COMPUTER PROGRAMS FOR CALCULATING AND PLOTTING THE STABILITY CHARACTERISTICS OF A BALLOON TETHERED IN A WIND

by Robert M. Bennett, Samuel R. Bland, and L. Tracy Redd

Langley Research Center
Hampton, Va. 23665
Descriptions are presented for six related computer programs for calculating the stability characteristics of a balloon tethered in a steady wind. Equilibrium conditions, characteristic roots, and modal ratios are calculated for a range of discrete values of velocity for a fixed tether-line length. Separate programs are used (1) to calculate longitudinal stability characteristics, (2) to calculate lateral stability characteristics, (3) to plot the characteristic roots versus velocity, (4) to plot the characteristic roots in root-locus form, (5) to plot the longitudinal modes of motion, and (6) to plot the lateral modes of motion. The basic equations, program listings, and the input and output data for sample cases are presented, with a brief discussion of the overall operation and limitations. The programs are based on a linearized, stability-derivative type of analysis, including balloon aerodynamics, apparent mass, buoyancy effects, and static forces which result from the tether line.
## CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUMMARY</td>
<td>1</td>
</tr>
<tr>
<td>INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>SYMBOLS</td>
<td>2</td>
</tr>
<tr>
<td>GENERAL DESCRIPTION OF PROGRAMS</td>
<td>5</td>
</tr>
<tr>
<td>LONGITUDINAL STABILITY PROGRAM</td>
<td>6</td>
</tr>
<tr>
<td>Longitudinal Equations of Motion</td>
<td>8</td>
</tr>
<tr>
<td>Definition of Program Variables</td>
<td>14</td>
</tr>
<tr>
<td>Input Required for Longitudinal Stability Program</td>
<td>18</td>
</tr>
<tr>
<td>Limitations and Diagnostic Messages</td>
<td>22</td>
</tr>
<tr>
<td>Listing of Input Data Cards for Sample Case</td>
<td>23</td>
</tr>
<tr>
<td>Listing of Longitudinal Stability Program</td>
<td>24</td>
</tr>
<tr>
<td>Printing of Files Containing Calculated Results for Sample Case</td>
<td>40</td>
</tr>
<tr>
<td>LATERAL STABILITY PROGRAM</td>
<td>53</td>
</tr>
<tr>
<td>Lateral Equations of Motion</td>
<td>55</td>
</tr>
<tr>
<td>Input Required for Lateral Stability Program</td>
<td>57</td>
</tr>
<tr>
<td>Listing of Input Data Cards for Sample Case</td>
<td>61</td>
</tr>
<tr>
<td>Listing of Lateral Stability Program</td>
<td>62</td>
</tr>
<tr>
<td>Printing of Files Containing Calculated Results for Sample Case</td>
<td>70</td>
</tr>
<tr>
<td>PROGRAM FOR PLOTTING FREQUENCIES AND</td>
<td>85</td>
</tr>
<tr>
<td>DECAY RATES VERSUS WIND VELOCITY</td>
<td></td>
</tr>
<tr>
<td>General Description</td>
<td>85</td>
</tr>
<tr>
<td>Definitions of Program Variables</td>
<td>85</td>
</tr>
<tr>
<td>Listing of Program</td>
<td>89</td>
</tr>
<tr>
<td>PROGRAM FOR PLOTTING ROOTS IN ROOT-LOCUS</td>
<td>91</td>
</tr>
<tr>
<td>FORM WITH WIND VELOCITY AS A PARAMETER</td>
<td></td>
</tr>
<tr>
<td>General Description</td>
<td>91</td>
</tr>
<tr>
<td>Listing of Program</td>
<td>93</td>
</tr>
<tr>
<td>PROGRAM FOR PLOTTING LONGITUDINAL MODES OF MOTION</td>
<td>95</td>
</tr>
<tr>
<td>Definition of Program Variables</td>
<td>96</td>
</tr>
<tr>
<td>Description of Input Data</td>
<td>97</td>
</tr>
<tr>
<td>Listing of Input Data Cards for Sample Case</td>
<td>98</td>
</tr>
<tr>
<td>Listing of Program</td>
<td>99</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
</tr>
<tr>
<td>---------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>PROGRAM FOR PLOTTING LATERAL MODES OF MOTION</td>
<td>102</td>
</tr>
<tr>
<td>Listing of Input Data Cards for Sample Case</td>
<td>103</td>
</tr>
<tr>
<td>Listing of Program</td>
<td>104</td>
</tr>
<tr>
<td>APPENDIX — DESCRIPTIONS OF SELECTED SUBROUTINES</td>
<td>107</td>
</tr>
<tr>
<td>Basic Subroutines</td>
<td>107</td>
</tr>
<tr>
<td>Plotting Subroutines</td>
<td>132</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>145</td>
</tr>
</tbody>
</table>
COMPUTER PROGRAMS FOR CALCULATING AND PLOTTING
THE STABILITY CHARACTERISTICS OF A BALLOON
TETHERED IN A WIND

