACH.GE.1.) J1 = 3
      C(J1-1)   = 0.
      D(J1-1)   = 0.
      S         = 1.
      II        = 1
      I         = I2  +1
      DO 12 J=2,KY
      R0(J)     = 1.
      R1(J)     = 1.
      G1(J)     = G10(J)
   12 G2(J)     = G20(J)
   21 IP        = I  +II
      IM        = I  -II
      J2        = KY
      IF (IV(I,K).LT.2.AND.I.GT.LX) J2 = NY
      LV        = IABS(1  -IABS(IV(I,K)))
      R0(KY)    = AMIN0(LV,IABS(IV(I,K)))
      R1(KY)    = LV
      AC        = T1*A1(I)*C1(K)
      DO 32 J=J1,J2
      AB        = -T1*A1(I)*B1(J)
      BC        = -T1*B1(J)*C1(K)
      YP        = S0(I,K)  +B0(J)
      A         = 1.  -R0(J)  +A0(I)*A0(I)  +YP*YP
      H         = R0(J)/A
      FH        = R0(J)*A
      DGI       = S*(G(IP,J,L)  -G1(J))
      DGJ       = G(I,J+1,L)  -G(I,J-1,L)
      DGK       = G(I,J,L+1)  -GK1(I,J)
      DGII      = G(I+1,J,L)  -G(I,J,L)  -G(I,J,L)  +G(I-1,J,L)
     1            +A3(I)*DGI
      DGJJ      = G(I,J+1,L)  -G(I,J,L)  -G(I,J,L)  +G(I,J-1,L)
     1            -B3(J)*DGJ
      DGKK      = G(I,J,L+1)  -G(I,J,L)  -G(I,J,L)  +G(I,J,L-1)
     1            +C3(K)*DGK
      DGIJ      = G(I+1,J+1,L)  -G(I-1,J+1,L)
     1            -G(I+1,J-1,L)  +G(I-1,J-1,L)
      DGIK      = G(I+1,J,L+1)  -G(I+1,J,L-1)
     1            -G(I-1,J,L+1)  +G(I-1,J,L-1)
```

```
      DGJK       = G(I,J+I,L+1)  -G(I,J-1,L+1)
     1             -G(I,J+1,L-1)  +G(I,J-1,L-1)
      GY         = -B1(J)*DGJ
      U          = A1(I)*DGI  -SX(I)*GY  +CA*A0(I)  +SA*YP
      V          = GY  +SA*A0(I)  -CA*YP
      W          = R0(J)*(C1(K)*DGK  -SZ(I)*GY  +SYAW)
      QXY        = H*(U*U  +V*V)
      QQ         = QXY  +W*W
      AA         = DIM(AA0,.2*QQ)
      HZ         = FH*SZ(I)
      F          = 1.  +SX(I)*SX(I)  +SZ(I)*HZ
      AV         = V  -U*SX(I)  -W*HZ
      UU         = H*U*U
      VV         = H*AV*AV
      WW         = FH*W*W
      UV         = H*U*AV
      VW         = AV*W
      UW         = U*W
      AXX        = R1(J)*(AA  -UU)
      AZZ        = FH*AA  -WW
      R          = -(AXX*SXX(I)  +AZZ*SZZ(I)  -(UW  +UW)*SXZ(I))*GY
     1             -T1*H*(CA*(U*U  -V*V)  +(SA  +SA)*U*V
     2                   -QXY*(U*A0(I)  +V*YP))
      AXT        = ABS(U*A1(I))
      AYT        = ABS(AV*B1(J))
      AZT        = ABS(FH*W*C1(K))
      A          = R0(J)*BETA*AA/AMAX1(AXT,AYT,AZT,(1.  -R0(J)))
      AXT        = A*AXT
      AYT        = A*AYT
      AZT        = A*AZT
      IF (QQ.GE.AA) GO TO 33
      AXX        = AXX*A2(I)
      AYY        = (F*AA  -VV)*B2(J)
      AZZ        = AZZ*C2(K)
      AXY        = -R1(J)*(AA*SX(I)  +UV)*(AB  +AB)
      AYZ        = -(AA*HZ  +VW)*(BC  +BC)
      AXZ        = -UW*(AC  +AC)
      BP         = AYY
      BM         = AYY
      B          = -AYY  -AYY  -Q1*(AXX  +AZZ)
      R          = AXX*DGII  +AYY*DGJJ  +AZZ*DGKK
     1             +AXY*DGIJ  +AYZ*DGJK  +AXZ*DGIK  +R
      GO TO 35
   33 NS         = NS  +1
      AXX        = UU*A2(I)
      AYY        = VV*B2(J)
      AZZ        = WW*C2(K)
      AXY        = 8.*S*UV*AB
      AYZ        = 8.*VW*BC
      AXZ        = 8.*S*UW*AC
      BXX        = (QQ  -UU)*A2(I)
      BYY        = (F*QQ  -VV)*B2(J)
      BZZ        = (FH*QQ  -WW)*C2(K)
      BXY        = -(QQ*SX(I)  +UV)*(AB  +AB)
      BYZ        = -(QQ*HZ  +VW)*(BC  +BC)
```

```
      BXZ          = -UW*(AC   +AC)
      AQ           = AA/QQ
      DELTAG       = BXX*DGII   +BYY*DGJJ   +BZZ*DGKK
     1               +BXY*DGIJ   +BYZ*DGJK   +BXZ*DGIK
      DGII         = G(I,J,L)   -G(IM,J,L)   -G(IM,J,L)   +G2(J)
     1               +A3(I)*UGI
      DGJJ         = G(I,J,L)   -G(I,J-1,L)   -G(I,J-1,L)   +G(I,J-2,L)
     1               -B3(J)*DGJ
      DGKK         = G(I,J,L)   -G(I,J,L-1)   -G(I,J,L-1)   +GK2(I,J)
     1               +C3(K)*DGK
      DGIJ         = G(I,J,L)   -G(IM,J,L)
     1               -G(I,J-1,L)   +G(IM,J-1,L)
      DGIK         = G(I,J,L)   -G(I,J,L-1)
     1               -G(IM,J,L)   +G(IM,J,L-1)
      DGJK         = G(I,J,L)   -G(I,J,L-1)
     1               -G(I,J-1,L)   +G(I,J-1,L-1)
      GSS          = AXX*DGII   +AYY*DGJJ   +AZZ*DGKK
     1               +AXY*DGIJ   +AYZ*DGJK   +AXZ*DGIK
      BP           = AQ*BYY
      BM           = BP   -(AQ   -1.)*(AYY   +AYY   +AXY   +AYZ)
      B            = -BP   -BP   -Q2*AQ*(BXX   +BZZ)
     1               +(AQ   -1.)*(2.*(AXX   +AYY   +AZZ)   +AXY   +AYZ   +AXZ)
      R            = (AQ   -1.)*GSS   +AQ*DELTAG   +R
   35 IF (ABS(R).LE.ABS(FR)) GO TO 37
      FR           = R
      IR           = I
      JR           = J
      KR           = K
   37 R            = R   -AXT*(G1(J)   -G(IM,J,L))
     1               -AZT*(GK1(I,J)   -G(I,J,L-1))
      B            = B   -AXT   -AYT   -AZT
      BM           = BM   +AYT
      B            = 1./(B   -BM*C(J-1))
      C(J)         = B*BP
   32 D(J)         = B*(R   -BM*D(J-1))
      CG           = 0.
      J            = J2
      DO 42 M=J1,J2
      CG           = D(J)   -C(J)*CG
      IF (ABS(CG).LE.ABS(DG)) GO TO 43
      DG           = CG
      IG           = I
      JG           = J
      KG           = K
   43 G2(J)        = G1(J)
      G1(J)        = G(I,J,L)
      GK2(I,J)     = GK1(I,J)
      GK1(I,J)     = G(I,J,L)
      G(I,J,L)     = G(I,J,L)   -CG
   42 J            = J   -1
      IF (IV(I,K).LT.2) GO TO 51
      DGI          = S*(G(IP,KY,L)   -G2(KY))
      DGK          = G(I,KY,L+1)   -GK2(I,KY)
      U            = A1(I)*DGI   +CA*A0(I)   +SA*S0(I,K)
      W            = C1(K)*DGK   +SYAW
```

```
      FH          = A0(I)*A0(I)  +S0(I,K)*S0(I,K)
      V           = B1(KY)*(1.  +SX(I)*SX(I)  +FH*SZ(I)*SZ(I))
      G(I,KY+1,L) = G(I,KY-1,L)
     1              -(CA*SU(I,K)  -SA*A0(I)  +U*SX(I)  +FH*W*SZ(I))/V
      IF (I.NE.ITE1(K)) GO TO 61
      M           = NX  +2  -I
      E           = G(M,KY,L)  -G(I,KY,L)
      NO          = NO  +1
      E0(NO)      = E0(NO)  +P3*(E  -E0(NO))
      N           = NO
      GO TO 61
   51 IF (I.GT.LX) GO TO 61
      E           = 0.
      IF (IV(I,K).NE.1) GO TO 57
      ZZ          = Z  -TYAW*A0(I)*A0(I)
   53 IF (ZZ.GE.Z0(N-1)) GO TO 55
      N           = N  -1
      GO TO 53
   55 R           = (ZZ  -Z0(N-1))/(Z0(N)  -Z0(N-1))
      E           = R*E0(N)  +(1.  -R)*E0(N-1)
   57 M           = NX  +2  -I
      G(I,KY+1,L) = G(M,KY-1,L)  -E
      G(M,KY+1,L) = G(I,KY-1,L)  +E
      GK2(M,KY)   = GK1(M,KY)
      GK1(M,KY)   = G(M,KY,L)
      G(M,KY,L)   = G(I,KY,L)  +E
   61 IF (I.EQ.NX) GO TO 71
      IF (I.EQ.2) RETURN
      I           = I  +II
      GO TO 21
   71 S           = -1.
      II          = -1
      I           = I1  -1
      DO 72 J=2,KY
      G1(J)       = G30(J)
   72 G2(J)       = G40(J)
      GO TO 21
      END




      SUBROUTINE  VELO  (K,L,SV,SM,CP,X,Y)
C     CALCULATES SURFACE SPEED SV,MACH NUMBER SM,PRESSURE COEFFICIENT CP
C     AND COORDINATES X,Y AT SPAN STATION K
      COMMON      G(193,26,4),SEP1,
     1            A0(193),SEP2,A1(193),SEP3,A2(193),SEP4,A3(193),SEP5,
     2            B0(26),SEP6,B1(26),SEP7,B2(26),SEP8,B3(26),SEP9,
     3            C0(33),SEP10,C1(33),SEP11,C2(33),SEP12,C3(33),SEP13,
     4            S0(193,33),SEP14,E0(129),SEP15,Z0(129),SEP16,
     5            IV(193,33),SEP17,ITE1(33),SEP18,ITE2(33),SEP19,
     6            NX,NY,NZ,KTE1,KTE2,KSYM,SCAL,SCALZ,
     7            YAW,CYAW,SYAW,ALPHA,CA,SA,FMACH,N1,N2,N3,IO
      DIMENSION   SV(1),SM(1),CP(1),X(1),Y(1)
C     ITE1 AND ITE2 ARE LOWER AND UPPER TRAILING EDGE POINTS
```

