605202

EXTENSION OF THE DOUGLAS NEUMANN PROGRAM TO PROBLEMS OF LIFTING, INFINITE CASCADES

by

JOSEPH P. GIESING

Report No. LB 31653 Revised 2 July 1964

THIS RESEARCH WAS CARRIED OUT UNDER THE BUREAU OF SHIPS FUNDAMENTAL HYDROMECHANICS RESEARCH PROGRAM, NS 715-102 ADMINISTERED BY THE DAVID TAYLOR MODEL BASIN

DOUGLAS AIRCRAFT DIVISION

EXTENSION OF THE DOUGLAS NEUMANN PROGRAM TO PROBLEMS OF LIFTING, INFINITE CASCADES

by

JOSEPH P. GIESING

Report No. LB 31653 Revised 2 July 1964

THIS RESEARCH WAS CARRIED OUT UNDER THE BUREAU OF SHIPS FUNDAMENTAL HYDROMECHANICS RESEARCH PROGRAM, NS 715-102, ADMINISTERED BY THE DAVID TAYLOR MODEL BASIN

Contract Nonr 4308(00)

1.0 SUMMARY

The Two-Dimensional Douglas Neumann Program for calculating the potential flow about bodies of arbitrary shape has been extended to handle lifting, infinite cascades. The resulting very general program allows a wide range of heretofore intractable problems to be solved. Essentially, the program can handle any problem in which the flow pattern repeats indefinitely along an axis. In the Douglas Neumann program this axis is the y-axis. This permits the calculation of the flow about a lifting or nonlifting cascade having any stagger angle and spacing and having arbitrary blade geometry. The program can also calculate the flow about more than one cascade. Thus it can handle the interaction problem of two or more parallel cascades.

The Douglas Cascade Program is compared with other theoretical methods, special analytical cases, and experimental data. Program details, among which are input-output format and FORTRAN listing are given in the appendices.

2.0 TABLE OF CONTENTS

3.0 LIST OF FIGURES

4.0 NOTATION

A	Kutta-condition matrix defined in appendix A
c	chord length of cascade blade
c_o	complex coordinate of body element midpoint
c_1	complex coordinate of body element first endpoint
c_2	complex coordinate of body element second endpoint
C_L	lift coefficient per cascade blade normalized with U the average velocity modulus, $\frac{2\Gamma}{Uc}$. If some other normalizing velocity, U_η is used the lift coefficient is $\frac{2\Gamma U}{U_\eta^2 c}$. When a set of cascades is considered, C_L is the lift coefficient for the set.
C_p	pressure coefficient having the average onset-flow velocity as the normalizing velocity
$\vec{i}$, $\vec{j}$	unit vectors in the x-and y-directions, respectively
i	complex unit, $\sqrt{-1}$
j, k	midpoint and element subscripts, respectively
K	complex source strength, $K = M + i\Gamma$
M	source strength for point source
$\vec{n}$, $\vec{t}$	unit vectors in the normal and tangential directions, respectively
NC	number of cascades considered
q	surface location of a source
s	surface location of a general point
SP	cascade spacing, the flow pattern repeats with this spacing along a prescribed axis
U	modulus of $\vec{V}$
U_I	modulus of $\vec{V}_I$
U_η	modulus of any normalizing velocity $\vec{V}_\eta$

$\vec{V}$	average onset-flow velocity $= \frac{\vec{V}_I + \vec{V}_E}{2}$
$\vec{V}(q,s)$	velocity at s due to a source at q
$\vec{V}_E$	exit velocity, velocity at $x = \infty$
$\vec{V}_I$	inlet velocity, velocity at $x = -\infty$
$\vec{W}(k,s)$	invluence of element k at point s
W_{jk}	influence of element k at the midpoint of element j, $W_{jk} = X_{jk} - iY_{jk}$
x,y	coordinates of a general point
z	x + iy, complex coordinate of a general point
α	average angle of attack measured in the counterclockwise direction
α_E	exit angle of attack
α_k	angle of attack of the kth element
α_I	inlet angle of attack
$\Delta\alpha$	cascade turning angle
Γ	circulation about an individual cascade blade
$\Delta V0_m$	effect of a uniform onset flow at zero angle of attack on the Kutta condition of the mth cascade in a system of several cascades
$\Delta V90_m$	effect of a uniform onset flow at 90° angle of attack on the Kutta condition of the mth cascade in a system of several cascades
ΔV_{mn}	effect on the Kutta condition of the mth cascade due to circulatory flow about the nth cascade in a system of several cascades
ζ	$\xi + i\eta$, complex coordinate of a source
θ	stagger angle (see figure 3), measured clockwise
σ	surface source distribution
γ	airfoil trailing-edge angle

5.0 INTRODUCTION

The Douglas Neumann Program is a powerful tool for determining the potential flow about one or more arbitrary two-dimensional lifting bodies. The Neumann program is rigorous in the sense that the exact solution is approached in the limit as the number of points describing the body goes to infinity. This powerful method has now been extended to calculate the flow about infinite two-dimensional lifting cascades. Cascade data are used to advantage when working with compressor stages, turning vanes, or propeller blades. A. J. Acosta (Calif. Inst. of Tech.) and H. P. Linhardt (reference 1) have concluded that the use of cascade theory to predict propeller characteristics is accurate even when the propeller is extreme in configuration. These authors tested a low-aspect-ratio axial-flow pump propeller. Comparison of the experimental results with three-dimensional vortex and two-dimensional cascade theory showed the simple cascade theory to be superior.

A cascade is defined as simply a series of identical bodies equally spaced and identically oriented. There are no restrictions on the body, shape, spacing, and stagger angle. The cascade of bodies may be lifting or nonlifting. In general, the program can handle any problem in which the flow pattern repeats along an axis from plus to minus infinity. Figure 3 shows a double cascade along with a graphical representation of the cascade parameters.

Shown in figures 4, 5, 6, and 7 are some of the extreme configurations a cascade might take, given the flexibility this program affords. The program can also be used to calculate the flow past a series of cascade stages, i.e., more than one cascade. The stages, however, cannot move relative to each other, since this would be an unsteady-flow problem.

In some cases the effect of boundary-layer displacement thickness on the cascade blade is important. In these cases the displacement thickness can be added to the blade and the resulting thicker blade can be used in the cascade program. This displacement-thickness technique has been tried successfully on a single isolated airfoil.

The Douglas Newmann program with its cascade modification calculates the velocity and pressure distribution normalized to average velocity and lift coefficient and moment coefficient per cascade blade. Also calculated are the inlet and exit velocities and the cascade turning angle. Appendix C gives the FORTRAN listing and all the details of the input and output of the Douglas Cascade Program.

6.0 THEORY

The technique employed by the Neumann Program to solve the fluid-flow problem is to apply a source distribution of appropriate strength on the surface of the body in such a way that the flow normal to the surface of the body is either zero or prescribed. This technique is described in great detail in references 2, 3, and 4. When the Neumann boundary condition is applied, an integral equation in source strength σ is obtained. This integral equation is

$$-\vec{V}_\infty \cdot \vec{n} = \sigma(s) + \int_{BODY} \sigma(q)\, A(q,s)\, dq \tag{1}$$

where $A(q,s) = \vec{n} \cdot \vec{V}(q,s)$ and V_∞ is the onset flow.

In the unmodified program $V(q,s)$ is the familiar velocity at s due to a unit source at q. If x,y are coordinates associated with s and ξ, η with q, then

$$\vec{V}(q,s) = \vec{i}\, V_x + \vec{j}\, V_Y$$

where

$$V_x = \frac{(x-\xi)}{(x-\xi)^2 + (y-\eta)^2} \tag{2}$$

$$V_Y = \frac{(y-\eta)}{(x-\xi)^2 + (y-\eta)^2}$$

or, in complex notation,

$$V_x - i\, V_Y = \frac{1}{Z(s) - \zeta(q)}$$

To modify the program to handle infinite cascades a new velocity at a point s due to a unit source at q in a cascade is used.

This "new" velocity is the sum of the velocities due to a row of sources equally spaced along the y axis. The sum is the series representation of the hyperbolic cotangent function (see Lamb, reference 5, page 71). Thus the cascade velocity is

$$V_X - iV_Y = \frac{1}{2SP} \coth \left\{ \frac{\pi}{SP} \left[Z(s) - \zeta(q) \right] \right\}$$

or

$$V_X = \frac{\frac{1}{2SP} \cosh\left[\frac{\pi}{SP}(x-\xi)\right] \sinh\left[\frac{\pi}{SP}(x-\xi)\right]}{\left\{\sin\left[\frac{\pi}{SP}(y-\eta)\right]\right\}^2 + \left\{\sinh\left[\frac{\pi}{SP}(x-\xi)\right]\right\}^2} \tag{3}$$

$$V_Y = \frac{\frac{1}{2SP} \cos\left[\frac{\pi}{SP}(y-\eta)\right] \sin\left[\frac{\pi}{SP}(y-\eta)\right]}{\left\{\sin\left[\frac{\pi}{SP}(y-\eta)\right]\right\}^2 + \left\{\sinh\left[\frac{\pi}{SP}(x-\xi)\right]\right\}^2}$$

where SP is the cascade spacing. The cascade vortex velocity is (Lamb, page 244) simply the cascade source velocity rotated 90° clockwise.

The technique employed to solve the integral equation given in (1) is to approximate the body surface by straight-line elements. The source strength is assumed constant along any one element, but it varies from element to element. Now if the Neumann boundary condition is applied at the midpoints of each of these elements, equation (1) can be written as

$$-\vec{V}_{\infty_j} \cdot \vec{n}_j = \sigma_j + \sum_{k=1}^{N} \sigma_k \int_{\text{ELEM. }k} A_j(q)\, dq \tag{4}$$

Here we note that $A(q,s)$ is now written $A_j(q)$ since the positions of the general points are now fixed as the element mid-points. Referring to the definition of $A(q,s)$ given for (1), we write

$$A_j(q) = \vec{n} \cdot \vec{V}_j(q) \tag{5}$$

$\vec{V}_j(q)$ is the velocity at the mid-point of the jth element due to a unit source at q.

Let the quantity $\vec{W}_{jk}$ be defined as

$$\vec{W}_{jk} = \int_{\text{ELEM. } k} \vec{V}_j(q)\, dq = \vec{i}X + \vec{i}Y \tag{6}$$

For convenience the complex form of $\vec{W}(q,s)$ or $\vec{W}_j(q)$ and $\vec{V}(q,s)$ or $\vec{V}_j(q)$ will be adopted and used henceforth. Substituting the cascade-velocity source function found in (3) into (6), we have the following

$$W_{jk} = X - iY = \int_{\text{ELEM. } k} V_j(q)\, dq = \int_{\text{ELEM. } k} \frac{1}{2SP} \coth\left\{\frac{\pi}{2P}\left[Z_j - \zeta(q)\right]\right\} dq \tag{7}$$

The kth element over which the integration is to be performed is shown in figure 1. From figure 1 the following relations are evident

$$\begin{aligned} \zeta(q) &= C_o + q\, e^{i\alpha_k} \\ d\zeta &= dq\, e^{i\alpha_k} \end{aligned} \tag{8}$$

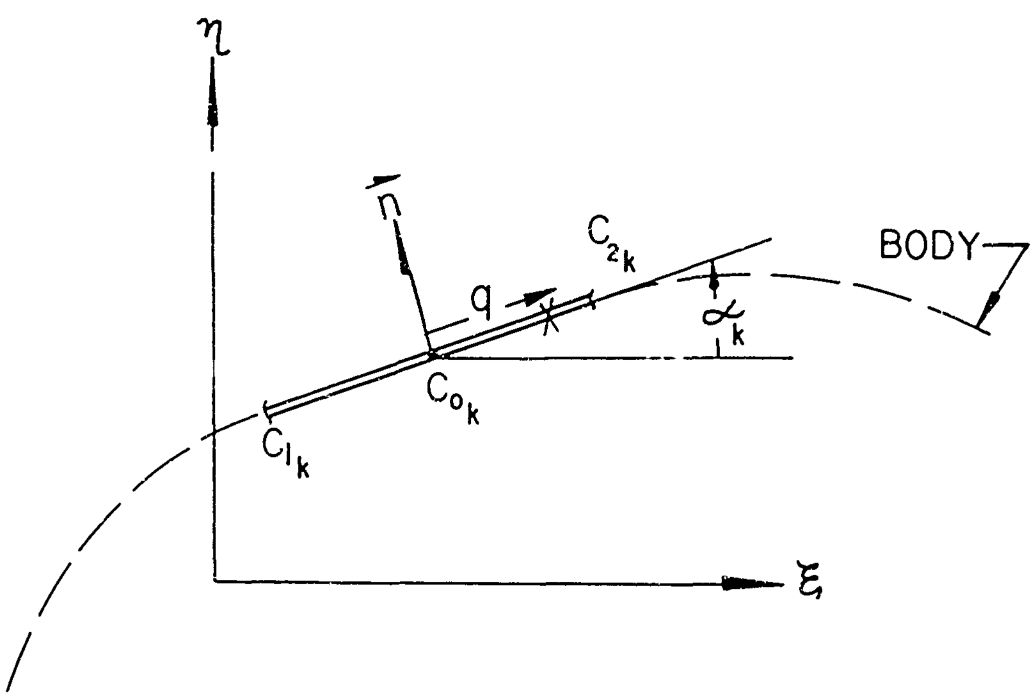


Figure 1.- A typical straight line element of the body surface.

If the relations of (8) are used, equation (7) may be rewritten as follows:

$$W = e^{-i\alpha_k} \int_{C_{1k}}^{C_{2k}} \frac{1}{2SP} \coth\left[\frac{\pi}{SP}(z_j - \zeta)\right] d\zeta$$

which upon integration becomes

$$W = X - iY = \frac{e^{-i\alpha_k}}{2\pi} \ln\left\{\frac{\sinh\left[\frac{\pi}{SP}(z_j - C_{1k})\right]}{\sinh\left[\frac{\pi}{SP}(z_j - C_{2k})\right]}\right\} \tag{9}$$

Equation (9) is the basic cascade source function used in the Douglas Cascade Program. It represents the complex velocity at the jth element midpoint z_j, due to the kth source element in cascade.

If we now use the definition given in equation (6), equation (4) can be rewritten thus:

$$-\vec{V}_{\infty j} \cdot \vec{n}_j = \sum_{k=1}^{N} \vec{W}_{jk} \cdot \vec{n}_j \, \sigma_k = \sum_{k=1}^{N} A_{jk} \, \sigma_k \tag{10}$$

Equation (10) is then solved using equation (9) for the unknown σ_k. Once the σ_k values are known, the velocity and pressure anywhere in the flow field can be calculated.

To solve the general case of a lifting cascade at any angle of attack, "basic" flows are calculated and superimposed in such a way that the correct angle of attack is obtained and the Kutta condition is satisfied. These "basic" flows are the following:

1) Flow at zero angle of attack.
2) Flow at 90° angle of attack.
3) Circulatory flow for each cascade.

Equation (10) is solved for each one of the basic onset flows; here V_∞ is the onset flow in the equation. Superposition of solutions is possible because the potential equation is linear and the boundary condition on the cascade blades is homogeneous. The details of the superposition technique are given in Appendix A.

In a cascade there is an infinite number of airfoils or blades, each having a circulation. Therefore since the cascade runs along the y-axis, there exists an upwash an infinite distance upstream of the lifting cascade and a downwash an infinite distance downstream. At these distances, the lifting airfoils act like a row of equally spaced vortices. The magnitude of the upwash and downwash due to this row of vortices can be deduced from Lamb (reference 5) as

$$V_{up} = -V_{down} = \frac{\Gamma}{2SP} = \frac{U_\eta^2 C_{L_\eta} C}{4U\, SP} \tag{11}$$

where Γ is the circulation per cascade body and SP is the cascade spacing. U_η is any convenient normalizing velocity. In the Douglas program U_η is just U. (When more than one cascade is involved, Γ is replaced by $\sum_{m=1}^{NC} \Gamma_m$; NC is the number of cascades. Also C_L is the lift coefficient for the set of cascades.) The cascade therefore turns the flow. The upwash and downwash and the cascade turning angle are shown in figure 3. For a cascade, the lift vector is normal to the average onset-flow velocity vector. The inlet and exit velocities can be determined by using (11) for V_{up} and by referring to the vector diagram of figure 3. All velocities will be normalized with the modulus of the average velocity. The inlet and exit velocities are written as follows:

$$\vec{V}_I = \frac{\vec{V}}{U} + \vec{j}\,\frac{\Gamma}{2SPU} = \frac{\vec{V}}{U} + \vec{j}\,\frac{C_L c}{4SP} \tag{12}$$

$$\vec{V}_E = \frac{\vec{V}}{U} - \vec{j}\,\frac{\Gamma}{2SPU} = \frac{\vec{V}}{U} - \vec{j}\,\frac{C_L c}{4SP}$$

The inlet and exit angles of attack (see figure 2) can be written as

$$\alpha_I = \tan^{-1}\left[\frac{\sin\alpha + \frac{\Gamma}{2SP}}{\cos\alpha}\right] = \tan^{-1}\left[\frac{\sin\alpha + \frac{C_L c}{4SP}}{\cos\alpha}\right]$$

$$\alpha_E = \tan^{-1}\left[\frac{\sin\alpha - \frac{\Gamma}{2SP}}{\cos\alpha}\right] \tag{13}$$

Thus the turning angle is

$$\Delta_\alpha = \alpha_I - \alpha_E = \tan^{-1}\left[\frac{\frac{\Gamma}{SP}\cos\alpha}{1 - \left(\frac{\Gamma}{2SP}\right)^2}\right] = \tan^{-1}\left[\frac{\frac{C_L c}{2SP}\cos\alpha}{1 - \left(\frac{C_L c}{4SP}\right)^2}\right] \tag{14}$$

The <u>average</u> onset flow modulus U is used in the definition of all of the hydrodynamic coefficients and variables in the Douglas Cascade Program. In some cases it may be desirable to normalize the velocities involved with the inlet velocity modulus, U_I. In that case we write

$$\frac{V_{up}}{U_I} = \left(\frac{C_{L_I} c}{4SP}\right)\left(\frac{U_I}{U}\right)$$

$$C_{L_I} = \frac{2U\Gamma}{U_I^2 c}$$

$$\text{MOD.}\left(\frac{\vec{V}_I}{U_I}\right) = 1 \tag{15}$$

$$\frac{\vec{V}_E}{U_I} = \frac{\vec{V}_I}{U_I} - \vec{j}\left(\frac{C_{L_I} c}{2SP}\right)\left(\frac{U_I}{U}\right)$$

The exit angle of attack can be obtained from the exit velocity vector. Also the turning angle can be calculated, since the inlet and exit angle of attack are known. To convert from the system normalized with the average velocity to the one using the inlet velocity as the normalizing factor, the following set of conversions can be used:

$$C_{L_I} = \left(\frac{U}{U_I}\right)^2 C_L$$

$$C_{p_I} = 1 - \left(\frac{V}{U}\right)^2\left(\frac{U}{U_I}\right)^2 \tag{16}$$

$$\frac{U_I}{U} = \frac{\left(\tan\alpha_I - \tan\alpha_E\right)\left(\frac{2SP}{c}\cos\alpha_I\right)}{C_{L_I}}$$

7.0 EXAMPLES AND COMPARISONS

In order to give an idea of the wide class of problems the Program can handle, several cases have been calculated. Figures 3, 4, 5, 6, and 7 present a range of configurations and geometries that can be handled with ease.

To show the accuracy of the Program, comparisons with exact solutions and experimental data are presented. Figures 8, 9, 10, and 11 give these comparisons.

Figure 12 compares the Douglas Cascade Program with a theory developed by Garrick, reference 7.

These figures are now described in detail.

7.1 TANDEM CIRCLES

To illustrate the fact that the Cascade Program can be used for problems not necessarily associated with the usual lifting cascade, the flow about an infinite number of nonlifting bodies in tandem was calculated (see figure 4). In this particular case the bodies are circles; however, any body can be used.

This tandem arrangement can be recognised as simply a cascade at 90° angle of attack rotated 90° so that the axis of repetition, the y-axis, becomes the x-axis. Recall that any problem where the flow pattern repeats indefinitely in one direction can be handled by the Douglas Program.

7.2 MULTIPLE-CASCADES

Figures 5 and 6 are included to illustrate the multiple-cascade capability.

Shown in figure 5 are three parallel cascades. The central cascade has circulation, while the other two do not. It would be a mistake to say that because they have no circulation the two outside cascades are nonlifting. The resultant forces in an interaction problem are not determined by the circulation alone.

It is true that the total lift of the entire set is proportional to the total circulation of the set and that the net drag of the set is zero. However, this is a gross effect for the set and does not hold for the individual members. As is noted in figure 5 the total drag coefficient of the first and last cascade is the negative of the drag coefficient on the second cascade. The total lift coefficient of the cascade set is 22.7.

The trailing edge of the central circle is at $\cdot 30^{\circ}$ from the horizontal and the inlet angle of attack for the cascade set is 43°.

For multiple cascades several limitations must be kept in mind. First, the cascades must be parallel. Second, there can be only one spacing associated with all of the cascades. In general, all the cascades of a set must have the same spacing; however, certain exceptions to this can be made. One of these exceptions is illustrated in figure 6. In this case the spacing of the second cascade is exactly half the spacing of the first. This effect was obtained by putting <u>two</u> cascade bodies in the second cascade. The two bodies of the second cascade are placed one above the other and spaced at exactly one-half the spacing of the first cascade. This process can easily be generalized to many cascades, and the result is that the spacing ratio of two or more cascades will be a rational number. It is noted that there is still only <u>one</u> spacing associated with both cascades of figure 6. This spacing is shown in the figure.

Since neither cascade has circulation, the set has no net lift. However, bodies 2 and 3, of the second cascade, have lift equal but opposite in sign thus the total lift is zero. The drag of the first cascade is equal to the thrust of the second. The drag coefficient is 0.546.

7.3 ANALYTIC TEST CASES

The Douglas Neumann Program with its cascade modification can calculate the flow about any cascade profile. To test this claim, the flow was calculated about the blade section, profile "A", shown in figure 7. This extreme shape was generated from a circle by using a series of conformal transformations. Appendix B gives the details of these transformations. The pressure coefficients obtained from the transformation method are exact. Figures 8a and 8b give the exact and calculated pressure distributions over the blade in a cascade at two different angles of attack. Generally the agreement is good for such an extreme blade shape; however, if greater accuracy were desired, more coordinate points describing the body would be needed. To prevent crowding, not all of the points calculated by the program are shown in these figures. For an example where all of the points are plotted see figures 12a and 12b.

The blade section shown, profile "C" in figure 9, was obtained through a series of conformal transformations in the same manner as the blade of figure 7 (see Appendix B). Also, figures 9a and 9b show the exact and calculated pressure distributions over the blade in cascade at two angles of attack. Agreement between analytic and calculated pressure distributions is better than that of figures 8a and 8b as is to be expected, since the shape is less extreme.

7.4 EXPERIMENTAL COMPARISONS

Shown in figures 10a, 10b and 10c are calculated and experimentally obtained pressure distributions for an NACA 65-010 cascade blade at three values of lift coefficient. The experimental data were taken from reference 6. Figure 11 shows the experimental and calculated lift coefficients as functions of the "effective" angle-of-attack for the cascade.

The "effective" angle of attack is simply the stagger angle plus the inlet angle of attack.

The calculated and experimental pressure distributions agree quite well except for a small region near the trailing edge. The descrepancies near the trailing edge are probably due to boundary-layer-thickness effects.

7.5 COMPARISON OF THE DOUGLAS METHOD WITH A METHOD DUE TO I. E. GARRICK

I. E. Garrick (reference 7) applied a straight-line cascade transformation in series with a Theodorsen-type transformation to map an airfoil in cascade onto a single circle. Once the transformation was obtained, the pressure distribution over the cascade airfoil could be found. Accuracy can easily be lost in the process of carrying out the operations involved, because of the peculiar nature of the straight-line cascade transformation. This may explain some of the discrepancies between the two methods.

Garrick calculated the flow over an NACA 4412 airfoil in cascade. Shown in figures 12a and 12b are pressure distributions calculated by Garrick and Douglas for the 4412 cascade at two lift coefficients. In figures 12a and 12b all points at which the Program executed a calculation are shown. This will serve as a indication of the number of coordinates used by the Program in the solution of the problem.

8.0 ACKNOWLEDGEMENTS

I wish to acknowledge the contribution made by Mr. Thomas Clissold who constructed the computer program for the Cascade Method and wrote the program input, output description found in Appendix C.

APPENDIX A

Basic Solutions

Each "basic solution" is a solution of the potential-fluid-flow problem for a cascade of bodies with a given onset flow. Once determined, these "basic solutions" are combined in such a way that the desired flow at infinity is obtained and the Kutta condition on each cascade is satisfied. The "basic solutions" needed for this combination procedure are:

1) Flow about the several cascades at zero angle of attack,
2) Flow about the several cascades at 90° angle of attack,
3) Flow about the cascades due to the presence of circulation in the <u>first</u> cascade,
4) Flow about the cascades due to the presence of circulation in the <u>second</u> cascade, and so forth.

To obtain circulation about a cascade profile, a unit vortex is placed within the profile. This vortex serves as the onset flow for the circulatory "basic solution".

In a cascade the flow pattern repeats indefinitely along one axis, in this case the y-axis. Therefore it is only necessary to deal with one of the cascade blades. If the Kutta condition holds for one blade of a cascade, it holds for all of them. Thus, in dealing with the cascade, only one blade will be considered.

Each basic solution violates the Kutta condition. A measure of this violation is the difference ΔV of the tangential velocities above and below the trailing edge. The ΔV's of the basic solutions are denoted: $\Delta V0$, $\Delta V90$, and ΔV_{mn}; where $m = 1, 2 \cdots NC$ and $n = 1, 2 \cdots NC$. Here ΔV_{mn} is the effect on the Kutta condition of the mth cascade due to circulation in the nth cascade. NC is the number of cascades. Added together, the basic solutions must satisfy the Kutta condition on each cascade and also give the desired flow at infinity.

