N72-24991

NASA CONTRACTOR REPORT

NASA CR-2016

NASA CR-2016

FLIGHT TESTING TECHNIQUES FOR THE EVALUATION OF LIGHT AIRCRAFT STABILITY DERIVATIVES

A Review and Analysis

by Frederick O. Smetana, Delbert C. Summey, and W. Donald Johnson

Prepared by
NORTH CAROLINA STATE UNIVERSITY
Raleigh, N. C. 27607
for Langley Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • MAY 1972

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-2016		
4. Title and Subtitle FLIGHT TESTING TECHNIQUES FOR THE EVALUATION OF LIGHT AIRCRAFT STABILITY DERIVATIVES - A REVIEW AND ANALYSIS		5. Report Date May 1972 6. Performing Organization Code
7. Author(s) Frederick O. Smetana, Delbert C. Summey, and W. Donald Johnson		8. Performing Organization Report No.
9. Performing Organization Name and Address North Carolina State University Raleigh, N.C. 27607		10. Work Unit No. 736-05-10-01 11. Contract or Grant No. NAS1-9603
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		13. Type of Report and Period Covered Contractor Report 14. Sponsoring Agency Code
15. Supplementary Notes		

16. Abstract

Techniques quoted in the literature for the extraction of stability derivative information from flight test records are reviewed. A recent technique developed at NASA's Langley Research Center was regarded as the most productive yet developed. Results of tests of the sensitivity of this procedure to various types of data noise and to the accuracy of the estimated values of the derivatives are reported. Computer programs for providing these initial estimates are given. The literature review also includes a discussion of flight test measuring techniques, instrumentation, and piloting techniques.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Stability derivative extraction Stability derivatives Computed program	Unclassified - Unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	112	$3.00

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

TABLE OF CONTENTS

GENERAL INTRODUCTION

A previous study (ref. 1) considered means for predicting the influence of configuration changes on the riding and handling qualities of a light aircraft. A reader of that study will note that the values of many significant parameters cannot be estimated with great precision despite the fact that the entire analysis assumes only very small perturbations from equilibrium. Flight testing is therefore necessary to establish the validity of the analysis and to determine the riding and handling qualities for larger excursions from equilibrium. Conceivably, flight testing could also prove useful in developing improved parameter prediction techniques by helping to establish the correct parameter values for a given configuration. To serve this latter function, however, the flight data must be taken as accurately as possible then interpreted consistently and correctly.

Flight testing is here regarded by the authors as the terminal portion of the complete riding and handling qualities design task. For this reason the present work was developed as a supplement to the previous study. To serve this function, the work employs a similar approach and may therefore be somewhat more analysis-oriented than is usual in discussions of stability and control flight testing. This emphasis, however, seems consistent with the finding that the parameter extraction procedure used to operate on accurately measured data is far more significant than the manner in which the pilot performs the test.

Consistent with the plan of the previous study, the present work gives a review of methods found in the literature for extracting both static stability derivatives and dynamic stability derivatives from flight data. No discussion is presented of methods for establishing the compliance with the FAR's or Military Specifications on light aircraft handling. These requirements are discussed in the previous study in terms of suitable values for the appropriate stability derivatives.

Following this portion of the review, there is a brief discussion of the instrumentation and instrument installation techniques needed to procure the data from which the stability derivatives can be extracted. The piloting procedures found to assist the data reduction are also indicated.

The next section treats in some detail a very sophisticated method for extracting stability derivatives from flight data which has been under development by the National Aeronautics and Space Administration for a number of years. This method takes advantage of modern computer technology to obtain a high degree of precision at reasonable cost. The technique includes a provision for removing certain types of noise from the signals. It is specially adapted here for use with light aircraft.

Finally, it may be noted that the previous study provided computer programs for calculating the frequency and damping ratio of an aircraft's oscillatory modes and the time constants of its aperiodic modes if one has the values of the stability derivatives in the transfer function. Another program for calculating the time histories of the various motions, given the frequencies, damping ratios, time constants and transfer function numerators, was also provided. For the present study, the computerization has been extended to the calculation of predicted values for the stability derivatives given the aircraft geometry. The methods upon which the computer programs are based were detailed in the previous study.

In addition to making the prediction of light aircraft motions now entirely a mechanical task, the new programs simplify the task of extracting the stability derivatives from flight motions. The NASA procedure mentioned above seems to be locally convergent. Thus, initial values for the stability derivatives not too far from their correct values are necessary to insure convergence. These are provided by the new programs. They are described in detail in Appendix B.

The reader will observe that the mathematical basis of the method recommended for extraction of stability derivatives from measurements of aircraft flight motions is not elementary. This is perhaps unfortunate because this study is intended for use by engineers whose preparation may not have included instruction in relatively complex numerical computation procedures or the mathematical theory associated with fitting a set of equations with undetermined coefficients to experimental data, particularly where the number of undetermined coefficients exceeds the number of independent equations. Two factors led the authors to persist in this choice despite the obvious obstacle. The first was their conviction that even with the best methods and instrumentation available it is difficult to extract derivative values that are accurate and reliable. Inferior data and data reduction techniques are often not worth the effort expended since the results obtained with them usually fail to offer a reliable standard against which to compare theoretical predictions. Anything less than a high level effort is probably best left undone.

The second reason the recommended technique was pursued was that it has been so programmed that little mathematical sophistication is required to <u>use</u> it. Some consideration of the physics involved, however, is needed to obtain reliable results. The user must appreciate the fact that a maneuver which does not excite a particular motion strongly is not very suitable for extracting derivative values associated with that motion. For example, an aileron pulse is less useful for finding such derivatives as L_r, N_r, and N_β than is a rudder pulse. Also, a short flight record is inappropriate for extracting the derivatives which are dominant in the phugoid mode. Once these factors are recognized along with the deleterious influence of noise, phase shifts, lack of resolution, and error in the flight records it becomes a fairly mechanical procedure to extract reliable values of the stability derivatives. The necessary steps are related in detail. For someone interested in examining the rigor of the procedure, sufficient detail is provided along with pertinent references so that he can reach a judgment on this point.

The reader will also note that most of the flight criteria usually taken to be indicative of light aircraft handling qualities are not discussed at all. The values of parameters such as the variation of stick force with speed, while important in helping a pilot evaluate the handling characteristics of an aircraft offer little opportunity to extract information on the precise influence of geometric or inertial changes since equivalent expressions involve a combination of several of the usual stability derivatives. Because the ultimate purpose of the present work is to improve the ease and accuracy with which the light aircraft design process is carried out, it was felt that only those procedures which offered a resonable prospect of serving this purpose effectively should be discussed at this time. It is hoped that the following review and analysis is consistent with this aim.

LITERATURE REVIEW —

TECHNIQUES FOR EXTRACTING STABILITY DERIVATIVES FROM FLIGHT TEST DATA

INTRODUCTION

Interest in obtaining values of the stability derivatives by testing the aircraft in actual flight was evident as early as 1925. By that time both the Aeronautical Research Committee and the National Committee for Aeronautics had begun to conduct such programs. Because riding and handling quality requirements are constantly growing more stringent and because the costs of conducting extensive development programs are steadily rising, there has been a mounting desire to improve the accuracy of analytical predictions of the motion of an aircraft in response to a given control input. Once the stability and aerodynamic parameters are extracted from actual flight data, the parameter prediction, such as described in reference 2, can then either be verified or modified to give estimates closer to the flight values.

The major difficulty in finding the longitudinal (or lateral) stability derivatives is that only three differential equations must be solved for ten or twelve unknowns; this problem exists even after the equations have been linearized.* In addition there is the problem of noise in the flight data. Stability derivatives can never be any more accurate than the flight records from which they are taken. Thus, improvements in the technique used to solve for the stability derivatives become important only after good flight records have been secured.

In the last thirty years, many techniques to extract these derivatives have been employed, sometimes with excellent results. The general reliability of the techniques, however, could never be convincingly demonstrated. With the advent of transonic and supersonic aircraft and missiles and their stability and control problems coupled with the difficulty of wind tunnel measurement in these flow regimes, the availability of stability derivatives derived from flight data assumed a heightened importance. Fortunately, the concurrent development of modern high speed computers has made possible the development of more rigorous techniques which heretofore were prohibitively complex because of the many involved mathematical computations required.

The literature review which follows discusses some of the more important of these developments. Although several important references may have been omitted, it is felt that a sufficient number have been included to insure thoroughness.

---

* The linearized equations of longitudinal and lateral motion with the dimensional stability derivatives are given on pages 53 and 39 respectively. The derivation of these equations, as well as the appropriate transfer functions, can be found in reference 1.

EARLY METHODS

Prior to the 1940's dynamic stability tests were concerned chiefly with determining the damping and frequency of aircraft oscillation. NACA Report 442 (ref. 3), a 1932 study by Soule and Wheatley, compares the theoretical and measured longitudinal stability characteristics of an airplane. The linearized longitudinal equations of motion were used to obtain the longitudinal characteristic equation in terms of the dimensional stability derivatives. Approximate factorization of the biquadratic* was used to obtain two quadratic equations: one of the quadratic equations represented a short-period, heavily-damped oscillation; the second, a long-period, lightly-damped oscillation. Soule and Wheatley argued that it is with the lightly-damped oscillation that instability is most likely to occur; therefore, it is usually necessary to investigate only this phase of the motion. From the long-period or phugoid quadratic, equations could be given for both the period and damping coefficient of this mode in terms of the dimensional stability derivatives. Using theoretical formulas to estimate the dimensional stability derivatives, the theoretical damping coefficient and period were obtained. Flight tests of a Doyle 0-2 airplane were then made to measure the period and damping experimentally. The period and damping coefficient were determined by direct measurements of the oscillation characteristics of u, w, and θ both for power-off and power-on conditions. The authors (ref. 3) decided that since u, w, and θ are interdependent variables, the periods of their variations with time are necessarily the same, although they may not be in phase; thus, the period and damping can be determined by studying the behavior of only one variable. Airspeed was chosen as the one most convenient for study. The period of oscillation was found by measuring the time interval between two consecutive peaks of a time history of velocity. The damping coefficient was approximated by estimating the decrease in velocity at two consecutive time history peaks. Based on a comparison between the experimental values of both the damping coefficient and the period of oscillation it was concluded that the theory of longitudinal stability based on the assumption of small oscillations gives satisfactory results for practical studies of longitudinal stability.

In 1950 a survey of methods for determining stability parameters from dynamic flight measurement was conducted (ref. 4). Most of the methods were concerned with determining transfer function coefficients which are certain combinations of stability derivatives. The transfer functions investigated were derived from the linearized longitudinal pitching velocity and normal acceleration equations. Experimentally-determined transfer functions were compared with analytical models. A least squares procedure was applied to

---

* the fourth order characteristic equation of motion formed by expansion of the denominator determinant common to all the longitudinal transfer functions

obtain those values of the coefficients which cause the analytical model to match the data most closely. The ratio of coefficient error to that of the basic data was also obtained in this fashion. A number of methods for obtaining good first approximations to the coefficients, a step which aids the convergence of the least square procedure, were discussed.

A more in-depth literature survey on dynamic stability and control research was made in 1951 by Cornell Aeronautical Laboratory (ref. 5), with the following overall appraisal:

> While it is by no means easy to evaluate overall progress in the field of dynamic response measurements, the following may be stated: Steady state oscillation and transient input flight methods are now available which yield equivalent and repeatable response data. Methods for reduction to the derivatives have been demonstrated for both the longitudinal and lateral cases. For the airplanes tested close agreement in all cases has not been obtained between the measured responses and best estimates based on static high speed wind tunnel data and theory. Though there is no conclusive evidence as to where the differences may lie, there is a growing belief in the validity of the flight measurements. Because of the additional processing involved in extracting the derivatives from the measured response data, it would be expected, and has generally been found true, that larger discrepancies may exist between the measured and estimated derivatives. While there is every reason to believe that the approach is basically sound, and generally applicable to projected performance ranges and design parameters, further experience and refinement are desirable.

As Cornell's survey notes, considerable interest was shown in the late 1940's in forced oscillation tests. In this procedure the elevator is oscillated steadily at some frequency and amplitude. After the aircraft's motion had become steady, the amplitude of the normal acceleration response and its phase relationship to the forcing function is measured. The elevator excitation is then changed to a new frequency and amplitude and the response measured. Angle of attack and pitch angle responses can also be determined at the same time. From these tests functional relationships such as the variation of normal acceleration amplitude and phase angle per unit elevator deflection with frequency of oscillation can be determined. This type of response presentation, or transfer function, is commonly used in describing the dynamic properties of mechanical or electrical systems, and is directly useful in the synthesis (and stability determination) of a complete system in which the airplane is a component. This was an attempt to rely heavily upon the mathematical and experimental techniques of electrical engineers, which provided a well-developed basis for the handling of dynamic phenomena. The flight time needed to measure a complete frequency response, however, is excessive at least by modern standards. For this reason Fourier transform techniques were developed to determine the harmonic content of transient responses. The amplitude and phase of the constituent sine waves in a normal acceleration time history are compared to the amplitude and phase of the constituent sine waves in the elevator deflection time history to form the transfer function as before. The entire transfer function, or frequency response, however, can be obtained from the response to a single elevator pulse. This is a substantial saving in flight time achieved at the cost of a substantial increase in the complexity of data reduction.

To insure a minimum change in airspeed during flight testing, a double pulse* was recommended. Excellent agreement is possible between information taken on different flights and days if sound instrumentation, carefully calibrated, is used. The author (ref. 5) felt that static, full-scale wind tunnel tests would be invaluable for checking derivatives from a dynamic test. Actual flight tests at a Mach number of 0.7 were conducted, showing good results in extracting stability derivatives.

In 1951 Shinbrot (ref. 6) gave a method for the calculation of stability coefficients, which are made up of stability derivatives, from transient response data. The calculation of these coefficients of the linear differential equations of motion was based on the classical least-squares curve-fitting method.** The method is quite cumbersome to use for some inputs. The initial approximations to the coefficients were obtained by a method (ref. 7) requiring graphical differentiation of time histories, which, in some cases, may cause large errors.

The next year, Shinbrot delineated several methods for curve-fitting a set of data by least squares in his investigation of curve-fitting techniques (ref. 8). The pitching velocity was described as a sum of exponentials with complex exponents. The coefficients of the exponentials, as well as their exponents, were combinations of the stability derivatives of the airplane; least squares was then applied to obtain coefficient values, and an analytical relation between the coefficients and the derivatives was used to evaluate the derivatives or combinations of derivatives. Nine months later (ref. 9) he discussed some of the errors encountered using least squares and other curve-fitting techniques. In this report he warns against using only the pitching velocity excited by

---

* a rapid motion of the elevator first in one direction and then in the other

** Usually one wishes to determine several unknown parameters from only 3 equations of motion. By evaluating the three equations at a number of times, one can form many equations in the unknown parameters. For example, suppose $\dot{u} = Au + Bw + Cq$ and time histories are available for $\dot{u}$, u, w, and q. Then many equations in A, B, and C can be formed by evaluating the above equation at many different points in time thus giving more equations than unknowns. A least squares procedure can then readily be used to find the best values for A, B, and C which satisfy the data set.

an elevator pulse to calculate all the derivatives in the pitching velocity equation. No more should be expected from such an analysis than the period and damping parameters.

In 1954 Shinbrot (ref. 10) developed a general theory of the so-called "equations-of-motion" methods* for the analysis of linear dynamical systems and then extended it to apply to non-linear systems. A variation of the "Fourier transform" method for analysis of linear systems was combined with the non-linear methods to produce an improved technique for obtaining stability derivatives of both linear and non-linear systems. As the report notes, one important advantage of the new method is that the dependency on initial values of the derivatives, found in earlier methods, is entirely eliminated. This advantage is of particular importance when systems of higher order than the second are considered.

Twelve years after Shinbrot's method was published in TN 3288, Burns (ref. 11) wrote of his experience with it in estimating stability derivatives. Based on the flight testing of two aircraft, it was found that reliable results were obtained only when the unknowns in each longitudinal or lateral equation were reduced to two. Burns gave three recommendations which he felt might be helpful in future flight test programs: 1) to evaluate control derivatives, the initial control input should be sufficiently rapid for the effect of disturbances within the duration of the control input to be relatively small; 2) to evaluate damping derivatives, one cycle of the motion is sufficient; and 3) to evaluate normal force derivatives, the acceleration equation should be used.

In 1951 Donegan and Pearson presented what was termed a matrix method for determining the longitudinal stability coefficients of an airplane.** They first integrated the linearized longitudinal equations of motion so that no derivative terms remained. They then integrated measured values of the angle of attack, pitch angle, normal acceleration, and control surface deflection angle numerically for different values of time. By substituting measurements into the integral equations, a system of simultaneous equations in the unknown coefficients is created. These unknown coefficients can then be found by solving the simultaneous equations. Once the coefficients have been found, some of the stability derivatives which make up the coefficients may be approximated by making certain assumptions. An attractive feature of this method is that integrations tend to smooth out noise. A method is also given to obtain the frequency response of the airplane.

---

* Measured values of the dependent variables at various times are substituted into the differential equations, the general forms of which are assumed. For each time, 3 equations in the unknown parameters are generated. The process is continued until sufficient equations are available to overspecify the unknown parameters.

** first given in NACA TN-2370 which was later superseded by NACA TR-1070 (ref. 12)

In 1954 Donegan followed the theme of the work in TR 1070 by presenting three matrix methods for determining the longitudinal stability derivatives from transient flight data (ref. 13). The methods differ in complexity with the most general method requiring four measurements in time history form and the least general method requiring only two time history measurements, together with the assumptions that $C_{m_{\dot{\alpha}}}/C_{m_q}$ = constant and $C_{m_{\delta_e}} = (\ell_t/c)C_{L_{\delta_e}}$. The results of these methods depended in large measure on accurate instrument measurements and required considerable computation to yield adequate engineering answers.

In 1955 Donegan *et al.* (ref. 14) obtained lateral stability derivatives by curve-fitting forced oscillation responses with a vector representation of the linearized lateral equations of motion. The dependent variables (β, ϕ, etc.) of the equations were considered to be vectors of amplitude ratio R and phase angle Φ. By equating real and imaginary parts, the three equations of lateral motion can be separated into six equations. These six equations are then fitted to the flight data by a least squares procedure. The coefficients evaluated by this means are combinations of the stability derivatives.

In the early to mid-1950's several other notable reports treated the extraction of stability derivatives. Reference 15 presents a method for deriving time-response and frequency-response data for angle of attack and normal accelerations at the c.g., when these data are measured at non-c.g. locations and pitching velocity is not measured. The method appears particularly applicable when insturments cannot be placed in the most desirable locations.

In 1954 Sternfield (ref. 16) presented a vector method approach to the analysis of the dynamic lateral stability of aircraft, making possible a physical visualization of the contribution of the various stability derivatives and mass characteristics to the overall motion of the airplane.

Eggleston and Mathews also presented TR 1204 (ref. 17) in 1954 evaluating some of the methods previously published for determining transfer functions and frequency response of aircraft from flight data. In general these methods may be classed as: 1) analysis of the frequency response resulting from a sinusoidal control surface input, 2) analysis of the frequency response by using Fourier transforms to convert the transient response to an arbitrary input into the frequency domain, and 3) analysis of the transient through the use of least-squares solutions of the coefficients of an assumed equation (curve-fitting methods). The investigation revealed that the curve-fitting methods (Donegan-Pearson and exponential-approximation methods) appear to be less critical to inputs having regions of low harmonic content than Fourier methods and present the frequency response as analytical transfer functions. Fourier methods indicate characteristics of frequency response that may be missed in curve-fitting methods because of the limitations on the assumed form of the equations. For manual calculations, the Donegan-Pearson method appears best suited for highly damped systems in response to arbitrary inputs, and the Fourier method offers comparable results but requires lengthy calculations.

Wilkie (ref. 18) presents a statistical extraction method for aircraft stability coefficients based on a maximum-liklihood parameter-estimation technique. In general no significant difference in extraction accuracy was observed between the integral method of Shinbrot, the statistical method, and the derivative method, provided the various data required by each method were available to equal precision.

An analysis of longitudinal response to unstable aircraft is given in reference 19. Methods of obtaining stability derivatives from flight records, as well as possible improvement for the methods are given. Mention is made of the fact that it may be advantageous to assume values of the least important derivatives and then calculate the others by least squares.

In 1959 another review of the activities in the field of aircraft dynamic stability derivatives was undertaken (ref. 20). While the author mentioned several common methods for analyzing full-scale flight data, the major portion of the report was concerned with a detailed discussion of the techniques used to obtain dynamic measurements in wind tunnels. Reference 21 is also concerned with dynamic longitudinal measurements in a wind tunnel. Procedures are given by which the longitudinal damping derivative may be obtained in the wind tunnel.

MODERN METHODS

A United States Air Force report by Rampy and Berry (ref. 22) treated the determination of stability derivatives from flight test data by operation analog matching. In conventional analog matching the aircraft equations of motion are programmed on an analog computer to provide a mathematical model of the aircraft against which to compare the motions recorded from an actual flight test. Theoretical or wind tunnel values of the stability derivatives are used as initial estimates of their flight values. These and other basic variables (airspeed, moments of inertia, etc.) are then "fed" into the computer as constants. The flight test inputs (*e.g.*, control surface deflections) are reproduced on function generating equipment for introduction into the mathematical model. The computer calculates responses to these inputs and records them on a strip chart or oscilloscope for comparison with actual flight test time histories. Differences between the computer and aircraft responses are attributed to errors in the estimated values of the stability derivatives. The values of the stability derivatives used in the computer are then changed using a trial and error process until the computed responses match the flight records. The stability derivative values producing this match are then noted.

The process is very time-consuming because it may be necessary to match a considerable number of time histories to obtain generally valid values and the cataloging of the various influences becomes difficult. It should also be mentioned that experience has indicated that small errors in initial conditions read from the flight test records affect the solution noticably. The effect of initial condition errors is to shift the amplitude or rotate a response time history rather than to change its general shape.

Although the technique is relatively simple and straight-forward, hours or days may be spent before a satisfactory match is obtained. The quality of the match depends on the experience of the operator and the "goodness" of the flight data. In an effort to reduce the large amount of time required for data reduction, hardware and analytical techniques which achieve high speed, repetitive operation of the analog computer, such as reported in reference 22, have been developed. These permit automatic application of initial conditions, introduction of forcing functions, and then computation of solutions for a predetermined time interval. At the end of this interval, the computer stops the solution, resets, applies the same initial conditions and forcing functions, and repeats the computation. The sequence rate is fast enough to make the solution appear as a stationary wave when displayed on the oscilloscope. A permanent record of the solution may be obtained by photographing the oscilloscope. The stability derivatives, represented by potentiometers, can be adjusted while the computer is operating in the high speed repetitive mode. Thus, a change in a stability derivative would cause a different solution to appear immediately on the oscilloscope. By scaling the flight test record to the size of the oscilloscope trace and graphing it on transparent material, one can readily determine when a satisfactory match has been obtained.

In 1966 Wolowicz produced an important work (ref. 23) in which he discussed various factors that influence the determination of stability and control derivatives and other behavior characteristics from flight data. Techniques are given for estimating both horizontal and vertical c.g. location as well as moments of inertia. Wolowicz points out that in flight testing the method of analysis selected governs the control input*, and the magnitude and duration of the input influences the magnitude of the perturbation. He points out that in well-performed pulse maneuvers with lightly damped oscillation, it is possible to determine a 2-second period to within 0.02 seconds. Good accuracy in damping can be obtained for damping ratios less than 0.2. The accuracy of the period and damping measurements becomes rather poor for damping ratios greater than about 0.3.

In considering a method for reducing flight data to the desired stability derivatives, Wolowicz indicates that many of the approximate expressions and the time-vector method depend upon control-fixed, free-oscillation data which are not usuable when damping is high; thus, data with high damping are usually investigated by a least squares technique or analog matching. Application of many of the simpler equations for determining derivatives requires an evaluation of the period and damping; whereas, application of the time-vector method requires, in addition, the determination of amplitude and phase relationships. These quantities are obtained from the free-oscillation portion of the pulse maneuver. The damping ratio, undamped natural frequency, and phase relationships can be obtained for both short period and phugoid free-oscillations by relations given in the text (ref. 23).

Wolowicz obtained good approximations for some of the longitudinal stability derivatives by keeping only the dominant terms when the equations of motion had been solved for a particular derivative. $C_{m_{\delta_e}}$ can be determined from the initial portion (approximately 0.2 seconds) of a rapid pulse maneuver by:

$$C_{m_{\delta_e}} = \frac{I_{yy}}{\frac{1}{2}\rho U^2 S c} \frac{\Delta \dot{q}}{\Delta \delta_e} \quad . \tag{1}$$

In a similar manner $C_{N_{\delta_e}}$ can be obtained by the relation:

$$C_{N_{\delta_e}} = \frac{W}{\frac{1}{2}\rho U^2 S} \frac{\Delta a_n}{\Delta \delta_e} \quad . \tag{2}$$

Once $C_{N_{\delta_e}}$ is known, $C_{L_{\delta_e}}$ can be approximated. The approximation for $C_{m_{\delta_e}}$ should result in no more than 5% error while $C_{L_{\delta_e}}$ should result in no more than 10% error. For both, accuracy is improved if the peak control input and acceleration response are used disregarding the phase lag between the two. It has been found that the time difference in peak values of control input and acceleration response is primarily the result of instrument phase lag.

---

* This is done primarily to take advantage of certain simplifications in the analysis provided by the use of special control inputs. More general methods are often independent of the type and quality of control input.

Analysis by this method requires instruments with flat response characteristics extending to relatively high frequencies (8 cycles/second).

An approximation is also given for evaluating C_{N_α} from the short-period free oscillation data of the airplane with control fixed:

$$C_{N_\alpha} \simeq C_L \left|\frac{\Delta a_n}{\Delta \alpha}\right| . \qquad (3)$$

In this expression the pitching-velocity and the angle-of-attack-rate term have been neglected in the short period form of the normal-force equation. For conventional, low performance aircraft, $C_{L_\alpha} \simeq C_{N_\alpha}$ for small values of α. An approximation is also given for $(C_{N_q} + C_{N_{\dot\alpha}})$, but it is quite difficult to evaluate. The static derivative C_{m_α} can be approximated to within 3% accuracy from the relation

$$C_{m_\alpha} \simeq -\frac{I_{yy}}{\frac{1}{2}\rho U^2 Sc}\, \omega^2_{n_{s.p.}} \quad . \qquad (4)$$

An equation is also given for the sum of C_{m_q} and $C_{m_{\dot\alpha}}$:

$$(C_{m_q} + C_{m_{\dot\alpha}}) = \frac{2I_{yy}}{mc^2}\left[C_{N_\alpha} - 4\tau\left(\frac{0.693}{(T_{\frac{1}{2}})_{s.p.}}\right)\right] , \qquad (5)$$

where τ = time parameter, $m/\rho US$. Separating the two derivatives with any accuracy is quite difficult; however the phugoid damping derivatives C_{D_u} and C_{L_u} can also be obtained by using formulas given in Wolowicz's report:

$$\begin{aligned} \frac{U\partial C_C}{\partial u} + \frac{2C_C}{\cos\alpha\cos\beta} &= \frac{4\zeta_{ph}\omega_{n_{ph}}}{\rho US} \\ \frac{U\partial C_N}{\partial u} + \frac{2C_N}{\cos\alpha\cos\beta} &= \frac{2\omega^2_{n_{ph}} m}{g\rho S} \quad . \end{aligned} \qquad (6)$$

Substitution of $C_N = C_L \cos\alpha + C_D \sin\alpha$ and $C_C = C_D \cos\alpha - C_L \sin\alpha$ into the above relations enables one to find C_{D_u} and C_{L_u}.

Wolowicz also gives some short approximations for the lateral stability derivatives. However, because of the more complex behavior of the airplane and the larger number of derivatives involved, the lateral-directional control and stability derivatives are not as readily and reliably determined by the use of approximate equations as are the longitudinal derivatives. Readers interested in lateral approximations should consult the report.

Along with outlining the approximations for both the longitudinal and lateral stability derivatives, Wolowicz also discussed the application of the analog-matching technique to flight data. It is indicated that when flight data preclude the successful use of the graphical time-vector technique or the approximate equations, and when time and expense will not permit the use of an experimentation with more sophisticated techniques, recourse is usually taken to the analog computer to determine the derivative values that provide the best match of the analog time history with the flight time history of a maneuver. Use of the analog computer should only be considered when other techniques cannot be applied. An example of the analog matching method was given for a high-performance aircraft in which the accuracies in determining the derivatives were based on the amount the derivatives could be changed before a trend toward mismatch became evident. For the longitudinal motion the accuracies for C_{N_α}, C_{m_α}, $C_{m_{\delta_e}}$, and (C_{m_q} + $C_{m_{\dot{\alpha}}}$) were found to be 10%, 5%, 10%, and 20% to 30% respectively, for a strong pull-up and release maneuver. Typical accuracies for the lateral stability derivatives based on well-conditioned releases from sideslip maneuvers were found to be 5%, 15%, 5% to 30%, and 5% to 15% for C_{n_β}, C_{ℓ_β}, C_{n_r}, and $C_{\ell_{\delta_a}}$ respectively.

In 1967 Rubin *et al.* (ref. 24) presented the steps necessary to derive the regression differential equation for a set of unknown parameters. The method was based on classical regression, that branch of statistics wherein relationships among a number of different stochastic variables are found. Classical regression consists of finding the coefficients or constants which minimize the error criterion, usually a squared function of the error. An example is given in which this method was employed to find the aerodynamic stability derivatives for the lateral motions of an airplane. Rubin *et al.* felt that the lack of connection between the papers on parameter identification and statistical regression analysis has led to much confusion among readers, if not the writers of these papers.

In the same year a Canadian report by Howard (ref. 25) presented a refined version of the equations of motion technique to determine the lateral stability and control derivatives of a STOL aircraft. This refined technique incorporates an allowance for unknown constant errors (inertia errors) in the measured quantities. It was believed that this allowance made a significant contribution to the overall accuracy of the method.

In 1969 three reports were published which may be valuable when discussing techniques for reducing flight test data. Reference 26 by Clinkenbeard *et al.* deals with the instrumentation necessary for extracting stability derivatives from V/STOL aircraft with a discussion of a possible method to analyze flight data. Analog matching and curve-fitting the equations of motion by least squares appeared to be the only techniques which would permit analysis of the non-linear equations. Since the analog matching technique is cumbersome, time consuming, and requires sound engineering judgment, the least squares technique was judged to be the more valuable for reducing the flight data.

A differential correction method for the identification of airplane parameters is given in reference 27. The method employs an iteration procedure and can be applied to both linear and non-linear differential equations.

> The differential correction method uses a criterion function that is quadratic in the difference between the measurement vector and the model output vector, and it is minimized to obtain the parameter estimates in the following way. The model output is expanded in a Taylor series for model parameter perturbations about an estimate of the parameter vector. Only first-order terms of the Taylor series expansion are retained.

The Taylor series expansion is used to obtain an approximate expression for the perturbations of the criterion function due to plant parameter perturbations. This approximate expression is minimized to obtain a correction to the estimate of the parameters. The technique used to compute the model output perturbations (that is, the sensitivity vectors) is described in this paper and is believed to be new. This technique improves the speed and accuracy of the differential correction method.

The differential correction method is not guaranteed to converge, but this fact did not cause a serious problem in samples tested thus far. Example plots comparing measured data to fitted data are given in the report, with good results indicated.

The third report by Larson and Fleck (ref. 28) describes the method of quasilinearization--a combination of high-speed digital computer capabilities with established linearization techniques resulting in a new method of identifying parameters. The method is essentially an efficient means for evaluating parameters existing in a set of algebraic or differential equations. The procedure is iterative in that the unknown parameters are estimated initially and then corrected until an error function is minimized. Larson and Fleck feel the mathematical concepts are well-known, but the combination of these mathematical concepts with the high-speed digital computer yields new and useful results.

An excellent comparison of methods for determining stability derivatives from flight data is given in a paper (ref. 29) published in 1969 by Taylor *et al.* The purpose of the paper was to compare a modified Newton-Raphson method* developed by Taylor and Iliff (ref. 30) with existing methods. The Newton-Raphson technique was developed to enable the use of a priori information and to automatically adjust bias terms and initial conditions to compensate for errors. The method converges rapidly to minimize the weighted mean square fit error. The a priori information may be based on wind tunnel data or upon previously analyzed flight data. The a priori values are also weighted so that, for a weighting of zero, the a priori values are ignored, and, for an infinite weighting, the flight data are ignored. The Newton-Raphson method resembles the procedure often followed in analog matching in which, initially, the wind-tunnel values are used and changes made to improve the fit are weighted against the departure from the wind-tunnel values. The method has been computerized and a detailed description of the program is scheduled to be released in late 1971 or early 1972 as a NASA Technical Note (ref. 31).

The attractiveness of this curve fit procedure is enhanced by a recitation of the limitation of some of the other methods. Taylor *et al.* felt that Wolowicz's approximate formulas had many disadvantages, *e.g.*,

---

* See page 37 for a more detailed discussion.

only some of the primary unknown coefficients of stability and control derivatives can be determined. In addition the forms of response that can be analyzed are very restrictive, *i.e.*, effects of controls must be either dominant or negligible. The analog matching technique is not recommended because the skill and technique of the operator is a factor in the resulting estimates. Although the regression methods of least squares and of Shinbrot involve no manual operation as does analog matching, nor are they limited in the coefficients that can be obtained, experience has indicated the variance of estimated coefficients to be excessive. Time histories from reference 29 which indicate the results of applying each of these methods to the problem of solving for the lateral stability derivatives have been reproduced in the present work (fig. 1 thru 7). These plots, which even include wind-tunnel data, are useful when discussing the accuracy of the derivatives obtained from the flight data. The Newton-Raphson (very similar to quasilinearization) method gave a resulting fit of the flight data which was superior to that of the least squares, Shinbrot, and analog matching methods. The Newton-Raphson method was employed to solve the problem of poor convergence which may occur when there are several unknowns. One important advantage of this method compared with the least squares method is that it is not necessary that all components of the state variables and their time derivatives be measured. The method has already been successfully applied to the problem of finding both lateral and longitudinal stability derivatives of airplanes such as the XB-70.

Even with the good results obtained from the Newton-Raphson, Taylor *et al.* are quick to point out that raw flight data must still be screened and edited manually before any method of obtaining stability derivatives is applied.

A very recent paper by Chapman and Kirk (ref. 32) discusses still another improved least squares method of matching analytical solutions to flight records. In this approach the error function to be minimized contains corrections to the calculated values of the dependent variable in addition to the usual difference between the measured and calculated values. The corrections are the first term in a Taylor series expansion of the dependent variable in terms of the unknown coefficients of the differential equation. The error-with-correction is squared and the sum of these squares, taken at a number of points in time, is minimized. Evaluation of the partial derivatives in the correction terms is by the method of parametric differentiation, a short description of which is given in the paper. The authors report rapid convergence to acceptable values in the cases evaluated thus far, four of which are reported in the paper. A particular point is made in the paper that "if the starting solution does not roughly describe the experimental data, divergence of the solution most often occurs." The procedure suggested for obtaining a starting solution is to integrate the differential equation a sufficient number of times to remove the highest order derivative, that is, to change it into an integral equation. The integrals are then evaluated numerically from the experimental data. By varying the interval of integration a set of equations can be obtained from which the values of the unknown coefficients are extracted by the method of least squares.

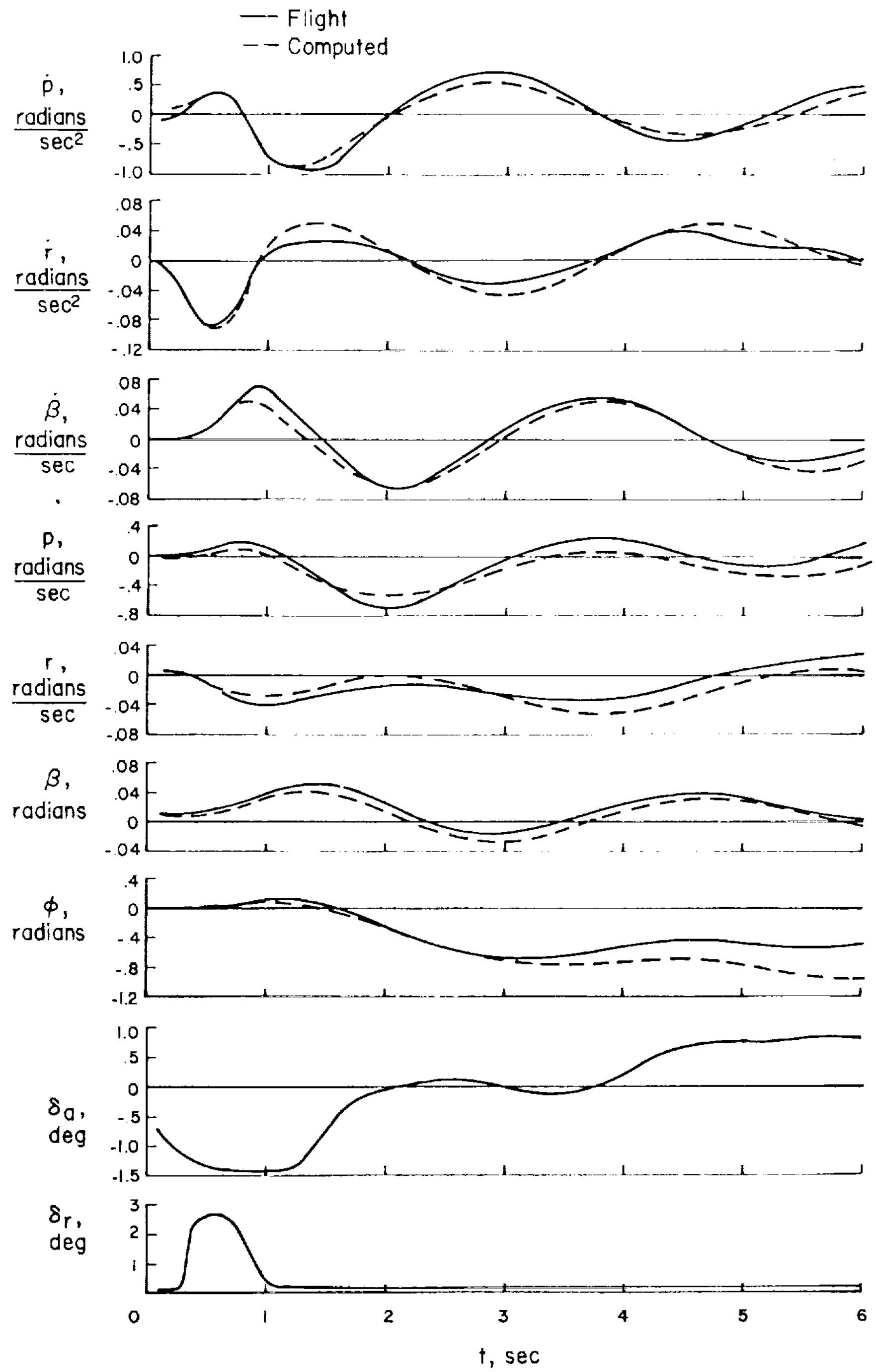

Figure 1. Comparison of time histories measured in flight and computed by using wind-tunnel coefficients.

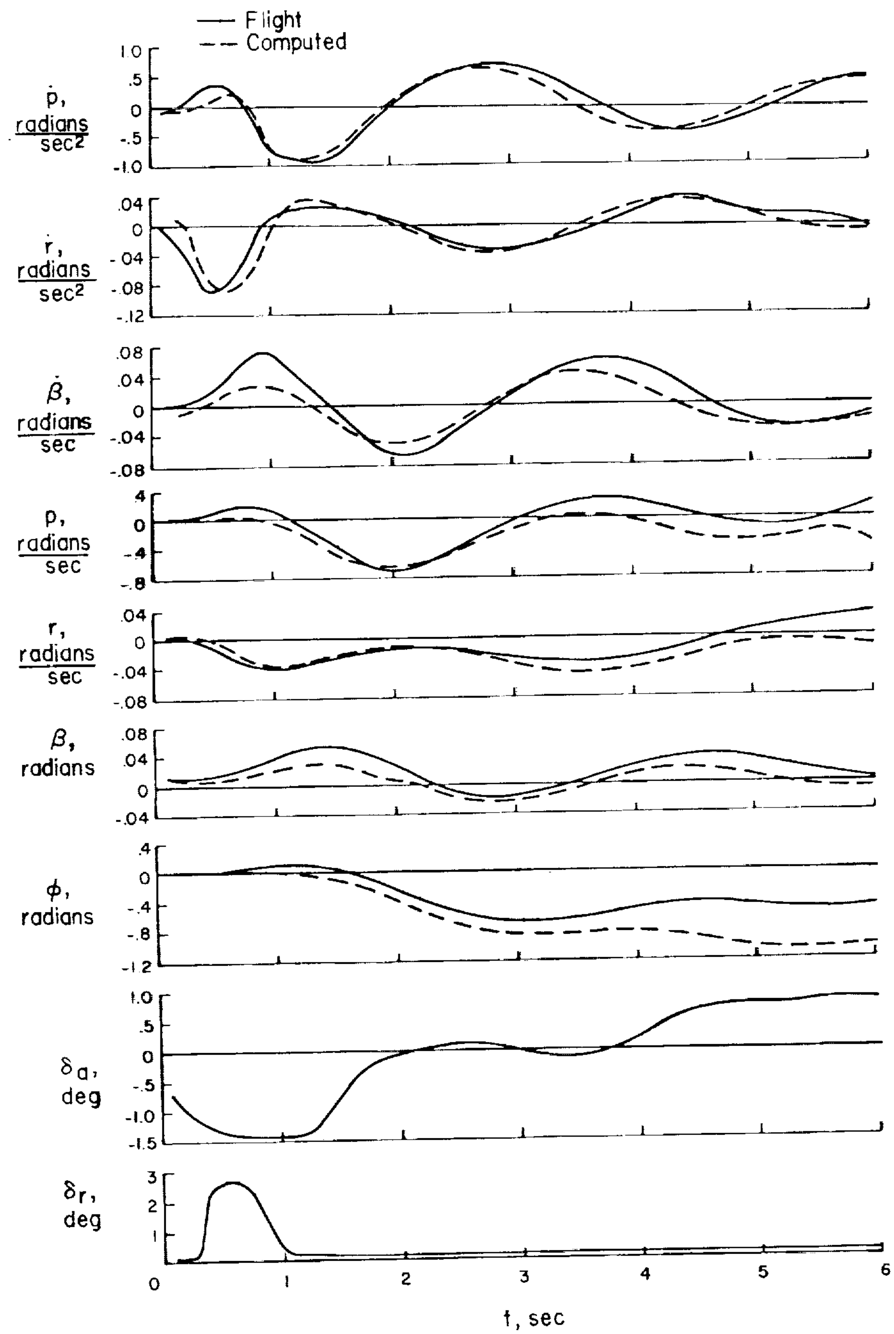

Figure 2. Comparison of time histories measured in flight and computed by using coefficients obtained from flight data by the method of simple equations (ref. 23).

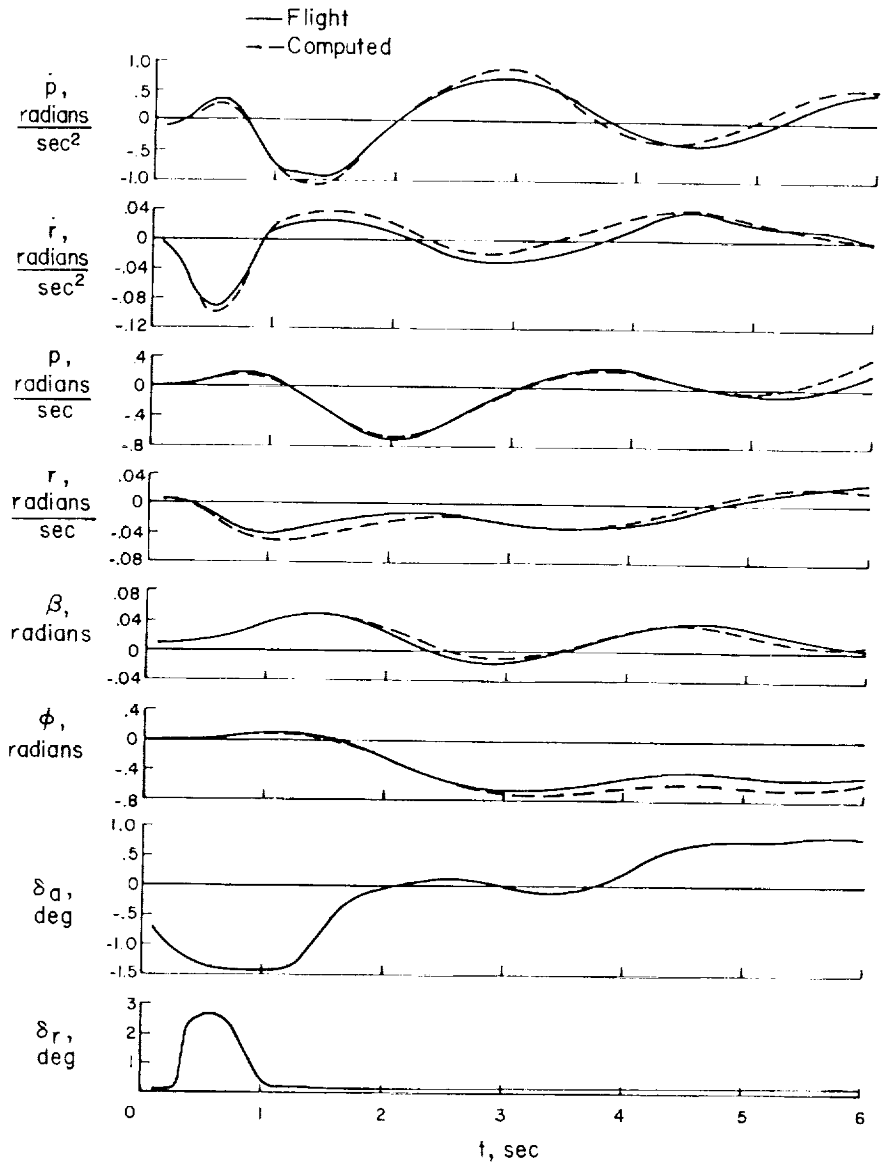

Figure 3. Analog match of time histories measured in flight.

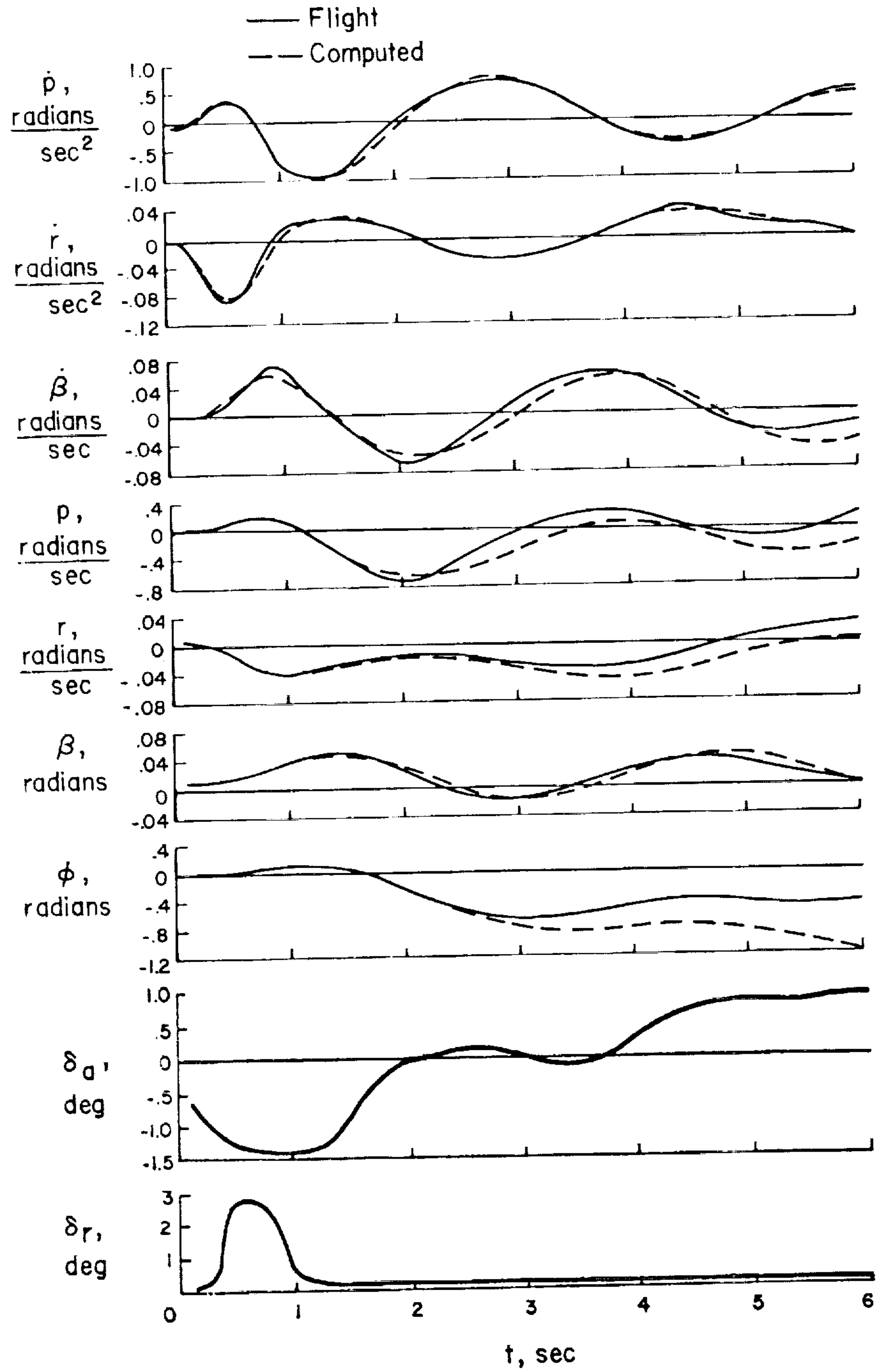

Figure 4. Comparison of time histories measured in flight and computed by using coefficients obtained from flight data with the least-squares method.