```
      I1          = ITE1(K)
      I2          = ITE2(K)
C     SURFACE INDEX IS NY +1
      J           = NY  +1
      Q1          = .2*FMACH**2
      T1          = 1./(.7*FMACH**2)
      DO 12 I=I1,I2
C     DETERMINE MULTIPLIER H OF SQUARE ROOT TRANSFORMATION
      YP          = S0(I,K)   +B0(J)
      H           = A0(I)*A0(I)   +YP*YP
C     DETERMINE SLOPES SX,SZ OF SHEARING TRANSFORMATION
      DSI         = S0(I+1,K)   -S0(I-1,K)
      DSK         = S0(I,K+1)   -S0(I,K-1)
      SX          = A1(I)*DSI
      SZ          = C1(K)*DSK
C     DETERMINE FIRST DIFFERENCES OF POTENTIAL
      DGI         = G(I+1,J,L)   -G(I-1,J,L)
      DGJ         = G(I,J+1,L)   -G(I,J-1,L)
      DGK         = G(I,J,L+1)   -G(I,J,L-1)
C     DETERMINE VELOCITY COMPONENTS U,V,W
      U           = A1(I)*DGI   +SX*B1(J)*DGJ   +CA*A0(I)   +SA*YP
      V           = -B1(J)*DGJ   +SA*A0(I)   -CA*YP
      W           = C1(K)*DGK   +SZ*B1(J)*DGJ   +SYAW
C     DETERMINE SURFACE SPEED SV,MACH NUMBER SM,PRESSURE COEFFICIENT CP
      QQ          = 0.
      IF (H.GT.1.E-6) QQ = (U*U   +V*V)/H   +W*W
      Q           = SQRT(QQ)
      IF (U.LT.0.) Q = -Q
      SV(I)       = Q
      QQ          = 1.   +Q1*(1.   -QQ)
      SM(I)       = FMACH*Q/SQRT(QQ)
      CP(I)       = T1*(QQ**3.5   -1.)
C     DETERMINE SURFACE COORDINATES X,Y
      X(I)        = .5*SCAL*(A0(I)**2   -S0(I,K)**2)
   12 Y(I)        = SCAL*A0(I)*S0(I,K)
      RETURN
      END



      SUBROUTINE  CPLOT (I1,I2,FMACH,X,Y,CP)
C     PLOTS CP AT EQUAL INTERVALS IN THE MAPPED PLANE
      DIMENSION   KODE(2),LINE(100),X(1),Y(1),CP(1)
      DATA        KODE/1H ,1H+/
      IWRIT       = 6
      WRITE (IWRIT,2)
    2 FORMAT(50H0PLOT OF CP AT EQUAL INTERVALS IN THE MAPPED PLANE/
     1       10H0      X    ,10H       Y    ,10H      CP  )
      CP0         = ((1.   +.2*FMACH**2)**3.5   -1.)/(.7*FMACH**2)
C     CP0 IS STAGNATION PRESSURE COEFFICIENT
C     SET LINE TO BLANK SYMBOLS
      DO 12 I=1,100
   12 LINE(I)     = KODE(1)
      DO 22 I=I1,I2
```

```
C     SET K PROPORTIONAL TO CP
      K          = 30.*(CP0   -CP(I))   +4.5
C     SET ELEMENT K OF LINE TO + SYMBOL
      LINE(K)    = KODE(2)
      WRITE (IWRIT,610) X(I),Y(I),CP(I),LINE
   22 LINE(K)    = KODE(1)
      RETURN
  610 FORMAT(3F10.4,100A1)
      END



      SUBROUTINE  FORCF (I1,I2,X,Y,CP,AL,CHORD,XM,CL,CD,CM)
C     CALCULATES SECTION LIFT,DRAG AND MOMENT COEFFICIENTS
C     BY TRAPEZOIDAL INTEGRATION OF SURFACE PRESSURE
      DIMENSION    X(1),Y(1),CP(1)
      RAD        = 57.2957795130823
      ALPHA      = AL/RAD
      CL         = 0.
      CD         = 0.
      CM         = 0.
      N          = I2  -1
      DO 12 I=I1,N
      DX         = (X(I+1)   -X(I))/CHORD
      DY         = (Y(I+1)   -Y(I))/CHORD
      XA         = (.5*(X(I+1)   +X(I))   -XM)/CHORD
      YA         = .5*(Y(I+1)   +Y(I))/CHORD
      CPA        = .5*(CP(I+1)   +CP(I))
      DCL        = -CPA*DX
      DCD        = CPA*DY
      CL         = CL   +DCL
      CD         = CD   +DCD
   12 CM         = CM   +DCD*YA   -DCL*XA
C     ROTATE CL AND CD TO DIRECTION OF FREE STREAM
      DCL        = CL*COS(ALPHA)   -CD*SIN(ALPHA)
      CD         = CL*SIN(ALPHA)   +CD*COS(ALPHA)
      CL         = DCL
      RETURN
      END



      SUBROUTINE  TOTFOR(KTE1,KTE2,CHORD,SCL,SCD,SCM,C0,SCALZ,XM,
     1                   CL,CD,CMP,CMR,CMY)
C     CALCULATES TOTAL LIFT,DRAG AND MOMENT COEFFICIENTS
C     IN DIRECTION NORMAL TO LEADING EDGE
C     BY TRAPEZOIDAL INTEGRATION OF SECTION FORCE COEFFICIENTS
C     SPANWISE FORCE IS NOT CALCULATED
C     CMP IS PITCHING MOMENT COEFFICIENT REFERRED TO MEAN CHORD
C     CMR IS ROLLING MOMENT COEFFICIENT REFERRED TO SEMI-SPAN
C     CMY IS YAWING MOMENT COEFFICIENT REFERRED TO SEMI-SPAN
      DIMENSION    CHORD(1),SCL(1),SCD(1),SCM(1),C0(1)
      SPAN       = SCALZ*(C0(KTE2)   -C0(KTE1))
```

```
      CL         = 0.
      CD         = 0.
      CMP        = 0.
      CMR        = 0.
      CMY        = 0.
      S          = 0.
      N          = KTE2  -1
      DO 12 K=KTE1,N
      DZ         = .5*SCALZ*(C0(K+1)  -C0(K))
      Z          = .5*SCALZ*(C0(K+1)  +C0(K))
      CL         = CL  +DZ*(SCL(K+1)*CHORD(K+1)  +SCL(K)*CHORD(K))
      CD         = CD  +DZ*(SCD(K+1)*CHORD(K+1)  +SCD(K)*CHORD(K))
      CMP        = CMP  +DZ*(SCM(K+1)*CHORD(K+1)**2  +SCM(K)*CHORD(K)**2)
      CMR        = CMR  +Z*DZ*(SCL(K+1)*CHORD(K+1)  +SCL(K)*CHORD(K))
      CMY        = CMY  +Z*DZ*(SCD(K+1)*CHORD(K+1)  +SCD(K)*CHORD(K))
   12 S          = S  +DZ*(CHORD(K+1)  +CHORD(K))
      CL         = CL/S
      CD         = CD/S
      CMP        = CMP*SPAN/S**2  +XM*CL
      CMR        = (CMR  +CMR)/(S*SPAN)
      CMY        = (CMY  +CMY)/(S*SPAN)
      RETURN
      END



      SUBROUTINE  CHARTY
C     GENERATES MACH NO CHARTS IN PLANE OF WING PLANFORM
      COMMON        G(193,26,4),SEP1,
     1              A0(193),SEP2,A1(193),SEP3,A2(193),SEP4,A3(193),SEP5,
     2              B0(26),SEP6,B1(26),SEP7,B2(26),SEP8,B3(26),SEP9,
     3              C0(33),SEP10,C1(33),SEP11,C2(33),SEP12,C3(33),SEP13,
     4              S0(193,33),SEP14,E0(129),SEP15,Z0(129),SEP16,
     5              IV(193,33),SEP17,ITE1(33),SEP18,ITE2(33),SEP19,
     6              NX,NY,NZ,KTE1,KTE2,KSYM,SCAL,SCALZ,
     7              YAW,CYAW,SYAW,ALPHA,CA,SA,FMACH,N1,N2,N3,IO
      DIMENSION     LV(33)
      IWRIT      = 6
      LX         = NX/2  +1
      MX         = NX  +1
      KY         = NY  +1
      MY         = NY  +2
      Q1         = .2*FMACH**2
      DO 2 K=2,NZ
      LV(K)      = MY
      IF (IV(LX,K).LE.0) LV(K) = KY
    2 CONTINUE
      WRITE (IWRIT,12)
   12 FORMAT(42H0UPPER SURFACE MACH NO CHART IN WING PLANE)
      LI         = 1
      IM         = NX
   21 DO 22 L=1,3
      BUFFER IN (N1,1) (G(1,1,L),G(MX,MY,L))
      IF (UNIT(N1).GT.0.) GO TO 101
```

```
22    CONTINUE
      K           = 1
31    K           = K  +1
      L           = LV(K)
      N           = 1
      II          = NX/2  +1
      KI          = 0
      JJ          = 2
      KJ          = 1
33    I           = II
      J           = JJ
      IF (J.LT.L) GO TO 35
      J           = NY
35    YP          = SO(I,K)  +B0(J)
      H           = A0(I)*A0(I)  +YP*YP
      DSI         = SO(I+1,K)  -SO(I-1,K)
      DSK         = SO(I,K+1)  -SO(I,K-1)
      SX          = A1(I)*DSI
      SZ          = C1(K)*DSK
      DGI         = G(I+1,J,2)  -G(I-1,J,2)
      DGJ         = G(I,J+1,2)  -G(I,J-1,2)
      DGK         = G(I,J,3)  -G(I,J,1)
      IF (J.LT.L) GO TO 37
      M           = NX  +2  -I
      DGI         = .5*(DGI  +G(M-1,J,2)  -G(M+1,J,2))
      DGJ         = .5*(DGJ  +G(M,J-1,2)  -G(M,J+1,2))
      DGK         = .5*(DGK  +G(M,J,3)  -G(M,J,1))
37    U           = A1(I)*DGI  +SX*B1(J)*DGJ  +CA*A0(I)  +SA*YP
      V           = -B1(J)*DGJ  +SA*A0(I)  -CA*YP
      W           = C1(K)*DGK  +SZ*B1(J)*DGJ  +SYAW
      QQ          = (U*U  +V*V)/H  +W*W
      F           = 1.  +Q1*(1.  -QQ)
      Q           = 0.
      IF (F.GT.0.) Q = SQRT(QQ/F)
      IF (LI*(I  -II)  +J  -JJ) 41,45,43
41    Q0          = Q
      I           = I  +LI
      J           = KY
      GO TO 35
43    Q           = .5*(Q  +Q0)
      L           = MY
45    N           = N  +1
      IV(N,K)     = 100.*FMACH*Q
      IF (II.EQ.IM) GO TO 51
      IF (JJ.NE.KY) GO TO 47
      KI          = LI
      KJ          = 0
47    II          = II  +KI
      JJ          = JJ  +KJ
      GO TO 33
51    IF (K.EQ.NZ) GO TO 61
      DO 52 I=1,MX
      DO 52 J=1,MY
      G(I,J,1)    = G(I,J,2)
52    G(I,J,2)    = G(I,J,3)
```

```
      BUFFER IN (N1,1) (G(1,1,3),G(MX,MY,3))
      IF (UNIT(N1).GT.0.) GO TO 101
      GO TO 31
   61 DO 62 I=2,N
   62 WRITE (IWRIT,610) (IV(I,K),K=2,NZ)
      IF (LI.EQ.-1) GO TO 91
      REWIND N1
      LI          = -1
      IM          = 2
      WRITE (IWRIT,600)
      WRITE (IWRIT,72)
   72 FORMAT(42H0LOWER SURFACE MACH NO CHART IN WING PLANE)
      GO TO 21
   91 IO          = 1
      RETURN
  101 IO          = 0
      RETURN
  600 FORMAT(1H1)
  610 FORMAT(1X,32I4)
      END



      SUBROUTINE  REFIN
C     INTERPOLATES POTENTIAL AT MESH POINTS OF REFINED GRID
C     SUCCESSIVE SLICES OF THE G ARRAY ,REPRESENTING VALUES OF POTENTIAL
C     ON X-Y PLANES AT SUCCESSIVE VALUES OF Z,ARE READ
C     FROM ONE DISC FILE,UPDATED,AND WRITTEN ON A SECOND DISC FILE
      COMMON      G(193,26,4),SEP1,
     1            A0(193),SEP2,A1(193),SEP3,A2(193),SEP4,A3(193),SEP5,
     2            B0(26),SEP6,B1(26),SEP7,B2(26),SEP8,B3(26),SEP9,
     3            C0(33),SEP10,C1(33),SEP11,C2(33),SEP12,C3(33),SEP13,
     4            S0(193,33),SEP14,E0(129),SEP15,Z0(129),SEP16,
     5            IV(193,33),SEP17,ITE1(33),SEP18,ITE2(33),SEP19,
     6            NX,NY,NZ,KTE1,KTE2,KSYM,SCAL,SCALZ,
     7            YAW,CYAW,SYAW,ALPHA,CA,SA,FMACH,N1,N2,N3,IO
      MX          = NX  +1
      KY          = NY  +1
      MY          = NY  +2
      MZ          = NZ  +1
      MX0         = NX/2  +1
      MY0         = NY/2  +2
      MZ0         = NZ/2  +1
      K           = 1
C     INTERPOLATE POTENTIAL ARRAY G
C     READ SLICE OF POTENTIAL ARRAY FROM FIRST DISC FILE
   11 BUFFER IN (N1,1) (G(1,1,1),G(MX0,MY0,1))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N1).GT.0.) GO TO 401
C     SHIFT VALUES TO LOCATIONS IN NEW GRID
      J           = NY/2  +1
      JJ          = KY
   21 I           = MX0
      II          = MX
```