COMBINATION OF BASIC SOLUTIONS

For a set of isolated airfoils the following set of linear equations in the unknown circulation strengths satisfies the Kutta conditions on each airfoil. The uniform onset flow at infinity is of speed U and angle of attack α. The set of linear equations with the unknown Γ_m is

$$\begin{array}{l}
\Delta VO_1 U\cos\alpha + \Delta V90_1 U\sin\alpha + \Delta V_{11}\Gamma_1 + \Delta V_{12}\Gamma_2 + \cdots = 0 \\
\Delta VO_2 U\cos\alpha + \Delta V90_2 U\sin\alpha + \Delta V_{21}\Gamma_1 + \Delta V_{22}\Gamma_2 + \cdots = 0 \\
\vdots \\
\Delta VO_{NC} U\cos\alpha + \Delta V90_{NC} U\sin\alpha + \Delta V_{NC1}\Gamma_1 + \Delta V_{NC2}\Gamma_2 + \cdots = 0
\end{array}$$

or, in matrix form,

$$\begin{bmatrix} \Delta V_{11} & \Delta V_{12} & \cdots \\ \Delta V_{21} & & \\ \vdots & & \\ \Delta V_{NC1} & & \end{bmatrix} \begin{Bmatrix} \Gamma_1 \\ \Gamma_2 \\ \vdots \\ \Gamma_{NC} \end{Bmatrix} = -U \begin{Bmatrix} \Delta VO_1\cos\alpha + \Delta V90_1\sin\alpha \\ \Delta VO_2\cos\alpha + \Delta V90_2\sin\alpha \\ \vdots \\ \Delta VO_{NC}\cos\alpha + \Delta V90_{NC}\sin\alpha \end{Bmatrix} \qquad (A1)$$

where

α = angle of attack of the set of airfoils

Γ = circulation strength per airfoil

For a set of cascades there is a complicating factor, and (A1) cannot be used directly. The angle α, the average angle of attack, is not necessarily known. What is known is any one of the following:

1) Cascade or set of cascades at a prescribed average angle of attack (α).
2) Cascade or set of cascades at a prescribed inlet angle of attack (α_I).
3) Cascade or set of cascades at a prescribed lift coefficient (C_L).
4) Cascade or set of cascades with a prescribed turning angle ($\Delta\alpha$).

These cases are mutually exclusive. For example: if α_I is prescribed the other three cannot be prescribed.

For case (1) equation (A1) can be used directly. For cases (2), (3), and (4) there is an additional unknown, namely, α. Thus an additional equation must be found for these cases. The additional equations needed for cases (2), (3), and (4), respectively, are:

$$\cos\alpha \tan\alpha_I - \sin\alpha - \frac{1}{2SP}\sum_{m=1}^{NC}\Gamma_m = 0 \tag{A2-2}$$

$$\frac{2}{Uc}\sum_{m=1}^{NC}\Gamma_m - C_L = 0 \tag{A2-3}$$

$$\tan(\Delta\alpha) = \frac{\frac{\cos\alpha}{SP}\sum_{m=1}^{NC}\Gamma_m}{1-\left[\frac{1}{2SP}\sum_{m=1}^{NC}\Gamma_m\right]^2} = 0 \tag{A2-4}$$

With some rearranging of the linear equations, (A2-2) and (A2-3) can be incorporated into the set of linear equations (A1). However, (A2-4) cannot be incorporated and must be solved iteratively together with (A1).

When α_I is the desired input, equation (A2-2) can be incorporated into (A1). The resulting matrix equation is

$$\begin{bmatrix} U\Delta V90_1 & \Delta V_{11} & \Delta V_{12} \cdots \Delta V_{2NC} \\ \vdots & \vdots & \vdots \quad\quad \vdots \\ U\Delta V90_{NC} & \Delta V_{NC1} & \Delta V_{NC2} \cdots \Delta V_{NCNC} \\ 1 & \frac{1}{2SP} & \frac{1}{2SP} \cdots \frac{1}{2SP} \end{bmatrix} \begin{Bmatrix} \tan\alpha \\ \Gamma_1/\cos\alpha \\ \Gamma_2/\cos\alpha \\ \vdots \\ \Gamma_{NC}/\cos\alpha \end{Bmatrix} = \begin{Bmatrix} -U\Delta VO_1 \\ \vdots \\ -U\Delta VO_{NC} \\ \tan\alpha_I \end{Bmatrix} \tag{A3}$$

When the lift coefficient C_L is input, equation (2-3) can be used with (A1) in the following manner. First we may write

$$\begin{bmatrix} U\Delta V90_1 & \Delta V_{11} & \Delta V_{12} & \cdots & \Delta V_{1NC} \\ \vdots & \vdots & \vdots & & \vdots \\ U\Delta V90_{NC} & \Delta V_{NC1} & \Delta V_{NC2} & & \Delta V_{NCNC} \\ 0 & \frac{2}{C} & \frac{2}{C} & \cdots & \frac{2}{C} \end{bmatrix} \begin{Bmatrix} \tan\alpha \\ \Gamma_1/\cos\alpha \\ \Gamma_2/\cos\alpha \\ \vdots \\ \Gamma_{NC}/\cos\alpha \end{Bmatrix} = \begin{Bmatrix} -U\Delta VO_1 \\ -U\Delta VO_2 \\ \vdots \\ -U\Delta VO_{NC} \\ C_L/\cos\alpha \end{Bmatrix} \quad (A4)$$

In the above set of linear equations notice that $\cos\alpha$, with α an unknown, still appears on the right-hand side of the equation. To solve this set of equations, define A as

$$A = \begin{bmatrix} U\Delta V90_1 & \Delta V_{11} & \Delta V_{12} & \cdots \\ \vdots & \vdots & & \\ U\Delta V90_{NC} & \Delta V_{NC1} & \cdots & \\ 0 & \frac{2}{C} & \cdots & \frac{2}{C} \end{bmatrix}$$

Let A^{-1}, the inverse of A, be defined as

$$A^{-1} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1NC} \\ a_{21} & & & \vdots \\ \vdots & & & \\ a_{NC1} & \cdots & & a_{NCNC} \end{bmatrix}$$

Then (A4) becomes

$$\begin{Bmatrix} \tan\alpha \\ \Gamma_1/\cos\alpha \\ \vdots \\ \Gamma_{NC}/\cos\alpha \end{Bmatrix} = A^{-1} \begin{Bmatrix} -U\Delta VO_1 \\ -U\Delta VO_L \\ \vdots \\ -U\Delta VO_{NC} \\ C_L/\cos\alpha \end{Bmatrix}$$

and therefore

$$\tan\alpha = \Delta VO_1 \, a_{11} + \Delta VO_2 \, a_{12} + \cdots (C_L/\cos\alpha) \, a_{1NC} \quad (A5)$$

Equation (A5) can then be used to solve for $\cos\alpha$. Equations (A4) and (A5) represent the solution when C_L is input.

As has been stated before, (A1) and (A2-4) must be solved iteratively when the turning angle $\Delta\alpha$ is desired.

When dealing with the usual case of a single cascade, the equations become very simple. Equation (A1) reduces to one equation and the matrix of (A3) is of order two and its solution is trivial.

APPENDIX B

Analytic Cascade Test Cases

In order to check the Douglas Cascade Program, several exact cascade solutions were generated by conformal transformation methods. Specifically, a circle was mapped by a series of transformation functions into a profile in cascade. The final shape could not be determined exactly ahead of time but certain characteristics could be controlled.

The first in the series of mapping functions was the Karman-Trefftz transformation. This function maps a circle in the complex S-plane onto an airfoil shape in the complex Q-plane. The transformation is

$$\frac{Q - rd}{Q + rd} = \left(\frac{S - d}{S + d}\right)^r \qquad \text{(B1)}$$

The real constant r determines γ, the trailing-edge angle, by the relation

$$\gamma = \pi \, (2 - r) \qquad \text{(B2)}$$

The final airfoil shape is determined by the location of the circle with reference to the S coordinate system. The real constant d is the distance from the origin to the intersection of the circle with the real S axis. This intersection maps to the trailing edge of the airfoil. This mapping is very similar to the Joukowski transformation, in that a displacement of the circle toward the negative real S-axis produces thickness and a displacement toward the positive imaginary S-axis produces positive camber. Thus these three parameters determine the Karman-Trefftz airfoil: (1) the trailing-edge-angle constant, r; (2) the radius of the circle, a; and (3) the location of the center of the circle in the S coordinate system. For profile "A" $r = 1.8$, a = 1.0 and the center location is $(-0.1, 1/\sqrt{2}\,i)$. For profile "C" r = 1.85, a = 1.0, and the center location is (-0.02, 0.51).

The second and last transformation takes the Karman-Trefftz airfoil in the Q-plane into some profile in a cascade in the final z-plane by using the following transformation:

$$z = \ln\left(\frac{Q - A}{Q - B}\right) \tag{B3}$$

The spacing of the resulting cascade is always 2π, and A and B are complex quantities. The singular points at A and B must be outside of the airfoil and must not touch its surface. Essentially, the points A and B in the Q-plane go to plus and minus infinity, respectively, in the z-plane. The closer the points are to the airfoil surface, the longer the cascade profile. For convenience of computation, the points A and B are not selected in the airfoil plane but in the circle plane. In the circle plane they are called A' and B'. For profile "A"

A' = 0.[illegible]57 + 0i, B' = -1.1 + 0.[illegible]i. For profile "C"

A' = 1.0[illegible]60[illegible] + 0i, B' = -1.08[illegible]0[illegible] + 0i. To determine the coordinates of A and B the transformation of (B6) is used with A' and B' in place of S and A and B in place of Q.

To obtain the flow field in the z-plane, the complex potential must be differentiated.

If F is this complex potential and w the complex velocity,

$$w(z) = \frac{dF(z)}{dz} = \frac{dF(S)}{dS}\,\frac{dS}{dQ}\,\frac{dQ}{dz} \tag{B4}$$

The term $\frac{dS}{dQ}$ can be obtained from (B1)

$$\frac{dS}{dQ} = \frac{S^2 - d^2}{Q^2 - r^2 d^2} \tag{B5}$$

The term $\frac{dQ}{dz}$ can be obtained from (B3)

$$\frac{dQ}{dz} = \frac{Q^2 - Q(A + B) + AB}{A - B} \tag{B6}$$

The derivative $\frac{dF(S)}{dS}$ is the complex velocity in the circle plane; i.e., it represents the flow field about the circle. The question to consider is: in what flow field is the circle immersed? The answer lies in the transformation (B3), i.e., the mapping function that takes the Karman-Trefftz airfoil to a profile in cascade. It can be shown that a source at B and a sink at A in the airfoil plane give a uniform flow from minus to plus infinity in the cascade plane. It can also be shown that a vortex at each of these points gives a vertical component to the uniform flow in the cascade plane. Therefore the following relations hold:

$$\begin{aligned}
U_I\cos\alpha_I &= \text{source strength, } M, \text{ at } B \\
U_I\sin\alpha_I &= \text{vortex strength, } \Gamma, \text{ at } B \\
U_E\cos\alpha_E &= \text{sink strength, } M, \text{ at } A \\
-U_E\sin\alpha_E &= \text{vortex strength, } \Gamma, \text{ at } A
\end{aligned} \tag{B7}$$

To preserve continuity of mass in the cascade plane, the source strength at B must be equal to the sink strength at A. The flow in which the cylinder is immersed is then generated by a source and vortex at B and a sink and vortex at A.

It can be shown that the circulation about the cascade profile is the negative difference of the strengths of the vortexices located at A and B. Thus all circulations add up to zero. The onset flow has now been determined and therefore the flow about a circle in this onset flow can be determined. The complex potential function for this flow is

$$F = K_1 \ln(S-A) + \overline{K}_1 \ln\left(S - \frac{a^2}{\overline{A}}\right) + K_2 \ln(S-B) + \overline{K}_2 \ln\left(S - \frac{a^2}{\overline{B}}\right) \qquad (B8)$$

where a is the circle radius and K is the complex source strength $M + \Gamma i$. M is the source mass flow and Γ is the circulation. The complex velocity is just the derivative of F and is

$$\frac{dF}{dS} = \frac{K_1}{S - A} + \frac{\overline{K}_1}{S - \frac{a^2}{\overline{A}}} + \frac{K_2}{S - B} + \frac{\overline{K}_2}{S - \frac{a^2}{\overline{B}}} \qquad (B9)$$

$$K_1 = U_1(\cos\alpha_I + i \sin\alpha_I), \quad K_2 = -U_I(\cos\alpha_I + i \frac{U_E}{U_I} \sin\alpha_E)$$

If the expressions for the derivities of (B[illegible]), (B[illegible]) and (B9) are substituted into equation (A9) and if S takes the values of the coordinates of the circle the velocity over the surface of the cascade body in the z-plane can be determined. In these formulas the velocities are normalized with the inlet velocity modulus U_I.

APPENDIX C

Program Input and Output

A summary of the Program input is presented before the detailed explanation is given.

As is stated in the text, one or more cascade bodies of arbitrary shape can be handled by the Program. However, there are three practical restrictions on the input, as follows:

(1) The cascade body or bodies must be of finite thickness.

(2) The maximum number of bodies with circulation is 8.

(3) The maximum number of points describing all of the bodies and off-body points is 500.

Each body considered is a lifting body with a stagnation point at the trailing edge, unless otherwise specified in the input. It is assumed in the program that the first coordinate point input is the trailing edge. The surface coordinates are input, starting from the trailing edge and progressing around the body in the clockwise direction. The last point of a body must be the first point repeated. However, if the body is non-closed and non-lifting the last point is not the first point repeated.

In addition to calculating the flow on the cascade-body surface, the Program can calculate the flow at points in the flow field. The coordinates of these off-body points are input in the same manner as the coordinates of the cascade body.

All coordinates, whether body-surface or off-body coordinates, may be scaled, rotated, and translated. The Program executes these operations in the order named.

For diagnostic purposes the two matrices $A_{jk} = \vec{W}_{jk} \cdot \vec{n}$ and $B_{jk} = \vec{W}_{jk} \cdot \vec{t}$ can be printed out.

Input Data

Each case must consist of a header card, case control data, body control data, and coordinate data. The header card contains the description of the case, control flags, and case number. The case control data specify certain constants used in the computation. The body control data specify the amount of coordinate data being input and constants used to modify the coordinate data. The coordinate data describe either the two-dimensional cross section of the body or off-body points. All data cards must contain sequence numbers in card columns 77 through 80, so that the data may be sorted. If any data cards are found to be out of sequence, the program will discontinue execution.

The data must be arranged in the following order (see figure 2):

1. Header Card (1 card)
2. Case Control Card (1 card)
3. Body Control Cards (2 cards)
4. Coordinate Data Cards (Variable number of cards)

Items 3 and 4 are repeated for each body (if more than one body is being considered) and for off-body points, if any. The Y coordinates for each body, or off-body points, must always start on a new card, and must always follow the X coordinates. Additional cases may be run by placing additional sets (items 1 through 4) one after another.

If additional cases are to be run using some or all of the previous untransformed coordinate data, the "Subcase" capability is used. All that need be input are the body control cards, with the "Subcase" flag marked, that will transform the coordinates of the previous case in a manner desired for the present case.

If additional bodies are to be added their coordinates are input in the normal manner, without the "Subcase" flag, following the body control cards mentioned above. If, in additional cases, bodies are to be deleted or replaced these bodies must appear last in the sequence of bodies in the original case. To delete body coordinates simply omit the body control cards for that body.

When replacing a body for a subcase simply introduce the new body control cards and coordinates in the place of the replaced body.

As an illustration, if it is desired to run two Cascades, call them A and B, and then to delete B and run A alone, or with a new body C, the following procedure is followed.

Input A as the first body and B as the second as shown in figure 2. Then write new header and case control cards for the second case, placing them in back of the y cards for body B. The body control cards for A are written with the "Subcase" flag marked. To omit body B simply omit the body control cards for body B. If a new body C is to be input in place of B, its body control cards followed by its coordinate cards would be input following the body control cards for A.

A second type of subcase capability exists. If only the <u>case</u> control data is to be changed for a second case simply mark the flag in card column 8 and write out the new case control data. For example, if the calculation is desired at a second value of C_L simply write an additional header card and case control card with the new value of C_L. This may be repeated indefinitely.

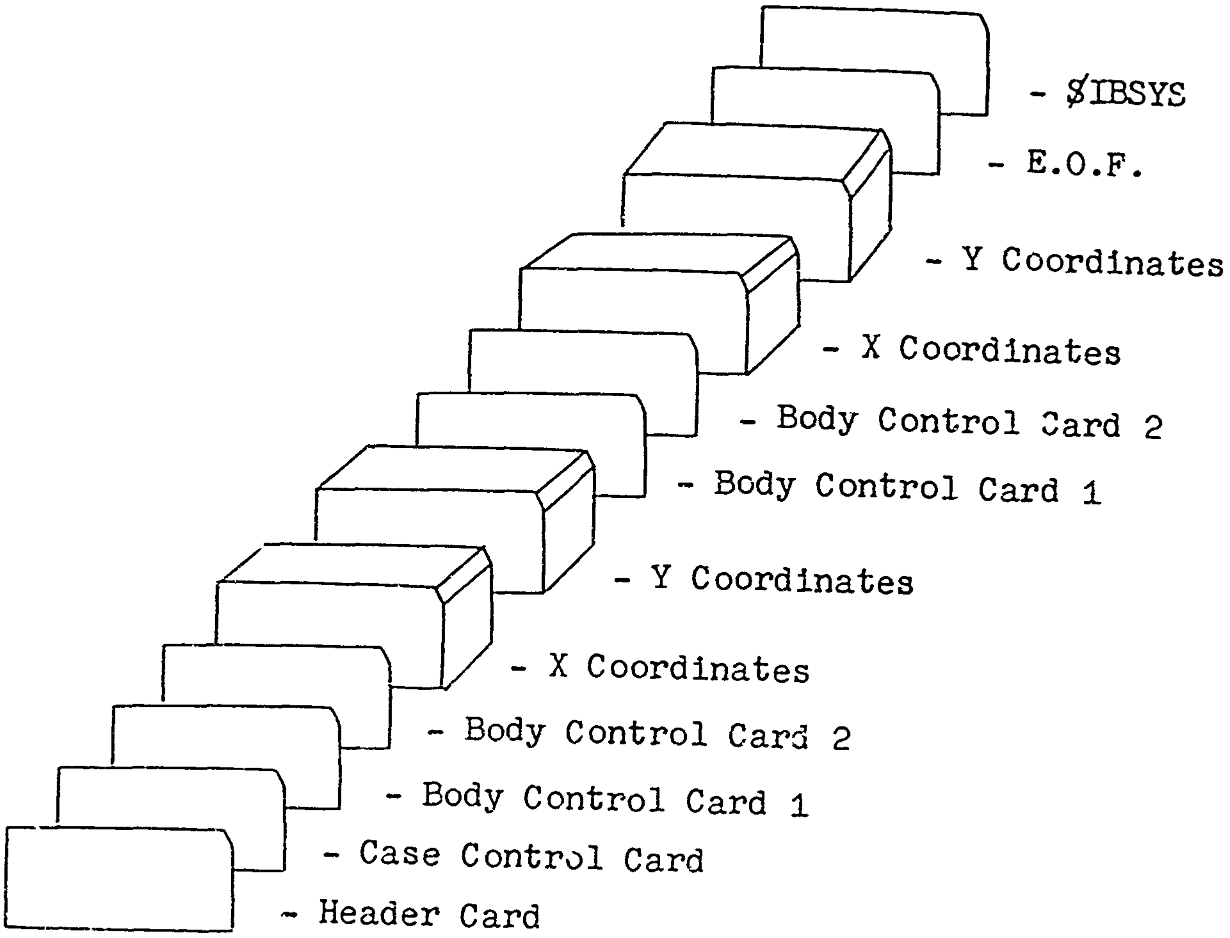


Figure 2. - Arrangement of data cards

Every complete job must be followed by an end-of-file-card (7-8 punch in card column 1) and a $IBSYS card ($IBSYS in card columns 1-6). Another case may be run by placing a header card, case control card, etc., after the last Y coordinates of the first case. Although not part of the data, an end-of-file-card must always follow the program deck, so that the data for any job are always preceded by and followed by end-of-file cards.

Header Card:

Card column 1 must always be filled with any nonzero integer that indicates the number of bodies being input. This integer must not be larger than 8.

Card columns 2-12, when punched with any nonzero integer, activate flags that indicate the following:

Card column:

2	Flow is to be determined at points off the body.
3	α is input for use in the combination equations for airfoils.
4	$\Delta\alpha$ is input for use in the combination equations for airfoils.
5	Inlet α is input for use in the combination equations for airfoils.
6	C_L is input for use in the combination equations for airfoils.
7	The matrix of influence coefficients is to be printed out.
8	Go directly to combination solution using basic velocity solutions of the previous case.
9-12	Not used.
13-60	This description of the case will be printed on each section of output.
63-68	This case number will be printed at the beginning of the output.
77-80	A sequence number must appear in these columns.

Case Control Data:

All of the input items defined in this section, with the exception of CHORD, are assumed to be zero if no value is input.

CHORD	The chord length to be used in computations for this case. It will be assumed to be 1.0 if no value is input. Any value input must appear with a decimal point.
SPACING	The spacing between the bodies of the infinite cascade. A decimal point must be specified.

C_L	The lift coefficient to be used in the combination equations for airfoils. A decimal point must be specified.
α	The angle of attack (in degrees) to be used in the combination equations for airfoils. A decimal point must be specified.
INLET α	The inlet angle (in degrees) to be used in the combination equations for airfoils. A decimal point must be specified.
$\Delta\alpha$	The cascade turning angle (in degrees) to be used in the combination equations for cascades. A decimal point must be specified.

Body Control Data:

Body Control Card 1

NN	The number of points on the body being input. The sum of all the NN for all bodies in a case must not be greater than 500. This number must not specify a decimal point, and it must be punched at the far right of its field (right-justified).
MX	The factor used to multiply all x coordinates. It is assumed to be 1.0 if no value is input. A decimal point must be specified.
MY	The factor used to multiply all y coordinates. Otherwise, same as MX.
THETA - Stagger Angle	The angle (in degrees) through which all points are to be rotated about the origin in the clockwise direction (stagger angle). A decimal point must be specified.

ADDX	The constant to be added to all x coordinates. A decimal point must be specified.
ADDY	The constant to be added to all y coordinates. A decimal point must be specified.

Body Control Card 2

BDN	The body sequence number. This number must be a nonzero integer if body coordinates follow; it is zero only if off-body coordinates follow.
NLF	This is a flag that must be any nonzero integer only if the body whose coordinates follow is to be considered a nonlifting or noncirculatory body.
SUBCASE	This is a flag that directs the program to use the unmodified coordinates of the body of the previous case. It must be any nonzero integer.
XMC YMC	The coordinates of the moment center to be used when the combination equations for airfoils are used. A decimal point must be specified.

Coordinate Data:

In inputting the body coordinates, it is essential that the coordinate data start at the body trailing edge, progress around the body in the clockwise direction and that the last point input be the first point repeated for a closed body only. The example problem illustrates this procedure in figure 13b and 13c.

X	The x coordinates of the points defining the body. (The x coordinates of the off-body points if BDN is zero). A

decimal point must be specified.

Y The y coordinates of the points defining the body. (The y coordinates of the off-body points if BDN is zero). A decimal point must be specified.

The x coordinates must precede the y coordinates in the deck arrangement. There must be NN x coordinates and NN y coordinates.

Output:

All sections of output, with the exception of the matrix printout, is preceded by the following header:

DOUGLAS AIRCRAFT COMPANY

LONG BEACH DIVISION

The information contained on the header card and the case control card is printed on the first page of output. On this page, FLAG 2 corresponds to card column 2 on the header card, FLAG 3 to card column 3, and so on.

The next section of output consists of basic data, i.e., control and coordinate data, for each body. The body control data are printed out first; the column headers follow:

X Y DELTA S SUMDS D ALPHA

The X and Y columns list the modified x and y coordinates and the midpoints of the element formed by two consecutive body points. A modified coordinate is one that has been scaled, rotated and translated according to the input. The column headed by DELTA S specified the length of the element formed by two consecutive body points, SUMDS shows a running sum of the DELTA S column, and D ALPHA shows the angle between two consecutive elements.

If off-body points are input, the following column headers are printed out after the basic data for all of the bodies:

X-OFF Y-OFF

These columns merely list the modified off-body points.

If FLAG 7 is punched (card column 7 on the header card), the A_{jk} and B_{jk} matrices are printed out after the basic data. Both matrices are printed out in row order across the page.

If off-body points are being run, A_{jk} and B_{jk} off-body matrices are formed and are printed out after the on-body matrices.

The next section of output consists of the original x and y coordinates and the midpoints of the elements formed by these coordinates, the velocities at the midpoints of the elements (V), and the corresponding pressure coefficients (CP) and source densities (SIGMA). This output is repeated for each onset flow i.e., basic solution.

The combination solution follows the basic solutions. The combination solution is a suitable combination of the basic solutions in such a way that the Kutta condition is met and the other input requirements satisfied. The format is exactly the same as that of the preceding solutions for one onset flow, except that the SIGMA column is replaced by a DELTA S column, which specifies the lengths of the original, unmodified elements. Also shown are computed and input constants that apply to the combination solution: spacing, alpha, X_{mc}, Y_{mc}, inlet alpha, exit alpha, inlet velocity, exit velocity, and delta alpha.

The last section of output shows the solution at the points off the body. The column header for this section is simply:

X Y VXL VYL

where the X and Y columns list the off-body points and the VXL and VYL columns list the X and Y components of velocity at the specified off-body point.

Example Problem

To illustrate the input procedure and to show and example of computed output, an example problem is presented. The problem consists of a cascade of circles of unit radius, spaced three radii apart. The cascade is at an average angle of attack of 10°. The only coordinate modification is a rotation of 180° to place the first coordinate point at the desired trailing-edge position which, in this problem, is on the x-axis. (see figure 13a) The circle is composed of thirty coordinate points spaced equally around the perimeter. The input coordinates (see figure 13b and 13c) progress in the clockwise direction.

The example problem also shows the input and output for four off-body points. The coordinates of these points are input in the same manner as the circle coordinates (see figures 13d and 13e) except that the flag BDN is marked 0.

The output for the example problem is shown in figures 14a through f. Figure 14a shows the basic data, i.e., the input data and the transformed coordinates. The basic data are given on the first three pages of output. Figures 14b, c, and d show the output sheets that give the three basic flow solutions: the solution at 0° angle of attack, the solution at 90° angle of attack, and the solution due to a circulatory flow. Figure 14e shows the output sheet that gives the combination solution for an average angle of attack of 10°. Also shown on the output of figure 14e are the following:

(1) Inlet and exit velocity and angle of attack

(2) Cascade lift and moment coefficient and the x and y force coefficients

(3) Spacing

(4) Cascade turning angle

Figure 14f shows the output sheet for the off-body point velocities.

Figure 15 gives a complete FORTRAN IV listing of the Douglas Cascade Program. Ten tape units are needed on the computer for this program.

REFERENCES

1. Acosta, A. J., Lindhardt, H. D.: Note on the Application of Cascade Theory to Design of Axial-Flow Pumps. ASME Paper No. 62-WA-222 Nov. 1963.

2. Smith, A. M. O., and Pierce, Jesse: - Exact Solution of the Neumann Problem. Calculation of non-Circulatory Plane and Axially Symmetric Flows About or Within Arbitrary Boundaries. Douglas Aircraft Company Report No. ES 26988, April 1958.

3. Hess, J. L.: - Calculation of Potential Flow About Bodies of Revolution Having Axes Perpendicular to the Free Stream Direction. Douglas Aircraft Company Report No. ES 29812, May 1960.