By Robert M. Bennett, Samuel R. Bland,
and L. Tracy Redd
Langley Research Center

SUMMARY

Descriptions are presented for six related computer programs for calculating the stability characteristics of a balloon tethered in a steady wind. Equilibrium conditions, characteristic roots, and modal ratios are calculated for a range of discrete values of velocity for a fixed tether-line length. Separate programs are used (1) to calculate longitudinal stability characteristics, (2) to calculate lateral stability characteristics, (3) to plot the characteristic roots versus velocity, (4) to plot the characteristic roots in root-locus form, (5) to plot the longitudinal modes of motion, and (6) to plot the lateral modes of motion. The basic equations, program listings, and the input and output data for sample cases are presented, with a brief discussion of the overall operation and limitations. The programs are based on a linearized, stability-derivative type of analysis, including balloon aerodynamics, apparent mass, buoyancy effects, and static forces which result from the tether line.

INTRODUCTION

Tethered balloons are used for many purposes such as supporting antennas or carrying measuring instruments aloft. Dynamic instabilities during high-wind conditions often limit the operation or utility of these devices. Although some limited early stability work (refs. 1 to 2, for example) and more recently some work including cable dynamics (refs. 3 to 6, for example) has been done, a systematic study of the factors involved in the stability of tethered balloons is apparently lacking. The Langley Research Center has undertaken a research study to develop methods for stability analysis. Portions of this study are given in references 7 to 10.

The purpose of this report is to list and describe the operation and use of six related computer programs for calculating the stability characteristics of a balloon.

1
tethered in a steady wind and for plotting the results. The analysis on which the pro-
gams are based is given in reference 8. It is essentially a linearized, stability-
derivative type of analysis of dynamic motions in either the longitudinal plane or in the
lateral plane. Buoyancy forces, aerodynamic apparent masses, and the static spring
forces resulting from the tether cable are included, in addition to the usual static and
dynamic aerodynamic terms.

A listing of each program is given along with a listing of input and output data for a
sample case. The overall operation of the programs and some of their limitations are
discussed. Usage descriptions of several of the basic subroutines are given in the
appendix.

SYMBOLS

In addition to the symbol definitions given here, symbols relating to the input data
are more specifically defined in the sections of this report which describe the input data
required by the programs.

A matrix of coefficients of acceleration terms in equations of motion

B matrix of coefficients of velocity or rate terms in equations of motion

C matrix of coefficients of displacement terms in equations of motion

$C_D$ drag coefficient, $\frac{D}{\rho V^2 S/2}$

$C_L$ lift coefficient, $\frac{L}{\rho V^2 S/2}$

$C_I$ rolling-moment coefficient, $\frac{\text{Rolling moment}}{\rho V^2 S c /2}$

$C_m$ pitching-moment coefficient, $\frac{M}{\rho V^2 S c /2}$

$C_n$ yawing-moment coefficient, $\frac{\text{Yawing moment}}{\rho V^2 S c /2}$

$C_Y$ side-force coefficient, $\frac{\text{Side force}}{\rho V^2 S /2}$

$c$ reference length
\( D, L, M \)  

- drag force, lift force, and pitching moment, respectively (see fig. 2)

\( d_c \)  

tether cable diameter (see fig. 2)

\( h_{br} \)  

- component of distance from reference point to center of buoyancy (see fig. 2)

\( h_{cg} \)  

- component of distance from reference point to center of mass (see fig. 2)

\( h_{sr} \)  

- component of distance from reference point to center of mass of balloon structure (see fig. 2)

\( I_y \)  

- pitching moment of inertia about balloon center of mass

\( k \)  

- tether derivatives defined by equations (16) and (18) and equations (22) and (23)

\( l_{br} \)  

- component of distance from reference point to center of buoyancy (see fig. 2)

\( l_{cg} \)  

- component of distance from reference point to center of mass (see fig. 2)

\( l_{sr} \)  

- component of distance from reference point to center of mass of balloon structure (see fig. 2)

\( l_{tr} \)  

- component of distance from reference point to attachment point of tether line (see fig. 2)

\( m_T \)  

- mass of balloon structure and contained gas

\( m_{x,a}, m_{y,a}, m_{z,a} \)  

- aerodynamic apparent mass in body-reference X-axis, Y-axis, and Z-axis directions, respectively

\( n \)  

- number of degrees of freedom or quantity defined by equation (12a)

\( p \)  

- perturbation roll rate

\( \ddot{p} \)  

- quantity defined by equation (12b)

\( q \)  

- perturbation pitch rate

\( q_i \)  

- generalized coordinate

\( \ddot{q} \)  