```
   31 G(II,JJ,1)   = G(I,J,1)
      I           = I  -1
      II          = II  -2
      IF (I.GT.0) GO TO 31
      J           = J  -1
      JJ          = JJ  -2
      IF (J.GT.0) GO TO 21
C     INTERPOLATE IN X
      DO 42 J=1,KY,2
      DO 42 I=2,NX,2
   42 G(I,J,1)    = .5*(G(I+1,J,1)  +G(I-1,J,1))
C     INTERPOLATE IN Y
      DO 52 I=1,MX
      DO 54 J=2,NY,2
   54 G(I,J,1)    = .5*(G(I,J+1,1)  +G(I,J-1,1))
   52 G(I,MY,1) = 0.
C     WRITE SLICE OF INTERPOLATED POTENTIAL ARRAY ON SECOND DISC FILE
      BUFFER OUT(N2,1) (G(1,1,1),G(MX,MY,1))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N2).GT.0.) GO TO 401
C     INCREMENT Z
      K           = K  +1
      IF (K.LE.MZ0) GO TO 11
      REWIND N1
      REWIND N2
C     READ FIRST TWO SLICES OF POTENTIAL ARRAY FROM SECOND DISC FILE
      BUFFER IN (N2,1) (G(1,1,1),G(MX,MY,1))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N2).GT.0.) GO TO 401
      BUFFER IN (N2,1) (G(1,1,3),G(MX,MY,3))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N2).GT.0.) GO TO 401
C     WRITE FIRST SLICE OF POTENTIAL ARRAY ON FIRST DISC FILE
      BUFFER OUT(N1,1) (G(1,1,1),G(MX,MY,1))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N1).GT.0.) GO TO 401
      K           = 1
C     INCREMENT Z
  111 K           = K  +1
C     INTERPOLATE IN Z
      DO 112 J=1,MY
      DO 112 I=1,MX
  112 G(I,J,2)    = .5*(G(I,J,1)  +G(I,J,3))
C     WRITE TWO SLICES OF INTERPOLATED POTENTIAL ARRAY
C     ON FIRST DISC FILE
      DO 122 L=2,3
      BUFFER OUT(N1,1) (G(1,1,L),G(MX,MY,L))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N1).GT.0.) GO TO 401
  122 CONTINUE
      IF (K.EQ.MZ0) GO TO 201
C     SHIFT SLICES OF POTENTIAL ARRAY
      DO 132 J=1,MY
      DO 132 I=1,MX
  132 G(I,J,1)    = G(I,J,3)
```

```
C     READ SLICE OF POTENTIAL ARRAY FROM SECOND DISC FILE
      BUFFER IN (N2,1) (G(1,1,3),G(MX,MY,3))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N2).GT.0.) GO TO 401
      GO TO 111
  201 REWIND N1
      REWIND N2
C     SET VALUES OF POTENTIAL AT DUMMY POINTS BEHIND BOUNDARY
C     READ FIRST THREE SLICES OF POTENTIAL ARRAY FROM FIRST DISC FILE
      DO 202 L=1,3
      BUFFER IN (N1,1) (G(1,1,L),G(MX,MY,L))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N1).GT.0.) GO TO 401
  202 CONTINUE
C     WRITE FIRST SLICE OF POTENTIAL ARRAY ON SECOND DISC FILE
      BUFFER OUT(N2,1) (G(1,1,1),G(MX,MY,1))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N2).GT.0.) GO TO 401
      TYAW         = .5*SCAL*SYAW/CYAW
      N0           = KTE1  -1
      E0(N0)       = 0.
      K            = 2
  211 Z            = SCALZ*C0(K)
      N            = N0
      I            = MX0  +1
      IF (K.LT.KTE1.OR.K.GT.KTE2) GO TO 231
      I1           = ITE1(K)
      I2           = ITE2(K)
C     ITE1 AND ITE2 ARE LOWER AND UPPER TRAILING EDGE POINTS
      DO 212 I=I1,I2
C     DETERMINE SLOPES SX,SZ OF SHEARING TRANSFORMATION
      DSI          = S0(I+1,K)  -S0(I-1,K)
      DSK          = S0(I,K+1)  -S0(I,K-1)
      SX           = A1(I)*DSI
      SZ           = C1(K)*DSK
      DGI          = G(I+1,KY,2)  -G(I-1,KY,2)
      DGK          = G(I,KY,3)  -G(I,KY,1)
      U            = A1(I)*DGI  +CA*A0(I)  +SA*S0(I,K)
      W            = C1(K)*DGK  +SYAW
      FH           = A0(I)*A0(I)  +S0(I,K)*S0(I,K)
      V            = B1(KY)*(1.  +SX*SX  +FH*SZ*SZ)
C     SET POTENTIAL BELOW SURFACE TO SATISFY BOUNDARY CONDITION
  212 G(I,KY+1,2) = G(I,KY-1,2)
     1                 -(CA*S0(I,K)  -SA*A0(I)  +U*SX  +FH*W*SZ)/V
      N0           = N0  +1
C     RESET TRAILING EDGE JUMP E0 IN POTENTIAL
      E0(N0)       = G(I2,KY,2)  -G(I1,KY,2)
      N            = N0
      I            = I1
      IF (K.NE.KTE2.OR.YAW.LE.0.) GO TO 231
C     RESET JUMP E0 IN POTENTIAL ALONG SIDE EDGE OF DOWNSTREAM TIP
  221 I            = I  +1
      M            = NX  +2  -I
      N0           = N0  +1
      E0(N0)       = G(M,KY,2)  -G(I,KY,2)
```

```
      IF (I.LT.MXO) GO TO 221
      I          = I1
  231 I          = I  -1
      E          = 0.
      IF (IV(I,K).NE.1) GO TO 237
C     IV = 1 INDICATES VORTEX POINT
C     INTERPOLATE JUMP E IN POTENTIAL
C     TO SET POTENTIAL AT DUMMY POINT BELOW VORTEX SHEET
      ZZ         = Z  -TYAW*AO(I)**2
  233 IF (ZZ.GE.ZO(N-1)) GO TO 235
      N          = N  -1
      GO TO 233
  235 R          = (ZZ  -ZO(N-1))/(ZO(N)  -ZO(N-1))
      E          = R*EO(N)  +(1.  -R)*EO(N-1)
  237 M          = NX  +2  -I
      G(I,KY+1,2) = G(M,KY-1,2)  -E
      G(M,KY+1,2) = G(I,KY-1,2)  +E
      IF (IV(I,K).NE.-1) GO TO 241
C     IV = -1 INDICATES POINT JUST BEYOND EDGE OF WING OR VORTEX SHEET
C     RENORMALIZE POTENTIAL ON EITHER SIDE OF CUT AT MEAN VALUE
      G(I,KY,2)   = .5*G(I,KY,1)  +.25*(G(I,KY,3)  +G(M,KY,3))
      IF (IV(I,K+1).LT.1)
     1G(I,KY,2)   = .5*G(I,KY,3)  +.25*(G(I,KY,1)  +G(M,KY,1))
      G(M,KY,2)   = G(I,KY,2)
      G(I,KY-1,2) = .5*(G(I,KY,2)  +G(I,KY-2,2))
      G(M,KY-1,2) = .5*(G(M,KY,2)  +G(M,KY-2,2))
  241 IF (I.GT.2) GO TO 231
      IF (K.EQ.NZ) GO TO 261
C     SHIFT SLICES OF POTENTIAL ARRAY
      DO 252 J=1,MY
      DO 252 I=1,MX
      G(I,J,1)  = G(I,J,2)
  252 G(I,J,2)  = G(I,J,3)
C     WRITE SLICE OF UPDATED POTENTIAL ARRAY ON SECOND DISC FILE
      BUFFER OUT(N2,1) (G(1,1,1),G(MX,MY,1))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N2).GT.0.) GO TO 401
C     READ SLICE OF POTENTIAL ARRAY FROM FIRST DISC FILE
      BUFFER IN (N1,1) (G(1,1,3),G(MX,MY,3))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N1).GT.0.) GO TO 401
C     INCREMENT Z
      K          = K  +1
      GO TO 211
  261 EO(NO+1)   = 0.
C     WRITE LAST TWO SLICES OF POTENTIAL ARRAY ON SECOND DISC FILE
      DO 262 L=2,3
      BUFFER OUT(N2,1) (G(1,1,L),G(MX,MY,L))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N2).GT.0.) GO TO 401
  262 CONTINUE
      REWIND N1
      REWIND N2
C     COPY FINAL VALUES OF POTENTIAL ON FIRST DISC FILE
      DO 302 K=1,MZ
```

```
      BUFFER IN (N2,1) (G(1,1,1),G(MX,MY,1))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N2).GT.0.) GO TO 401
      BUFFER OUT(N1,1) (G(1,1,1),G(MX,MY,1))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N1).GT.0.) GO TO 401
  302 CONTINUE
C     SET IO TO INDICATE SUCCESSFUL COMPLETION
      IO         = 1
      RETURN
C     SET IO TO INDICATE DISC FAILURE
  401 IO         = 0
      RETURN
      END




      SUBROUTINE  SMOO
C     SMOOTHS POTENTIAL
C     BY REPLACING THE VALUE AT EACH POINT BY A WEIGHTED AVERAGE
C     OF THE VALUES AT NEIGHBOURING POINTS
C     SUCCESSIVE SLICES OF THE G ARRAY ,REPRESENTING VALUES OF POTENTIAL
C     ON X-Y PLANES AT SUCCESSIVE VALUES OF Z,ARE READ
C     FROM ONE DISC FILE,UPDATED,AND WRITTEN ON A SECOND DISC FILE
      COMMON      G(193,26,4),SEP1,
     1            A0(193),SEP2,A1(193),SEP3,A2(193),SEP4,A3(193),SEP5,
     2            B0(26),SEP6,B1(26),SEP7,B2(26),SEP8,B3(26),SEP9,
     3            C0(33),SEP10,C1(33),SEP11,C2(33),SEP12,C3(33),SEP13,
     4            S0(193,33),SEP14,E0(129),SEP15,Z0(129),SEP16,
     5            IV(193,33),SEP17,ITE1(33),SEP18,ITE2(33),SEP19,
     6            NX,NY,NZ,KTE1,KTE2,KSYM,SCAL,SCALZ,
     7            YAW,CYAW,SYAW,ALPHA,CA,SA,FMACH,N1,N2,N3,IO
      MX         = NX   +1
      KY         = NY   +1
      MY         = NY   +2
      MZ         = NZ   +1
C     SET SMOOTHING PARAMETERS
      PX         = 1./6.
      PY         = 1./6.
      PZ         = 1./6.
C     READ FIRST THREE SLICES OF POTENTIAL ARRAY FROM FIRST DISC FILE
      DO 2 L=1,3
      BUFFER IN (N1,1) (G(1,1,L),G(MX,MY,L))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N1).GT.0.) GO TO 51
    2 CONTINUE
C     WRITE FIRST SLICE OF POTENTIAL ARRAY ON SECOND DISC FILE
      BUFFER OUT(N2,1) (G(1,1,1),G(MX,MY,1))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N2).GT.0.) GO TO 51
      K          = 1
C     INCREMENT Z
   11 K          = K  +1
C     GENERATE SMOOTHED VALUES OF POTENTIAL FOR MIDDLE SLICE
```