4. Hess, J. L., Smith, A. M. O.: Calculation of Non-Lifting Potential Flow About Arbitrary Three-Dimensional Bodies. Douglas Report ES 40622, March 1962.

5. Lamb H.: Hydrodynamics. Cambridge University Press, 1932.

6. Herrig, L. J., Emery, J. C., Erwin, J. R.: Systematic Two-Dimensional Cascade Tests of NACA 65-Series Compressor Blades at Low Speeds. NACA TN 3916, Feb. 1957.

7. Garrick, I. E.: On the plane potential flow Past a Lattice of Arbitrary Airfoils. NACA Report 788, 1944.

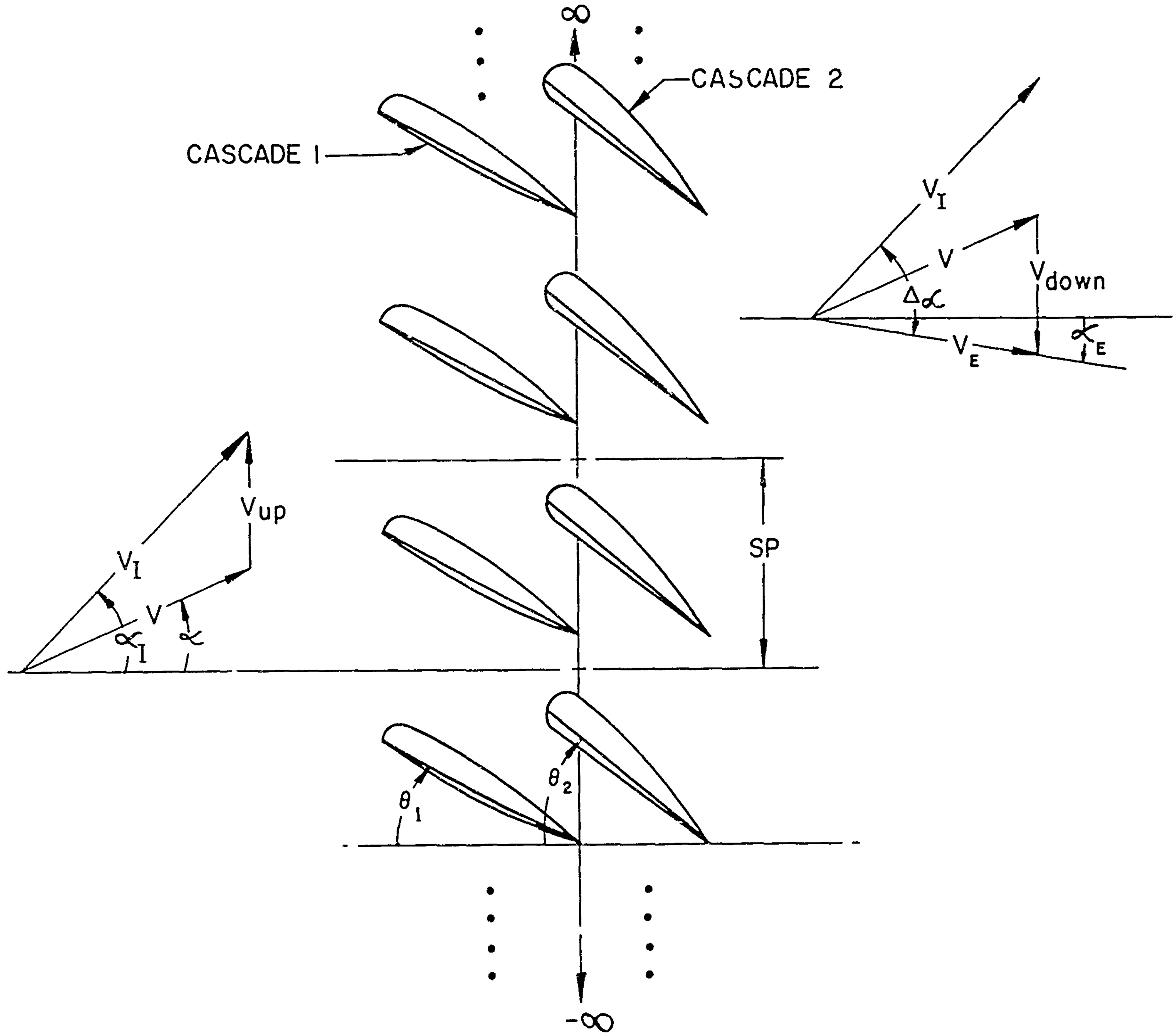


Figure 3. - Vector diagram showing inlet, average, and exit velocities and angles of attack. The cascade parameters, spacing and stagger are also shown.

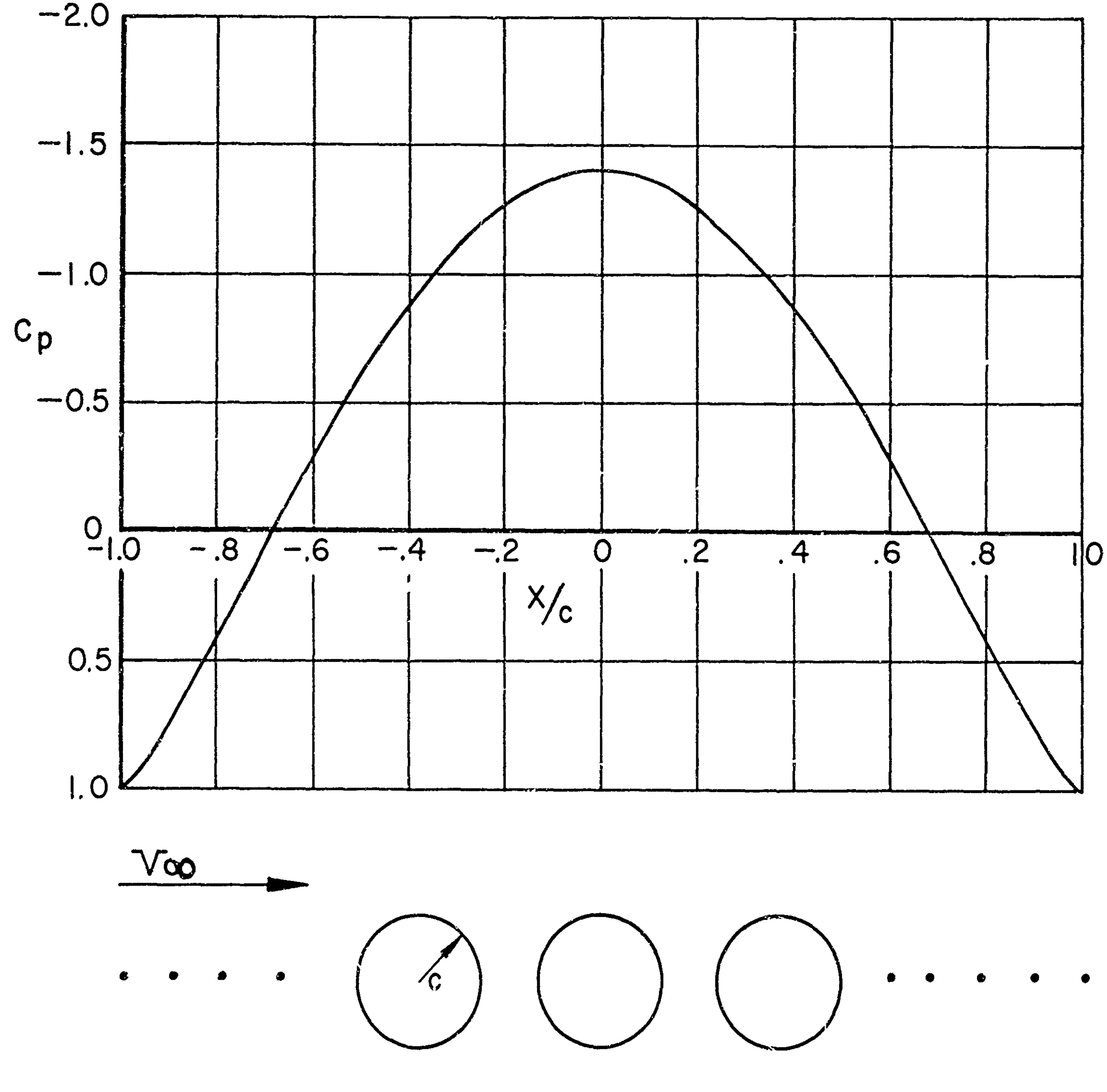


Figure 4. - Pressure distribution on a circle in tandem with an infinite number of similar circles.

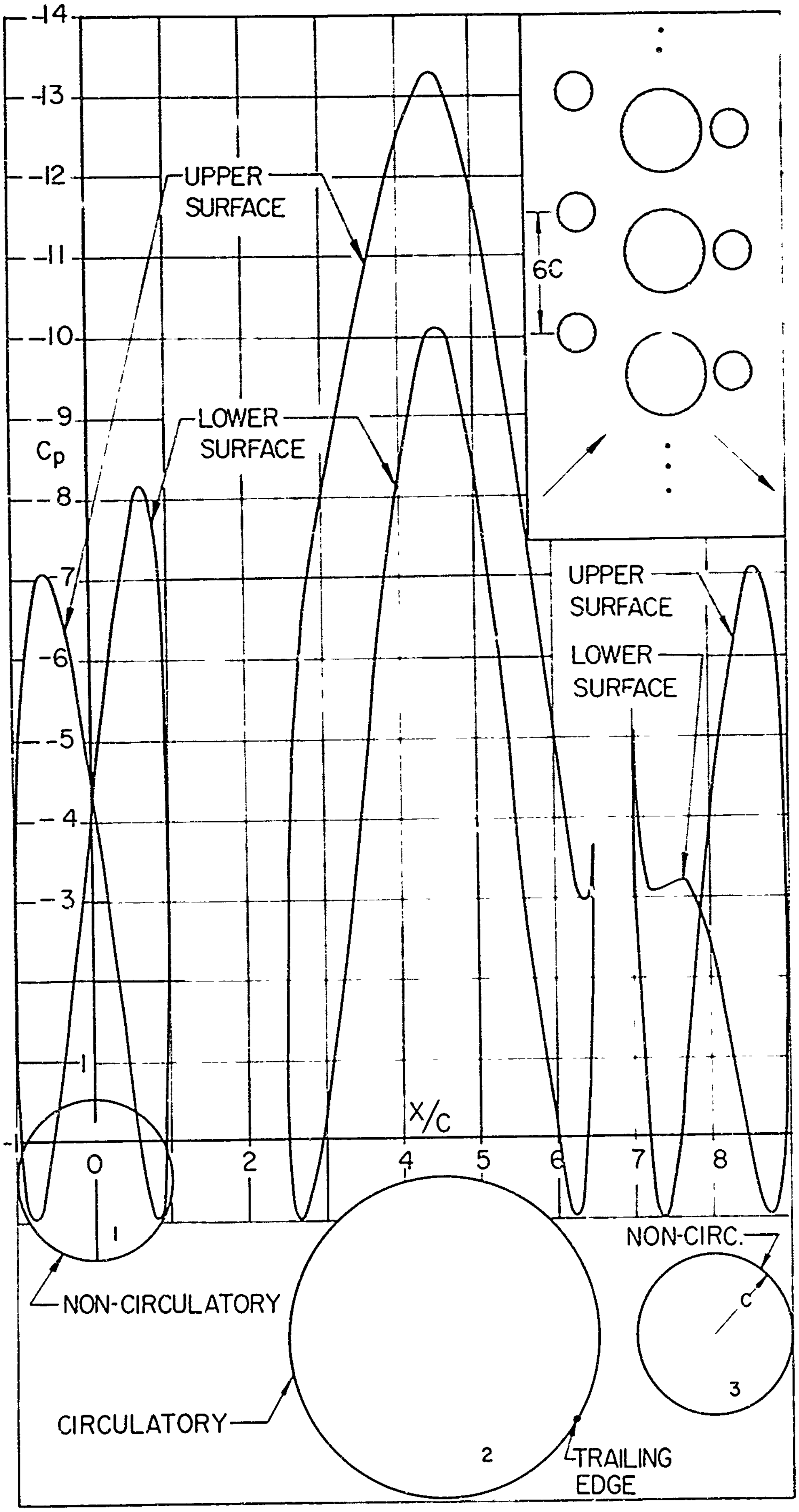


Figure 5. Pressure distributions on three circles in cascade. The central cascade has circulation. SP = 6 chord, $\alpha_1 = 43°$, $C_{L_1} = -0.08$, $C_{D_1} = 1.36$, $C_{L_2} = 19.22$, $C_{D_2} = -1.44$, $C_{L_3} = 3.48$, $C_{D_3} = 0.08$

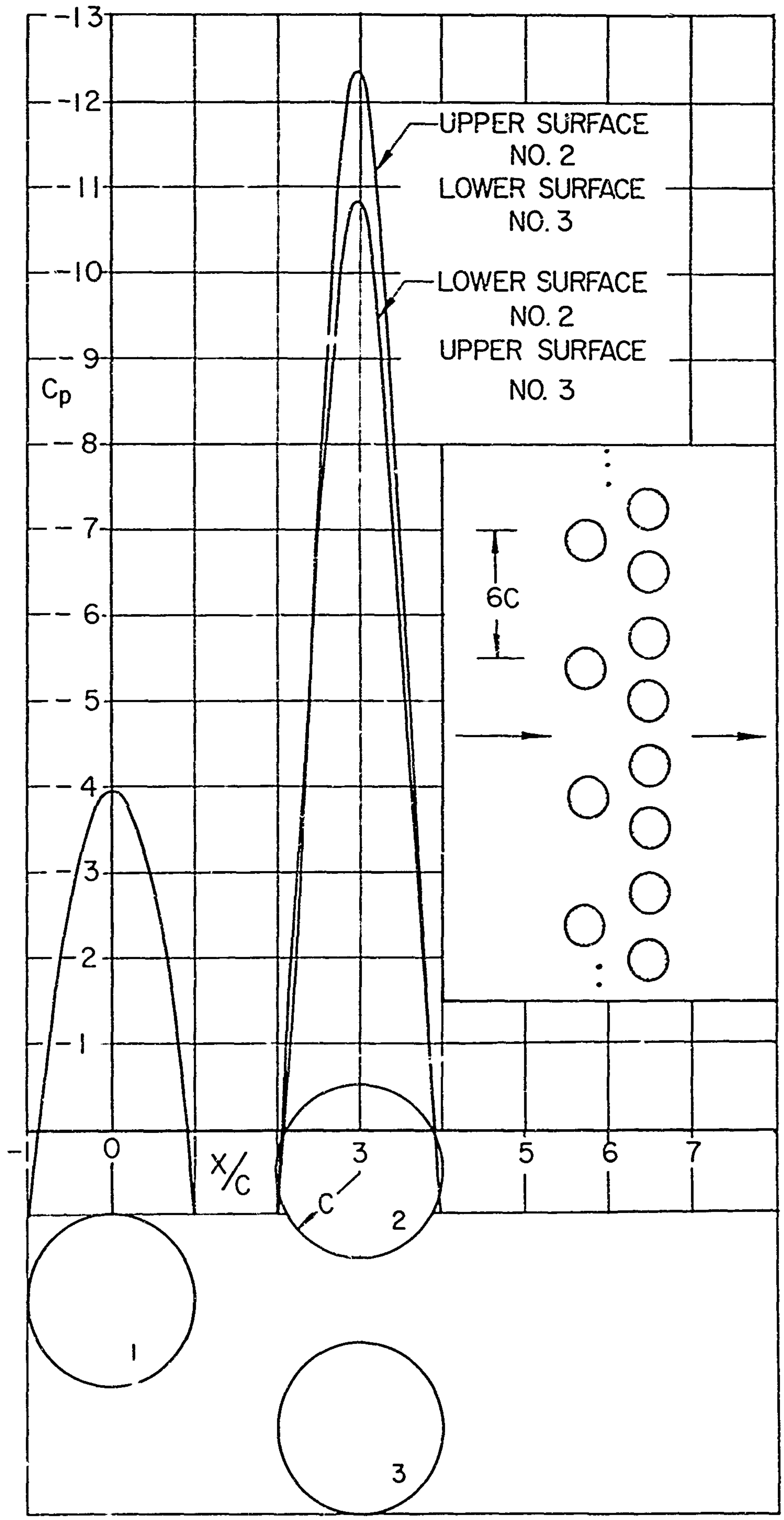


Figure 6. Pressure distributions on three circles in cascade. The arrangement shown simulates two cascades with a spacing ratio, $SP_1/SP_2 = 2$. Both cascades are non-circulatory. $SP = 6$, $\alpha_1 = 0$, $C_{L_1} = 0$, $C_{D_1} = 0.546$, $C_{L_2} = 1.31$, $C_{D_2} = -0.273$, $C_{L_3} = -1.31$ $C_{D_3} = -0.273$.

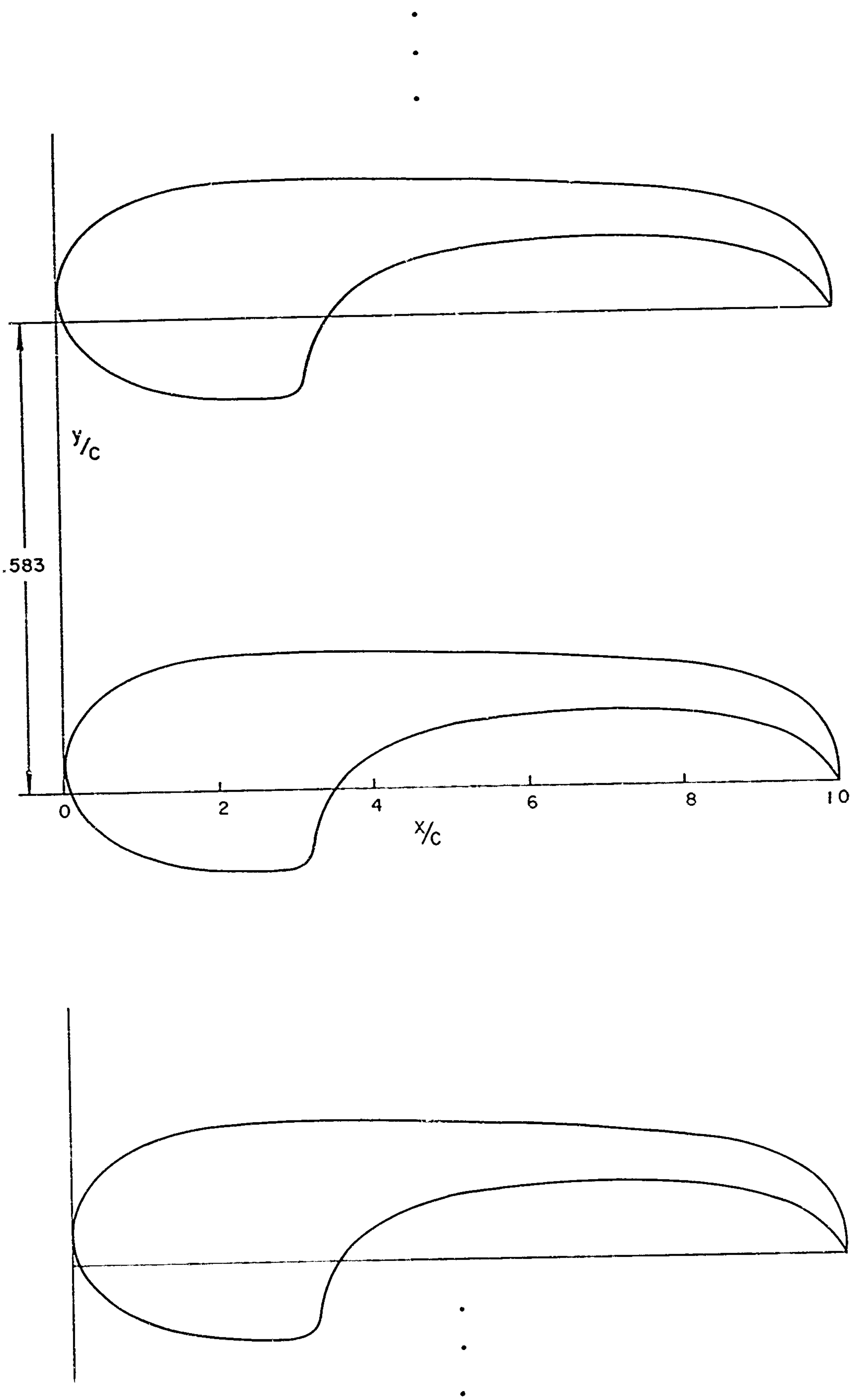


Figure 7. - Analytic cascade profile "A."

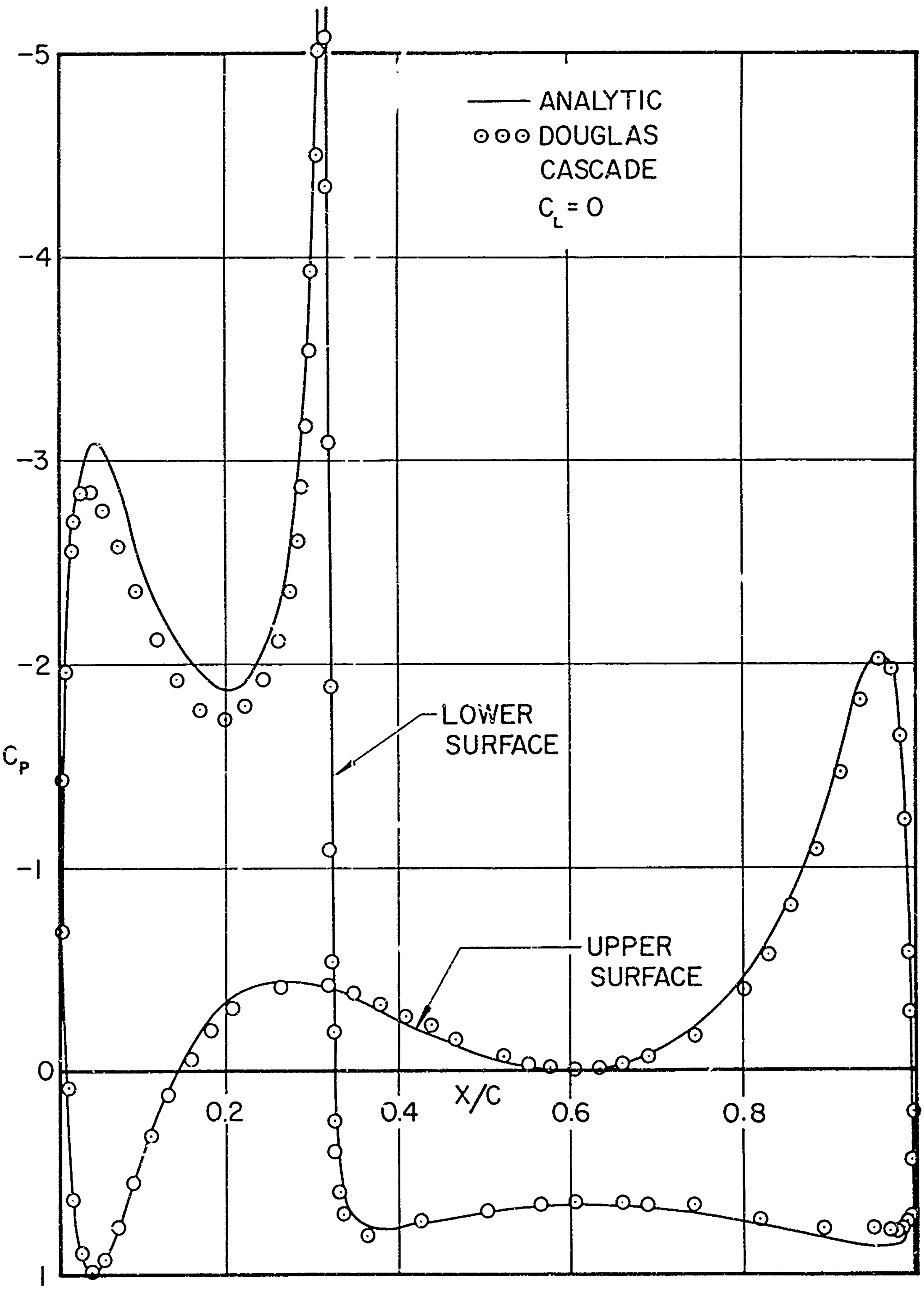


Figure 8. - Comparison of analytic and calculated pressure distributions on profile "A" in cascade. (a) $C_L = 0$, $\theta = 0°$, $\alpha_1 = -45°$, SP = 0.538

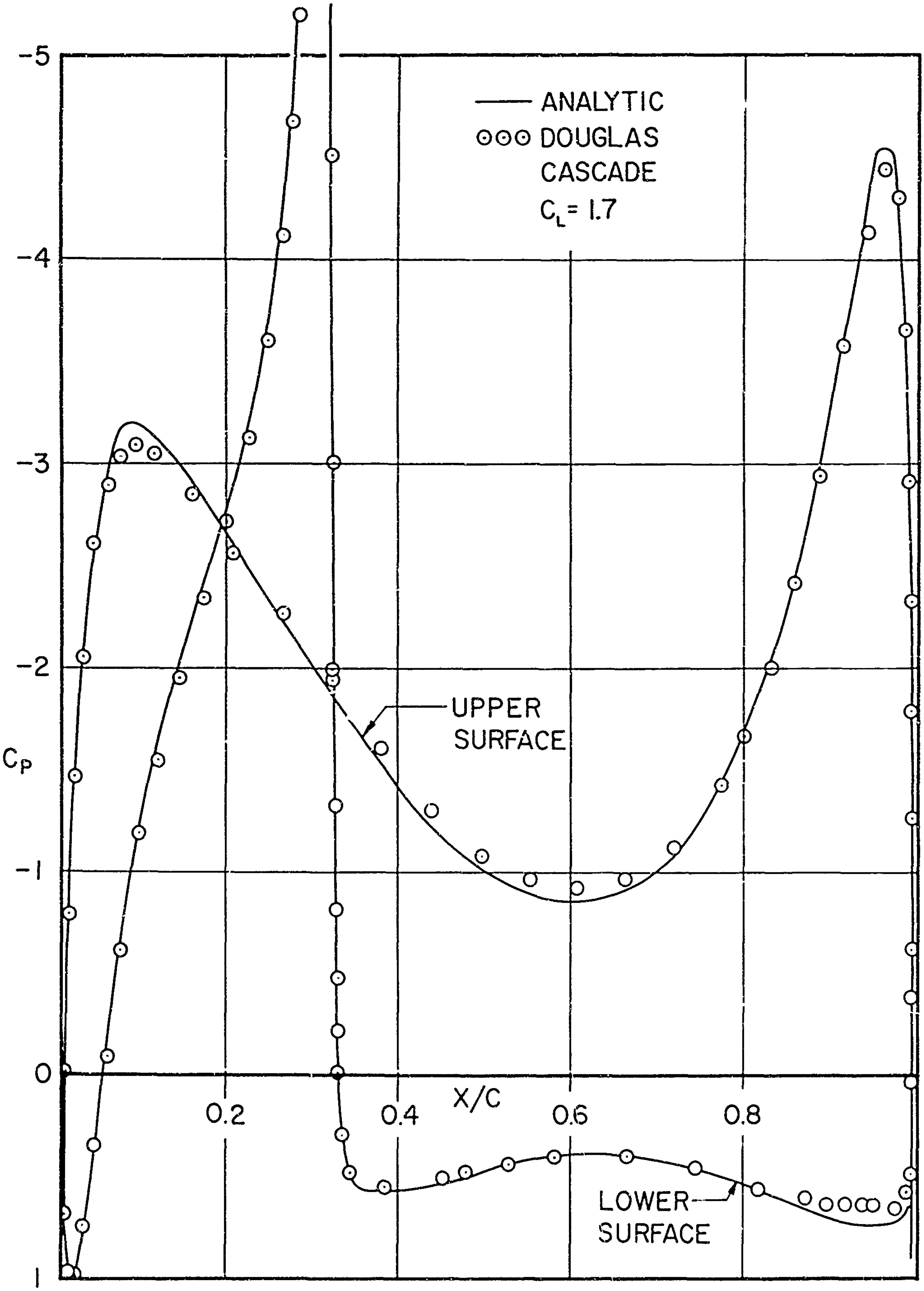


Figure 8. - Continued (b) $C_L = 1.7$, $\theta = 0°$, $\alpha_1 = 22.8°$, SP = 0.538