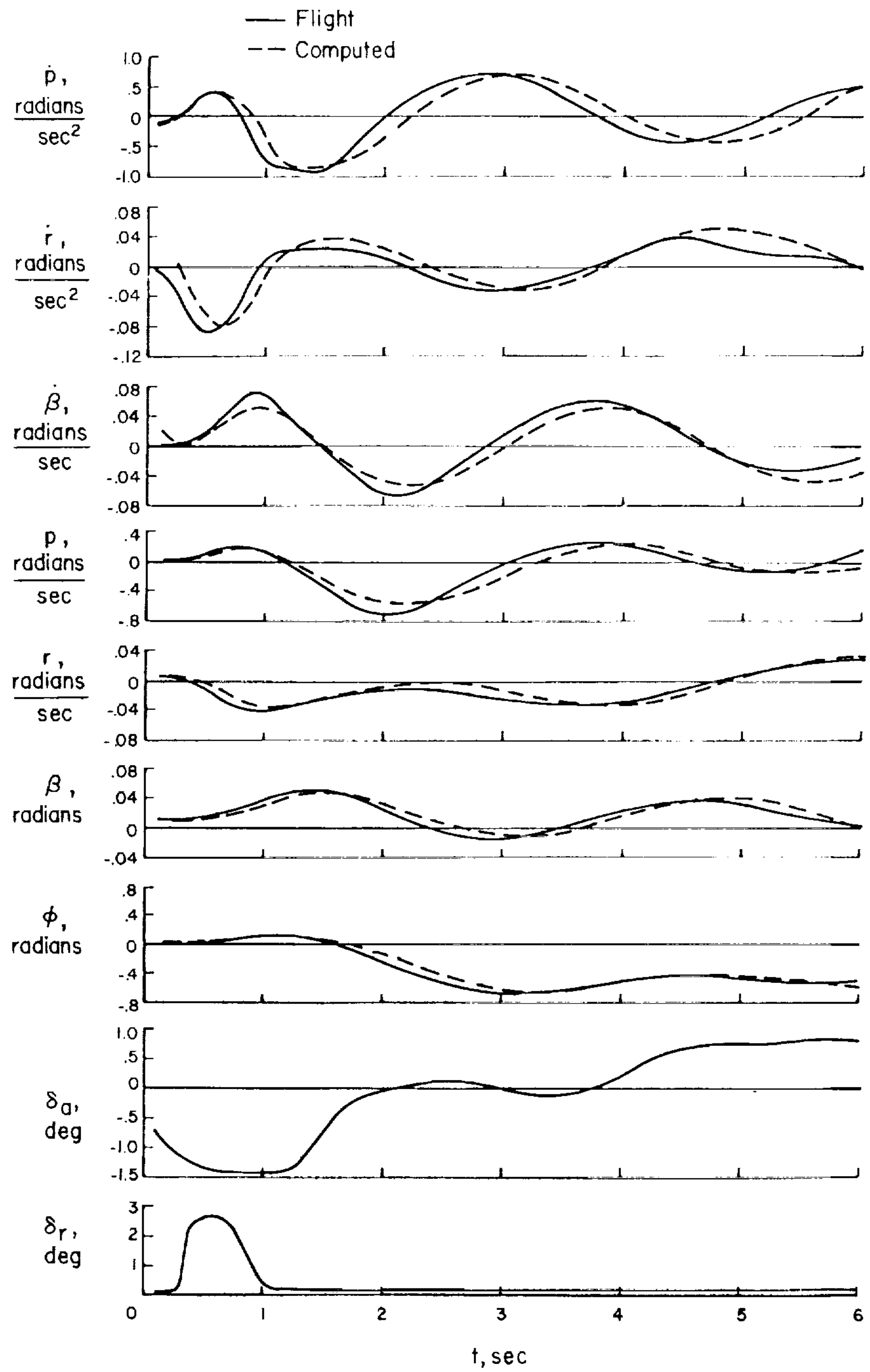

Figure 5. Comparison of time histories measured in flight and computed by using coefficients obtained from flight data by Shinbrot method (ref. 10).

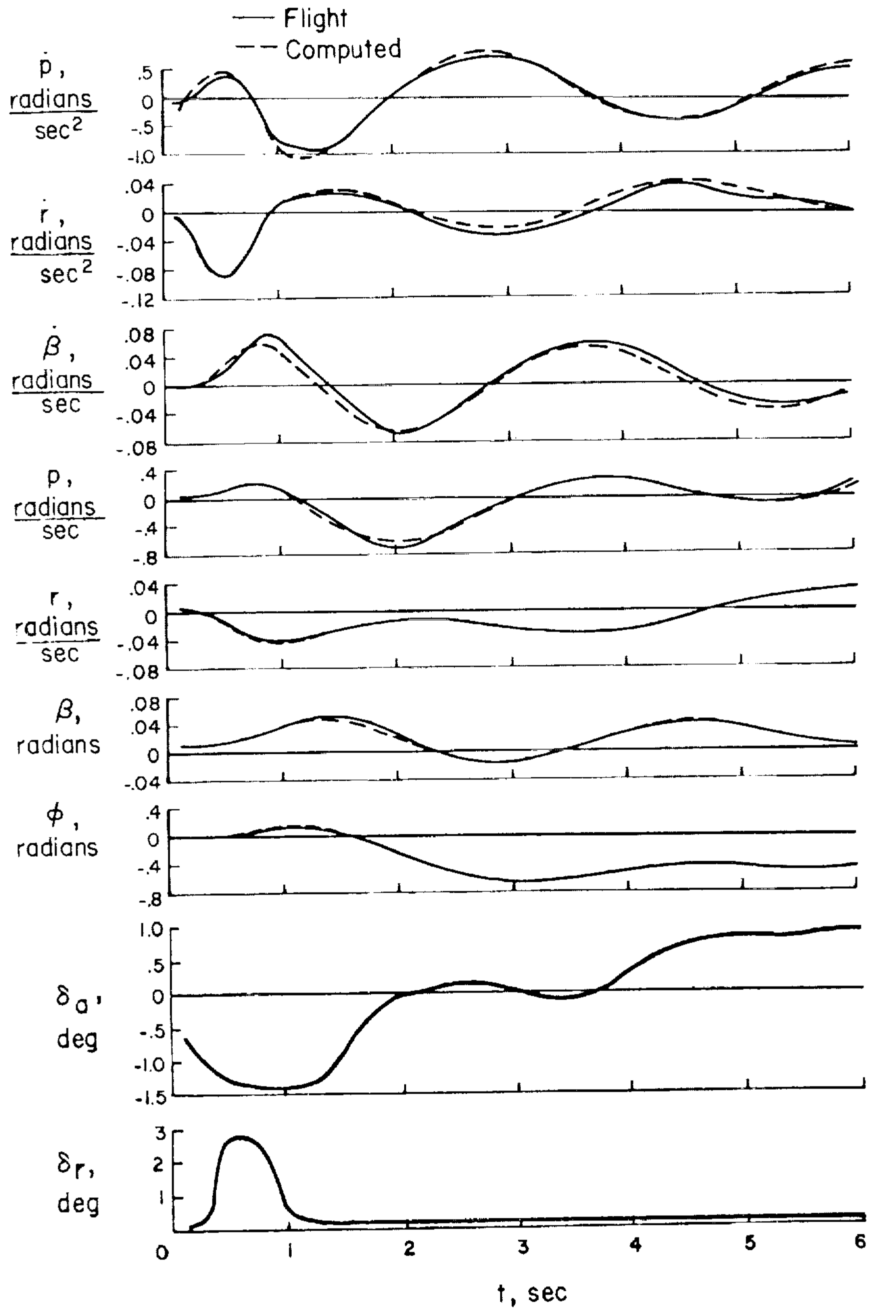

Figure 6. Comparison of time histories measured in flight and computed by using coefficients obtained from flight data with the Newton-Raphson method without a priori information.

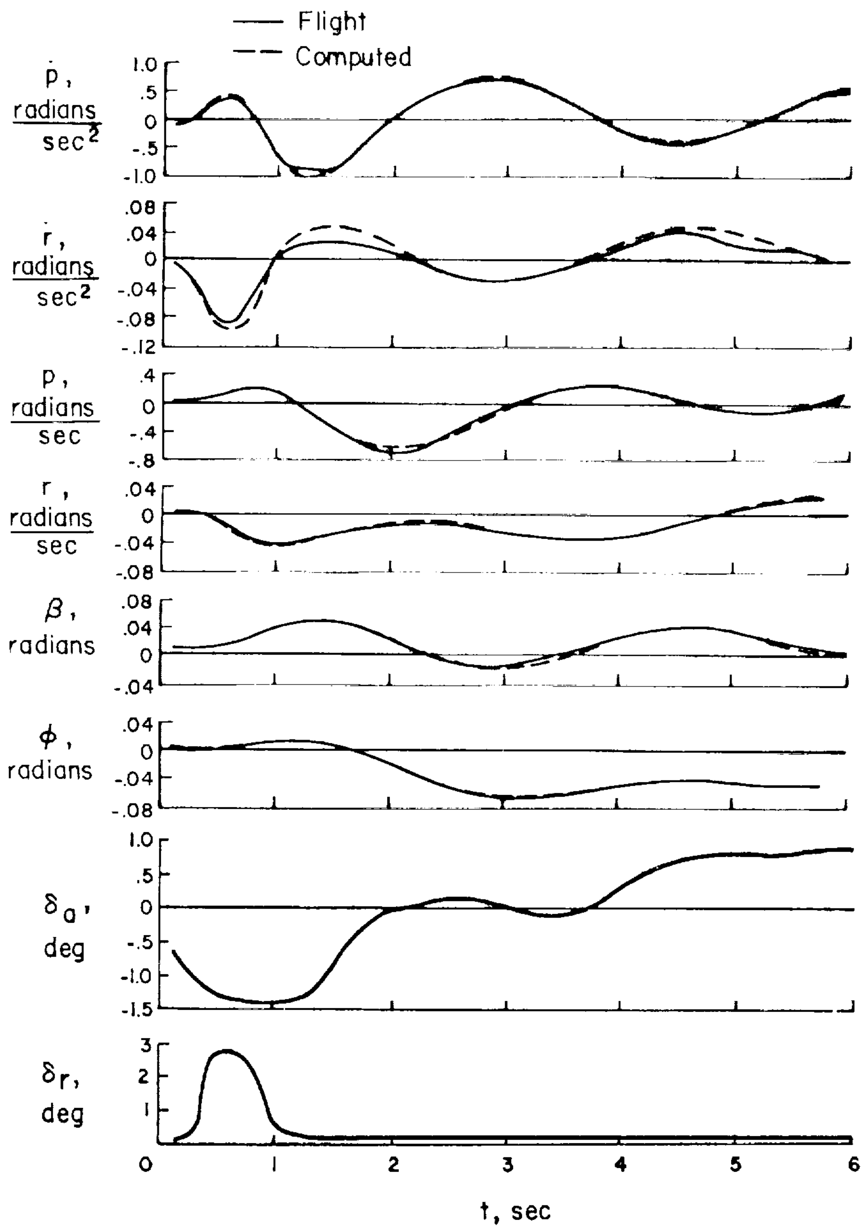

Figure 7. Comparison of time histories measured in flight and computed by using coefficients obtained from flight data with the Newton-Raphson method with a priori information.

The computer time required to obtain convergence for a problem with linear aerodynamics was about one minute on a IBM 7094. The other three problems reported involve various non-linearities.

At least one other advanced technique of extracting stability derivative values from flight data is currently under development at NASA's Langley Research Center. Called the Maximum Liklihood Method, the procedure employs a variational technique to minimize the error function. It is anticipated that a description of this procedure will appear within the next year as a NASA TN. One would also expect that additional refinement will appear from time to time because of the importance to good design of reliable flight test derivative values.

The experiences reported in some of the modern papers seem to confirm the view that for systems which are well represented by linear equations, for example a light aircraft in most of its flight maneuvers, one can expect to obtain good results with recent derivative extraction procedures provided

(1) the original data is accurate, relatively noise free, and readable to three significant figures,

(2) all accelerations, as well as velocities and displacements, are measured so as to reduce the computation's dependence on any particular measurement and to eliminate the need to compute accelerations by numerical differentiation of possibly degraded velocity data.

It follows that the fewer the data channels available and the more complex the equations required to describe the motion, the less likely one is to obtain satisfactory results.

LITERATURE REVIEW—

INSTRUMENTATION FOR STABILITY

AND CONTROL FLIGHT TESTING

GENERAL REMARKS

The present discussion considers the questions: (1) What parameters must be measured in order to determine the values of the stability derivatives from flight data? (2) What instruments does one use to measure the required parameters?

During the discussion, two statements are held to axiomatic: (1) The more independent parameters one can measure directly with comparable quality, the more unambiguously and the more deterministically one can assess the values of a given number of stability derivatives. (2) The ultimate accuracy of any stability derivative extraction procedure is limited by the signal-to-noise ratio of the measuring instruments.

Probably the most common deficiency of flight test instrumentation systems today is the use of inferior data transducers. Money saved there is penny-wise and dollar-foolish. Compromises are often made in order to obtain more channels of data and to make possible more rapid data reduction; but it must be remembered that no amount of massaging can make really poor data good while a smaller quantity of good data can often serve many purposes. On light aircraft where flight time is relatively inexpensive such compromises cannot be justified as cost effective.

It follows from (1) above and the fact that there are many more stability derivatives which one would wish to evaluate than independent equations of motion, that one should measure as many of the independent parameters describing the motion as possible. Those which are readily measured include:

(a) Angles

angle of attack, α
angle of sideslip, β
pitch angle, θ
roll angle, ϕ
yaw angle, ψ

(b) Velocities

airspeed, V
roll rate, p
yaw rate, r
pitch rate, q

(c) Accelerations

longitudinal acceleration, a_x
vertical acceleration, a_z
lateral acceleration, a_y
roll acceleration, $\dot{p}$
yaw acceleration, $\dot{r}$
pitch acceleration, $\dot{q}$.

In addition to these 15 primary measurements one can measure latitude and longitude as functions of time with navigational devices and vertical height as a function of time with an altimeter. If the wind velocities and wind shears are known, then suitable integrals of the accelerations can be equated to the velocities; and integrals of the velocities can be related to the position. Such comparisons are quite valuable in establishing the creditability of the primary measurements.

If the weight and thrust during steady flight are also known, then the resultant static aerodynamic forces are readily established from the kinematics of the situation. These forces are then easily reduced to coefficient (stability derivative) form.

Finally, to establish the forcing functions applied to the aircraft, it is necessary that the positions of the aerodynamic control surfaces be measured as functions of time.

The instruments available for measuring the parameters enumerated above are discussed briefly in the following sections. It was felt that a detailed treatise on each instrument type was beyond the scope of the present discussion. The operating principle of each type is related and some typical measuring accuracies are given. Precautions to be observed in application are also stated.

Additional discussion of the instrumentation requirements of stability and control flight testing may be found in references 33 and 34. The discussion found in reference 23 relates NASA practice in this area.

MEASUREMENT TECHNIQUES

Aerodynamic Angles

α and β are defined as the angles which the aircraft reference line makes with the tangent to the flight path. If the aircraft did not disturb the flow field locally, then the entire flow would always be parallel to the tangent to the flight path. But because the local flow has a different direction from that at "infinity", one must either correct a local measurement for this deviation (which depends on C_L, primarily) or place the sensor in a region where the flow deviation from its "true" value is negligably small. For a small, single-engine aircraft the most convenient location, aerodynamically, is at the end of a boom mounted from a wing tip. A boom extending about six feet in front of the wing tip is generally adequate. It should be determined beforehand, however, that boom bending and wing torsion are within the limits expected of the α and β measurements.

Sensors are generally of two types: fixed or movable. The movable type is typically a vane. A large flat plate or wedge is mounted at the trailing edge of a rod. The nose of the rod is weighted so that the rod and plate are mass balanced about a pivot point. A shaft, attached to the rod at the pivot point, leads to a position transducer which also serves to secure the rod to the aircraft. The aerodynamic characteristics of the rod are such that a very powerful moment is developed if the rod does not align itself with the local flow. One also desires that the ratio of this aerodynamic moment to rod inertia be very large so that the vane will accurately follow high frequency disturbances. A natural frequency of 100 radians/second is achievable (ref. 35). Alignment accuracy is generally on the order of 0.1° in a carefully-constructed device. A position transducer capable of resolution to one part in 400 is usually required to take advantage of this accuracy.

Fixed, pressure-sensing angle-measuring devices are capable of the same accuracy, resolution, and response as movable vanes. They have the advantage of being inherently more reliable because they have no moving parts external to the aircraft. There are no bearings to sieze or static frictions to overcome. Fixed devices are not as sensitive to small distortions in their geometry, and they can be fabricated to receive smaller stresses from external loads. Their external dimensions are also more compact. Their principal disadvantage lies in the number and cost of the required pressure transducer(s). This may be seen from the fact that angle-of-attack (or angle-of-sideslip) is a direct function of the ratio of two pressure differences. Because of this the cost is about a factor of ten higher than for the movable vane.

References 36 and 37 present a comprehensive discussion of the aerodynamics of α and β sensors and contain a bibliography of earlier work. It may be mentioned in passing that no significant works on the subject seem to have appeared in the last 14 years. The authors have also been unable to find any description of recent tranducer developments suitable for use with these devices, particularly the pressure-sensing versions. The technology for advanced versions of previous transducers, however, is known to exist, and it remains only to undertake their development.

Angular Orientation in Inertial Space

The development of inertial navigators for intercontinental ballistic missiles led to many improvements in the design and construction of gyroscopes. Many of these techniques are evident in the so-called vertical gyros now finding increasing use in light aircraft. These devices, which can measure both pitch and roll, are provided with inertial vertical references to reduce drift whenever the aircraft is in unaccelerated flight. A rate-sensing switch, a long time constant, or unique construction prevents the "erection" system from producing errors during maneuvers. Accuracies are typically within 0.1°. Drift is less than 7.5°/hour.

The roll-stabilized, directional gyro with a magnetic flux gate performs substantially the same function for yaw angle measurements.

Since these free or two-degree-of-freedom gyros ideally are not sensitive to angular or linear accelerations, they may be mounted anywhere within the airframe and will give the same indication.

A very helpful, succinct discussion of the theory of operation of these devices and their present state-of-development is given in reference 38. Generally, for accuracies superior to those quoted above and for lower drift rates, the use of "stable tables" or inertial navigator platforms is recommended. The various types platforms are also discussed briefly in reference 38.

Airspeed

The correct determination of indicated airspeed requires the existence of (1) a pitot pressure source located in an area free of wakes and propeller slipstream and (2) a static pressure source located in such a position that the local pressure is the same as in the free stream at all airspeeds. These conditions are seldom met in light aircraft. The static pressure source is usually located on the fuselage in the cabin area where a measurable "position error"--variable with lift coefficient and sideslip--exists. Further, cabin instrumentation is generally inadequate for accurate, responsive interpretations of dynamic pressure as airspeed and of static pressure as pressure altitude. Finally, proper attention is seldom given to balancing the pitot and static lines so that the pneumatic lags are equal.

What is required for dynamic measurements is a quality airspeed head--one which is guaranteed to have an inherent static pressure error no greater than 1% of dynamic pressure for $\alpha < 30^{\circ}$ and $\beta < 10^{\circ}$--mounted at the end of a 6-foot-long, rigid boom, itself located at the wing tip of the aircraft. The static orifices in the airspeed head should be no less than 4 in number, each with a diameter of .070 inches. One-fourth inch or larger pneumatic lines should be used and the pressure transducers should be located at the base of the boom or even in the boom if possible. The pressure transducers should not be sensitive to acceleration and should have a basic accuracy of $\frac{1}{4}$ in. of H_2O (62.2 newtons/meter^2). With such an instrument it is possible to measure speed changes of 2.16 mph accurately and thus determine the x-direction derivatives.

Reference 39 gives a rather complete discussion of this history of airspeed heads, results of an extensive series of wind tunnel tests on a variety of airspeed head configurations, and a semi-empirical procedure for modifying the head configuration to offset the aircraft's position error. Rosemount Engineering Corporation (ref. 40 and 41) has for some time marketed airspeed probes utilizing another method of position error compensation.

Angular Velocities

Angular velocity components are almost always measured with rate gyroscopes. These are gyroscopes which are constrained to one-degree-of-freedom, and their displacement about the output axis is proportional to the angular rate input to the input axis. (Position-measuring gyros, on the other hand, have two-degrees-of-freedom, that is, two gimbals.) To measure the three components of angular velocity three rate gyroscopes are required. Generally, precision is better and drift rate lower than with two-degree-of-freedom gyros. The most common type of signal pickoff used with gyros is a synchro.

Accelerations

In an accelerometer a mass is positioned in the case by two springs. When the case is accelerated the inertia of the mass makes it move relative to the case. If one restrains the mass's motion to a straight line this becomes the devices' axis of sensitivity. By measuring the displacement of the mass relative to the case and knowing the spring constants one can calculate the acceleration. The three components of linear acceleration are readily measured with devices of this type. Accelerometers used with inertial navigators typically can sense accelerations as low as 10^{-5} g; thus, these devices are often the most accurate instruments in the entire flight test instrument repertoire and should therefore be used extensively. Careful filtering of the output signal may be necessary because accelerometers will also respond to vibrations of their supporting structure. These vibrations can be induced by the engine, structural resonance, and atmospheric and boundary layer turbulence.

Care must be exercised in the mounting of accelerometers. If they are located off the c.g. they will indicate a component due to the angular velocity of the aircraft: $a = \ell\omega^2$, where a is the contribution to the total acceleration, ℓ is the distance from the actual c.g. to the accelerometer mounting and ω is the component of the aircraft's angular velocity in the plane described by ℓ and the accelerometer's axis of sensitivity.

It will be noted that if the accelerometer mass is mounted on a shaft and constrained by two torsional springs it becomes an angular accelerometer. Unfortunately, such devices have not found the widespread application they deserve. Since the measurements can be made with high precision and low noise, they are excellent as additional, independent data channels for use in improving the reliability of stability derivative extraction procedures. Raw angular acceleration data is also useful for estimating the aerodynamic moments produced by control surface deflections.

Control Surface Position

Generally, something as simple as the wiper of a potentiometer or a synchro is connected to the control surface torque tube for position indication. The potentiometer must be capable of resolution of about one part in 500 to maintain accuracy comparable to that of other elements in the measuring system. Calibration is usually carried out with an accurate protractor. Operationally, the major concern is for the noise introduced into the signal by structural vibrations.

Weight

Measurements of in-flight weight are usually acomplished by measuring first the weight of the dry aircraft on the ground. The fuel volume and its specific gravity are then noted as is the payload. The fuel consumed up to a given time is then subtracted from the starting weight to find the weight at that time. A fuel totalizer (integrating flowmeter) is usually used for this purpose. Through the use of such means, the weight at any time can be determined to within a pound or two.

Thrust

Direct thrust measurements on propeller-driven aircraft are extremely difficult to make. An indirect method is usually employed. This involves a knowledge of the airspeed and the power delivered to the airstream. Hence, the engine test cell data for the given engine manifold conditions must be known as well as the propeller characteristics when installed on the sample airplane. Knowledge of the thrust in steady level flight, of course, is tantamount to a measurement of aircraft drag.

Signal Conditioning and Recording

For many years a substantial effort has been devoted to improving the techniques for in-flight recording of the indications of data transducers. The techniques of course are applicable to missile and space craft testing as well as to aircraft testing. The objectives have been to (1) improve the signal-to-noise ratio, (2) increase the data packing density on a given quantity of recording media, and (3) record the data in a form compatible with automated data reduction procedures. Inevitably this effort led to digital encoding schemes using magnetic tape as the recording medium. Since most transducers are normally considered to provide an analog output signal, some form of analog-to-digital converter must be used. Care must also be taken to scale the signal for best signal-to-noise ratio. References 42 and 43 document in a very detailed fashion the design analysis used to arrive at an advanced, digital flight data system. Although the system was intended for V/STOL aircraft, much of the computer software, error analyses, data recording techniques, etc. are equally applicable to other aircraft types.

The sophistication of such a system is justified primarily by the very high cost of flight time and the large amount of data in addition to flight dynamics which must be acquired on each flight. Frequently, for light aircraft, the latter situation is not present and the cost of flight time and additional data reduction time are less than the cost of complex signal processing and recording equipment. In these circumstances, an analog recording of 12 in. wide oscillograph paper running at a speed of 5 inches per second is quite sufficient if the individual traces have a maximum amplitude of, say ±3 in., for the expected maneuvers. The traces can be read by hand with sufficient accuracy for later digital processing.

PILOTING TECHNIQUES

It is perhaps an obvious truism that the excitation which the pilot applies to the aircraft should be tailored to the character of the data he desires to obtain. For example, if one is interested in measuring $C_{\ell_{\delta_a}}$ he should perform a maneuver in which $C_{\ell_{\delta_a}}$ is a dominant factor--such as a rapid roll. Since it is difficult to determine C_{ℓ_p} and $C_{\ell_{\delta_a}}$ individually in a steady roll, it is preferable to make precise measurements of $\dot{p}$ and δ_a at the onset of a roll where the damping due to roll is still small. For most aircraft, the time during which this is possible is very short--on the order of 50 milliseconds. Thus, to employ this technique it is necessary to use instrumentation capable of accurately recording rapid transients. Further, the pilot must extend the control surface in such a fashion that the high frequency content of the responses are well excited in order to obtain a satisfactory signal-to-noise ratio. In other words, the pilot must attempt to apply a pulsed aileron deflection resembling a delta function.

Many derivatives, on the other hand, can only be evaluated from changes in the equilibrium aerodynamic forces and moments. Rapid control surface pulses do not excite the aircraft motions in a way that permits accurate extraction of these so-called static derivatives; their extraction is therefore facilitated by the use of long control surface pulses, *i.e.*, pulses where the excitation of the aircraft near zero frequency is substantial.

Because of this dissimilarity in excitation requirements, it is usually preferable to extract derivative values from responses obtained with a range in pulse widths, giving more weight to the values obtained with the appropriate excitation. Generally, pulses are performed from a trimmed condition in smooth air. So-called double pulses--consecutive pulses of equal and opposite amplitude--are frequently employed so that the aircraft will not depart greatly from its original condition. Recording of the aircraft motion in response to a pulse disturbance is generally continued for a period of 15 to 30 seconds in order to define adequately the low frequency components of the motion. Pulse amplitude is usually kept small so that the assumption of small perturbations is not violated. Increasing amplitudes can be employed to determine the point at which significant inertial or aerodynamic non-linearities are introduced. It is, of course, desirable to employ the largest input compatible with the small perturbation assumption to obtain the greatest signal-to-noise ratio. Larger inputs may be used with non-linear analyses to define second order effects and cross-couplings.

Elevator pulses are employed to excite the longitudinal responses (u, α, θ, etc.) while both rudder and aileron pulses are used to excite the lateral-directional responses (β, ϕ, ψ, etc.).

A RECOMMENDED PROCEDURE FOR EXTRACTING STABILITY DERIVATIVES FROM LIGHT AIRCRAFT FLIGHT DATA

INTRODUCTION

Because of its versatility and ease of application, the modified Newton-Raphson technique of Taylor *et al.* (ref. 29) was deemed most suitable for the reduction of light aircraft flight test data. A detailed examination was therefore conducted to determine the constraints on its application.* An important consideration was the degree of instrumentation accuracy necessary to establish reliable aircraft parameters. For this purpose, the tehcnique was used to test simulated flight data and investigate the amount of noise that actual test data could contain and still be useful. These results can aid in establishing instrument specifications.

---

* A copy of the computer deck and program listing for the Newton-Raphson technique was obtained from Lawrence W. Taylor. The program is written in Fortran and required only minor modifications to run on an IBM 360/75. A detailed description of the computer program and its operation appears in a forthcoming NASA TN (ref. 31). Copies of the program may be obtained from L. W. Taylor, NASA Langley, Hampton, Virginia.

LATERAL

The Newton-Raphson method, as employed by Taylor, is a means of selecting those parameter values which best fit an assumed model to a data set according to particular error criterion. The error criterion is more general than the classical least squares criterion in that it permits the fit error to $\dot{p}$, $\dot{r}$, β, and ϕ to be minimized as well as the fit error to p, r, and $\dot{\beta}$. The technique also enables one to use a priori values of the stability derivatives, bias terms, and initial conditions to improve the fit of the equations to flight test data. It is also possible to extract the stability parameters from incomplete flight data, a distinct advantage over several other techniques. The reader is directed to reference 29 for a more detailed exposition of the theory of this technique.

For investigative purposes, some "flight" time histories were computed by the following procedure. The linearized lateral equations of motion,

$$\begin{aligned}
\dot{p} &= L_p p + L_r r + L_\beta \beta + L_{\delta_a}\delta_a + L_{\delta_r}\delta_r, \\
\dot{r} &= N_p p + N_r r + N_\beta \beta + N_{\delta_a}\delta_a + N_{\delta_r}\delta_r, \\
\dot{\beta} &= \frac{Y_p}{U_o} p + \left(\frac{Y_r}{U_o} - 1\right) r + Y_v \beta + \frac{g}{U_o}\phi + \frac{Y_{\delta_a}}{U_o}\delta_a + \frac{Y_{\delta_r}}{U_o}\delta_r, \\
\dot{\phi} &= p ,
\end{aligned} \tag{7}$$

with I_{xz} assumed zero, were solved in the Laplace domain and time histories calculated by the method of residues as given in reference 44. Values for the dimensional stability derivatives used as coefficients for equations (7) were those of a typical light aircraft, the Cessna 182. The values of p, r, β, ϕ, $\dot{p}$, $\dot{r}$, and $\dot{\beta}$ resulting from steps of 3^o and 20^o were tabulated at intervals of 0.025 seconds for a period of 10 seconds. These responses are plotted as solid curves in the figures showing the fit obtained by the Newton-Raphson technique. Because aileron deflection was assumed to be zero, values for L_{δ_a} and N_{δ_a} could not be determined. Based on results of a sensitivity analysis presented in reference 1, Y_p, Y_r, Y_{δ_a}, and Y_{δ_r} were taken to be zero. Thus, the problem reduces to a system of three equations containing nine unknown parameters, as shown below:

$$\begin{aligned}
\dot{p} &= L_p p + L_r r + L_\beta \beta + L_{\delta_r}\delta_r, \\
\dot{r} &= N_p p + N_r r + N_\beta \beta + N_{\delta_r}\delta_r, \\
\dot{\beta} &= -r + Y_v \beta + \frac{g}{U_o}\phi, \\
\dot{\phi} &= p .
\end{aligned} \tag{8}$$

The Newton-Raphson technique was required to fit the computed time histories of p, r, β, ϕ, $\dot{p}$, $\dot{r}$, and $\dot{\beta}$ for various situations. Equations (8) can be written in matrix form as:

$$\begin{bmatrix} \dot{p} \\ \dot{r} \\ \dot{\beta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} L_p & L_r & L_\beta & 0 \\ N_p & N_r & N_\beta & 0 \\ 0 & -1 & Y_v & \frac{g}{U_o} \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ r \\ \beta \\ \phi \end{bmatrix} + \begin{bmatrix} L_{\delta_r} \\ N_{\delta_r} \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} \delta_r \end{bmatrix} \tag{9}$$

A short-hand version is given by

$$\dot{\underline{X}} = A\underline{X} + B\underline{U} \quad . \tag{10}$$

The dimensional stability derivatives used in the A and B matrices of equation (10) as a starting point for the first iteration will be referred to as the initial values of the stability parameters. The Newton-Raphson technique minimizes the fit error, J, one part of which is the weighted mean square difference between the system responses and responses from the model of the system. The technique simultaneously minimizes the difference between the computed and the a priori values of the parameters. A priori values and initial values are the two ways available for introducing background knowledge of the stability parameters into the computational scheme.

The Newton-Raphson technique was first applied to the computed data using zero initial and no a priori values. After ten iterations, the fit error remained large and had converged to an erroneous set of stability parameters which gave a poor fit of the aircraft dynamics. This tendency toward local convergence probably results from portions of the aircraft's response being under-excited by the rudder step. Possibly, convergence to realistic values of the parameters would occur from zero initial values for data obtained after disturbing the aircraft with more violent actuation of the controls. Figure 8 shows the results of this attempt to fit the computed data due to a rudder step of three degrees (0.0524 radians).

Following some initial fluctuations the fit error, J, (see figure 9) levels out and, after seven iterations, indicates no significant improvement of the fit. This suggests that the technique has converged, but to unreliable values of the stability parameters (figure 8). Examination of fit error, J, versus iteration number in figure 9 and the time histories in figure 8, indicates that additional information concerning the values of the dimensional stability derivatives is necessary to obtain an adequate fit of the data.

Thus, an effort was made to develop techniques for providing initial approximations of these parameter values, which could, in turn, be used as inputs to the Newton-Raphson technique in either the role of a priori values for the parameter values in the error criterion or as initial values of the A and B matrices.

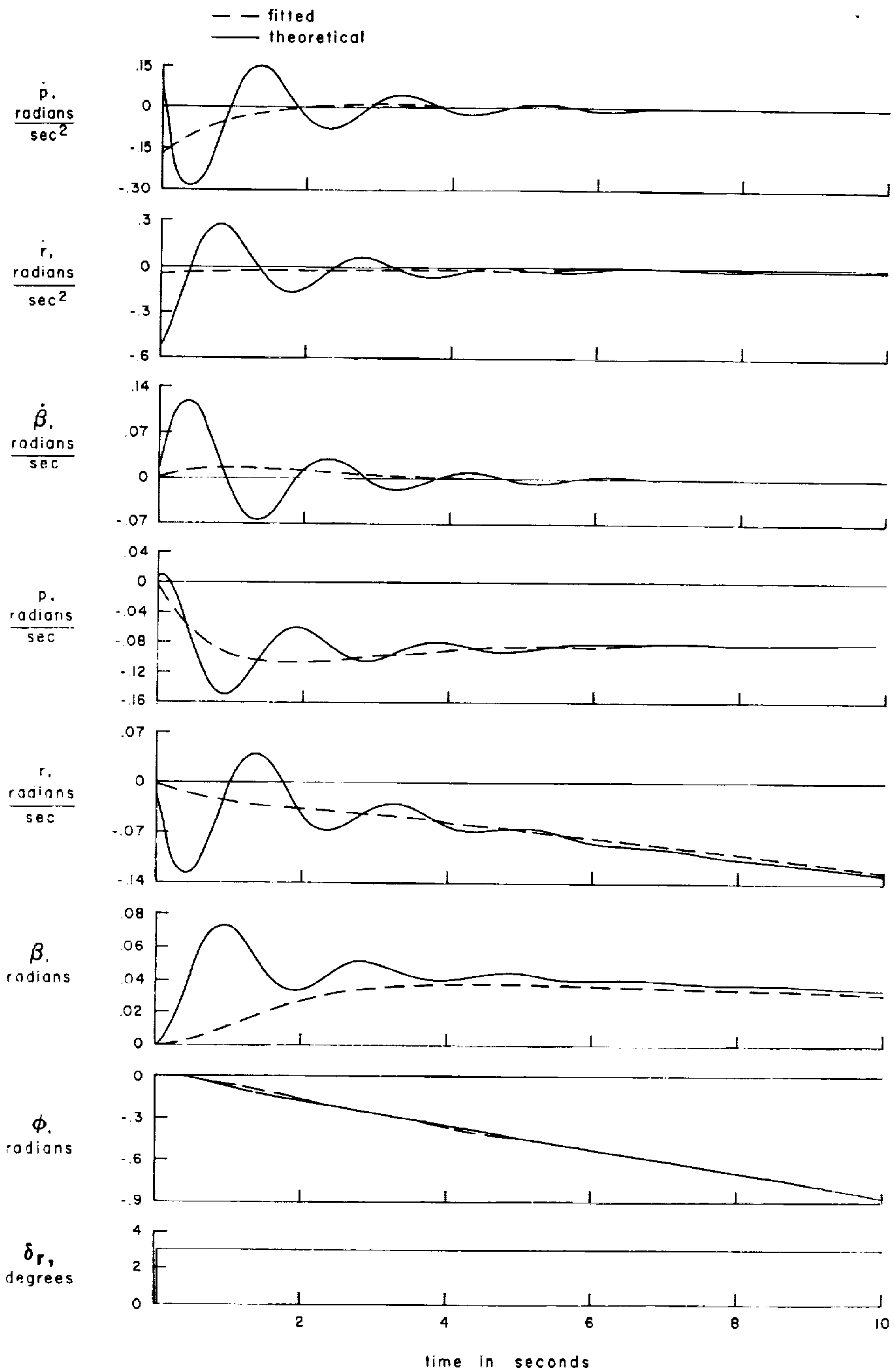

Figure 8. Comparison of time histories resulting from zero initial values and no a priori information.

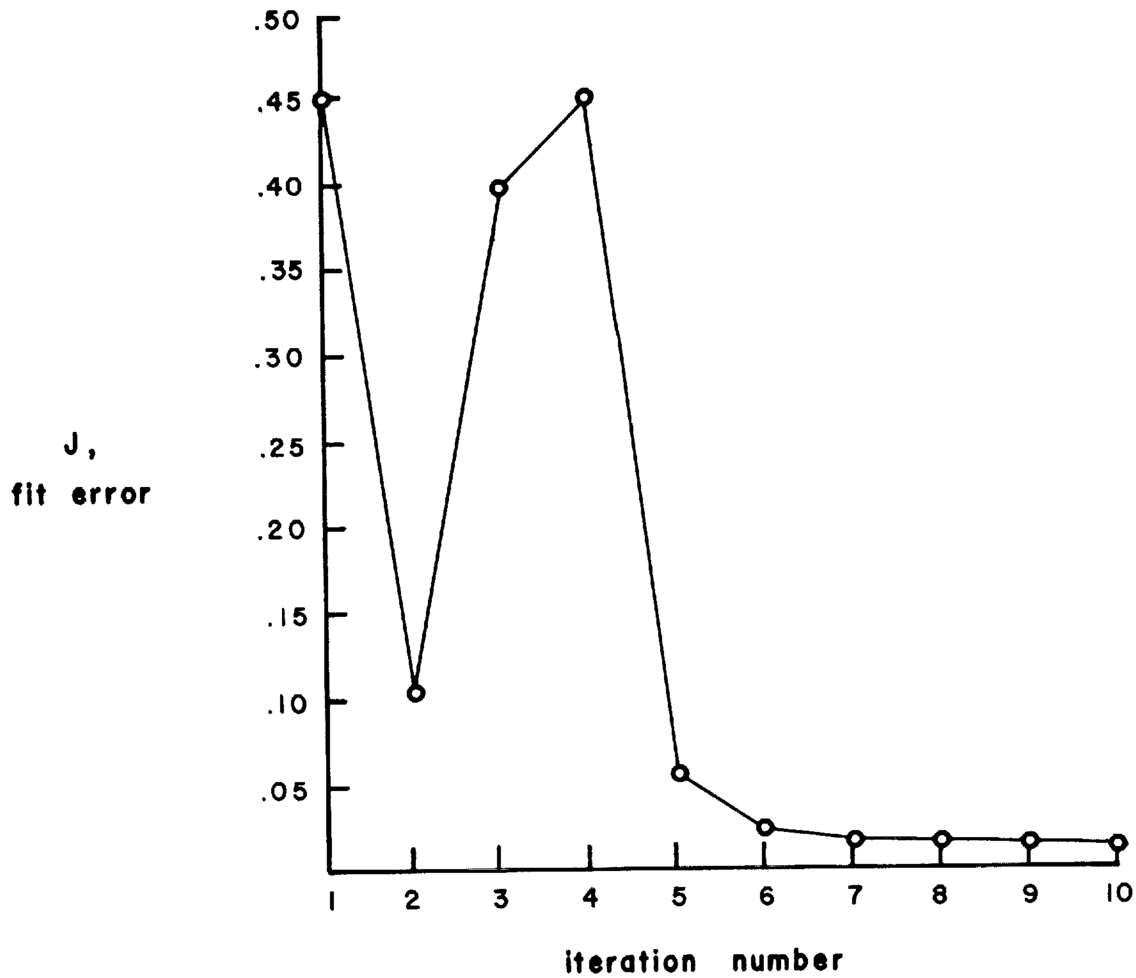

Figure 9. Convergence of the fit error for zero initial values and no a priori information.

To estimate values for the dimensional derivatives, the methods for calculating the non-dimensional stability derivatives presented in reference 1 were reviewed, and the procedure considered most accurate for each derivative was programmed for the digital computer. These procedures were divided to form two programs, one for the longitudinal mode and one for the lateral. A simple polynomial curve-fitting scheme was used to describe the methods which rely on information from experimental or theoretical graphs. By using these polynomial curve-fits and the included interpolation procedures, it is felt that data obtained from these programs is as accurate as that estimated from the actual graphs. Once the methods for estimating all of the non-dimensional stability derivatives were computerized, the dimensional stability derivatives could be calculated by simply "inputting" certain inertial and geometric information to the programs. Program listings and sample outputs are presented in Appendix B.

With these programs to provide prior knowledge of the stability parameters, the Newton-Raphson technique converged more rapidly and gave realistic values for the stability parameters. The procedure for reducing flight test data would now include inputting to these programs pertinent geometrical and inertial characteristics and flight condition data for the aircraft being evaluated. These programs would then produce values for the stability parameters, by theoretical methods gleaned from the literature, for use as either initial or a priori input values in the Newton-Raphson technique.

The majority of the input data necessary for estimating prior values of the stability derivatives can be obtained from a three-view drawing of the aircraft under investigation. Inertial characteristics are normally available and flight condition data such as speed and altitude are readily determined. In addition to calculating the non-dimensional and dimensional stability derivatives, the programs were designed to evaluate the coefficients of the transfer function, extract transfer function poles and zeros by factoring the numerator and denominator polynomials, and calculate information necessary to describe the frequency response of the airplane.

Advantages of using the previously described computer programs for determining values of the dimensional stability parameters from flight test data are easily shown. For demonstrative purposes, values of the stability parameters used to compute the simulated flight test data of figure 8 were randomly varied by 25% both positively and negatively and used as inputs to the Newton-Raphson technique as both a priori and initial values.

First, a priori values (parameter values used in the error criterion) within 25% of the actual and zero initial values of the parameters were inserted into the computational routine and an attempt was made to fit the simulated flight test data of figure 8. The fit error, figure 10, indicates that convergence was obtained, but examination of the two sample traces in figure 11 shows that the dynamics of the airframe are not matched. Next, zero a priori values and initial values within 25% either positively or negatively of the actual were introduced into the Newton-Raphson technique, and a fit of the data from figure 8 was again attempted. The simulated flight test data was matched very closely, as evidenced by the examples of figure 12. The fit of response variables not shown in figure 12 was equally good, as indicated by figure 13, a plot of fit error versus iteration number. A comparison of figures 10 and 13 indicates that the fit error decreases by more than three orders of magnitude when the same prior knowledge of the stability parameters is inserted into the computational routine as initial values rather than as a priori values. In addition to demonstrating small fit errors and agreement with computed time histories, the technique should also determine values for the dimensional stability derivatives accurately. Table 1 presents a comparison of the results achieved from various approaches.

The actual values of the stability parameters in the second column of table 1 were those used to generate the simulated flight test data. Consequently, they represent values of the parameters which the technique attempts to recover. The fit error obtained when these values were

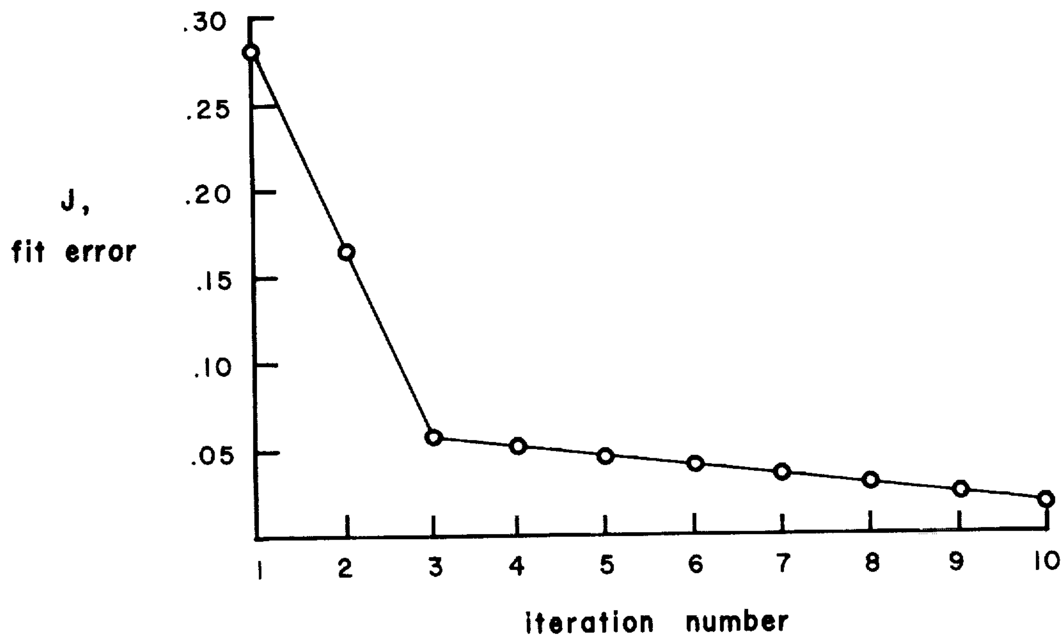

Figure 10. Convergence of the fit error for zero initial values and a priori values in error by ±25%.

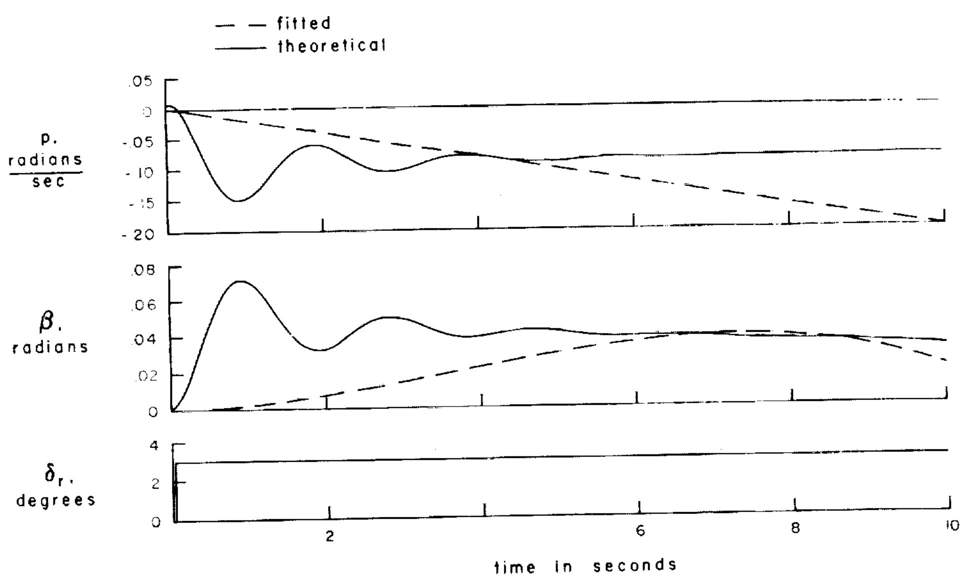

Figure 11. Comparison of example time histories resulting from zero initial values and a priori values in error by ±25%.