- quantity defined by equation (12c)
\( r \)  perturbation yaw rate

\( S \)  reference area

\( T_0, T_1 \)  tension of tether cable at lower and upper ends, respectively (see fig. 2)

\( t \)  time

\( t_{tr} \)  component of distance from reference point to attachment point of tether line (see fig. 2)

\( u \)  perturbation velocity along stability \( X \)-axis

\( V \)  velocity

\( W_S \)  structural weight of balloon (see fig. 2)

\( w_c \)  weight per unit length of tether cable

\( x, y, z \)  coordinate displacements in body-fixed stability-axis system with origin at center of mass

\( x_1, z_1 \)  coordinates of balloon center of mass (see fig. 2)

\( \alpha \)  perturbation angle of attack

\( \alpha_t \)  trim angle of attack

\( \beta \)  angle of sideslip

\( \gamma_0, \gamma_1 \)  angles between the horizontal and tether cable, respectively (see fig. 2)

\( \theta, \phi, \psi \)  angular perturbations about the \( X \)-, \( Y \)-, and \( Z \)-axis, respectively

\( \lambda \)  characteristic root

\( \text{Im}(\lambda) \)  frequency

\( \text{Re}(\lambda) \)  decay rate
\( \lambda \) variable defined by equation (12d)

\( \rho \) air density

\( \tau \) variable defined by equation (12e)

Subscript:

\( R \) reference point

Dots over variables indicate differentiation with respect to time.

**GENERAL DESCRIPTION OF PROGRAMS**

A linearized, stability-derivative type of analysis, such as the one considered here, generally results in a system of simultaneous, linear, ordinary, second-order differential equations with constant coefficients. In order to examine the stability of such a system, a solution of the form \( q_i = \tilde{q}_i e^{\lambda t} \) for exponentially varying motion is assumed, where \( q_i \) is a generalized coordinate and \( \tilde{q}_i \) is a complex constant. The resulting stability determinant is of order \( n \times n \), where \( n \) is the number of degrees of freedom, and has elements that are quadratic in \( \lambda \). Thus the determinant has \( 2n \) characteristic roots or eigenvalues. For solution of the stability determinant, the use of a standard eigenvalue computer subroutine requires a transformation to a \( 2n \times 2n \) determinant with \( \lambda \) on the principal diagonal only (see description of subroutine QUADDET in the appendix). The sign of the real part of \( \lambda \), \( \text{Re}(\lambda) \), signifies growth or decay of a mode of motion of the system, with \( \text{Re}(\lambda) > 0 \) indicating a growing motion (instability). Additional insight about the modes of motion of the dynamic system can also be obtained by substituting each characteristic value into the stability determinant and solving for the associated modal ratios or eigenvector elements.

The six programs described herein calculate the characteristic roots and modal ratios for a range of discrete values of velocity for a fixed tether-line length and plot the results. These programs are:

1. Program STABLTY for longitudinal stability calculations
2. Program STBLTY2 for lateral stability calculations
3. Program VPLOT for plotting frequencies \( \text{Re}(\lambda) \) and decay rates \( \text{Im}(\lambda) \) versus wind velocity
4. Program RTLOCUS for plotting roots in root-locus form with wind velocity as a parameter
Program CALBALM for plotting longitudinal modes of motion

Program CALBLM2 for plotting lateral modes of motion

A block diagram illustrating the relationship of these programs and their use is given in figure 1. Although the programs can be operated in any consistent system of units, the constants and labels are generally given in the SI unit system. The pertinent cards are labeled SI UNITS in the comments field (cols. 73 to 80).

The programs are written basically in FORTRAN language for the CDC 6000 series machines with the Langley Research Center version of the SCOPE 3.0 operating system and the RUN compiler. Some of the system library subroutines used by these programs are in the COMPASS language. A FORTRAN simulator for one of the more essential subroutines (MASCNT) is included in the appendix to facilitate use on other systems.

The programs were designed for use through a low-speed terminal system with the program stored on a data cell system at the central computer complex. Efficient usage of the low-speed terminal requires keeping the INPUT and OUTPUT files to minimum length. Thus, the results of the calculations are written onto disk files and routed after execution (fig. 1). In addition, many of the programs have their data for execution included in DATA statements and only the necessary case data or changes to the nominal case are read. Although written in a form suitable for construction of a single program with several levels of overlays, the programs have been left separate and are used sequentially with multiple executions and disk communication between programs. The zero-level overlay is used, however, to reduce the field length for loading. In this form the largest program (lateral version of STABLTY) loads and executes with a field length of 31000 bytes. Typical execution time for STABLTY is about 45 seconds of central processor unit time for one case of 100 velocity increments.