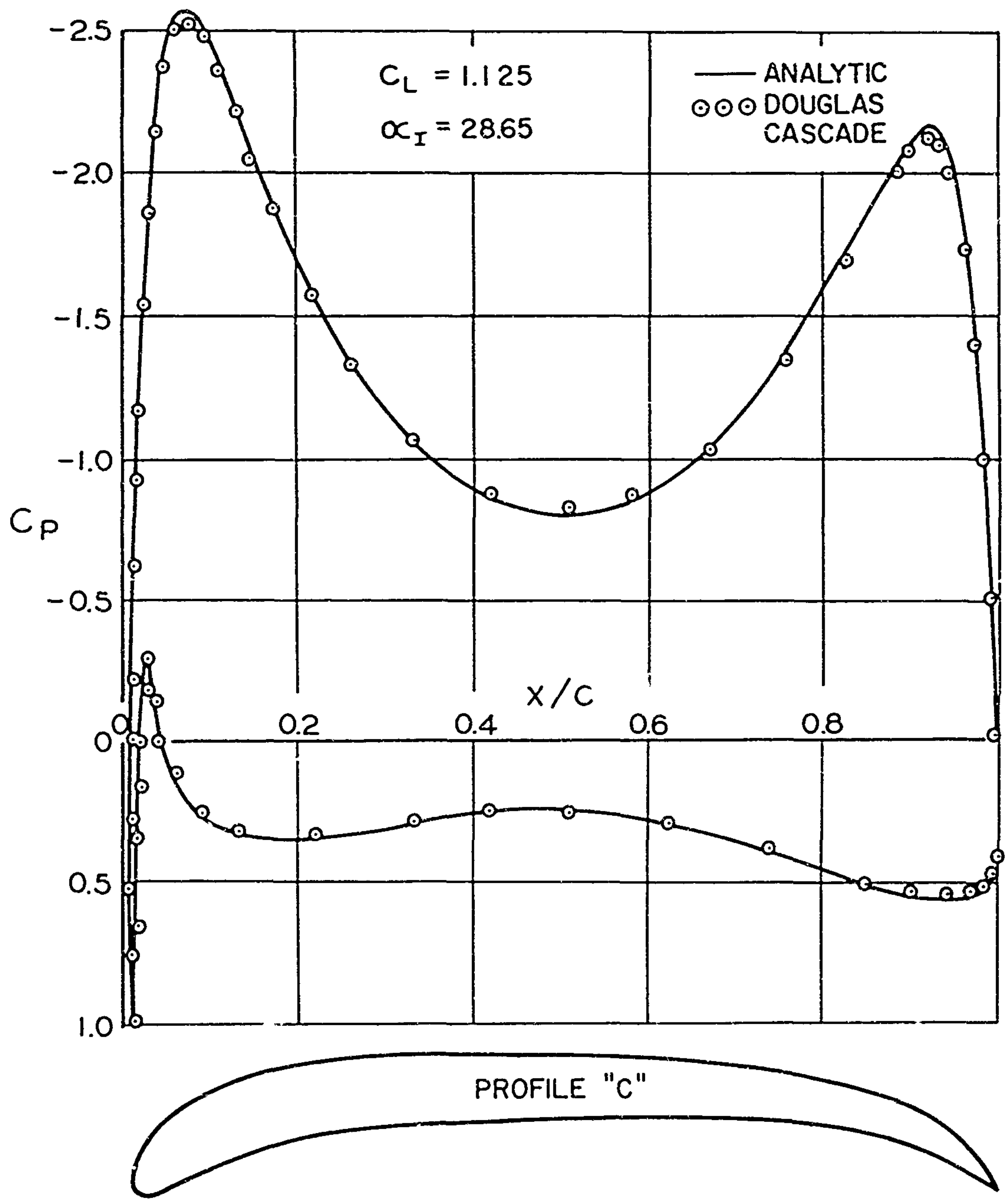


Figure 9. Comparison of analytic and calculated pressure distribution on profile "C" in cascade. (a) $C_L = 1.125$, $\theta = 0°$, $\alpha_1 = 28.65°$, SP = 0 795

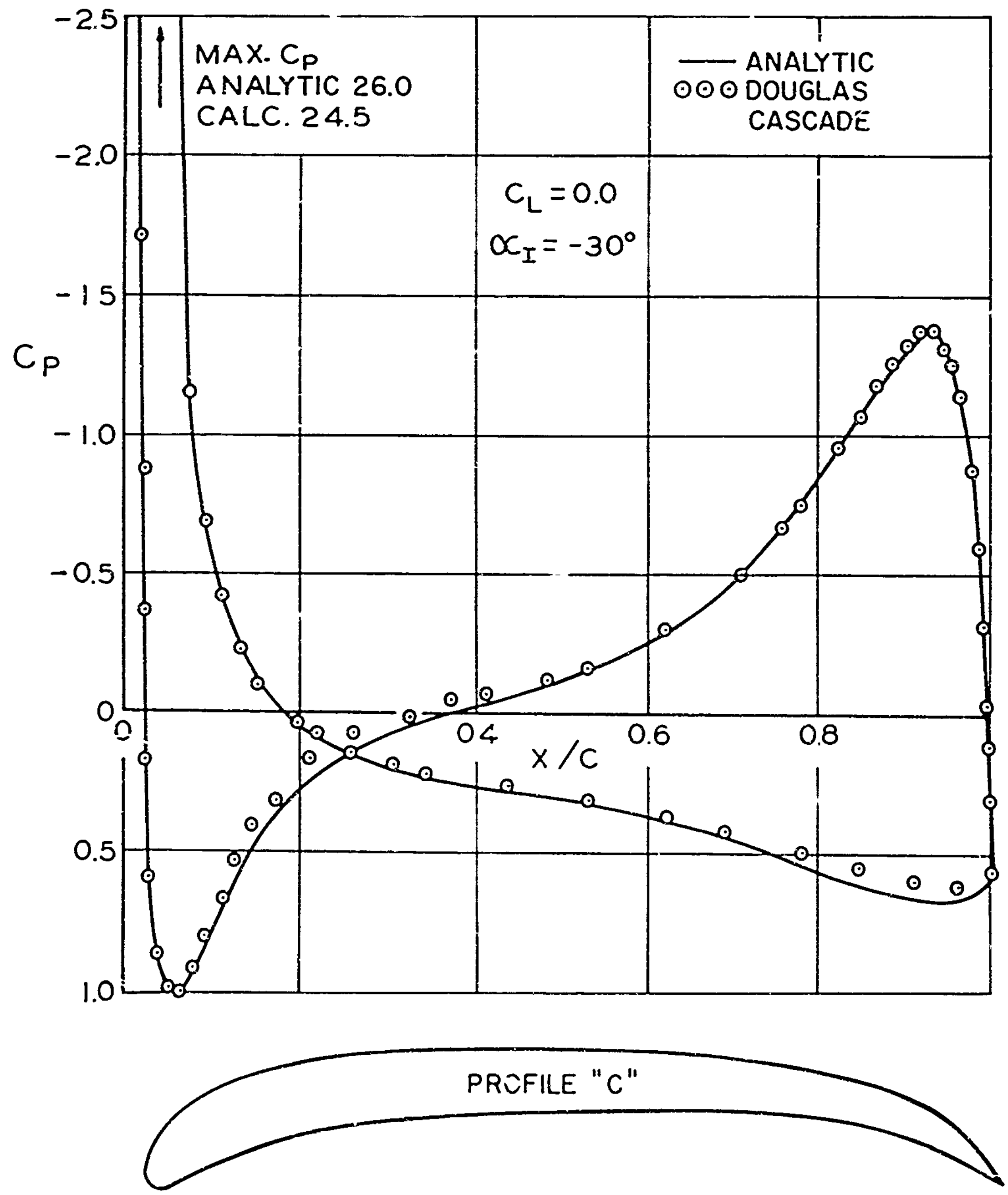


Figure 9.- Continued (b) $C_L = 0$, $\theta = 0°$, $\alpha_1 = -30°$, SP = 0.795

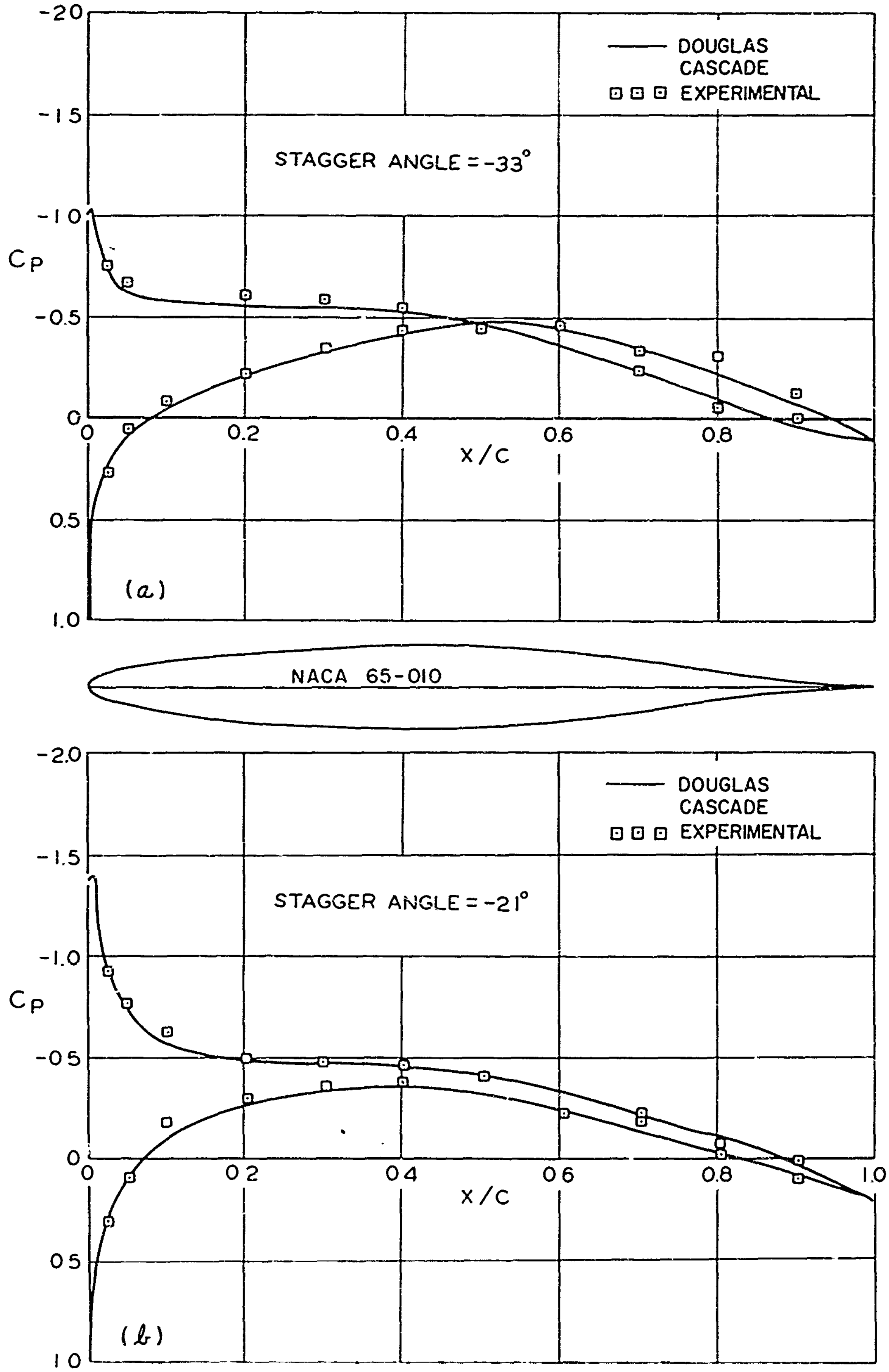


Figure 10.- Comparison of calculated and experimental pressure distributions on an NACA 65-010 airfoil in cascade. (a) $C_L = -0.135$, $\theta = -33°$, $\alpha_1 = 30°$, SP = 1.0. (b) $C_L = 0.2$, $\theta = -21°$, $\alpha_1 = 30°$, SP = 1.0

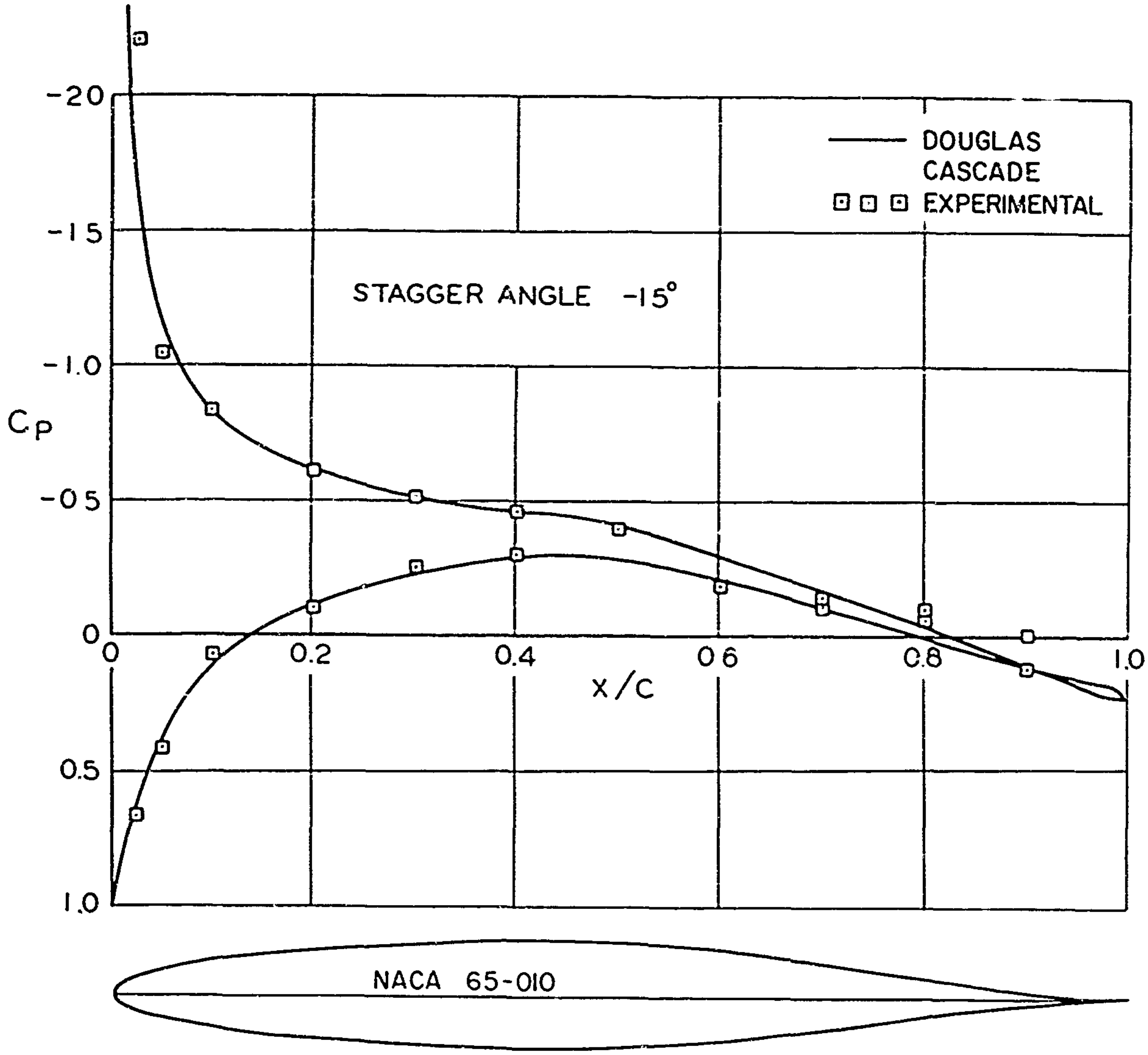


Figure 10.- Continued (c) $C_L = 0.355$, $\theta = -15°$, $\alpha_1 = 30°$, SP = 1.0

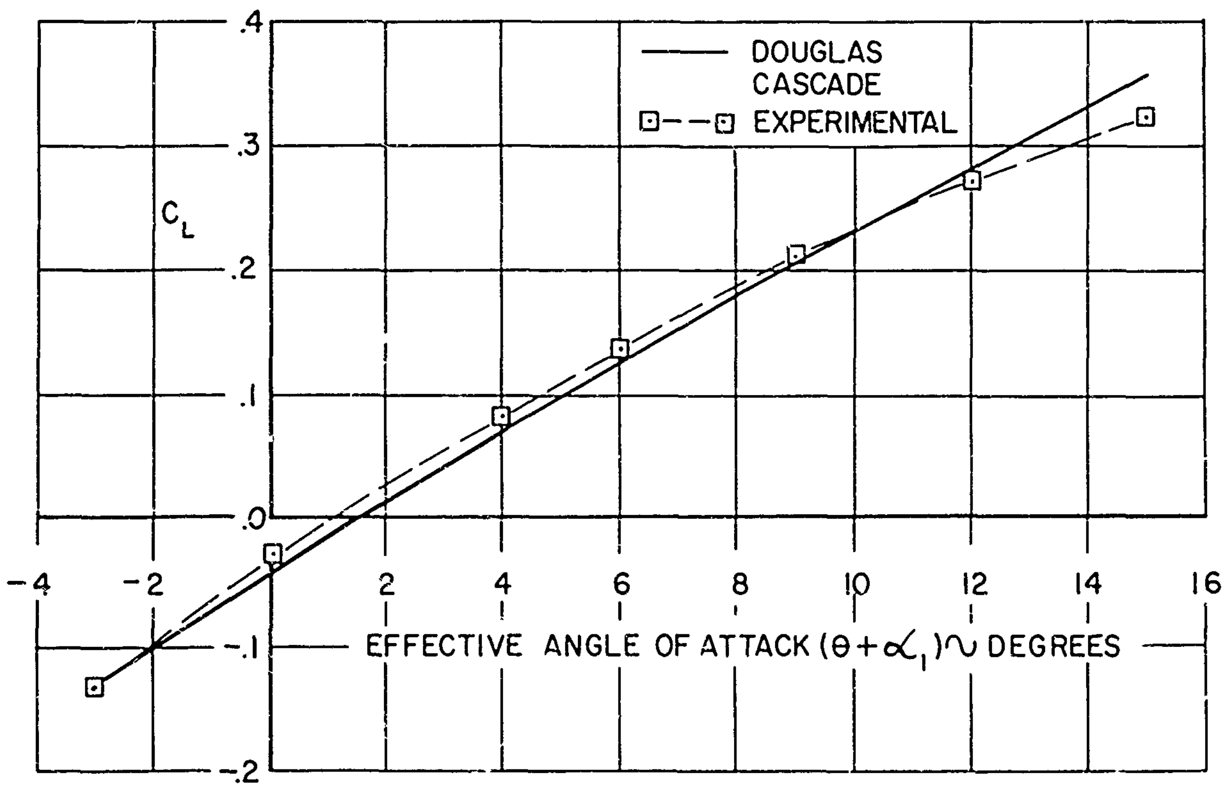


Figure 11.- Comparison of calculated and experimental lift coefficient versus "effective" angle of attack for the NACA 65-010 airfoil in cascade.

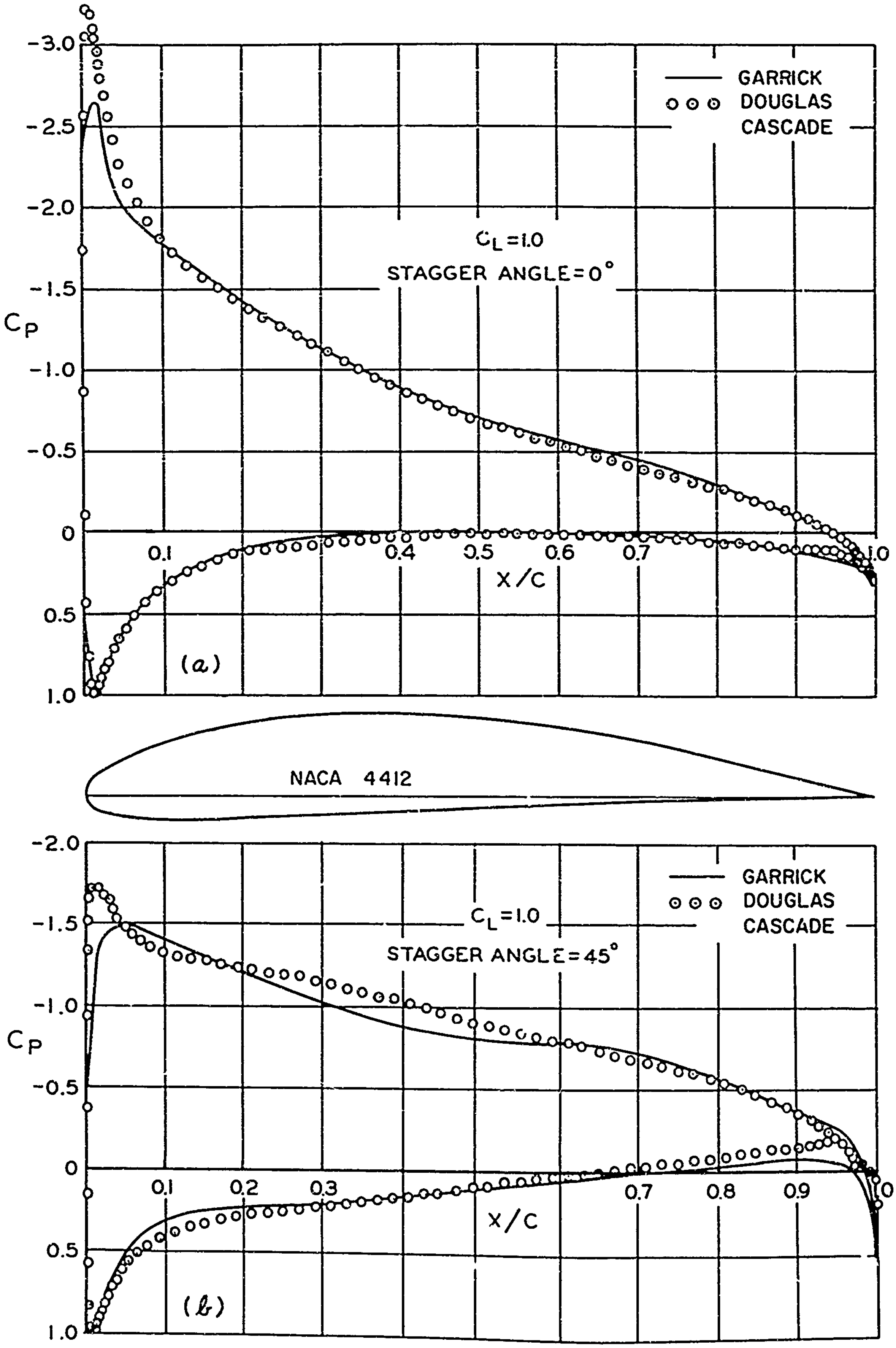


Figure 12. - Comparison of the pressure distribution, as calculated by I. E. Garrick and by Douglas, of an NACA 4412 airfoil in cascade. (a) $C_L = 1.0$, $\theta = 0°$, SP = 0.968

(b) $C_L = 1.0$, $\theta = 45°$, SP = 1.096

FORTRAN FIXED POINT INPUT

ENGINEER ________ PHONE ________ DATE ________

PUNCH IN ALL CARDS CASE ONE LOG NO. PROGRAM NO. 22Y

Flags	Description of case	SEQ.
	2-D CASCADE TEST PROBLEM	

SPACING						SEQ.
3.0		10.0				0001

(a)

PUNCH IN ALL CARDS CASE ONE LOG NO PROGRAM NO

						SEQ NO
NN	NX	NY	THETA			
31			120.0			0002
EDN	NLP	SUBCASE	XXC	XYC		
1						0003
X_1	X_2	X_3	X_4	X_5	X_6	
.99999999	.97814759	.91354544	.809017	.6691306	.49999999	0004
X_7	X_8	X_9				
.30901701	.10452847	.10452847	.30901699	.5	.66913061	0007
.80901699	.91354545	.97814760	.99999999	.9781476	.91354544	0005
.80901699	.6691306	.5	.30901699	.10452847	.10452847	0009
.309017	.5	.66913061	.809017	.91354544	.9781476	0010
.99999999						0012

DIRECTIONS FOR KEYPUNCH: DO NOT PUNCH BLANK COLUMNS —— NO UNDERPUNCHES IN SIGN FIELDS

(b)

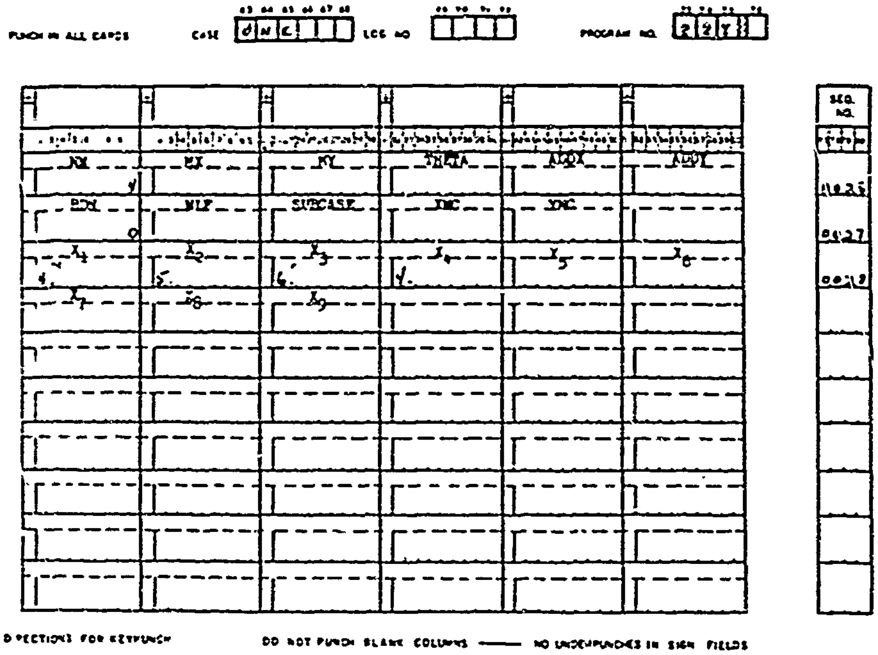
PUNCH IN ALL CARDS CASE ONE LOG NO PROGRAM NO 22Y

						SEQ NO
Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	
.00000001	.20791169	.40673664	.58778523	.74314482	.86602539	0013
Y_7	Y_8	Y_9				
.9510565	.99452189	.99452189	.95105651	.8660254	.74314483	0020
.58778525	.40673664	.20791169	0.	.20791169	.40673664	0021
.58778525	.74314483	.8660254	.95105651	.99452189	.99452189	0022
.9510565	.86602539	.74314481	.58778523	.40673664	.20791168	0024
.00000001						0025

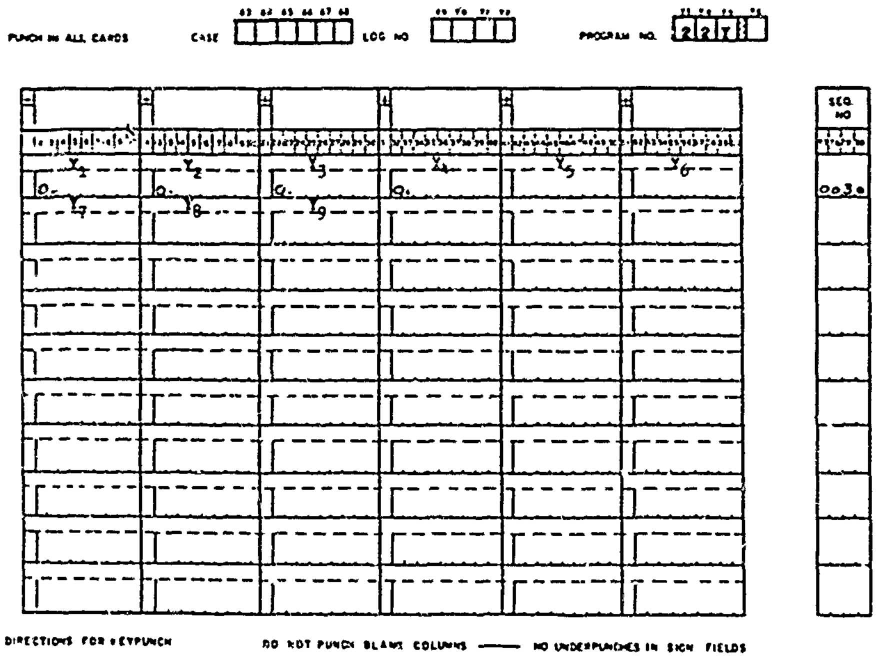
DIRECTIONS FOR KEYPUNCH: DO NOT PUNCH BLANK COLUMNS —— NO UNDERPUNCHES IN SIGN FIELDS

(c)

Figure 13. - Example problem control and data input sheets. (a) Header card and case control data. (b) Control data and x coordinates for the cascade body. (c) y coordinates for cascade body.