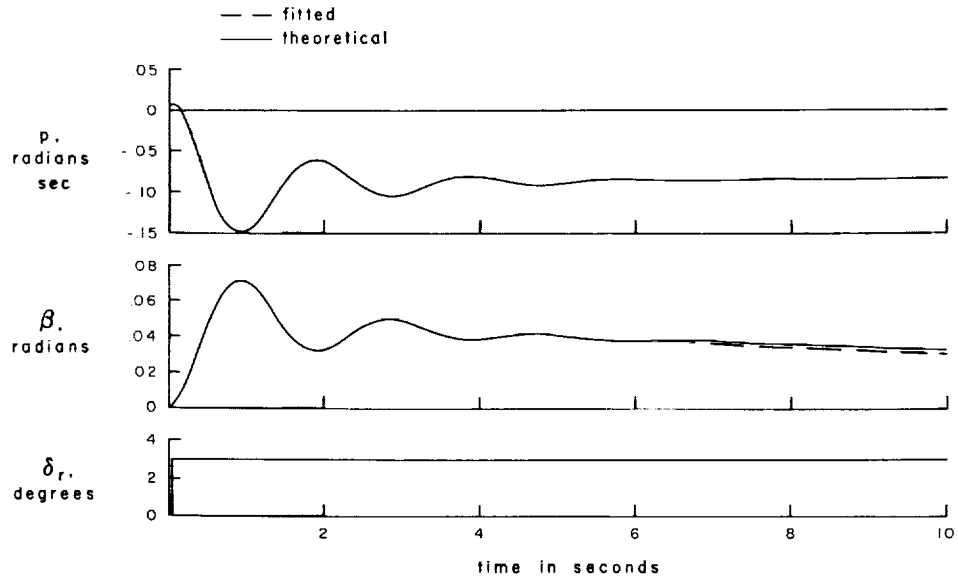

Figure 12. Comparison of example time histories resulting from initial values in error by ±25% and no a priori information.

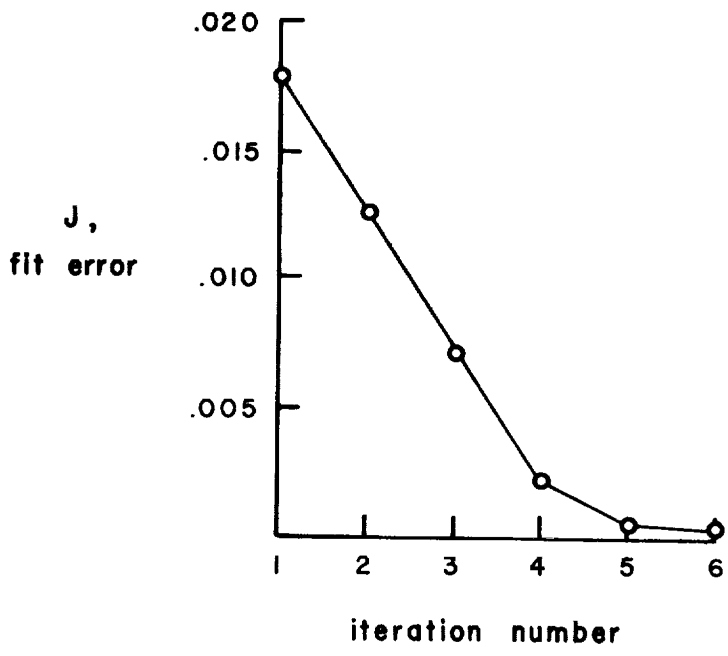

Figure 13. Convergence of the fit error for initial values in error by ±25% and no a priori information.

Derivative	Actual	Zero initial values Zero a priori values	Initial values in error by ±25%	A priori values in error by ±25%
L_p	-12.45	-1.15	-11.8	-10.02
L_r	2.54	-.141	2.56	2.89
L_β	-28.77	1.78	-27.3	-20.44
L_{δ_r}	4.75	-3.23	4.6	3.70
N_p	-.372	-.254	-.62	-.577
N_r	-1.26	-.009	-1.27	-.958
N_β	10.07	.198	9.49	12.72
N_{δ_r}	-10.25	-.785	-10.2	-8.20
Y_v	-.146	-.046	-.146	-.139
Fit error	$.47 \times 10^{-7}$	.0158	$.33 \times 10^{-6}$	.01718

Table 1. Comparison of lateral coefficient values.

inserted in the A and B matrices is theoretically zero, but appears as a small number due to machine round-off. The third column depicts values for the parameters obtained when no prior information is inserted into the routine. These parameters correspond to the fit error and time histories presented in figures 9 and 8, respectively. The fourth column lists values for the stability derivatives obtained when each parameter was varied by 25% either positively or negatively and then inserted in the Newton-Raphson technique as initial values. Figures 12 and 13 illustrate the time histories and fit error, respectively, obtained for this attempt. These same initial estimates were inserted into the computer program as a priori values, and the resulting parameters appear in column five. Figure 11 indicates that these parameters fail to match the dynamics of the aircraft.

In applications to actual flight data, the programs given in Appendix B would be used to generate the best available predictions of the theoretical values of all the stability derivatives for the particular aircraft and flight condition. These derivatives are then used as the initial estimates in the extraction procedure.

The stability derivative sensitivity analysis presented in reference 1 indicates that four stability derivatives (L_p, L_β, N_r, and N_β) are most influential in determining lateral stability. An investigation of column four in table 1, input of initial values within 25%, shows that each of these major derivatives was recovered within 6% of actual value. It should also be noted that, even though some of the unimportant derivatives such as N_p are in error by as much as 67%, the time histories are closely matched. After examination of table 1, one must conclude that, for best results, prior information concerning the stability parameters is best used as initial values in the A and B matrices of equation (10).

From the preceding results, it seems possible to determine reliable values of the stability parameters, provided good data and theoretical estimates of the parameters within ±25% of the actual value are available. It is the writers' opinion that the techniques gleaned from the literature and computerized in the programs of Appendix B are capable of estimating the important dimensional stability derivatives this accurately. Thus, the problem of obtaining usable data remains of primary concern.

Since the results of all parameter identification procedures depend heavily on the quality of test data available, instrumentation is basic to any analysis. Some of the more common instrument-induced errors include random noise, calibration errors, mounting inaccuracies, instrument bias, and time lags, among others. Consequently, to achieve reliable results, the data must be conditioned by compensating for instrument shortcomings. Reference 26 by Clinkenbeard *et al.* provides an in-depth investigation of methods for obtaining knowledge of instrumentation errors and techniques used to compensate for these errors prior to extraction of the stability parameters. In the present study, several of the errors in data acquisition deemed most likely to occur in light aircraft flight tests were considered.

First, the effect on parameter evaluation of using data which contains random noise is investigated. An ideal situation would be to provide the instrumentation engineer with a chart of the type and maximum amount of data noise permissible to obtain the important stability parameters within a certain accuracy. However, the variety of noise types, methods of noise compensation, and techniques for stability parameter determination make such a categorization impossible at this time. Instead, by use of the Newton-Raphson technique, an attempt was made to correlate parameter evaluation accuracy with the amount of allowable noise of the more prevalent type. This was accomplished by generating exact time histories from known dimensional stability derivatives and attempting to retrieve these known coefficients from the time history after it had been contaminated by random noise. The generated time histories, shown in figure 8, were contaminated with random noise having a normal distribution with zero mean and a standard deviation of unity. This noise was scaled to be a given percentage of the absolute value of the largest peak for each of the input time histories. Therefore, 5% random noise implies that at each data point a random amount was added to the time history corresponding to 5% of the largest value in the recording interval.

Consider first figure 14 which shows an attempt to fit generated data containing 5% random noise using the Newton-Raphson technique. Initial values of the stability parameters in error by 25%, either positively or negatively, were inserted in the technique as a starting point for the first iteration.

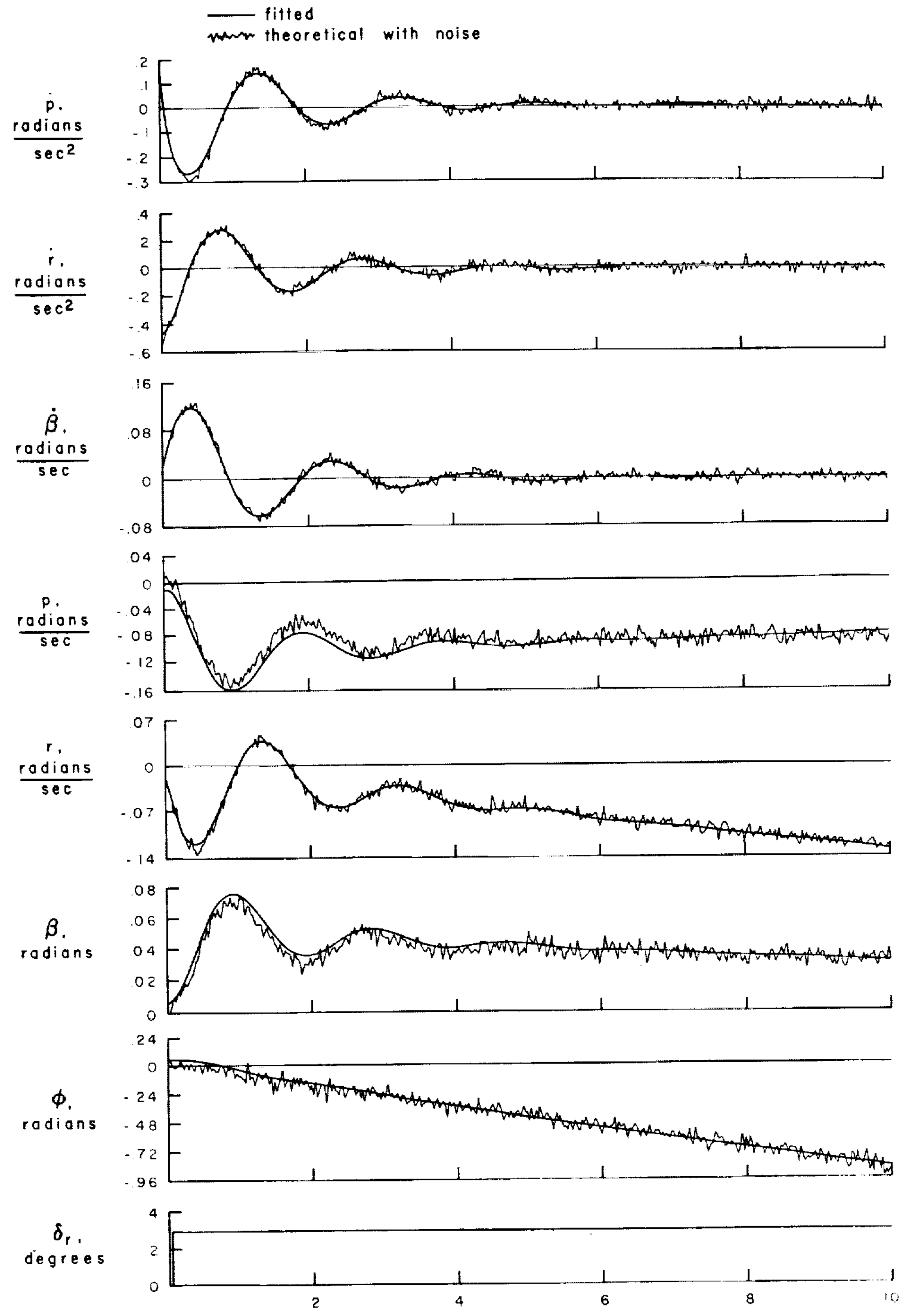

Figure 14. Comparison of time histories resulting from an attempt to fit data containing 5% random noise.

Convergence was obtained and a good match of the time histories achieved, with the exception of slight deviations in the traces of p and β as figure 14 illustrates. However, large errors occurred for some of the more important parameters such as N_β, L_β, L_p, and L_{δ_r}. These deviations probably result from a noise-induced error magnification associated with the non-uniqueness problem encountered when solving for more unknowns than given equations.

To determine the effect on parameter evaluation of random noise of various magnitudes, the computed flight test data was also contaminated with 3% and 10% random error. The noise was again normally distributed with zero mean and standard deviation of one. Also in keeping with the study of 5% noise, initial values of the coefficients in error by ±25% were used as a starting point for the first iteration. Table 2 presents a comparison of the accuracy with which the stability parameters were retrieved for varying levels of random noise. As expected, table 2 reveals that parameter evaluation accuracy decreases rapidly as the magnitude of random noise contained in the data increases. The values of the more important stability parameters recovered from data contaminated with 3% random noise seem partially acceptable with the possible exception of L_p, L_β, and N_β. However, the coefficients extracted from data containing 5% (time histories in figure 14) and 10% noise are totally unacceptable as table 2 indicates. Therefore, any instrument used to measure aircraft response must induce less than 3% noise or else extensive data smoothing is mandatory.

One then concludes that even though a good fit of the contaminated data seems to have been obtained, the parameter values may be unreliable. Therefore, even data with noise which is "well-behaved", meaning normally

Derivative	Actual	0% random noise	3% random noise	5% random noise	10% random noise
L_p	-12.45	-11.8	-14.76	-16.72	-20.54
L_r	2.54	2.56	2.47	2.26	.953
L_β	-28.77	-27.3	-33.2	-37.04	-45.28
L_{δ_r}	4.75	4.6	4.13	3.33	-.482
N_p	-.372	-.62	-2.87	-2.87	-12.84
N_r	-1.26	-1.27	-1.22	-1.23	-2.02
N_β	10.07	9.49	4.40	.0443	-17.15
N_{δ_r}	-10.25	-10.2	-10.2	-10.44	-12.8
Y_v	-.146	-.146	-.167	-.1825	-.244

Table 2. Effect of random noise on coefficient values.

distributed with zero mean, must be conditioned to smooth out the noise before any analysis is performed; otherwise, the validity of the results must be questioned.

The effect on parameter evaluation of biased instrument readings was then considered. This problem arises when a particular instrument, for example the gyro measuring p, is in error by a constant amount, called a bias. This type error is normally the result of misalignment or failure to initially null the instrument to zero. A high signal-to-noise ratio is probably the easiest and most efficient way of minimizing the effect of bias errors. This was clearly seen by adding a constant bias increment to the time histories resulting from rudder steps of three and twenty degrees. Because of the high signal-to-noise ratio the coefficients obtained from fitting the twenty degree responses were only slightly affected; however, the bias increments caused significant errors in parameter estimates from the fit of responses to a three degree rudder step. Table 3 provides a comparison of the effect on parameter identification of bias errors in several of the response traces to a three degree rudder step. Examination of table 3 reveals that the effect of constant bias increments is heavily dependent upon the response variable in error. For example a bias of $3^o/sec^2$ in $\dot{r}$ has negligible effect on the coefficients except for Y_v; whereas a bias of $3^o/sec$ in r creates large errors in most of the parameters. The table indicates that L_r, L_{δ_r}, N_p, and Y_v are more sensitive than the other parameters to a bias error in p. Likewise, a bias error in ϕ seems more influential in the determination of L_r, N_p, and Y_v. Bias errors in β give rise to large discrepancies in L_{δ_r}, N_{δ_r}, and Y_v. In considering the effects

Derivative	Actual	Bias error of $3^o/sec$ in p	Bias error of $3^o/sec$ in r	Bias error of 3^o in β	Bias error of 3^o in ϕ	Bias error of $3^o/sec^2$ in $\dot{p}$	Bias error of $3^o/sec^2$ in $\dot{r}$
L_p	-12.45	-10.88	-18.20	-12.6	-13.81	-15.63	-12.78
L_r	2.54	2.10	5.55	2.76	2.04	2.14	2.61
L_β	-28.77	-30.80	-47.43	-29.3	-31.73	-36.07	-29.52
L_{δ_r}	4.75	9.834	1.021	34.1	5.02	4.76	4.87
N_p	-.372	.0294	6.78	-.424	.017	-.384	-.4
N_r	-1.26	-1.43	-3.85	-1.34	-1.09	-1.28	-1.35
N_β	10.07	11.02	31.52	10.02	10.83	10.09	10.02
N_{δ_r}	-10.25	-10.94	-9.82	-20.3	-10.26	-10.28	-10.3
Y_v	-.146	-.178	1.154	-.066	-.314	-.147	-.07

Table 3. Effect of bias error on coefficient values.

of constant bias increments, one should remember that in addition to the control surface derivatives, the most important lateral parameters are L_p, L_r, N_β, and N_r. Therefore, the effect of bias errors on these stability coefficients should be carefully considered; whereas differences in parameters of minor importance, such as N_p, may not significantly affect the theoretical aircraft model. In summary, if instrument bias errors are not removed prior to extraction of the stability parameters, serious discrepancies in calculated coefficients may be present. However, by having previous knowledge of instrument inadequacies, the aerodynamicist can remove the effect of constant bias error when preconditioning the test data or compensate for it during the extraction procedure.

The effect on parameter evaluation of another prevalent instrument error, the simple time lag was considered. These time delays often result from servo or filter characteristics. For demonstrative purposes the basic computed flight test data of aircraft response to a three-degree rudder step, including a time lag of $1/(s + 1)$ in the $\dot{\beta}$ trace was analyzed by the Newton-Raphson technique. Initial values of the stability parameters with errors of ±25% were used as a starting point for the first iteration, and the time histories were errorless except for the time lag of $\dot{\beta}$. Figure 15 presents the $\dot{\beta}$ time history, before contamination by the time lag, the contaminated

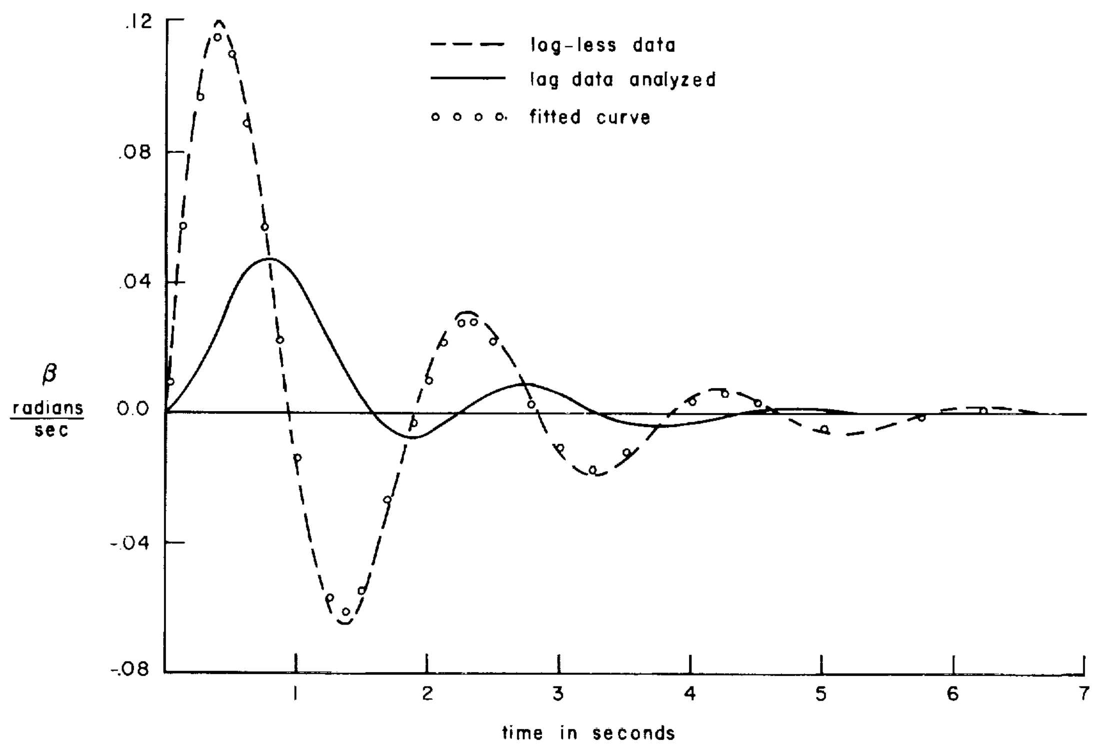

Figure 15. Comparison of time histories resulting from an attempt to fit data containing a time lag.

time history, and the trace obtained when the lag-induced data were analyzed. The results are very intriguing because the time history computed using the extracted coefficients closely matches the original lag-less data rather than that containing the time lag, the data set actually analyzed. Perhaps this oddity becomes more reasonable when one recalls that each of the other response variables were errorless, and thus the effect of a time lag in one variable does not destroy the overall time history match.

The comparison of parameter values presented in table 4 indicates that even though the time lag produces error in the extraction procedure, the resulting stability derivative values are acceptable. For example, with the exception of N_p*, the largest inaccuracy occurred in determining L_p which is within 13% of the actual.

Derivative	Actual	No time lag	$\dot{\beta}$ lag of $\frac{1}{s+1}$
L_p	-12.45	-11.8	-10.9
L_r	2.54	2.56	2.76
L_β	-28.77	-27.3	-26.44
L_{δ_r}	4.75	4.6	4.53
N_p	-.372	-.620	-.774
N_r	-1.26	-1.27	-1.28
N_β	10.07	9.49	8.98
N_{δ_r}	-10.25	-10.2	-9.7
Y_v	-.146	-.146	-.158

Table 4. Effect of time lag coefficient values.

---

* Accurate values of N_p are not to be expected because the input used to excite the aircraft was a rudder step. An aileron input is required to produce responses from which accurate values of this derivative can be extracted.

LONGITUDINAL

For the sake of completeness a description of how the procedure for extracting stability derivatives from flight data would be applied to longitudinal motion is included. While the discussion which follows is similar to that already presented for the lateral mode, applying the extraction procedure to the longitudinal mode is a more complex problem warranting separate investigation.

The linearized longitudinal equations of motion (11) were solved in the Laplace domain,and "flight" time histories were calculated by the method of residues as given in reference 44.

$$
\begin{aligned}
\dot{u} &= X_u u + X_w w + X_q q - 32.2\,\theta + X_{\delta_e}\delta_e, \\
(1 - Z_{\dot{w}})\dot{w} &= Z_u u + Z_w w + (U_o + Z_q)q + Z_{\delta_e}\delta_e, \\
\dot{q} &= M_u u + M_w w + M_{\dot{w}}\dot{w} + M_q q + M_{\delta_e}\delta_e, \\
\dot{\theta} &= q\ .
\end{aligned}
\tag{11}
$$

Values of the dimensional stability derivatives used in equations (11) were those of a typical light aircraft, the Cessna 182. The values of u, w, q, θ, $\dot{u}$, $\dot{w}$, and $\dot{q}$ resulting from an elevator step of one degree were tabulated at intervals of 0.1 seconds for a period of 40 seconds. These responses are plotted as solid curves in the figures showing the fit obtained by the Newton-Raphson technique. Based on the results of a sensitivity analysis presented in reference 1, $Z_{\dot{w}}$ was taken to be zero. Thus, the problem becomes a system of three equations containing thirteen unknown parameters, as shown below:

$$
\begin{aligned}
\dot{u} &= X_u u + X_w w + X_q q - 32.2\,\theta + X_{\delta_e}\delta_e, \\
\dot{w} &= Z_u u + Z_w w + (U_o + Z_q)q + Z_{\delta_e}\delta_e, \\
\dot{q} &= M_u u + M_w w + M_{\dot{w}}\dot{w} + M_q q + M_{\delta_e}\delta_e, \\
\dot{\theta} &= q\ .
\end{aligned}
\tag{12}
$$

Equations (12) were modified to give the same general form used for the lateral solution, equation (9). This modification necessitated eliminating $\dot{w}$ from the right hand side of the $\dot{q}$ equation, removing the dependence of $\dot{q}$ on $\dot{w}$. This is accomplished by substituting $\dot{w}$ (given by the second equation of (12)) into the $\dot{q}$ equation yielding equations of the form:

$$
\begin{aligned}
\dot{u} &= X_u u + X_w w + X_q q - 32.2\,\theta + X_{\delta_e}\delta_e, \\
\dot{w} &= Z_u u + Z_w w + (U_o + Z_q)q + Z_{\delta_e}\delta_e, \\
\dot{q} &= (M_u + Z_u M_{\dot{w}})u + (M_w + M_{\dot{w}} Z_w)w + \left[M_q + M_{\dot{w}}(U_o + Z_q)\right]q + (M_{\delta_e} + Z_{\delta_e} M_{\dot{w}})\delta_e, \\
\dot{\theta} &= q\ .
\end{aligned}
\tag{13}
$$

Equations (13) can be written in matrix form as:

$$\begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} X_u & X_w & X_q & -32.2 \\ Z_u & Z_w & (U_o + Z_q) & 0 \\ (M_u + Z_u M_{\dot{w}}) & (M_w + M_{\dot{w}} Z_w) & (M_q + M_{\dot{w}}(U_o + Z_q)) & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ w \\ q \\ \theta \end{bmatrix} \tag{14}$$

$$+ \begin{bmatrix} X_{\delta_e} \\ Z_{\delta_e} \\ M_{\delta_e} + Z_{\delta_e} M_w \\ 0 \end{bmatrix} \begin{bmatrix} \delta_e \end{bmatrix}$$

Equations (12) had thirteen unknown dimensional stability derivatives. The technique employed to cast the equations into the form required by the computation procedure resulted in combining $M_{\dot{w}}$ with other derivatives; thus, only twelve coefficients can be determined. Another relation must therefore be specified to permit evaluation of the individual derivatives. For light aircraft, the relation

$$Cm_{\delta_e} = - \frac{c}{\ell_t} CL_{\delta_e} \tag{15}$$

is usually used to evaluate Cm_{δ_e}. The corresponding equation for the dimensional derivative is

$$M_{\delta_e} = \frac{\ell_t m}{I_{yy}} Z_{\delta_e} \ . \tag{16}$$

From this relation and the values obtained from the Newton-Raphson method for the twelve coefficients, each of the dimensional stability derivatives can be evaluated.

Even though the procedure indicates that every derivative can be evaluated, problems exist in obtaining reliable values of all the derivatives because the coefficients determined by the $\dot{q}$ equation are functions of two or three derivatives instead of one derivative as in the lateral mode. One such problem encountered is that of obtaining an acceptable value for $M_{\dot{w}}$, a very important derivative in the longitudinal mode. Equation (14) indicates that Z_{δ_e} will be determined. Then using the equation (16) which relates M_{δ_e} and Z_{δ_e}, $M_{\dot{w}}$ can be written as:

$$M_{\dot{w}} = \frac{K}{Z_{\delta_e}} - \frac{\ell_t m}{I_{yy}} \ , \tag{17}$$

where K is the coefficient obtained from the Newton-Raphson technique for the term $(M_{\delta_e} + Z_{\delta_e} M_{\dot{w}})$. Examination of the equation for $M_{\dot{w}}$ reveals that two numbers which are on the order of 1.0 in magnitude must be subtracted to obtain a value of $M_{\dot{w}}$, which itself will be on the order of 0.01 for a light aircraft. Since two large numbers are subtracted to obtain a value which is two orders of magnitude smaller than either of the original numbers, the accuracy achieved in calculating $M_{\dot{w}}$, a very important derivative, may be quite poor.

First, the Newton-Raphson technique was applied to the generated "flight" data using no previous parameter estimates as a starting point for the first iteration or as a priori values. Since the rows of matrix A in equation (10) become dependent when the parameters are identically zero, the initial values were set at ±0.1 depending on whether the actual coefficient was positive or negative to simulate the case of no prior knowledge. Figure 16 denotes that after six iterations no significant improvement of the fit occurred. Figure 17 illustrates the result of this attempt to fit the computed data due to an elevator step of one degree. Convergence occurred and, with the exception of θ (pitch angle), the time histories are closely matched. Figures 16

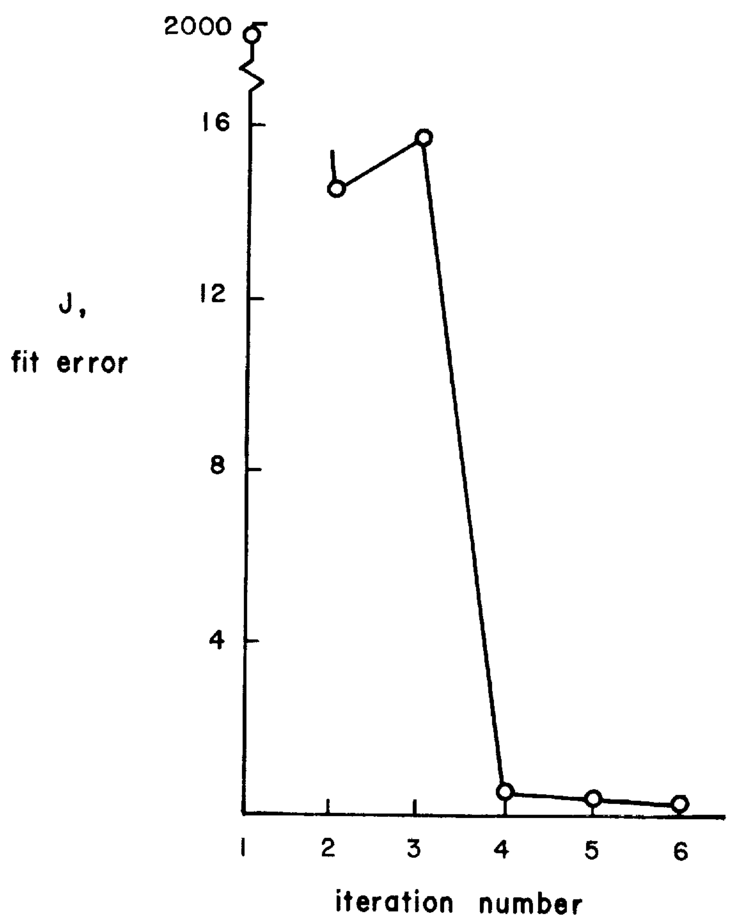

Figure 16. Convergence of fit error for initial values of ±0.1 and no a priori information.

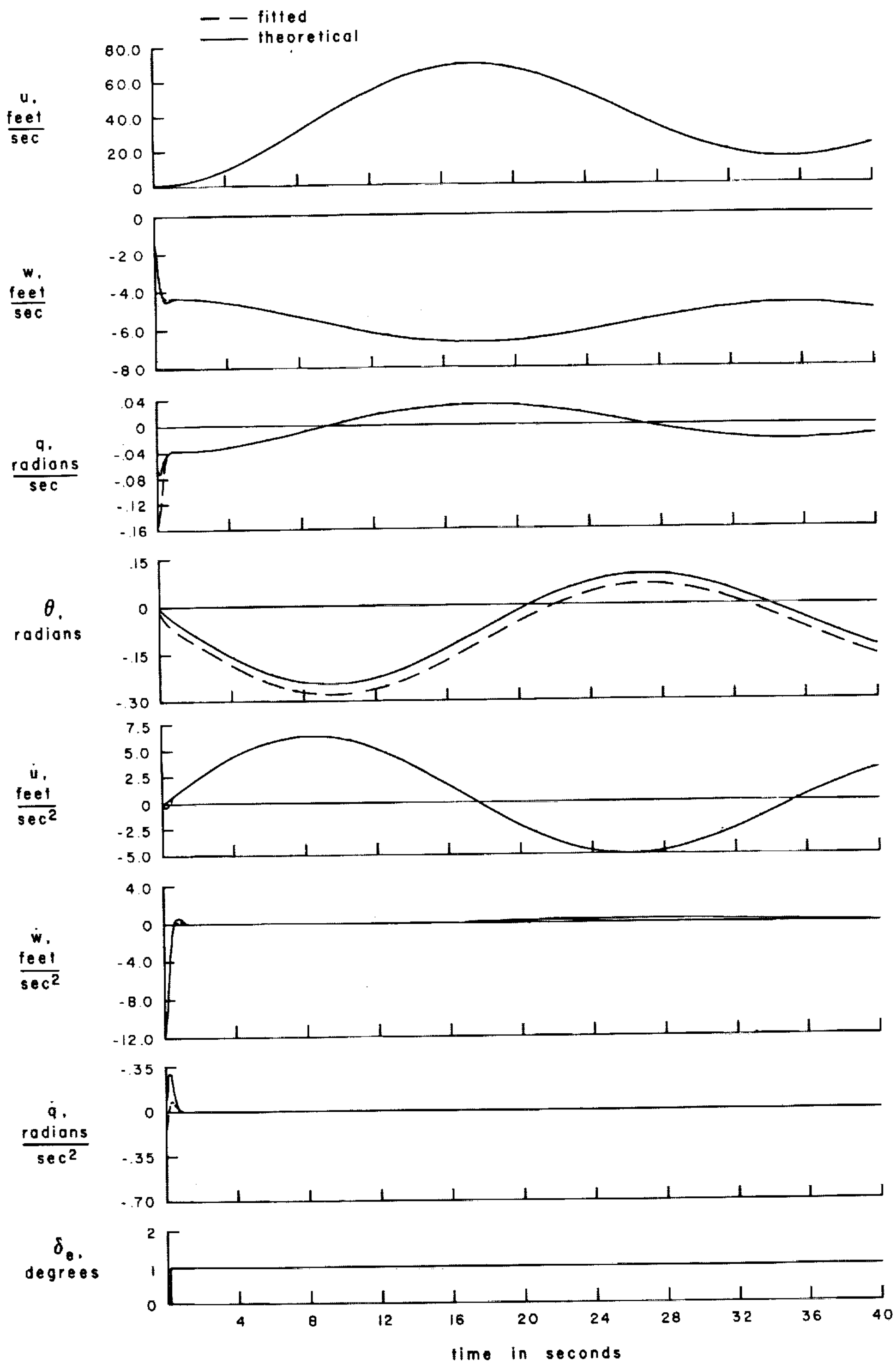

Figure 17. Comparison of time histories resulting from initial values of ±0.1 and no a priori information.

and 17 give the preliminary indication that reliable values for the coefficient were extracted. That this was <u>not</u> the case is revealed by examination of table 5. The coefficient values are totally unacceptable even though convergence was obtained, and the time histories of figure 17 are reasonably well matched. Thus it seems necessary to obtain additional information concerning values of the stability coefficients, to ensure that satisfactory parameters are extracted.

Derivative	Actual	Initial values of ±0.1 No a priori values
X_u	-.0295	-.0241
X_w	.0871	.8675
X_q	0.0	16.31
Z_u	-.2933	-.1132
Z_w	-2.2	-.1652
$U_o + Z_q$	214.5	105.9
$M_u + Z_u M_{\dot{w}}$	.0024	-.0035
$M_w + M_{\dot{w}} Z_w$	-.1066	-.3947
$M_q + M_{\dot{w}} U_o$	-6.024	-9.941
X_{δ_e}	-6.188	167.3
Z_{δ_e}	-44.32	199.9
$M_{\delta_e} + Z_{\delta_e} M_{\dot{w}}$	-39.14	-117.6

Table 5. Comparison of actual longitudinal coefficients with those obtained without prior information.

This prior knowledge of the longitudinal parameters may be obtained from the program listed in Appendix B. This program is a computerization of the theoretical methods for calculating stability derivatives gleaned from the literature and deemed most accurate for each parameter. The reader is directed to reference 1 for a detailed description of these methods. A general discussion of this program was previously included in the presentation of lateral results. With this program to provide prior information concerning coefficient values, the Newton-Raphson technique extracted much more reliable values of the coefficients as is shown below.

First, values of the stability parameters used to compute the "theoretical" flight test data of figure 17 were randomly varied by 25% both positively and negatively and then used as inputs to the Newton-Raphson technique as both a priori and initial values.

A priori values (parameter values used in the error criterion) within 25% of the actual and initial values of ±0.1 were inserted into the Newton-Raphson procedure in an attempt to fit the "flight" test data of figure 17. The fit error (see figure 18) indicates that convergence was obtained. No significant improvement in the fit occurred after eight iterations. However, the sample time histories of figure 19 show some deviation between the actual and the fitted traces.

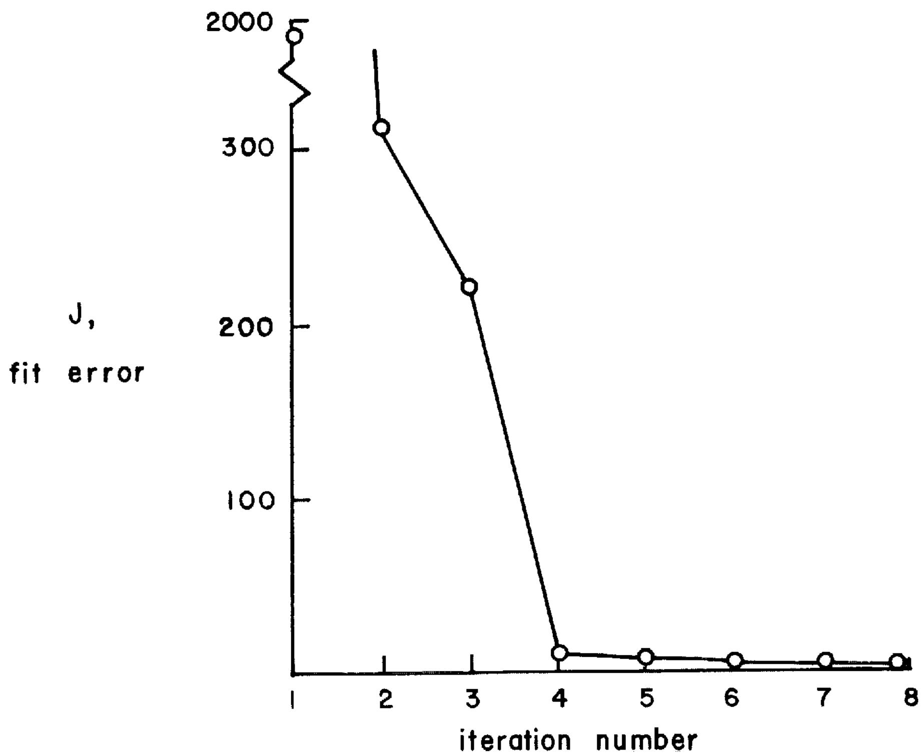

Figure 18. Convergence of fit error for initial values of ±0.1 and a priori values in error by ±25%.

Next, a fit of the data in figure 17 was attempted using zero a priori values and initial values within ±25% of the actual as a starting point . for the first iteration. The examples in figure 20 show that the simulated test data was matched very closely. Figure 21 of fit error, J, indicates that the response variables not included as examples in figure 20 were matched equally well. A comparison of figures 18 and 21 indicates a reduction in fit error of approximately three orders of magnitude when the same prior knowledge of the stability coefficients is inserted into the computational scheme as initial values rather than as a priori values. The final decision, as to which method of using prior information is more benefical, was based on a comparison of parameter evaluation accuracies presented in table 6.

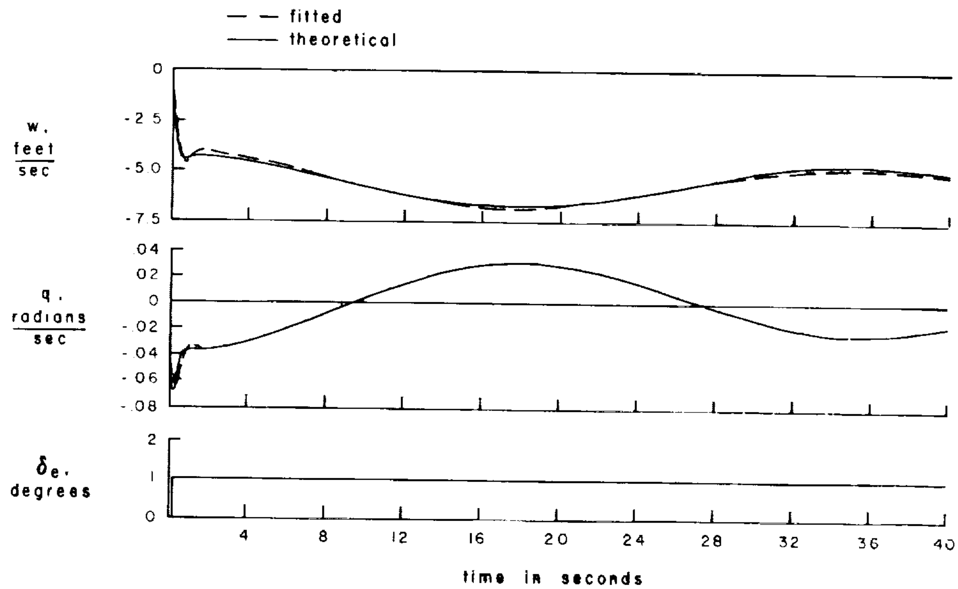

Figure 19. Comparison of example time histories resulting from initial values of ±0.1 and a priori values in error by ±25%.

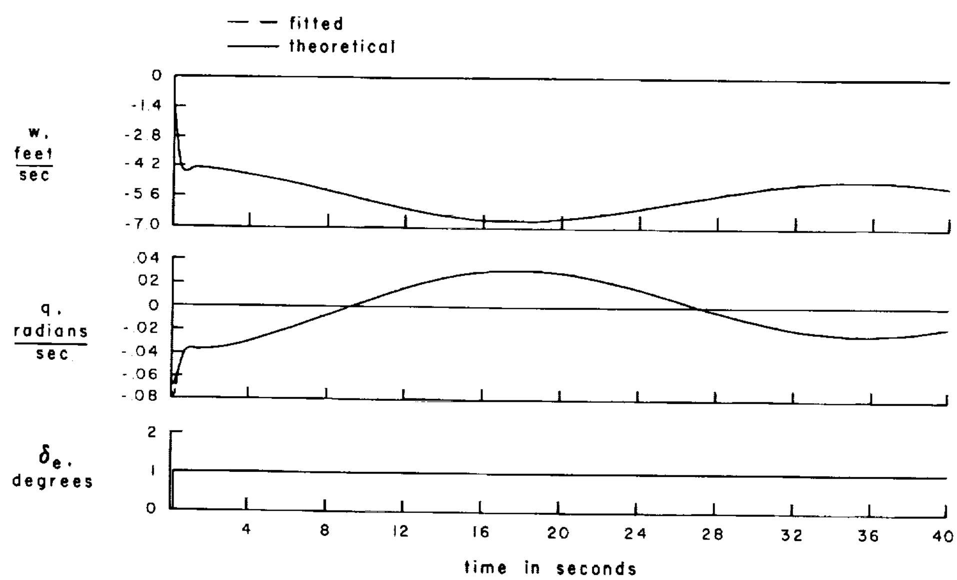

Figure 20. Comparison of example time histories resulting from initial values in error by ±25% and no a priori information.

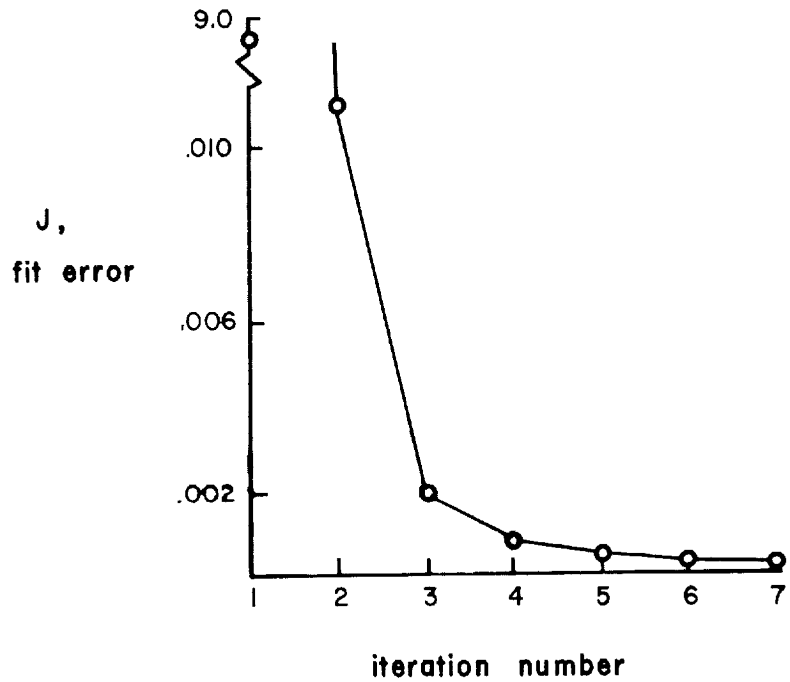

Figure 21. Convergence of fit error for initial values in error by ±25% and no a priori information.

The values in the second column were those used to generate the computed flight test data. Consequently, they represent values of the coefficients which the Newton-Raphson technique attempts to recover. The third column depicts parameter values obtained when the actual coefficients were randomly varied by ±25% and inserted in the computer program as a priori values. The fourth column lists values obtained when these same initial estimates were inserted into the computational routine as a starting point for the first iteration. This table indicates that best results are obtained when theoretical estimates are inserted in the extraction procedure as initial values rather than as a priori values.

In application of this procedure to actual flight data, the programs given in Appendix B would provide theoretical predictions of all the stability derivatives for the particular aircraft and flight condition. These derivatives are then used as initial estimates in the extraction procedure.

The stability derivative sensitivity analysis presented in reference 1 deemed Z_w, M_q, M_w, $M_{\dot{w}}$ as the stability derivatives most influential in determining longitudinal motions. An investigation of column four in table 6 shows that even with initial parameter estimates within 25% of actual, large errors may exist in several of the more important parameters. Therefore, it seems necessary either to approximate the longitudinal coefficients with less than 25% error initially or to reduce the number of unknowns to be determined.

Derivative	Actual	Initial values of ±0.1 A priori values within ±25%	Initial values within ±25% No a priori values
X_u	-.0295	-.02884	-.02916
X_w	.0871	.1337	.1188
X_q	0.0	-.4997	.2681
Z_u	-.2933	-.3030	-.2417
Z_w	-2.2	-2.123	-1.745
$U_o + Z_q$	214.5	218.2	179.2
$M_u + Z_u M_{\dot{w}}$	.0024	-.0001826	.001882
$M_w + M_{\dot{w}} Z_w$	-.1066	-.0698	-.1391
$M_q + M_{\dot{w}} U_o$	-6.024	-2.830	-6.587
X_{δ_e}	-6.188	3.876	-5.317
Z_{δ_e}	-44.32	6.187	-15.19
$M_{\delta_e} + Z_{\delta_e} M_{\dot{w}}$	-39.14	-22.42	-48.35

Table 6. Effect of initial and a priori values on longitudinal coefficients.

It may be noted that in general as the difference between the number of unknown parameters and the number of equations increases, the more non-unique the solutions become. For example, the more the order of a polynomial used to fit a given data set exceeds the number of data points, the more freedom one has in choosing the coefficients. In the lateral case one attempts to extract nine coefficients from three equations. Here, one tries to recover the proper values for twelve coefficients from three equations. It is not surprising, therefore, that even with error-less data it was more difficult to recover the longitudinal parameters accurately.