The four plotting programs described are written for a CalComp Electro-Mechanical Plotter, using the Langley Research Center (LRC) plotting system computer software. Relatively high-quality, hard-copy plots are produced by this system. The basic plotting subroutines are not given, but writeups are included in the appendix in order to facilitate program conversion for other systems.

LONGITUDINAL STABILITY PROGRAM

The longitudinal stability program has been adapted from an essentially general program for calculating the eigenvalues of a stability determinant with elements that are quadratic in the eigenvalue. The main program primarily calls working subroutines and handles a portion of the input and output. It contains one main loop for incrementing and varying the wind velocity. The coefficient matrices are generated by calling subroutine
Figure 1.- Block diagram of programs for calculating and plotting stability characteristics of a tethered balloon.
INICOEF which contains the entry point VCOEF. Calculations that are independent of the wind velocity are performed by INICOEF, including the reading of the NAMELIST for input data. The velocity-dependent calculations are performed within the entry point VCOEF. For each value of velocity, the balloon equilibrium conditions such as tether-line angles, height, downstream distance (fig. 2), and aerodynamic trim conditions are calculated. The static aerodynamic coefficients $C_D$, $C_L$, and $C_m$ are calculated by function subprograms that are to be written by the user to describe the aerodynamics of the balloon configuration to be analyzed. These functions are not restricted to linear functions and are written for a reference point (fig. 2). The program transfers the coefficients to the center of mass for use in the calculations, thereby facilitating parameter studies.

The eigenvalues are sorted in the order of ascending frequencies $\text{Im}(\lambda)$ within the main program. Since the present system has real coefficient matrices, complex roots or eigenvalues occur only as complex pairs. For each such pair, the root with positive frequency contains all the needed information. Thus the conjugate root with a negative value of frequency is generally not printed on the files for output.

A symbol cycling technique is used that has been found to be helpful to relate printed and plotted results. A single symbol (plus signs in the plotting programs herein) is used for plotting all points except for every tenth increment of velocity. For every tenth velocity increment, the results are plotted using the standard NASA symbol sequence of circle, square, etc., and the name of the symbol is printed on the printed results. The indexing parameters for the symbol cycling are set up in the main program and are written on tape 7 for later use in plotting, and the symbol name is written on tapes 8 and 11 (blanks on tapes 8 and 11 if plus signs are to be used). This technique was taken from an unpublished flutter program written by Robert N. Desmarais of the Langley Research Center.

Longitudinal Equations of Motion

The equations of longitudinal motion written about the center of mass are (see ref. 9):

$$\ddot{x} + \frac{\rho V S}{2m_x} (2C_D + C_D \alpha) \dot{x} + \frac{k_{xx}}{m_x} x + \frac{\rho V S}{2m_x} (C_D \alpha - C_L) \dot{z} + \frac{k_{xz}}{m_x} z$$

$$+ \left( \frac{k_x \theta}{m_x} + \frac{\rho V^2 S C_D \alpha}{2m_x} \right) \theta = 0$$

(1)
Figure 2.- Sketch of the balloon identifying pertinent dimensional relationships. (All arrows are pointing in the positive direction.)
where the mass and buoyancy terms in equations (1) to (3) are given by

\[ m_x = m_T + m_{x,a} \cos^2 \alpha_t + m_{z,a} \sin^2 \alpha_t \]  
(4)

\[ m_z = m_T + m_{x,a} \sin^2 \alpha_t + m_{z,a} \cos^2 \alpha_t + \frac{\rho S \bar{c}}{4} C_{L\alpha} \]  
(5)

and

\[ M_{s1} = \left[ (l_{br} - l_{cg}) B + (l_{sr} + l_{cg}) W_S \right] \sin \alpha_t \]
\[ + \left[ (h_{cg} - h_{br}) B + (h_{sr} - h_{cg}) W_S \right] \cos \alpha_t \]  
(6)

The coefficients of equations (1) to (6) are evaluated for the steady equilibrium or trim conditions and thus the values of \( \alpha_t, z_1, T_0, \gamma_0 \) etc. (fig. 2) are required. The value of \( \alpha_t \) is calculated by Newton iteration of the following equation (in subroutine TRIM) which is implicit in \( \alpha_t \) and results from combining the lift, drag, and pitching-moment trim equations (ref. 9):

\[ h_{k1} \left( C_L \frac{\rho V^2 S}{2} + B - W_S \right) - M_{s2} - (h_{k2} C_D + \bar{c} C_m) \frac{\rho V^2 S}{2} = 0 \]  
(7)
where \( h_{k1}, h_{k2}, \) and \( M_{s2} \) are functions of \( \alpha_t \) given by

\[
\begin{align*}
    h_{k1} &= (t_{tr} - l_{cg}) \cos \alpha_t + (t_{tr} - h_{cg}) \sin \alpha_t \\
    h_{k2} &= (t_{tr} - h_{cg}) \cos \alpha_t - (l_{tr} - l_{cg}) \sin \alpha_t
\end{align*}
\]  