```
      DO 12 J=3,NY
      DO 14 I=2,NX
   14 G(I,J,4)  = (1.  -PX  -PY  -PZ)*G(I,J,2)
     1             +.5*PX*(G(I+1,J,2)  +G(I-1,J,2))
     2             +.5*PY*(G(I,J+1,2)  +G(I,J-1,2))
     3             +.5*PZ*(G(I,J,3)  +G(I,J,1))
      G(1,J,4)  = G(1,J,2)
   12 G(MX,J,4) = G(MX,J,2)
C     LEAVE BOUNDARY VALUES UNCHANGED
      DO 16 I=1,MX
      G(I,1,4)  = G(I,1,2)
      G(I,2,4)  = G(I,2,2)
      G(I,KY,4) = G(I,KY,2)
   16 G(I,MY,4) = G(I,MY,2)
C     WRITE SLICE OF UPDATED POTENTIAL ARRAY ON SECOND DISC FILE
      BUFFER OUT(N2,1) (G(1,1,4),G(MX,MY,4))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N2).GT.0.) GO TO 51
      IF (K.EQ.NZ) GO TO 31
C     SHIFT SLICES OF POTENTIAL ARRAY
      DO 22 J=1,MY
      DO 22 I=1,MX
      G(I,J,1)  = G(I,J,2)
   22 G(I,J,2)  = G(I,J,3)
C     READ SLICE OF POTENTIAL ARRAY FROM FIRST DISC FILE
      BUFFER IN (N1,1) (G(1,1,3),G(MX,MY,3))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N1).GT.0.) GO TO 51
      GO TO 11
C     WRITE LAST SLICE OF UPDATED POTENTIAL ARRAY ON SECOND DISC FILE
   31 BUFFER OUT(N2,1) (G(1,1,3),G(MX,MY,3))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N2).GT.0.) GO TO 51
      REWIND N1
      REWIND N2
C     COPY FINAL VALUES OF POTENTIAL ON FIRST DISC FILE
      DO 42 K=1,MZ
      BUFFER IN (N2,1) (G(1,1,1),G(MX,MY,1))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N2).GT.0.) GO TO 51
      BUFFER OUT(N1,1) (G(1,1,1),G(MX,MY,1))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N1).GT.0.) GO TO 51
   42 CONTINUE
C     SET IO TO INDICATE SUCCESSFUL COMPLETION
      IO        = 1
      RETURN
C     SET IO TO INDICATE DISC FAILURE
   51 IO        = 0
      RETURN
      END
```

```
      SUBROUTINE  SPLIF(M,N,S,F,FP,FPP,FPPP,KM,VM,KN,VN,MODE,FQM,IND)
C     CUBIC SPLINE
C     SPLINE IS FITTED TO DATA ARRAY F AT NODES S
C     FROM INDEX M TO INDEX N
C     KM = 1,2 OR 3 INDICATES THAT FIRST,SECOND OR THIRD DERIVATIVE
C     IS GIVEN VALUE VM AT POINT M
C     KN = 1,2 OR 3 INDICATES THAT FIRST,SECOND OR THIRD DERIVATIVE
C     IS GIVEN VALUE VN AT POINT N
C     IF MODE = 0 NODAL VALUES OF FIRST,SECOND AND THIRD DERIVATIVES
C     OF SPLINE ARE STORED IN FP,FPP AND FPPP ARRAYS
C     SO THAT FITTED VALUE AT A DISTANCE H BEYOND A NODE IS
C     F  +FP*H  +FPP*H**2/2.  +FPPP*H**3/6
C     IF MODE GT 0 FPPP IS GIVEN THE NODAL VALUES OF THE INTEGRAL OF F
C     INSTEAD OF ITS THIRD DERIVATIVE, STARTING WITH THE VALUE FQM
C     THEN THE THIRD DERIVATIVE CAN BE RECOVERED AS
C     (FPP(I+1)  -FP(I))/(S(I+1)  -S(I))
C     IND IS SET EQUAL TO 0 IF S IS NOT A MONOTONE ARRAY
      DIMENSION    S(1),F(1),FP(1),FPP(1),FPPP(1)
      IND          = 0
      K            = IABS(N  -M)
      IF (K  -1) 81,81,1
    1 K            = (N  -M)/K
      I            = M
      J            = M  +K
      DS           = S(J)  -S(I)
      D            = DS
      IF (DS) 11,81,11
   11 DF           = (F(J)  -F(I))/DS
      IF (KM  -2) 12,13,14
   12 U            = .5
      V            = 3.*(DF  -VM)/DS
      GO TO 25
   13 U            = 0.
      V            = VM
      GO TO 25
   14 U            = -1.
      V            = -DS*VM
      GO TO 25
   21 I            = J
      J            = J  +K
      DS           = S(J)  -S(I)
      IF (D*DS) 81,81,23
   23 DF           = (F(J)  -F(I))/DS
      B            = 1./(DS  +DS  +U)
      U            = B*DS
      V            = B*(6.*DF  -V)
   25 FP(I)        = U
      FPP(I)       = V
      U            = (2.  -U)*DS
      V            = 6.*DF  +DS*V
      IF (J  -N) 21,31,21
   31 IF (KN  -2) 32,33,34
   32 V            = (6.*VN  -V)/U
      GO TO 35
   33 V            = VN
```

```
      GO TO 35
   34 V            = (DS*VN  +FPP(I))/(1.  +FP(I))
   35 B            = V
      D            = DS
   41 DS           = S(J)  -S(I)
      U            = FPP(I)  -FP(I)*V
      FPPP(I)      = (V  -U)/DS
      FPP(I)       = U
      FP(I)        = (F(J)  -F(I))/DS  -DS*(V  +U  +U)/6.
      V            = U
      J            = I
      I            = I  -K
      IF (J  -M) 41,51,41
   51 I            = N  -K
      FPPP(N)      = FPPP(I)
      FPP(N)       = B
      FP(N)        = DF  +D*(FPP(I)  +B  +B)/6.
      IND          = 1
      IF (MODE) 81,81,61
   61 FPPP(J)      = FQM
      V            = FPP(J)
   71 I            = J
      J            = J  +K
      DS           = S(J)  -S(I)
      U            = FPP(J)
      FPPP(J)      = FPPP(I)  +.5*DS*(F(I)  +F(J)  -DS*DS*(U  +V)/12.)
      V            = U
      IF (J  -N)  71,81,71
   81 RETURN
      END



      SUBROUTINE  INTPL(MI,NI,SI,FI,M,N,S,F,FP,FPP,FPPP,MODE)
C     INTERPOLATION USING PIECEWISE TAYLOR SERIES
C     AS GENERATED BY CUBIC SPLINE OR ITS INTEGRAL
C     VALUES F,FP,FPP AND FPPP OF FUNCTION AND ITS FIRST,SECOND
C     AND THIRD DERIVATIVES ARE GIVEN AT NODES S
C     FROM INDEX M TO INDEX N
C     INTERPOLATED VALUES FI ARE GENERATED AT POINTS SI
C     FROM INDEX MI TO INDEX NI
C     IF MODE GT 0 A CORRECTION IS ADDED
C     FOR A PIECEWISE CONSTANT FOURTH DERIVATIVE
C     SO THAT INTEGRAL OF CUBIC SPLINE IS EVALUATED EXACTLY
      DIMENSION    SI(1),FI(1),S(1),F(1),FP(1),FPP(1),FPPP(1)
      K            = IABS(N  -M)
      K            = (N  -M)/K
      I            = M
      MIN          = MI
      NIN          = NI
      D            = S(N)  -S(M)
      IF (D*(SI(NI)  -SI(MI))) 11,13,13
   11 MIN          = NI
      NIN          = MI
```

```
   13 KI         = IABS(NIN  -MIN)
      IF (KI) 21,21,15
   15 KI         = (NIN  -MIN)/KI
   21 II         = MIN  -KI
      C          = 0.
      IF (MODE) 31,31,23
   23 C          = 1.
   31 II         = II  +KI
      SS         = SI(II)
   33 I          = I  +K
      IF (I  -N) 35,37,35
   35 IF (D*(S(I)  -SS)) 33,33,37
   37 J          = I
      I          = I  -K
      SS         = SS  -S(I)
      FPPPP      = C*(FPPP(J)  -FPPP(I))/(S(J)  -S(I))
      FF         = FPPP(I)  +.25*SS*FPPPP
      FF         = FPP(I)  +SS*FF/3.
      FF         = FP(I)  +.5*SS*FF
      FI(II)     = F(I)  +SS*FF
      IF (II  -NIN) 31,41,31
   41 RETURN
      END


      SUBROUTINE  GRAPH (IPLOT,I1,I2,X,Y,CP,TITLE,FMACH,YA,AL,
     1                   Z,CL,CD,CHORD0)
C     GENERATES CALCOMP PLOT OF WING SECTION
C     AND SURFACE PRESSURE COEFFICIENT
      DIMENSION   X(1),Y(1),CP(1),TITLE(12),R(12)
      IF (IPLOT.GE.0) GO TO 11
C     INITIALIZE PLOTTER IF IPLOT LT 0
      CALL PLOTSBL(1000,24HANTONY JAMESON   X109403)
C     DEFINE ORIGIN
      CALL PLOT(1.25,1.0,-3)
C     WRITE TITLE AND FLOW PARAMETERS
   11 ENCODE(48,12,R(1)) TITLE
   12 FORMAT(12A4)
      CALL SYMBOL(0.,0.,.14,R,0.,48)
      ENCODE(40,14,R(1)) FMACH,YA,AL
   14 FORMAT(4HM = ,F8.3,3X,6HYAW = ,F5.2,3X,6HALP = ,F5.2)
      CALL SYMBOL(0.,-.25,.14,R,0.,40)
      ENCODE(40,16,R(1)) Z,CL,CD
   16 FORMAT(4HZ = ,F8.2,3X,5HCL = ,F6.4,3X,5HCD = ,F6.4)
      CALL SYMBOL(0.,-.5,.14,R,0.,40)
C     SCALE AND TRANSLATE PROFILE
      XMIN       = X(I1)
      DO 22 I=I1,I2
   22 XMIN       = AMIN1(X(I),XMIN)
      SCALE      = 5./CHORD0
      DO 24 I=I1,I2
      X(I)       = (X(I)  -XMIN)*SCALE  +.5
   24 Y(I)       = Y(I)*SCALE  +.75
```

```
C     LOCATE FRONT STAGNATION POINT
      CPMAX      = 0.
      IMAX       = I1
      N1         = I1  +1
      N2         = I2  -1
      DO 26 I=N1,N2
      IF (CP(I).LE.CPMAX) GO TO 26
      CPMAX      = CP(I)
      IMAX       = I
   26 CONTINUE
      N          = I2  -I1  +1
C     DRAW PROFILE
      CALL LINE(X(I1),Y(I1),N,1,0,1,0.,1.,0.,1.)
C     SHIFT ORIGIN
      CALL PLOT(0.,4.5,-3)
C     DRAW VERTICAL AXIS
      CALL AXIS(0.,-3.,2HCP,2,8.,90.,1.2,-.4,0)
C     MARK CRITICAL PRESSURE COEFFICIENT ON AXIS
      CPC        = (((5.  +FMACH**2)/6.)**3.5  -1.)/(.7*FMACH**2)
      IF (CPC.GE.-2.0) CALL SYMBOL(0.,-2.5*CPC,.40,15,0.,-1)
C     PLOT LOWER SURFACE PRESSURE COEFFICIENT
      DO 32 I=I1,IMAX
      IF (CP(I).LT.-2.0) GO TO 32
      CALL SYMBOL(X(I),-2.5*CP(I),.07,3,45.,-1)
   32 CONTINUE
C     PLOT UPPER SURFACE PRESSURE COEFFICIENT
      DO 34 I=IMAX,I2
      IF (CP(I).LT.-2.0) GO TO 34
      CALL SYMBOL(X(I),-2.5*CP(I),.07,3,0.,-1)
   34 CONTINUE
C     SHIFT ORIGIN FOR NEXT PLOT
      CALL PLOT(12.,-4.5,-3)
      RETURN
      END