The effect of data containing random noise on evaluation of longitudinal stability parameters was considered. In a manner similar to that used in studying noise effects on the recovery of lateral parameters, the theoretical data of figure 17 was contaminated with 5% random noise having zero mean and standard deviation of unity. Then using initial values within ±25% as a starting point for the first iteration, an attempt was made to fit this data with the Newton-Raphson technique. Figure 22 illustrates the closeness with which the noisy data was matched after ten iterations. A good match of the time histories was achieved with the exception of slight deviations in

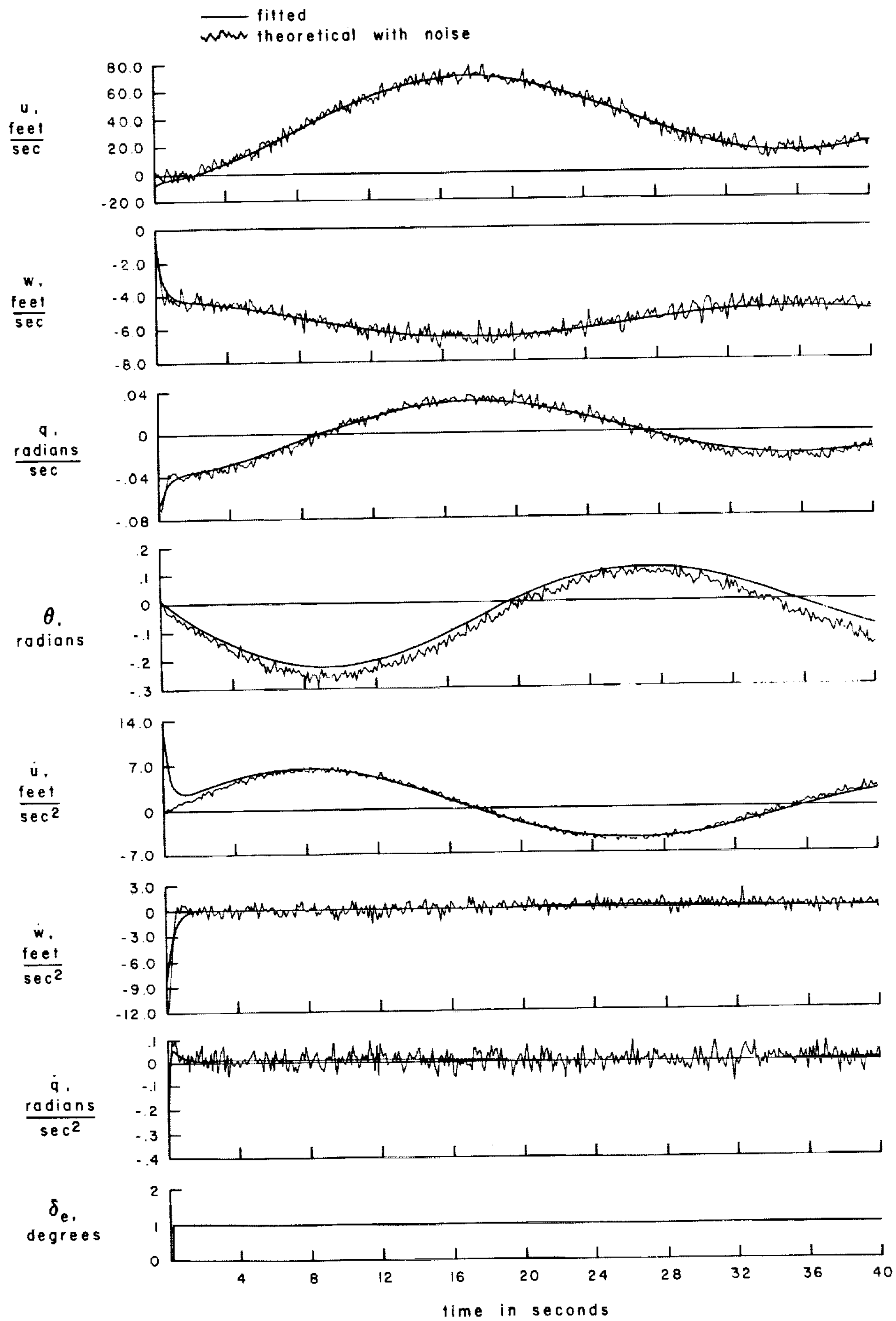

Figure 22. Comparison of time histories resulting from an attempt to fit data containing 5% random noise.

the traces of θ and $\dot{u}$. However, such large errors were obtained when evaluating the stability coefficients, that no further investigation of noise effects on longitudinal parameters was attempted.

CONCLUDING REMARKS

The application of a modified Newton-Raphson technique to the problem of obtaining both lateral and longitudinal stability derivatives of a typical light airplane from flight test data has been presented. For rapid convergence to reliable values, it is important to use initial extimates of the derivatives which will closely approximate those possessed by the aircraft. Computer programs which will give acceptable initial estimates for both the longitudinal and lateral stability derivatives of a light aircraft are listed in Appendix B. Application of the Newton-Raphson technique was found to give good results when the initial estimates of the lateral stability derivative values were within 25% of their actual value; however, even more accurate estimates are necessary to obtain good results from the longitudinal mode.

Because no unique set of stability derivatives can be determined from a situation with more unknowns than equations, such as exists here, the most effective use of the technique requires the exercise of more judgment than one would wish and makes its use by the inexperienced somewhat less than routine. As one might expect, convergence will be easier to obtain as the number of stability derivatives to be recovered in reduced, thus giving a more determinant system of equations. In the previous analysis, Y_p, Y_r, Y_{δ_a}, Y_{δ_r}, and $Z_{\dot{w}}$ were assigned values because, in general, aircraft motion is relatively insensitive to variations in these particular derivatives (ref. 1). Further study of the sensitivity analysis indicates that it may also be practical to assume theoretical values for other derivatives, yielding a smaller number of derivatives to be determined. In reference 1 it was found that the derivatives which proved to be of major importance for the longitudinal analysis were Z_w, M_w, $M_{\dot{w}}$, and M_q, while for the lateral analysis they were N_β, N_r, L_β, and L_p. Therefore, theoretically estimating derivatives such as X_u, Z_u, M_u, X_w, $X_{\dot{w}}$, and Z_q for the longitudinal mode will reduce the number of parameters to be identified and improve convergence. For the lateral mode, if the input is predominately due to a rudder deflection, a theoretical value of N_p would reduce the number of unknown derivatives; however, if the input is dominated by an aileron deflection, a theoretical value of L_r may be assumed.

The program described makes it possible to favor in the extraction procedure those measurements or maneuvers deemed to be more reliable. However, information from other sources--insturment calibrations, previous experience, etc.--must be used to take advantage of this flexibility. For the studies conducted here, all quantities were taken to be equally important.

REFERENCES

1. Smetana, Frederick O.; Summey, Delbert C.; and Johnson, W. Donald: "Riding and Handling Qualities of Light Aircraft--A Review and Analysis". NASA CR-1975, 1972.

2. Hoak, D. E.; and Ellison, D. E.: "USAF Stability and Control Datcom". October 1960 (Rev. August 1968).

3. Soule, Hartley A.; and Wheatley, John B.: "A Comparison Between the Theoretical and Measured Longitudinal Stability Characteristics of an Airplane". NACA TR-442, 1932, 16 pages.

4. Greenberg, Harry: "A Survey of Methods for Determining Stability Parameters of an Airplane from Dynamic Flight Measurements". NACA TN-2340, April 1951, 36 pages.

5. Milliken, W. F., Jr.: "Dynamic Stability and Control Research". Cornell Aeronautical Laboratory Report, CAL-39, September 3-14, 1951, 58 pages.

6. Shinbrot, Marvin: "A Least Squares Curve Fitting Method with Applications to the Calculation of Stability Coefficients from Transient-Response Data". NACA TN-2341, April 1951, 52 pages.

7. Briggs, Benjamin R.; and Jones, Arthur L.: "Techniques for Calculating Parameters of Nonlinear Dynamic Systems from Response Data". NACA TN-2977, July 1953, 67 pages.

8. Shinbrot, Marvin: "A Description and a Comparison of Certain Nonlinear Curve-Fitting Techniques, with Applications to the Analysis of Transient-Response Data". NACA TN-2622, February 1952, 41 pages.

9. Shinbrot, Marvin: "An Analysis of the Errors in Curve-Fitting Problems with an Application to the Calculation of Stability Parameters from Flight Data". NACA TN-2820, November 1952, 29 pages.

10. Shinbrot, Marvin: "On the Analysis of Linear and Nonlinear Dynamical Systems from Transient-Response Data". NACA TN-3288, December 1954, 51 pages.

11. Burns, B. R. A.: "Experience with Shinbrot's Method of Transient Response. Analysis for the Extraction of Stability and Control Derivatives". AGARD Flight Mechanics Specialists Meeting--Stability and Control, September 1966, 43 pages.

12. Donegan, James J.; and Pearson, Henry A.: "Matrix Method of Determining the Longitudinal-Stability Coefficients and Frequency Response of an Aircraft from Transient Flight Data". NACA TR-1070, 1952, 11 pages.

13. Donegan, James J.: "Matrix Methods for Determining the Longitudinal-Stability Derivatives of an Airplane from Transient Flight Data". NACA TR-1169, 1954, 19 pages.

14. Donegan, James J.; Robinson, Samuel W., Jr.; and Gates, Ordway B., Jr.: "Determination of Lateral-Stability Derivatives and Transfer-Function Coefficients from Frequency-Response Data for Lateral Motions". NACA TR-1225, 1955, 19 pages.

15. Eggleston, John M.: "A Method of Deriving Frequency-Response Data for Motion of the Center of Gravity from Data Measured on an Aircraft at Locations other than the Center of Gravity". NACA TN-3021, October, 1953, 25 pages.

16. Sternfield, L.: "A Vector Method Approach to the Analysis of the Dynamic Lateral Stability of Aircraft". Journal of Aeronautical Science, Vol. 21, April 1954, pp. 251-256.

17. Eggleston, John M.; and Mathews, Charles W.: "Application of Several Methods for Determining Transfer Functions and Frequency Response of Aircraft from Flight Data". NACA TR-1204, 1954, 17 pages.

18. Wilkie, Lloyd E.: "Final Report on Extraction of Aircraft Stability Coefficients from Flight Test Data and Theoretical and Experimental Studies on Selected Problems of High Speed Aerodynamics and Dynamic Stability". WADC Technical Report 57-723, December 1957, 132 pages.

19. Eckhart, Franklin F.; and Harper, Robert P., Jr.: "Analysis of Longitudinal Responses of Unstable Aircraft". Cornell Aeronautical Laboratory Report, CAL Report No. BA-1610-F-1, September 30, 1964, 81 pages.

20. Orlik-Rückemann, K: "Methods of Measurement of Aircraft Dynamic Stability Derivatives". National Research Laboratories Report LR-254, July 1959, 57 pages.

21. Tufts, Orren B.: "Dynamic Derivative Measurements in a Wind Tunnel". Institute of the Aerospace Sciences, SMF Fund Paper FF-35, 1963, pp. 45-48.

22. Rampy, Johnny M.; and Berry, Donald T.: "Determination of Stability Derivatives from Flight Test Data by Means of Operation Analog Matching". 11th Annual Air Force Science and Engineering Symposium, Brooks Air Force Base, Texas, October 20-24, 1964, 17 pages.

23. Wolowicz, Chester H.: "Considerations in the Determination of Stability and Control Derivatives and Dynamic Characteristics from Flight Data". AGARD Report 549 - Part I, 1966, 166 pages.

24. Rubin, Arthur I.; Driban, Stanley; and Miessner, Wayne W.: "Regression Analysis and Parameter Identification". Simulation, July 1967, pp. 39-47.

25. Howard, J.: "The Determination of Lateral Stability and Control Derivatives from Flight Data". Canadian Aeronautics and Space Journal, March 1967, pp. 127-134.

26. Clinkenbeard, Ivan L.; Hill, Richard W.; and Rooker, James L.: "Instrumentation for the Extraction of V/Stol Stability Derivatives". Instrumentation in the Aerospace Industry, Vol. 15, May 1969, pp. 224-231.

27. Dolbin, Benjamin H., Jr.: "A Differential Correction Method for the Identification of Airplane Parameters from Flight Test Data". National Electronics Coference, December 8-10, 1969, pp. 90-94.

28. Larson, Duane B.; and Fleck, John T.: "Quasilinearization Techniques". National Electronics Conference, Chicago, Illinois, Proceedings, December 8-10, 1969, pp. 95-100.

29. Taylor, Lawrence W., Jr.; Iliff, Kenneth W.; and Powers, Bruce G.: "A Comparison of Newton-Raphson and Other Methods for Determining Stability Derivatives from Flight Data". AIAA Third Flight Test Conference, Paper 69-315, March 10-12, 1969, 15 pages.

30. Taylor, Lawrence W., Jr.; and Iliff, Kenneth W.: "A Modified Newton-Raphson Method for Determining Stability Derivatives from Flight Data". 2nd International Conference on Computing Methods in Optimization Problems, September 9-13, 1968, 16 pages.

31. Taylor, Lawrence W.: "A System Identification Using a Modified Newton-Raphson Method--A Fortran Program". NASA TN D-6734, 1972.

32. Chapman, Gary T.; and Kirk, Donn B.: "A Method for Extracting Aerodynamic Coefficients from Free-Flight Data". AIAA Journal, Vol. 8, no. 4, April 1970, pp. 753-758.

33. Rose, R.: "Stability and Control Flight Testing--Some of the Test Instrumentation Requirements". AGARD Flight Test Instrumentation, 1967, pp. 65-75.

34. Burns, B. R. A.: "The Effects of Instrumentation Errors on Stability Derivative Measurements". Flight Test Instrumentation, Vol. III, Pergamon Press, 1964, pp. 23-37.

35. Richardson, Norman R.: "Dynamic and Static Wind-Tunnel Tests of a Flow-Direction Vane". NASA TN D-6193, April, 1971, 21 pages.

36. Smetana, Frederick O.; and Stuart, Jay Wm., Jr.: "A Study of Angle-of-Attack Angle-of-Sideslip Pitot-Static Probes". WADC Technical Report 57-234, January 1957, 113 pages.

37. Smetana, Frederick O.; and Headley, Jack W.: "A Further Study of Angle-of-Attack Angle-of-Sideslip Pitot-Static Probes". WADC Technical Report 57-234, Part II, June 1958, 159 pages.

38. Kearfott Systems Division, General Precision Systems, Inc.: "Gyros Platforms Accelerometers, Technical Information for the Engineer". 7th edition, November 1967, 66 pages.

39. Smetana, Frederick O.: "Investigation of Free-Stream Pressure and Stagnation Pressure Measurement from Transonic and Supersonic Aircraft, Final Report". WADC Technical Report 55-238, April 1958, 95 pages.

40. Rosemount Engineering Company: "Rosemount Engineering Company Bulletin 2661". October 1966, 11 pages.

41. Rosemount Engineering Company: "Rosemount Engineering Company Bulletin 116010". 4 pages.

42. Hill, R. W.; and Bolling, N. F.: "V/Stol Flight Test Instrumentation Requirements for Extraction of Aerodynamic Coefficients". Technical Rep. AFFDL-TR-68-154, Vol. II, April 1969, 68 pages.

43. Hill, R. W.; Clinkenbeard, I. L.; and Bolling, N. F.: "V/Stol Flight Test Instrumentation Requirements for Extraction of Aerodynamic Coefficients". Technical Rep. AFFDL-TR-68-154, Vol. I, December 1968, 317 pages.

44. Clark, Robert N.: Introduction to Automatic Control Systems. John Wiley and Sons, Inc., 1964, 460 pages.

APPENDIX A

SYMBOLS

A	stability matrix
a_n	normal acceleration
B	control matrix
C	wing force parallel to the airplane reference line
C_C	coefficient of wing force parallel to the airplane reference line $\frac{C}{\frac{1}{2}\rho U^2 S}$
C_D	drag coefficient $\frac{D}{\frac{1}{2}\rho U^2 S}$
C_{D_q}	$\frac{\partial C_D}{\partial(\frac{qc}{2U})}$
C_{D_u}	$\frac{U}{2}\frac{\partial C_D}{\partial u}$
C_{D_α}	$\frac{\partial C_D}{\partial \alpha}$
$C_{D_{\dot{\alpha}}}$	$\frac{\partial C_D}{\partial(\frac{\dot{\alpha}c}{2U})}$
$C_{D_{\delta_e}}$	$\frac{\partial C_D}{\partial \delta_e}$
C_L	lift coefficient $(L/\frac{1}{2}\rho U^2 S)$
C_{L_u}	$\frac{U}{2}\frac{\partial C_L}{\partial u}$
C_{L_α}	$\frac{\partial C_L}{\partial \alpha}$
$C_{L_{\dot{\alpha}}}$	$\frac{\partial C_L}{\partial(\frac{\dot{\alpha}c}{2U})}$
$C_{L_{\delta_e}}$	$\frac{\partial C_L}{\partial \delta_e}$

C_ℓ	rolling moment coefficient $(L/\frac{1}{2}\rho U^2 Sb)$
C_{ℓ_p}	$\dfrac{\partial C_\ell}{\partial(\frac{pb}{2U})}$
C_{ℓ_r}	$\dfrac{\partial C_\ell}{\partial(\frac{rb}{2U})}$
C_{ℓ_β}	$\dfrac{\partial C_\ell}{\partial \beta}$
$C_{\ell_{\delta_a}}$	$\dfrac{\partial C_\ell}{\partial \delta_a}$
$C_{\ell_{\delta_r}}$	$\dfrac{\partial C_\ell}{\partial \delta_r}$
C_m	pitching-moment coefficient $(M/\frac{1}{2}\rho U^2 Sc)$
C_{m_q}	$\dfrac{\partial C_m}{\partial(\frac{qc}{2U})}$
C_{m_α}	$\dfrac{\partial C_m}{\partial \alpha}$
$C_{m_{\dot\alpha}}$	$\dfrac{\partial C_m}{\partial(\frac{\dot\alpha c}{2U})}$
$C_{m_{\delta_e}}$	$\dfrac{\partial C_m}{\partial \delta_e}$
C_N	coefficient of wing force normal to the airplane reference line
C_{N_α}	$\dfrac{\partial C_N}{\partial \alpha}$
C_n	yawing-moment coefficient $(N/\frac{1}{2}\rho U^2 Sb)$
C_{n_p}	$\dfrac{\partial C_n}{\partial(\frac{pb}{2U})}$
C_{n_r}	$\dfrac{\partial C_n}{\partial(\frac{rb}{2U})}$
C_{n_β}	$\dfrac{\partial C_n}{\partial \beta}$

$C_{n\delta_a}$	$\frac{\partial C_n}{\partial \delta_a}$
$C_{n\delta_r}$	$\frac{\partial C_n}{\partial \delta_r}$
C_T	thrust coefficient $(T/\frac{1}{2}\rho U^2 S)$
C_{y_p}	$\frac{\partial C_y}{\partial(\frac{pb}{2U})}$
C_{y_r}	$\frac{\partial C_y}{\partial(\frac{rb}{2U})}$
C_{y_β}	$\frac{\partial C_y}{\partial \beta}$
$C_{y\delta_a}$	$\frac{\partial C_y}{\partial \delta_a}$
$C_{y\delta_r}$	$\frac{\partial C_y}{\partial \delta_r}$
c	mean aerodynamic chord
c.g.	airplane center of gravity
D	drag force
g	acceleration due to gravity (32.2 ft/sec^2)
I_{xx}	moment of inertia about the x-axis
I_{yy}	moment of inertia about the y-axis
I_{zz}	moment of inertia about the z-axis
I_{xz}	product of inertia
J	fit error
L	lift or rolling moment
L_p	$\frac{\rho U S b^2}{4 I_{xx}} \frac{\partial C_\ell}{\partial(\frac{pb}{2U})}$
L_r	$\frac{\rho U S b^2}{4 I_{xx}} \frac{\partial C_\ell}{\partial(\frac{rb}{2U})}$

L_v	$\frac{\rho USb}{2I_{xx}} \frac{\partial C_\ell}{\partial \beta}$
L_β	$U_o L_v$
L_{δ_a}	$\frac{\rho U^2 Sb}{2I_{xx}} \frac{\partial C_\ell}{\partial \delta_a}$
L_{δ_r}	$\frac{\rho U^2 Sb}{2I_{xx}} \frac{\partial C_\ell}{\partial \delta_r}$
ℓ_t	length from c.g. to tail quarter chord
M	pitching-moment about the c.g.
M_q	$\frac{\rho USc^2}{4I_{yy}} \frac{\partial C_m}{\partial (\frac{qc}{2U})}$
M_u	$\frac{\rho USc}{I_{yy}} (\frac{U}{2} \frac{\partial C_m}{\partial u} + C_m)$
M_w	$\frac{\rho USc}{2I_{yy}} \frac{\partial C_m}{\partial \alpha}$
$M_{\dot{w}}$	$\frac{\rho Sc^2}{4I_{yy}} \frac{\partial C_m}{\partial (\frac{\dot{\alpha} c}{2U})}$
M_{δ_e}	$\frac{\rho U^2 Sc}{2I_{yy}} \frac{\partial C_m}{\partial \delta_e}$
m	mass in slugs
N	wing force normal to the airplane reference line or yawing moment
N_p	$\frac{\rho USb^2}{4I_{zz}} \frac{\partial C_n}{\partial (\frac{pb}{2U})}$
N_r	$\frac{\rho USb^2}{4I_{zz}} \frac{\partial C_n}{\partial (\frac{rb}{2U})}$
N_v	$\frac{\rho USb}{2I_{zz}} \frac{\partial C_n}{\partial \beta}$
N_β	$U_o N_v$

N_{δ_a}	$\frac{\rho U^2 Sb}{2I_{zz}} \frac{\partial C_n}{\partial \delta_a}$
N_{δ_r}	$\frac{\rho U^2 Sb}{2I_{zz}} \frac{\partial C_n}{\partial \delta_r}$
p	rolling velocity
q	pitching velocity
r	yawing velocity
S	wing area
s	Laplace variable
T	thrust
$(T_{\frac{1}{2}})_{s.p.}$	time in seconds required for absolute value of transient short-period oscillation to damp to one-half amplitude
U	airplane velocity
$\underline{U}$	the control vector
U_o	equilibrium airspeed
u	perturbation from equilibrium airspeed
W	airplane weight
w	perturbation from equilibrium vertical velocity
$\underline{X}$	the computed state vector
X_q	$-\frac{\rho USc}{4m} \frac{\partial C_D}{\partial (\frac{qc}{2U})}$
X_u	$\frac{\rho US}{m}(-\frac{U}{2} \frac{\partial C_D}{\partial u} - C_D)$
X_w	$\frac{\rho US}{2m}(C_L - \frac{\partial C_D}{\partial \alpha})$
$X_{\dot{w}}$	$-\frac{\rho Sc}{4m} \frac{\partial C_D}{\partial (\frac{\dot{\alpha} c}{2U})}$
X_{δ_e}	$-\frac{\rho U^2 S}{2m} \frac{\partial C_D}{\partial \delta_e}$

Y_p	$\frac{\rho USb}{4m}\frac{\partial C_y}{\partial(\frac{pb}{2U})}$
Y_r	$\frac{\rho USb}{4m}\frac{\partial C_y}{\partial(\frac{rb}{2U})}$
Y_v	$\frac{\rho US}{2m}\frac{\partial C_y}{\partial\beta}$
Y_{δ_a}	$\frac{\rho U^2S}{2m}\frac{\partial C_y}{\partial\delta_a}$
Y_{δ_r}	$\frac{\rho U^2S}{2m}\frac{\partial C_y}{\partial\delta_r}$
Z_q	$-\frac{\rho USc}{4m}\frac{\partial C_L}{\partial(\frac{qc}{2U})}$
Z_u	$\frac{\rho US}{m}(-\frac{U}{2}\frac{\partial C_L}{\partial u} - C_L)$
Z_w	$\frac{\rho US}{2m}(-\frac{\partial C_L}{\partial\alpha} - C_D)$
$Z_{\dot{w}}$	$-\frac{\rho Sc}{4m}\frac{\partial C_L}{\partial(\frac{\dot{\alpha}c}{2U})}$
Z_{δ_e}	$-\frac{\rho U^2S}{2m}\frac{\partial C_L}{\partial\delta_e}$
α	angle of attack
β	sideslip angle
δ_a	aileron deflection
δ_e	elevator deflection
δ_r	rudder deflection
ζ_{ph}	phugoid damping ratio
θ	pitch angle
π	3.1416
ρ	density

ϕ	roll angle
ψ	yaw angle
$\omega_{n_{ph}}$	phugoid natural frequency
$\omega_{n_{s.p.}}$	short period natural frequency

A dot over a quantity denotes the time derivative of that quantity.