(8a)  

(8b)

and

\[
M_{s2} = \left[ (l_{br} - l_{cg}) B + (l_{sr} + l_{cg}) W_s \right] \cos \alpha_t
\]

\[
- \left[ (h_{cg} - h_{br}) B + (h_{sr} - h_{cg}) W_s \right] \sin \alpha_t
\]

(9)

and \( C_D, C_L, \) and \( C_m \) are also functions of \( \alpha_t \). It might be noted that the increments \( \Delta C_D, \Delta C_L, \) and \( \Delta C_m \), which are used in the program for parametric studies, are included in the trim calculation as constants but are not considered as functions of \( \alpha_t \).

The aerodynamic coefficients in equations (1) to (3) and equation (7) are referred to the center of mass. The coefficients are given in the program about a reference point (fig. 2) and are transferred to the center of mass by the following relations in subroutines TRIM and DERTRAN:

\[
C_{Lq} = C_{Lq,R} + 2 \frac{x_t}{c} C_{L\alpha} - 2 \frac{z_t}{c} (2C_L + C_{Lu})
\]

(10a)

\[
C_m = C_{m,R} - \frac{x_t}{c} C_L + \frac{z_t}{c} C_D
\]

(10b)

\[
C_{mu} = C_{mu,R} - \frac{x_t}{c} C_{Lu} + \frac{z_t}{c} C_{Du}
\]

(10c)

\[
C_{m\alpha} = C_{m\alpha,R} - \frac{x_t}{c} (C_{L\alpha} + C_D) - \frac{z_t}{c} (C_L - C_{D\alpha})
\]

(10d)

\[
C_{m\alpha} = C_{m\alpha,R} - \frac{x_t}{c} C_{L\alpha}
\]

(10e)

\[
C_{mq} = C_{mq,R} - \frac{x_t}{c} (C_{Lq,R} - 2C_{m\alpha}) - 2 \frac{z_t}{c} (2C_m + C_{mu})
\]

(10f)
Calculation of the equilibrium tether conditions. - The equilibrium tether conditions are required for calculation of the tether derivatives as subsequently discussed. The equilibrium tether conditions are calculated as follows and are based on the analysis of reference 2. The values of $T_1$ and $\gamma_1$ (see fig. 2) are determined from the following equations, which are manipulations of the lift and drag trim equations:

$$T_1 = \left[\left(C_D \frac{\rho V^2 S}{2}\right)^2 + \left(B - W_s + C_L \frac{\rho V^2 S}{2}\right)^2\right]^{1/2}$$

$$\gamma_1 = \cos^{-1}\left(C_D \frac{\rho V^2 S}{2T_1}\right)$$

The velocity $V$ and the cable parameters $l_c$, $d_c$, $C_{D_c}$, and $w_c$ are specified. Let

$$n = C_{D_c} \frac{\rho V^2 d_c}{2}$$

$$\bar{p} = \frac{w_c}{2n}$$

$$\bar{q} = \sqrt{1 + \bar{p}^2}$$

From the analysis of the tether, a variable $\bar{\lambda}$ is also defined as

$$\bar{\lambda}(\gamma_u) = \int_0^{\gamma_u} \left[\frac{\tau(\gamma)}{\sin^2 \gamma + 2\bar{p} \cos \gamma}\right] d\gamma$$

where

$$\tau(\gamma) = \left(\frac{\bar{q} + \bar{p} - \cos \gamma}{\bar{q} - \bar{p} + \cos \gamma}\right) \frac{\bar{p}}{\bar{q}}$$

$\gamma$ is the cable angle, and $\gamma_u$ is the value of $\gamma$ associated with the upper limit of the integral in equation (12d). Since $\gamma_1$ is known from equation (11b), $\bar{\lambda}_1 = \bar{\lambda}(\gamma_1)$ is calculated (subroutine TETHER) from equation (12d), using numerical integration (subroutine ROMBERG), and $\tau_1 = \tau(\gamma_1)$ is calculated from equation (12e). The related value of $\bar{\lambda}_0$ is given by (ref. 2):
\[ \lambda_0 = \lambda_1 - \frac{n \tau_1 t_c}{T_1} \]  
(13)

where \( \lambda_0 = \lambda(\gamma_0) \). Thus, the left-hand side of equation (12d) is known from equation (13) and can be evaluated for \( \gamma_u = \gamma_0 \). This is done in subroutine TETHER using Newton iteration (subroutine NEWINT) and numerical integration (subroutine ROMBERG). The values of \( T_0 \) and \( z_1 \) are given by (ref. 2):

\[ T_0 = \frac{T_1 \tau_0}{\tau_1} \]  
(14a)

\[ z_1 = \frac{T_1 - T_0}{w_c} \]  
(14b)

with \( \tau_0 = \tau(\gamma_0) \). The value of \( x_1 \) is given by (ref. 2):

\[ x_1 = \frac{T_1}{n \tau_1} \int_{\gamma_0}^{\gamma_1} \left[ \frac{\tau(y)}{\sin^2 y + 2 \phi \cos y} \right] \cos y \, dy \]  
(15)

which is also integrated numerically using subroutine ROMBERG (in subroutine TETHER).