      SUBROUTINE  THREED(IPLOT,SV,SM,CP,X,Y,TITLE,YA,AL,
     1                   VLD,CL,CD,CHORDD)
C     GENERATES DRAWING OF WING AND THREE DIMENSIONAL PLOTS
C     OF PRESSURE COEFFICIENT OVER UPPER AND LOWER SURFACES
      COMMON      G(193,26,4),SEP1,
     1            A0(193),SEP2,A1(193),SEP3,A2(193),SEP4,A3(193),SEP5,
     2            B0(26),SEP6,B1(26),SEP7,B2(26),SEP8,B3(26),SEP9,
     3            C0(33),SEP10,C1(33),SEP11,C2(33),SEP12,C3(33),SEP13,
     4            S0(193,33),SEP14,E0(129),SEP15,Z0(129),SEP16,
     5            IV(193,33),SEP17,ITE1(33),SEP18,ITE2(33),SEP19,
     6            NX,NY,NZ,KTE1,KTE2,KSYM,SCAL,SCALZ,
     7            YAW,CYAW,SYAW,ALPHA,CA,SA,FMACH,N1,N2,N3,I0
      DIMENSION   X(1),Y(1),SV(1),SM(1),CP(1),TITLE(12),R(12)
      LX         = NX/2  +1
      MX         = NX  +1
      MY         = NY  +2
C     SET HORIZONTAL AND VERTICAL SHIFTS
```

```
      SX          = 2.
      TX          = 3.5
      SY          = 2.75
      DY          = 8./NZ
      IF (IPLOT.GE.0) GO TO 1
C     INITIALIZE PLOTTER IF IPLOT LT 0
      CALL PLOTSBL(1000,24HANTONY JAMESON   X109403)
C     DEFINE ORIGIN
      CALL PLOT(1.25,1.0,-3)
    1 M           = 1
C     WRITE TITLE FOR DRAWING OF WING
      ENCODE(12,2,R(1))
    2 FORMAT(12HVIEW OF WING)
      CALL SYMBOL(2.,.5,.14,R,0.,12)
C     READ FIRST THREE SLICES OF POTENTIAL ARRAY FROM DISC FILE
   11 DO 12 L=1,3
      BUFFER IN (N1,1) (G(1,1,L),G(MX,MY,L))
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (UNIT(N1).GT.0.) GO TO 101
   12 CONTINUE
      K           = 2
C     INCREMENT Z
   21 K           = K  +1
      IF (K.GT.KTE2) GO TO 61
C     SHIFT SLICES OF POTENTIAL ARRAY
      DO 22 J=1,MY
      DO 22 I=1,MX
      G(I,J,1)    = G(I,J,2)
   22 G(I,J,2)    = G(I,J,3)
C     READ SLICE OF POTENTIAL ARRAY FROM DISC FILE
      BUFFER IN (N1,1) (G(1,1,3),G(MX,MY,3))
      IF (UNIT(N1).GT.0.) GO TO 101
C     GIVE UP IN EVENT OF DISC FAILURE
      IF (K.LT.KTE1) GO TO 21
      I1          = ITE1(K)
      I2          = ITE2(K)
C     CALCULATE SURFACE SPEED SV,MACH NUMBER SM,PRESSURE COEFFICIENT CP
C     AND COORDINATES X,Y OF WING SECTION
      CALL VELO (K,2,SV,SM,CP,X,Y)
      IF (K.GT.KTE1) GO TO 41
C     WRITE TITLE AND FLOW PARAMETERS
C     BEFORE GENERATING PLOT AT FIRST SPAN STATION
      ENCODE(48,32,R(1)) TITLE
   32 FORMAT(12A4)
      CALL SYMBOL(.5,0.,.14,R,0.,48)
      ENCODE(40,34,R(1)) FMACH,YA,AL
   34 FORMAT(4HM = ,F8.3,3X,6HYAW = ,F5.2,3X,6HALP = ,F5.2)
      CALL SYMBOL(.5,-.25,.14,R,0.,40)
      ENCODE(40,36,R(1)) VLD,CL,CD
   36 FORMAT(6HL/D = ,F6.2,3X,5HCL = ,F6.4,3X,5HCD = ,F6.4)
      CALL SYMBOL(.5,-.5,.14,R,0.,40)
C     SCALE AND TRANSLATE COORDINATES AND PRESSURE COEFFICIENT
   41 XMIN        = X(I1)
      DO 42 I=I1,I2
   42 XMIN        = AMIN1(X(I),XMIN)
```

```
      SCALX      = 2.5/CHORDO
      SCALP      = -1.25
      DO 44 I=I1,I2
      X(I)       = (X(I)   -XMIN)*SCALX   +SX
      Y(I)       = Y(I)*SCALX  +SY
   44 CP(I)      = SCALP*CP(I)   +SY
C     INCREMENT VERTICAL SHIFT FOR NEXT SPAN STATION
      SY         = SY  +DY
      IF (M.EQ.2) GO TO 51
C     IF M = 1 DRAW WING SECTION
      N          = I2  -I1  +1
      CALL LINE(X(I1),Y(I1),N,1,0,1,0.,1.,0.,1.)
      GO TO 21
C     IF M = 2 PLOT PRESSURE COEFFICIENT OVER UPPER AND LOWER SURFACES
   51 N          = I2  -LX  +1
C     PLOT UPPER SURFACE COEFFICIENT AT LEFT SIDE OF PAGE
      CALL LINE(X(LX),CP(LX),N,1,0,1,0.,1.,0.,1.)
      N          = LX  -I1  +1
C     TRANSLATE X COORDINATES TO RIGHT
      DO 52 I=I1,LX
   52 X(I)       = X(I)   +TX
C     PLOT LOWER SURFACE COEFFICIENT AT RIGHT SIDE OF PAGE
      CALL LINE(X(I1),CP(I1),N,1,0,1,0.,1.,0.,1.)
      GO TO 21
   61 REWIND N1
      M          = M  +1
C     SHIFT ORIGIN FOR NEXT PLOT
      CALL PLOT(12.,0.,-3)
      IF (M.GT.2) GO TO 71
C     RESET HORIZONTAL AND VERTICAL SHIFTS
      SX         = 0.
      SY         = 2.75
C     WRITE TITLES FOR PRESSURE PLOTS
      ENCODE(24,62,R(1))
   62 FORMAT(24HUPPER SURFACE PRESSURE  )
      CALL SYMBOL(0.,.5,.14,R,0.,24)
      ENCODE(24,64,R(1))
   64 FORMAT(24HLOWER SURFACE PRESSURE  )
      CALL SYMBOL(3.5,.5,.14,R,0.,24)
      GO TO 11
C     SET IO TO INDICATE SUCCESSFUL COMPLETION
   71 IO         = 1
      RETURN
C     SET IO TO INDICATE DISC FAILURE
  101 IO         = 0
      RETURN
      END
```

7. Listing of Quasiconservation Option for Program H

This is a listing of an option for Program H that gives correct resolution of the shock conditions by using the theory described in Section 2 of Chapter I. The results obtained from the option agree almost perfectly with those of an exact, or full, conservation form of the finite difference scheme. We do not list the exact form because its computation time is about forty percent longer than the listed option. The option is based on a centered finite difference approximation of a quasilinear equation for the velocity potential ϕ combined with artificial viscosity terms in true conservation form. Further details about this new procedure will appear in a later publication. Our limited experience with it indicates that it does not give such a reliable overall simulation of boundary layer shock wave interaction as does the old Murman subroutine it replaces (see the seventh, eighth and ninth pages of the listing of Program H).

LISTING OF QUASICONSERVATION OPTION FOR PROGRAM H 07/18/74

```
C     ****TO USE THIS OPTION REPLACE THE SUBROUTINE MURMAN FOUND ****
C     ****ON PAGES 7 THRU 9 OF THE LISTING OF PROGRAM H BY THE   ****
C     ****FOLLOWING NEW VERSION.                                 ****
      SUBROUTINE MURMAN
C     SET UP COEFFICIENT ARRAYS FOR THE TRIDIAGONAL SYSTEM USED FOR LINE
C     RELAXATION AND COMPUTE THE UPDATED PHI ON THIS LINE
      COMMON PHI(162,31),FP(162,31),A(31),B(31),C(31),D(31),E(31)
     1 ,RP(31),RPP(31),R(31),RS(31),RI(31),AA(162),BB(162),CO(162)
     2 ,SI(162),PHIR(162),XC(162),YC(162),FM(162),ARCL(162).DSUM(162)
     3 ,ANGOLD(162),XOLD(162),YOLD(162),ARCOLD(162),DELOLD(162)
      COMMON /A/ PI,TP,RAD,EM,ALP,RN,PCH,XP,TC,CHD,DPHI,CL,RCL,YR
     1 ,XA,YA,TE,DT,DR,DELTH,DELR,RA,DCN,DSN,RA4,EPSIL,QCRIT,C1,C2
     2 ,C4,C5,C6,C7,BET,BETA,FSYM,VSEP,SEPM,TTLE(4),M,N,MM,NN,NSP
     3 ,IK,JK,IZ,ITYP,MODE,IS,NFC,NCY,NRN,NG,IDIM,N2,N3,N4,NT,IXX
     4 , NPTS,LL,I,LSEP,M4
      DIMENSION VV(35),RPO(35)
      DATA RPO/35*0./
      BETP = BETA+.25
C     DO THE BOUNDARY
      RP(1) = 0.
      RP(NN) = 0.
      KK = 0
      PHIO = PHI(I,2)-2.*DR*CO(I)
      PHIYP= PHI(I,2)-PHI(I,1)
      PHIYY = PHIYP+PHIO-PHI(I,1)
      PHIXX = PHI(I+1,1)+PHI(I-1,1)-PHI(I,1)-PHI(I,1)
      PHIXM = PHI(I+1,1)-PHI(I-1,1)
      PHIXP = PHI(I+1,2)-PHI(I-1,2)
C     CHECK FOR THE TAIL POINT
      IF (I.NE.MM) GO TO 10
      C(1) = (C1+C1)*RS(1)
      A(1) = -C(1)+XA*C1-C1
      D(1) = C1*(PHIXX+RS(1)*PHIYY+RA4*CO(I)-E(1))
      GO TO 40
   10 U = PHIXM*DELTH-SI(I)
      BQ = U/FP(I,1)
      QS = U*BQ
      CS = C1-C2*QS
      BQ = BQ*QS*(FP(I-1,1)-FP(I+1,1))
      X = RA4*(CS+QS)*CO(I)
      C(1) = (CS+CS)*RS(1)
      CMQS = CS-QS
      D(1) = CMQS*PHIXX+CS*RS(1)*PHIYY + RI(1)*BQ + X + RPP(1)
      PHIXT = BETP*ABS(U)+ABS(CMQS)
      IF (CMQS.GE.0.) GO TO 30
C     FLOW IS SUPERSONIC. BACKWARD DIFFERENCES
      KK = 1
      PHIXT = PHIXT-CMQS
      A(1) = -(C(1)+PHIXT)
      RPP(1) = CMQS*PHIXX
      D(1) = D(1)-PHIXT*E(1)-RPP(1)
      GO TO 40
C     FLOW SUBCRITICAL. CENTRAL DIFFERENCES
```