APPENDIX B

LONGITUDINAL GEOMETRIC PROGRAM

```
C
C     GIVEN VALUES OF THE STABILITY DERIVATIVES AND AIRCRAFT
C     CHARACTERISTICS THIS PROGRAM PERFORMS THE FOLLOWING:
C
C        1) CALCULATE NON-DIMENSIONAL STABILITY DERIVATIVES
C
C        2) CALCULATE DIMENSIONAL STABILITY DERIVATIVES
C
C        3) FORMS THE TRANSFER FUNCTIONS, TF(S)=N(S)/D(S)
C
C        4) SOLVE FOR ROOTS OF D(S) AND N(S)
C
C        5) CALCULATE NATURAL FREQUENCIES, DAMPING RATIOS, TIME TO DAMP
C           TO ONE-HALF AMPLITUDE, AND SETTLING TIME
C
C        6) PRODUCES INFORMATION NEEDED FOR BODE PLOT CONSTRUCTION
C
C
C     THE DERIVATION OF THE EQUATIONS OF MOTION ON WHICH THIS ANALYSIS IS
C     BASED WAS TAKEN FROM 'DYNAMICS OF THE AIRFRAME', BUREAU OF
C     AERONAUTICS REPORT, AE-61-4II.
C
C
C     THE ANALYSIS DESCRIBED ABOVE MUST MEET THE ASSUMPTIONS IMPOSED ON
C     THE EQUATIONS OF MOTION WHEN THEY WERE DERIVED.
C      THESE ASSUMPTIONS ARE:
C
C        1) THE AIRFRAME IS ASSUMED TO BE A RIGID BODY.
C        2) THE EARTH IS ASSUMED TO BE FIXED IN SPACE, AND, UNLESS
C           SPECIFICALLY STATED OTHERWISE, THE EARTH'S ATMOSPHERE IS
C           ASSUMED TO BE FIXED WITH RESPECT TO THE EARTH.
C        3) THE MASS OF THE AIRPLANE IS ASSUMED TO REMAIN CONSTANT FOR
C           THE DURATION OF ANY PARTICULAR DYNAMIC ANALYSIS.
C        4) THE X-Z PLANE IS ASSUMED TO BE A PLANE OF SYMMETRY.
C        5) THE DISTURBANCES FROM THE STEADY FLIGHT CONDITION ARE ASSUMED
C           TO BE SMALL ENOUGH SO THAT THE PRODUCTS AND SQUARES OF THE
C           CHANGES IN VELOCITIES ARE NEGLIGIBLE IN COMPARISON WITH THE
C           CHANGES THEMSELVES.  ALSO, THE DISTURBANCE ANGLES ARE ASSUMED
C           TO BE SMALL ENOUGH SO THAT THE SINES OF THESE ANGLES MAY BE
C           SET EQUAL TO THE ANGLES AND THE COSINES SET EQUAL TO ONE.
C           PRODUCTS OF THESE ANGLES ARE ALSO APPROXIMATELY ZERO AND CAN
C           BE NEGLECTED.  AND, SINCE THE DISTURBANCES ARE SMALL, THE
C           CHANGE IN AIR DENSITY ENCOUNTERED BY THE AIRPLANE DURING ANY
C           DISTURBANCE CAN BE CONSIDERED TO BE ZERO.
C        6) DURING THE STEADY FLIGHT CONDITION, THE AIRPLANE IS ASSUMED
C           TO BE FLYING WITH WINGS LEVEL AND ALL COMPONENTS OF VELOCITY
C           ZERO EXCEPT U SUB O.  W SUB O = 0 BECAUSE THE STABILITY AXES
C           WERE CHOSEN AS THE REFERENCE AXES.
C        7) THE FLOW IS ASSUMED TO BE QUASI-STEADY.
C
C
C      ICHECK(II) IS A SUBSCRIPTED VARIABLE WHICH PERMITS THE PROGRAMMER
C      TO READ IN THE VALUE OF A PARTICULAR STABILITY DERIVATIVE RATHER
C      THAN HAVING THE PROGRAM CALCULATE IT.  ICHECK(1) = 1 INDICATES
C      THAT CL IS READ IN.  EACH OF THE 21 DERIVATIVES HAS A SUBSCRIPT
C      NUMBER:  CD-2, CM-3, CT-4, CLA-5, CDA-6, CMA-7, CLDA-8, CDDA-9,
C      CMDA-10, CLQ-11, CDQ-12, CMQ-13, CLDE-14, CDDE-15, CMDE-16, CLU-17,
C      CDU-18, CMU-19, CTU-20, CTRPM-21.  EACH DERIVATIVE THAT IS READ IN
C      IS LISTED ON A SEPARATE CARD AFTER THE DATA CARD FOR ICHECK(II).
C      RHO - THE DENSITY OF AIR AT THE ALTITUDE WHICH THE AIRPLANE IS
C            FLYING IN SLUGS/FT**3.
C      U - THE SPEED OF THE AIRCRAFT IN FEET/SECOND.
C      MS - MS IS THE MASS OF THE AIRCRAFT IN SLUGS.
C      IYY - THE MOMENT OF INERTIA ABOUT THE Y AXIS IN SLUGS-FT**2.
C      THRUST - AIRCRAFT THRUST IN POUNDS.
C      ZJ - THE PERPENDICULAR DISTANCE FROM THE C.G. TO THE THRUST LINE,
C           POSITIVE FOR THE C.G. ABOVE THE THRUST LINE.
C      GCOSGM AND GSINGM ARE THE PRODUCTS OF THE ACCELERATION DUE TO
C      GRAVITY (ASSUMED = 32.2 FT/SEC**2 FOR THIS ALTITUDE RANGE) AND THE
C      COSINE AND SINE RESPECTIVELY OF THE INITIAL FLIGHT PATH ANGLE,
C      GAMMA, (USUALLY ZERO FOR LEVEL FLIGHT).
C      COSXZ - THE COSINE OF THE ANGLE MADE BETWEEN THE THRUST AXIS AND
C              THE WIND AXIS.
C      SINXZ - THE SINE OF THE ANGLE MADE BETWEEN THE THRUST AXIS AND THE
C              WIND AXIS.
C      S - WING AREA IN FT**2.
C      ST - HORIZONTAL TAIL AREA IN FT**2.
C      B - WING SPAN IN FEET.
C      BT - HORIZONTAL TAIL SPAN IN FEET.
C      EFF - EFFICIENCY OF THE HORIZONTAL TAIL.
C      CHE - MEAN AERODYNAMIC CHORD OF THE ELEVATOR.
C      TR - TAPER RATIO OF THE WING.
C      TRT - TAPER RATIO OF THE HORIZONTAL TAIL.
C      CL2DW - THE 2-D LIFT COEFFICIENT OF THE WING.
C      ITAIL - INCIDENCE ANGLE OF THE HORIZONTAL TAIL.
C      CLA2DW - 2-D LIFT CURVE SLOPE OF THE WING PER DEGREE.
C      CLA2DT - 2-D LIFT CURVE SLOPE OF THE HORIZONTAL TAIL PER DEGREE.
C      CDPIE - AIRPLANE PARASITE DRAG COEFFICIENT.
C      CDA2DW - 2-D DRAG CURVE SLOPE OF THE WING PER RADIAN.
C      ALPHA - WING ANGLE OF ATTACK IN DEGREES.
C      IWING - INCIDENCE ANGLE OF THE WING IN DEGREES.
C      LT - LENGTH FROM THE C.G. TO THE QUARTER-CHORD OF THE HORIZONTAL
C           TAIL IN FEET.
C      LT1 - LENGTH FROM THE WING QUARTER-CHORD TO THE  QUARTER-CHORD OF
C            THE HORIZONTAL TAIL IN FEET.
C      XA - CHORDWISE DISTANCE FROM THE C.G. TO THE WING AERODYNAMIC
C           CENTER IN FEET (POSITIVE FOR THE C.G. BEHIND THE WING A.C.).
C      ZA - VERTICAL DISTANCE FROM THE C.G. TO THE WING AERODYNAMIC CEN-
C           TER IN FEET (POSITIVE FOR THE C.G. BELOW THE WING A.C.).
C      LB - LENGTH OF THE FUSELAGE IN FEET.
C      WFUS - MAXIMUM WIDTH OF THE FUSELAGE IN FEET.
C      XFUS - DISTANCE FROM THE FUSELAGE NOSE TO THE WING QUARTER-CHORD
C             IN FEET.
C      XCG - CHORDWISE DISTANCE FROM THE C.G. TO THE WING QUARTER-CHORD
C            IN FEET (POSITIVE FOR THE C.G. AHEAD OF THE WING QUARTER-
C            CHORD).
C      ZT - PERPENDICULAR DISTANCE FROM THE C.G. TO THE THRUST LINE
C           (POSITIVE FOR THRUST LINE BELOW THE C.G.).
C      SELEV - AREA OF THE ELEVATOR IN FT**2.
C      AR - THE WING ASPECT RATIO.
C      ART - THE HORIZONTAL TAIL ASPECT RATIO.
C      CH - THE WING CHORD.
C      CHT - THE HORIZONTAL TAIL CHORD.
C      TAU - THE CORRECTION FACTOR FOR INDUCED ANGLE OF THE WING.
C      TAU1 - THE CORRECTION FACTOR FOR INDUCED ANGLE OF THE WING FOR TR
C             = 1.0.
C      E1 - THE INDUCED-ANGLE SPAN EFFICIENCY FACTOR OF THE WING.
C      CLAW - THE 3-D WING LIFT CURVE SLOPE PER RADIAN.
C      TAUT - THE CORRECTION FACTOR FOR INDUCED ANGLE OF THE HORIZONTAL
C             TAIL.
C      TAU1T - THE CORRECTION FACTOR FOR INDUCED ANGLE OF THE HORIZONTAL
C              TAIL IF TRT = 1.0.
C      E1T - THE INDUCED-ANGLE SPAN EFFICIENCY FACTOR OF THE HORIZONTAL
C            TAIL.
C      DELTA - THE CORRECTION FACTOR FOR INDUCED DRAG OF THE WING.
C      DELTA1 - THE CORRECTION FACTOR FOR INDUCED DRAG OF THE WING FOR
C               TR = 1.0.
C      E - OSWALD'S SPAN EFFICIENCY FACTOR OF THE WING.
```

```
C        CLW - THE 3-D WING LIFT COEFFICIENT.
C        DW - THE DOWNWASH ANGLE AT THE HORIZONTAL TAIL.
C        ALPHAT - THE ANGLE OF ATTACK AT THE HORIZONTAL TAIL.
C        SRATIO - THE RATIO OF THE ELEVATOR AREA TO HORIZONTAL TAIL AREA.
C        DADDE - THE ELEVATOR EFFICIENCY FACTOR; CHANGE IN THE TAIL ANGLE
C                OF ATTACK DUE TO ELEVATOR DEFLECTION.
C        CLAT - THE 3-D HORIZONTAL TAIL LIFT-CURVE SLOPE PER RADIAN.
C        CLT - THE 3-D HORIZONTAL TAIL LIFT COEFFICIENT NEEDED FOR EQUILI-
C              BRIUM.
C        ALPHAR - THE ANGLE OF ATTACK REQUIRED TO ACHIEVE THE NECESSARY
C                 TAIL LIFT COEFFICIENT.
C        DALPHA - THE EFFECTIVE ANGLE OF ATTACK WHICH MUST BE SUPPLIED BY
C                 THE ELEVATOR TO OBTAIN EQUILIBRIUM.
C        DELTAE - THE ELEVATOR DEFLECTION REQUIRED FOR EQUILIBRIUM.
C        CL - TOTAL 3-D AIRPLANE LIFT COEFFICIENT.
C        CD - TOTAL 3-D AIRPLANE DRAG COEFFICIENT.
C        CM - TOTAL 3-D AIRPLANE PITCHING MOMENT.
C        CT - TOTAL 3-D AIRPLANE THRUST COEFFICIENT.
C        DEDA - THE CHANGE IN DOWNWASH WITH WING ANGLE OF ATTACK.
C        CLA - TOTAL 3-D AIRPLANE LIFT CURVE SLOPE PER RADIAN.
C        CDA - THE 3-D AIRPLANE DRAG CURVE SLOPE PER RADIAN.
C        CMAW - THE CHANGE IN WING PITCHING MOMENT DUE TO ANGLE OF ATTACK
C               PER RADIAN.
C        CMAT - THE CHANGE IN HORIZONTAL TAIL PITCHING MOMENT DUE TO ANGLE
C               OF ATTACK PER RADIAN.
C        XLB - THE RATIO OF THE DISTANCE FROM THE FUSELAGE NOSE TO THE WING
C              QUARTER-CHORD TO THE TOTAL FUSELAGE LENGTH.
C        KF - AN EMPIRICAL FACTOR FOR FUSELAGE OR NACELLE CONTRIBUTIONS TO
C             CMA.
C        CMAFUS - THE CHANGE IN FUSELAGE PITCHING MOMENT DUE TO ANGLE OF
C                 ATTACK PER RADIAN.
C        CMA - TOTAL AIRPLANE CHANGE IN PITCHING MOMENT DUE TO ANGLE OF
C              ATTACK PER RADIAN.
C        CLDA - CHANGE IN AIRPLANE LIFT COEFFICIENT WITH RATE OF CHANGE
C               OF ANGLE OF ATTACK.
C        CDDA - CHANGE IN AIRPLANE DRAG COEFFICIENT WITH RATE OF CHANGE OF
C               ANGLE OF ATTACK.
C        CMDA - CHANGE IN AIRPLANE PITCHING MOMENT COEFFICIENT WITH RATE OF
C               CHANGE OF ANGLE OF ATTACK.
C        CLQ - CHANGE IN AIRPLANE LIFT COEFFICIENT WITH PITCHING RATE.
C        CDQ - CHANGE IN AIRPLANE DRAG COEFFICIENT WITH PITCHING RATE.
C        CMQ - CHANGE IN AIRPLANE PITCHING MOMENT COEFFICIENT WITH PITCH-
C              ING RATE.
C        CLDE2D - THE 2-D ELEVATOR EFFECTIVENESS.
C        CLDE - CHANGE IN AIRPLANE LIFT COEFFICIENT WITH ELEVATOR DEFLEC-
C               TION PER RADIAN.
C        CDDE - CHANGE IN AIRPLANE DRAG COEFFICIENT WITH ELEVATOR DEFLEC-
C               TION PER RADIAN.
C        CMDE - CHANGE IN AIRPLANE PITCHING MOMENT COEFFICIENT WITH ELEVA-
C               TOR DEFLECTION PER RADIAN.
C        CLU, CDU, CMU, CTU - CHANGE IN LIFT, DRAG, PITCHING MOMENT, AND
C                             THRUST COEFFICIENTS WITH PERTURBATION
C                             AIRSPEED.
C        CTRPM - CHANGE IN THRUST COEFFICIENT WITH ENGINE SPEED.
C
C        ALTHOUGH THE PROGRAM DOES NOT CALCULATE THE STABILITY DERIVATIVES
C        DUE TO FLAP DEFLECTION THESE DERIVATIVES CAN BE CALCULATED AND
C        THEN READ INTO THE PROGRAM AS CLDE, CDDE, AND CMDE.
C
C
C        THE FOLLOWING CARDS IMPLY THAT THE PROGRAM IS EXECUTED IN DOUBLE
C        PRECISION.
C
         IMPLICIT REAL*8(A-H,O-Z)
         REAL*8 MS,IYY,MU,MW,MDW,MQ,MIN,NUS,NAS,NTHS,KGAIN,KC,KROOT,KKK,KD,
        $KN,NWS,LT,LT1,IWING,LB,KF,ITAIL
         COMPLEX*16 P,TST
         COMMON   WNSP,ZSP,T12SP,TO5SP,WNP,ZP,T12P,TO5P,WF(21),RRD(10),RRN(
        $10),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),AMPRDB(21),P
        $HOFG(21),KGAIN
         COMMON I
         DIMENSION NUS(5),NAS(5),NTHS(5),DS(6),ROOTR(10),ROOTI(10),C(5),KC(
        $4),QUF1(3),QUF2(3),RR1(2),RR2(2),RI1(2),RI2(2),CC(4),RR(3),COFFI(6
        $),RI(3),XR(3),XI(3),COF(3),RE(2),RIM(2),ROUT(1),COFF(2),KKK(21),NW
        $S(5),ICHECK(21),P(6)
C
         READ(1,100)(ICHECK(II),II=1,21)
     100 FORMAT(21I1)
         READ(1,101)RHO,U,MS,IYY,THRUST,ZJ,GCOSGM,GSINGM,COSXZ,SINXZ,S,ST,B
        $,BT,EFF,CHE,TR,TRT,CL2DW,ITAIL,CLA2DW,CLA2DT,CDPIE,CDA2DW,ALPHA,IW
        $ING,LT,LT1,XA,ZA,LB,WFUS,XFUS,XCG,ZT,SELEV
     101 FORMAT(8F10.5)
         AR=B**2/S
         ART=BT**2/ST
         CH=S/B
         CHT=ST/BT
         IF(TR.LT.0.0.OR.TR.GT.1.0)WRITE(3,200)TR
     200 FORMAT(//,1X,'*****  TR = ',F7.4,' IS OUTSIDE DESIRABLE RANGE OF 0
        $.0 TO 1.0 WHEN CALCULATING TAU OR DELTA  *****',//)
         IF((AR.LT.3.0.AND.TR.EQ.1.0).OR.(AR.GT.12.0.AND.TR.EQ.1.0))WRITE(3
        $,201)AR
     201 FORMAT(//,1X,'*****  AR = ',F7.4,' IS OUTSIDE DESIRABLE RANGE OF 3
        $.0 TO 12.0 FOR TR = 1.0 WHEN CALCULATING TAU1 OR DELTA1  *****',//
        $)
         IF(TR.EQ.1.0)GO TO 1
         TAU=6526.0*TR**13-33936.5*TR**12+55708.3*TR**11-40275.9*TR**10-550
        $22.8*TR**9+134075.0*TR**8-128452.0*TR**7+71152.4*TR**6-24169.6*TR*
        $*5+4910.63*TR**4-541.19*TR**3+27.1966*TR**2-1.31155*TR+0.170212
         E1=1.0/(1.0+TAU)
         GO TO 2
       1 TAU1=0.0000297115*AR**4-0.000811747*AR**3+0.0071717*AR**2-0.002988
        $53*AR+1.07739
         E1=1.0/TAU1
       2 CLAW=(57.3*CLA2DW)/(1.0+CLA2DW*57.3/(3.1416*E1*AR))
         IF(TRT.LT.0.0.OR.TRT.GT.1.0)WRITE(3,202)TRT
     202 FORMAT(//,1X,'***** TRT = ',F7.4,' IS OUTSIDE DESIRABLE RANGE OF 0
        $.0 TO 1.0 WHEN CALCULATING TAUT  *****',//)
         IF((ART.LT.3.0.AND.TRT.EQ.1.0).OR.(ART.GT.12.0.AND.TRT.EQ.1.0))WRI
        $TE(3,203)AR
     203 FORMAT(//,1X,'*****  ART = ',F7.4,' IS OUTSIDE DESIRABLE RANGE OF
        $3.0 TO 12.0 FOR TRT = 1.0 WHEN CALCULATING TAU1T  *****',//)
         IF(TRT.EQ.1.0)GO TO 3
         TAUT=6526.0*TRT**13-33936.5*TRT**12+65708.3*TRT**11-40275.9*TRT**1
        $0-55022.8*TRT**9+134075.0*TRT**8-128452.0*TRT**7+71152.4*TRT**6-24
        $169.6*TRT**5+4910.63*TRT**4-541.19*TRT**3+27.1966*TRT**2-1.31155*T
        $RT+0.170212
         E1T=1.0/(1.0+TAUT)
         GO TO 4
       3 TAU1T=0.0000297115*ART**4-0.000811747*ART**3+0.0071717*ART**2-0.00
        $298853*ART+1.07739
         E1T=1.0/TAU1T
       4 IF(TR.EQ.1.0)GO TO  5
         DELTA=2.48637*TR**6-9.29906*TR**5+14.0692*TR**4-11.1199*TR**3+5.01
        $493*TR**2-1.24262*TR+0.141122
         E=1.0/(1.0+DELTA)
         GO TO 6
       5 DELTA1=-0.0000143113*AR**3+0.0000158924*AR**2+0.0113969*AR+.988661
         E=1.0/DELTA1
```

```
    6 IF(ICHECK(1).EQ.1)GO TO 7
      CLW=CL20W/(1.0+2.0/AR)
      DW=(20.0)*CLW*(((1.0/TR)**0.3)/(AR**0.725))*(3.0*CH/LT1)**0.25
      ALPHAT=ALPHA-IWING+ITAIL-DW
      SRATIO=SELEV/ST
      IF(SRATIO.LT.0.0.OR.SRATIO.GT.0.7)WRITE(3,204)SRATIO
  204 FORMAT(//,1X,'*****  SRATIO = ',F6.4,' IS OUTSIDE DESIRABLE RANGE
     $OF 0.0 TO 0.7 WHEN CALCULATING DADDE  *****',//)
      DADDE=21.7949*SRATIO**5-46.4744*SRATIO**4+36.9347*SRATIO**3-14.259
     $*SRATIO**2+3.70551*SRATIO-0.000057815
      CLAT=(57.3*CLA2DT)/(1.0+CLA2DT*57.3/(3.1416*E1T*ART))
      CLT=CLW*XA*S/(LT*ST*EFF)
      ALPHAR=CLT/(CLAT/57.3)
      DALPHA=ALPHAR-ALPHAT
      DELTAE=DALPHA/DADDE
      CL=CLW+CLT*(ST/S)*EFF
      GO TO 8
    7 READ(1,102)CL
  102 FORMAT(F10.5)
    8 IF(ICHECK(2).EQ.1)GO TO 9
      CD=CDPIE+CL**2/(3.1416*E*AR)
      GO TO 10
    9 READ(1,102)CD
   10 IF(ICHECK(3).EQ.1)GO TO 11
      CM=(THRUST/(0.5*RHO*U**2*S))*(ZT/CH)
      GO TO 12
   11 READ(1,102)CM
   12 IF(ICHECK(4).EQ.1)GO TO 13
      CT=THRUST/(0.5*RHO*U**2*S)
      GO TO 14
   13 READ(1,102)CT
   14 IF(ICHECK(5).EQ.1)GO TO 15
      CLAW=(57.3*CLA2DW)/(1.0+CLA2DW*57.3/(3.1416*E1*AR))
      CLAT=(57.3*CLA2DT)/(1.0+CLA2DT*57.3/(3.1416*E1T*ART))
      DEDA=20.0*(CLAW/57.3)*(((1.0/TR)**0.3)/AR**0.725)*(3.0*CH/LT1)**0.
     $25
      CLA=CLAW
      GO TO 16
   15 READ(1,102)CLA
   16 IF(ICHECK(6).EQ.1)GO TO 17
      CDA=CDA2DW+2.0*CL*CLA/(3.1416*E*AR)
      GO TO 18
   17 READ(1,102)CDA
   18 IF(ICHECK(7).EQ.1)GO TO 19
      CMAW=CLA*((1.0+(2.0*CL/(3.1416*E*AR)))*((ALPHA-IWING)/57.3)+CD/CLA)
     $*(XA/CH)+(2.0*CL/(3.1416*E*AR)-(ALPHA-IWING)/57.3-CL/CLA)*(ZA/CH))
      CMAT=CLAT*(1.0-DEDA)*(ST/S)*(LT/CH)*EFF
      XLB=XFUS/LB
      IF(XLB.LT.0.08.OR.XLB.GT.0.65)WRITE(3,205)XLB
  205 FORMAT(//,1X,'*****  XLB = ',F6.5,' IS OUTSIDE DESIRABLE RANGE OF
     $0.1 TO 0.65 WHEN CALCULATING KF  *****',//)
      KF=-19.6558*XLB**4+50.3465*XLB**3-26.5479*XLB**2+7.47299*XLB-0.464
     $064
      CMAFUS=KF*WFUS*WFUS*LB/(S*CH)
      CMA=CMAW+CMAFUS-CMAT
      GO TO 20
   19 READ(1,102)CMA
   20 IF(ICHECK(8).EQ.1)GO TO 21
      CLDA=2.0*CLAT*DEDA*(LT1/CH)*(ST/S)*EFF
      GO TO 22
   21 READ(1,102)CLDA
   22 IF(ICHECK(9).EQ.1)GO TO 23
      CDDA=0.0
      GO TO 24
   23 READ(1,102)CDDA
   24 IF(ICHECK(10).EQ.1)GO TO 25
      CMDA=-2.0*CLAT*DEDA*(LT/CH)*(LT1/CH)*(ST/S)*EFF
      GO TO 26
   25 READ(1,102)CMDA
   26 IF(ICHECK(11).EQ.1)GO TO 27
      CLQ=2.0*(XCG/CH)*CLA+2.0*(LT/CH)*CLAT*(ST/S)*EFF
      GO TO 28
   27 READ(1,102)CLQ
   28 IF(ICHECK(12).EQ.1)GO TO 29
      CDQ=0.0
      GO TO 30
   29 READ(1,102)CDQ
   30 IF(ICHECK(13).EQ.1)GO TO 31
      CMQ=-2.0*(XCG/CH**2)*DABS(XCG)*CLA-2.0*(LT/CH)**2*CLAT*(ST/S)*EFF
      GO TO 32
   31 READ(1,102)CMQ
   32 IF(ICHECK(14).EQ.1)GO TO 42
      CHECHT=CHE/CHT
      IF(CHECHT.LT.0.0.OR.CHECHT.GT.1.0)WRITE(3,206)CHECHT
  206 FORMAT(//,1X,'*****  CHE/CHT = ',F7.4,' IS OUTSIDE DESIRABLE RANGE
     $ OF 0.0 TO 1.0 WHEN CALCULATING CLDE20  *****',//)
      CLDE20=-0.0580767*(CHE/CHT)**4+0.146101*(CHE/CHT)**3-0.169823*(CHE
     $/CHT)**2+0.194084*(CHE/CHT)-0.00202833
      IF(ART.LT.0.0.OR.ART.GT.10.0)WRITE(3,207)ART
  207 FORMAT(//,1X,'*****  ART = ',F7.4,' IS OUTSIDE DESIRABLE RANGE OF
     $0.0 TO 10.0 WHEN CALCULATING CLDE  *****',//)
      AD2D=4.65842*(CHE/CHT)**4-10.9873*(CHE/CHT)**3+9.47521*(CHE/CHT)**
     $2-4.09969*(CHE/CHT)-0.0432959
      IF(AD2D.GE.-0.1)GO TO 33
      IF(AD2D.LT.-0.1.AND.AD2D.GE.-0.2)GO TO 34
      IF(AD2D.LT.-0.2.AND.AD2D.GE.-0.3)GO TO 35
      IF(AD2D.LT.-0.3.AND.AD2D.GE.-0.4)GO TO 36
      IF(AD2D.LT.-0.4.AND.AD2D.GE.-0.5)GO TO 37
      IF(AD2D.LT.-0.5.AND.AD2D.GE.-0.6)GO TO 38
      IF(AD2D.LT.-0.6.AND.AD2D.GE.-0.7)GO TO 39
      IF(AD2D.LT.-0.7.AND.AD2D.GE.-0.8)GO TO 40
      AD1=0.0000255224*ART**4-0.00080707*ART**3+0.00935254*ART**2-0.0487
     $466*ART+1.12006
      AD2=1.0
      AD=AD1+(AD2-AD1)*(AD2D-(-0.8))/(-0.2)
      GO TO 41
   33 AD=-0.0000580764*ART**5+0.00203744*ART**4-0.0284903*ART**3+0.20378
     $3*ART**2-0.802715*ART+2.77199
      GO TO 41
   34 AD1=-0.0000580764*ART**5+0.00203744*ART**4-0.0284903*ART**3+0.2037
     $83*ART**2-0.802715*ART+2.77199
      AD2=-0.000858209*ART**3+0.0224724*ART**2-0.206709*ART+1.81719
      AD=AD1+(AD2-AD1)*(AD2D-(-0.1))/(-0.1)
      GO TO 41
   35 AD1=-0.000858209*ART**3+0.0224724*ART**2-0.206709*ART+1.81719
      AD2=0.000132994*ART**4-0.00394386*ART**3+0.0450323*ART**2-0.249244
     $*ART+1.69816
      AD=AD1+(AD2-AD1)*(AD2D-(-0.2))/(-0.1)
      GO TO 41
   36 AD1=0.000132994*ART**4-0.00394386*ART**3+0.0450323*ART**2-0.245244
     $*ART+1.69816
      AD2=-0.00000724351*ART**6+0.000235239*ART**5-0.00285365*ART**4+0.0
     $150882*ART**3-0.0214656*ART**2-0.108685*ART+1.47429
      AD=AD1+(AD2-AD1)*(AD2D-(-0.3))/(-0.1)
      GO TO 41
   37 AD1=-0.00000724351*ART**6+0.000235239*ART**5-0.00285365*ART**4+0.0
     $150882*ART**3-0.0214656*ART**2-0.108685*ART+1.47429
      AD2=-0.0000116B1*ART**5+0.000430597*ART**4-0.00621814*ART**3+0.045
```

```
      $6723*ART**2-0.184917*ART+1.41226
       AD=AD1+(AD2-AD1)*(AD2D-(-0.4))/(-0.1)
       GO TO 41
   38 AD1=-0.000011681*ART**5+0.000430597*ART**4-0.00621814*ART**3+0.045
      $6723*ART**2-0.184917*ART+1.41226
       AD2=-0.0000073708*ART**5+0.000307952*ART**4-0.00489224*ART**3+0.03
      $80384*ART**2-0.154526*ART+1.32116
       AD=AD1+(AD2-AD1)*(AD2D-(-0.5))/(-0.1)
       GO TO 41
   39 AD1=-0.0000073708*ART**5+0.000307952*ART**4-0.00489224*ART**3+0.03
      $80384*ART**2-0.154526*ART+1.32116
       AD2=0.0000243494*ART**4-0.000255415*ART**3+0.0057768*ART**2-0.048
      $7234*ART+1.16569
       AD=AD1+(AD2-AD1)*(AD2D-(-0.6))/(-0.1)
       GO TO 41
   40 AD1=0.0000243494*ART**4-0.000255415*ART**3+0.0057768*ART**2-0.048
      $7234*ART+1.16569
       AD2=0.0000255244*ART**4-0.00080707*ART**3+0.00935254*ART**2-0.0487
      $466*ART+1.12006
       AD=AD1+(AD2-AD1)*(AD2D-(-0.7))/(-0.1)
   41 CLDE=CLDE2D*(CLAT/(CLA2DT))*AD*(ST/S)*EFF
       CLIN=CLDE
       GO TO 43
   42 READ(1,102)CLDE
       CLIN=CLDE
   43 IF(ICHECK(15).EQ.1)GO TO 44
       CDDE=0.0
       CDIN=CDDE
       GO TO 45
   44 READ(1,102)CDDE
       CDIN=CDDE
   45 IF(ICHECK(16).EQ.1)GO TO 46
       CMDE=(-LT/CH)*CLDE
       CMIN=CMDE
       GO TO 47
   46 READ(1,102)CMDE
       CMIN=CMDE
   47 IF(ICHECK(17).EQ.1)GO TO 48
       CLU=0.0
       GO TO 49
   48 READ(1,102)CLU
   49 IF(ICHECK(18).EQ.1)GO TO 50
       CDU=0.0
       GO TO 51
   50 READ(1,102)CDU
   51 IF(ICHECK(19).EQ.1)GO TO 52
       CMU=0.0
       GO TO 53
   52 READ(1,102)CMU
   53 IF(ICHECK(20).EQ.1)GO TO 54
       CTU=0.0
       GO TO 55
   54 READ(1,102)CTU
   55 IF(ICHECK(21).EQ.1)GO TO 56
       CTRPM=0.0
       GO TO 57
   56 READ(1,102)CTRPM
   57 WRITE(3,208)RHO,U,MS,IYY,THRUST,ZJ,GCOSGM,GSINGM,COSXZ,SINXZ
  208 FORMAT(15(/),9X,114('*')/9X,'*',112X,'*',/9X,'*',39X,'PERTINENT AI
      $RPLANE CHARACTERISTICS',39X,'*',/9X,'*',39X,34('-'),39X,'*',/9X,'*
      $',112X,'*',/9X,'*',14X,'DENSITY (SLUGS/FT**3)',5X,'=',F10.5,10X,'V
      $ELOCITY (FT/SEC)',9X,'=',F10.5,14X,'*',/9X,'*',14X,'MASS (SLUGS)'
      $,14X,'=',F10.5,10X,'IYY (SLUG-FT**2)',10X,'=',F10.5,14X,'*',/9X
      $,'*',14X,'THRUST (POUNDS)',11X,'=',F10.5,10X,'ZJ (FT)',19X,'=',F
      $10.5,14X,'*',/9X,'*',14X,'G*COS(GAMMA) (FT/SEC/SEC)',1X,'=',F10.
      $5,10X,'G*SIN(GAMMA) (FT/SEC/SEC)',1X,'=',F10.5,14X,'*',/9X,'*',1
      $4X,'COS(XZ)',19X,'=',F10.5,10X,'SIN(XZ)',19X,'=',F10.5,14X,'*')
       WRITE(3,209)S,ST,B,BT,CH,CHT,AR,ART,TR,TRT,ALPHA,ALPHAT,IWING,ITAI
      $L
  209 FORMAT(9X,'*',112X,'*',/9X,'*',14X,'WING AREA (FT**2)',9X,'=',F1
      $0.5,10X,'HORZ. TAIL AREA (FT**2)',3X,'=',F10.5,14X,'*',/9X,'*',1
      $4X,'WING SPAN (FT)',12X,'=',F10.5,10X,'HORZ. TAIL SPAN (FT)',6X,
      $'=',F10.5,14X,'*',/9X,'*',14X,'WING CHORD (FT)',11X,'=',F10.5,10
      $X,'HORZ. TAIL CHORD (FT)',5X,'=',F10.5,14X,'*',/9X,'*',14X,'WING A
      $SPECT RATIO',9X,'=',F10.5,10X,'HORZ. TAIL ASPECT RATIO',3X,'='
      $,F10.5,14X,'*',/9X,'*',14X,'WING TAPER RATIO',10X,'=',F10.5,10X,
      $'HORZ. TAIL TAPER RATIO',4X,'=',F10.5,14X,'*',/9X,'*',14X, 'WING A
      $LPHA (DEGREES)',6X,'=',F10.5,10X,'TAIL ALPHA (DEGREES)',6X,'=',F10
      $.5,14X,'*',/9X,'*',14X,'IWING (DEGREES)',11X,'=',F10.5,10X,'ITAIL
      $(DEGREES)',11X,'= ,F10.5,14X,'*')
       WRITE(3,210)DW,DEWA,DELTAE,SELEV,EFF,CHE,CLA2DW,CLA2DT,CDPIE,CDA2D
      $W,CL2DW
  210 FORMAT(9X,'*',14X,'DOWNWASH ANGLE (DEGREES)',2X,'=',F10.5,10X,'DOW
      $NWASH/ALPHA',12X,'=',F10.5,14X,'*',/9X,'*',14X,'ELEVATOR ANGLE (DE
      $GREES)',2X,'=',F10.5,10X,'ELEVATOR AREA (FT**2)',5X,'=',F10.5,14X,
      $'*',/9X,'*',14X,'TAIL EFFICIENCY',11X,'=',F10.5,10X,'ELEVATOR CHOR
      $D (FT)',7X,'=',F10.5,14X,'*',/9X,'*',14X,'2-D WING CLA',14X,'=',
      $F10.5,10X,'2-D TAIL CLA',14X,'=',F10.5,14X,'*',/9X,'*',14X,'CDPIE'
      $,21X,'=',F10.5,10X,'2-D WING CDA',14X,'=',F10.5,14X,'*',/9X,'*',
      $14X,'2-D WING CL',15X,'=',F10.5,61X,'*')
       WRITE(3,211)LB,WFUS,LT,LT1,XA,ZA,XFUS,XCG,ZT
  211 FORMAT(9X,'*',112X,'*',/9X,'*',51X,'DISTANCES',52X,'*',/9X,'*',112
      $X,'*',/9X,'*',6X,'LENGTH OF FUSELAGE (FT)',11X,'=',F10.5,10X,'WIDT
      $H OF FUSELAGE (FT)',12X,'=',F10.5,6X,'*',/9X,'*',6X,'C.G. TO TAIL
      $QUARTER-CHORD (FT)',3X,'=',F10.5,10X,'WING TO TAIL QUARTER-CHORD (
      $FT)'3X,'=',F10.5,6X,'*',/9X,'*',6X,'C.G. TO WING A.C.(CHORDWISE) (
      $FT)', 1X,'=',F10.5,10X,'C.G. TO WING A.C.(VERTICAL) (FT)',2X,'=',F
      $10.5,6X,'*',/9X,'*',6X,'NOSE TO WING QUARTER-CHORD (FT)',3X,'=',F1
      $0.5,10X,'C.G. TO WING QUARTER-CHORD (FT)',3X,'=',F10.5,6X,'*',/9X,
      $'*',6X,'C.G. TO THRUST AXIS (FT)',10X,'=',F10.5,61X,'*',/9X,'*',11
      $2X,'*',/9X,114('*'))
       WRITE(3,212)
  212 FORMAT('1',5(/),2X,128('*'),/,2X,'*',126X,'*',/,2X,'*',46X, 'LONGI
      $TUDINAL STABILITY DERIVATIVES',46X,'*',/,2X,'*',46X,34('-'),46X,'*
      $',/,2X,'*',126X,'*',/,2X,'*',126X,'*')
       WRITE(3,213)CL,CLA,CLDA,CLQ,CLDE,CLU,CT,CD,CDA,CDDA,CDQ,CDDE,CDU,C
      $TU,CM,CMA,CMDA,CMQ,CMDE,CMU,CTRPM
  213 FORMAT(2X,128('*'),/2X,'*',126X,'*',/,2X,'*',4X,'CL =',F8.4,4X,'CL
      $A =',F8.4,4X,'CLDA =',F8.4,4X,'CLQ =',F8.4,4X,'CLDE =',F8.4,4X,
      $'CLU =',F8.4,4X,'CT    =',F8.4,4X,'*',/,2X,'*',4X,'CD ='F8.4,4X,
      $'CDA =',F8.4,4X,'CDDA =',F8.4,4X,'CDQ =',F8.4,4X,'CDDE =',F8.4,4X,
      $'CDU =',F8.4,4X,'CTU   =',F8.4,4X,'*',/,2X,'*',4X,'CM =',F8.4,4X,
      $'CMA =',F8.4,4X,'CMDA =',F8.4,4X,'CMQ =',F8.4,4X,'CMDE =',F8.4,4X
      $,'CMU =',F8.4,4X,'CTRPM =',F8.4,4X,'*',/,2X,'*',126X,'*',/,2X,128(
      $'*'),///)
C
       LCOUNT=0
C
C      LCOUNT IS A COUNTER WHICH PREVENTS MULTIPLE CALCULATIONS OF THE
C      CHARACTERISTIC EQUATION.
C
C      THE READ STATEMENT BELOW IS THE INPUT FOR THE VARIABLES EXPLAINED
C      BELOW.
C
   58 READ(1,103)K,NUMER,ACC
  103 FORMAT(2I3,E14.7)
       IF(NUMER.EQ.0)CALL EXIT
C
```

```
C      CLIN, CDIN, AND CMIN ARE THE STABILITY DERIVATIVES DUE TO CONTROL
C      SURFACE DEFLECTIONS.  CLIN IS EITHER THE PARTIAL OF CL WITH
C      RESPECT TO ELEVATOR DEFLECTION OR FLAP DEFLECTION.  THE VALUE OF K
C      DETERMINES WHETHER IT IS CONCERNED WITH FLAPS OR ELEVATOR.
C
C      IF K IS GIVEN THE VALUE 1 THEN CLIN, CDIN, AND CMIN ARE PARTIAL
C      DERIVATIVES WITH RESPECT TO ELEVATOR DEFLECTION. IF K=2 THEN THE
C      PARTIALS ARE TAKEN WITH RESPECT TO FLAP DEFLECTION.
C
C      NUMER DEFINES WHICH TRANSFER FUNCTION WE ARE INTERESTED IN, (U,
C      ALPHA,THETA,AND W).  NUMER=1 GIVES VARIATIONS IN U, NUMER=2 GIVES
C      VARIATIONS IN ALPHA, NUMER=3 GIVES VARIATIONS IN THETA, AND
C      NUMER=4 GIVES VARIATIONS IN W.
C
C      THUS, FOR EXAMPLE, IF K=2 AND NUMER=2 THE TRANSFER FUNCTION WILL
C      BE VARIATION IN ALPHA DUE TO DEFLECTION OF THE FLAPS.
C
C      ACC IS THE ACCURACY USED THROUGHOUT THE PROGRAM TO COMPARE TWO
C      NUMBERS TO SEE IF THEY ARE SUFFICIENTLY CLOSE.
C
C      IT SHOULD BE NOTICED THAT THE PROGRAM IS SET-UP TO CALCULATE THE
C      DIMENSIONAL STABILITY DERIVATIVES AND THE DENOMINATOR POLYNOMIAL
C      ONLY ONCE; THEREFORE, THE VALUE OF K SHOULD REMAIN CONSTANT FOR
C      A PARTICULAR DATA SET.
C
       LCOUNT=LCOUNT+1
       IF(LCOUNT.GE.2)GO TO 63
       IF(K.NE.1)GO TO 59
       WRITE(3,214)
   214 FORMAT(15X,98('*'),/,15X,'*',96X,'*',/,15X,'*   ',31X,'RESPONSE TO
      $ELEVATOR DEFLECTION',30X,'   *')
       GO TO 60
    59 WRITE(3,215)
   215 FORMAT(15X,98('*'),/,15X,'*',96X,'*',/,15X,'*   ',33X,'RESPONSE TO
      $FLAP DEFLECTION',32X,'   *')
    60 WRITE(3,216)CLIN,CDIN,CMIN,K,ACC
   216 FORMAT(15X,'*',96X,'*',/,15X,'*',96X,'*',/15X,'*   ','CLIN =',F10.6
      $,4X,'CDIN =',F10.6,4X,'CMIN =',F10.6,4X,'K =',I2,4X,'ACC =',F14.8,
      $6X,'*',/,15X,'*',96X,'*',/,15X,98('*'))
C
C      CALCULATION OF DIMENSIONAL STABILITY DERIVATIVES
C
C      D AND DC ARE JUST CONSTANTS USED TO CALCULATE THE DIMENSIONAL
C      STABILITY DERIVATIVES.
C
       D=RHO*U*S/MS
       DC=RHO*U*S*CH/IYY
C
C      DIMENSIONAL STABILITY DERIVATIVES
C
       XU=D*(-(CDU+CD))
       ZU=D*(-(CLU+CL))
       MU=DC*(CMU+CM)
       TU=D*(CTU+CT)
       XW=(D/2.0)*(CL-CDA)
       ZW=(D/2.0)*(-(CLA+CD))
       MW=(DC/2.0)*CMA
       ZDW=-RHO*S*CH*CLDA/(4.0*MS)
       XDW=-RHO*S*CH*CDDA/(4.0*MS)
       MDW=RHO*S*CH*CH*CMDA/(4.0*IYY)
       XQ=-RHO*U*S*CH*CDQ/(4.0*MS)
       ZQ=-RHO*U*S*CH*CLQ/(4.0*MS)
       MQ=DC*CH*CMQ/4.0
       TRPM=30*CH*D*CTRPM
       XIN=-D*CDIN*U/2.0
       ZIN=-D*CLIN*U/2.0
       MIN=DC*CMIN*U/2.0
       IF(LCOUNT.GT.1)GO TO 61
       WRITE(3,217)XU,ZU,MU,TU,XW,ZW,MW,XDW,ZDW,MDW,XQ,ZQ,MQ,TRPM,XIN,
      $ZIN,MIN
   217 FORMAT(///2X,123('*'),/,2X,'*',121X,'*',/,2X,'*',44X,'DIMENSIONAL
      $STABILITY DERIVATIVES',44X,'*',/,2X,'*',121X,'*',/,2X,'*',2X,
      $'  XU =',F10.5,4X,'  ZU =',F10.5,4X,'  MU =',F10.5,4X,'  TU =',
      $F10.5,4X,'  XW =',F10.5,4X,'  ZW =',F10.5,3X,'*',/,2X,'*',3X,
      $ ' MW =',F10.5,4X,' XWD =',F10.5,4X,' ZDW =',F10.5,4X,' MDW =',
      $F10.5,4X,'  XQ =',F10.5,4X,'  ZQ =',F10.5,3X,'*',/,2X,'*',2X,
      $'  MQ =',F10.5,4X,'TRPM =',F10.5,4X,' XIN =',F10.5,4X,' ZIN =',F10
      $.5,4X,' MIN =',F10.5,23X,'*',/,2X,'*',121X,'*',/,2X,123('*'))
C
C      COEFFICIENTS FOR TRANSFER FUNCTION
C
C      A1, A2, A3, A4, A5 ARE CONSTANTS USED TO SIMPLIFY THE CALCULATION
C      OF THE DENOMINATOR AND NUMERATOR COEFFICIENTS.
C
    61 A1=TU*COSXZ+XU
       A2=SINXZ*TU-ZU
       A3=ZJ*MS*TU/IYY+MU
       A4=1.0-ZDW
       A5=ZQ+U
C
C      DENOMINATOR COEFFICIENTS, DS(5)S**4+DS(4)S**3+DS(3)S**2+DS(2)S+
C      DS(1), WHERE S REPRESENTS THE LAPLACIAN OPERATOR.
C
       DS(5)=A4
       DS(4)=-A4*(MQ+A1)-ZW-MDW*A5+XDW*A2
       DS(3)=A1*(MQ*A4+ZW+MDW*A5)-A3*(XDW*A5+XQ*A4)+MQ*ZW-A2*(MQ*XDW-XW-
      $XQ*MDW)+MDW*GSINGM-MW*A5
       DS(2)=GSINGM*(XDW*A3+MW-MDW*A1)+GCOSGM*(-A2*MDW+A3*A4)+A3*(-XW*A5
      $+ZW*XQ)-A2*(-XQ*MW+XW*MQ)+A1*(-MQ*ZW+MW*A5)
       DS(1)=GCOSGM*(MW*(-A2)-ZW*A3)+GSINGM*(XW*A3-MW*A1)
C
C      THE DO LOOP BELOW DETERMINES THE ORDER OF THE POLYNOMIAL IN THE
C      DENOMINATOR, MD.
C
       DO 62 I=1,5
       IF(DABS(DS(I)).GT.ACC)MD=I-1
    62 CONTINUE
       IF(MD.NE.0)GO TO 63
C
C      IF MD = 0, THERE IS NO CHARACTERISTIC EQUATION, THEREFORE THE
C      PROGRAM IS TERMINATED.
C
       CALL EXIT
C
C      NUMERATOR COEFFICIENTS FOR U VARIATION, NUS(4)S**3+NUS(3)S**2+
C      NUS(2)S+NUS(1)
C
    63 NUS(4)=XIN*A4+ZIN*XDW
       NUS(3)=A4*(XQ*MIN-MQ*XIN)+A5*(XDW*MIN-MDW*XIN)-ZW*XIN+ZIN*(XW-XDW
      $*MQ+XQ*MDW)
       NUS(2)=A5*(-MW*XIN+MIN*XW)+GSINGM*(MDW*XIN-XDW*MIN)-GCOSGM*(MDW
      $*ZIN+A4*MIN)+XIN*MQ*ZW+ZIN*(MW*XQ-MQ*XW)-XQ*ZW*MIN
       NUS(1)=GSINGM*(XIN*MW-XW*MIN)+GCOSGM*(ZW*MIN-MW*ZIN)
C
C      NUMERATOR COEFFICIENTS FOR ANGLE OF ATTACK VARIATION, NAS(4)S**3+
C      NAS(3)S**2+NAS(2)S+NAS(1)
C
       NAS(4)=ZIN/U
```

```
      NAS(3)=-ZIN*(A1+MQ)/U+MIN*A5/U+XIN*(-A2)/U
      NAS(2)=MQ*ZIN*A1/U+XIN*A5*A3/U+MIN*XQ*(-A2)/U-ZIN*XQ*A3/U-MIN*A5*A
     $1/U-MIN*GSINGM/U+XIN*MQ*(A2)/U
      NAS(1)=-XIN*GSINGM*A3/U+MIN*GCOSGM*A2/U+ZIN*GCOSGM*A3/U+MIN*GSINGM
     $*A1/U
C
C     COEFFICIENTS FOR NUMERATOR FOR PITCH VARIATION, NTHS(3)S**2+
C     NTHS(2)S+NTHS(1)
C
      NTHS(3)=MIN*A4+ZIN*MDW
      NTHS(2)=A1*(-MIN*A4-ZIN*MDW)+A2*(MIN*XDW-XIN*MDW)+A3*(ZIN*XDW+XIN*
     $A4)-MIN*ZW+ZIN*MW
      NTHS(1)=A1*(MIN*ZW-MW*ZIN)+A2*(MIN*XW-XIN*MW)+A3*(ZIN*XW-XIN*ZW)
C
C     COEFFICIENTS FOR NUMERATOR FOR W VARIATION,  NWS(4)S**3 +
C     NWS(3)S**2 + NWS(2)S + NWS(1)
C
      NWS(4)=NAS(4)*U
      NWS(3)=NAS(3)*U
      NWS(2)=NAS(2)*U
      NWS(1)=NAS(1)*U
      IF(LCOUNT.GT.1)GO TO 64
      WRITE(3,218)DS(5),DS(4),DS(3),DS(2),DS(1),NUS(4),NUS(3),NUS(2),NUS
     $(1),NAS(4),NAS(3),NAS(2),NAS(1),NTHS(3),NTHS(2),NTHS(1),NWS(4),NWS
     $(3),NWS(2),NWS(1)
  218 FORMAT(////2X,123('*'),/,2X,'*',121X,'*',/,2X,'*',32X,'POLYNOMIAL
     $COEFFICIENTS FOR THE DENOMINATOR AND NUMERATOR',32X,'*',/2X,'*',32
     $X,57('-'),32X,'*',/,2X,'*',121X,'*',/,2X,'*',3X,'DS(5)    =',F10.4
     $,5X,'DS(4)    =',F10.4,5X,'DS(3)    =',F10.4,5X,'DS(2)    =',F10.4,5X
     $,'DS(1)    =',F10.4,3X,'*',/,2X,'*',121X,'*',/,2X,'*',27X,'NUS(4)
     $=',F10.4,5X,'NUS(3)   =',F10.4,5X,'NUS(2)   =',F10.4,5X,'NUS(1)   ='
     $,F10.4,3X,'*',/,2X,'*',121X,'*',/,2X,'*',27X,'NAS(4)   =',F10.4,5X,
     $'NAS(3)   =',F10.4,5X,'NAS(2)   =',F10.4,5X,'NAS(1)   =',F10.4,3X,
     $'*',/2X,'*',121X,'*',/,2X,'*',51X,'NTHS(3) =',F10.4,5X,'NTHS(2) ='
     $,F10.4,5X,'NTHS(1) =',F10.4,3X,'*',/,2X,'*',121X,'*',/,2X,'*',27X,
     $'NWS(4)   =',F10.4,5X,'NWS(3)   =',F10.4,5X,'NWS(2)   =',F10.4,5X,
     $'NWS(1)   =',F10.4,3X,'*',/,2X,'*',121X,'*',/,2X,123('*'))
C
C     THE 4 'IF' STATEMENTS BELOW TELL WHICH SET OF NUMERATOR
C     COEFFICIENTS TO USE DEPENDING ON THE VALUE OF NUMER.
C
   64 IF(NUMER.EQ.1)GO TO 65
      IF(NUMER.EQ.2)GO TO 67
      IF(NUMER.EQ.3)GO TO 69
      IF(NUMER.EQ.4)GO TO 71
   65 WRITE(3,219)
  219 FORMAT('1',    130('*'),/,1X,'*',128X,'*',/,1X,'*',128X,'*',/,1X,'*
     $',52X,'SOLUTION FOR U VARIATION',52X,'*',/,1X,'*',52X,24('-'),52X,
     $'*',/,1X,'*',128X,'*',/,1X,'*',128X,'*',/,1X,'*',55X,'DENOMINATOR
     $ROOTS',56X,'*',/,1X,'*',128X,'*')
C
C     THE DO LOOP BELOW DETERMINES THE ORDER OF THE POLYNOMIAL IN THE
C     NUMERATOR, MN.
C
      DO 66 I=1,4
      IF(DABS(NUS(I)).GT.ACC)MN=I-1
   66 CONTINUE
      IF(MN.NE.0)GO TO 73
      WRITE(3,220)MN
  220 FORMAT(1X,'MN=',I3)
      CALL EXIT
   67 WRITE(3,221)
  221 FORMAT('1',    130('*'),/,1X,'*',128X,'*',/,1X,'*',128X,'*',/,1X,'*
     $',45X,'SOLUTION FOR ANGLE OF ATTACK VARIATION',45X,'*',/,1X,'*',45
     $X,38('-'),45X,'*',/,1X,'*',128X,'*',/,1X,'*',128X,'*',/,1X,'*',55X
     $,'DENOMINATOR ROOTS',56X,'*',/,1X,'*',128X,'*')
      DO 68 I=1,4
      IF(DABS(NAS(I)).GT.ACC)MN=I-1
   68 CONTINUE
      IF(MN.NE.0)GO TO 73
      WRITE(3,220)MN
      CALL EXIT
   69 WRITE(3,222)
  222 FORMAT('1',    130('*'),/,1X,'*',128X,'*',/,1X,'*',128X,'*',/,1X,'*
     $',55X,'SOLUTION FOR PITCH VARIATION',          45X,'*',/,1X,'*',55
     $X,28('-'),45X,'*',/,1X,'*',128X,'*',/,1X,'*',128X,'*',/,1X,'*',55X
     $,'DENOMINATOR ROOTS',56X,'*',/,1X,'*',128X,'*')
      DO 70 I=1,3
      IF(DABS(NTHS(I)).GT.ACC)MN=I-1
   70 CONTINUE
      IF(MN.NE.0)GO TO 73
      WRITE(3,220)MN
      CALL EXIT
   71 WRITE(3,223)
  223 FORMAT('1',    130('*'),/,1X,'*',128X,'*',/,1X,'*',129X,'*',/,1X,'*
     $',55X,'SOLUTION FOR   W   VARIATION',          45X,'*',/,1X,'*',55
     $X,28('-'),45X,'*',/,1X,'*',128X,'*',/,1X,'*',129X,'*',/,1X,'*',55X
     $,'DENOMINATOR ROOTS',56X,'*',/,1X,'*',128X,'*')
      DO 72 I=1,4
      IF(DABS(NWS(I)).GT.ACC)MN=I-1
   72 CONTINUE
      IF(MN.NE.0)GO TO 73
      WRITE(3,220)MN
      CALL EXIT
C
C     GETROT IS A SUBROUTINE WHICH, USING OTHER SUBROUTINES, CALCULATES
C     ROOTS, DAMPING RATIOS, AND NATURAL FREQUENCIES, AND THESE ARE
C     TRANSFERRED TO THE MAINLINE BY USE OF A 'COMMON' STATEMENT.
C
   73 CALL GETROT(DS,MD,RRD,RID)
C
C     FOR DAMPING RATIOS GREATER THAN ONE(A NON-OSCILLATORY MODE) THE
C     FOLLOWING FOUR CARDS PREVENT TAKING THE SQUARE ROOT OF A NEGATIVE
C     NUMBER WHEN CALCULATING THE DAMPED NATURAL FREQUENCY.  IF THE
C     DAMPING RATIOS ARE GREATER THAN ONE THEN THE DAMPED NATURAL
C     FREQUENCIES REMAIN 0.0.
C
      WDSP=0.0
      WDP=0.0
      IF(DABS(ZSP).GT.1.0) GO TO 74
      WDSP=WNSP*DSQRT(1.0-ZSP*ZSP)
   74 IF(DABS(ZP).GT.1.0) GO TO 75
      WDP=WNP*DSQRT(1.0-ZP*ZP)
   75 WRITE(3,224)(J,RRD(J),RID(J),J=1,MD)
  224 FORMAT(1X,'*',46X,'ROOT(',I1,') = ',F10.5,'   +J ',F10.5,47X,'*')
      WRITE(3,225)
  225 FORMAT(1X,'*',128X,'*',/,1X,'*',128X,'*',/,1X,'*',34X,'NATURAL FRE
     $Q',5X,'DAMPING RATIO',4X,'TIME FOR 1/2 DAMPING',9X,'SETTLING TIME'
     $,18X,'*',/,1X,'*',31X,'UNDAMPED    DAMPED',80X,'*')
      WRITE(3,226)WNSP,WDSP,ZSP,T12SP,T05SP
  226 FORMAT(1X,'*',18X,'SHORT PERIOD',F9.5,F10.5,2X,F10.5,11X,F10.5,14X
     $,F10.5,22X,'*',/,1X,'*',128X,'*')
      WRITE(3,227)WNP,WDP,ZP,T12P,T05P
  227 FORMAT(1X,'*',18X,'PHUGOID',4X,F10.5,F10.5,2X,F10.5,11X,F10.5,14X,
     $F10.5,22X,'*',/,1X,'*',128X,'*',/,1X,'*',35X,58('-'),35X,'*',/,1X,
     $'*',128X,'*')
C
C     CALCULATION OF VALUES OF WF FOR FUTURE USE IN THE BODE ROUTINE.
```

```
C     INCLUDED ARE SELECTED VALUES OF WF(.01,.1,1.0,10.0,100.0,1000.0)
C     PLUS 5 VALUES AROUND EACH NATURAL FREQUENCY(2 ABOVE, 2 BELOW, AND
C     THE NATURAL FREQUENCY) TO INCREASE DATA IN THE BODE PLOT CRITICAL
C     AREAS.
C
      WF(1)=.01
      DO 76 IJ=2,6
      IM1=IJ-1
      WF(IJ)=WF(IM1)*10.0
   76 CONTINUE
C
C     KWF IS THE NUMBER OF NATURAL FREQUENCIES TO BE USED IN THE BODE
C     ROUTINE.
C
      KWF=6
C
C     II AND IK ARE COUNTERS USED TO DETERMINE THE MAXIMUM VALUE OF KWF
C     DEPENDING ON THE NUMBER OF NATURAL FREQUENCIES IN BOTH THE
C     NUMERATOR AND THE DENOMINATOR OF A PARTICULAR TRANSFER FUNCTION.
C
      II=0
      IK=0
      IF(I.EQ.0)GO TO 78
      IF(I.EQ.2) GO TO 77
      WF(12)=0.9*WNP
      WF(13)=0.75*WNP
      WF(14)=WNP
      WF(15)=1.1*WNP
      WF(16)=1.25*WNP
      IK=1
      KWF=16
   77 WF(7)=.9*WNSP
      WF(8)=.75*WNSP
      WF(9)=WNSP
      WF(10)=1.1*WNSP
      WF(11)=1.25*WNSP
      II=1
      IF(KWF.EQ.16)GO TO 78
      KWF=11
C
C     GETROT IS USED TO FIND ROOTS OF A PARTICULAR NUMERATOR DEPENDING
C     ON THE VALUE OF NUMER.
C
   78 IF(NUMER.EQ.1)GO TO 79
      IF(NUMER.EQ.2)GO TO 80
      IF(NUMER.EQ.4)GO TO 81
      CALL GETROT(NTHS,MN,RRN,RIN)
      GO TO 82
   79 CALL GETROT(NUS,MN,RRN,RIN)
      GO TO 82
   80 CALL GETROT(NAS,MN,RRN,RIN)
      GO TO 82
   81 CALL GETROT(NWS,MN,RRN,RIN)
   82 WRITE(3,228)
  228 FORMAT(1X,'*',56X,'NUMERATOR ROOTS',57X,'*',/,1X,'*',128X,'*')
      WRITE(3,224)(J,RRN(J),RIN(J),J=1,MN)
      WRITE(3,225)
C
C     IF THE DAMPING RATIO HAS AN ABSOLUTE VALUE GREATER THAN ONE(A NON-
C     OSCILLATORY MODE), THEN A DAMPED NATURAL FREQUENCY IS NOT
C     CALCULATED FOR THE NUMERATOR.  THEREFORE, WDSP AND WDP ARE LEFT AS
C     ZERO.
C
      WDSP=0.0
      WDP=0.0
      IF(DABS(ZSP).GT.1.0) GO TO 83
      WDSP=WNSP*DSQRT(1.0-ZSP*ZSP)
   83 IF(DABS(ZP).GT.1.0) GO TO 84
      WDP=WNP*DSQRT(1.0-ZP*ZP)
C
C     THE TWO WRITE STATEMENTS BELOW PRINT THE PERTINENT INFORMATION FOR
C     OSCILLATORY MODES IN THE NUMERATOR.  IF THE NATURAL FREQUENCIES
C     ARE PRINTED AS ZERO THE MODE IS NON-OSCILLATORY.
C
   84 WRITE(3,229)WNSP,WDSP,ZSP,T12SP,T05SP
      WRITE(3,229)WNP,WDP,ZP,T12P,T05P
  229 FORMAT(1X,'*',28X,F9.4,F11.4,3X,F10.4,11X,F10.5,14X,F10.5,22X,'*',
     $/,1X,'*',128X,'*')
      WRITE(3,230)
  230 FORMAT(1X,'*',35X,58('-'),35X,'*',/,1X,'*',128X,'*',/,1X,'*',53X,'
     $BODE PLOT INFORMATION',54X,'*',/,1X,'*',128X,'*',/,1X,'*',21X,'FRE
     $QUENCY',26X,'AMPLITUDE RATIO',25X,'PHASE ANGLE',21X,'*',/,1X,'*',1
     $5X,'RAD/SEC',8X,'CYCLES/SEC',14X,'PURE',10X,'DECIBELS',15X,'RADIAN
     $S',8X,'DEGREES',15X,'*')
C
C     THE NEXT FEW CARDS ARE A ROUTINE TO FIND MORE VALUES OF WF FOR THE
C     BODE PLOT ROUTINE DEPENDING ON WHETHER OR NOT THE NUMERATOR HAS
C     ANY OSCILLATORY MODES.  THE FREQUENCIES AND THE VALUES OF KWF ARE
C     CHOSEN IN THE SAME MANNER AS THOSE OF THE DENOMINATOR PREVIOUSLY
C     MENTIONED.
C
C     MD1 AND MN1 ARE USED TO PREVENT HAVING ZERO SUBSCRIPTS WHEN
C     CALCULATING THE NUMERATOR AND DENOMINATOR GAINS FOR THE BODE PLOT
C     SUBROUTINE.
C
      MD1=MD+1
      MN1=MN+1
      IF(I.EQ.0)GO TO 87
C
C     II AND IK ARE COUNTERS USED TO DETERMINE THE MAXIMUM VALUE OF KWF.
C
      IF(II.EQ.0.AND.IK.EQ.0)GO TO 86
      IF(II.EQ.1.AND.IK.EQ.0)GO TO 85
      WF(17)=.9*WNSP
      WF(18)=.75*WNSP
      WF(19)=WNSP
      WF(20)=1.1*WNSP
      WF(21)=1.25*WNSP
      KWF=21
      GO TO 87
   85 WF(12)=.9*WNSP
      WF(13)=.75*WNSP
      WF(14)=WNSP
      WF(15)=1.1*WNSP
      WF(16)=1.25*WNSP
      KWF=16
      GO TO 87
   86 WF(7)=.9*WNSP
      WF(8)=.75*WNSP
      WF(9)=WNSP
      WF(10)=1.1*WNSP
      WF(11)=1.25*WNSP
      KWF=11
   87 IF(NUMER.NE.1)GO TO 88
C
C     THE GAIN (KGAIN) FOR THE ROOT LOCUS PLOTS IS CALCULATED FROM THE
C     COEFFICIENTS OF THE HIGHEST ORDER TERM IN THE DENOMINATOR AND
C     NUMERATOR, TF=KGAIN(S-A)(S-B)/(S-C)(S-D), WHERE A AND B ARE ROOTS
```

```
C     OF THE NUMERATOR AND C AND D ARE ROOTS OF THE DENOMINATOR.
C
      KGAIN=NUS(MN1)/DS(MD1)
      GO TO 91
   88 IF(NUMER.NE.2)GO TO 89
      KGAIN=NAS(MN1)/DS(MD1)
      GO TO 91
   89 IF(NUMER.NE.3)GO TO 90
      KGAIN=NTHS(MN1)/DS(MD1)
      GO TO 91
   90 KGAIN=NWS(MN1)/DS(MD1)
   91 KKWF=KWF
C
C     THE NEXT 12 CARDS RANK THE WF'S IN ASCENDING ORDER.
C
   92 WAX=WF(1)
      LK=1
      DO 93 JD=2,KKWF
      IF(WF(JD).GE.WAX)LK=JD
      IF(WF(JD).GE.WAX)WAX=WF(JD)
   93 CONTINUE
      WSAV=WF(KKWF)
      WF(KKWF)=WAX
      WF(LK)=WSAV
      KKWF=KKWF-1
      IF(KKWF.EQ.1)GO TO 94
      GO TO 92
C
C     BODE IS THE SUBROUTINE WHICH CALCULATES AMPLITUDE RATIO AND PHASE
C     ANGLE FOR EACH WF.  THE INFORMATION IS TRANSFERRED TO THE MAINLINE
C     BY THE USE OF A 'COMMON' STATEMENT.
C
   94 CALL BODE(MD,MN,KWF)
      WRITE(3,231)(WF(I),WCYCLE(I),AMPR(I),AMPRDB(I),PHASE(I),PHDEG(I),I
     $=1,KWF)
  231 FORMAT(1X,'*',13X,F10.5,6X,F10.5,12X,F10.2,6X,F10.5,12X,F10.5,6X,F
     $10.5,13X,'*')
      WRITE(3,232)
  232 FORMAT(1X,'*',128X,'*',/,1X,130('*'))
      GO TO 58
      END


      SUBROUTINE GETROT(COFFI,M,ROOTR,ROOTI)
C
C     GETROT IS A SUBROUTINE WHICH, USING OTHER SUBROUTINES, CALCULATES
C     ROOTS, DAMPING RATIOS, AND NATURAL FREQUENCIES, AND THESE ARE
C     TRANSFERRED TO THE MAINLINE BY USE OF A 'COMMON' STATEMENT.
C
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 MS,IYY,MU,MW,MDW,MQ,MIN,NUS,NAS,NTHS,KGAIN,KC,KROOT,KKK,KD,
     $KN,NWS
      COMPLEX*16 P,TST
      COMMON    WNSP,ZSP,T12SP,T05SP,WNP,ZP,T12P,T05P,WF(21),RRD(10),RRN(
     $10),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),AMPRDB(21),P
     $HDEG(21),KGAIN
      COMMON I
      DIMENSION NUS(5),NAS(5),NTHS(5),DS(6),ROOTR(10),ROOTI(10),C(5),KC(
     $4),QUF1(3),QUF2(3),RR1(2),RR2(2),RI1(2),RI2(2),CC(4),RR(3),COFFI(6
     $),RI(3),XR(3),XI(3),COF(3),RE(2),RIM(2),ROOT(1),COFF(2),KKK(21),NW
     $S(5),P(6)
C
C     THE 3 'IF' STATEMENTS BELOW DECIDE WHICH ROOT-EXTRACTION
C     SUBROUTINE TO CALL DEPENDING ON THE VALUE OF M(THE ORDER OF THE
C     POLYNOMIAL).
C
      IF(M.EQ.4)GO TO 3
      IF(M.EQ.3)GO TO 2
      IF(M.EQ.2)GO TO 1
C
C     THE SUBROUTINES SINGLE, QUAD, CUBE, AND FOURTH SOLVE (THE SOLUTION
C     IS ACHIEVED IN CLOSED FORM AND THUS REQUIRES NO ITERATIVE
C     PROCEDURE) FOR ROOTS OF FIRST, SECOND, THIRD, AND FOURTH ORDER
C     POLYNOMIALS, RESPECTIVELY.
C
      CALL SINGLE(COFFI,ROOTR,ROOTI)
      GO TO 4
    1 CALL QUAD(COFFI,ROOTR,ROOTI)
      GO TO 4
    2 CALL CUBE(COFFI,ROOTR,ROOTI)
      GO TO 4
    3 CALL FOURTH(COFFI,ROOTR,ROOTI)
C
C     THE FOLLOWING CARDS TEST THE ROOTS OF THE POLYNOMIAL TO CHECK THE
C     ACCURACY OF THE ROOT SOLVER SUBROUTINES.  IF THE VALUE OF TST IS
C     TOO LARGE A WARNING MESSAGE IS PRINTED.
C
    4 DO 6 I=1,M
      P(I)=DCMPLX(ROOTR(I),ROOTI(I))
      ZERO=0.0
      TST=DCMPLX(COFFI(1),ZERO)
      MJ=M+1
      DO 5 J=2,MJ
      TST=TST+COFFI(J)*P(I)**(J-1)
    5 CONTINUE
      IF(CDABS(TST).LE.ACC)GO TO 6
      WRITE(3,100)P(I),TST
  100 FORMAT(1X,'ROOT = ',2G15.8,'    WHEN SUBSTITUTED INTO ITS POLYNOMI
     $AL FAILED TO COME WITHIN ACC OF 0.0.',/,1X,'THIS VALUE DIFFERED FR
     $OM ZERO BY ',2G15.8,'   THIS IMPLIES EITHER A ROUNDOFF ERROR WHEN
     $TESTING THE ROOTS',/,1X,'(ACC TOO SMALL) OR THE VALUE 0.01 USED TO
     $ COMPARE WITH TEST IN SUBROUTINE FOURTH IS TOO LARGE.')
    6 CONTINUE
C
C     I IS A COUNTER WHICH DETERMINES THE NUMBER OF ROOTS WHICH HAVE
C     BOTH A REAL AND AN IMAGINARY PART.
C
C
C     THE NEXT 25 CARDS IS A PROCEDURE WHICH POSITIONS ROOTS WITH BOTH
C     REAL AND IMAGINARY PARTS IN THE FIRST L POSITIONS AND THE ROOTS
C     WITH ZERO IMAGINARY PARTS IN THE NEXT KK POSITIONS.  FOR EXAMPLE
C     IF THERE ARE 4 ROOTS, TWO WITH ONLY REAL PARTS AND TWO COMPLEX,
C     THE COMPLEX ROOTS WILL BE IN POSITIONS 1 AND 2 AND THE REAL ROOTS
C     WILL BE IN POSITIONS 3 AND 4.  L, K, AND KK ARE COUNTERS USED TO
C     FACILITATE THIS PROCEDURE.
C
C     I IS A COUNTER WHICH DETERMINES THE NUMBER OF ROOTS WHICH HAVE
C     BOTH A REAL AND AN IMAGINARY PART.
C
      I=M
C
C     N IS A COUNTER WHICH PREVENTS THE ORDER OF THE POLYNOMIAL FROM
C     BEING DESTROYED.
C
      N=M
      L=1
      K=M+1
```

```
      KK=0
      DO 8 J=1,N
      IF(DABS(ROOTI(J)).GT.ACC)GO TO 7
      I=I-1
      ROOTI(K)=ROOTI(J)
      ROOTR(K)=ROOTR(J)
      K=K+1
      KK=KK+1
      GO TO 8
    7 ROOTI(L)=ROOTI(J)
      ROOTR(L)=ROOTR(J)
      L=L+1
    8 CONTINUE
      IF(KK.EQ.0)GO TO 10
    9 KI=K-KK
      ROOTI(N)=ROOTI(KI)
      ROOTR(N)=ROOTR(KI)
      N=N-1
      KK=KK-1
      IF(KK.EQ.0)GO TO 10
      GO TO 9
C
C     AT THIS POINT THERE ARE I ROOTS THAT HAVE BOTH A REAL AND AN
C     IMAGINARY PART AND (M-I) ROOTS WITH JUST A REAL PART.
C
C     THIS PART OF THE PROGRAM DETERMINES THE LARGEST REAL PART OF THE
C     ROOTS AND RANKS THEM FROM THE BOTTOM IN THE I POSITIONS AVAILABLE.
C
   10 RMAX=ROOTR(1)
      K=1
      IF(I.EQ.0) GO TO 13
      DO 11 J=1,I
      IF(DABS(ROOTR(J)).GT.RMAX)K=J
      IF(DABS(ROOTR(J)).GT.RMAX)RMAX=ROOTR(J)
   11 CONTINUE
      RSAVER=ROOTR(I)
      RSAVEI=ROOTI(I)
      ROOTR(I)=ROOTR(K)
      ROOTI(I)=ROOTI(K)
      ROOTR(K)=RSAVER
      ROOTI(K)=RSAVEI
      N=I-1
      DO 12 J=1,N
      IF(DABS(RMAX-ROOTR(J)).LE.ACC)L=J
   12 CONTINUE
      RSAVER=ROOTR(N)
      RSAVEI=ROOTI(N)
      ROOTR(N)=ROOTR(L)
      ROOTI(N)=ROOTI(L)
      ROOTR(L)=RSAVER
      ROOTI(L)=RSAVEI
C
C     THE OUTPUT FOR BOTH NUMERATOR AND DENOMINATOR IS PRINTED IN A FORM
C     WHICH REQUIRES TWO OSCILLATORY MODES.  IF ONE OR BOTH OF THE MODES
C     ARE NON-OSCILLATORY THEN THE FOLLOWING PROCEDURE IS USED:
C          1)   THE DAMPING RATIO IS CHOSEN TO BE THE SMALLER MAGNITUDE OF
C               THE REAL ROOTS, SINCE THIS ROOT WILL DOMINATE IN THE TIME
C               DOMAIN (A NEGATIVE DAMPING RATIO WOULD INDICATE AN
C               UNSTABLE MODE).
C          2)   THE TIME TO DAMP TO 50% AND 5% OF THE AMPLITUDE ARE
C               CALCULATED BASED ON THE ABOVE DAMPING RATIO.  THUS, FOR AN
C               UNSTABLE SYSTEM THESE TIMES WILL BE NEGATIVE.
C
C     THE REMAINING PORTION OF GETROT CALCULATES THE NATURAL FREQUENCIES
C     (WNP & WNSP), DAMPING RATIOS(ZP & ZSP), TIME TO DAMP TO 1/2
C     AMPLITUDE(T12P & T12SP), AND SETTLING TIME(T05P & T05SP).  THE
C     SETTLING TIME IS THE TIME TO DAMP TO 5% OF THE ORIGINAL AMPLITUDE.
C     THE SUFFIXES P AND SP REFER TO OSCILLATORY MODES FOR THE NUMERATOR
C     OR THE DENOMINATOR DEPENDING ON THE EQUATION BEING SOLVED .
C
C     THE SHORT PERIOD AND PHUGOID NATURAL FREQUENCIES ARE DETERMINED BY
C     A RANKING OF THE MAGNITUDE OF THE REAL AND IMAGINARY PARTS OF THE
C     ROOTS, THE LARGER MAGNITUDE REPRESENTS THE SHORT PERIOD MODE.  IF
C     THERE IS ONLY ONE OSCILLATORY MODE THIS MODE IS REFERRED TO AS THE
C     SHORT PERIOD MODE AND THE PHUGOID MODE NATURAL FREQUENCY IS
C     PRINTED AS ZERO.  WHEN GETROT IS USED FOR A NUMERATOR POLYNOMIAL
C     THE SHORT PERIOD INFORMATION IS PRINTED AS A NUMERATOR OSCILLATORY
C     MODE(SINCE A CUBIC IS THE LARGEST NUMERATOR POLYNOMIAL POSSIBLE,
C     THERE WILL BE ONLY ONE OSCILLATORY MODE AT MOST).
C
   13 IF(M.EQ.1)GO TO 17
      IF(M.EQ.2.AND.I.EQ.0)GO TO 18
      IF(M.EQ.2.AND.I.EQ.2)GO TO 21
      IF(M.EQ.3.AND.I.EQ.0)GO TO 22
      IF(M.EQ.3.AND.I.EQ.2)GO TO 26
      IF(M.EQ.4.AND.I.EQ.0)GO TO 27
      IF(M.EQ.4.AND.I.EQ.2)GO TO 32
      WN1=DSQRT(ROOTR(3)*ROOTR(3)+ROOTI(3)*ROOTI(3))
      WN2=DSQRT(ROOTR(1)*ROOTR(1)+ROOTI(1)*ROOTI(1))
      IF(WN1.GT.WN2)GO TO 14
      WNSP=WN2
      NX=20
      WNP=WN1
      GO TO 15
   14 WNSP=WN1
      NX=10
      WNP=WN2
   15 IF(NX.NE.20)GO TO 16
      ZSP=-ROOTR(1)/WNSP
      T05SP=(2.9957)/(ZSP*WNSP)
      T12SP=(.693147)/(ZSP*WNSP)
      ZP=-ROOTR(3)/WNP
      T12P=(.693147)/(ZP*WNP)
      T05P=(2.9957)/(ZP*WNP)
      GO TO 35
   16 ZSP=-ROOTR(3)/WNSP
      T05SP=(2.9957)/(ZSP*WNSP)
      T12SP=(.693147)/(ZSP*WNSP)
      ZP=-ROOTR(1)/WNP
      T05P=(2.9957)/(ZP*WNP)
      T12P=(.693147)/(ZP*WNP)
      GO TO 35
   17 WNSP=0.0
      WNP=0.0
      ZSP=-ROOTR(1)
      ZP=0.0
      T12SP=.693147/ZSP
      T05SP=2.9957/ZSP
      T05P=0.0
      T12P=0.0
      GO TO 35
   18 IF(DABS(ROOTR(1)).GT.DABS(ROOTR(2)))GO TO 19
      ZSP=-ROOTR(1)
      ZP=-ROOTR(2)
      GO TO 20
   19 ZSP=-ROOTR(2)
      ZP=-ROOTR(1)
   20 WNSP=0.0
```

```
      WNP=0.0
      T12P=.693147/ZP
      T05P=2.9957/ZP
      T12SP=.693147/ZSP
      T05SP=2.9957/ZSP
      GO TO 35
   21 WNP=0.0
      ZP=0.0
      T12P=0.0
      T05P=0.0
      WNSP=DSQRT(ROOTR(1)*ROOTR(1)+ROOTI(1)*ROOTI(1))
      ZSP=-ROOTR(1)/WNSP
      T12SP=.693147/(ZSP*WNSP)
      T05SP=2.9957/(ZSP*WNSP)
      GO TO 35
   22 RAX=ROOTR(1)
      K=1
      DO 23 J=1,3
      IF(DABS(ROOTR(J)).GT.RAX)K=J
      IF(DABS(ROOTR(J)).GT.RAX)RAX=ROOTR(J)
   23 CONTINUE
      RSAV=ROOTR(3)
      ROOTR(3)=RAX
      ROOTR(K)=RSAV
      IF(DABS(ROOTR(1)).GT.DABS(ROOTR(2)))GO TO 24
      ZSP=-ROOTR(1)
      ZP=-ROOTR(2)
      GO TO 25
   24 ZSP=-ROOTR(2)
      ZP=-ROOTR(1)
   25 WNSP=0.0
      WNP=0.0
      T12P=.693147/ZP
      T05P=2.9957/ZP
      T12SP=.693147/ZSP
      T05SP=2.9957/ZSP
      GO TO 35
   26 WNP=0.0
      ZP=-ROOTR(3)
      T12P=.693147/ZP
      T05P=2.9957/ZP
      WNSP=DSQRT(ROOTR(1)*ROOTR(1)+ROOTI(1)*ROOTI(1))
      ZSP=-ROOTR(1)/WNSP
      T12SP=.693147/(ZSP*WNSP)
      T05SP=2.9957/(ZSP*WNSP)
      GO TO 35
   27 RAX=ROOTR(1)
      K=1
      DO 28 J=1,4
      IF(DABS(ROOTR(J)).GT.RAX)K=J
      IF(DABS(ROOTR(J)).GT.RAX)RAX=ROOTR(J)
   28 CONTINUE
      RSAV=ROOTR(4)
      ROOTR(4)=RAX
      ROOTR(K)=RSAV
      RAX=ROOTR(1)
      K=1
      DO 29 J=1,3
      IF(DABS(ROOTR(J)).GT.RAX)K=J
      IF(DABS(ROOTR(J)).GT.RAX)RAX=ROOTR(J)
   29 CONTINUE
      RSAV=ROOTR(3)
      ROOTR(3)=RAX
      ROOTR(K)=RSAV
      IF(DABS(ROOTR(1)).GT.DABS(ROOTR(2)))GO TO 30
      ZSP=-ROOTR(1)
      ZP=-ROOTR(2)
      GO TO 31
   30 ZSP=-ROOTR(2)
      ZP=-ROOTR(1)
   31 WNSP=0.0
      WNP=0.0
      T12P=.693147/ZP
      T05P=2.9957/ZP
      T12SP=.693147/ZSP
      T05SP=2.9957/ZSP
      GO TO 35
   32 WNSP=DSQRT(ROOTR(1)*ROOTR(1)+ROOTI(1)*ROOTI(1))
      ZSP=-ROOTR(1)/WNSP
      WNP=0.0
      T12SP=.693147/(ZSP*WNSP)
      T05SP=2.9957/(ZSP*WNSP)
      IF(DABS(ROOTR(3)).GT.DABS(ROOTR(4)))GO TO 33
      ZP=-ROOTR(3)
      GO TO 34
   33 ZP=-ROOTR(4)
   34 T12SP=(2.9957)/(ZSP*WNSP)
      T05SP=.693147/(ZSP*WNSP)
   35 RETURN
      END


      SUBROUTINE FOURTH(C,ROOTR,ROOTI)
C
C     THIS SUBROUTINE FACTORS A FOURTH ORDER POLYNOMIAL BY A CLOSED FORM
C     PROCEDURE WHICH FORMS 2 QUADRATIC FACTORS AND THEN CALLS A
C     QUADRATIC FACTORING SUBROUTINE, QUAD, TO OBTAIN THE FOUR ROOTS.
C     THE PROCEDURE WAS TAKEN FROM 'INTRODUCTION TO THE THEORY OF
C     EQUATIONS' BY N.B. CONKWRIGHT.
C
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 MS,IYY,MU,MW,MOW,MQ,MIN,NUS,NAS,NTHS,KGAIN,KC,KROOT,KKK,NWS
      DIMENSION C(5),KC(4),QUF1(3),QUF2(3),RR1(2),RR2(2),RI1(2),RI2(2),R
     $OOTR(10),ROOTI(10)
      P=C(4)/C(5)
      Q=C(3)/C(5)
      R=C(2)/C(5)
      S=C(1)/C(5)
      KC(4)=1.0
      KC(3)=-.5*Q
      KC(2)=.25*(P*R-4.0*S)
      KC(1)=.125*(4.0*Q*S-P*P*S-R*R)
      HKC=(3.0*KC(4)*KC(2)-KC(3)*KC(3))/9.0*KC(4)*KC(4)
      GKC=(2.0*KC(3)*KC(3)*KC(3)-9.0*KC(4)*KC(3)*KC(2)+27.0*KC(4)*KC(4)*
     $KC(1))/(27.0*KC(4)*KC(4)*KC(4))
      RAD=GKC*GKC+4.0*HKC*HKC*HKC
      IF(RAD.LT.0.0)GO TO 3
      UKC=((-GKC+DSQRT(GKC*GKC+4.0*HKC*HKC*HKC))/2.0)
      RTUKC=DABS(UKC)**.333333333333333
      UKC=DSIGN(RTUKC,UKC)
      VKC=-HKC/UKC
      KROOT=UKC+VKC-KC(3)/(3.0*KC(4))
    1 B=DSQRT(KROOT*KROOT-S)
      A=DSQRT(2.0*KROOT+P*P*.25-Q)
      TEST=2.0*A*B+R-KROOT*P
      IF(TEST.LE..01)GO TO 2
      A=-A
```

```
      TEST=2.0*A*B+R-KROOT*P
      IF(TEST.LE..01)GO TO 2
      A=-A
      B=-B
      TEST=2.0*A*B+R-KROOT*P
      IF(TEST.LE..01)GO TO 2
      A=-A
    2 QUF1(3)=1.0
      QUF2(3)=1.0
      QUF1(2)=.5*P-A
      QUF2(2)=.5*P+A
      QUF1(1)=KROOT-B
      QUF2(1)=KROOT+B
      CALL QUAD(QUF1,RR1,RI1)
      ROOTR(1)=RR1(1)
      ROOTR(2)=RR1(2)
      ROOTI(1)=RI1(1)
      ROOTI(2)=RI1(2)
      CALL QUAD(QUF2,RR2,RI2)
      ROOTR(3)=RR2(1)
      ROOTR(4)=RR2(2)
      ROOTI(3)=RI2(1)
      ROOTI(4)=RI2(2)
      RETURN
    3 THETA=DARCOS(-GKC/(2.0*DSQRT(-HKC*HKC*HKC)))
      TH3=THETA/3.0
      KROOT = 2.0*DCOS(TH3)*DSQRT(-HKC) - KC(3)/(3.0*KC(4))
      GO TO 1
      END
```

```
      SUBROUTINE CUBE(CC,XR,XI)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION CC(4),RR(3),RI(3),XR(3),XI(3)
C
C     THIS SUBROUTINE FACTORS A THIRD ORDER POLYNOMIAL BY A CLOSED FORM
C     PROCEDURE GIVEN IN 'INTRODUCTION TO THE THEORY OF EQUATIONS' BY
C     CONKWRIGHT AND MODIFIED BY THE PROCEDURE GIVEN IN 'STANDARD MATH
C     TABLES' BY CHEMICAL RUBBER COMPANY.
C
      H=(3.0*CC(4)*CC(2)-CC(3)*CC(3))/(9.0*CC(4)*CC(4))
      G=(2.0*CC(3)*CC(3)*CC(3)-9.0*CC(4)*CC(3)*CC(2)+27.0*CC(4)*CC(4)*CC
     $(1))/(27.0*CC(4)*CC(4)*CC(4))
      RAD=G*G+4.0*H*H*H
      IF(RAD.LT.0.0)GO TO 3
      UROOT3=(-G+DSQRT(G*G+4.0*H*H*H))/2.0
      IF(UROOT3.LT.0.0)GO TO 1
      UROOT=UROOT3**.333333333333333
      GO TO 2
    1 UROOT=(-UROOT3)**.333333333333333
      UROOT=-UROOT
    2 VROOT=-H/UROOT
      RR(1)=UROOT+VROOT
      RR(2)=-(UROOT+VROOT)/2.0
      RR(3)=-(UROOT+VROOT)/2.0
      RI(1)=0.0
      RI(2)=(UROOT-VROOT)*(.866025403784)
      RI(3)=-(UROOT-VROOT)*(.866025403784)
      XR(1)=RR(1)-CC(3)/(3.0*CC(4))
      XR(2)=RR(2)-CC(3)/(3.0*CC(4))
      XR(3)=RR(3)-CC(3)/(3.0*CC(4))
      XI(1)=RI(1)
      XI(2)=RI(2)
      XI(3)=RI(3)
      GO TO 4
    3 THETA=DARCOS(-G/(2.0*DSQRT(-H*H*H)))
      TH3=THETA/3.0
      XI(1)=0.0
      XI(2)=0.0
      XI(3)=0.0
      XR(1)=2.0*DCOS(TH3)*DSQRT(-H)
      XR(2)=2.0*DCOS(TH3+2.094395102392)*DSQRT(-H)
      XR(3)=2.0*DCOS(TH3+4.188790204784)*DSQRT(-H)
      SUB=CC(3)/(3.0*CC(4))
      XR(1)=XR(1)-SUB
      XR(2)=XR(2)-SUB
      XR(3)=XR(3)-SUB
    4 RETURN
      END
```

```
      SUBROUTINE QUAD(COF,RE,RIM)
C
C     THIS SUBROUTINE FACTORS A SECOND ORDER POLYNOMIAL BY USING THE
C     QUADRATIC FORMULA.
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION COF(3),RE(2),RIM(2)
      DIS=COF(2)*COF(2)-4.0*COF(3)*COF(1)
      IF(DIS.LT.0.0) GO TO 1
      RE(1)=(-COF(2)+DSQRT(DIS))/(2.0*COF(3))
      RE(2)=(-COF(2)-DSQRT(DIS))/(2.0*COF(3))
      RIM(1)=0.0
      RIM(2)=0.0
      GO TO 2
    1 RE(1)=-COF(2)/(2.0*COF(3))
      RE(2)=RE(1)
      RIM(1)=(-DSQRT(-DIS))/(2.0*COF(3))
      RIM(2)=(DSQRT(-DIS))/(2.0*COF(3))
    2 RETURN
      END
```

```
      SUBROUTINE SINGLE(COFF,ROOT,ROOTI)
C
C     THIS SUBROUTINE FACTORS A FIRST ORDER POLYNOMIAL.
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION ROOT(1),COFF(2),ROOTI(1)
      ROOT(1)=-COFF(1)/COFF(2)
      ROOTI(1)=0.0
      RETURN
      END
```

```
      SUBROUTINE BODE(L,JJ,KWF)
C
C     THIS IS A SUBROUTINE WHICH CALCULATES THE INFORMATION NEEDED TO
C     CONSTRUCT A BODE PLOT ONCE THE NUMERATOR AND DENOMINATOR
C     POLYNOMIALS HAVE BEEN FACTORED.  THE INFORMATION IS TRANSFERRED TO
C     THE MAINLINE BY USING A 'COMMON' STATEMENT.  THE FREQUENCIES ARE
C     GIVEN BOTH IN RADIANS PER SECOND AND CYCLES PER SECOND.  THE
C     AMPLITUDE RATIO IS GIVEN BOTH AS A PURE NUMBER AND IN DECIBELS.
C     THE PHASE ANGLE IS GIVEN BOTH IN DEGREES AND RADIANS.
```

```
C
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 MS,IYY,MU,MW,MDW,MQ,MIN,NUS,NAS,NTHS,KGAIN,KC,KROOT,KKK,KD,
     $KN,NWS,KDSAVE,KNSAVE
      COMMON   WNSP,ZSP,T12SP,T05SP,WNP,ZP,T12P,T05P,WF(21),RRD(10),RRN(
     $10),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),AMPRDB(21),P
     $HDEG(21),KGAIN
      COMMON I
      DIMENSION KKK(21)
      DO 9 M=1,KWF
      AMPD=1.0
      KD=1.0
      PHASED=0.0
      DO 4 J=1,L
      IF(RRD(J).EQ.0.0)GO TO 1
      GO TO 3
    1 AMPD=AMPD*DABS(WF(M)-RID(J))
      IF(WF(M)-RID(J).LT.0.0)GO TO 2
      PHASED=PHASED-3.1415926536/2.0
      GO TO 4
    2 PHASED=PHASED+3.1415926536/2.0
      GO TO 4
    3 AMPD=DSQRT(((-WF(M)+RID(J))/RRD(J))**2+1.0)*AMPD
      PHASED=-(DATAN((-RID(J)+WF(M))/(-RRD(J))))+PHASED
      KDSAVE=KD
      KD=-RRD(J)*KD
      IF(KDSAVE*KD.LT.0.0)PHASED=PHASED-3.1415926536
    4 CONTINUE
      PHASEN=0.0
      KN=KGAIN
      AMPN=1.0
      DO 8 I=1,JJ
      IF(RRN(I).EQ.0.0)GO TO 5
      GO TO 7
    5 AMPN=AMPN*DABS(WF(M)-RIN(I))
      IF(WF(M)-RIN(I).LT.0.0)GO TO 6
      PHASEN=PHASEN+3.1415926536/2.0
      GO TU 8
    6 PHASEN=PHASEN-3.1415926536/2.0
      GO TO 8
    7 AMPN=DSQRT(((-WF(M)+RIN(I))/RRN(I))**2+1.0)*AMPN
      PHASEN=DATAN((-RIN(I)+WF(M))/(-RRN(I)))+PHASEN
      KNSAVE=KN
      KN=-RRN(I)*KN
      IF(KNSAVE*KN.LT.0.0)PHASEN=PHASEN+3.1415926536
    8 CONTINUE
      KKK(M)=KN/KD
      AMPR(M)=DABS(KKK(M))*AMPN/AMPD
      AMPRDB(M)=20.0*DLOG10(AMPR(M))
      PHASE(M)=PHASED+PHASEN
      PHDEG(M)=PHASE(M)*57.295779513
      WCYCLE(M)=WF(M)/(2.0*3.1415926536)
    9 CONTINUE
      RETURN
      END
```