(d)



(e)

Figure 13. - Continued (d) Control data and x coordinates for off body points.
(e) y coordinates for off-body points.

DOUGLAS AIRCRAFT COMPANY
LONG BEACH DIVISION

PROGRAM 22Y -- 2-D CASCADE

PAGE 1

***** CASE CONTROL DATA *****

2-D CASCADE TEST PROBLEM

CASE NO. ONE

```
BODIES  =  1
FLAG 2  =  1
FLAG 3  =  1
FLAG 4  =  0
FLAG 5  =  0
FLAG 6  =  0
FLAG 7  =  0
FLAG 8  =  0
FLAG 9  =  0
FLAG 10 =  0
FLAG 11 =  0
FLAG 12 =  0

      SPACING =  3.00000000
           CL =  -0.
        ALPHA =  10.00000000
  INLET ALPHA =  -0.
  DELTA ALPHA =  -0.
        CHORD =  1.00000000
```

DOUGLAS AIRCRAFT COMPANY
LONG BEACH DIVISION

2-D CASCADE TEST PROBLEM

PAGE 3

```
NN =  4     NLF =  -0      MX =  -0.        MY =  -0.
THETA =  -0.             ADDX =  -0.      ADDY =  -0.
  XMC =  -0.              YMC =  -0.
```

OFF-BODY COORDINATES (TRANSFORMED)

	X-OFF	Y-OFF
1	4.00000000	-0.
2	5.00000000	-0.
3	6.00000000	-0.
4	4.00000000	-0.

DOUGLAS AIRCRAFT COMPANY
LONG BEACH DIVISION

2-D CASCADE TEST PROBLEM

PAGE 2

```
NN =  31     NLF =  -0      MX =  -0.        MY =  -0.
THETA =  130.00000000     ADDX =  -0.      ADDY =  -0.
  XMC =  -0.               YMC =  -0.
```

ON-BODY COORDINATES (TRANSFORMED)
BODY NO. 1

	X	Y	DELTA S	SUMDS	D ALPHA
1	0.99999997	0.00000004			
	0.98907378	-0.10395580	0.20905692	0.20905692	
2	0.97814758	-0.20791165			-12.00000286
	0.94584651	-0.30732413	0.20905693	0.41811385	
3	0.91354544	-0.40673660			-11.99999607
	0.86128122	-0.49726090	0.20905689	0.62717074	
4	0.80901700	-0.58778519			-12.00000370
	0.73907380	-0.66546498	0.20905694	0.83622768	
5	0.66913060	-0.74314477			-11.99999344
	0.58456530	-0.80458507	0.20905692	1.04528460	
6	0.50000001	-0.86602537			-12.00000370
	0.40450852	-0.90854092	0.20905690	1.25434150	
7	0.30901702	-0.95105647			-11.99999857
	0.20677276	-0.97278917	0.20905693	1.46339843	
8	0.10452849	-0.99452187			-12.00000036
	0.00000002	-0.99452187	0.20905694	1.67245536	
9	-0.10452843	-0.99452188			347.99999619
	-0.20677269	-0.97278919	0.20905691	1.88151225	
10	-0.30901695	-0.95105650			-11.99999690
	-0.40450846	-0.90854095	0.2090569	2.09056917	
11	-0.49999997	-0.86602540			-12.00000036
	-0.58456527	-0.80458511	0.20905691	2.29962608	
12	-0.66913058	-0.74314483			-12.00000715
	-0.73907376	-0.66546504	0.20905691	2.50868297	
13	-0.80901694	-0.58778526			-11.99999011
	-0.86128118	-0.49726096	0.20905692	2.71773988	
14	-0.91354542	-0.40673666			-12.00000036
	-0.94584650	-0.30732419	0.20905692	2.92679679	
15	-0.97814758	-0.20791171			-12.00000715
	-0.98907370	-0.10395586	0.20905691	3.13585371	
16	-0.99999998	-0.00000002			-11.99999523
	-0.98907378	0.10395581	0.20905691	3.34491062	
17	-0.97814759	0.20791166			-12.00000548
	-0.94584651	0.30732413	0.20905692	3.55396754	
18	-0.91354544	0.40673660			-11.99999344
	-0.86128122	0.49726091	0.20905691	3.76302442	
19	-0.80901700	0.58778521			-12.00000203
	-0.73907380	0.66546500	0.20905694	3.97208133	
20	-0.66913060	0.74314480			-11.99999857
	-0.58456530	0.80458508	0.20905691	4.18113822	
21	-0.50000001	0.86602537			-12.00000560
	-0.40450850	0.90854093	0.20905693	4.39019513	
22	-0.30901700	0.95105648			-11.99999332
	-0.20677276	0.97278918	0.20905690	4.59925199	
23	-0.10452849	0.99452187			-12.00000143
	-0.00000004	0.99452187	0.20905694	4.80830890	
24	0.10452843	0.99452188			-12.00000310
	0.20677270	0.97278918	0.20905692	5.01736581	
25	0.30901697	0.95105650			-11.99999726
	0.40450846	0.90854094	0.20905692	5.22642273	
26	0.49999997	0.86602539			-12.00000048
	0.58456527	0.80458510	0.20905692	5.43547964	
27	0.66913058	0.74314481			-11.99999940
	0.73907377	0.66546503	0.20905691	5.64453655	
28	0.80901697	0.58778524			-12.00000072
	0.86128119	0.49726095	0.20905689	5.85359341	
29	0.91354541	0.40673666			-11.99999440
	0.94584650	0.30732418	0.20905694	6.06265032	
30	0.97814758	0.20791170			-12.00003099
	0.98907373	0.10395586	0.20905691	6.27170716	
31	0.99999989	0.00000001			

Figure 14. - Program output sheets for example problem. (a) Input or basic data.

This Document Contains
Missing Page/s That Are
Unavailable In The
Original Document

OR are
Blank pgs
that have
Been Removed

**BEST
AVAILABLE COPY**

DOUGLAS AIRCRAFT COMPANY
LONG BEACH DIVISION

2-D CASCADE TEST PROBLEM

CASE ONE
STREAMFLOW SOLUTION
UNTRANSFORMED COORDINATES

	X	Y	V	CP	SIGMA
1	-0.99999999	-0.00000001			
	-0.98907378	0.10395583	-0.27628037	0.92366915	-3.13563100
2	-0.97814759	0.20791168			
	-0.94584651	0.30732416	-0.82911973	0.31256047	-3.05742222
3	-0.91354544	0.40673663			
	-0.86128122	0.49726093	-1.38354529	-0.91419758	-2.89666861
4	-0.80901700	0.58778523			
	-0.73907379	0.66546501	-1.94094200	-2.76725584	-2.63964686
5	-0.66913059	0.74314480			
	-0.58456529	0.80458509	-2.49449280	-5.22249430	-2.25525227
6	-0.49999999	0.86602539			
	-0.40450849	0.90854094	-3.00859588	-8.05164909	-1.69504222
7	-0.30901700	0.95105650			
	-0.20677274	0.97278919	-3.39735663	-10.54203200	-0.92603669
8	-0.10452846	0.99452189			
	0.	0.99452189	-3.54606095	-11.57454824	-0.00000057
9	0.10452846	0.99452189			
	0.20677272	0.97278919	-3.39735693	-10.54203403	0.92603564
10	0.30901699	0.95105650			
	0.40450849	0.90854095	-3.00859627	-8.05165148	1.69504154
11	0.50000000	0.86602540			
	0.58456530	0.80458511	-2.49449351	-5.22249788	2.25525099
12	0.66913061	0.74314483			
	0.73907379	0.66546503	-1.94094250	-2.76725775	2.63964698
13	0.80901697	0.58778524			
	0.86128121	0.49726094	-1.38354616	-0.91419996	2.89666799
14	0.91354544	0.40673663			
	0.94584652	0.30732416	-0.82912060	0.31255904	3.05742201
15	0.97814760	0.20791169			
	0.98907379	0.10395584	-0.27628082	0.92366891	3.13563150
16	0.99999999	0.			
	0.98907379	-0.10395584	0.27628004	0.92366934	3.13563138
17	0.97814760	-0.20791169			
	0.94584651	-0.30732416	0.82911979	0.31256037	3.05742210
18	0.91354544	-0.40673663			
	0.86128121	-0.49726094	1.38354519	-0.91419728	2.89666864
19	0.80901699	-0.58778524			
	0.73907379	-0.66546503	1.94094194	-2.76725560	2.63964716
20	0.66913059	-0.74314483			
	0.58456530	-0.80458511	2.49449295	-5.22249502	2.25525236
21	0.50000000	-0.86602540			
	0.40450849	-0.90854095	3.00859606	-8.05165017	1.69504198
22	0.30901697	-0.95105650			
	0.20677272	-0.97278919	3.39735678	-10.54203308	0.92603677
23	0.10452846	-0.99452189			
	-0.	-0.99452189	3.54606113	-11.57454944	0.00000054
24	-0.10452846	-0.99452189			
	-0.20677273	-0.97278919	3.39735699	-10.54203439	-0.92603590
25	-0.30901700	-0.95105650			
	-0.40450849	-0.90854094	3.00859636	-8.05165195	-1.69504163
26	-0.50000000	-0.86602539			
	-0.58456530	-0.80458509	2.49449340	-5.22249728	-2.25525141
27	-0.66913061	-0.74314480			
	-0.73907380	-0.66546501	1.94094259	-2.76725811	-2.63964647
28	-0.80901700	-0.58778523			
	-0.86128122	-0.49726093	1.38354608	-0.91419975	-2.89666826
29	-0.91354544	-0.40673663			
	-0.94584651	-0.30732416	0.82912064	0.31255897	-3.05742183
30	-0.97814760	-0.20791168			
	-0.98907375	-0.10395583	0.27627993	0.92366940	-3.13563240
31	-0.99999990	0.00000001			

Figure 14. - Continued (b) Solution at 0° angle of attack

2-D CASCADE TEST PROBLEM

CASE ONE
90-DEGREE FLOW SOLUTION
UNTRANSFORMED COORDINATES

	X	Y	V	CP	SIGMA
1	-0.99999999	-0.00000001			
	-0.98907378	0.10395583	-1.53488527	-1.35587278	0.18791410
2	-0.97814759	0.20791168			
	-0.94584651	0.30732416	-1.44404635	-1.08526984	0.54974247
3	-0.91354544	0.40673663			
	-0.86128122	0.49726093	-1.27005473	-0.61303900	0.86980452
4	-0.80901700	0.58778523			
	-0.73907379	0.66546501	-1.02985202	-0.06059517	1.12206495
5	-0.66913059	0.74314480			
	-0.58456529	0.80458509	-0.75146042	0.43530724	1.28728521
6	-0.49999999	0.86602539			
	-0.40450849	0.90854094	-0.47140203	0.77778012	1.36378153
7	-0.30901700	0.95105650			
	-0.20677274	0.97278919	-0.22030935	0.95146379	1.37810113
8	-0.10452846	0.99452189			
	0.	0.99452189	-0.00000036	0.99999999	1.37502626
9	0.10452846	0.99452189			
	0.20677272	0.97278919	0.22030865	0.95146409	1.37810123
10	0.30901699	0.95105650			
	0.40450849	0.90854095	0.47140139	0.77778073	1.36378188
11	0.50000000	0.86602540			
	0.58456530	0.80458511	0.75145990	0.43530802	1.28728572
12	0.66913061	0.74314483			
	0.73907379	0.66546503	1.02985173	-0.06059459	1.12206513
13	0.80901697	0.58778524			
	0.86128121	0.49726094	1.27005431	-0.61303794	0.86980546
14	0.91354544	0.40673663			
	0.94584652	0.30732416	1.44404624	-1.08526954	0.54974321
15	0.97814760	0.20791169			
	0.98907379	0.10395584	1.53488547	-1.35587338	0.18791476
16	0.99999999	0.			
	0.98907379	-0.10395584	1.53488551	-1.35587353	-0.18791421
17	0.97814760	-0.20791169			
	0.94584651	-0.30732416	1.44404645	-1.08527014	-0.54974271
18	0.91354544	-0.40673663			
	0.86128121	-0.49726094	1.27005479	-0.61303915	-0.86980455
19	0.80901699	-0.58778524			
	0.73907379	-0.66546503	1.02985221	-0.06059557	-1.12206489
20	0.66913059	-0.74314483			
	0.58456530	-0.80458511	0.75146053	0.43530708	-1.28728540
21	0.50000000	-0.86602540			
	0.40450849	-0.90854095	0.47140203	0.77778012	-1.36378179
22	0.30901697	-0.95105650			
	0.20677272	-0.97278919	0.22030934	0.95146379	-1.37810113
23	0.10452846	-0.99452189			
	-0.	-0.99452187	0.00000038	0.99999999	-1.37502635
24	-0.10452846	-0.99452189			
	-0.20677273	-0.97278919	-0.22030871	0.95146407	-1.37810141
25	-0.30901700	-0.95105650			
	-0.40450849	-0.90854094	-0.47140150	0.77778063	-1.36378205
26	-0.50000000	-0.86602539			
	-0.58456530	-0.80458509	-0.75146005	0.43530780	-1.28728585
27	-0.66913061	-0.74314480			
	-0.73907380	-0.66546501	-1.02985184	-0.06059480	-1.12206538
28	-0.80901700	-0.58778523			
	-0.86128122	-0.49726093	-1.27005456	-0.61303858	-0.86980533
29	-0.91354544	-0.40673663			
	-0.94584651	-0.30732416	-1.44404650	-1.08527029	-0.54974335
30	-0.97814760	-0.20791168			
	-0.98907375	-0.10395583	-1.53488554	-1.35587361	-0.18791355
31	-0.99999990	0.00000001			

Figure 14. - Continued (c) Solution at 90° angle of attack

DOUGLAS AIRCRAFT COMPANY
LONG BEACH DIVISION

2-D CASCADE TEST PROBLEM

CASE ONE
NON-UNIFORM ONSET FLOW SOLUTION NO. 1
UNTRANSFORMED COORDINATES

	X	Y	V	CP	SIGMA
1	-0.99999999	-0.00000001			
	-0.98907378	0.10395583	0.25337435	0.93580144	-0.02108791
2	-0.97814759	0.20791168			
	-0.94584651	0.30732416	0.23853764	0.94309979	-0.06055873
3	-0.91354544	0.40673663			
	-0.86128122	0.49726093	0.21024562	0.95579678	-0.09188694
4	-0.80901700	0.58778523			
	-0.73907379	0.66546501	0.17160053	0.97055326	-0.10970391
5	-0.66913059	0.74314480			
	-0.58456529	0.80458509	0.12796328	0.98362540	-0.10935894
6	-0.49999999	0.86602539			
	-0.40450849	0.90854094	0.08709732	0.99241406	-0.08856266
7	-0.30901700	0.95105650			
	-0.20677274	0.97278919	0.05785099	0.99665326	-0.04960032
8	-0.10452846	0.99452189			
	0.	0.99452189	0.04723441	0.99776891	-0.00000000
9	0.10452846	0.99452189			
	0.20677272	0.97278919	0.05785099	0.99665326	0.04960030
10	0.30901699	0.95105650			
	0.40450849	0.90854095	0.08709731	0.99241406	0.08856265
11	0.50000000	0.86602540			
	0.58456530	0.80458511	0.12796327	0.98362540	0.10935894
12	0.66913061	0.74314483			
	0.73907379	0.66546503	0.17160054	0.97055325	0.10970389
13	0.80901697	0.58778524			
	0.86128121	0.49726094	0.21024559	0.95579679	0.09188698
14	0.91354544	0.40673663			
	0.94584652	0.30732416	0.23853765	0.94309978	0.06055875
15	0.97814760	0.20791169			
	0.98907379	0.10395584	0.25337436	0.93580143	0.02108794
16	0.99999999	0.			
	0.98907379	-0.10395584	0.25337438	0.93580142	-0.02108793
17	0.97814760	-0.20791169			
	0.94584651	-0.30732416	0.23853766	0.94309978	-0.06055875
18	0.91354544	-0.40673663			
	0.86128121	-0.49726094	0.21024563	0.95579677	-0.09188694
19	0.80901699	-0.58778524			
	0.73907379	-0.66546503	0.17160054	0.97055325	-0.10970390
20	0.66913059	-0.74314483			
	0.58456530	-0.80458511	0.12796328	0.98362540	-0.10935894
21	0.50000000	-0.86602540			
	0.40450849	-0.90854095	0.08709731	0.99241406	-0.08856265
22	0.30901697	-0.95105650			
	0.20677272	-0.97278919	0.05785100	0.99665326	-0.04960031
23	0.10452846	-0.99452189			
	-0.	-0.99452189	0.04723441	0.99776891	-0.00000000
24	-0.10452846	-0.99452189			
	-0.20677273	-0.97278919	0.05785100	0.99665326	0.04960032
25	-0.30901700	-0.95105650			
	-0.40450849	-0.90854094	0.08709732	0.99241405	0.08856265
26	-0.50000000	-0.86602539			
	-0.58456530	-0.80458509	0.12796328	0.98362540	0.10935895
27	-0.66913061	-0.74314480			
	-0.73907380	-0.66546501	0.17160055	0.97055325	0.10970390
28	-0.80901700	-0.58778523			
	-0.86128122	-0.49726093	0.21024563	0.95579677	0.09188696
29	-0.91354544	-0.40673663			
	-0.94584651	-0.30732416	0.23853769	0.94309977	0.06055877
30	-0.97814760	-0.20791168			
	-0.98907375	-0.10395583	0.25337437	0.93580142	0.02108780
31	-0.99999990	0.00000001			

Figure 14.- Continued (d) Circulatory flow solution

2-D CASCADE TEST PROBLEM

SPACING = 3.00000000 ALPHA = 9.99999988 DELTA ALPHA = 19.60917735

INLET ALPHA = 19.51188397 V INLET = 1.04480879 XMC = 0.

EXIT ALPHA = -0.09729340 V EXIT = 0.98480917 YMC = 0.

COMBINED VELOCITIES
BODY NO. 1 UNTRANSFORMED COORDINATES

	X	Y	VC	CP	DELTA S
1	-0.99999999	-0.00000001			
	-0.98907378	0.10395583	-0.27208283	0.92597093	0.20905692
2	-0.97814759	0.20791168			
	-0.94584651	0.30732416	-0.81635635	0.33356231	0.20905693
3	-0.91354544	0.40673663			
	-0.86128122	0.49726093	-1.36190663	-0.85478967	0.20905690
4	-0.80901700	0.58778523			
	-0.73907379	0.66546501	-1.90977612	-2.64724481	0.20905694
5	-0.66913059	0.74314480			
	-0.58456529	0.80458509	-2.45247807	-5.01464844	0.20905693
6	-0.49999999	0.86602539			
	-0.40450849	0.90854094	-2.95312697	-7.72095883	0.20905690
7	-0.30901700	0.95105650			
	-0.20677274	0.97278919	-3.32314461	-10.04329002	0.20905693
8	-0.10452846	0.99452189			
	0.	0.99452189	-3.44250140	-10.85081577	0.20905694
9	0.10452846	0.99452189			
	0.20677272	0.97278919	-3.24663243	-9.54062212	0.20905691
10	0.30901699	0.95105650			
	0.40450849	0.90854095	-2.78941125	-6.78081506	0.20905693
11	0.50000000	0.86602540			
	0.58456530	0.80458511	-2.19149938	-3.80266953	0.20905692
12	0.66913061	0.74314483			
	0.73907379	0.66546503	-1.55211282	-1.40905419	0.20905691
13	0.80901697	0.58778524			
	0.86128121	0.49726094	-0.92082220	0.15208647	0.20905692
14	0.91354544	0.40673663			
	0.94584652	0.30732416	-0.31484519	0.90087250	0.20905692
15	0.97814760	0.20791169			
	0.98907379	0.10395584	0.26097684	0.93189109	0.20905692
16	0.99999999	0.			
	0.98907379	-0.10395584	0.80514307	0.35174464	0.20905692
17	0.97814760	-0.20791169			
	0.94584651	-0.30732416	1.31820282	-0.73765868	0.20905692
18	0.91354544	-0.40673663			
	0.86128121	-0.49726094	1.80423087	-2.25524902	0.20905691
19	0.80901699	-0.58778524			
	0.73907379	-0.66546503	2.27079713	-4.15651959	0.20905694
20	0.66913059	-0.74314483			
	0.58456530	-0.80458511	2.72169322	-6.40761393	0.20905691
21	0.50000000	-0.86602540			
	0.40450849	-0.90854095	3.13636646	-8.83679450	0.20905694
22	0.30901697	-0.95105650			
	0.20677272	-0.97278919	3.44485435	-10.86702144	0.20905690
23	0.10452846	-0.99452189			
	-0.	-0.99452189	3.54187548	-11.54488182	0.20905694
24	-0.10452846	-0.99452189			
	-0.20677273	-0.97278919	3.36834201	-10.34572780	0.20905692
25	-0.30901700	-0.95105650			
	-0.40450849	-0.90854094	2.97265065	-7.83665180	0.20905691
26	-0.50000000	-0.86602539			
	-0.58456530	-0.80458509	2.46071422	-5.05511445	0.20905693
27	-0.66913061	-0.74314480			
	-0.73907380	-0.66546501	1.91313393	-2.66008145	0.20905692
28	-0.80901700	-0.58778523			
	-0.86128122	-0.49726093	1.36314641	-0.85816813	0.20905690
29	-0.91354544	-0.40673663			
	-0.94584651	-0.30732416	0.81669162	0.33301480	0.20905694
30	-0.97814760	-0.20791168			
	-0.98907375	-0.10395583	0.27208282	0.92597093	0.20905691
31	-0.99999990	0.00000001			

CY = 2.09703341 CX = -0.35218436 CM = -0.00000016

CL = 2.10384563

Figure 14.- Continued (e) Combined solutions for cascade body

DOUGLAS AIRCRAFT COMPANY
LONG BEACH DIVISION

2-D CASCADE TEST PROBLEM

SPACING = 3.00000000 ALPHA = 9.99999988

OFF-BODY POINT VELOCITIES

	X	Y	VXL	VYL
1	4.00000000	-0.	0.98323388	-0.00167368
2	5.00000000	-0.	0.98461396	-0.00167261
3	6.00000000	-0.	0.98478390	-0.00167250
4	4.00000000	-0.	0.98323388	-0.00167368