```
   30 A(1) = XA*CMQS-C(1)-PHIXT
      D(1) = D(1)-PHIXT*E(1)
      RPP(1) = 0.
C     DO NON-BOUNDARY POINTS
   40 DO 60 J = 2,N
      PHIXX = PHI(I+1,J)+PHI(I-1,J)-PHI(I,J)-PHI(I,J)
      DU = PHIXP
      PHIXP = PHI(I+1,J+1)-PHI(I-1,J+1)
      PHIXY = PHIXP-PHIXM
      PHIXM = DU
      DU = DU*DELTH
      PHIYM = PHIYP
      PHIYP = PHI(I,J+1)-PHI(I,J)
      PHIYY = PHIYP-PHIYM
      U = R(J)*DU-SI(I)
      DV = R(J)*(PHI(I,J+1)-PHI(I,J-1))*DELR
      V = DV*R(J)-CO(I)
      VV(J) = V
      RAV = R(J)*RA*V
      BQ = 1./FP(I,J)
      BQU = BQ*U
      US = BQU*U
      UV = (BQU+BQU)*V
      VS = BQ*V*V
      QS = US+VS
      CS = C1-C2*QS
      CMVS = CS-VS
      CMUS = CS-US
C     COMPUTE CONTRIBUTION OF RIGHT-HAND SIDE FROM LOW ORDER TERMS
      D(J) =RA4*((CMVS+US-VS)*DV-UV*DU)+RI(J)*QS*BQ*(U*(FP(I-1,J)-
     1 FP(I+1,J))+RAV*(FP(I,J-1)-FP(I,J+1)))
      UV = .5*BQU*RAV
      C(J) =  RS(J)*CMVS
      B(J) = C(J)
      D(J) = D(J)+C(J)*PHIYY-UV*PHIXY+CMUS*PHIXX+RPP(J)
      CSQS = CS/QS
      CMQS = CSQS-1.
      PHIXT = BETP*ABS(U)
      PHIYT = BETP*ABS(RAV)
      IF (CMQS.GE.0.) GO TO 50
C     SUPERSONIC FLOW, USE BACKWARD DIFFERENCING
      KK = KK+1
      PHIXT = PHIXT-CMQS*(US+US+ABS(UV))+CSQS*VS
      PHIYT = PHIYT-CMQS*(RS(J)*(VS+VS)+ABS(UV))
      B(J) = RS(J)*CSQS*US
      C(J) = B(J)+PHIYT
      A(J) = -(C(J)+B(J)+PHIXT)
      RPP(J) = CMQS*(US*PHIXX + UV*PHIXY)
      RP(J) = CMQS*RS(J)*VS*PHIYY+RPO(J)
      RPO(J) = CMQS*UV*PHIXY
      GO TO 60
C     SUBSONIC FLOW, USE CENTRAL DIFFERENCES
   50 C(J) = C(J)+PHIYT
      PHIXT = PHIXT+CMUS
      A(J) =  XA*CMUS-B(J)-C(J)-PHIXT
```

```
      RPP(J) = 0.
      RP(J) = RPO(J)
      RPO(J) = 0.
   60 D(J) = D(J)-PHIXT*E(J)-RPP(J)-RP(J)
      DO 70 J = 2,N
      IF (VV(J).LT.0.) GO TO 72
      D(J) = D(J)+RP(J+1)
      GO TO 70
   72 BQ = B(J)
      B(J) = C(J)
      C(J) = BQ
      D(J) = D(J)+RP(J-1)
   70 CONTINUE
   75 NSP = NSP+KK
C     SOLVE THE TRIDIAGONAL SYSTEM
      CALL TRID
      RETURN
      END
```

8. Listing of Update for Design Programs B and D

We start with an update for the glossary of Tape 7 parameters which appears on pages 105 and 106 of Volume I.

The following two parameters have been redefined:

NRN	Integer. ABS(NRN) is the run number. If NRN is negative the paths in the hodograph plane are plotted. If NRN > 1000 the Calcomp plots are done on blank paper on the CIMS CDC 6600.
TR	Real. Between 0. and 1. it specifies the relative location of the artificial tail between trailing streamlines $\psi = 0$. If TR > 1 the new model of the tail is chosen.

The following two parameters have been added:

NCR	Integer. Number of constraints. If omitted or zero it will default to 7. NCR is added on Card 1, Columns 61-65 of Table 1, page 107 of Volume I.
TE	Real. Tail extension parameter. If TE $\geq$ 0 points up to TE will be printed and plotted in D. For TE set to zero or omitted TE defaults to 1. If TE is negative and TR $\leq$ 1, TE is set to 1+0.3*CD. If TE is negative and TR > 1, TE is set to 1. TE is added on Card 10, Columns 71-80 of Table 1, page 107 of Volume I.

```
PAGE 129 DELETE LINES 15 AND 16

PAGE 129 DELETE LINE 27 AND REPLACE BY THE FOLLOWING
   READ (N7,50) NP,NRN,MRP,A,AA,GAMMA,EM,BP,CD,TR,NCR
   IF (NCR.EQ.0) NCR = 7

PAGE 129 DELETE LINE 43 AND REPLACE BY THE FOLLOWING
   READ (N7,70) (FF(I),I = 1,64)
   IF (FF(64).EQ.0.) FF(64) = 1.

PAGE 130 DELETE LINE 18 AND REPLACE BY THE FOLLOWING
50 FORMAT (3I5,9F5.3,I5)

PAGE 130 INSERT AFTER LINE 43 THE FOLLOWING
   IF (AIMAG(CMP(4)).EQ.0.) CMP(4) = -CD/(4.*AIMAG(CMP(2))*SQR
  1 T(CABS(ONE-CE(2))))

PAGE 131 INSERT AFTER LINE 34 THE FOLLOWING
   BB(8,NRP) = FF(64)

PAGE 131 DELETE LINES 56 AND 57 AND REPLACE BY THE FOLLOWING
   IF(AIMAG(CMP(4)).NE.0) X1(2,1)= X1(2,1)*TMP*CMP(5)/CABS(CMP
  1 (5))

PAGE 131 DELETE LINE 59 AND REPLACE BY THE FOLLOWING
   C(17) = 4.*B*REAL(CMP(2)*X1(2,1)*CMP(6))
   C(16) = CD*C(17)
   IF (C(17).EQ.0.) C(16) = 2.*CD
   BB(4,NRP) =AMAX1(0.,C(1)*(1.-TR)*C(16))
   TEMP(20) = (0.,0.)
   IF (TR.GT.1.) TEMP(20) = C(16)/(4.*CMP(2))
   TMP = X1(2,1)*TAO(2,2)-TEMP(20)
   C(16) = .5*C(16)

PAGE 132 DELETE LINES  3 THRU  5

PAGE 133 DELETE LINE 49 AND REPLACE BY THE FOLLOWING
   CALL ABORT

PAGE 133 DELETE LINE 51 AND REPLACE BY THE FOLLOWING
   CALL ABORT

PAGE 134 INSERT AFTER LINE 12 THE FOLLOWING
   DATA NBMAX /0/
   ISW = 0

PAGE 134 INSERT AFTER LINE 30 THE FOLLOWING
   IF (ISW.GT.0) CALL ADJ(1,1,A)

PAGE 134 INSERT AFTER LINE 36 THE FOLLOWING
   ISW = KK
```

```
 PAGE 134 INSERT AFTER LINE 42 THE FOLLOWING
      NBMAX = MAX0(NBMAX,NB)

 PAGE 134 DELETE LINE 49 AND REPLACE BY THE FOLLOWING
 80 WRITE (N2,130) NBMAX

 PAGE 134 DELETE LINE 58 AND REPLACE BY THE FOLLOWING
130 FORMAT (13H OUT OF PATHS/18H LONGEST PATH HAS ,I3,7H POINTS)

 PAGE 135 DELETE LINES  9 AND 10 AND REPLACE BY THE FOLLOWING
      EQUIVALENCE (CB1,C(29)),(CM1,C(19)),(CB2,C(37)),(CB3,C(33))

 PAGE 135 INSERT AFTER LINE 11 THE FOLLOWING
      CM2 = .999*CM1

 PAGE 135 INSERT AFTER LINE 41 THE FOLLOWING
      IF (NK.GT.7)  BB(7,NRP) = 0.
      TE = BB(8,NRP)

 PAGE 137 DELETE LINES 12 AND 13 AND REPLACE BY THE FOLLOWING
      SUM = 400.*XR*FLOAT(NR)/(SUM*FLOAT(NRP))

 PAGE 138 DELETE LINES  5 THRU  9 AND REPLACE BY THE FOLLOWING
      READ (N7,240) (D(J),J = 1,8)
      WRITE (N1,240) (D(J),J = 1,8)
      NK = NK-1
      WRITE (N1,250) NK, (LC(J+1), J = 1,NK)
      WRITE (N1,200) (FF(J), J = 1,62),XR,TE

 PAGE 138 INSERT AFTER LINE 40 THE FOLLOWING
250 FORMAT (16I5)

 PAGE 140 DELETE LINES 44 AND 45 AND REPLACE BY THE FOLLOWING
      IF(LL.EQ.0)CALLCUSP(X1(5,N),T(5,N),TAO(5,N),U(5,N),X3(5,N))

 PAGE 141 INSERT AFTER LINE  4 THE FOLLOWING
      SS(6) = CLOG(ETA(N))
      C(20) = AIMAG(SS(6))

 PAGE 141 DELETE LINE 41

 PAGE 141 DELETE LINE 43 AND REPLACE BY THE FOLLOWING
      SS(4) = -TT(5)*SS(3)+(U(3,N)+TEMP(20))*SS(6)+X3(1,N)

 PAGE 142 INSERT AFTER LINE 40 THE FOLLOWING
      C20 = C(20)

 PAGE 142 INSERT AFTER LINE 47 THE FOLLOWING
      C(20) = C20

 PAGE 142 INSERT AFTER LINE 55 THE FOLLOWING
      COMPLEX TEMP
      COMMON /C/ TEMP(20)
```

```
PAGE 143 DELETE LINE 11 AND REPLACE BY THE FOLLOWING
    R = TAO(4,I)*X1(2,I)-TEMP(20)

PAGE 143 INSERT AFTER LINE 20 THE FOLLOWING
    COMPLEX TEMP
    COMMON /C/ TEMP(20)

PAGE 143 DELETE LINE 32 AND REPLACE BY THE FOLLOWING
    R = TT(8)*Y(4)-TEMP(20)

PAGE 145 DELETE LINES 46 AND 47 AND REPLACE BY THE FOLLOWING
    T3(K) = 2./(1./T1(K,I-1)+1./T2(K))
10  T2(K) = 2./(1./T1(K,I)+1./T2(K))

PAGE 146 INSERT AFTER LINE  5 THE FOLLOWING
    B = B-TEMP(20)/ET

PAGE 146 DELETE LINE 27 AND REPLACE BY THE FOLLOWING
   1 U(3,I)-TEMP(20)-TEMP(20),U(4,I)))

PAGE 146 DELETE LINE 51 AND REPLACE BY THE FOLLOWING
    E = CONJG(1./ETA(N))

PAGE 147 DELETE LINE  5 AND REPLACE BY THE FOLLOWING
    GE = PF*(U(3,J-1)+U(3,J)+B*(U(4,J-1)+U(4,J)))+TEMP(20)

PAGE 147 DELETE LINE 35 AND REPLACE BY THE FOLLOWING
    B=CLOG(CMPLX(COS(C(20)),-SIN(C(20)))*ETA(I))+CMPLX(0.,C(20))
    C(20) = AIMAG(B)

PAGE 147 DELETE LINE 41 AND REPLACE BY THE FOLLOWING
    B = CLOG(CMPLX(COS(C(20)),-SIN(C(20)))*ETA(I))
    Q = REAL(B)
    C(20) = AIMAG(B)+C(20)
    Y2 = REAL(X(1))*Q-AIMAG(TEMP(20))*C(20)

PAGE 147 DELETE LINE 46 AND REPLACE BY THE FOLLOWING
    AO(4)  = (REAL(-TEMP(3)*B+Y2+X3(1,I))+SI(3))*SI(4)

PAGE 147 DELETE LINE 58 AND REPLACE BY THE FOLLOWING
    A(4,M) = (REAL(-TEMP(3)*B+Y2+X3(1,I))+SI(3))*SI(4)

PAGE 150 DELETE LINE 47 AND REPLACE BY THE FOLLOWING
C     TAIL LOGS

PAGE 150 DELETE LINE 51 AND REPLACE BY THE FOLLOWING
C     NOSE LOGS

PAGE 151 DELETE LINE 44 AND REPLACE BY THE FOLLOWING
    TT(4) = -TEMP(3)*E+(Y(1)+TEMP(20))*Q+X3(1,I)