SAMPLE OUTPUT

```
                    PERTINENT AIRPLANE CHARACTERISTICS
                    ----------------------------------

DENSITY (SLUGS/FT**3)         =   0.00205    VELOCITY (FT/SEC)              = 219.00000
MASS (SLUGS)                  =  82.30000    IYY (SLUG-FT**2)               =1346.00000
THRUST (POUNDS)               =   0.0        ZJ (FT)                        =   0.0
G*COS(GAMMA) (FT/SEC/SEC)     =  32.20000    G*SIN(GAMMA) (FT/SEC/SEC)      =   0.0
COS(XZ)                       =   0.99810    SIN(XZ)                        =   0.06105

WING AREA (FT**2)             = 174.00000    HORZ. TAIL AREA (FT**2)        =  33.71000
WING SPAN (FT)                =  35.83000    HORZ. TAIL SPAN (FT)           =  11.54000
WING CHORD (FT)               =   4.85627    HORZ. TAIL CHORD (FT)          =   3.35442
WING ASPECT RATIO             =   7.37310    HORZ. TAIL ASPECT RATIO        =   3.44024
WING TAPER RATIO              =   0.69500    HORZ. TAIL TAPER RATIO         =   0.65000
WING ALPHA (DEGREES)          =   1.50000    TAIL ALPHA (DEGREES)           =  -4.60634
IWING (DEGREES)               =   1.50000    ITAIL (DEGREES)                =  -3.00000
DOWNWASH ANGLE (DEGREES)      =   1.60634    DOWNWASH/ALPHA                 =   0.42101
ELEVATOR ANGLE (DEGREES)      =   7.70657    ELEVATOR AREA (FT**2)          =  16.61000
TAIL EFFICIENCY               =   0.85000    ELEVATOR CHORD (FT)            =   1.44000
2-D WING CLA                  =   0.10300    2-D TAIL CLA                   =   0.10000
CDPIE                         =   0.02690    2-D WING CDA                   =   0.0
2-D WING CL                   =   0.39000

                              DISTANCES

LENGTH OF FUSELAGE (FT)             =  25.00000   WIDTH OF FUSELAGE (FT)             =   4.17000
C.G. TO TAIL QUARTER-CHORD (FT)     =  14.60000   WING TO TAIL QUARTER-CHORD (FT)    =  14.63000
C.G. TO WING A.C.(CHORDWISE)(FT)    =   0.11630   C.G. TO WING A.C.(VERTICAL) (FT)   =   1.67000
NOSE TO WING QUARTER-CHORD (FT)     =   6.84000   C.G. TO WING QUARTER-CHORD (FT)    =  -0.11630
C.G. TO THRUST AXIS (FT)            =   0.0
```

```
                    LONGITUDINAL STABILITY DERIVATIVES
                    ----------------------------------

CL =  0.3093   CLA =  4.6080   CLDA =  1.7419   CLQ =  3.9168    CLDE =  0.4268   CLU =  0.0   CT    =  0.0
CD =  0.0311   CDA =  0.1256   CDDA =  0.0      CDQ =  0.0       CDDE =  0.0      CDU =  0.0   CTU   =  0.0
CM =  0.0      CMA = -0.8853   CMDA = -5.2370   CMQ =-12.4337    CMDE = -1.2830   CMU =  0.0   CTRPM =  0.0
```

```
                    RESPONSE TO ELEVATOR DEFLECTION

CLIN =  0.426765    CDIN =  0.0       CMIN = -1.283038    K = 1    ACC =    0.00010000
```

```
                    DIMENSIONAL STABILITY DERIVATIVES

XU =  -0.02951    ZU  =  -0.29327    MU  =   0.0        TU  =   0.0        XW  =   0.08707   ZW =  -2.19951
MW =  -0.12463    XWD =   0.0        ZDW =  -0.00916    MDW =  -0.00817    XQ  =   0.0       ZQ =  -4.50914
MQ =  -4.25034    TRPM =  0.0        XIN =   0.0        ZIN = -44.31254    MIN = -39.55799
```

POLYNOMIAL COEFFICIENTS FOR THE DENOMINATOR AND NUMERATOR

DS(5) = 1.0092	DS(4) = 8.2719	DS(3) = 36.3491	DS(2) = 1.2503	DS(1) = 1.1769
	NUS(4) = 0.0	NUS(3) = -3.8581	NUS(2) = 518.6315	NUS(1) = 2623.8309
	NAS(4) = -0.2023	NAS(3) = -39.6095	NAS(2) = -1.1685	NAS(1) = -1.7357
		NTHS(3) = -39.5580	NTHS(2) = -82.6526	NTHS(1) = -3.4144
	NWS(4) = -44.3125	NWS(3) = -8674.4787	NWS(2) = -255.9103	NWS(1) = -373.5542

SOLUTION FOR U VARIATION

DENOMINATOR ROOTS

ROOT(1) = -0.01350 +J -0.18006
ROOT(2) = -0.01350 +J 0.18006
ROOT(3) = -4.08483 +J -4.36792
ROOT(4) = -4.08483 +J 4.36792

	NATURAL FREQ UNDAMPED	NATURAL FREQ DAMPED	DAMPING RATIO	TIME FOR 1/2 DAMPING	SETTLING TIME
SHORT PERIOD	5.98034	4.36792	0.68304	0.16969	0.73337
PHUGOID	0.18058	0.18006	0.07530	50.97879	220.32436

NUMERATOR ROOTS

ROOT(1) = 139.30842 +J 0.0
ROOT(2) = -4.88185 +J 0.0

NATURAL FREQ UNDAMPED	NATURAL FREQ DAMPED	DAMPING RATIO	TIME FOR 1/2 DAMPING	SETTLING TIME
0.0	0.0	4.8819	0.14198	0.61364
0.0	0.0	-139.3084	-0.00498	-0.02150

BODE PLOT INFORMATION

FREQUENCY RAD/SEC	FREQUENCY CYCLES/SEC	AMPLITUDE RATIO PURE	AMPLITUDE RATIO DECIBELS	PHASE ANGLE RADIANS	PHASE ANGLE DEGREES
0.01000	0.00159	2236.24	66.99036	3.13292	179.50305
0.10000	0.01592	3193.30	70.08479	3.01880	172.96468
0.13543	0.02155	4936.19	73.86785	2.88477	165.28514
0.16252	0.02587	9558.39	79.60770	2.51695	144.21071
0.18058	0.02874	14815.61	83.41439	1.56521	89.67985
0.19863	0.03161	8342.88	78.42632	0.66173	37.91454
0.22572	0.03592	3762.97	71.51061	0.31593	18.10136
1.00000	0.15915	76.79	37.70663	-0.00784	-0.44934
4.48526	0.71385	4.41	12.89714	-0.45025	-25.79757
5.38231	0.85662	3.01	9.56579	-0.61691	-35.34656
5.98034	0.95180	2.36	7.44828	-0.72296	-41.42273
6.57837	1.04698	1.86	5.39652	-0.82033	-47.00142
7.47543	1.18975	1.33	2.45109	-0.94670	-54.24210
10.00000	1.59155	0.57	-4.85312	-1.18940	-68.14785
99.99998	15.91549	0.03	-63.65482	-2.16309	-123.76412
999.99979	159.15494	0.00	-108.26810	-2.99986	-171.87934