Figure 14. - Continued (f) Combined solutions for off-body points

```
$RESTORE
$EXECUTE       IBJOB
$IBJOB
$IBFTC 22YA
      COMMON  IM, NER, NT, NB, NCFLG, RPI, R2PI, SP, CL, ALPHA, FALPHA
     1, DALFA, CHORD, FLG02, FLG03, FLG04, FLG05, FLG06, FLG07, FLG08
     2, FLG09, FLG10, FLG11, FLG12, ND, NLF, SUMDS, XHC, YHC, ADDY, HEDR
     3, CASE
      COMPLEX  IM
      INTEGER  FLG02, FLG03, FLG04, FLG05, FLG06, FLG07
     1, FLG08, FLG09, FLG10, FLG11, FLG12
   10 CALL PART1
      IF ( FLG08 .NE. 0 )  GO TO 60
      CALL PART2
      IF ( NT .LE. 135 )  GO TO 20
      CALL PART3
      GO TO 30
   20 CALL PART4
   30 CALL PART5
   60 CALL PART6
      GO TO 10
      END
$IBMAP 22YZ     5,NOLIST,NOREF,M94
        ENTRY     .UN09.
.UN09.  PZE       UNIT09
UNIT09  FILE      9,INOUT,BIN
        END
$IBMAP 22YY     5,NOLIST,NOREF,M94
        ENTRY     .UN10.                                          13ZA
.UN10.  PZE       UNIT10                                          13Z
UNIT10  FILE      10,INOUT,BIN                                    13ZA
        END                                                       13ZA
$IBMAP 27 X     5,NOLIST,NOREF,M94
        F /RY     .UN11.
.UN11.  . .C      UNIT11
UNIT11  FILE      11,INOUT,BIN
        END
$IBMAP 22YW     5,NOLIST,NOREF,M94
        ENTRY     .UN12.                                          13ZA
.UN12.  PZE       UNIT12                                          13ZA
UNIT12  FILE      12,INOUT,BIN                                    13ZA
        END                                                       13ZA
$IBMAP 22YV     5,NOLIST,NOREF,M94
        ENTRY     .UN13.
.UN13.  PZE       UNIT13
UNIT13  FILE      13,INOUT,BIN
        END
$ORIGIN         ALPHA
$IBFTC 22YB
      SUBROUTINE  PART1                                          22YB
      COMMON  IM, NER, NT, NB, NCFLG, RPI, R2PI, SP, CL, ALPHA, FALPHA
     1, DALFA, CHORD, FLG02, FLG03, FLG04, FLG05, FLG06, FLG07, FLG08
     2, FLG09, FLG10, FLG11, FLG12, ND, NLF, SUMDS, XHC, YHC, ADDY, HEDR
     3, CASE
      DIMENSION  X(500), Y(500), XMP(499), YMP(499),  ALFA(499),      22YB
     1 RSUS(499), SINA(499), COSA(499), DELS(499), DALF(498), Z(499),  22YB
     2 Q(500), HEDR(8), NO(10), NLF(10), SUMDS(10), XHC(8), YHC(8),    22YB
     3 ADDY(8), FLG02(1)
      COMPLEX  IM, Z, Q                                               22YB
      INTEGER  BDN, SUBKS, SEQ1, SEQ2                                 22YB
      INTEGER  FLG02, FLG03, FLG04, FLG05, FLG06, FLG07
     1, FLG08, FLG09, FLG10, FLG11, FLG12
      REAL  MX, MY                                                    22YB
      IM = (0., 1.)                                                   22YB
      READ (5, 4)  NB, FLG02, FLG03, FLG04, FLG05, FLG06, FLG07,
     1 FLG08, FLG09, FLG10, FLG11, FLG12, HEDR, CASE, SEQ1
    4 FORMAT (12I1, 8A6, 2X A6, 8X I4 )
      DO 50 I = 1, 11
   50 FLG02(I) = IABS( FLG02(I) )
      READ (5, 8)  SP, CL, ALPHA, FALPHA, DALFA, CHORD, SEQ2
    8 FORMAT (6F10.0, 16X I4)
      IF ( SEQ2 .GE. SEQ1 )  GO TO 80
   60 PRINT 9
    9 FORMAT (1H0/49HDATA OUT OF SEQUENCE. SORT DATA ON 77-80. RELOAD. )
      STOP
   80 SEQ1 = SEQ2
      IF ( CHORD .EQ. 0. )  CHORD = 1.                                22YB
      WRITE (6, 12)  HEDR, CASE, NB, FLG02, FLG03, FLG04, FLG05, FLG06,
     1 FLG07, FLG08, FLG09, FLG10, FLG11, FLG12, SP, CL, ALPHA, FALPHA,
     2 DALFA, CHORD
   12 FORMAT (1H1 25X 26HDOUGLAS  AIRCRAFT  COMPANY / 28X 21HLONG  BEACH
     1  DIVISION /// 6X 26HPROGRAM 22Y -- 2-D CASCADE // 11X 29H***** CA
     2SE CONTROL DATA ***** /// 6X 8A6, 4X 10HCASE NO.  A6 //6X 9HBODIES22YB
     3  =I3/ 6X 9HFLAG 2  =I3/ 6X 9HFLAG 3  =I3/ 6X 9HFLAG 4  = I3/ 6X
     4 9HFLAG 5  = I3/ 6X 9HFLAG 6  = I3/ 6X 9HFLAG 7  = I3/ 6X 9HFLAG 8
     5  = I3/ 6X 9HFLAG 9  = I3/ 6X 9HFLAG 10 = I3/ 6X 9HFLAG 11 = I3/
     6 6X 9HFLAG 12 =I3// 11X 10H SPACING = F13.8/ 16X 5H CL = F13.8/
     7 13X 8H ALPHA = F13.8/ 7X 14H INLET ALPHA = F13.8/ 7X 14H DELTA AL
     8PHA = F13.8/ 13X 8H CHORD = F13.8 )
      IF ( FLG08 .NE. 0 )  RETURN
      NT = 0                                                          22YB
      NCFLG = 2                                                       22YB
      REWIND 4                                                        22YB
      REWIND 9                                                        22YB
      REWIND 12                                                       22YB
      REWIND 13
      DO 120 I = 1, 10                                                22YB
      SUMDS(I) = 0.                                                   22YB
      NLF(I) = 0.                                                     22YB
  120 ND(I) = 0.                                                      22YB
      K2 = NB                                                         22YB
      IF ( FLG02 .NE. 0 )  K2 = NB + 1
      DO 2000 L = 1, K2                                               22YB
      READ (5, 15)  NN, MX, MY, THETA, ADDX, ADDY(L), SEQ2
   15 FORMAT (5X I5, 5F10.0, 16X I4 )
      IF ( SEQ2 .LT. SEQ1 )  GO TO 60
      SEQ1 = SEQ2
      READ (5, 16) BDN, NLFF, SUBKS, XHC(L), YHC(L), ...
   16 FORMAT ( 3(5X I5), 2F10.0, 26X I4 )
      IF ( SEQ2 .LT. SEQ1 )  GO TO 60
      SEQ1 = SEQ2
      ND(L) = NN                                                      22YB
      NLF(L) = NLFF                                                   22YB
      NT = NT + NN                                                    22YB
```

Figure 15.- FORTRAN listing of the Douglas cascade program.

```
      IF ( NLFF .EQ. 0 ) NCFLG = NCFLG + 1                              22Y8
      IF ( BDY .EQ. 0 )  NCFLG = NCFLG - 1                              22Y8
      IF ( SUBKS .EQ. 0 ) GO TO 140                                     22Y8
      READ (13) ( X(I), I = 1, NN )
      READ (13) ( Y(I), I = 1, NN )
      IF ( BDY .EQ. 0 ) GO TO 150                                       22Y8
      READ (13)
      READ (13)
      GO TO 150                                                         22Y8
  140 DO 142 I = 1, NN, 6
      READ (5, 20)  X(I), X(I+1), X(I+2), X(I+3), X(I+4), X(I+5), SEQ2
   20 FORMAT (6F10.0, 16X I4)
      IF ( SEQ2 .LT. SEQ1 )  GO TO 60
  142 SEQ1 = SEQ2
      DO 144 I = 1, NN, 6
      READ (5, 20)  Y(I), Y(I+1), Y(I+2), Y(I+3), Y(I+4), Y(I+5), SEQ2
      IF ( SEQ2 .LT. SEQ1 )  GO TO 60
  144 SEQ1 = SEQ2
      WRITE (13) ( X(I), I = 1, NN )
      WRITE (13) ( Y(I), I = 1, NN )
  150 M = NN - 1                                                        22Y8
      IF ( BDY .EQ. 0 )  GO TO 200                                      22Y8
      IF ( SUBKS .NE. 0 )  GO TO 200                                    22Y8
      DO 160 I = 1, M                                                   22Y8
      XMP(I) = ( X(I+1) + X(I) ) / 2.                                   22Y8
  160 YMP(I) = ( Y(I+1) + Y(I) ) / 2.                                   22Y8
      WRITE (13) ( XMP(I), I = 1, M )
      WRITE (13) ( YMP(I), I = 1, M )
  200 WRITE (6, 24)  HEDR, NN, NLF(L), MX, MY, THETA, ADDX, ADDY(L),    22Y8
     1 XMC(L), YMC(L)                                                   22Y8
   24 FORMAT (1H1 25X 26HDOUGLAS  AIRCRAFT  COMPANY / 28X 21HLONG  BEACH22Y8
     1  DIVISION /// 5X 8A6 // 5X 4HNN = I4, 4X 5HNLF = I4, 5X 4HMX =   22Y8
     2 F13.8, 4X 4HMY = F13.9 / 5X 7HTHETA = F13.8, 4X 6HADDX = F13.8,  22Y8
     3 2X 6HADDY = F13.8 / 7X 5HXMC = F13.8, 5X 5HYMC = F13.8 )
      IF ( BDY .EQ. 0 )  GO TO 220                                      22Y8
      WRITE (6, 28) BDY                                                 22Y8
   28 FORMAT (1H0 4X 33HON-BODY COORDINATES (TRANSFORMED) / 9H BODY NO.
     1 I3 // 11X 2H X 13X 1HY 11X 7HDELTA S 7X 5HSUMDS 8X 7HD ALPHA // )
      GO TO 240                                                         22Y8
  220 WRITE (6, 32)                                                     22Y8
   32 FORMAT (1H0 4X 34HOFF-BODY COORDINATES (TRANSFORMED) //10X 5HX-OFF
     1 9X 5HY-OFF // )
  240 IF ( MX .EQ. 0. )  GO TO 260                                      22Y8
      DO 250 I = 1, NN                                                  22Y8
  250 X(I) = X(I) + MX                                                  22Y8
      XMC(L) = XMC(L) + MX                                              22Y8
  260 IF ( MY .EQ. 0. )  GO TO 290                                      22Y8
      DO 270 I = 1, NN                                                  22Y8
  270 Y(I) = Y(I) + MY                                                  22Y8
      YMC(L) = YMC(L) + MY                                              22Y8
  280 IF ( THETA .EQ. 0. )  GO TO 300                                   22Y8
      THETA = THETA / 57.2957795                                        22Y8
      CSTHT = COS( THETA )                                              22Y8
      SNTHT = SIN( THETA )                                              22Y8
      DO 290 I = 1, NN                                                  22Y8
      T1 = X(I)                                                         22Y8
      X(I) = T1*CSTHT + Y(I)*SNTHT                                      22Y8
  290 Y(I) = Y(I)*CSTHT - T1*SNTHT                                      22Y8
      T1 = XMC(L)                                                       22Y8
      XMC(L) = T1*CSTHT + YMC(L)*SNTHT                                  22Y8
      YMC(L) = YMC(L)*CSTHT - T1*SNTHT                                  22Y8
  300 IF ( ADDX .EQ. 0. ) GO TO 320                                     22Y8
      DO 310 I = 1, NN                                                  22Y8
  310 X(I) = X(I) + ADDX                                                22Y8
      XMC(L) = XMC(L) + ADDX                                            22Y8
  320 T1 = ADDY(L)                                                      22Y8
      IF ( T1 .EQ. 0. )  GO TO 340                                      22Y8
      DO 330 I = 1, NN                                                  22Y8
  330 Y(I) = Y(I) + T1                                                  22Y8
      YMC(L) = YMC(L) + T1                                              22Y8
  340 IF ( CHORD .EQ. 1. )  GO TO 360                                   22Y8
      DO 350 I = 1, NN                                                  22Y8
      X(I) = X(I)/CHORD                                                 22Y8
  350 Y(I) = Y(I)/CHORD                                                 22Y8
      XMC(L) = XMC(L)/CHORD                                             22Y8
      YMC(L) = YMC(L)/CHORD                                             22Y8
  360 IF ( BDY .EQ. 0 )  GO TO 500                                      22Y8
      DO 380 I = 1, NN                                                  22Y8
  380 RSDS(I) = 0.                                                      22Y8
      SUMS = 0.                                                         22Y8
      DO 400 I = 1, M                                                   22Y8
      T1 = X(I+1) - X(I)                                                22Y8
      T2 = Y(I+1) - Y(I)                                                22Y8
      XMP(I) = (X(I+1) + X(I)) / 2.                                     22Y8
      YMP(I) = (Y(I+1) + Y(I)) / 2.                                     22Y8
      TDS = SQRT ( T1*T1 + T2*T2 )                                      22Y8
      DELS(I) = TDS                                                     22Y8
      SUMS = SUMS + TDS                                                 22Y8
      RSDS(I) = SUMS                                                    22Y8
      SUMDS(L) = SUMDS(L) + TDS                                         22Y8
      COSA(I) = T1 / TDS                                                22Y8
      SINA(I) = T2 / TDS                                                22Y8
      ALFA(I) = ATAN2 (T2, T1)                                          22Y8
      Z(I) = CMPLX ( XMP(I), YMP(I) )                                   22Y8
  400 Q(I) = CMPLX ( X(I), Y(I) )                                       22Y8
      Q(NN) = CMPLX ( X(NN), Y(NN) )                                    22Y8
      WRITE (12) ( XMP(I), I = 1, M )                                   22Y8
      WRITE (12) ( YMP(I), I = 1, M )                                   22Y8
      WRITE (12) ( DELS(I), I = 1, M )                                  22Y8
      M = NN - 2                                                        22Y8
      DO 420 I = 1, M                                                   22Y8
  420 DALF(I) = ( ALFA(I+1)-ALFA(I) ) * 57.2957795                      22Y8
      WRITE (6, 36)  X(1), Y(1), XMP(1), YMP(1), DELS(1), RSDS(1)       22Y8
   36 FORMAT (1H  3H  1  2F14.8,  / 4X  4F14.8 )                        22Y8
      M = NN - 1                                                        22Y8
      WRITE (6, 40) ( I, X(I), Y(I), DALF(I-1), XMP(I), YMP(I), DELS(I) 22Y8
     1 , RSDS(I), I = 2, M )                                            22Y8
   40 FORMAT (1H  I3, 2F14.8, 28X F14.8 / 4X  4F14.8 )                  22Y8
      WRITE (6, 44)  NN, X(NN), Y(NN)                                   22Y8
   44 FORMAT (1H  I3, 2F14.8 )                                          22Y8
      GO TO 600                                                         22Y8
  500 WRITE (6, 48) ( I, X(I), Y(I), I = 1, NN )                        22Y8
   48 FORMAT (1H  I3, 2F14.8 )                                          22Y8
```

Figure 15. - Continued

```
      M = NN                                                         22YB
      DO 550 I = 1, NN                                               22YB
  550 Z(I) = CMPLX( X(I), Y(I) )                                     22YB
  600 WRITE (9)  (Z(I), I = 1, M )                                   22YB
      IF ( BDN .EQ. 0 )  GO TO 2000                                  22YB
      WRITE (9) ( SINA(I), I = 1, M )                                22YB
      WRITE (4) ( SINA(I), I = 1, M )                                22YB
      WRITE (9) ( COSA(I), I = 1, M )                                22YB
      WRITE (4) ( COSA(I), I = 1, M )                                22YB
      WRITE (9) ( Q(I), I = 1, NN )                                  22YB
 2000 CONTINUE                                                       22YB
      NT = NT - NB - ND(NB+1)                                        22YB
C     NT = TOTAL NO. OF ELEMENTS                                     22YB
      RETURN                                                         22YB
      END                                                            22YB
$ORIGIN       ALPHA
$IBFTC 22YC
      SUBROUTINE  PART2                                              22YC
      COMMON  IM, NER, NT, NB, NCFLG, RPI, R2PI, SP, CL, ALPHA, FALPHA
     1, DALFA, CHORD, FLG02, FLG03, FLG04, FLG05, FLG06, FLG07, FLG08
     2, FLG09, FLG10, FLG11, FLG12, ND, NLF, SUMDS, XPC, YPC, ADDY, HEDR
      INTEGER  FLG02, FLG03, FLG04, FLG05, FLG06, FLG07
     1, FLG08, FLG09, FLG10, FLG11, FLG12
      DIMENSION  Z(499), Q(500), SINA(499), COSA(500), ND(10),       22YC
     1 VNS(4000), VTS(4000), A(499), B(499), SUMDS(10), NLF(10)      22YC
      COMPLEX  IM, Z, Q, W1, W2, TF, T2, T1, CLOG, CSINH
      RPI = 0.31830989                                               22YC
      R2PI = 0.15915494                                              22YC
      REWIND  9                                                      22YC
      REWIND 10                                                      22YC
      REWIND 11                                                      22YC
      M = 1                                                          22YC
      N = ND(1) - 1                                                  22YC
      M1 = 1                                                         22YC
      N1 = ND(1)                                                     22YC
      DO 100 L = 1, NB                                               22YC
      READ (9) (Z(I), I = M, N)                                      22YC
      READ (9) (SINA(I), I = M, N)                                   22YC
      READ (9) (COSA(I), I = M, N)                                   22YC
      READ (9) (Q(I), I = M1, N1)                                    22YC
      M = N + 1                                                      22YC
      N = N + ND(L+1) - 1                                            22YC
      M1 = N1 + 1                                                    22YC
  100 N1 = N1 + ND(L+1)                                              22YC
      K = NB * NT                                                    22YC
      DO 200 I = 1, K                                                22YC
      VNS(I) = 0.                                                    22YC
  200 VTS(I) = 0.                                                    22YC
      NPFLG = 0                                                      22YC
      A2 = 0.                                                        22YC
      B2 = 0.                                                        22YC
      L = NT                                                         22YC
  500 DO 1500 J = 1, L                                               22YC
      M1 = 1                                                         22YC
      N1 = ND(1) - 1                                                 22YC
      J1 = J - L                                                     22YC
      J2 = 0                                                         22YC
      J4 = 0                                                         22YC
      T1 = COSA(J) - IM*SINA(J)                                      22YC
      DO 1200 I = 1, NB                                              22YC
      J1 = J1 + L                                                    22YC
      J4 = J4 + 1                                                    22YC
      TF = CSINH( 3.14159265*(Z(J)-Q(M1)) / SP )
      DO 1000 K = M1, N1                                             22YC
      J2 = J2 + 1                                                    22YC
      IF ( SP .GT. 0. )  GO TO 650                                   22YC
      CALL FORM1 ( J, K, J2, Z, Q, SINA, COSA, W1 )                  22YC
      IF ( SP .EQ. 0. )  GO TO 700                                   22YC
      CALL FORM2 ( J, K, J2, Z, Q, SINA, COSA, W2 )                  22YC
      IF ( NPFLG .NE. 0 )   GO TO 550                                22YC
      T2 = CONJG(W2) * T1                                            22YC
      A2 = AIMAG( T2 )                                               22YC
      B2 = REAL ( T2 )                                               22YC
      GO TO 720                                                      22YC
  550 A2 = - AIMAG( W2 )                                             22YC
      B2 = REAL( W2 )                                                22YC
      GO TO 750                                                      22YC
  650 CALL SPGTO ( J, K, J2, SP, TF, Z, Q, SINA, COSA, W1 )
  700 IF ( NPFLG .NE. 0 )  GO TO 750                                 22YC
  720 T2 = CONJG(W1) * T1                                            22YC
      A1 = AIMAG( T2 )                                               22YC
      IF (J .EQ. J2)  A1 = ABS( A1 )                                 22YC
      B1 = REAL( T2 )                                                22YC
      GO TO 800                                                      22YC
  750 A1 = - AIMAG( W1 )                                             22YC
      B1 = REAL( W1 )                                                22YC
  800 VNS(J1) = VNS(J1) - B1 + B2                                    22YC
      VTS(J1) = VTS(J1) + A1 - A2                                    22YC
      A(J2) = A1 + A2                                                22YC
 1000 B(J2) = B1 + B2                                                22YC
      VNS(J1) = VNS(J1) / SUMDS(J4)                                  22YC
      VTS(J1) = VTS(J1) / SUMDS(J4)                                  22YC
      M1 = N1 + 2                                                    22YC
 1200 N1 = N1 + ND(I+1)                                              22YC
      WRITE (10) (A(I), I = 1, NT)                                   22YC
      WRITE (10) (B(I), I = 1, NT)                                   22YC
      IF ( FLG07 .EQ. 0 )  GO TO 1300
      WRITE (6, 5) (J, (A(I), I = 1, NT) )                           22YC
    5 FORMAT (1H0 12H   AJK    ROW I4 // (6F15.8) )                  22YC
      WRITE (6, 10) (J, (B(I), I = 1, NT) )                          22YC
   10 FORMAT (1H0 12H   BJK   ROW  I4 // (6F15.8) )                  22YC
 1300 IF ( NT .LE. 135 .OR. NPFLG .NE. 0 )  GO TO 1500               22YC
      WRITE (11) ( A(I), I = 1, NT )                                 22YC
      WRITE (11) ( B(I), I = 1, NT )                                 22YC
 1500 CONTINUE                                                       22YC
      M = 1                                                          22YC
      N = L                                                          22YC
      DO 2000 J = 1, NB                                              22YC
      IF ( NLF(J) .NE. 0 ) GO TO 1800                                22YC
      WRITE (4)  (VNS(I), I = M, N)                                  22YC
      WRITE (4)  (VTS(I), I = M, N)                                  22YC
 1800 M = N + 1                                                      22YC
 2000 N = N + L                                                      22YC
      IF ( FLG07 .EQ. 0 )  GO TO 3000
```

Figure 15. - Continued

```
      N = NB + L                                                        22YC
      WRITE (6, 15) ( VNS(I), I = 1, N )                                22YC
      WRITE (6, 20) ( VTS(I), I = 1, N )                                22YC
   15 FORMAT (1HO/10X 3HVNS /// (6F15.8) )                              22YC
   20 FORMAT (1HO / 10X 3HVTS /// (6F15.8) )                            22YC
 3000 IF ( FLG02 .EQ. 0 .OR. NPFLG .NE. 0 )  RETURN
      NPFLG = 1                                                         22YC
      L = ND(NB+1)                                                      22YC
      READ (9) ( Z(I), I = 1, L )                                       22YC
      K = NB + L
      DO 3100 I = 1, K
      VNS(I) = 0.
 3100 VTS(I) = 0.
      GO TO 500                                                         22YC
      END                                                               22YC
$IBFTC 22YD
      SUBROUTINE  FORM1  ( J, K, J2, Z, Q, SINA, COSA, W )              22YC
      COMMON  IM, NER, NT, NB, NCFLG, RPI, R2PI, SP, CL, ALPHA, FALPHA
      COMPLEX  IM, Z, Q, W, CLOG                                        22YC
      DIMENSION   Z(499), Q(500), SINA(499), COSA(499)                  22YC
      W = CLOG ( (Z(J)-Q(K)) / (Z(J)-Q(K+1)) )                          22YC
      W = ( COSA(J2) - IM*SINA(J2) ) * R2PI * W                         22YC
      RETURN                                                            22YC
      END                                                               22YC
$IBFTC 22YE
      SUBROUTINE  FORM2  ( J, K, J2, Z, Q, SINA, COSA, W )              22YC
      COMMON  IM, NER, NT, NB, NCFLG, RPI, R2PI, SP, CL, ALPHA, FALPHA
      COMPLEX  IM, Z, Q, W, CLOG                                        22YC
      DIMENSION  Z(499), Q(500), SINA(499), COSA(499)                   22YC
      W = CLOG( (Z(J)-CONJG(Q(K))) / ( Z(J)-CONJG(Q(K+1))) )            22YC
      W = ( COSA(J2) + IM*SINA(J2) ) * R2PI * W                         22YC
      RETURN                                                            22YC
      END                                                               22YC
$IBFTC 22YF
      SUBROUTINE  SPGTO ( J, K, J2, SP, TF, Z, Q, SINA, COSA, W )
      COMMON  IM, NER, NT, NB, NCFLG, RPI, R2PI
      COMPLEX  IM, Z, Q, W, TF, TN, CLOG, CSINH
      DIMENSION  Z(499), Q(500), SINA(499), COSA(499)                   22YC
      TN = CSINH( 3.14159265*(Z(J)-Q(K+1)) / SP )
      W = ( COSA(J2) - IM*SINA(J2) ) * CLOG ( TF/TN ) * R2PI
      TF = TN
      RETURN                                                            22YC
      END                                                               22YC
$IBFTC 22YU
      COMPLEX FUNCTION  CSINH ( Z )
      COMPLEX  Z, CEXP
      CSINH = ( CEXP(Z) - CEXP(-Z) ) / 2.0
      RETURN
      END
$ORIGIN        ALPHA
$IBFTC 22YG
      SUBROUTINE  PART3                                                 22YG
      COMMON  IM, NER, NT, NB, NCFLG, RPI, R2PI, SF, CL, ALPHA, FALPHA
     1, DALFA, CHORD, FLG02, FLG03, FLG04, FLG05, FLG06, FLG07, FLG08
     2, FLG09, FLG10, FLG11, FLG12, ND, NLF, SUMDS, XMC, YMC, ADDY, HEDR
      INTEGER  FLG02, FLG03, FLG04, FLG05, FLG06, FLG07
     1, FLG08, FLG09, FLG10, FLG11, FLG12
```

```
      COMPLEX  IM                                                       22YG
      DIMENSION  A(499), R(459,10), SIG(499,10), KFLAG(10), DSIG1(10),  22YG
     1 ND(10), NLF(10)                                                  22YG
      REWIND  3                                                         22YG
      REWIND  4                                                         22YG
      REWIND 10                                                         22YG
      REWIND 11                                                         22YG
      M = 1                                                             22YG
      N = ND(1) - 1                                                     22YG
      DO 100 K = 1, NB                                                  22YG
      READ (4)   ( R(I,1), I = M, N )                                   22YG
      READ (4)   ( R(I,2), I = M, N )                                   22YG
      M = N + 1                                                         22YG
  100 N = N + ND(K+1) - 1                                               22YG
C     PRECEDING READS IN  SINES, COSINES. ONSET FLOWS NEXT (IF ANY).    22YG
      IF ( NCFLG .LE. 2 )  GO TO 180
      DO 150 J = 3, NCFLG                                               22YG
      READ (4)   ( R(I,J), I = 1, NT )                                  22YG
  150 READ (4)                                                          22YG
  180 DO 200 J = 2, NCFLG
      DO 200 I = 1, NT                                                  22YG
  200 R(I,J) = - R(I,J)                                                 22YG
  250 NTU = 10                                                          22YG
      ITER = 0                                                          22YG
      NCONV = 0                                                         22YG
      DO 300 J = 1, NCFLG                                               22YG
      KFLAG(J) = 0                                                      22YG
      DO 300 I = 1, NT                                                  22YG
  300 SIG(I,J) = 0.                                                     22YG
  400 DO 450 I = 1, NCFLG                                               22YG
  450 DSIG1(I) = 0.                                                     22YG
      DO 800 I = 1, NT                                                  22YG
      IF ( NTU .EQ. 10 )  GO TO 500                                     22YG
      READ (11) ( A(L), L = 1, NT )                                     22YG
      READ (11)                                                         22YG
      GO TO 600                                                         22YC
  500 READ (10) ( A(L), L = 1, NT )                                     22YG
      READ (10)                                                         22YG
  600 DO 800 J = 1, NCFLG                                               22YG
      IF ( KFLAG(J) .NE. 0 )  GO TO 800                                 22YG
      SUM = 0.                                                          22YG
      DO 700 L = 1, NT                                                  22YG
  700 SUM = SUM + A(L)*SIG(L,J)                                         22YG
      DSIG2 = ( R(I,J) - SUM ) / A(I)                                   22YG
      SIG(I,J) = SIG(I,J) + DSIG2                                       22YG
      IF ( ABS( DSIG2 ) .GT. DSIG1(J) ) DSIG1(J) = ABS( DSIG2 )
  800 CONTINUE                                                          22YG
      REWIND 10                                                         22YG
      REWIND 11                                                         22YG
      ITER = ITER + 1                                                   22YG
      DO 1000 J = 1, NCFLG                                              22YG
      IF ( KFLAG(J) .NE. 0 )  GO TO 1000                                22YG
      IF ( DSIG1(J) .GE. 1.E-4 )  GO TO 1000
      KFLAG(J) = ITER                                                   22YG
      NCONV = NCONV + 1                                                 22YG
      IF ( NCONV .EQ. NCFLG )  GO TO 1400                               22YG
 1000 CONTINUE                                                          22YG
```