PAGE 156 DELETE LINE  5 AND REPLACE BY THE FOLLOWING
    READ (N1,40) NP,NRN,MRP,EM,BP,TR,NK
    FAC = 1.
```

```
      IF (IABS(NRN).GT.999) FAC = .5
```

PAGE 156 INSERT AFTER LINE 11 THE FOLLOWING

```
      TE = AIMAG (GG)
```

PAGE 156 DELETE LINES 31 THRU 35 AND REPLACE BY THE FOLLOWING

```
      NRN = ISIGN(MOD(IABS(NRN),1000),NRN)
      CALL CPLOT ((3.0,2.0),-3)
C     SF WILL BE THE CHORD LENGTH IN INCHES
      SF = 5.
C     DEFAULT TAIL EXTENSION TO 1 IF TR.GT.1 OTHERWISE TE=1+.6*DY
      IF (TE.LE.0.) TE = 1.+.6*AMAX1(0.,SIGN(CC(6),1.00001-TR))
      IF ((TR.GT.1.).OR.(CC(6).LE.0.)) TE = -TE
      CALL GRF (NN,NNX,TE)
```

PAGE 156 DELETE LINES 39 AND 40 AND REPLACE BY THE FOLLOWING

```
      XMAX = 22.*FAC
      CALL CPLOT (CMPLX(.5*XMAX,4.5),-3)
      SIZE = .14
      REWIND N3
      READ (N3,90) (PG(I),I = 1,6)
      CALL CSYMBL ((-3.0,-3.5),PG,60)
      SIZE = .07
```

PAGE 156 DELETE LINE 43 AND REPLACE BY THE FOLLOWING

```
      CALL XYAXES ((0.,0.),1.+3.*XMAX/11.,1.+3.*XMAX/11.,4.4/XMAX)
```

PAGE 156 DELETE LINE 45 AND REPLACE BY THE FOLLOWING

```
      CALL XYAXES ((0.,0.),1.+XMAX/11.,1.+XMAX/11.,4.4/XMAX)
```

PAGE 156 DELETE LINE 48 AND REPLACE BY THE FOLLOWING

```
   40 FORMAT (3I5,20X,3F5.3,5X,F5.3)
```

PAGE 156 DELETE LINES 57 AND 58 AND REPLACE BY THE FOLLOWING

```
  110 FORMAT (8F10.5)
```

PAGE 157 DELETE LINE 1 AND REPLACE BY THE FOLLOWING

```
      SUBROUTINE GRF (NN,NNX,TE)
```

PAGE 157 DELETE LINE 11 AND REPLACE BY THE FOLLOWING

```
      DATA ZO,MX,LM,MXMAX,NNXMAX,K /0.,3,250,500,250,0/
```

PAGE 157 INSERT AFTER LINE 12 THE FOLLOWING

```
      XT = ABS(TE)
```

PAGE 158 DELETE LINE 35 AND REPLACE BY THE FOLLOWING

```
  160 CALL SORT (MX-1,TE)
```

PAGE 160 DELETE LINES 41 THRU 45 AND REPLACE BY THE FOLLOWING

```
      READ (N1,40) PSI(1),PX,(PSI(I), I = 2,14)
      IF (PSI(2).EQ.1H1) PSI(2) = PX
      READ (N1,50) (PSI(I), I = 15,30)
      IF (NK.GT.15) READ (N1,50) (PSI(I), I = 31,46)
```

PAGE 160 DELETE LINES 49 AND 50 AND REPLACE BY THE FOLLOWING

```
      IA = IABS(II)
      DO 10 J = 1,IA
```

PAGE 160 DELETE LINE 55 AND REPLACE BY THE FOLLOWING

```
      IF (II.GT.0) T = T*CSQRT(1.+BP*BP/(T*T),ONE)-BP
```

PAGE 161 DELETE LINE 1 AND REPLACE BY THE FOLLOWING

```
      JJ = 15 + IABS(NK)
```

PAGE 161 DELETE LINE 3 AND REPLACE BY THE FOLLOWING

```
      WRITE (N2,100) (FF(J),J = 1,64)
```

PAGE 161 DELETE LINE 9 AND REPLACE BY THE FOLLOWING

```
40    FORMAT (1XA4,2A1,A3,1XA4,9F5.3,1XA4)
```

PAGE 161 DELETE LINES 14 AND 15 AND REPLACE BY THE FOLLOWING

```
90    FORMAT (///38X,6HTAPE 7///4XA4,A1,A3,A4,F6.2,2F5.2,F6.2,2F6.3
     1 ,F7.3,F6.3,F5.2,A4/4X,16A4/4X,16A4)
```

PAGE 161 DELETE LINE 20 AND REPLACE BY THE FOLLOWING

```
     2 F5.3,3X3HDY=F5.3,3X,4HT/C=F5.3/////35X,14HTAPE 6, PATH 0/)
```

PAGE 161 DELETE LINE 25 AND REPLACE BY THE FOLLOWING

```
      SUBROUTINE SORT (N,TE)
```

PAGE 161 DELETE LINES 37 AND 38 AND REPLACE BY THE FOLLOWING

```
C     CHANGE CPOR AND CPSF TO CHANGE CP ORIGIN AND SCALE FACTOR
      DATA CPMAX,CPOR,CPSF/3.,4.5,.4/
      IF (EM.LE..7) CPOR = 4.0
      IF (EM.GE..8) CPOR = 5.0
      YMN = 0.
```

PAGE 161 INSERT AFTER LINE 40 THE FOLLOWING

```
      IF (C(3,J).GT..8) GO TO 10
```

PAGE 161 DELETE LINES 42 AND 43 AND REPLACE BY THE FOLLOWING

```
      YMX = AMAX1(YMX,C(4,J))
10    CONTINUE
      IF (TE.GT.0.) GO TO 15
C     ADD TAIL POINT ON LOWER SURFACE
      TE = 1.
      N = N+1
      DO 12 K = 1,4
12    C(K,N) = C(K,1)
      C(4,1) = C(4,1)+CC(6)
      C(5,N) = 100.
15    TC = (YMX-YMN)/TE
      CC(5) = CC(5)/TE
      CC(6) = CC(6)/TE
      NPTS = N
      IF (TE.GT.1.) NPTS = N-1
```

```
PAGE 161 DELETE LINE 48 AND REPLACE BY THE FOLLOWING
   IF (C(5,N).EQ.100.)  C(5,N) = C(5,1) + .000001
   K = IABS(NRN)
   WRITE (N3,90) RR,K,NPTS

PAGE 161 DELETE LINE 51 AND REPLACE BY THE FOLLOWING
   CALL XYAXES(CMPLX(-.5,CPOR),1.+1./CPSF,10.-YOR-CPOR,-CPSF)
   SFX = SF

PAGE 161 DELETE LINE 57 AND REPLACE BY THE FOLLOWING
   YMX = CPOR-PE(N+1)/CPSF

PAGE 162 DELETE LINES  5 AND  6 AND REPLACE BY THE FOLLOWING
   CALL CSYMBL ((-.5,-1.0),RR,60)
   SF = SFX

PAGE 162 INSERT AFTER LINE 28 THE FOLLOWING
   C(3,J) = C(3,J)/TE
   C(4,J) = C(4,J)/TE

PAGE 162 INSERT AFTER LINE 34 THE FOLLOWING
   IF (TE.GT.1.) GO TO 65

PAGE 162 DELETE LINES 37 THRU 40 AND REPLACE BY THE FOLLOWING
   IF (C(3,2).EQ.1.) CALL CSYMBL(C(3,1),15,-1)
65 ANG = 0.
   YOR = YOR+CPOR
   SS = 1./(SF*CPSF)
   CALL CSYMBL (CMPLX(C(3,1),-SS*C(5,1)),11,-1)
   DO 70 K = 2,N

PAGE 162 DELETE LINE 46 AND REPLACE BY THE FOLLOWING
80 FORMAT (3H M=,F4.3,5X,3HCL=,F5.3,5X,3HDY=F4.3,6X4HT/C=F4.3)

PAGE 163 DELETE LINE  7 AND REPLACE BY THE FOLLOWING
   BAD(S) = CSQRT(CONJG(DD)-ES(S),X)

PAGE 163 DELETE LINE  9 AND REPLACE BY THE FOLLOWING
   SF = SX*XMAX/22.

PAGE 163 DELETE LINE 30

PAGE 163 DELETE LINE 34 AND REPLACE BY THE FOLLOWING
   GO TO 45
25 IF (MOD(-NN,3).NE.1) GO TO 32

PAGE 163 DELETE LINE 58 AND REPLACE BY THE FOLLOWING
   GO TO 50
45 SIZE = .28

PAGE 164 DELETE LINE 19 AND REPLACE BY THE FOLLOWING
   IF (NP.GT.0) GO TO 120
```

```
   PAGE 164 DELETE LINE 24 AND REPLACE BY THE FOLLOWING
  120 IF (NN.GT.0) RETURN
C     SKIP PAST DATA ON TAPE1
      READ (N1,110) (X, I = 1,7)
C     READ AND PLOT PATHS
   50 READ (N1,110) KK,L,IM,((A(I,J),I = 1,IM),PSI(J), J = 1,L)
C     ****CHECK FOR END OF FILE****
      IF (EOF(N1).NE.0) RETURN
C     CHECK FOR SUPERSONIC PATH
      IF (KK.GT.0) GO TO 150
      IF (NRN.GT.0) GO TO 50
C     PLOT THE PATH OR FORK
      IF (L.LE.1) GO TO 50
      CALL CPLOT (A(5,1),3)
      DO 140 J = 2,L
  140 CALL CPLOT (A(5,J),2)
  150 IF (L.NE.1) GO TO 50
C     CHECK TO SEE IF SUPERSONIC PATHS WERE WRITTEN ON TAPE1
      IF (KK.GE.9) GO TO 50
      READ (N1,110) IA,(ETA(I),I = 1,IA)
      READ (N1,110) IB,(SEE(I),I = 1,IB)
      NN = -KK
      X = ETA(IA)
      GO TO 25

   PAGE 168 DELETE LINE  1 AND REPLACE BY THE FOLLOWING
      N = MOD(IABS(NRN),1000)

   PAGE 168 DELETE LINE  5 AND REPLACE BY THE FOLLOWING
      IF (IABS(NRN).GT.1000) GO TO 50
      CALL PLOTS (60,ID)
      RETURN
C     PLOT ON UNLINED PAPER
   50 CALL PLOTSBL (60,ID)