**Calculation of the tether derivatives.** - The tether derivatives or spring constants \( k_{xx} \), \( k_{xz} \), \( k_{zx} \), and \( k_{zz} \) required for equations (1) to (3) are also calculated using the analysis of reference 2. These derivatives are expressed in terms of the equilibrium tether conditions as follows:

\[ k_{xx} = \frac{1}{\delta} \left[ T_1 \cos \gamma_1 (\sin \gamma_1 - \sin \gamma_0) + n \left( z_1 - t_1 \sin \gamma_0 \right) \sin^3 \gamma_1 \right] \]  
(16a)

\[ k_{xz} = \frac{1}{\delta} \left[ T_1 \cos \gamma_1 (\cos \gamma_0 - \cos \gamma_1) + n \left( t_1 \cos \gamma_0 - x_1 \right) \sin^3 \gamma_1 \right] \]  
(16b)

\[ k_{zx} = \frac{1}{\delta} \left[ T_1 \sin \gamma_1 (\sin \gamma_1 - \sin \gamma_0) \right. \]  

\[ - \left. \left( w_c + n \sin^2 \gamma_1 \cos \gamma_1 \right) \left( z_1 - t_1 \sin \gamma_0 \right) \right] \]  
(16c)

\[ k_{zz} = \frac{1}{\delta} \left[ T_1 \sin \gamma_1 (\cos \gamma_0 - \cos \gamma_1) \right. \]  

\[ - \left. \left( w_c + n \sin^2 \gamma_1 \cos \gamma_1 \right) \left( t_1 \cos \gamma_0 - x_1 \right) \right] \]  
(16d)
where

$$\delta = x_1 (\sin \gamma_1 - \sin \gamma_0) + z_1 (\cos \gamma_0 - \cos \gamma_1) - l_1 \sin (\gamma_1 - \gamma_0)$$  \hspace{1cm} (17)$$

The tether derivatives related to pitch angle $\theta$ and referred to the center of mass are

$$k_{\theta x} = h_{k_2}k_{xx} - h_{k_1}k_{zx} \hspace{1cm} (18a)$$

$$k_{\theta z} = h_{k_2}k_{xz} - h_{k_1}k_{zz} \hspace{1cm} (18b)$$

$$k_{x\theta} = h_{k_2}k_{xx} - h_{k_1}k_{xz} \hspace{1cm} (18c)$$

$$k_{z\theta} = h_{k_2}k_{zx} - h_{k_1}k_{zz} \hspace{1cm} (18d)$$

$$k_{\theta \theta} = k_{\theta \theta_D} + k_{\theta \theta_{T_1}} \hspace{1cm} (18e)$$

$$k_{\theta \theta_D} = h_{k_2}^2k_{xx} - h_{k_2}h_{k_1}(k_{xz} + k_{zx}) + h_{k_1}^2k_{zz} \hspace{1cm} (18f)$$

$$k_{\theta \theta_{T_1}} = h_{k_2}(T_1 \sin \gamma_1) + h_{k_1}(T_1 \cos \gamma_1) \hspace{1cm} (18g)$$

where $h_{k_1}$ and $h_{k_2}$ are defined by equations (8).

**Definition of Program Variables**

Some of the principal FORTRAN variable names are given and defined in the following sections. Where there are corresponding mathematical symbols, these symbols are also listed.