Figure 15. - Continued

```
      IF ( ITER .EQ. 100 )  GO TO 1400                                  22YG
      IF ( NTU .EQ. 10 )  GO TO 1200                                    22YG
      NTU = 10                                                          22YG
      GO TO 400                                                         22YG
 1200 NTU = 11                                                          22YG
      GO TO 400                                                         22YG
 1400 DO 1600 J = 1, NCFLG                                              22YG
      IF ( KFLAG(J) .NE. 0 )  GO TO 1500                                22YG
      WRITE (6, 10)                                                     22YG
   10 FORMAT (1HO 7X 36H NO CONVERGENCE AFTER 100 ITERATIONS )          22YG
      GO TO 1600                                                        22YG
 1500 WRITE (6, 15)  KFLAG(J)                                           22YG
   15 FORMAT (1HO 5X I5, 1X 36H ITERATIONS REQUIRED FOR CONVERGENCE )   22YG
 1600 WRITE (3. ( SIG(I,J), I = 1, NT )                                 22YG
      RETURN                                                            22YG
      END                                                               22YG
$ORIGIN       ALPHA
$IBFTC 22YH
      SUBROUTINE  PART4                                                 22YH
      COMMON  IM, NER, NT, NB, NCFLG, RPI, R2PI, SP, CL, ALPHA, FALPHA
     1, DALFA, CHORD, FLG02, FLG03, FLG04, FLG05, FLG06, FLG07, FLG08
     2, FLG09, FLG10, FLG11, FLG12, ND, NLF, SUMDS, XMC, YMC, ADDY, HEDR
      INTEGER  FLG02, FLG03, FLG04, FLG05, FLG06, FLG07
     1, FLG08, FLG09, FLG10, FLG11, FLG12
      COMPLEX  IM                                                       22YH
      DIMENSION  A(135,135), R(135,10), ND(10), NLF(10)                 22YH
      REWIND  3                                                         22YH
      REWIND  4                                                         22YH
      REWIND 10                                                         22YH
      M = 1                                                             22YH
      N = ND(1) - 1                                                     22YH
      DO 100 K = 1, NB                                                  22YH
      READ (4) ( R(I,1), I = M, N )                                     22YH
      READ (4) ( R(I,2), I = M, N )                                     22YH
      M = N + 1                                                         22YH
  100 N = N + ND(K+1) - 1                                               22YH
C     PRECEDING REАDS IN SINES, COSINES. ONSET FLOWS NEXT (IF ANY).    22YH
      IF ( NCFLG .LE. 2 )  GO TC 180
      DO 150 J = 3, NCFLG                                               22YH
      READ (4) ( R(I,J), I = 1, NT )                                    22YH
  150 READ (4)                                                          22YH
  180 DO 200 J = 2, NCFLG
      DO 200 I = 1, NT                                                  22YH
  200 R(I,J) = - R(I,J)                                                 22YH
  250 DO 300 I = 1, NT                                                  22YH
      READ (10) ( A(I,J), J = 1, NT )                                   22YH
  300 READ (10)                                                         22YH
      DET = 1.                                                          22YH
      CALL MIS1 ( A, NT, 135, R, NCFLG, NERR, DET )                     22YH
      IF ( NERR .NE. 0 )  WRITE (6, 350)                                22YH
  350 FORMAT (1H1 23H MATRIX AJK IS SINGULAR )                          22YH
      DO 500 J = 1, NCFLG                                               22YH
  500 WRITE (3) ( R(I,J), I = 1, NT )                                   22YH
      RETURN                                                            22YH
      END                                                               22YH
$IBFTC 22YT
```

```
      SUBROUTINE MIS1 ( A, N, NDD, B, M, NERR, D )
      DIMENSION  A(1), B(1)                                             MIS10130
      EQUIVALENCE  (I,FI), (K,FK)                                       MIS10140
      NERR =  1                                                         MIS10150
      ND = NDD                                                          MIS10160
   10 DO 90  I=1,N                                                      MIS10180
      AIJMAX = A(I)                                                     MIS10200
      IJMAX = I                                                         MIS10210
      DO 25  J=2,N                                                      MIS10220
      IJ = I + (J-1)*ND                                                 MIS10230
      IF  ( ABS ( A(IJ) )  -  ABS  ( AIJMAX ) )   25,25,20              MIS10240
   20 AIJMAX = A(IJ)                                                    MIS10250
      IJMAX = IJ                                                        MIS10260
   25 CONTINUE                                                          MIS10270
      IF  (AIJMAX)  30,999,30                                           MIS10290
   30 DO 35  J=1,N                                                      MIS10310
      IJ = I + (J-1)*ND                                                 MIS10320
   35 A(IJ) = A(IJ)/AIJMAX                                              MIS10330
      D = D * AIJMAX                                                    MIS10340
      DO 40  J=1,M                                                      MIS10360
      IJ = I + (J-1)*ND                                                 MIS10370
   40 B(IJ) = B(IJ)/AIJMAX                                              MIS10380
      DO 70  K=1,N                                                      MIS10420
      IF  (K-I)  50,70,50                                               MIS10430
   50 KJMAX = IJMAX + (K-I)                                             MIS10440
      ARAT  =  -A(KJMAX)                                                MIS10450
      KJ = K                                                            MIS10460
      IJ = I                                                            MIS10470
      DO 60   J=1,N                                                     MIS10480
      IF  (A(IJ))  55,58,55                                             MIS10490
   55 A(KJ) = ARAT*A(IJ) + A(KJ)                                        MIS10500
   58 KJ = KJ + ND                                                      MIS10510
   60 IJ = IJ + ND                                                      MIS10520
      A(KJMAX) = 0.0                                                    MIS10530
      KJ = K                                                            MIS10540
      IJ = I                                                            MIS10550
      DO 69 J=1,M                                                       MIS10560
      IF (B(IJ))  65,68,65                                              MIS10570
   65 B(KJ) = ARAT*B(IJ) + B(KJ)                                        MIS10580
   68 KJ = KJ + ND                                                      MIS10590
   69 IJ = IJ + ND                                                      MIS10600
   70 CONTINUE                                                          MIS10610
      KJ = IJMAX - I+1                                                  MIS10620
   90 A(KJ) = FI                                                        MIS10660
      DO 100  I=1,N                                                     MIS10700
      K = I                                                             MIS10710
   93 I1 = K*ND - ND + 1                                                MIS10720
      FK = A(I1)                                                        MIS10730
      IF (K-I)  93,100,95                                               MIS10750
   95 IJ = I                                                            MIS10800
      IK = K                                                            MIS10810
      DO 99  J=1,M                                                      MIS10820
      A(2) = B(IJ)                                                      MIS10830
      B(IJ) = B(IK)                                                     MIS10840
      B(IK) = A(2)                                                      MIS10850
      IJ = IJ + ND                                                      MIS10860
   99 IK = IK + ND                                                      MIS10870
```

Figure 15. - Continued

```
  100 CONTINUE                                                           MIS10880
      NERR  =  0                                                         MIS10890
  999 RETURN                                                             MIS10900
      END                                                                MIS10910
$ORIGIN        ALPHA
$IBFTC 22YI
      SUBROUTINE  PART5
      COMMON  IM, NER, NT, NB, NCFLG, RPI, R2PI, SP, CL, ALPHA, FALPHA
     1, DALFA, CHORD, FLG02, FLG03, FLG04, FLG05, FLG06, FLG07, FLG08
     2, FLG09, FLG10, FLG11, FLG12, ND, NLF, SUMDS, XMC, YMC, ADDY, HEDR
     3, CASE
      INTEGER  FLG02, FLG03, FLG04, FLG05, FLG06, FLG07
     1, FLG08, FLG09, FLG10, FLG11, FLG12
      COMPLEX  IM                                                        22YI
      EQUIVALENCE  ( T, CP )                                             22YI
      DIMENSION  B(499), VT(499,10), SIG(499,10), T(499,10), CP(499,11)  22YI
     1, ND(10), NLF(10), X(500), Y(500), XMP(499), YMP(499)              22YI
     2, SUMDS(10), XMC(8), YMC(8), ADDY(8), HEDR(8)                      22YI
      REWIND  3                                                          22YI
      REWIND  4                                                          22YI
      REWIND 10                                                          22YI
      REWIND 11                                                          22YI
      REWIND 12                                                          22YI
      REWIND 13
      M = 1                                                              22YI
      N = ND(1) - 1                                                      22YI
      DO 100 K = 1, NB                                                   22YI
      READ (4) ( T(I,2), I = M, N )                                      22YI
      READ (4) ( T(I,1), I = M, N )                                      22YI
C     READS IN SINES AND COSINES                                         22YI
      M = N + 1                                                          22YI
  100 N = N + ND(K+1) - 1                                                22YI
      IF ( NCFLG .LE. 2 )  GO TO 200                                     22YI
      DO 150 J = 3, NCFLG                                                22YI
      READ (4)                                                           22YI
  150 READ (4) ( T(I,J), I = 1, NT )                                     22YI
  200 DO 250 J = 1, NCFLG                                                22YI
  250 READ (3) ( SIG(I,J), I = 1, NT )                                   22YI
      DO 400 I = 1, NT                                                   22YI
      READ (10)                                                          22YI
      READ (10) ( B(L), L = 1, NT )                                      22YI
      DO 400 J = 1, NCFLG                                                22YI
      PR = 0.                                                            22YI
      DO 300 L = 1, NT                                                   22YI
  300 PR = PR + B(L)*SIG(L,J)                                            22YI
      VT(I,J) = PR + T(I,J)                                              22YI
  400 CP(I,J) = 1. - VT(I,J)**2                                          22YI
      DO 500 J = 1, NCFLG                                                22YI
  500 WRITE (11) ( VT(I,J), I = 1, NT )                                  22YI
      M = 1                                                              22YI
      N = ND(1)                                                          22YI
      M1 = 1                                                             22YI
      N1 = ND(1) - 1                                                     22YI
      DO 700 J = 1, NB                                                   22YI
      READ (13) ( X(I), I = M, N )
      READ (13) ( Y(I), I = M, N )
      READ (13) ( XMP(I), I = M1, N1 )
      READ (13) ( YMP(I), I = M1, N1 )
      M = N + 1                                                          22YI
      N = N + ND(J+1)                                                    22YI
      M1 = N1 + 1                                                        22YI
  700 N1 = N1 + ND(J+1) - 1                                              22YI
      DO 2500 L = 1, NCFLG                                               22YI
      I = 1                                                              22YI
      J = 1                                                              22YI
      M = 1                                                              22YI
      N = ND(M)                                                          22YI
      LCTR = 22                                                          22YI
      K = L - 2                                                          22YI
 1000 WRITE (6, 1100)  HEDR, CASE
 1100 FORMAT (1H1 25X 26HDOUGLAS  AIRCRAFT  COMPANY / 28X 21HLONG  BEACH22YI
     1  DIVISION //// 5X 8A6 // 6H CASE  A6 )
      IF ( K .GT. 0 )   GO TO 1500                                       22YI
      IF ( K .EQ. 0 )   GO TO 1300                                       22YI
      WRITE (6, 1200)                                                    22YI
 1200 FORMAT (1H 19HSTREAMFLOW SOLUTION )
      GO TO 1700                                                         22YI
 1300 WRITE (6, 1400)                                                    22YI
 1400 FORMAT (1H 23H90-DEGREE FLOW SOLUTION )
      GO TO 1700                                                         22YI
 1500 WRITE (6, 1600)  K                                                 22YI
 1600 FORMAT (1H 35HNON-UNIFORM ONSET FLOW SOLUTION NO. I3 )
 1700 WRITE (6, 1800)                                                    22YI
 1800 FORMAT (1H 25HUNTRANSFORMED COORDINATES // 12X 1HX 13X 1HY 14X 1HV
     1 12X 2HCP 11X 5HSIGMA // )
 2000 WRITE (6, 2100) I, X(I), Y(I), XMP(J), YMP(J), VT(J,L)             22YI
     1, CP(J,L), SIG(J,L)                                                22YI
 2100 FORMAT (1H  I3, 2F14.8 / 4X 5F14.8 )                               22YI
      I = I + 1                                                          22YI
      J = J + 1                                                          22YI
      IF ( I .EQ. N )  GO TO 2200                                        22YI
      IF ( I .LE. LCTR)  GO TO 2000                                      22YI
      LCTR = LCTR + 22                                                   22YI
      GO TO 1000                                                         22YI
 2200 M = M + 1                                                          22YI
      N = N + ND(M)                                                      22YI
      WRITE (6, 2300)  I, X(I), Y(I)                                     22YI
 2300 FORMAT (1H  I3, 2F14.8 // )                                        22YI
      I = I + 1                                                          22YI
      IF ( J .GE. NT )  GO TO 2500                                       22YI
      GO TO 2000                                                         22YI
 2500 CONTINUE                                                           22YI
      RETURN                                                             22YI
      END                                                                22YI
$ORIGIN        ALPHA,REW
$IBFTC 22YJ
      SUBROUTINE  PART6                                                  22YJ
      COMMON  IM, NER, NT, NB, NCFLG, RPI, R2PI, SP, CL, ALPHA, FALPHA
     1, DALFA, CHORD, FLG02, FLG03, FLG04, FLG05, FLG06, FLG07, FLG08
     2, FLG09, FLG10, FLG11, FLG12, ND, NLF, SUMDS, XMC, YMC, ADDY, HEDR
     3, CASE
      COMPLEX  IM                                                        22YJ
      INTEGER  FLG02, FLG03, FLG04, FLG05, FLG06, FLG07
     1, FLG08, FLG09, FLG10, FLG11, FLG12
```

Figure 15. - Continued

```
      DIMENSION  VC(499), X(500), Y(500), XMP(499), YMP(499), XM(499)   22YJ
     1, YM(499), SINA(499), COSA(499), CP(499), DELS(499), ND(10)       22YJ
     2, NLF(10), SUMDS(10), GAM(9), XMC(8), YMC(8), HEDR(8), ADDY(8)    22YJ
     3, DVT(9,10), DV(9,9), DELSUT(507), SIGT(499), XTEMP(3992)         22YJ
     4, YTEMP(3992), VXL(499), VYL(499), XIJ(499), YIJ(499), DVA(9,8)   22YJ
     5, DVX(9,10), ZTEMP(500)
      EQUIVALENCE  (GAM, DVT), (DV, DVT(10)), (DVA, DVT(19))            22YJ
     1, (VXL, XMP), (VYL, YMP), (SIGT, XM), (XIJ, SINA), (YIJ, COSA)    22YJ
     2, (ZTEMP, XTEMP(500))                                             22YJ
      IF ( FLG08 .EQ. 0 )  GO TO 40
      IF ( FLG02 .EQ. 0 )  GO TO 40
      REWIND 10
      J = 2 * NT
      DO 30 I = 1, J
   30 READ (10)
C **  LOOP SKIPS BOTH ON-BODY MATRICES
   40 NER = 0
      DET = 1.                                                          22YJ
      K = NCFLG - 2                                                     22YJ
      REWIND  3                                                         22YJ
      REWIND  4                                                         22YJ
      REWIND 11                                                         22YJ
      REWIND 12                                                         22YJ
      REWIND 13                                                         22YJ
      DO 50 J = 1, 10                                                   22YJ
      DO 50 I = 1, 9                                                    22YJ
      DVX(I,J) = 0.
   50 DVT(I,J) = 0.                                                     22YJ
      IF (FLG03 .EQ. 0 .AND. FLG04 .EQ. 0 .AND. FLG06 .EQ. 0) GO TO 1000
      IF ( FLG04 .NE. 0 )  ALPHA = DALFA
      IF ( FLG06 .NE. 0 )  ALPHA = 0.
      ALPHA = ALPHA / 57.2957795                                        22YJ
      CSALF = COS(ALPHA)                                                22YJ
      SNALF = SIN(ALPHA)                                                22YJ
      READ (11) ( XTEMP(I), I = 1, NT )                                 22YJ
      READ (11) ( YTEMP(I), I = 1, NT )                                 22YJ
      M = 1                                                             22YJ
      N = ND(1) - 1                                                     22YJ
      DO 100 I = 1, NB                                                  22YJ
      GAM(I) = (XTEMP(M) + XTEMP(N))*CSALF                              22YJ
      GAM(I) = - ( GAM(I) + (YTEMP(M)+YTEMP(N)) *SNALF )                22YJ
      M = N+1                                                           22YJ
  100 N = N + ND(I+1) - 1                                               22YJ
      IF ( K .EQ. 0 )  GO TO 160                                        22YJ
      DO 150 J = 1, K
      READ (11) ( ZTEMP(I), I = 1, NT )                                 22YJ
      M = 1                                                             22YJ
      N = ND(1) - 1                                                     22YJ
      DO 150 I = 1, NB                                                  22YJ
      DVA(I,J) = ZTEMP(M) + ZTEMP(N)
      DVX(I,J) = DVA(I,J)
      M = N+1                                                           22YJ
  150 N = N + ND(I+1) - 1
  160 REWIND 11                                                         22YJ
      IF ( K .EQ. NB )  GO TO 200                                       22YJ
      M = NCFLG - 1                                                     22YJ
      DO 170 I = M, NB                                                  22YJ
      DVA(I,I) = 1.
  170 DVX(I,I) = 1.
  200 CALL MISS (DVA, NB, 9, GAM, 1, NER, DET)                          22YJ
      IF ( NER .NE. 0 )  WRITE (6, 210)                                 22YJ
  210 FORMAT (1H1 38HCOMBINATION PROGRAM MATRIX IS SINGULAR )           22YJ
      IF ( FLG03 .NE. 0 )  GO TO 5000
      IF ( FLG06 .NE. 0 )  GO TO 2000
      DALFA = DALFA / 57.2957795
      T = 0.                                                            22YJ
      DO 220 I = 1, K                                                   22YJ
  220 T = T + GAM(I)                                                    22YJ
      T = .5*T / SP                                                     22YJ
      DAL1 = ATAN2( 2.*T*CSALF, 1.- T*T )
      IF ( ABS(DAL1 - DALFA) .LT. .0005 )  GO TO 420
      ALF1 = ALPHA                                                      22YJ
      ALF2 = ALF1 + 2.*(DALFA-DAL1)                                     22YJ
      ITER = 0                                                          22YJ
  250 CSALF = COS( ALF2 )                                               22YJ
      SNALF = SIN( ALF2 )                                               22YJ
      M = 1                                                             22YJ
      N = ND(1) - 1                                                     22YJ
      DO 300 I = 1, NB                                                  22YJ
      GAM(I) = ( XTEMP(M)+XTEMP(N)) * CSALF                             22YJ
      GAM(I) = - ( GAM(I) + (YTEMP(M)+YTEMP(N)) *SNALF )                22YJ
      M = N + 1                                                         22YJ
  300 N = N + ND(I+1) - 1                                               22YJ
      DO 320 I = 1, NB
      DO 320 J = 1, NB
  320 DVA(I,J) = DVX(I,J)
      DET = 1.                                                          22YJ
      CALL MISS (DVA, NB, 9, GAM, 1, NER, DET)                          22YJ
      IF ( NER .NE. 0 )  WRITE (6, 210)
      T = 0.                                                            22YJ
      DO 350 I = 1, K                                                   22YJ
  350 T = T + GAM(I)                                                    22YJ
      T = .5*T / SP                                                     22YJ
      DAL2 = ATAN2( 2.*T*CSALF, 1.- T*T )
      IF ( ABS(DAL2 - DALFA) .GE. .0005 )  GO TO 450
  400 ALPHA = ALF2
  420 DALFA = DALFA * 57.2957795
      GO TO 5000                                                        22YJ
  450 ITER = ITER + 1                                                   22YJ
      IF ( ITER .LT. 20 )  GO TO 550                                    22YJ
      WRITE (6, 500)                                                    22YJ
  500 FORMAT (1H1 65HALPHA COMPUTATION ITERATIONS EXCEED 20. LAST VALUE 22YJ
     1OF ALPHA USED. )                                                  22YJ
      GO TO 400
  550 T = ALF1*(DAL2 - DALFA) / (DAL2 - DAL1)                           22YJ
     1  + ALF2*(DALFA - DAL1) / (DAL2 - DAL1)                           22YJ
      ALF1 = ALF2                                                       22YJ
      ALF2 = T                                                          22YJ
      DAL1 = DAL2                                                       22YJ
      GO TO 250                                                         22YJ
 1000 IF ( FLG05 .NE. 0 )  GO TO 1005
      WRITE (6, 1002)
 1002 FORMAT (1H1 30HFLAGS 3, 4, 5, AND 6 ARE ZERO. //
     1 36H COMBINATION PROGRAM CANNOT PROCEED. )
```

Figure 15. - Continued

```
      :ETURN
 1005 IF ( FALPHA .NE. 90.0 .AND. FALPHA .NE. 270. )  GO TC 1020          22YJ
      WRITE (6, 1010)                                                     22YJ
 1010 FORMAT (1H1 32HUNALLOWAB    ..ET ALPHA IS INPUT )                   22YJ
      RETURN                                                              22YJ
 1020 FALPHA = FALPHA / 57.2957                                           22YJ
      TNA = SIN(FALPHA) / COS(FA    ))
      DO 1040 J = 1, NCFLG                                                22YJ
      READ (11) (XTEMP(I), I - 1, NT)                                     22YJ
      M = 1                                                               22YJ
      N = ND(1) - 1                                                       22YJ
      DO 1040 I = 1, NB                                                   22YJ
      DVT(I,J) = XTEMP(M) + XTEMP(N)                                      22YJ
      M = N+1                                                             22YJ
 1040 N = N+ND(I+1) - 1                                                   22YJ
      REWIND 11                                                           22YJ
      I = NB + 1                                                          22YJ
      DVT(I,1) = - TNA
      DVT(I,2) = 1.                                                       22YJ
      IF ( K .EQ. 0 )  GO TO 1120                                         22YJ
      F = .5/SP                                                           22YJ
      DO 1100 J = 3, NCFLG                                                22YJ
 1100 DVT(I,J) = F                                                        22YJ
 1120 DO 1140 J = 1, I                                                    22YJ
 1140 GAM(J) = - GAM(J)                                                   22YJ
      IF ( K .EQ. NB ) GO TO 1200                                         22YJ
      DO 1160 J = NCFLG, I
 1160 DV(J-1,J) = 1.
 1200 CALL MISS (DV, I, 9, GAM, 1, NER, DET)                              22YJ
      IF ( NER .NE. 0 )  WRITE (6, 210)                                   22YJ
C     GAM(1) = TAN (ALPHA),   GAM(2) = GAM(1)/COS ALPHA--SOLVE FOR GAM(1)22YJ
      ALPHA = ATAN( GAM(1) )                                              22YJ
      CSALF = COS( ALPHA )                                                22YJ
      DO 1220 I = 1, K                                                    22YJ
 1220 GAM(I) = GAM(I+1) * CSALF                                           22YJ
      GO TO 5000                                                          22YJ
 2000 CL1 = 0.
      DO 2020 I = 1, K
 2020 CL1 = CL1 + GAM(I)
      CL1 = 2.*CL1
      IF ( ABS(CL - CL1) .LT. .0005)  GO TO 5C00
      ALF1 = 0.
      ALF2 = .2 * (CL - CL1)
      ITER = 0
 2050 CSALF = COS( ALF2 )
      SNALF = SIN( ALF2 )
      M = 1
      N = ND(1) - 1
      DO 2100 I = 1, NB
      GAM(I) = ( XTEMP(M)+XTEMP(N) ) * CSALF
      GAM(I) = - ( GAM(I) + (YTEMP(M)+YTEMP(N))*SNALF )
      M = N + 1
 2100 N = N + ND(I+1) - 1
      DO 2150 I = 1, NB
      DO 2150 J = 1, NB
 2150 DVA(I,J) = DVX(I,J)
      DET = 1.
      CALL MISS ( DVA, NB, 9, GAM, 1, NER, DET )
      IF ( NER .NE. 0 )  WRITE (6, 210)
      CL2 = 0.
      DO 2200 J = 1, K
 2200 CL2 = CL2 + GAM(J)
      CL2 = 2.*CL2
      IF ( ABS(CL2 - CL) .LT. .0005 )  GO TO 2600
      ITER = ITER + 1                                                     22YJ
      IF (ITER .GE. 20)  GO TO 2550
      T = ALF1*(CL2 - CL) / (CL2 - CL1) + ALF2*(CL - CL1) / (CL2 - CL1) 22YJ
      ALF1 = ALF2                                                         22YJ
      ALF2 = T                                                            22YJ
      CL1 = CL2                                                           22YJ
      GO TO 2050
 2550 WRITE (6, 500)                                                      22YJ
 2600 ALPHA = ALF2                                                        22YJ
 5000 IF ( K .EQ. NB )  GO TO 5040                                        22YJ
      M = NCFLG - 1                                                       22YJ
      DO 5020 I = M, NB                                                   22YJ
 5020 GAM(I) = 0.                                                         22YJ
 5040 READ (11) ( XTEMP(I), I = 1, NT )
      SNALF = SIN( ALPHA )                                                22YJ
      READ (11) ( YTEMP(I), I = 1, NT )
      DO 5070 I = 1, NT
 5070 VC(I) = XTEMP(I)*CSALF + YTEMP(I)*SNALF
      IF ( K .EQ. 0 )  GO TO 5150                                         22YJ
      DO 5100 L = 1, K                                                    22YJ
      READ (11) ( XTEMP(I), I = 1, NT )                                   22YJ
      DO 5100 I = 1, NT                                                   22YJ
 5100 VC(I) = VC(I) + XTEMP(I)*GAM(L)                                     22YJ
 5150 DO 5200 I = 1, NT                                                   22YJ
 5200 CP(I) = 1. - VC(I)*VC(I)                                            22YJ
      M = 1                                                               22YJ
      N = ND(1) - 1                                                       22YJ
      M1 = 1                                                              22YJ
      N1 = ND(1)                                                          22YJ
      DO 5250 J = 1, NB                                                   22YJ
      READ (4) ( SINA(I), I = M, N )                                      22YJ
      READ (13) ( X(I), I = M1, N1 )                                      22YJ
      READ (4) ( COSA(I), I = M, N )                                      22YJ
      READ (13) ( Y(I), I = M1, N1 )                                      22YJ
      READ (13) ( XMP(I), I = M, N )                                      22YJ
      READ (13) ( YMP(I), I = M, N )                                      22YJ
      READ (12) ( XM(I), I = M, N )                                       22YJ
      READ (12) ( YM(I), I = M, N )                                       22YJ
      READ (12) ( DCLS(I), I = M, N )                                     22YJ
      M1 = N1 + 1                                                         22YJ
      N1 = N1 + ND(J+1)                                                   22YJ
      M = N + 1                                                           22YJ
 5250 N = N + ND(J+1) - 1                                                 22YJ
      J = NT + NB                                                         22YJ
      DO 5300 I = 1, J                                                    22YJ
 5300 DELSUT(I) = SQRT( (X(I+1)-X(I))**2 + (Y(I+1)-Y(I))**2 )             22YJ
      GT = 0.
      DO 5350 I = 1, K
 5350 GT = GT + GAM(I)
```