   PAGE 168 DELETE LINE 19 AND REPLACE BY THE FOLLOWING
     1 12.00/
```

Vol. 59: J. A. Hanson, Growth in Open Economics. IV, 127 pages. 1971. DM 16,-

Vol. 60: H. Hauptmann, Schätz- und Kontrolltheorie in stetigen dynamischen Wirtschaftsmodellen. V, 104 Seiten. 1971. DM 16,-

Vol. 61: K. H. F. Meyer, Wartesysteme mit variabler Bearbeitungsrate. VII, 314 Seiten. 1971. DM 24,-

Vol. 62: W. Krelle u. G. Gabisch unter Mitarbeit von J. Burgermeister, Wachstumstheorie. VII, 223 Seiten. 1972. DM 20,-

Vol. 63: J. Kohlas, Monte Carlo Simulation im Operations Research. VI, 162 Seiten. 1972. DM 16,-

Vol. 64: P. Gessner u. K. Spremann, Optimierung in Funktionenräumen. IV, 120 Seiten. 1972. DM 16,-

Vol. 65: W. Everling, Exercises in Computer Systems Analysis. VIII, 184 pages. 1972. DM 18,-

Vol. 66: F. Bauer, P. Garabedian and D. Korn, Supercritical Wing Sections. V, 211 pages. 1972. DM 20,-

Vol. 67: I. V. Girsanov, Lectures on Mathematical Theory of Extremum Problems. V, 136 pages. 1972. DM 16,-

Vol. 68: J. Loeckx, Computability and Decidability. An Introduction for Students of Computer Science. VI, 76 pages. 1972. DM 16,-

Vol. 69: S. Ashour, Sequencing Theory. V, 133 pages. 1972. DM 16,-

Vol. 70: J. P. Brown, The Economic Effects of Floods. Investigations of a Stochastic Model of Rational Investment Behavior in the Face of Floods. V, 87 pages. 1972. DM 16,-

Vol. 71: R. Henn und O. Opitz, Konsum- und Produktionstheorie II. V, 134 Seiten. 1972. DM 16,-

Vol. 72: T. P. Bagchi and J. G. C. Templeton, Numerical Methods in Markov Chains and Bulk Queues. XI, 89 pages. 1972. DM 16,-

Vol. 73: H. Kiendl, Suboptimale Regler mit abschnittweise linearer Struktur. VI, 146 Seiten. 1972. DM 16,-

Vol. 74: F. Pokropp, Aggregation von Produktionsfunktionen. VI, 107 Seiten. 1972. DM 16,-

Vol. 75: GI-Gesellschaft für Informatik e.V. Bericht Nr. 3. 1. Fachtagung über Programmiersprachen · München, 9-11, März 1971. Herausgegeben im Auftrag der Gesellschaft für Informatik von H. Langmaack und M. Paul. VII, 280 Seiten. 1972. DM 24,-

Vol. 76: G. Fandel, Optimale Entscheidung bei mehrfacher Zielsetzung. 121 Seiten. 1972. DM 16,-

Vol. 77: A. Auslender, Problemes de Minimax via l'Analyse Convexe et les Inégalités Variationelles: Théorie et Algorithmes. VII, 132 pages. 1972. DM 16,-

Vol. 78: GI-Gesellschaft für Informatik e.V. 2. Jahrestagung, Karlsruhe, 2.-4. Oktober 1972. Herausgegeben im Auftrag der Gesellschaft für Informatik von P. Deussen. XI, 576 Seiten. 1973. DM 36,-

Vol. 79: A. Berman, Cones, Matrices and Mathematical Programming. V, 96 pages. 1973. DM 16,-

Vol. 80: International Seminar on Trends in Mathematical Modelling, Venice, 13-18 December 1971. Edited by N. Hawkes. VI, 288 pages. 1973. DM 24,-

Vol. 81: Advanced Course on Software Engineering. Edited by F. L. Bauer. XII, 545 pages. 1973. DM 32,-

Vol. 82: R. Saeks, Resolution Space, Operators and Systems. X, 267 pages. 1973. DM 22,-

Vol. 83: NTG/GI-Gesellschaft für Informatik, Nachrichtentechnische Gesellschaft. Fachtagung „Cognitive Verfahren und Systeme", Hamburg, 11.-13. April 1973. Herausgegeben im Auftrag der NTG/GI von Th. Einsele, W. Giloi und H.-H. Nagel. VIII, 373 Seiten. 1973. DM 28,-

Vol. 84: A. V. Balakrishnan, Stochastic Differential System I. Filtering and Control. A Function Space Approach. V, 252 pages. 1973. DM 22,-

Vol. 85: T. Page, Economics of Involuntary Transfers: A Unified Approach to Pollution and Congestion Externalities. XI, 159 pages. 1973. DM 18,-

Vol. 86: Symposium on the Theory of Scheduling and Its Applications. Edited by S. E. Elmaghraby. VIII, 437 pages. 1973. DM 32,-

Vol. 87: G. F. Newell, Approximate Stochastic Behavior of n-Server Service Systems with Large n. VIII, 118 pages. 1973. DM 16,-

Vol. 88: H. Steckhan, Güterströme in Netzen. VII, 134 Seiten. 1973. DM 16,-

Vol. 89: J. P. Wallace and A. Sherret, Estimation of Product. Attributes and Their Importances. V, 94 pages. 1973. DM 16,-

Vol. 90: J.-F. Richard, Posterior and Predictive Densities for Simultaneous Equation Models. VI, 226 pages. 1973. DM 20,-

Vol. 91: Th. Marschak and R. Selten, General Equilibrium with Price-Making Firms. XI, 246 pages. 1974. DM 22,-

Vol. 92: E. Dierker, Topological Methods in Walrasian Economics. IV, 130 pages. 1974. DM 16,-

Vol. 93: 4th IFAC/IFIP International Conference on Digital Computer Applications to Process Control, Zürich/Switzerland, March 19-22, 1974. Edited by M. Mansour and W. Schaufelberger. XVIII, 544 pages. 1974. DM 36,-

Vol. 94: 4th IFAC/IFIP International Conference on Digital Computer Applications to Process Control, Zürich/Switzerland, March 19-22, 1974. Edited by M. Mansour and W. Schaufelberger. XVIII, 546 pages. 1974. DM 36,-

Vol. 95: M. Zeleny, Linear Multiobjective Programming. XII, 220 pages. 1974. DM 20,-

Vol. 96: O. Moeschlin, Zur Theorie von Neumannscher Wachstumsmodelle. XI, 115 Seiten. 1974. DM 16,-

Vol. 97: G. Schmidt, Über die Stabilität des einfachen Bedienungskanals. VII, 147 Seiten. 1974. DM 16,-

Vol. 98: Mathematical Methods in Queueing Theory. Proceedings of a Conference at Western Michigan University, May 10-12, 1973. Edited by A. B. Clarke. VII, 374 pages. 1974. DM 28,-

Vol. 99: Production Theory. Edited by W. Eichhorn, R. Henn, O. Opitz, and R. W. Shephard. VIII, 386 pages. 1974. DM 32,-

Vol. 100: B. S. Duran and P. L. Odell, Cluster Analysis. A survey. VI, 137 pages. 1974. DM 18,-

Vol. 101: W. M. Wonham, Linear Multivariable Control. A Geometric Approach. X, 344 pages. 1974. DM 30,-

Vol. 102: Analyse Convexe et Ses Applications. Comptes Rendus, Janvier 1974. Edited by J.-P. Aubin. IV, 244 pages. 1974. DM 25,-

Vol. 103: D. E. Boyce, A. Farhi, R. Weischedel, Optimal Subset Selection. Multiple Regression, Interdependence and Optimal Network Algorithms. XIII, 187 pages. 1974. DM 20,-

Vol. 104: S. Fujino, A Neo-Keynesian Theory of Inflation and Economic Growth. V, 96 pages. 1974. DM 18,-

Vol. 105: Optimal Control Theory and its Applications. Part I. Proceedings of the Fourteenth Biennual Seminar of the Canadian Mathematical Congress. University of Western Ontario, August 12-25, 1973. Edited by B. J. Kirby. VI, 425 pages. 1974. DM 35,-

Vol. 106: Optimal Control Theory and its Applications. Part II. Proceedings of the Fourteenth Biennial Seminar of the Canadian Mathematical Congress. University of Western Ontario, August 12-25, 1973. Edited by B. J. Kirby. VI, 403 pages. 1974. DM 35,-

Vol. 107: Control Theory, Numerical Methods and Computer Systems Modelling. International Symposium, Rocquencourt, June 17-21, 1974. Edited by A. Bensoussan and J. L. Lions. VIII, 757 pages. 1975. DM 53,-

Vol. 108: F. Bauer et al., Supercritical Wing Sections II. A Handbook. V, 296 pages. 1975. DM 28,-

Ökonometrie und Unternehmensforschung
Econometrics and Operations Research

Vol. I	Nichtlineare Programmierung. Von H. P. Künzi und W. Krelle unter Mitwirkung von W. Oettli. – Mit 18 Abbildungen. XV, 221 Seiten. 1962. Geb. DM 38,–
Vol. II	Lineare Programmierung und Erweiterungen. Von G. B. Dantzig. Ins Deutsche übertragen und bearbeitet von A. Jaeger. – Mit 103 Abbildungen. XVI, 712 Seiten. 1966. Geb. DM 68,–
Vol. III	Stochastic Processes. By M. Girault. – With 35 figures. XII, 126 pages. 1966. Cloth DM 28,–
Vol. IV	Methoden der Unternehmensforschung im Versicherungswesen. Von K.-H. Wolff. – Mit 14 Diagrammen. VIII, 266 Seiten. 1966. Geb. DM 49,–
Vol. V	The Theory of Max-Min and its Application to Weapons Allocation Problems. By John M. Danskin. – With 6 figures. X, 126 pages. 1967. Cloth DM 32,–
Vol. VI	Entscheidungskriterien bei Risiko. Von H. Schneeweiss. – Mit 35 Abbildungen. XII, 214 Seiten. 1967. Geb. DM 48,–
Vol. VII	Boolean Methods in Operations Research and Related Areas. By P. L. Hammer (Ivănescu) and S. Rudeanu. With a preface by R. Bellman. – With 25 figures. XVI, 329 pages. 1968. Cloth DM 46,–
Vol. VIII	Strategy for R & D: Studies in the Microeconomics of Development. By Th. Marschak, Th. K. Glennan JR., and R. Summers. – With 44 figures. XIV, 330 pages. 1967. Cloth DM 56,80
Vol. IX	Dynamic Programming of Economic Decisions. By M. J. Beckmann. – With 9 figures XII, 143 pages. 1968. Cloth DM 28,–
Vol. X	Input-Output-Analyse. Von J. Schumann. – Mit 12 Abbildungen. X, 311 Seiten. 1968. Geb. DM 58,–
Vol. XI	Produktionstheorie. Von W. Wittmann. – Mit 54 Abbildungen. VIII, 177 Seiten. 1968. Geb. DM 42,–
Vol. XII	Sensivitätsanalysen und parametrische Programmierung. Von W. Dinkelbach. – Mit 20 Abbildungen. XI, 190 Seiten. 1969. Geb. DM 48,–
Vol. XIII	Graphentheoretische Methoden und ihre Anwendungen. Von W. Knödel. – Mit 24 Abbildungen. VIII, 111 Seiten. 1969. Geb. DM 38,–
Vol. XIV	Praktische Studien zur Unternehmensforschung. Von E. Nievergelt, O. Müller, F. E. Schlaepfer und W. H. Landis. – Mit 82 Abbildungen. XII, 240 Seiten. Geb. DM 58,–
Vol. XV	Optimale Reihenfolgen. Von H. Müller-Merbach. – Mit 43 Abbildungen. IX, 225 Seiten. 1970. Geb. DM 60,–
Vol. XVI	Preispolitik der Mehrproduktenunternehmung in der statischen Theorie. Von R. Selten. – Mit 20 Abbildungen. VIII, 195 Seiten. 1970. Geb. DM 64,–
Vol. XVII	Information Theory for Systems Engineers. By L. P. Hyvärinen. – With 42 figures. VIII, 197 pages. 1970. Cloth DM 44,–
Vol. XVIII	Unternehmensforschung im Bergbau. Von F. L. Wilke. – Mit 29 Abbildungen. VIII, 150 Seiten. 1972. Geb. DM 54,–

This series aims to report new developments in mathematical economics, econometrics, operations research, and mathematical systems, research and teaching – quickly, informally and at a high level. The type of material considered for publication includes:

1. Preliminary drafts of original papers and monographs
2. Lectures on a new field, or presenting a new angle on a classical field
3. Seminar work-outs
4. Reports of meetings, provided they are
   a) of exceptional interest and
   b) devoted to a single topic.

Texts which are out of print but still in demand may also be considered if they fall within these categories.

The timeliness of a manuscript is more important than its form, which may be unfinished or tentative. Thus, in some instances, proofs may be merely outlined and results presented which have been or will later be published elsewhere. If possible, a subject index should be included. Publication of Lecture Notes is intended as a service to the international scientific community, in that a commercial publisher, Springer-Verlag, can offer a wider distribution to documents which would otherwise have a restricted readership. Once published and copyrighted, they can be documented in the scientific literature.

Manuscripts

Manuscripts should comprise not less than 100 pages.

They are reproduced by a photographic process and therefore must be typed with extreme care. Symbols not on the typewriter should be inserted by hand in indelible black ink. Corrections to the typescript should be made by pasting the amended text over the old one, or by obliterating errors with white correcting fluid. Authors receive 75 free copies and are free to use the material in other publications. The typescript is reduced slightly in size during reproduction; best results will not be obtained unless the text on any one page is kept within the overall limit of 18x26.5 cm (7x10½ inches). The publishers will be pleased to supply on request special stationery with the typing area outlined.

Manuscripts in English, German or French should be sent directly to Springer-Verlag New York or Springer-Verlag Heidelberg

Springer-Verlag, D-1000 Berlin 33, Heidelberger Platz 3
Springer-Verlag, D-6900 Heidelberg 1, Neuenheimer Landstraße 28–30
Springer-Verlag, 175 Fifth Avenue, New York, NY 10010/USA

ISBN 3-540-07029-X
ISBN 0-387-07029-X