LATERAL GEOMETRIC PROGRAM

```
C
C
C     GIVEN VALUES OF THE AIRCRAFT GEOMETRY AND OTHER PERTINENT DATA,
C     THIS PROGRAM PERFORMS THE FOLLOWING:
C
C
C        1) CALCULATE NON-DIMENSIONAL STABILITY DERIVATIVES
C
C        2) CALCULATE DIMENSIONAL STABILITY DERIVATIVES
C
C        3) FORMS THE TRANSFER FUNCTIONS, TF(S)=N(S)/D(S)
C
C        4) SOLVE FOR ROOTS OF D(S) AND N(S)
C
C        5) CALCULATE NATURAL FREQUENCIES, DAMPING RATIOS, TIME TO DAMP
C           TO ONE-HALF AMPLITUDE, AND SETTLING TIME
C
C        6) PRODUCES INFORMATION NEEDED FOR BODE PLOT CONSTRUCTION
C
C
C
C     THE DERIVATION OF THE EQUATIONS OF MOTION ON WHICH THIS ANALYSIS IS
C     BASED WAS TAKEN FROM 'DYNAMICS OF THE AIRFRAME', BUREAU OF
C     AERONAUTICS REPORT, AE-61-4II.
C
C     THE ANALYSIS DESCRIBED ABOVE MUST MEET THE ASSUMPTIONS IMPOSED ON
C     THE EQUATIONS OF MOTION WHEN THEY WERE DERIVED.
C     THESE ASSUMPTIONS ARE:
C
C        1) THE AIRFRAME IS ASSUMED TO BE A RIGID BODY.
C
C        2) THE EARTH IS ASSUMED TO BE FIXED IN SPACE, AND, UNLESS
C           SPECIFICALLY STATED OTHERWISE, THE EARTH'S ATMOSPHERE IS
C           ASSUMED TO BE FIXED WITH RESPECT TO THE EARTH.
C
C        3) THE MASS OF THE AIRPLANE IS ASSUMED TO REMAIN CONSTANT FOR
C           THE DURATION OF ANY PARTICULAR DYNAMIC ANALYSIS.
C
C        4) THE X-Z PLANE IS ASSUMED TO BE A PLANE OF SYMMETRY.
C
C        5) THE DISTURBANCES FROM THE STEADY FLIGHT CONDITION ARE ASSUMED
C           TO BE SMALL ENOUGH SO THAT THE PRODUCTS AND SQUARES OF THE
C           CHANGES IN VELOCITIES ARE NEGLIGIBLE IN COMPARISON WITH THE
C           CHANGES THEMSELVES.  ALSO, THE DISTURBANCE ANGLES ARE ASSUMED
C           TO BE SMALL ENOUGH SO THAT THE SINES OF THESE ANGLES MAY BE
C           SET EQUAL TO THE ANGLES AND THE COSINES SET EQUAL TO ONE.
C           PRODUCTS OF THESE ANGLES ARE ALSO APPROXIMATELY ZERO AND CAN
C           BE NEGLECTED.  AND, SINCE THE DISTURBANCES ARE SMALL, THE
C           CHANGE IN AIR DENSITY ENCOUNTERED BY THE AIRPLANE DURING ANY
C           DISTURBANCE CAN BE CONSIDERED TO BE ZERO.
C
C        6) DURING THE STEADY FLIGHT CONDITION, THE AIRPLANE IS ASSUMED
C           TO BE FLYING WITH WINGS LEVEL AND ALL COMPONENTS OF VELOCITY
C           ZERO EXCEPT U SUB 0.  W SUB 0 = 0 BECAUSE THE STABILITY AXES
C           WERE CHOSEN AS THE REFERENCE AXES.
C
C        7) THE FLOW IS ASSUMED TO BE QUASI-STEADY.
C
C
C
C      THE PERTINENT AIRPLANE CHARACTERISTICS ARE DEFINED AS FOLLOWS:
C
C      GCOSGM AND GSINGM ARE THE PRODUCTS OF THE ACCELERATION DUE TO
C      GRAVITY (ASSUME = 32.2 FT/SEC**2 FOR THIS ALTITUDE RANGE) AND THE
C      COSINE AND SINE RESPECTIVELY OF THE INITIAL FLIGHT PATH ANGLE,
C      GAMMA, (USUALLY ZERO FOR LEVEL FLIGHT).
C
C      IZZ IS THE MOMENT OF INERTIA ABOUT THE Z AXIS (FT-LBS-SEC*SEC)
C
C      IXZ IS THE PRODUCT OF INERTIA (FT-LBS-SEC*SEC)
C
C      IXX IS THE MOMENT OF INERTIA ABOUT THE X AXIS (FT-LBS-SEC*SEC)
C
C      MS IS THE MASS OF THE AIRPLANE (SLUGS)
C
C      S IS THE WING AREA OF THE AIRPLANE (SQUARE FEET)
C
C      RHO IS THE DENSITY AT THE ALTITUDE AT WHICH THE AIRPLANE IS FLYING
C
C      U IS THE SPEED OF THE AIRCRAFT IN FEET PER SECOND
C
C      CH IS THE MEAN AERODYNAMIC CHORD OF THE WING (FEET)
C
C      B IS THE WING SPAN (FEET)
C
C      CL IS THE AIRPLANE LIFT COEFFICIENT
C
C      SA IS THE WING SWEEP ANGLE (POSITIVE AFT, IN RADIANS)
C
C      DIH IS THE WING DIHEDRAL ANGLE (POSITIVE UP, IN DEGREES)
C
C      ZW IS THE DISTANCE FROM BODY CENTERLINE TO QUARTER-CHORD POINT OF
C      EXPOSED WING ROOT CHORD (POSITIVE FOR QUARTER-CHORD POINT BELOW
C      THE BODY CENTERLINE, FEET)
C
C      FUSVOL IS THE VOLUME OF THE FUSELAGE (CUBIC FEET)
C
C      H IS THE MAXIMUM BODY HEIGHT AT WING-BODY INTERSECTION (FEET)
C
C      SV IS THE AREA OF THE VERTICAL TAIL(SQUARE FEET)
C
C      BV IS THE SPAN OF THE VERTICAL TAIL(FEET)
C
C      R1 IS THE RADIUS OF THE FUSELAGE IN THE VICINITY OF THE VERTICAL
C      TAIL(FEET)
C
C      TR IS THE WING TAPER RATIO (TIP CHORD/ROOT CHORD)
C
C      ZV IS THE DISTANCE FROM THE CENTER OF PRESSURE OF THE VERTICAL
C      TAIL TO THE AIRPLANE'S X-AXIS (POSITIVE FOR VERTICAL TAIL ABOVE
C      THE X-AXIS, FEET)
C
C      ETAV IS THE EFFICIENCY FACTOR OF THE VERTICAL TAIL
C
C      SBS IS THE BODY SIDE AREA OF THE FUSELAGE(SQUARE FEET)
C
C      LF IS THE LENGTH OF THE FUSELAGE(FEET)
C
C      LT IS THE LENGTH FROM C.G. TO CENTER OF PRESSURE OF THE TAIL, FEET
C
C      XM IS THE DISTANCE FROM THE NOSE TO THE C.G.(FEET)
C
C      H1 IS THE FUSELAGE HEIGHT MEASURED AT 1/4 LF FROM THE NOSE(FEET)
C
C      H2 IS THE FUSELAGE HEIGHT MEASURED AT 3/4 LF FROM THE NOSE(FEET)
C
C      W IS THE MAXIMUM WIDTH OF THE FUSELAGE(FEET)
C
```

```
C     SAH IS THE SWEEP ANGLE OF THE HORIZONTAL TAIL(IN RADIANS)
C
C     CLA2DW IS TWO-DIMENSIONAL LIFT CURVE SLOPE OF THE WING(PER RADIAN)
C
C     BH IS THE SPAN OF THE HORIZONTAL TAIL, FEET
C
C     SH IS THE AREA OF THE HORIZONTAL TAIL, SQUARE FEET
C
C     TRH IS TAPER RATIO OF THE HORIZONTAL TAIL (TIP CHORD/ROOT CHORD)
C
C     CLA2DH IS THE TWO-DIMENSIONAL LIFT CURVE SLOPE OF THE HORIZONTAL
C     TAIL (PER RADIAN)
C
C     BA IS THE SPAN OF AN AILERON(ON ONE SIDE), FEET
C
C     CA IS THE AILERON CHORD, FEET
C
C     SR IS THE AREA OF THE RUDDER, SQUARE FEET
C
C     ALPHA IS THE ANGLE OF ATTACK AT WHICH THE AIRPLANE IS OPERATING(IN
C     RADIANS)
C
C     CDO IS THE PARASITE DRAG OF THE AIRCRAFT
C
C     YI IS THE DISTANCE FROM THE BODY CENTERLINE TO THE INBOARD EDGE
C     OF THE AILERON, FEET
C
C     THE FOLLOWING CARDS DEFINE VARIABLES USED TO APPROXIMATE FUSELAGE
C     VOLUME BY DIVIDING THE VOLUME INTO 4 PRISMOIDS.
C
C     HNOSE IS THE FUSELAGE HEIGHT IN THE NOSE REGION,FEET
C
C     WNOSE IS THE FUSELAGE WIDTH IN THE NOSE REGION,FEET
C
C     HFCY IS THE FUSELAGE HEIGHT AT THE FRONT OF THE CANOPY, FEET
C
C     WFCY IS THE FUSELAGE WIDTH AT THE FRONT OF THE CANOPY, FEET
C
C     LFCY IS THE LENGTH ALONG THE BODY CENTERLINE FROM NOSE TO FRONT OF
C     CANOPY, FEET
C
C     LMH IS THE LENGTH ALONG THE BODY CENTERLINE FROM NOSE TO POINT OF
C     MAXIMUM FUSELAGE HEIGHT, FEET
C
C     HBCY IS THE FUSELAGE HEIGHT AT THE BACK OF THE CANOPY, FEET
C
C     WBCY IS THE FUSELAGE WIDTH AT THE BACK OF THE CANOPY, FEET
C
C     LBCY IS THE LENGTH ALONG THE BODY CENTERLINE FROM NOSE TO BACK OF
C     CANOPY, FEET
C
C
C
C     THE FOLLOWING CARDS IMPLY THAT THE PROGRAM IS EXECUTED IN DOUBLE
C     PRECISION.
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMPLEX*16 P,TST
      REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
     1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU,LFCY,LMH,LBCY
      COMMON WNSP,ZSP,T12SP,T05SP,WNP,ZP,T12P,T05P,WF(21),RRD(10),RRN(10
     1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
     1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,R1,TR,ZV,YI,ETAV,B,SBS,CL
     1AVT,XM,H1,H2,W,WP,RHO,CLA2DW,SAH,BH,SH,TRH,CLA2DH,BA,CA,CH,SR,ALPH
     1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
     1DA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
      DIMENSION NS(5),NPHI(5),NPSI(5),DS(6),ROOTR(10),ROOTI(10),C(5),KC(
     14),QUF1(3),QUF2(3),RR1(2),RR2(2),RI1(2),RI2(2),CC(4),RR(3),COFFI(6
     1),RI(3),XR(3),XI(3),COF(3),RE(2),RIM(2),ROOT(1),COFF(2),KKK(21),KE
     1Y(15),P(6)
C
C     READ IN PERTINENT AIRPLANE CHARACTERISTICS
C
      READ (1,1) CL,SA,DIH,ZW,H,S,SV,BV,R1,TR,ZV,ETAV,B,LF,LT,XM,H1,H2,W
     1,CLA2DW,SAH,BH,SH,TRH,CLA2DH,BA,CA,SR,ALPHA,CDO,RHO,YI,U,MS,GCOSGM
     1,GSINGM,IXX,IXZ,IZZ,HNOSE,WNOSE,HFCY,WFCY,LFCY,LMH,HBCY,WBCY,LBCY
    1 FORMAT (8F10.6/8F10.6/8F10.6/8F10.6/8F10.6/8F10.6)
C
C     CALCULATE EFFECTIVE ASPECT RATIO OF VERTICAL TAIL AND LIFT CURVE
C     SLOPE OF VERTICAL TAIL
C
      AE=1.55*BV*BV/SV
      IF (AE.GE.0.0.AND.AE.LE.6.5) GO TO 3
      WRITE (3,2) AE
    2 FORMAT (1X,'VALUE OF AE = ',G11.4,'   USED TO CALCULATE CLAVT IS O
     1UTSIDE PREFERRED RANGE OF 0.0 TO 6.5')
    3 CLAVT=.0000397694*AE**5-.00069754*AE**4+.00463464*AE**3-.0158634*A
     1E*AE+.0401979*AE+.00037328
      CLAVT=CLAVT*57.2958
C
C     CALCULATE ASPECT RATIO OF WING AND HORIZONTAL TAIL
C
      AR=B*B/S
      ARH=BH*BH/SH
C
C     CALCULATE THE MEAN AERODYNAMIC CHORD
C
      CH=S/B
C
C     ESTIMATE BODY SIDE AREA USING FOUR TRAPEZOIDS
C
      SBS=(HNOSE+HFCY)*LFCY/2.+(H+HFCY)*(LMH-LFCY)/2.+(H+HBCY)*(LBCY-LMH
     1)/2.+(HBCY+2.*R1)*(LF-LBCY)/2.
C
C     ESTIMATE FUSELAGE VOLUME USING FOUR PRISMOIDS
C
      V1=LFCY*(2.0*(HNOSE*WNOSE+HFCY*WFCY)+HFCY*WNOSE+HNOSE*WFCY)
      V2=(LMH-LFCY)*(2.0*(HFCY*WFCY+H*W)+H*WFCY+HFCY*W)
      V3=(LBCY-LMH)*(2.0*(HBCY*WBCY+H*W)+H*WBCY+W*HBCY)
      V4=(LF-LBCY)*(2.0*(HBCY*WBCY+(2.0*R1)*(2.0*R1))+HBCY*2.0*R1+WBCY*2
     1.0*R1)
      FUSVOL=(V1+V2+V3+V4)/6.0
C
C     NEXT THREE CARDS DETERMINE WHETHER THE AIRPLANE IS LOW WING(WP=3.0
C     MID WING(WP=2.0), OR HIGH WING(WP=1.0)
C
      R=ZW/H
      WP=2.0
      IF (R.GE..25)WP=3.0
      IF (R.LT.-.25)WP=1.0
C
C     WRITE PERTINENT AIRPLANE CHARACTERISTICS AS DEFINED IN THE INITIAL
C     PART OF THIS PROGRAM
C
      WRITE (3,4) RHO,S,MS,GCOSGM,U,CH,B,GSINGM,IXX,IXZ,IZZ,CL,SA,DIH,ZW
     1,FUSVOL,H,SV,BV,R1,TR,ZV,ETAV,SBS,LF,LT,XM,H1,H2,W,SAH,CLA2DW,BH,S
     1H,TRH,CLA2DH,BA,CA,SR,ALPHA,CDO,YI,HNOSE,WNOSE
    4 FORMAT (1X,121('*'),/,1X,'*',119X,'*',/,1X,'*',40X,'PERTINENT AIRP
```

```
     1LANE CHARACTERISTICS',45X,'*',/,1X,'*',40X,34('-'),45X,'*',/,1X,'*
     1',119X,'*',/,1X,'*',17X,'RHO =',F10.6,3X,'WING AREA =',F10.6,3X,'M
     1ASS =',F10.6,3X,'G*COS(GAMMA) =',F10.6,17X,'*',/,1X,'*',17X,'U   =
     1',F10.4,3X,'CHORD      =',F10.6,3X,'SPAN =',F10.4,3X,'G*SIN(GAMMA)
     1=',F10.6,17X,'*'/,1X,'*',17X,'IXX =',F10.4,3X,'IXZ        =',F9.4,4
     1X,'IZZ  =',F9.4,4X,'CL',11X,'=',F10.6,17X,'*',/,1X,'*',17X,'SA  ='
     1,F10.6,3X,'DIH',7X,'=',F10.6,3X,'ZW   =',F10.6,3X,'FUSVOL',7X,'=',
     1F10.6,17X,'*',/,1X,'*',17X,'H   =',F10.6,3X,'SV',8X,'=',F10.6,3X,'
     1BV   =',F10.6,3X,'K1',11X,'=',F10.6,17X,'*',/,1X,'*',17X,'TR  =',F
     110.6,3X,'ZV',8X,'=',F10.6,3X,'ETAV =',F10.6,3X,'SBS',10X,'=',F10.6
     1,17X,'*',/,1X,'*',17X,'LF  =',F10.6,3X,'LT',8X,'=',F10.6,3X,'XM
     1=',F10.6,3X,'H1',11X,'=',F10.6,17X,'*',/,1X,'*',17X,'H2  =',F10.6,
     13X,'W',9X,'=',F10.6,3X,'SAH  =',F10.6,3X,'CLA2DW',7X,'=',F10.6,17X
     1,'*',/,1X,'*',17X,'B4  =',F10.6,3X,'SH',8X,'=',F10.6,3X,'TRH  =',F
     110.6,3X,'CLA2DH',7X,'=',F10.6,17X,'*',/,1X,'*',17X,'BA  =',F10.6,3
     1X,'CA',8X,'=',F10.6,3X,'SR   =',F10.6,3X,'ALPHA',8X,'=',F10.6,17X,
     1'*',/,1X,'*',17X,'CDO =',F10.6,3X,'Y1',8X,'=',F10.6,3X,'HNOSE=',F1
     10.6,3X,'WNOSE',8X,'=',F10.6,17X,'*')
      WRITE (3,5) HFCY,WFCY,LFCY,LMH,HBCY,WBCY,LBCY
    5 FORMAT (1X,'*',17X,'HFCY=',F10.6,3X,'WFCY',6X,'=',F10.6,3X,'LFCY =
     1',F10.6,3X,'LMH',10X,'=',F10.6,17X,'*',/,1X,'*',17X,'HBCY=',F10.6,
     13X,'WBCY',6X,'=',F10.6,3X,'LBCY =',F10.6,44X,'*',/,1X,'*',119X,'*'
     1,/,1X,121('*'),/)
C
C     THE FOLLOWING READ STATEMENT ENABLES ONE TO READ IN A VALUE FOR A
C     PARTICULAR STABILITY DERIVATIVE OR GROUP OF DERIVATIVES, THEREBY
C     OVER RIDING THE CALCULATED VALUE, BY SIMPLY PUTTING A ONE (1.0) IN
C     COLUMN 5 FOR CYB, 10 FOR CLB, 15 FOR CNB, 20 FOR CLP, 25 FOR CYP,
C     30 FOR CNP, 35 FOR CYR, 40 FOR CLR, 45 FOR CNR, 50 FOR CYDA, 55
C     FOR CLDA, 60 FOR CNDA, 65 FOR CYDR, 70 FOR CLDR, OR 75 FOR CNDR,
C     AND THEN PLACING THE VALUES TO BE READ, IN ORDER, (ONE PER CARD)
C     BEHIND THIS KEY CARD.
C
      READ (1,6) (KEY(JI),JI=1,15)
    6 FORMAT (15I5)
C
C     THE FOLLOWING CARDS CALL INDIVIDUAL SUBROUTINES WHICH COMPUTE
C     VALUES FOR THE NON-DIMENSIONAL STABILITY DERIVATIVES
C
      CALL SCYB
      IF (KEY(1).EQ.0) GO TO 8
      READ (1,7) CYB
    7 FORMAT (F10.6)
    8 CALL SCLB
      IF (KEY(2).EQ.0) GO TO 9
      READ (1,7) CLB
    9 CALL SCNB
      IF (KEY(3).EQ.0) GO TO 10
      READ (1,7) CNB
   10 CALL SCLP
      IF (KEY(4).EQ.0) GO TO 11
      READ (1,7) CLP
   11 CALL SCYP
      IF (KEY(5).EQ.0) GO TO 12
      READ (1,7) CYP
   12 CALL SCNP
      IF (KEY(6).EQ.0) GO TO 13
      READ (1,7) CNP
   13 CALL SCYR
      IF (KEY(7).EQ.0) GO TO 14
      READ (1,7) CYR
   14 CALL SCLR
      IF (KEY(8).EQ.0) GO TO 15
      READ (1,7) CLR
   15 CALL SCNR
      IF (KEY(9).EQ.0) GO TO 16
      READ (1,7) CNR
   16 CALL SCYDA
      IF (KEY(10).EQ.0) GO TO 17
      READ (1,7) CYDA
   17 CALL SCLDA
      IF (KEY(11).EQ.0) GO TO 18
      READ (1,7) CLDA
   18 CALL SCNDA
      IF (KEY(12).EQ.0) GO TO 19
      READ (1,7) CNDA
   19 CALL SCYDR
      IF (KEY(13).EQ.0) GO TO 20
      READ (1,7) CYDR
   20 CALL SCLDR
      IF (KEY(14).EQ.0) GO TO 21
      READ (1,7) CLDR
   21 CALL SCNDR
      IF (KEY(15).EQ.0) GO TO 22
      READ (1,7) CNDR
   22 WRITE (3,23)
   23 FORMAT (1X,121('*'),/,1X,'*',119X,'*',/,1X,'*',44X,'LATERAL STABIL
     1ITY DERIVATIVES',46X,'*',/,1X,'*',44X,29('-'),46X,'*',/,1X,'*',119
     1X,'*',/,1X,'*',119X,'*')
      WRITE (3,24) CYB,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CNDA,CY
     1DR,CLDR,CNDR
   24 FORMAT (1X,'*  CYB  =',F10.6,4X,'CLB   =',F10.6,4X,'CNB =',F10.6,4
     1X,'CYP =',F10.6,4X,'CLP  =',F10.6,4X,'CNP  =',F10.6,'  *',/,1X,'*
     1 CYR  =',F10.6,4X,'CLR   =',F10.6,4X,'CNR =',F10.6,4X'CYDA =',F9.6
     1,4X,'CLDA =',F10.6,4X,'CNDA =',F10.6,'  *',/,1X,'*  CYDR =',F10.6,
     14X,'CLDR  =',F10.6,4X,'CNDR =',F9.6,59X,'  *',/,1X,'*',119X,'*',/,
     11X,121('*'),/)
      LCOUNT=0
C
C     LCOUNT IS A COUNTER WHICH PREVENTS MULTIPLE CALCULATIONS OF THE
C     CHARACTERISTIC EQUATION.
C
C     THE READ STATEMENT BELOW IS THE INPUT FOR THE VARIABLES EXPLAINED
C     BELOW.
   25 READ (1,26) K,NUMER,ABC
   26 FORMAT (2I3,E14.7)
      IF (NUMER.EQ.0) CALL EXIT
C
      IF (K.EQ.1) GO TO 27
      CYIN=CYDA
      CLIN=CLDA
      CNIN=CNDA
      GO TO 28
   27 CYIN=CYDR
      CLIN=CLDR
      CNIN=CNDR
C
C     CYIN, CLIN, AND CNIN ARE THE STABILITY DERIVATIVES DUE TO CONTROL
C     SURFACE DEFLECTIONS.  CYIN IS EITHER THE PARTIAL OF CY WITH
C     RESPECT TO RUDDER DEFLECTION OR AILERON DEFLECTION. THE VALUE OF K
C     DETERMINES WHETHER IT IS CONCERNED WITH RUDDER OR AILERON.
C
C     IF K IS GIVEN THE VALUE 1 THEN CYIN, CLIN, AND CNIN ARE PARTIAL
C     DERIVATIVES WITH RESPECT TO RUDDER DEFLECTION. IF K=2 THEN THE
C     PARTIALS ARE TAKEN WITH RESPECT TO AILERON DEFLECTION.
C
C     NUMER DEFINES WHICH TRANSFER FUNCTION WE ARE INTERESTED IN, (BETA,
C     PHI, PSI). NUMER=1 GIVES VARIATIONS IN BETA, NUMER=2 GIVES
```

```
C     VARIATIONS IN PHI, AND NUMER=3 GIVES VARIATIONS IN PSI.
C
C     THUS, FOR EXAMPLE, IF K=2 AND NUMER=2 THE TRANSFER FUNCTION WILL
C     BE VARIATION IN PHI DUE TO DEFLECTION OF THE AILERONS.
C
C     IT SHOULD BE NOTICED THAT THE PROGRAM IS SET-UP TO CALCULATE THE
C     DIMENSIONAL STABILITY DERIVATIVES AND THE DENOMINATOR POLYNOMIAL
C     ONLY ONCE; THEREFORE, THE VALUE OF K SHOULD REMAIN CONSTANT FOR
C     A PARTICULAR DATA SET.
C
C     ACC IS THE ACCURACY USED THROUGHOUT THE PROGRAM TO COMPARE TWO
C     NUMBERS TO SEE IF THEY ARE SUFFICIENTLY CLOSE.
C
   28 LCOUNT=LCOUNT+1
      IF (LCOUNT.GE.2) GO TO 37
      IF (K.NE.1) GO TO 30
      WRITE (3,29)
   29 FORMAT (15X,98('*'),/,15X,'*',96X,'*',/,15X,'*  ',32X,'RESPONSE TO
     1 RUDDER DEFLECTION',31X,'  *')
      GO TO 32
   30 WRITE (3,31)
   31 FORMAT (15X,98('*'),/,15X,'*',96X,'*',/,15X,'*  ',31X,'RESPONSE TO
     1 AILERON DEFLECTION',31X,'  *')
   32 WRITE (3,33) CYIN,CLIN,CNIN,K,ACC
   33 FORMAT (15X,'*',96X,'*',/,15X,'*',96X,'*',/15X,'*  ','CYIN =',F10.
     16,4X,'CLIN =',F10.6,4X,'CNIN =',F10.6,4X,'K =',I2,4X,'ACC =',F14.8
     1,6X,'*',/,15X,'*',96X,'*',/,15X,98('*'))
C
C     CALCULATION OF DIMENSIONAL STABILITY DERIVATIVES
C
      A=RHO*U*S
      F=RHO*U*S*B
      H=RHO*U*S*B*B
      D=RHO*U*U*S
      E=RHO*U*U*S*B
C
C     A, F, H, D, E ARE JUST CONSTANTS USED TO CALCULATE THE DIMENSIONAL
C     STABILITY DERIVATIVES.
C
      YV=A*CYB/(2.0*MS)
      LV=F*CLB/(2.0*IXX)
      LB=U*LV
      NV=F*CNB/(2.0*IZZ)
      NB=U*NV
      YP=F*CYP/(4.0*MS)
      LP=H*CLP/(4.0*IXX)
      NP=H*CNP/(4.0*IZZ)
      YR=F*CYR/(4.0*MS)
      LR=H*CLR/(4.0*IXX)
      NR=H*CNR/(4.0*IZZ)
      YIN=D*CYIN/(2.0*MS)
      LIN=E*CLIN/(2.0*IXX)
      NIN=E*CNIN/(2.0*IZZ)
C
C     THE FOLLOWING WRITE STATEMENT GIVES THE DIMENSIONAL STABILITY
C     DERIVATIVES.
C
      WRITE (3,34) YV,LB,NB,YP,LP,NP,YR,LR,NR,YIN,LIN,NIN
   34 FORMAT (/2X,123('*'),/,2X,'*',121X,'*',/,2X,'*',44X,'DIMENSIONAL S
     1TABILITY DERIVATIVES',44X,'*',/,2X,'*',121X,'*',/,2X,'*',2X,'  YV
     1=',F10.5,4X,'  LB =',F10.5,4X,'  NB =',F10.5,4X,'  YP =',F10.5,4X,
     1'  LP =',F10.5,4X,'  NP =',F10.5,3X,'*',/,2X,'*',3X,' YR =',F10.5,
     14X,'  LR =',F10.5,4X,'  NR =',F10.5,4X,' YIN =',F10.5,4X,' LIN =',
     1F10.5,4X,' NIN =',F10.5,3X,'*',/,2X,'*',121X,'*',/,2X,123('*'))
C
C     COEFFICIENTS FOR TRANSFER FUNCTION
C
C     DENOMINATOR COEFFICIENTS, DS(5)S**4+DS(4)S**3+DS(3)S**2+DS(2)S+
C     DS(1), WHERE S REPRESENTS THE LAPLACIAN OPERATOR.
C
      DS(5)=1.0-(IXZ*IXZ)/(IXX*IZZ)
      DS(4)=YV*((IXZ*IXZ)/(IXX*IZZ)-1.0)-LP-NR-(IXZ/IXX)*NP-(IXZ/IZZ)*LR
      DS(3)=YV*(LP+NR+(IXZ/IXX)*NP+(IXZ/IZZ)*LR)+NR*LP+LB*((IXZ/IZZ)*(1.
     10-YR/U)-YP/U)+NB*((1.0-YR/U)-(IXZ/IXX)*(YP/U))-NP*LR
      DS(2)=-NB*(LR*YP/U+(IXZ/IXX)*(GCOSGM/U)+GSINGM/U+LP*(1.0-YR/U))+LB
     1*(NP*(1.0-YR/U)-(IXZ/IZZ)*GSINGM/U-GCOSGM/U+NR*YP/U)+YV*(NP*LR-NR*
     1LP)
      DS(1)=LB*(NR*GCOSGM/U-NP*GSINGM/U)+NB*(LP*GSINGM/U-LR*GCOSGM/U)
C
C     NUMERATOR COEFFICIENTS FOR SIDESLIP VARIATION, NS(4)S**3+NS(3)S**2
C     +NS(2)S+NS(1)
C
      NS(4)=(YIN/U)*(1.0-(IXZ*IXZ)/(IXX*IZZ))
      NS(3)=(YIN/U)*(-NR-LP-IXZ/IZZ-IXZ*NP/IXX)+LIN*(YP/U-(IXZ/IZZ)*(1.0
     1-YR/U))+NIN*((YP/U)*(IXZ/IXX)-(1.0-YR/U))
      NS(2)=(YIN/U)*(LP*NR-LR*NP)+LIN*(GCOSGM/U-YP*NR/U-NP*(1.0-YR/U)+(G
     1SINGM/U)*(IXZ/IZZ))+NIN*(GCOSGM*IXZ/(U*IXX)+YP*LR/U+LP*(1.0-YR/U)+
     1GSINGM/U)
      NS(1)=(NIN/U)*(LR*GCOSGM-LP*GSINGM)+(LIN/U)*(NP*GSINGM-NR*GCOSGM)
C
C     NUMERATOR COEFFICIENTS FOR ROLLING VARIATION, NPHI(4)S**3+
C     NPHI(3)S**2+NPHI(2)S+NPHI(1)
C
      NPHI(4)=LIN+NIN*IXZ/IXX
      NPHI(3)=LIN*(-YV-NR)+NIN*(LR-IXZ*YV/IZZ)+(YIN/U)*(LB+IXZ*NB/IXX)
      NPHI(2)=YV*(LIN*NR-NIN*LR)-(YIN/U)*(LB*NR-LR*NB)+(1.0-YR/U)*(LIN*N
     1B-NIN*LB)
      NPHI(1)=(GSINGM/U)*(NIN*LB-LIN*NB)
C
C     COEFFICIENTS FOR NUMERATOR FOR YAWING VARIATION, NPSI(4)S**3
C     +NPSI(3)S**2+NPSI(2)S+NPSI(1)
C
      NPSI(4)=NIN+IXZ*LIN/IZZ
      NPSI(3)=NIN*(-YV-LP)+LIN*(-IXZ*YV/IZZ+NP)+(YIN/U)*(NB+IXZ*LB/IZZ)
      NPSI(2)=NIN*(YV*LP-YP*LB/U)+LIN*(YP*NB/U-NP*YV)+(YIN/U)*(NP*LB-LP*
     1NB)
      NPSI(1)=(GCOSGM/U)*(LIN*NB-NIN*LB)
C
C     THE FOLLOWING WRITE STATEMENT GIVES THE COEFFICIENT OF THE
C     NUMERATOR AND DENOMINATOR.
C
      WRITE (3,35) DS(5),DS(4),DS(3),DS(2),DS(1),NS(4),NS(3),NS(2),NS(1)
     1,NPHI(4),NPHI(3),NPHI(2),NPHI(1),NPSI(4),NPSI(3),NPSI(2),NPSI(1)
   35 FORMAT (/2X,123('*'),/,2X,'*',121X,'*',/,2X,'*',32X,'POLYNOMIAL CO
     1EFFICIENTS FOR THE DENOMINATOR AND NUMERATOR',32X,'*',/2X,'*',32X,
     157('-'),32X,'*',/,2X,'*',121X,'*',/,2X,'*',3X,'DS(5)   =',F10.4,5X
     1,'DS(4)   =',F10.4,5X,'DS(3)   =',F10.4,5X,'DS(2)   =',F10.4,5X,'D
     1S(1)   =',F10.4,3X,'*',/,2X,'*',121X,'*',/,2X,'*',27X,'NS(4)   =',
     1F10.4,5X,'NS(3)   =',F10.4,5X,'NS(2)   =',F10.4,5X,'NS(1)   =',F10
     1.4,3X,'*',/,2X,'*',121X,'*',/,2X,'*',27X,'NPHI(4) =',F10.4,5X,'NPH
     1I(3) =',F10.4,5X,'NPHI(2) =',F10.4,5X,'NPHI(1) =',F10.4,3X,'*',/2X
     1,'*',121X,'*',/,2X,'*',27X,'NPSI(4) =',F10.4,5X,'NPSI(3) =',F10.4,
     15X,'NPSI(2) =',F10.4,5X,'NPSI(1) =',F10.4,3X,'*',/,2X,'*',121X,'*'
     1,/,2X,123('*'))
C     THE DO LOOP BELOW DETERMINES THE ORDER OF THE POLYNOMIAL IN THE
C     DENOMINATOR, ND.
C
      DO 36 I=1,5
```

```
      IF (DABS(DS(I)).GT.ACC)MD=I-1
   36 CONTINUE
      IF (MD.NE.0) GO TO 37
      CALL EXIT
C
C     IF MD = 0, THERE IS NO CHARACTERISTIC EQUATION, THEREFORE THE
C     PROGRAM IS TERMINATED.
C
C     THE 3 'IF' STATEMENTS BELOW TELL US WHICH SET OF NUMERATOR
C     COEFFICIENTS TO EVALUATE DEPENDING ON THE VALUE OF NUMER.
C
   37 IF (NUMER.EQ.1) GO TO 38
      IF (NUMER.EQ.2) GO TO 41
      IF (NUMER.EQ.3) GO TO 45
   38 WRITE (3,39)
   39 FORMAT ('1',130('*'),/,1X,'*',128X,'*',/,1X,'*',128X,'*',/,1X,'*',
     149X,'SOLUTION FOR SIDESLIP VARIATION',48X,'*',/,1X,'*',49X,31('-')
     1,48X,'*',/,1X,'*',128X,'*',/,1X,'*',128X,'*',/,1X,'*',55X,'DENOMIN
     1ATOR ROOTS',56X,'*',/,1X,'*',128X,'*')
C
C     THE DO LOOP BELOW DETERMINES THE ORDER OF THE POLYNOMIAL IN THE
C     NUMERATOR, MN.
C
      DO 40 I=1,4
      IF (DABS(NS(I)).GT.ACC)MN=I-1
   40 CONTINUE
      IF (MN.NE.0) GO TO 48
      CALL EXIT
   41 WRITE (3,42)
   42 FORMAT ('1',130('*'),/,1X,'*',128X,'*',/,1X,'*',128X,'*',/,1X,'*',
     148X,'SOLUTION FOR ROLLING VARIATION',50X,'*',/,1X,'*',48X,30('-'),
     150X,'*',/,1X,'*',128X,'*',/,1X,'*',128X,'*',/,1X,'*',55X,'DENOMINA
     1TOR ROOTS',56X,'*',/,1X,'*',128X,'*')
C
      DO 43 I=1,4
      IF (DABS(NPHI(I)).GT.ACC)MN=I-1
   43 CONTINUE
      IF (MN.NE.0) GO TO 48
      WRITE (3,44) MN
   44 FORMAT (1X,'MN=',I3)
      CALL EXIT
   45 WRITE (3,46)
   46 FORMAT ('1',130('*'),/,1X,'*',128X,'*',/,1X,'*',128X,'*',/,1X,'*',
     145X,'SOLUTION FOR YAWING VARIATION',54X,'*',/,1X,'*',45X,29('-'),5
     14X,'*',/,1X,'*',128X,'*',/,1X,'*',128X,'*',/,1X,'*',55X,'DENOMINAT
     1OR ROOTS',56X,'*',/,1X,'*',128X,'*')
      DO 47 I=1,4
      IF (DABS(NPSI(I)).GT.ACC)MN=I-1
   47 CONTINUE
      IF (MN.NE.0) GO TO 48
      WRITE (3,44) MN
      CALL EXIT
C
C     GETROT IS A SUBROUTINE WHICH, USING OTHER SUBROUTINES, CALCULATES
C     ROOTS, DAMPING RATIOS, AND NATURAL FREQUENCIES, AND THESE ARE
C     TRANSFERRED TO THE MAINLINE BY USE OF A 'COMMON' STATEMENT.
C
   48 CALL GETROT(DS,MD,RRD,RID)
C
C     THE FOLLOWING FOUR CARDS ADD A ZERO DENOMINATOR ROOT IF THE
C     TRANSFER FUNCTION IS NOT FOR SIDESLIP.
C
      IF (NUMER.EQ.1) GO TO 49
      MD=MD+1
      RRD(MD)=0.0
      RID(MD)=0.0
   49 WRITE (3,50) (J,RRD(J),RID(J),J=1,MD)
   50 FORMAT (1X,'*',46X,'ROOT(',I1,') = ',F10.5,'  +J ',F10.5,47X,'*')
      WRITE (3,51)
   51 FORMAT (1X,'*',128X,'*',/,1X,'*',128X,'*',/,1X,'*',34X,'NATURAL FR
     1EQ',5X,'DAMPING RATIO',4X,'TIME FOR 1/2 DAMPING',9X,'SETTLING TIME
     1',18X,'*',/,1X,'*',31X,'UNDAMPED   DAMPED',80X,'*')
C
C     FOR DAMPING RATIOS GREATER THAN ONE(A NON-OSCILLATORY MODE) THE
C     FOLLOWING FOUR CARDS PREVENT TAKING THE SQUARE ROOT OF A NEGATIVE
C     NUMBER WHEN CALCULATING THE DAMPED NATURAL FREQUENCY.  IF THE
C     DAMPING RATIOS ARE GREATER THAN ONE THEN THE DAMPED NATURAL
C     FREQUENCIES REMAIN 0.0.
C
      WDSP=0.0
      WDP=0.0
      IF (DABS(ZSP).GT.1.0) GO TO 52
      WDSP=WNSP*DSQRT(1.0-ZSP*ZSP)
   52 WRITE (3,53) WNSP,WDSP,ZSP,T12SP,TO5SP
   53 FORMAT (1X,'*',18X,'DUTCH ROLL  ',F9.4,F9.4,F10.5,11X,F10.5,14X,F1
     10.5,25X,'*',/,1X,'*',128X,'*')
C
C     CALCULATION OF VALUES OF WF FOR FUTURE USE IN THE BODE ROUTINE.
C     INCLUDED ARE SELECTED VALUES OF WF(.01,.1,1.0,10.0,100.0,1000.0)
C     PLUS 5 VALUES AROUND EACH NATURAL FREQUENCY(2 ABOVE, 2 BELOW, AND
C     THE NATURAL FREQUENCY) TO INCREASE DATA IN THE BODE PLOT CRITICAL
C     AREAS.
C
      WF(1)=.01
      DO 54 IJ=2,6
      IM1=IJ-1
      WF(IJ)=WF(IM1)*10.0
   54 CONTINUE
C
C     KWF IS THE NUMBER OF NATURAL FREQUENCIES TO BE USED IN THE BODE
C     ROUTINE.
C
      KWF=6
C
C     II AND IK ARE COUNTERS USED TO DETERMINE THE MAXIMUM VALUE OF KWF
C     DEPENDING ON THE NUMBER OF NATURAL FREQUENCIES IN BOTH THE
C     NUMERATOR AND THE DENOMINATOR OF A PARTICULAR TRANSFER FUNCTION.
C
      II=0
      IK=0
      IF (I.EQ.0) GO TO 56
      IF (I.EQ.2) GO TO 55
      WF(12)=0.9*WNP
      WF(13)=0.75*WNP
      WF(14)=WNP
      WF(15)=1.1*WNP
      WF(16)=1.25*WNP
      IK=1
      KWF=16
   55 WF(7)=.9*WNSP
      WF(8)=.75*WNSP
      WF(9)=WNSP
      WF(10)=1.1*WNSP
      WF(11)=1.25*WNSP
      II=1
      IF (KWF.EQ.16) GO TO 56
      KWF=11
C
```

```
C     GETROT IS USED TO FIND ROOTS OF A PARTICULAR NUMERATOR DEPENDING
C     ON THE VALUE OF NUMER.
C
   56 IF (NUMER.EQ.1) GO TO 57
      IF (NUMER.EQ.2) GO TO 58
      CALL GETROT(NPSI,MN,RRN,RIN)
      GO TO 59
   57 CALL GETROT(NS,MN,RRN,RIN)
      GO TO 59
   58 CALL GETROT(NPHI,MN,RRN,RIN)
   59 WRITE (3,60)
   60 FORMAT (1X,'*',56X,'NUMERATOR ROOTS',57X,'*',/,1X,'*',128X,'*')
      WRITE (3,50) (J,RRN(J),RIN(J),J=1,MN)
      IF (NUMER.NE.3) GO TO 63
      WRITE (3,51)
C
C     IF THE DAMPING RATIO HAS AN ABSOLUTE VALUE GREATER THAN ONE(A NON-
C     OSCILLATORY MODE), THEN A DAMPED NATURAL FREQUENCY IS NOT
C     CALCULATED FOR THE NUMERATOR.  THEREFORE, WDSP AND WDP ARE LEFT AS
C     ZERO.
C
      WDSP=0.0
      WCP=0.0
      IF (DABS(ZSP).GT.1.0) GO TO 61
      WDSP=WNSP*DSQRT(1.0-ZSP*ZSP)
C
C     THE TWO WRITE STATEMENTS BELOW PRINT THE PERTINENT INFORMATION FOR
C     OSCILLATORY MODES IN THE NUMERATOR.  IF THE NATURAL FREQUENCIES
C     ARE PRINTED AS ZERO THE MODE IS NON-OSCILLATORY.
C
   61 WRITE (3,62) WNSP,WDSP,ZSP,T12SP,T05SP
   62 FORMAT (1X,'*',28X,F9.4,F11.4,3X,F10.4,11X,F10.5,14X,F10.5,22X,'*'
     1,/,1X,'*',128X,'*')
   63 WRITE (3,64)
   64 FORMAT (1X,'*',35X,58('-'),35X,'*',/,1X,'*',128X,'*',/,1X,'*',53X,
     1'BODE PLOT INFORMATION',54X,'*',/,1X,'*',128X,'*',/,1X,'*',21X,'FR
     1EQUENCY',26X,'AMPLITUDE RATIO',25X,'PHASE ANGLE',21X,'*',/,1X,'*',
     115X,'RAD/SEC',8X,'CYCLES/SEC',14X 'PURE',10X,'DECIBELS',15X,'RADIA
     1NS',8X,'DEGREES',15X,'*')
C
C     THE NEXT FEW CARDS ARE A ROUTINE TO FIND MORE VALUES OF WF FOR THE
C     BODE PLOT ROUTINE DEPENDING ON WHETHER OR NOT THE NUMERATOR HAS
C     ANY OSCILLATORY MODES.  THE FREQUENCIES AND THE VALUES OF KWF ARE
C     CHOSEN IN THE SAME MANNER AS THOSE OF THE DENOMINATOR PREVIOUSLY
C     MENTIONED.
C
C     MD1 AND MN1 ARE USED TO PREVENT HAVING ZERO SUBSCRIPTS WHEN
C     CALCULATING THE NUMERATOR AND DENOMINATOR GAINS FOR THE BODE PLOT
C     SUBROUTINE.
C
      MD1=MD+1
      IF (NUMER.NE.1)MD1=MD
      MN1=MN+1
      IF (I.EQ.0) GO TO 67
C
C     II AND IK ARE COUNTERS USED TO DETERMINE THE MAXIMUM VALUE OF KWF.
C
      IF (II.EQ.0.AND.IK.EQ.0) GO TO 66
      IF (II.EQ.1.AND.IK.EQ.0) GO TO 65
      WF(17)=.9*WNSP
      WF(18)=.75*WNSP
      WF(19)=WNSP
      WF(20)=1.1*WNSP
      WF(21)=1.25*WNSP
      KWF=21
      GO TO 67
   65 WF(12)=.9*WNSP
      WF(13)=.75*WNSP
      WF(14)=WNSP
      WF(15)=1.1*WNSP
      WF(16)=1.25*WNSP
      KWF=16
      GO TO 67
   66 WF(7)=.9*WNSP
      WF(8)=.75*WNSP
      WF(9)=WNSP
      WF(10)=1.1*WNSP
      WF(11)=1.25*WNSP
      KWF=11
   67 IF (NUMER.NE.1) GO TO 68
      KGAIN=NS(MN1)/DS(MD1)
C
C     THE GAIN (KGAIN) FOR THE ROOT LOCUS PLOTS IS CALCULATED FROM THE
C     COEFFICIENTS OF THE HIGHEST ORDER TERM IN THE DENOMINATOR AND
C     NUMERATOR, TF=KGAIN(S-A)(S-B)/(S-C)(S-D), WHERE A AND B ARE ROOTS
C     OF THE NUMERATOR AND C AND D ARE ROOTS OF THE DENOMINATOR.
C
      GO TO 70
   68 IF (NUMER.NE.2) GO TO 69
      KGAIN=NPHI(MN1)/DS(MD1)
      GO TO 70
   69 KGAIN=NPSI(MN1)/DS(MD1)
   70 KKWF=KWF
C
C     THE NEXT 12 CARDS RANK THE WF'S IN ASCENDING ORDER.
C
   71 WAX=WF(1)
      LK=1
      DO 72 JD=2,KKWF
      IF (WF(JD).GE.WAX)LK=JD
      IF (WF(JD).GE.WAX)WAX=WF(JD)
   72 CONTINUE
      WSAV=WF(KKWF)
      WF(KKWF)=WAX
      WF(LK)=WSAV
      KKWF=KKWF-1
      IF (KKWF.EQ.1) GO TO 73
      GO TO 71
C
C     BODE IS THE SUBROUTINE WHICH CALCULATES AMPLITUDE RATIO AND PHASE
C     ANGLE FOR EACH WF.  THE INFORMATION IS TRANSFERRED TO THE MAINLINE
C     BY THE USE OF A 'COMMON' STATEMENT.
C
   73 CALL BODE(MD,MN,KWF)
      WRITE (3,74) (WF(I),WCYCLE(I),AMPR(I),AROB(I),PHASE(I),PHDEG(I),I=
     11,KWF)
   74 FORMAT (1X,'*',13X,F10.5,6X,F10.5,12X,F10.2,6X,F10.5,12X,F10.5,6X,
     1F10.5,13X,'*')
      WRITE (3,75)
   75 FORMAT (1X,'*',128X,'*',/,1X,130('*'))
      IF (NUMER.NE.1)MD=MD-1
      GO TO 25
      END


      SUBROUTINE GETROT(COFFI,M,ROOTR,ROOTI)
C
```

```
C       GETROT IS A SUBROUTINE WHICH, USING OTHER SUBROUTINES, CALCULATES
C       ROOTS, DAMPING RATIOS, AND NATURAL FREQUENCIES, AND THESE ARE
C       TRANSFERRED TO THE MAINLINE BY USE OF A 'COMMON' STATEMENT.
C
        IMPLICIT REAL*8(A-H,O-Z)
        COMPLEX*16 P,TST
        REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
       1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU
        COMMON WNSP,ZSP,T12SP,T05SP,WNP,ZP,T12P,T05P,WF(21),RRD(10),RRN(10
       1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHOEG
       1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,R1,TR,ZV,YI,ETAV,B,SBS,CL
       1AVT,XM,H1,H2,W,WP,RHO,CLA2OW,SAH,BH,SH,TRH,CLA2OH,BA,CA,CH,SR,ALPH
       1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
       1DA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
        DIMENSION MUS(5),NAS(5),NTHS(5),DS(6),ROOTR(10),ROOTI(10),C(5),KC(
       14),QUF1(3),QUF2(3),RR1(2),RR2(2),RI1(2),RI2(2),CC(4),RR(3),COFFI(6
       1),RI(3),XR(3),XI(3),COF(3),RE(2),RIM(2),ROOT(1),COFF(2),KKK(21),P(
       16)
C
C       THE 3 'IF' STATEMENTS BELOW DECIDE WHICH ROOT-EXTRACTION
C       SUBROUTINE TO CALL DEPENDING ON THE VALUE OF M(THE ORDER OF THE
C       POLYNOMIAL).
C
        IF (M.EQ.4) GO TO 3
        IF (M.EQ.3) GO TO 2
        IF (M.EQ.2) GO TO 1
C
C       THE SUBROUTINES SINGLE, QUAD, CUBE, AND FOURTH SOLVE (THE SOLUTION
C       IS ACHIEVED IN CLOSED FORM AND THUS REQUIRES NO ITERATIVE
C       PROCEDURE) FOR ROOTS OF FIRST, SECOND, THIRD, AND FOURTH ORDER
C       POLYNOMIALS, RESPECTIVELY.
C
        CALL SINGLE(COFFI,ROOTR,ROOTI)
        GO TO 4
      1 CALL QUAD(COFFI,ROOTR,ROOTI)
        GO TO 4
      2 CALL CUBE(COFFI,ROOTR,ROOTI)
        GO TO 4
      3 CALL FOURTH(COFFI,ROOTR,ROOTI)
C
C       THE FOLLOWING CARDS TEST THE ROOTS OF THE POLYNOMIAL TO CHECK THE
C       ACCURACY OF THE ROOT SOLVER SUBROUTINES.  IF THE VALUE OF TST IS
C       TOO LARGE A WARNING MESSAGE IS PRINTED.
C
      4 DO 7 I=1,M
        P(I)=DCMPLX(ROOTR(I),ROOTI(I))
        ZERO=0.0
        TST=DCMPLX(COFFI(1),ZERO)
        MJ=M+1
        DO 5 J=2,MJ
        TST=TST+COFFI(J)*P(I)**(J-1)
      5 CONTINUE
        IF (CDABS(TST).LE.ACC) GO TO 7
        WRITE (3,6) P(I),TST
      6 FORMAT (1X,'ROOT = ',2G15.8,'    WHEN SUBSTITUTED INTO ITS POLYNOM
       1IAL FAILED TO COME WITHIN ACC OF 0.0.',/,1X,'THIS VALUE DIFFERED F
       1ROM ZERO BY ',2G15.8,'   THIS IMPLIES EITHER A ROUNDOFF ERROR WHEN
       1 TESTING THE ROOTS',/,1X,'(ACC TOO SMALL) OR THE VALUE 0.01 USED T
       1O COMPARE WITH TEST IN SUBROUTINE FOURTH IS TOO LARGE.')
      7 CONTINUE
C
C       THE NEXT 25 CARDS IS A PROCEDURE WHICH POSITIONS ROOTS WITH BOTH
C       REAL AND IMAGINARY PARTS IN THE FIRST L POSITIONS AND THE ROOTS
C       WITH ZERO IMAGINARY PARTS IN THE NEXT KK POSITIONS.  FOR EXAMPLE
C       IF THERE ARE 4 ROOTS, TWO WITH ONLY REAL PARTS AND TWO COMPLEX,
C       THE COMPLEX ROOTS WILL BE IN POSITIONS 1 AND 2 AND THE REAL ROOTS
C       WILL BE IN POSITIONS 3 AND 4.  L, K, AND KK ARE COUNTERS USED TO
C       FACILITATE THIS PROCEDURE.
C
C       I IS A COUNTER WHICH DETERMINES THE NUMBER OF ROOTS WHICH HAVE
C       BOTH A REAL AND AN IMAGINARY PART.
C
        I=M
C
C       N IS A COUNTER WHICH PREVENTS THE ORDER OF THE POLYNOMIAL FROM
C       BEING DESTROYED.
C
        N=M
        L=1
        K=M+1
        KK=0
        DO 9 J=1,N
        IF (DABS(ROOTI(J)).GT.ACC) GO TO 8
        I=I-1
        ROOTI(K)=ROOTI(J)
        ROOTR(K)=ROOTR(J)
        K=K+1
        KK=KK+1
        GO TO 9
      8 ROOTI(L)=ROOTI(J)
        ROOTR(L)=ROOTR(J)
        L=L+1
      9 CONTINUE
        IF (KK.EQ.0) GO TO 11
     10 KI=K-KK
        ROOTI(N)=ROOTI(KI)
        ROOTR(N)=ROOTR(KI)
        N=N-1
        KK=KK-1
        IF (KK.EQ.0) GO TO 11
        GO TO 10
C
C       AT THIS POINT THERE ARE I ROOTS THAT HAVE BOTH A REAL AND AN
C       IMAGINARY PART AND (M-I) ROOTS WITH JUST A REAL PART.
C
C       THIS PART OF THE PROGRAM DETERMINES THE LARGEST REAL PART OF THE
C       ROOTS AND RANKS THEM FROM THE BOTTOM IN THE I POSITIONS AVAILABLE.
C
     11 RMAX=ROOTR(1)
        K=1
        IF (I.EQ.0) GO TO 14
        DO 12 J=1,I
        IF (DABS(ROOTR(J)).GT.RMAX)K=J
        IF (DABS(ROOTR(J)).GT.RMAX)RMAX=ROOTR(J)
     12 CONTINUE
        RSAVER=ROOTR(I)
        RSAVEI=ROOTI(I)
        ROOTR(I)=ROOTR(K)
        ROOTI(I)=ROOTI(K)
        ROOTR(K)=RSAVER
        ROOTI(K)=RSAVEI
        N=I-1
        DO 13 J=1,N
        IF (DABS(RMAX-ROOTR(J)).LE.ACC)L=J
     13 CONTINUE
        RSAVER=ROOTR(N)
        RSAVEI=ROOTI(N)
        ROOTR(N)=ROOTR(L)
```

```
      ROOTI(N)=ROOTI(L)
      ROOTR(L)=RSAVER
      ROOTI(L)=RSAVEI
C
C     THE OUTPUT FOR BOTH NUMERATOR AND DENOMINATOR IS PRINTED IN A FORM
C     WHICH REQUIRES TWO OSCILLATORY MODES.  IF ONE OR BOTH OF THE MODES
C     ARE NON-OSCILLATORY THEN THE FOLLOWING PROCEDURE IS USED:
C          1)  THE DAMPING RATIO IS CHOSEN TO BE THE SMALLER MAGNITUDE OF
C              THE REAL ROOTS, SINCE THIS ROOT WILL DOMINATE IN THE TIME
C              DOMAIN (A NEGATIVE DAMPING RATIO WOULD INDICATE AN
C              UNSTABLE MODE).
C          2)  THE TIME TO DAMP TO 50% AND 5% OF THE AMPLITUDE ARE
C              CALCULATED BASED ON THE ABOVE DAMPING RATIO.  THUS, FOR AN
C              UNSTABLE SYSTEM THESE TIMES WILL BE NEGATIVE.
C
C     THE REMAINING PORTION OF GETROT CALCULATES THE NATURAL FREQUENCIES
C     (WNP & WNSP), DAMPING RATIOS(ZP & ZSP), TIME TO DAMP TO 1/2
C     AMPLITUDE(T12P & T12SP), AND SETTLING TIME(T05P & T05SP).  THE
C     SETTLING TIME IS THE TIME TO DAMP TO 5% OF THE ORIGINAL AMPLITUDE.
C     THE SUFFIXES P AND SP REFER TO OSCILLATORY MODES FOR THE NUMERATOR
C     OR THE DENOMINATOR DEPENDING ON THE EQUATION BEING SOLVED .
C
C     THE NATURAL FREQUENCIES, WNSP AND WNP, ARE DETERMINED BY A RANKING
C     OF THE MAGNITUDE OF THE REAL AND IMAGINARY PARTS OF THE ROOTS.
C     THE LARGER MAGNITUDE REPRESENTS WNSP.  IF THERE IS ONLY ONE
C     OSCILLATORY MODE THIS MODE IS REFERRED TO AS THE 'SP' SUFFIX MODE
C     AND THE 'P' SUFFIX MODE IS PRINTED AS ZERO.  WHEN GETROT IS USED
C     FOR A NUMERATOR POLYNOMIAL OR LATERAL DENOMINATOR THE 'SP'
C     INFORMATION IS PRINTED AS A NUMERATOR OR DENOMINATOR OSCILLATORY
C     MODE WHEN ONLY ONE OSCILLATORY MODE IS PRESENT.
C
   14 IF (M.EQ.1) GO TO 18
      IF (M.EQ.2.AND.I.EQ.0) GO TO 19
      IF (M.EQ.2.AND.I.EQ.2) GO TO 22
      IF (M.EQ.3.AND.I.EQ.0) GO TO 23
      IF (M.EQ.3.AND.I.EQ.2) GO TO 27
      IF (M.EQ.4.AND.I.EQ.0) GO TO 28
      IF (M.EQ.4.AND.I.EQ.2) GO TO 33
      WN1=DSQRT(ROOTR(3)*ROOTR(3)+ROOTI(3)*ROOTI(3))
      WN2=DSQRT(ROOTR(1)*ROOTR(1)+ROOTI(1)*ROOTI(1))
      IF (WN1.GT.WN2) GO TO 15
      WNSP=WN2
      NX=20
      WNP=WN1
      GO TO 16
   15 WNSP=WN1
      NX=10
      WNP=WN2
   16 IF (NX.NE.20) GO TO 17
      ZSP=-ROOTR(1)/WNSP
      T05SP=(2.9957)/(ZSP*WNSP)
      T12SP=(.693147)/(ZSP*WNSP)
      ZP=-ROOTR(3)/WNP
      T12P=(.693147)/(ZP*WNP)
      T05P=(2.9957)/(ZP*WNP)
      GO TO 34
   17 ZSP=-ROOTR(3)/WNSP
      T05SP=(2.9957)/(ZSP*WNSP)
      T12SP=(.693147)/(ZSP*WNSP)
      ZP=-ROOTR(1)/WNP
      T05P=(2.9957)/(ZP*WNP)
      T12P=(.693147)/(ZP*WNP)
      GO TO 34
   18 WNSP=0.0
      WNP=0.0
      ZSP=-ROOTR(1)
      ZP=0.0
      T12SP=.693147/ZSP
      T05SP=2.9957/ZSP
      T05P=0.0
      T12P=0.0
      GO TO 34
   19 IF (DABS(ROOTR(1)).GT.DABS(ROOTR(2))) GO TO 20
      ZSP=-ROOTR(1)
      ZP=-ROOTR(2)
      GO TO 21
   20 ZSP=-ROOTR(2)
      ZP=-ROOTR(1)
   21 WNSP=0.0
      WNP=0.0
      T12P=.693147/ZP
      T05P=2.9957/ZP
      T12SP=.693147/ZSP
      T05SP=2.9957/ZSP
      GO TO 34
   22 WNP=0.0
      ZP=0.0
      T12P=0.0
      T05P=0.0
      WNSP=DSQRT(ROOTR(1)*ROOTR(1)+ROOTI(1)*ROOTI(1))
      ZSP=-ROOTR(1)/WNSP
      T12SP=.693147/(ZSP*WNSP)
      T05SP=2.9957/(ZSP*WNSP)
      GO TO 34
   23 RAX=ROOTR(1)
      K=1
      DO 24 J=1,3
      IF (DABS(ROOTR(J)).GT.RAX)K=J
      IF (DABS(ROOTR(J)).GT.RAX)RAX=ROOTR(J)
   24 CONTINUE
      RSAV=ROOTR(3)
      ROOTR(3)=RAX
      ROOTR(K)=RSAV
      IF (DABS(ROOTR(1)).GT.DABS(ROOTR(2))) GO TO 25
      ZSP=-ROOTR(1)
      ZP=-ROOTR(2)
      GO TO 26
   25 ZSP=-ROOTR(2)
      ZP=-ROOTR(1)
   26 WNSP=0.0
      WNP=0.0
      T12P=.693147/ZP
      T05P=2.9957/ZP
      T12SP=.693147/ZSP
      T05SP=2.9957/ZSP
      GO TO 34
   27 WNP=0.0
      ZP=-ROOTR(3)
      T12P=.693147/ZP
      T05P=2.9957/ZP
      WNSP=DSQRT(ROOTR(1)*ROOTR(1)+ROOTI(1)*ROOTI(1))
      ZSP=-ROOTR(1)/WNSP
      T12SP=.693147/(ZSP*WNSP)
      T05SP=2.9957/(ZSP*WNSP)
      GO TO 34
   28 RAX=ROOTR(1)
      K=1
      DO 29 J=1,4
```

```
      IF (DABS(ROOTR(J)).GT.RAX)K=J
      IF (DABS(ROOTR(J)).GT.RAX)RAX=ROOTR(J)
   29 CONTINUE
      RSAV=ROOTR(4)
      ROOTR(4)=RAX
      ROOTR(K)=RSAV
      RAX=ROOTR(1)
      K=1
      DO 30 J=1,3
      IF (DABS(ROOTR(J)).GT.RAX)K=J
      IF (DABS(ROOTR(J)).GT.RAX)RAX=ROOTR(J)
   30 CONTINUE
      RSAV=ROOTR(3)
      ROOTR(3)=RAX
      ROOTR(K)=RSAV
      IF (DABS(ROOTR(1)).GT.DABS(ROOTR(2))) GO TO 31
      ZSP=-ROOTR(1)
      ZP=-ROOTR(2)
      GO TO 32
   31 ZSP=-ROOTR(2)
      ZP=-ROOTR(1)
   32 WNSP=0.0
      WNP=0.0
      T12P=.693147/ZP
      T05P=2.9957/ZP
      T12SP=.693147/ZSP
      T05SP=2.9957/ZSP
      GO TO 34
   33 WNSP=DSQRT(ROOTR(1)*ROOTR(1)+ROOTI(1)*ROOTI(1))
      ZSP=-ROOTR(1)/WNSP
      WNP=0.0
      T12SP=.693147/(ZSP*WNSP)
      T05SP=2.9957/(ZSP*WNSP)
   34 RETURN
      END


      SUBROUTINE FOURTH(C,ROOTR,ROOTI)
C
C     THIS SUBROUTINE FACTORS A FOURTH ORDER POLYNOMIAL BY A CLOSED FORM
C     PROCEDURE WHICH FORMS 2 QUADRATIC FACTORS AND THEN CALLS A
C     QUADRATIC FACTORING SUBROUTINE, QUAD, TO OBTAIN THE FOUR ROOTS.
C     THE PROCEDURE WAS TAKEN FROM 'INTRODUCTION TO THE THEORY OF
C     EQUATIONS' BY N.B. CONKWRIGHT.
C
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 MS,IYY,MU,MW,MDW,MQ,MIN,MUS,NAS,NTHS,KGAIN,KC,KROOT,KKK
      DIMENSION C(5),KC(4),QUF1(3),QUF2(3),RR1(2),RR2(2),RI1(2),RI2(2),R
     1OOTR(10),ROOTI(10)
      P=C(4)/C(5)
      Q=C(3)/C(5)
      R=C(2)/C(5)
      S=C(1)/C(5)
      KC(4)=1.0
      KC(3)=-.5*Q
      KC(2)=.25*(P*R-4.0*S)
      KC(1)=.125*(4.0*Q*S-P*P*S-R*R)
      HKC=(3.0*KC(4)*KC(2)-KC(3)*KC(3))/9.0*KC(4)*KC(4)
      GKC=(2.0*KC(3)*KC(3)*KC(3)-9.0*KC(4)*KC(3)*KC(2)+27.0*KC(4)*KC(4)*
     1KC(1))/(27.0*KC(4)*KC(4)*KC(4))
      RAD=GKC*GKC+4.0*HKC*HKC*HKC
      IF (RAD.LT.0.0) GO TO 3
      UKC=((-GKC+DSQRT(GKC*GKC+4.0*HKC*HKC*HKC))/2.0)
      RTUKC=(DABS(UKC))**0.333333333333333
      UKC=DSIGN(RTUKC,UKC)
      VKC=-HKC/UKC
      KROOT=UKC+VKC-KC(3)/(3.0*KC(4))
    1 B=DSQRT(KROOT*KROOT-S)
      A=DSQRT(2.0*KROOT+P*P*.25-Q)
      TEST=2.0*A*B+R-KROOT*P
      IF (TEST.LE..01) GO TO 2
      A=-A
      TEST=2.0*A*B+R-KROOT*P
      IF (TEST.LE..01) GO TO 2
      A=-A
      B=-B
      TEST=2.0*A*B+R-KROOT*P
      IF (TEST.LE..01) GO TO 2
      A=-A
    2 QUF1(3)=1.0
      QUF2(3)=1.0
      QUF1(2)=.5*P-A
      QUF2(2)=.5*P+A
      QUF1(1)=KROOT-B
      QUF2(1)=KROOT+B
      CALL QUAD(QUF1,RR1,RI1)
      ROOTR(1)=RR1(1)
      ROOTR(2)=RR1(2)
      ROOTI(1)=RI1(1)
      ROOTI(2)=RI1(2)
      CALL QUAD(QUF2,RR2,RI2)
      ROOTR(3)=RR2(1)
      ROOTR(4)=RR2(2)
      ROOTI(3)=RI2(1)
      ROOTI(4)=RI2(2)
      RETURN
    3 THETA=DARCOS(-GKC/(2.0*DSQRT(-HKC*HKC*HKC)))
      TH3=THETA/3.0
      KROOT=2.0*DCOS(TH3)*DSQRT(-HKC)-KC(3)/(3.0*KC(4))
      GO TO 1
      END


      SUBROUTINE CUBE(CC,XR,XI)
C
C     THIS SUBROUTINE FACTORS A THIRD ORDER POLYNOMIAL BY A CLOSED FORM
C     PROCEDURE GIVEN IN 'INTRODUCTION TO THE THEORY OF EQUATIONS' BY
C     CONKWRIGHT AND MODIFIED BY THE PROCEDURE GIVEN IN 'STANDARD MATH
C     TABLES' BY CHEMICAL RUBBER COMPANY.
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION CC(4),RR(3),RI(3),XR(3),XI(3)
      H=(3.0*CC(4)*CC(2)-CC(3)*CC(3))/(9.0*CC(4)*CC(4))
      G=(2.0*CC(3)*CC(3)*CC(3)-9.0*CC(4)*CC(3)*CC(2)+27.0*CC(4)*CC(4)*CC
     1(1))/(27.0*CC(4)*CC(4)*CC(4))
      RAD=G*G+4.0*H*H*H
      IF (RAD.LT.0.0) GO TO 3
      UROOT3=(-G+DSQRT(G*G+4.0*H*H*H))/2.0
      IF (UROOT3.LT.0.0) GO TO 1
      UROOT=UROOT3**.33333333333333
      GO TO 2
    1 UROOT=(-UROOT3)**.33333333333333
      UROOT=-UROOT
    2 VROOT=-H/UROOT
      RR(1)=UROOT+VROOT
      RR(2)=-(UROOT+VROOT)/2.0
```

```
      RR(3)=-(UROOT+VROOT)/2.0
      RI(1)=0.0
      RI(2)=(UROOT-VROOT)*(.866025403784)
      RI(3)=-(UROOT-VROOT)*(.866025403784)
      XR(1)=RR(1)-CC(3)/(3.0*CC(4))
      XR(2)=RR(2)-CC(3)/(3.0*CC(4))
      XR(3)=RR(3)-CC(3)/(3.0*CC(4))
      XI(1)=RI(1)
      XI(2)=RI(2)
      XI(3)=RI(3)
      GO TO 4
    3 THETA=DARCOS(-G/(2.0*DSQRT(-H*H*H)))
      TH3=THETA/3.0
      XI(1)=0.0
      XI(2)=0.0
      XI(3)=0.0
      XR(1)=2.0*DCOS(TH3)*DSQRT(-H)
      XR(2)=2.0*DCOS(TH3+2.094395102392)*DSQRT(-H)
      XR(3)=2.0*DCOS(TH3+4.188790204784)*DSQRT(-H)
      SUB=CC(3)/(3.0*CC(4))
      XR(1)=XR(1)-SUB
      XR(2)=XR(2)-SUB
      XR(3)=XR(3)-SUB
    4 RETURN
      END


      SUBROUTINE QUAD(COF,RE,RIM)
C
C     THIS SUBROUTINE FACTORS A SECOND ORDER POLYNOMIAL BY USING THE
C     QUADRATIC FORMULA.
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION COF(3),RE(2),RIM(2)
      DIS=COF(2)*COF(2)-4.0*COF(3)*COF(1)
      IF (DIS.LT.0.0) GO TO 1
      RE(1)=(-COF(2)+DSQRT(DIS))/(2.0*COF(3))
      RE(2)=(-COF(2)-DSQRT(DIS))/(2.0*COF(3))
      RIM(1)=0.0
      RIM(2)=0.0
      GO TO 2
    1 RE(1)=-COF(2)/(2.0*COF(3))
      RE(2)=RE(1)
      RIM(1)=(-DSQRT(-DIS))/(2.0*COF(3))
      RIM(2)=(DSQRT(-DIS))/(2.0*COF(3))
    2 RETURN
      END


      SUBROUTINE SINGLE(COFF,ROOT,ROOTI)
C     THIS SUBROUTINE FACTORS A FIRST ORDER POLYNOMIAL.
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION ROOT(1),COFF(2),ROOTI(1)
      ROOT(1)=-COFF(1)/COFF(2)
      ROOTI(1)=0.0
      RETURN
      END


      SUBROUTINE BODE(L,JJ,KWF)
C
C     THIS IS A SUBROUTINE WHICH CALCULATES THE INFORMATION NEEDED TO
C     CONSTRUCT A BODE PLOT ONCE THE NUMERATOR AND DENOMINATOR
C     POLYNOMIALS HAVE BEEN FACTORED.  THE INFORMATION IS TRANSFERRED TO
C     THE MAINLINE BY USING A 'COMMON' STATEMENT.  THE FREQUENCIES ARE
C     GIVEN BOTH IN RADIANS PER SECOND AND CYCLES PER SECOND.  THE
C     AMPLITUDE RATIO IS GIVEN BOTH AS A PURE NUMBER AND IN DECIBELS.
C     THE PHASE ANGLE IS GIVEN BOTH IN DEGREES AND RADIANS.
C
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
     1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU,KDSAVE,KNSAVE
      COMMON WNSP,ZSP,T12SP,T05SP,WNP,ZP,T12P,T05P,WF(21),RRD(10),RRN(10
     1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
     1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,RL,TR,ZV,YI,ETAV,B,SBS,CL
     1AVT,XM,H1,H2,W,WP,KMU,CLA2DW,SAH,BH,SH,TRH,CLA2DH,BA,CA,CH,SR,ALPH
     1A,COO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
     1CA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
      DIMENSION KKK(21)
      DO 9 M=1,KWF
      AMPD=1.0
      KD=1.0
      PHASED=0.0
      DO 4 J=1,L
      IF (RRD(J).EQ.0.0) GO TO 1
      GO TO 3
    1 AMPD=AMPD*DABS(WF(M)-RID(J))
      IF (WF(M)-RID(J).LT.0.0) GO TO 2
      PHASED=PHASED-3.1415926536/2.0
      GO TO 4
    2 PHASED=PHASED+3.1415926536/2.0
      GO TO 4
    3 AMPD=DSQRT(((-WF(M)+RID(J))/RRD(J))**2+1.0)*AMPD
      PHASED=-(DATAN((-RID(J)+WF(M))/(-RRD(J))))+PHASED
      KDSAVE=KD
      KD=-RRD(J)*KD
      IF (KDSAVE*KD.LT.0.0)PHASED=PHASED-3.1415926536
    4 CONTINUE
      PHASEN=0.0
      KN=KGAIN
      AMPN=1.0
      DO 8 I=1,JJ
      IF (RRN(I).EQ.0.0) GO TO 5
      GO TO 7
    5 AMPN=AMPN*DABS(WF(M)-RIN(I))
      IF (WF(M)-RIN(I).LT.0.0) GO TO 6
      PHASEN=PHASEN+3.1415926536/2.0
      GO TO 8
    6 PHASEN=PHASEN-3.1415926536/2.0
      GO TO 8
    7 AMPN=DSQRT(((-WF(M)+RIN(I))/RRN(I))**2+1.0)*AMPN
      PHASEN=DATAN((-RIN(I)+WF(M))/(-RRN(I)))+PHASEN
      KNSAVE=KN
      KN=-RRN(I)*KN
      IF (KNSAVE*KN.LT.0.0)PHASEN=PHASEN+3.1415926536
    8 CONTINUE
      KKK(M)=KN/KD
      AMPR(M)=DABS(KKK(M))*AMPN/AMPD
      ARDB(M)=20.0*DLOG10(AMPR(M))
      PHASE(M)=PHASED+PHASEN
      PHDEG(M)=PHASE(M)*57.295779513
      WCYCLE(M)=WF(M)/(2.0*3.1415926536)
    9 CONTINUE
      RETURN
```

```
      END

      SUBROUTINE SCYB
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
     1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU
      COMMON WNSP,ZSP,T12SP,T05SP,WNP,ZP,T12P,T05P,WF(21),RRD(10),RRN(10
     1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
     1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,R1,TR,ZV,YI,ETAV,B,SBS,CL
     1AVT,XM,H1,H2,W,WP,RHO,CLA2OW,SAH,BH,SH,TRH,CLA2OH,BA,CA,CH,SR,ALPH
     1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
     1DA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
      CYBWS=CL*CL*6.0*DTAN(SA)*DSIN(SA)/(3.14159*AR*(AR+4.0*DCOS(SA)))
      CYBWD=-.0001*DABS(DIH)*57.2958
      CLAFUS=.1
      BRA=FUSVOL**.66666666666666
      IF (ZW.LT.0.0) GO TO 1
      KI=1.0+.485*(ZW*2.0/H)
      GO TO 2
    1 KI=1.0-.85555*(ZW*2.0/H)
    2 CYBFUS=-KI*CLAFUS*BRA/S
      SWOPR=.724+3.06*SV/(S*(1.0+DCOS(SA)))+.4*ZW/H+.009*AR
      PAR=BV/(2.0*R1)
      KY=1.0+.166*(PAR-3.5)
      IF (PAR.LE.2.0)KY=.75
      IF (PAR.GE.3.5)KY=1.0
      CYBT=-KY*CLAVT*SWOPR*SV/S
      CYB=CYBWS+CYBWD+CYBFUS+CYBT
      RETURN
      END

      SUBROUTINE SCLB
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
     1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU
      COMMON WNSP,ZSP,T12SP,T05SP,WNP,ZP,T12P,T05P,WF(21),RRD(10),RRN(10
     1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
     1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,R1,TR,ZV,YI,ETAV,B,SBS,CL
     1AVT,XM,H1,H2,W,WP,RHO,CLA2OW,SAH,BH,SH,TRH,CLA2OH,BA,CA,CH,SR,ALPH
     1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
     1DA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
      IF (AR.GE.1.5.AND.AR.LE.10.) GO TO 2
      WRITE (3,1) AR
    1 FORMAT (1X,'VALUE OF AR = ',G11.4,2X,'USED TO CALCULATE CLB IS OUT
     1SIDE PREFERRED RANGE OF 1.5 TO 10.0')
    2 CLBOD=((-.20833-DSQRT(.0434-4.6296*(1.0052-AR)))/2.3148148+.7*(1.0
     1-TR)*(1.0-TR)*(AR-1.5)/6.5)*.0001
      CLBWD=CLBOD*DIH*57.2958
      CLBWWD=CL*(-1.25*(.71*TR+.29)/(AR*TR)+.05)
      CLBW=CLBWD+CLBWWD
      CLBVT=-CLAVT*SV*ZV*ETAV/(S*B)
      IF (WP.NE.1.0) GO TO 3
      DCLB=-.00044
      GO TO 5
    3 IF (WP.NE.2.0) GO TO 4
      DCLB=0.0
      GO TO 5
    4 DCLB=.00064
    5 CLB=CLBW+CLBVT+DCLB
      RETURN
```

```
      END

      SUBROUTINE SCNB
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
     1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU
      COMMON WNSP,ZSP,T12SP,T05SP,WNP,ZP,T12P,T05P,WF(21),RRD(10),RRN(10
     1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
     1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,R1,TR,ZV,YI,ETAV,B,SBS,CL
     1AVT,XM,H1,H2,W,WP,RHO,CLA2OW,SAH,BH,SH,TRH,CLA2OH,BA,CA,CH,SR,ALPH
     1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
     1DA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
      A=XM/LF
      IF (A.GE..1.AND.A.LE..8) GO TO 2
      WRITE (3,1) A
    1 FORMAT (1X,'VALUE OF XM/LF = ',G11.4,'   USED TO CALCULATE CNB IS
     1OUTSIDE PERFERRED RANGE OF 0.1 TO 0.8')
    2 X=(A+.2)*10.0
      XP=X-3.0
      R=LF*LF/SBS
      IF (R.GE.2.5.AND.R.LE.20.) GO TO 4
      WRITE (3,3) R
    3 FORMAT (1X,'VALUE OF LF*LF/SBS = ',G11.4,'   USED TO CALCULATE CNB
     1 IS OUTSIDE PREFERRED RANGE OF 2.5 TO 20.0')
    4 YO=-.0015609*R**3+.0675772*R**2-.999884*R+1.20213
      YP=.314*XP+YO
      Y=12.0+YP
      YP=Y-8.0
      AA=DSQRT(H1/H2)
      IF (AA.GE..8.AND.AA.LE.1.65) GO TO 6
      WRITE (3,5) AA
    5 FORMAT (1X,'VALUE OF SQRT(H1/H2) = ',G11.4,'  USED TO CALCULATE CN
     1B IS OUTSIDE PREFERRED RANGE OF 0.8 TO 1.65')
    6 XP=AA*YP
      AB=H/W
      IF (AB.GE..5.AND.AB.LE.2.0) GO TO 8
      WRITE (3,7) AB
    7 FORMAT (1X,'VALUE OF H/W = ',G11.4,'   USED TO CALCULATE CNB IS OU
     1TSIDE PREFERRED RANGE OF 0.5 TO 2.0')
    8 YP=-((H/W)**(.48*W/H))*XP
      Y1=YP
      RHO1=RHO*10000.
      MU=.34777*RHO1+29.23345
      RN=RHO1*U*LF/(MU*.0001)
      AC=RN*.000001
      IF (AC.GE.0.0.AND.AC.LE.80.0) GO TO 10
      WRITE (3,9) AC
    9 FORMAT (1X,'VALUE OF RN*10-6 = ',G11.4,'   USED TO CALCULATE CNB I
     1S OUTSIDE PREFERRED RANGE OF 0.0 TO 80.0')
   10 YP=AC/20.0
      X=.00833327*YP**3-.113571*YP*YP+1.02095*YP+.446857
      B1=X/2.0
      X=-2.0*(Y1-B1)
      KB=(X-1.0)*.0005
      CNB=-KB*SBS*LF*57.298/(S*B)-CYBT*LT/B
      RETURN
      END

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