Figure 15. - Continued

```
      T = GT/SP * .5
      IF ( FLG05 .EQ. 0 ) FALPHA= ATAN2 (SNALF+T , CSALF)             22YJ
      ALFEX = ATAN2 (SNALF - T, CSALF)                                 22YJ
      ALFEX = ALFEX * 57.2957795
      FALPHA = FALPHA * 57.2957795                                     22YJ
      ALPHA = ALPHA * 57.2957795
      IF ( FLG04 .EQ. 0 )  DALFA = FALPHA - ALFEX
      VIN = SQRT ( 1. + 2.*SNALF*T + T*T )                             22YJ
      VEX = SQRT ( 1. - 2.*SNALF*T + T*T )                             22YJ
      J = 1                                                            22YJ
      K1 = 1                                                           22YJ
      M = 1                                                            22YJ
      N = ND(1) - 1                                                    22YJ
      DO 5800 L = 1, NB                                                22YJ
      LCTR = 19
      CL = 0                                                           22YJ
      CD = 0                                                           22YJ
      CM = 0                                                           22YJ
      DO 5400 I = M, N                                                 22YJ
      T = CP(I) * DELS(I)                                              22YJ
      CL = CL - T*COSA(I)                                              22YJ
      CD = CD + T*SINA(I)                                              22YJ
 5400 CM = CM + T*(COSA(I)*(XM(I)-XMC(L)) + SINA(I)*(YM(I)-YMC(L)) )   22YJ
      I = 1                                                            22YJ
      K2 = ND(L)                                                       22YJ
 5500 WRITE (6, 5550)  HEDR, SP, ALPHA, DALFA, FALPHA, VIN, XMC(L),
     1 ALFEX, VEX, YMC(L), CASE, L
 5550 FORMAT (1H1 25X 26HDOUGLAS  AIRCRAFT  COMPANY / 28X 21HLONG  BEACH22YJ
     1  DIVISION /// 5X 8A6 // 5X 9HSPACING = F13.8, 5X 7HALPHA = F13.8
     2, 5X 13HDELTA ALPHA = F13.8 // 14H INLET ALPHA = F13.8, 3X
     3 9HV INLET = F13.8, 13X 5HXMC = F13.8 // 2X 12HEXIT ALPHA = F13.8,
     4 4X 8HV EXIT = F13.8, 13X 5HYMC = F13.8 // 6H CASE A6, 22H    COMBI
     5NED VELOCITIES / 10H  BODY NO. I3, 27H  UNTRANSFORMED COORDINATES
     6// 11X 1HX 13X 1HY 13X 2HVC 12X 2HCP 10X 7HDELTA S // )
 5600 WRITE (6, 5650) I, X(J), Y(J), XMP(K1), YMP(K1), VC(K1), CP(K1)  22YJ
     1, DELSUT(J)                                                      22YJ
 5650 FORMAT (1H  I3, 2F14.8 / 4X 5F14.8 )                             22YJ
      I = I + 1                                                        22YJ
      J = J + 1                                                        22YJ
      K1 = K1 + 1                                                      22YJ
      IF ( I .EQ. K2 )  GO TO 5700                                     22YJ
      IF ( I .LE. LCTR )  GO TO 5600                                   22YJ
      LCTR = LCTR + 19
      GO TO 5500                                                       22YJ
 5700 WRITE (6, 5650)  I, X(J), Y(J)                                   22YJ
      J = J + 1                                                        22YJ
      WRITE (6, 5750)  CL, CD, CM                                      22YJ
 5750 FORMAT (1H0 / 5X 4HCY = F13.8, 4X 4HCX = F13.8, 4X 4HCM = F13.8 ) 22YJ
      M = N + 1                                                        22YJ
 5800 N = N + ND(L+1) - 1                                              22YJ
C     K = NCFLG-2 = NUMBER OF GAMMAS                                   22YJ
      CL = 2.*GT
      WRITE (6, 5840)  CL
 5840 FORMAT (1H0 4X 4HCL = F13.8 )
 5850 IF ( FLG02 .EQ. 0 )  RETURN
      N = ND(NB+1)
      READ (3) ( XTEMP(I), I = 1, NT )
      READ (3) ( YTEMP(I), I = 1, NT )
      DO 5870 I = 1, NT
 5870 SIGT(I) = XTEMP(I)*CSALF + YTEMP(I)*SNALF
      IF ( K .EQ. 0 )  GO TO 6150
      DO 6000 J = 1, K
      READ (4)
      READ (4)
C ** PRECEDING 2 READS WILL SKIP ON-BODY NON-UNIFORM ONSET FLOW
      READ (3) ( XTEMP(I), I = 1, NT)
      DO 6000 I = 1, NT
 6000 SIGT(I) = SIGT(I) + XTEMP(I)*GAM(J)
      M = 1
      M1 = N
      DO 6100 J = 1, K
      READ (4) ( XTEMP(I), I = M, M1 )
      READ (4) ( YTEMP(I), I = M, M1 )
      M = M1 + 1
 6100 M1 = M1 + N
 6150 DO 6400 J = 1, N
      READ (10) ( YIJ(I), I = 1, NT )
      READ (10) ( XIJ(I), I = 1, NT )
      SUM1 = 0.
      SUM2 = 0.
      DO 6200 I = 1, NT
      T = SIGT(I)
      SUM1 = SUM1 + T*XIJ(I)
 6200 SUM2 = SUM2 + T*YIJ(I)
      IF ( K .EQ. 0 )  GO TO 6300
      N1 = J
      DO 6250 I = 1, K
      T = GAM(I)
      SUM1 = SUM1 + T*YTEMP(N1)
      SUM2 = SUM2 + T*XTEMP(N1)
 6250 N1 = N1 + N
 6300 VXL(J) = SUM1 + CSALF
 6400 VYL(J) = SUM2 + SNALF
      READ (13) ( X(I), I = 1, N )
      READ (13) ( Y(I), I = 1, N )
      LCTR = 45
      I = 1
 6500 WRITE (6, 6550)  HEDR, SP, ALPHA
 6550 FORMAT (1H1 25X 26HDOUGLAS  AIRCRAFT  COMPANY / 28X 21HLONG  BEACH
     1  DIVISION /// 5X 8A6 // 5X 9HSPACING = F13.8, 4X 7HALPHA = F13.8 22YJ
     2 ///28H    OFF-BODY POINT VELOCITIES // 11X 1HX 13X 1HY 12X 3HVXL
     3 11X 3HVYL // )
 6600 WRITE (6, 6650)  I, X(I), Y(I), VXL(I), VYL(I)
 6650 FORMAT (1H  I3, 4F14.8 )
      I = I + 1
      IF ( I .GT. N )  RETURN
      IF ( I .LE. LCTR )  GO TO 6600
      LCTR = LCTR + 45
      GO TO 6500
      END
$IBFTC 22YS
      SUBROUTINE MISS ( A, N, NDD, B, M, NERR, D )
      DIMENSION  A(1), B(1)                                            MIS10130
```

Figure 15. - Continued

```
      EQUIVALENCE  (I,FI), (K,FK)                                     MIS10140
      NERR =  1                                                       MIS10150
      ND = NDD                                                        MIS10160
   10 DO 90  I=1,N                                                    MIS10180
      AIJMAX = A(I)                                                   MIS10200
      IJMAX = I                                                       MIS10210
      DO 25  J=2,N                                                    MIS10220
      IJ = I + (J-1)*ND                                               MIS10230
      IF  ( ABS ( A(IJ) )  -  ABS  ( AIJMAX ) )   25,25,20            MIS10240
   20 AIJMAX = A(IJ)                                                  MIS10250
      IJMAX = IJ                                                      MIS10260
   25 CONTINUE                                                        MIS10270
      IF  (AIJMAX)  30,999,30                                         MIS10290
   30 DO 35  J=1,N                                                    MIS10310
      IJ = I + (J-1)*ND                                               MIS10320
   35 A(IJ) = A(IJ)/AIJMAX                                            MIS10330
      D = D * AIJMAX                                                  MIS10340
      DO 40  J=1,M                                                    MIS10360
      IJ = I + (J-1)*ND                                               MIS10370
   40 B(IJ) = B(IJ)/AIJMAX                                            MIS10390
      DO 70  K=1,N                                                    MIS10420
      IF  (K-I)  50,70,50                                             MIS10430
   50 KJMAX = IJMAX + (K-I)                                           MIS10440
      ARAT  =  -A(KJMAX)                                              MIS10450
      KJ = K                                                          MIS10460
      IJ = I                                                          MIS10470
      DO 60   J=1,N                                                   MIS10480
      IF  (A(IJ))  55,58,55                                           MIS10490
   55 A(KJ) = ARAT*A(IJ) + A(KJ)                                      MIS10500
   58 KJ = KJ + ND                                                    MIS10510
   60 IJ = IJ + ND                                                    MIS10520
      A(KJMAX) = 0.0                                                  MIS10530
      KJ = K                                                          MIS10540
      IJ = I                                                          MIS10550
      DO 69 J=1,M                                                     MIS10560
      IF (B(IJ))  65,68,65                                            MIS10570
   65 B(KJ) = ARAT*B(IJ) + B(KJ)                                      MIS10580
   68 KJ = KJ + ND                                                    MIS10590
   69 IJ = IJ + ND                                                    MIS10600
   70 CONTINUE                                                        MIS10610
      KJ = IJMAX - I+1                                                MIS10620
   90 A(KJ) = FI                                                      MIS10660
      DO 100  I=1,N                                                   MIS10700
      K = I                                                           MIS10710
   93 I1 = K*ND - ND + 1                                              MIS10720
      FK = A(I1)                                                      MIS10730
      IF (K-I)  93,100,95                                             MIS10750
   95 IJ = I                                                          MIS10800
      IK = K                                                          MIS10810
      DO 99  J=1,M                                                    MIS10820
      A(2) = B(IJ)                                                    MIS10830
      B(IJ) = B(IK)                                                   MIS10840
      B(IK) = A(2)                                                    MIS10850
      IJ = IJ + ND                                                    MIS10860
   99 IK = IK + ND                                                    MIS10870
  100 CONTINUE                                                        MIS10880

      NERR  =   0                                                     MIS10890
  999 RETURN                                                          MIS10900
      END                                                             MIS10910
```

Figure 15. - Continued

DISTRIBUTION LIST

REPORT NO. LB 31653

Chief of Naval Research
Fluid Dynamics Branch, (Code 438)
Department of the Navy
Washington 25, D.C. (2 copies)

Chief, Bureau of Naval Weapons
Dynamics Section (Code RAAD-222)
Attention: Mr. D. Michel
Washington 25, D.C.

Commander
U.S. Naval Ordnance Laboratory
White Oak, Maryland

Commander
U.S. Naval Ordnance Test Station
China Lake, California

Officer-in-charge, Pasadena Annex
U.S. Naval Ordnance Test Station
Oceanic Research (Code P-508)
3202 E. Foothill Blvd.
Pasadena 8, California (2 copies)

National Bureau of Standards
Washington 25, D.C.
Att: Dr. G. B. Schubauer, Chief
Fluid Mechanics Section

National Bureau of Standards
Washington 25, D.C.
Att: Dr. J. M. Franklin, Consultant

Director
Langley Research Center
Langley Field, Virginia
Att: Mr. I. E. Garrick

Director
Langley Research Center
Langley Field, Virginia
Att: Mr. D. J. Marten

National Research Council of Canada
Hydromechanics Laboratory
Ottawa 2, Canada

Mr. R. P. Godwin, Acting Chief
Office of Research & Development
Maritime Administration
441 G. Street, N.W.
Washington 25, D.C.

Office of Technical Services
OTS, Dept. of Commerce
Washington 25, D.C.

Chief, Bureau of Ships
Department of the Navy
Washington 25, D.C.
Att: Code 210L (3 copies)

Chief, Bureau of Ships
Department of the Navy
Washington 25, D.C.
Att: Code 420 (2 copies)

Chief, Bureau of Ships
Department of the Navy
Washington 25, D.C.
Att: Code 440 (3 copies)

Chief, Bureau of Ships
Department of the Navy
Washington 25, D.C.
Att: Code 430 (3 copies)

Chief, Bureau of Ships
Department of the Navy
Washington 25, D.C.
Att: Code 341B

Chief, Bureau of Ships
Department of the Navy
Washington 25, D.C.
Att: 345

Chief, Bureau of Ships
Department of the Navy
Washington 25, D.C.
Att: 422

Chief, Bureau of Ships
Department of the Navy
Washington 25, D.C.
Att: 644 (2 copies)

Commanding Officer and Director
David Taylor Model Basin
Washington 7, D.C. (65 copies)

Commander
Defense Documentation Center
Cameron Station
Alexandria, Virginia (20 copies)

Commander
Air Research and Development Command
Att: Mechanics Branch
Air Force Office of Scientific Research
14th and Constitution
Washington 25, .C.

Commander
Wright Air Development Division
Aircraft Laboratory
Att: Mr. W. Mykytow, Dynamics Branch
Wright-Patterson Air Force Base, Ohio

Commanding Officer
Office of Naval Research Branch Office
495 Summer Street
Boston, Massachusetts 02110

Commanding Officer
Office of Naval Research Branch Office
207 West 24th Street
New York, New York 10011

Commanding Officer
Office of Naval Research Branch Office
230 North Mighigan Avenue
Chicago, Illinois 60601

Commanding Officer
Office of Naval Research Branch Office
1000 Geary Street
San Francisco, California 94109

Commanding Officer
Office of Naval Research Branch Office
1030 East Green Street
Pasadena, California 91101

Commanding Officer
Office of Naval Research
Box 39, Navy #100, Fleet Post Office
New York, New York (5 copies)

UNIVERSITIES

Guggenheim Aeronautical Laboratory
California Institute of Technology
Pasadena 4, California
Att: Dr. Clark B. Millikan

California Institute of Technology
Pasadena 4, California
Att: Professor T. Y. Wu

California Institute of Technology
Pasadena, California
Att: Dr. M. S. Plesset

California Institute of Technology
Pasadena, California
Att: Dr. A. J. Acosta

California State Polytechnic College
Aeronautical Engineering
San Luis Obispo, California
Att: Professor Lester W. Gustafson

University of California
209 Mechanical Engineering Building
Berkeley 4, California
Att: Professor Edmund V. Laitone

University of California
Dept. of Engineering, Institute of
Engineering Research
Berkeley 4, California
Att: Dr. J. V. Wehausen

Case Institute of Technology
Cleveland, Ohio
Att: Dr. G. Kuerti

Cornell University
Graduate School of Aeronautical
Engineering
Ithaca, New York
Att: Dr. W. R. Sears

Cornell Aeronautical Laboratory
4455 Genesee Street
Buffalo, New York
Att: Mr. W. Targoff

Cornell Aeronautical Laboratory
4455 Genesee Street
Buffalo, New York
Att: Mr. R. White

University of Florida
Department of Aeronautical Engineering
Gainesville, Florida
Att: Professor William H. Miller

University of Florida
Department of Aeronautical Engineering
Gainesville, Florida
Att: Professor D. T. Williams

Harvard University
Dept. of Applied Physics and
Engineering Science
Cambridge 38, Massachusetts
Att: Dr. H. W. Emmons

Harvard University
Dept. of Applied Physics and
Engineering Science
Cambridge 38, Massachusetts
Att: Professor Carrier

Harvard University
Dept. of Applied Physics and
Engineering Science
Cambridge 38, Massachusetts
Att: Professor Goldstein

University of Illinois
Dept. of Aeronautical Engineering
Urbana, Illinois
Att: Dr. Allen I. Ormsbee

Professor J. M. Robertson
University of Illinois
Dept. of Theoretical and Applied
Mechanics
212 Talbot Laboratory
Urbana, Illinois

John Hopkins University
Department of Aeronautics
Baltimore 18, Maryland
Att: Dr. F. H. Clauser

Johns Hopkins University
Applied Physics Laboratory
P. O. Box 244-Rt. 1
Laurel, Maryland
Att: Technical Reports Office

Johns Hopkins University
Applied Physics Laboratory
8621 Georgia Avenue
Silver Springs, Maryland
Att: Dr. F. N. Frenkiel

Massachusetts Institute of Technology
Fluid Dynamics Research Laboratory
Cambridge 39, Massachusetts
Attn: Prof. H. Ashley

Massachusetts Institute of Technology
Fluid Dynamics Research Laboratory
Cambridge 39, Massachusetts
Attn: Prof. M. Landahl

Massachusetts Institute of Technology
Fluid Dynamics Research Laboratory
Cambridge 39, Massachusetts
Attn: Prof. J. Dugundji

Massachusetts Institute of Technology
Fluid Dynamics Research Laboratory
Cambridge 39, Massachusetts
Attn: Prof. A. Shapiro

University of Michigan
Dept. of Aeronautical Engineering
Ann Arbor, Michigan
Att: Professor Arnold Kuethe

University of Michigan
College of Engineering
Ann Arbor, Michigan
Att: Dr. Arthur G. Hansen

University of Michigan
College of Engineering
Dept. of Engineering Mechanics
Ann Arbor, Michigan
Att: Professor C. S. Yih

University of Minnesota
Dept. of Aeronautical Engineering
Rosemount Aeronautical Laboratories
Rosemount, Minnesota

Guggenheim School of Aeronautics
New York University
University Heights
New York 53, New York
Att: Librarian

North Carolina State College
Head of Mechanical Engineering Dept.
Raleigh, North Carolina
Att: Dr. R. W. Truitt

Ohio State University
Research Foundation
Columbus 10, Ohio
Att: Library

Pennsylvania State University
University Park, Pennsylvania
Att: Professor George F. Wislicenus

Purdue University
School of Aeronautical Engineering
Lafayette, Indiana
Att: Library

University of Southern California
Engineering Center
935 West 37th Street
Los Angeles 7, California
Att: Dr. Raymond Chuan

Stanford University
Department of Mathematics
Stanford, California
Att: Dr. B. Perry

Stanford University
Department of Mathematics
Stanford, California
Att: Dr. E. Y. Hsu

Stanford University
Department of Mathematics
Stanford, California
Att: Prof. I. Flugge-Lotz

Stevens Institute of Technology
5th and Hudson Street
Hoboken, New Jersey
Att: Dr. John P. Breslin

Stevens Institute of Technology
Hoboken, New Jersey
Att: Director, Davidson Laboratory

Stevens Institute of Technology
Davidson Laboratory
Hoboken, New Jersey
Att: Mr. C. J. Henry

Stevens Institute of Technology
Davidson Laboratory
Hoboken, New Jersey
Att: Mr. S. Tsakonas

University of Texas
Defense Research Laboratory
P. O. Box 8029
Austin, Texas
Att: Professor M. J. Thompson

University of Washington
Dept of Aeronautical Engineering
Seattle 5, Washington
Att: Professor R. E. Street

State University of Iowa
Iowa Institute of Hydraulic Research
Iowa City, Iowa
Att: Prof. L. Landweber

State University of Iowa
Iowa Institute of Hydraulic Research
Iowa City, Iowa
Att: Prof. Hunter Rouse

University of California
Dept of Naval Architecture
Berkeley, California
Att: Prof. H. A. Schade, Head

St. Anthony F lls Hydraulic Laboratory
University of Minnesota
Minneapolic, Minn.
Att: Prof. B. Silberman

St. Anthony Falls Hydraulic Laboratory
University of Minnesota
Minneapolis, Minn.
Att: Mr. J. N. Wetzel

Ordnance Research Laboratory
Pennsylvania State University
University Park, Penn.
Att: Dr. M. Sevik

Virginia Polytechnic Institute
Aerospace Engineering Dept.
Blacksburg, Virginia
Att: Dr. H. A. Hassan

INDUSTRIAL AND RESEARCH COMPANIES

Midwest Research Institite
425 Volker Blvd.
Kansas City 10, Missouri
Att: Mr. Zeydel

Director, Dept. of Mechanical Sciences
Southwest Research Institute
8500 Culebra Road
San Antonio 6, Texas

James Forrestal Research Center
Dept. of Aeronautical Engineering
Princeton, New Jersey
Att: Professor C. D. Perkins, Chairman

Department of Mechanical Sciences
Southwest Research Institute
8500 Culebra Road
San Antonio 6, Texas
Att: Dr. H. N. Abramson

Department of Mechanical Sciences
Southwest Research Institute
8500 Culebra Road
San Antonio 6, Texas
Att: Mr. G. Ransleben

Department of Mechanical Sciences
Southwest Research Institute
8500 Culebra Road
San Antonio 6, Texas
Att: Editor, Applied Mechanics Review

Boeing Airplane Co., Seattle Division
P. O. Box 3107
Seattle 14, Washington
Att: Mr. G. Schairer

Boeing Airplane Co., Seattle Division
P. O. Box 3107
Seattle 14, Washington
Att: Mr. M. J. Turner

Convair
P. O. Box 1950
San Diego 12, California
Att: Mr. A. D. MacLellan, Systems Dynamic Group

Convair
P. O. Box 1950
San Diego 12, California
Att: Mr. H. T. Brooke, Hydrodynamics Group

Electric Boat Division
General Dynamics Corp.
Groton, Conn.
Att: Mr. Robert McCandliss

General Applied Sciences Laboratories, Inc.
Merrick and Stewart Avenues
Westbury, Long Island, New York
Att: Dr. F. Lane

Gibbs and Cox, Inc.
21 West Street
New York, New York

Grumman Aircraft Engineering Corp.
Bethpage, Long Island, New York
Att: Mr. E. Baird

Grumman Aircraft Engineering Corp.
Bethpage, Long Island, New York
Att: Mr. E. Bower

Grumman Aircraft Engineering Corp.
Bethpage, Long Island, New York
Att: Mr. M. C. Tilgner

Grumman Aircraft Engineering Corp.
Dynamic Developments Division
Babylon, New York

President, Hydronautics, Inc.
Pindell School Road
Laurel, Howard County, Maryland
(2 copies)

Lockheed Aircraft Corp.
Missiles and Space Division
Palo Alto, California
Att: R. W. Kermeen

Lockheed Aircraft Corp.
Missiles and Space Division
Palo Alto, California
Att: L. A. Rodert

Technical Research Group, Inc.
Route 110
Melville, Long Island, New York
Att: Dr. Jack Kotik (2 copies)

The Rand Corporation
1700 Main Street
Santa Monica, California
Att: Dr. B. Parkin

The Rand Corporation
1700 Main Street
Santa Monica, California
Att: D. Morriss

Armour Research Foundation
10 West 35th Street
Chicago 16, Illinois
Att: Document Librarian

Avco-Everett Research Laboratory
2385 Revere Beach Parkway
Everett 49, Massachusetts
Att: Dr. Richard H. Levy

Avco Manufacturing Corp.
2385 Revere Beach Parkway
Everett 49, Massachusetts
Att: Dr. A. Kantrowitz

AVCO Research Library
2385 Revere Beach Parkway
Everett 49, Massachusetts

Aeronutronics Systems, Inc.
1234 Air Way
Glendale, California
Att: Dr. L. Kavanau

Chance-Vought Corporation
Dallas, Texas
Att: W. C. Schoolfield

General Dynamics Corporation
Electric Boat Division
Groton, Connecticut
Att: Frank Walker Caldwell

General Electric Company
2900 Campbell Avenue
Schenectady 6, New York
Att: Library

General Electric Company
Research Laboratory
Schenectady, New York
Att: Dr. H. T. Nagamatsu

Grumman Aircraft Engineering Corp.
Bethpage, Long Island, New York
Att: Fluid Mechanics Section

Hughes Aircraft Company
Florence Ave. at Teal Street
Culver City, California
Att: Dr. A. E. Puckett

United Aircraft Corporation
East Hartford, Connecticut
Att: Mr. J. G. Lee

McDonnell Aircraft Corporation
St. Louis, Missouri
Att: Library

Lockheed Missile Systems Division
Research and Development Laboratory
Sunnyvale, California
Att: Dr. W. Griffith

Norair
Division of Northrop Aircraft, Inc.
Hawthorne, California
Att: Dr. W. Pfenninger

Mr. K. E. Van Every
Norair, Division of Northrop Aircraft, Inc.
Hawthorne, California

Sulzer Bros., Inc.
19 Rector Street
New York 6, New York

North American Aviation, Inc.
4300 East 5th Avenue
Columbus, Ohio
Att: Mr. R. M. Crone

North American Aviation, Inc.
Aerophysics Department
12214 Lakewood Boulevard
Downey, California
Att: Dr. E. R. van Driest

Republic Aviation Corporation
Conklin Street
Farmingdale, Long Island, New York
Att: Dr. W. J. O'Donnell

Therm Incorporated
Aerophysics Section
Ithaca, New York
Att: D. E. Ordway, Head

MISCELLANEOUS

Pacific Aeronautical Library
Institute of the Aeronautical
Sciences
7660 Beverly Boulevard
Los Angeles 36, California

Applied Mechanics Review
Southwest Research Institute
8500 Culebra Road
San Antonio, Texas

Society of Naval Architects and
Marine Engineers
74 Trinity Place
New York, New York

Von K. Jacob
Computing Center
California Institute of Technology
1201 East California Street
Pasadena 3, California

11