```
      REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
     1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU
      COMMON WNSP,ZSP,T12SP,T05SP,WNP,ZP,T12P,T05P,WF(21),RRD(10),RRN(10
     1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
     1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,RL,TR,ZV,YI,ETAV,B,SBS,CL
     1AVT,XM,H1,H2,W,WP,RHO,CLA2DW,SAH,BH,SH,TRH,CLA2DH,BA,CA,CH,SR,ALPH
     1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
     1DA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
      IF (SA.GE.-.524.AND.SA.LE..524) GO TO 2
      WRITE (3,1) SA
    1 FORMAT (1X,'VALUE OF SA = ',G11.4,'   USED TO CALCULATE CYP IS OUT
     1SIDE PREFERRED RANGE OF -.524 TO .524 RADIANS')
    2 IF (AR.GE.5.) GO TO 4
      WRITE (3,3) AR
    3 FORMAT (1X,'VALUE OF AR = ',G11.4,'  USED TO CALCULATE CYP IS LESS
     1 THAN THE SUGGESTED MINIMUM OF 5.0;',/,1X,'HOWEVER, DUE TO THE NEG
     1LIGIBLE EFFECT OF THIS DERIVATIVE, ONE MAY IGNORE THE EFFECT OF SM
     1ALLER AR ON CYP')
    4 X=-(SA*57.3-30.0)*7.0/60.0
      Y=X+3.0*(1.0-TR**1.58495)
      CYPOCL=(4.0-Y)/10.0
      IF (DIH.GE.-10..AND.DIH.LE.10.) GO TO 6
      WRITE (3,5) DIH
    5 FORMAT (1X,'VALUE OF DIH = ',G11.4,'   USED TO CALCULATE CYP IS OU
     1TSIDE PREFERRED RANGE OF -10.0 TO 10.0 DEGREES')
    6 P=(DIH-10.)/17.5+.5
      CYP=CYPOCL*CL+P*CLP
      RETURN
      END


      SUBROUTINE SCLP
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
     1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU
      COMMON WNSP,ZSP,T12SP,T05SP,WNP,ZP,T12P,T05P,WF(21),RRD(10),RRN(10
     1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
     1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,RL,TR,ZV,YI,ETAV,B,SBS,CL
     1AVT,XM,H1,H2,W,WP,RHO,CLA2DW,SAH,BH,SH,TRH,CLA2DH,BA,CA,CH,SR,ALPH
     1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
     1DA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
      IF (AR.GE.0.0.AND.AR.LE.10.) GO TO 2
      WRITE (3,1) AR
    1 FORMAT (1X,'VALUE OF AR = ',G11.4,'  USED TO CALCULATE CLP IS OUTS
     1IDE PREFERRED RANGE OF 0.0 TO 10.0')
    2 CLPAO=(.00000929805*AR**5-.00031818*AR**4+.00379426*AR**3-.0157796
     1*AR*AR-.0499261*AR-.0500059)+.209*(1.0-TR)**2*(AR-1.0)/9.0
      CLPAOW=CLPAO*(AR+4.0*DCOS(SA))/((2.0*3.14159*AR/CLA2DW)+4.0*DCOS(S
     1A))
      CLPW=CLPAOW-(CL*CL/(8.0*3.14159*AR*DCOS(SA)**2))*(1.0+2.0*DSIN(SA)
     1**2*(AR+2.0*DCOS(SA))/(AR+4.0*DCOS(SA)))-CDO/8.0
      CLPAOH=(.00000929805*ARH**5-.00031818*ARH**4+.00379426*ARH**3-.015
     17796*ARH**2-.0499261*ARH-.0500059)+.209*(1.0-TRH)**2*(ARH-1.0)/9.0
      CLPH=((.5*SH*BH*BH)/(S*B*B))*(CLPAOH*(ARH+4.0*DCOS(SAH))/((2.0*3.1
     14159*ARH/CLA2DH)+4.0*DCOS(SAH)))
      CLPV=2.0*ZV*ZV*CYBT/(B*B)
      CLP=CLPW+CLPH+CLPV
      RETURN
      END


      SUBROUTINE SCNP
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
     1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU
      COMMON WNSP,ZSP,T12SP,T05SP,WNP,ZP,T12P,T05P,WF(21),RRD(10),RRN(10
     1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
     1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,RL,TR,ZV,YI,ETAV,B,SBS,CL
     1AVT,XM,H1,H2,W,WP,RHO,CLA2DW,SAH,BH,SH,TRH,CLA2DH,BA,CA,CH,SR,ALPH
     1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
     1DA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
      IF (AR.LE.16.0) GO TO 2
      WRITE (3,1) AR
    1 FORMAT (1X,'VALUE OF AR = ',G11.4,'   USED TO CALCULATE CNP IS OUT
     1SIDE PREFERRED RANGE OF 0.0 TO 16.0')
    2 CNPOCL=-.0000140468*AR**3+.000649352*AR*AR-.0122523*AR+.00192856
      CNPW=(CL*(AR+4.0)/(AR+4.0*DCOS(SA)))*(1.0+6.0*(1.0+DCOS(SA)/AR)*DT
     1AN(SA)*DTAN(SA)/12.0)*CNPOCL
      P=ZV*2.0/B
      IF (P.GE.0.0.AND.P.LE..6) GO TO 4
      WRITE (3,3) P
    3 FORMAT (1X,'VALUE OF ZV/(B/2) = ',G11.4,'   USED TO CALCULATE CNP
     1IS OUTSIDE PREFERRED RANGE OF 0.0 TO 0.6')
    4 SIG1=22360.7*P**9-69947.1*P**8+93669.5*P**7-70229.7*P**6+32373.8*P
     1**5-9487.01*P**4+1764.41*P**3-200.25*P*P+11.9843*P-.00142786
      IF (P.GE..157) GO TO 5
      SIG2=216915.*P**9-610122.*P**8+724643.*P**7-473060.*P**6+185140.*P
     1**5-44463.6*P**4+6444.43*P**3-531.134*P**2+21.1205*P+.0395735
      GO TO 6
    5 SIG2=.533183*P*P-.78216*P+.35117
    6 SIGMA1=SIG2+TR*(SIG1-SIG2)
      DELHOB=(ZV-(ZV*DCOS(ALPHA)-LT*DSIN(ALPHA)))/B
      SIGMA2=9.3*DELHOB*DELHOB*3.0/B
      CNPV=(1.00*CLAVT*SV*(ZV*DSIN(ALPHA)+LT*DCOS(ALPHA))/(S*B))*(2.0*(Z
     1V*DCOS(ALPHA)-LT*DSIN(ALPHA))/B-(SIGMA1+SIGMA2))
      CNP=CNPW+CNPV
      RETURN
      END


      SUBROUTINE SCYR
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
     1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU
      COMMON WNSP,ZSP,T12SP,T05SP,WNP,ZP,T12P,T05P,WF(21),RRD(10),RRN(10
     1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
     1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,RL,TR,ZV,YI,ETAV,B,SBS,CL
     1AVT,XM,H1,H2,W,WP,RHO,CLA2DW,SAH,BH,SH,TRH,CLA2DH,BA,CA,CH,SR,ALPH
     1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
     1DA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
      CYRW=.143*CL-.05
      CYRT=-2.0*LT*CYBT/B
      CYR=CYRW+CYRT
      RETURN
      END


      SUBROUTINE SCLR
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
     1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU
      COMMON WNSP,ZSP,T12SP,T05SP,WNP,ZP,T12P,T05P,WF(21),RRD(10),RRN(10
     1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
     1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,RL,TR,ZV,YI,ETAV,B,SBS,CL
```

```
 1AVT,XM,H1,H2,W,WP,RHO,CLA2DW,SAH,BH,SH,TRH,CLA2DH,BA,CA,CH,SR,ALPH
 1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
 1DA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
  IF (AR.GE.2.0.AND.AR.LE.12.) GO TO 2
  WRITE (3,1) AR
1 FORMAT (1X,'VALUE OF AR = ',G11.4,'    USED TO CALCULATE CLR IS OUT
 1SIDE PREFERRED RANGE OF 2.0 TO 12.0')
2 YP=-.0000247674*AR**4+.00194154*AR**3-.0468052*AR*AR+.456404*AR-.0
 1980299+2.15*(AR+32.4)*(TR**(1.12*(1.0-TR)))/44.4
  Y=YP+2.0
  IF (SA.GE.0.0.AND.SA.LE..524) GO TO 4
  WRITE (3,3) SA
3 FORMAT (1X,'VALUE OF SA = ',G11.4,'    USED TO CALCULATE CLR IS OUT
 1SIDE PREFERRED RANGE OF 0.0 TO 0.524 RADIANS')
4 X=100.0*Y/(100.0-SA*57.3)
  CLROCL=X/20.0
  CLRW=CLROCL*CL
  CLRT=-2.0*LT*ZV*CYBT/(B*B)
  CLR=CLRW+CLRT
  RETURN
  END


  SUBROUTINE SCNR
  IMPLICIT REAL*8(A-H,O-Z)
  REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
 1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU
  COMMON WNSP,ZSP,T12SP,TO5SP,WNP,ZP,T12P,TO5P,WF(21),RRD(10),RRN(10
 1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
 1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,R1,TR,ZV,YI,ETAV,B,SBS,CL
 1AVT,XM,H1,H2,W,WP,RHO,CLA2DW,SAH,BH,SH,TRH,CLA2DH,BA,CA,CH,SR,ALPH
 1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
 1DA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
  CNR=2.0*LT*LT*CYBT/(B*B)-(.33*(1.0+3.0*TR)*CDO/(2.0+2.0*TR)+.02*(1
 1.0-(AR-6.0)/13.0-(1.0-TR)/2.5)*CL*CL)
  RETURN
  END


  SUBROUTINE SCYDA
  IMPLICIT REAL*8(A-H,O-Z)
  REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
 1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU
  COMMON WNSP,ZSP,T12SP,TO5SP,WNP,ZP,T12P,TO5P,WF(21),RRD(10),RRN(10
 1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
 1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,R1,TR,ZV,YI,ETAV,B,SBS,CL
 1AVT,XM,H1,H2,W,WP,RHO,CLA2DW,SAH,BH,SH,TRH,CLA2DH,BA,CA,CH,SR,ALPH
 1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
 1DA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
  CYDA=0.0
  RETURN
  END


  SUBROUTINE SCLDA
  IMPLICIT REAL*8(A-H,O-Z)
  REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
 1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU
  COMMON WNSP,ZSP,T12SP,TO5SP,WNP,ZP,T12P,TO5P,WF(21),RRD(10),RRN(10
 1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
 1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,R1,TR,ZV,YI,ETAV,B,SBS,CL
 1AVT,XM,H1,H2,W,WP,RHO,CLA2DW,SAH,BH,SH,TRH,CLA2DH,BA,CA,CH,SR,ALPH
 1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
 1DA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
  IF (AR.GE.4.0.AND.AR.LE.12.0) GO TO 2
  WRITE (3,1) AR
1 FORMAT (1X,'VALUE OF AR = ',G11.4,'    USED TO CALCULATE CLDA IS O
 1UTSIDE PREFERRED RANGE OF 4.0 TO 12.0')
2 K=0
  P=YI*2.0/B
3 CLDABT=51.1702*P**8-238.294*P**7+451.162*P**6-444.985*P**5+241.797
 1*P**4-70.2549*P**3+10.5819*P*P-.408045*P+.0000119225
  CLDAAT=2.0833*P**5-6.1553*P**4+5.36932*P**3-.567235*P*P+.170341*P-
 1.000227237
  IF (AR.GT.6.0.AND.AR.LT.10.0) GO TO 4
  IF (AR.LE.6.0)CLDAOT=CLDABT
  IF (AR.GE.10.0)CLDAOT=CLDAAT
  GO TO 5
4 CLDAOT=CLDABT+(CLDAAT-CLDABT)*(AR-6.0)/4.0
5 IF (K.EQ.1) GO TO 6
  CLDALT=CLDAOT
  P=(YI+BA)*2.0/B
  K=K+1
  GO TO 3
6 CLDAOT=CLDAOT-CLDALT
  R=CA/CH
  IF (R.GE.0.0.AND.R.LE..4) GO TO 8
  WRITE (3,7) R
7 FORMAT (1X,'VALUE OF CA/CH = ',G11.4,'    USED TO CALCULATE CLDA IS
 1 OUTSIDE PREFERRED RANGE OF 0.0 TO 0.4')
8 T=-65.1062*R**4+50.8227*R**3-15.7949*R*R+3.53383*R+.000043467
  CLDA=CLDAOT*T
  RETURN
  END


  SUBROUTINE SCNDA
  IMPLICIT REAL*8(A-H,O-Z)
  REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
 1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU
  COMMON WNSP,ZSP,T12SP,TO5SP,WNP,ZP,T12P,TO5P,WF(21),RRD(10),RRN(10
 1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
 1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,R1,TR,ZV,YI,ETAV,B,SBS,CL
 1AVT,XM,H1,H2,W,WP,RHO,CLA2DW,SAH,BH,SH,TRH,CLA2DH,BA,CA,CH,SR,ALPH
 1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
 1CA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
  R=YI*2.0/B
  IF (AR.GE.3.0.AND.AR.LE.8.0) GO TO 2
  WRITE (3,1) AR
1 FORMAT (1X,'VALUE OF AR = ',G11.4,'    USED TO CALCULATE CNDA IS OU
 1TSIDE PREFERRED RANGE OF 3.0 TO 8.0')
2 IF (AR.LT.3.0) GO TO 3
  IF (AR.GE.3.0.AND.AR.LE.4.0) GO TO 4
  IF (AR.GT.4.0.AND.AR.LT.6.0) GO TO 5
  IF (AR.GT.6.0.AND.AR.LE.8.0) GO TO 6
  EF1=-.110526*R*R+.013893*R-.146355
  EF2=-.191365*R**3+.147284*R*R-.00039296*R-.119884
  GO TO 7
3 EF1=-.36
  EF2=-.0889976*R**3-.00666109*R*R+.0419053*R-.284843
  GO TO 7
4 AF1=-.36
  BF1=.0538037*R**3-.133855*R*R+.00627854*R-.262321
  EF1=AF1+(AR-3.0)*(BF1-AF1)
```

```
      AF2=-.0889976*R**3-.00666109*R*R+.0419053*R-.28483
      BF2=.0549387*R**3-.164298*R*R+.101864*R-.234861
      EF2=AF2+(AR-3.0)*(BF2-AF2)
      GO TO 7
    5 AF1=.0538037*R**3-.133855*R*R+.00627854*R-.262321
      BF1=-.0629823*R**3-.0173375*R*R-.0333427*R-.180026
      EF1=AF1+((AR-4.0)/2.0)*(BF1-AF1)
      AF2=.0549387*R**3-.164298*R*R+.101864*R-.234861
      BF2=-.108911*R**3+.0291149*R*R+.0440961*R-.160312
      EF2=AF2+((AR-4.0)/2.0)*(BF2-AF2)
      GO TO 7
    6 AF1=-.0629823*R**3-.0173375*R*R-.0333427*R-.180026
      BF1=-.110526*R*R+.013893*R-.146355
      EF1=AF1+((AR-6.0)/2.0)*(BF1-AF1)
      AF2=-.108911*R**3+.0291149*R*R+.0440961*R-.160312
      BF2=-.191365*R**3+.147284*R*R-.00039296*R-.119884
      EF2=AF2+((AR-6.0)/2.0)*(BF2-AF2)
    7 EFF=EF2+((TR-.25)/.75)*(EF1-EF2)
      CNDA=2.0*EFF*CL*CLDA
      RETURN
      END


      SUBROUTINE SCYDR
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
     1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU
      COMMON WNSP,ZSP,T12SP,TO5SP,WNP,ZP,T12P,TO5P,WF(21),RRD(10),RRN(10
     1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
     1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,R1,TR,ZV,YI,ETAV,B,SBS,CL
     1AVT,XM,H1,H2,W,WP,RHO,CLA2DW,SAH,BH,SH,TRH,CLA2DH,BA,CA,CH,SR,ALPH
     1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
     1DA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
      R=SR/SV
      IF (R.GE.0.0.AND.R.LE..7) GO TO 2
      WRITF (3,1) R
    1 FORMAT (1X,'VALUE OF SR/SV = ',G11.4,'   USED TO CALCULATE CYDR IS
     1 OUTSIDE PREFERRED RANGE OF 0.0 TO 0.7')
    2 T=21.7949*R**5-46.4744*R**4+36.9347*R**3-14.259*R*R+3.70551*R-.000
     1057815
      CYDR=CLAVT*T*SV/S
      RETURN
      END


      SUBROUTINE SCLDR
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
     1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU
      COMMON WNSP,ZSP,T12SP,TO5SP,WNP,ZP,T12P,TO5P,WF(21),RRD(10),RRN(10
     1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
     1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,R1,TR,ZV,YI,ETAV,B,SBS,CL
     1AVT,XM,H1,H2,W,WP,RHO,CLA2DW,SAH,BH,SH,TRH,CLA2DH,BA,CA,CH,SR,ALPH
     1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
     1DA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
      R=SR/SV
      IF (R.GE.0.0.AND.R.LE..7) GO TO 2
      WRITE (3,1) R
    1 FORMAT (1X,'VALUE OF SR/SV = ',G11.4,'   USED TO CALCULATE CLDR IS
     1 OUTSIDE PREFERRED RANGE OF 0.0 TO 0.7')
    2 T=21.7949*R**5-46.4744*R**4+36.9347*R**3-14.259*R*R+3.70551*R-.000
     1057815
      CLDR=CLAVT*T*SV*ZV/(S*B)
      RETURN
      END


      SUBROUTINE SCNDR
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 MS,KGAIN,KC,KROOT,KKK,KD,KN,IXZ,IXX,IZZ,LP,NR,NP,LR,LB,NB,L
     1IN,NIN,NPHI,NPSI,NS,LV,NV,KY,KI,LF,LT,KB,MU
      COMMON WNSP,ZSP,T12SP,TO5SP,WNP,ZP,T12P,TO5P,WF(21),RRD(10),RRN(10
     1),RID(10),RIN(10),AMPR(21),PHASE(21),ACC,WCYCLE(21),ARDB(21),PHDEG
     1(21),KGAIN,CL,SA,DIH,FUSVOL,ZW,H,S,SV,BV,R1,TR,ZV,YI,ETAV,B,SBS,CL
     1AVT,XM,H1,H2,W,WP,RHO,CLA2DW,SAH,BH,SH,TRH,CLA2DH,BA,CA,CH,SR,ALPH
     1A,CDO,AR,ARH,CYB,CYBT,CLB,CNB,CYP,CLP,CNP,CYR,CLR,CNR,CYDA,CLDA,CN
     1DA,CYDR,CLDR,CNDR,U,LF,LT,I,NUMER
      R=SR/SV
      IF (R.GE.0.0.AND.R.LE..7) GO TO 2
      WRITE (3,1) R
    1 FORMAT (1X,'VALUE OF SR/SV = ',G11.4,'   USED TO CALCULATE CNDR IS
     1 OUTSIDE PREFERRED RANGE OF 0.0 TO 0.7')
    2 T=21.7949*R**5-46.4744*R**4+36.9347*R**3-14.259*R*R+3.70551*R-.000
     1057815
      CNDR=-CLAVT*T*SV*LT*ETAV/(S*B)
      RETURN
      END
```

SAMPLE OUTPUT

```
                    PERTINENT AIRPLANE CHARACTERISTICS

RHO =  0.002050   WING AREA =174.000000   MASS = 82.300000   G*COS(GAMMA) = 32.200000
U   =  219.0000   CHORD     =   4.856266   SPAN =    35.8300  G*SIN(GAMMA) =  0.0
IXX =  948.0000   IXZ       =    0.0       IZZ  =1967.0000   CL           =  0.307000
SA  =  0.0        DIH       =   1.730000   ZW   = -1.835000  FUSVOL       =236.013974
H   =  4.850000   SV        =  18.570000   BV   =  5.750000  R1           =  0.730000
TR  =  0.700000   ZV        =   2.820000   ETAV =  0.850000  SBS          = 74.815850
LF  = 25.000000   LT        =  14.800000   XM   =  7.000000  H1           =  4.800000
H2  =  1.800000   W         =   4.020000   SAH  =  0.0       CLA2DW       =  5.900000
BH  = 11.600000   SH        =  38.710000   TRH  =  0.660000  CLA2DH       =  5.730000
BA  =  8.900000   CA        =   0.750000   SR   =  6.950000  ALPHA        =  0.0
CDO =  0.027900   YI        =   8.340000   HNOSE=  2.700000  WNOSE        =  2.800000
HFCY=  3.500000   WFCY      =   3.600000   LFCY =  3.120000  LMH          =  6.410000
HBCY=  2.900000   WBCY      =   3.100000   LBCY = 12.830000
```

```
                    LATERAL STABILITY DERIVATIVES

CYB  = -0.308006   CLB  = -0.089010   CNB =  0.064553   CYP = -0.037333   CLP  = -0.470774   CNP  = -0.029229
CYR  =  0.210292   CLR  =  0.095876   CNR = -0.099236   CYDA = 0.0        CLDA =  0.177230   CNDA = -0.016725
CYDR =  0.187369   CLDR =  0.014747   CNDR =-0.065786
```

```
                    RESPONSE TO RUDDER DEFLECTION

CYIN =  0.187369   CLIN =  0.014747   CNIN = -0.065786   K = 1   ACC =   0.00010000
```

```
                    DIMENSIONAL STABILITY DERIVATIVES

YV = -0.14618   LB = -28.77650   NB =  10.05814   YP  = -0.31741   LP  = -12.45045   NP  =  -0.37255
YR =  1.78795   LR =   2.53562   NR =  -1.26487   YIN = 19.47414   LIN =   4.76759   NIN = -10.25025
```

POLYNOMIAL COEFFICIENTS FOR THE DENOMINATOR AND NUMERATOR

DS(5) = 1.0000	DS(4) = 13.8615	DS(3) = 28.6321	DS(2) = 141.4946	DS(1) = 1.6019
	NS(4) = 0.0889	NS(3) = 11.3793	NS(2) = 130.5543	NS(1) = -2.9348
	NPHI(4) = 4.7676	NPHI(3) = -21.8223	NPHI(2) = -248.8831	NPHI(1) = 0.0
	NPSI(4) = -10.2503	NPSI(3) = -130.0004	NPSI(2) = -6.4676	NPSI(1) = -36.3188

SOLUTION FOR SIDESLIP VARIATION

DENOMINATOR ROOTS

ROOT(1) = -0.68772 +J -3.29297
ROOT(2) = -0.68772 +J 3.29297
ROOT(3) = -12.47471 +J 0.0
ROOT(4) = -0.01135 +J 0.0

	NATURAL FREQ		DAMPING RATIO	TIME FOR 1/2 DAMPING	SETTLING TIME
	UNDAMPED	DAMPED			
DUTCH ROLL	3.3640	3.2930	0.20444	1.00789	4.35597

NUMERATOR ROOTS

ROOT(1) = -12.76698 +J 0.0
ROOT(2) = -115.22309 +J 0.0
ROOT(3) = 0.02244 +J 0.0

BODE PLOT INFORMATION

FREQUENCY		AMPLITUDE RATIO		PHASE ANGLE	
RAD/SEC	CYCLES/SEC	PURE	DECIBELS	RADIANS	DEGREES
0.01000	0.00159	1.50	3.54993	1.99879	114.52220
0.10000	0.01592	0.94	-0.49735	0.32221	18.46137
1.00000	0.15915	1.01	0.06567	-0.09191	-5.26598
2.52301	0.40155	1.73	4.77733	-0.58051	-33.26095
3.02761	0.48186	2.24	6.98719	-1.06200	-60.84799
3.36402	0.53540	2.26	7.09669	-1.53733	-88.08257
3.70042	0.58894	1.86	5.41050	-1.97265	-113.02440
4.20502	0.66925	1.22	1.70727	-2.36648	-135.58929
10.00000	1.59155	0.12	-18.69283	-2.90905	-166.67604
99.99998	15.91549	0.00	-57.33840	-2.41558	-138.40248
999.99979	159.15494	0.00	-80.96231	-1.68440	-96.50884

TIME RESPONSE PROGRAM

```
      IMPLICIT REAL*4(N)
      DIMENSION NG(10),DG(10),ROOTR(10),ROOTI(10),OUTPUT(200),T(200),FOR
     1CE(6,2)
      DATA FORCE/'IMPU','STEP','RAMP','PULS','RAMP','SINU','LSE',' ',' '
     1,'E','STEP','SUID'/
    1 FORMAT (8I10)
    2 FORMAT (8F10.3)
    3 FORMAT (I10,2F10.3)
      RTIME=2.
      PTIME=2.
      W=1.
    4 READ (1,1, END=23) ING
      IF (ING)  23,23,5
    5 READ (1,2) (NG(I),I=1,ING)
      READ (1,1) IDGM1
      READ (1,2) (ROOTR(I),ROOTI(I),I=1,IDGM1)
      READ (1,3) IF,AMP,GAINO
      AM1=AMP/GAINO
      CALL TIME(IF,AM1,PTIME,RTIME,W,OUTPUT,T,IO,NG,ING,ROOTR,ROOTI,IDGM
     11,ICODE)
      WRITE (3,6)
    6 FORMAT ('1'////,10X,'THE COEFFICIENTS OF THE NUMERATOR'/)
      DO 7 I=1,ING
    7 WRITE (3,8) I,NG(I)
    8 FORMAT (10X,'NG(',I1,')=',F17.7)
      WRITE (3,9)
    9 FORMAT (////10X,'THE ROOTS OF THE DENOMINATOR'/)
      DO 10 I=1,IDGM1
   10 WRITE (3,11) I,ROOTR(I),ROOTI(I)
   11 FORMAT (10X,'ROOT(',I1,')=',F17.7,' + J ',F17.7)
      WRITE (3,12) IF,(FORCE(IF+1,I),I=1,2),AMP
   12 FORMAT (////10X,'THE FORCING FUNCTION INDICATOR (IF) =',I5,//10X,'
     1THIS IMPLIES THAT A ',2A4,' INPUT WAS USED.',//20X,'AMPLITUDE=',F1
     17.7)
      IF (IF.EQ.3) WRITE (3,13) PTIME
   13 FORMAT (/20X,'PTIME=',F17.7)
      IF (IF.EQ.4) WRITE (3,14) RTIME
   14 FORMAT (/20X,'RTIME WAS ADJUSTED TO ',F10.4,' SECONDS.')
      IF (IF.EQ.5) WRITE (3,15) W
   15 FORMAT (/20X,'FREQUENCY=',F17.7)
      IF (ICODE.NE.0) WRITE (3,16) ICODE
   16 FORMAT (//,' ICODE =',I10)
      IF (ICODE.EQ.1) WRITE (3,17)
   17 FORMAT (//10X,'THE COMPLEX PART OF THE OUTPUT VECTOR BECAME
     1 SIGNIFICANT')
      IF (ICODE.EQ.2) WRITE (3,18)
   18 FORMAT (//10X,'MULTIPLE ROOTS ENCOUNTERED')
      IF (ICODE.EQ.3) WRITE (3,19)
   19 FORMAT (//10X,'BAD ENTRY - CHECK POLYNOMIAL ORDERS OF T.F.')
      IF (ICODE.NE.0) GO TO 4
      WRITE (3,20)
   20 FORMAT (//,8X,'TIME',13X,'OUTPUT'/)
      DO 21 I=1,IO
   21 WRITE (3,22) T(I),OUTPUT(I)
   22 FORMAT (2F17.7)
      GO TO 4
   23 CALL EXIT
      END


C***********************************************************************
C
C     PURPOSE:
C
C         THE PURPOSE OF THIS SUBROUTINE IS TO DETERMINE THE TIME
C         RESPONSE OF AN INPUT TO A TRANSFER FUNCTION BY TAKING THE
C         INVERSE LAPLACE TRANSFORM BY THE METHOD OF RESIDUES.
C         INTRODUCTION TO AUTOMATIC CONTROL SYSTEMS - ROBERT N. CLARK
C         (PP 70 - 77).
C
C
C     VARIABLES:
C
C         IF - FORCING FUNCTION INDICATOR
C                 IF=0 IMPLIES AN IMPULSE
C                 IF=1 IMPLIES A STEP
C                 IF=2 IMPLIES A RAMP
C                 IF=3 IMPLIES A PULSE
C                 IF=4 IMPLIES A RAMPSTEP
C                 IF=5 IMPLIES A SINUSOID
C         AMP - AMPLITUDE OF THE FORCING FUNCTION IN RADIANS
C         GAINO - COEFFICIENT OF HIGHEST ORDER TERM OF THE DENOMINATOR
C                 POLYNOMIAL
C         PTIME - PULSE TIME (SPECIFIED FOR IF=3)
C         RTIME - RAMP TIME (SPECIFIED FOR IF=4)
C         W - FREQUENCY OF THE SINUSOIDAL INPUT (SPECIFIED FOR IF=5)
C         OUTPUT - VECTOR OF CALCULATED RESPONCE AMPLITUDE VALUES
C         T - VECTOR OF SEQUENTIAL TIME VALUES DIRECTLY RELATED TO OUTPUT
C         IO - NUMBER OF OUTPUT VALUES (CALCULATED)
C         NG - VECTOR OF NUMERATOR COEFFICIENTS OF THE FORM NG(1) +
C              NG(2)*S + ...
C         ING - DIMENSION OF THE NUMERATOR COEFFICIENTS (ORDER OF
C               NUMERATOR POLYNOMIAL + 1)
C         ROOTR - VECTOR OF REAL PARTS OF THE ROOTS
C         ROOTI - VECTOR OF IMAGINARY PARTS OF THE ROOTS
C         IDGM1 - NUMBER OF ROOTS (ORDER OF THE DENOMINATOR)
C         ICODE - RETURN CODE VARIABLE
C                 ICODE=0 IMPLIES NORMAL EXECUTION
C                 ICODE=1 IMPLIES THAT THE COMPLEX PART OF THE OUTPUT
C                         VECTOR BECAME SIGNIFICANT.
C                 ICODE=2 IMPLIES THAT MULTIPLE ROOTS WERE ENCOUNTERED
C                         IN THE DENOMINATOR.
C                 ICODE=3 IMPLIES THAT THE ORDER OF THE DENOMINATOR WAS
C                         NOT GREATER THAN THE ORDER OF THE NUMERATOR.
C                 ICODE=4 IMPLIES THAT THE FORCING FUNCTION INDICATOR
C                         WAS SPECIFIED INCORRECTLY
C
C
C     SUBROUTINES CALLED:
C
C         CPVAL - COMPLEX EVALUATION OF A POLYNOMIAL
C         TYME - RESPONSE BY THE METHOD OF RESIDUES
C
C
C     REMARKS:
C
C         THIS SUBROUTINE IS DESIGNED TO GENERATE THE TIME RESPONSE OF A
C         GENERAL OUTPUT FUNCTION XO(S) = XI(S)G(S).  IN THIS
C         EVALUATION TWO IMPORTANT ASSUMPTIONS ARE MADE.
C                 1)  THE ORDER OF THE DENOMINATOR OF THE OUTPUT FUNCTION
C                     MUST BE LARGER THAN THE ORDER OF THE NUMERATOR
C                     OF THE OUTPUT FUNCTION.
C                 2)  MULTIPLE ROOTS OF THE DENOMINATOR MAY NOT EXIST.
C
C***********************************************************************
C
      SUBROUTINE TIME(IF,AMP,PTIME,RTIME,W,OUTPUT,T,IO,NG,ING,ROOTR,ROOT
```

```
     1I,IDGM1,ICODE)
      IMPLICIT REAL*4(N)
      DIMENSION P(10),NG(10),K(10),OUT(200),OUTPUT(200),T(200),ROOTR(10)
     1,ROOTI(10),SAVE(200)
      COMMON TDEL,TMAX
      ICODE=4
      IF (IF.GT.5.OR.IF.LT.0) RETURN
      GAINDG=1./AMP
C
C        DETERMINE TMAX
C
      SMALL=1.E6
      DO 1 I=1,IDGM1
      ABSR=ABS(ROOTR(I))
      IF (ABSR.EQ.0.) GO TO 2
      IF (ABSR.LT.SMALL)SMALL=ABSR
    1 CONTINUE
      TMAX=6./SMALL
      GO TO (2,3,4,9,14),IF
      CALL TYME(OUTPUT,T,IO,NG,ING,ROOTR,ROOTI,IDGM1,GAINDG,ICODE)
      RETURN
    2 IDGM1=IDGM1+1
      ROOTR(IDGM1)=0.
      ROOTI(IDGM1)=0.
      CALL TYME(OUTPUT,T,IO,NG,ING,ROOTR,ROOTI,IDGM1,GAINDG,ICODE)
      RETURN
    3 IDGM1=IDGM1+1
      ROOTR(IDGM1)=.001
      ROOTI(IDGM1)=0.
      IDGM1=IDGM1+1
      ROOTR(IDGM1)=-.001
      ROOTI(IDGM1)=0.
      CALL TYME(OUTPUT,T,IO,NG,ING,ROOTR,ROOTI,IDGM1,GAINDG,ICODE)
      RETURN
    4 IDGM1=IDGM1+1
      ROOTR(IDGM1)=0.0
      ROOTI(IDGM1)=0.0
      GAINDG=GAINDG*PTIME
      CALL TYME(SAVE,T,IO,NG,ING,ROOTR,ROOTI,IDGM1,GAINDG,ICODE)
      IF (ICODE.NE.0) GO TO 8
      MTIME=PTIME/TDEL+0.5
      IF (MTIME.EQ.0) GO TO 6
      DO 5 I=1,MTIME
    5 OUTPUT(I)=SAVE(I)
    6 IP1=MTIME+1
      DO 7 I=IP1,IO
    7 OUTPUT(I)=SAVE(I)-SAVE(I-MTIME)
    8 RETURN
    9 I=RTIME/TDEL+.5
      RTIME=I*TDEL
      IDGM1=IDGM1+1
      ROOTR(IDGM1)=.001
      ROOTI(IDGM1)=0.0
      IDGM1=IDGM1+1
      ROOTR(IDGM1)=-.001
      ROOTI(IDGM1)=0.0
      GAINDG=GAINDG*RTIME
      CALL TYME(SAVE,T,IO,NG,ING,ROOTR,ROOTI,IDGM1,GAINDG,ICODE)
      IF (ICODE.NE.0) GO TO 13
      MTIME=RTIME/TDEL+0.5
      IF (MTIME.EQ.0) GO TO 11
      DO 10 I=1,MTIME
   10 OUTPUT(I)=SAVE(I)
   11 IP1=MTIME+1
      DO 12 I=IP1,IO
   12 OUTPUT(I)=SAVE(I)-SAVE(I-MTIME)
   13 RETURN
   14 IDGM1=IDGM1+1
      ROOTR(IDGM1)=0.
      ROOTI(IDGM1)=W
      IDGM1=IDGM1+1
      ROOTR(IDGM1)=0.
      ROOTI(IDGM1)=-W
      CALL TYME(OUTPUT,T,IO,NG,ING,ROOTR,ROOTI,IDGM1,GAINDG,ICODE)
      RETURN
      END
```

```
      SUBROUTINE TYME(OUTPUT,T,IO,NG,ING,ROOTR,ROOTI,IDGM1,GAINDG,ICODE)
      IMPLICIT COMPLEX*8(C,K),REAL*4(N)
      COMPLEX*8 P,S,OUT,OUT1
      DIMENSION P(10),NG(10),K(10),OUT(200),OUTPUT(200),T(200),ROOTR(10)
     1,ROOTI(10),TTEST(16)
      COMMON TDEL,TMAX
      DATA TTEST/.001,.0025,.005,.01,.025,.05,.1,.25,.5,1.,2.5,5.,10.,25
     1.,50.,100./
C
C        CHECK FUR BAD ENTRY
C
      IF (IDGM1.LT.ING) GO TO 10
C
C        CHECK FOR MULTIPLE ROOTS
C
      DO 1 I=1,IDGM1
      RRP1=ROOTR(I)+.001
      RRM1=ROOTR(I)-.001
      RIP1=ROOTI(I)+.001
      RIM1=ROOTI(I)-.001
      DO 1 J=1,IDGM1
      IF (I.EQ.J) GO TO 1
      RRJ=ROOTR(J)
      RIJ=ROOTI(J)
      IF (RRM1.LT.RRJ.AND.RRP1.GT.RRJ.AND.RIM1.LT.RIJ.AND.RIP1.GT.RIJ) G
     1O TO 9
    1 CONTINUE
      ICODE=0
      TDEL=TMAX/200
      DO 2 I=1,16
      IF (TDEL.LT.TTEST(I)) GO TO 3
    2 CONTINUE
      I=16
    3 TDEL=TTEST(I)
      DO 4 I=1,IDGM1
    4 P(I)=CMPLX(ROOTR(I),ROOTI(I))
C
C        DETERMINE THE K'S
C
      DO 6 J=1,IDGM1
      S=P(J)
      CALL CPVAL(KNUM,S,NG,ING)
      KJ=1.
      DO 5 L=1,IDGM1
      IF (L.EQ.J) GO TO 5
      KJ=KJ/(S-P(L))
    5 CONTINUE
      K(J)=KJ*KNUM/GAINDG
    6 CONTINUE
```

```
C
C        DETERMINE THE TIME RESPONSE
C
      IO=0
      T1=-TDEL
    7 IO=IO+1
      OUT1=(0.,0.)
      T1=T1+TDEL
      DO 8 J=1,IDGM1
      IF (ROUTR(J)*T1.LT.-100.) GO TO 8
      OUT1=OUT1+K(J)*CEXP(T1*P(J))
    8 CONTINUE
      OUTPUT(IO)=REAL(OUT1)
      UNREAL=AIMAG(OUT1)
      IF (ABS(UNREAL).GT..001)ICODE=1
      T(IO)=T1
      IF (T1.LT.TMAX) GO TO 7
      RETURN
    9 ICODE=2
      RETURN
   10 ICODE=3
      RETURN
      END


      SUBROUTINE CPVAL(RES,ARG,X,IDIMX)
      COMPLEX*8 RES,ARG
      DIMENSION X(20)
      RES=(0.,0.)
      J=IDIMX
    1 IF (J)  3,3,2
    2 RES=RES*ARG+X(J)
      J=J-1
      GO TO 1
    3 RETURN
      END
```

SAMPLE OUTPUT

```
THE COEFFICIENTS OF THE NUMERATOR

NG(1)=     2620.3518066
NG(2)=      294.7414551
NG(3)=      -54.7695923
NG(4)=       -6.2450991


THE ROOTS OF THE DENOMINATOR

ROOT(1)=       -0.0135900 + J        -0.1801000
ROOT(2)=       -0.0135900 + J         0.1801000
ROOT(3)=       -4.0771599 + J        -4.3687296
ROOT(4)=       -4.0771599 + J         4.3687296
ROOT(5)=        0.0       + J         0.0


THE FORCING FUNCTION INDICATOR (IF) =    1

THIS IMPLIES THAT A STEP     INPUT WAS USED.

          AMPLITUDE=        0.0500000

 TIME               OUTPUT

  0.0                0.0
  2.5000000          9.9579163
  5.0000000         39.2050629
  7.5000000         80.7292175
 10.0000000        125.6774750
 12.5000000        165.1146088
 15.0000000        191.7606049
 17.5000000        201.2922363
 20.0000000        192.9867096
 22.5000000        169.6253052
 25.0000000        136.7240753
 27.5000000        101.2853241
 30.0000000         70.3430634
 32.5000000         49.5980072
 35.0000000         42.4001617
 37.5000000         49.2545013
 40.0000000         67.9096069
 42.5000000         93.9751892
 45.0000000        121.9134674
 47.5000000        146.1875458
 50.0000000        162.3342743
```