**Variables in the main program.** - The variables used in the main program are given as follows:
<table>
<thead>
<tr>
<th>FORTRAN variable name</th>
<th>Mathematical symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>A, B, C</td>
<td>$n \times n$</td>
<td>coefficient matrices of the equations of motion, i.e., mass, rate, and displacement matrices, respectively; rows 1, 2, and 3 contain the coefficients of the x-, z-, and $\theta$-equations, respectively</td>
</tr>
<tr>
<td>CNOA, CNOEI</td>
<td></td>
<td>Turing's condition number of the matrices $A$ and $EIDET$, respectively, defined in terms of the norm as, for example: $CNOA = |A| \times |A^{-1}|/n$, $n = \text{Order of } A$</td>
</tr>
<tr>
<td>CROT, CRTSQ</td>
<td></td>
<td>a complex eigenvalue and its square, respectively</td>
</tr>
<tr>
<td>DELV</td>
<td>$\Delta V$</td>
<td>increment in velocity</td>
</tr>
<tr>
<td>EICOEF</td>
<td></td>
<td>complex coefficient array for eigenvector calculations; here, $2 \times 3$ and normalized by $2^\theta$ in degrees</td>
</tr>
<tr>
<td>EIDET</td>
<td></td>
<td>stability or eigenvalue determinant in expanded form, $2n \times 2n$</td>
</tr>
<tr>
<td>ID</td>
<td></td>
<td>ten-word alphanumeric array containing case identification card and date and time for processing of case</td>
</tr>
<tr>
<td>IK2</td>
<td></td>
<td>index for cycling symbols</td>
</tr>
<tr>
<td>NEGR</td>
<td></td>
<td>number of complex eigenvalues with negative frequencies plus one</td>
</tr>
<tr>
<td>NMP</td>
<td></td>
<td>number of modes processed; here set to 3</td>
</tr>
<tr>
<td>NTWO</td>
<td></td>
<td>$2 \times \text{NMP}$; order of linear eigenvalue determinant</td>
</tr>
<tr>
<td>NVEL</td>
<td></td>
<td>number of velocity increments</td>
</tr>
<tr>
<td>ROOTI</td>
<td>$\text{Im}(\lambda)$</td>
<td>array containing imaginary portion of eigenvalues (modal frequencies) for a given velocity</td>
</tr>
<tr>
<td>FORTRAN variable name</td>
<td>Mathematical symbol</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------</td>
<td>------------</td>
</tr>
<tr>
<td>ROOTR</td>
<td>Re(\lambda)</td>
<td>array containing real portion of eigenvalues for a given velocity in the same sequence as ROOTI</td>
</tr>
<tr>
<td>SYMBOL</td>
<td></td>
<td>eleven-word Hollerith array containing names of plotting symbols for symbol cycling</td>
</tr>
<tr>
<td>VEL</td>
<td>V</td>
<td>velocity</td>
</tr>
<tr>
<td>VMIN</td>
<td></td>
<td>minimum velocity</td>
</tr>
</tbody>
</table>

Variables in subroutine INICOEF.- Many of the variables used in INICOEF are listed in the NAMELIST and are defined in the section entitled "Input Required for Longitudinal Stability Program." Also, some have the same usage as in the main program. Other principal variables are given as follows:

<table>
<thead>
<tr>
<th>FORTRAN variable name</th>
<th>Mathematical symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALPHA</td>
<td>\alpha_t</td>
<td>trim angle of attack in radians</td>
</tr>
<tr>
<td>ALPHAD</td>
<td>\alpha_t</td>
<td>trim angle of attack in degrees</td>
</tr>
<tr>
<td>CD</td>
<td>C_D</td>
<td>drag coefficient at trim</td>
</tr>
<tr>
<td>CDA</td>
<td>C_D\alpha</td>
<td>\frac{\partial C_D}{\partial \alpha}</td>
</tr>
<tr>
<td>CDRAG</td>
<td>n</td>
<td>cable drag per unit length</td>
</tr>
<tr>
<td>CL</td>
<td>C_L</td>
<td>lift coefficient for trim</td>
</tr>
<tr>
<td>CLA</td>
<td>C_L\alpha</td>
<td>lift-curve slope</td>
</tr>
<tr>
<td>CLQ</td>
<td>C_Lq</td>
<td>lift pitching-rate derivative about center of mass, \frac{\partial C_L}{\partial \frac{\alpha}{2V}}</td>
</tr>
<tr>
<td>CM</td>
<td>C_m</td>
<td>pitching-moment coefficient about center of mass at trim</td>
</tr>
<tr>
<td>FORTRAN variable name</td>
<td>Mathematical symbol</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------</td>
<td>------------</td>
</tr>
<tr>
<td>CMA</td>
<td>$C_{m\alpha}$</td>
<td>pitching-moment derivative about center of mass, $\partial C_m/\partial \alpha$</td>
</tr>
<tr>
<td>CMAD</td>
<td>$C_{m\dot{\alpha}}$</td>
<td>moment $\alpha$-rate stability derivative about center of mass, $\partial C_m/\partial \dot{\alpha}$</td>
</tr>
<tr>
<td>CMQ</td>
<td>$C_{mq}$</td>
<td>moment pitching-rate stability derivative about center of mass, $\partial C_m/\partial q\dot{c}/2V$</td>
</tr>
<tr>
<td>GAMO</td>
<td>$\gamma_0$</td>
<td>cable angle at ground measured from horizontal</td>
</tr>
<tr>
<td>GAM1</td>
<td>$\gamma_1$</td>
<td>cable angle at tether point on bridle measured from horizontal</td>
</tr>
<tr>
<td>Q</td>
<td>$Q$</td>
<td>dynamic pressure, $\rho V^2/2$</td>
</tr>
<tr>
<td>SKTT</td>
<td>$k_{\theta\theta}$</td>
<td>total tether pitch spring in body-axis system for pitch about center of mass</td>
</tr>
<tr>
<td>SKTTD</td>
<td>$k_{\theta\theta_D}$</td>
<td>portion of SKTT due to displacement of tether point for pitch about center of mass</td>
</tr>
<tr>
<td>SKTTT</td>
<td>$k_{\theta\theta_T}$</td>
<td>portion of SKTT due to rotation of balloon relative to steady tension vector at tether point</td>
</tr>
<tr>
<td>SKTZ</td>
<td>$k_{\theta z}$</td>
<td>tether pitching moment due to $z$-displacement of balloon</td>
</tr>
<tr>
<td>SKTX</td>
<td>$k_{\theta x}$</td>
<td>tether pitching moment due to $x$-displacement of balloon</td>
</tr>
<tr>
<td>SKXT</td>
<td>$k_{x \theta}$</td>
<td>tether $x$-force due to pitching displacement</td>
</tr>
<tr>
<td>SKXX</td>
<td>$k_{xx}$</td>
<td>tether $x$-spring constant at tether point</td>
</tr>
<tr>
<td>SKXXZ</td>
<td>$k_{xz}$</td>
<td>tether $x$-force due to $z$-displacement</td>
</tr>
<tr>
<td>FORTRAN variable name</td>
<td>Mathematical symbol</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------</td>
<td>------------</td>
</tr>
<tr>
<td>SKZT</td>
<td>$k_{z\theta}$</td>
<td>tether $z$-force due to pitching displacement</td>
</tr>
<tr>
<td>SKZX</td>
<td>$k_{zX}$</td>
<td>tether $z$-force due to $x$-displacement</td>
</tr>
<tr>
<td>SKZZ</td>
<td>$k_{zz}$</td>
<td>tether $z$-spring constant at tether point</td>
</tr>
<tr>
<td>TO</td>
<td>$T_0$</td>
<td>tether cable tension at ground (see fig. 2)</td>
</tr>
<tr>
<td>T1</td>
<td>$T_1$</td>
<td>tether cable tension at bridle (see fig. 2)</td>
</tr>
<tr>
<td>XOC</td>
<td>$x_t/\bar{c}$</td>
<td>$x$-distance in stability-axis system from reference point to center of mass</td>
</tr>
<tr>
<td>X1</td>
<td>$x_1$</td>
<td>balloon horizontal displacement, positive in direction of wind</td>
</tr>
<tr>
<td>ZOC</td>
<td>$z_t/\bar{c}$</td>
<td>$z$-distance in stability-axis system from reference point to center of mass</td>
</tr>
<tr>
<td>Z1</td>
<td>$z_1$</td>
<td>balloon altitude</td>
</tr>
<tr>
<td>UNCRT</td>
<td></td>
<td>complex eigenvalue obtained by factoring diagonal quadratic element of stability determinant</td>
</tr>
</tbody>
</table>

It may also be noted that the tether subroutines called by VCOEF are written with FORTRAN variable names closely paralleling the mathematical notation of reference 2.

Input Required for Longitudinal Stability Program

The user-written function subroutines FCD, FCL, and FCMR describe the longitudinal static aerodynamic coefficients about the reference point for the configuration. In the present usage, the curve fits (ref. 8) to the measured coefficients as functions of angle of attack are used. The static coefficients $C_D$, $C_L$, and $C_{m,R}$ are associated with the function variable names. Angle of attack is passed as a formal parameter and the derivatives $C_{D\alpha'}$, $C_{L\alpha'}$, and $C_{m\alpha',R}$ are returned as formal parameters of the functions FCD, FCL, and FCMR, respectively. These functions must be replaced with functions appropriate for the configuration to be analyzed and are thus considered part of the input data.
For each case, one card of 80 characters of case identification is read in an 8A10 format, and a NAMELIST called LONGDTA is read. The FORTRAN variable names, their equivalent mathematical symbols, and their definitions are given as follows in the order the variables are listed in the NAMELIST, which is also the order for printing. All variables are preset in the program with DATA statements to the values for the reference configuration of the LRC balloon, and only changes need to be read with the NAMELIST. Thus, the program can be executed using no changes in the parameters in the NAMELIST.

<table>
<thead>
<tr>
<th>FORTRAN variable name</th>
<th>Mathematical symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDINS</td>
<td>$C_{D_{\text{ins}}}$</td>
<td>constant increment of $C_D$ (allows for $C_D$ of instrument package of balloon)</td>
</tr>
<tr>
<td>CLAD</td>
<td>$C_{L_\alpha}$</td>
<td>lift $\alpha$-rate stability derivative, $\partial C_L/\partial \alpha_c V$</td>
</tr>
<tr>
<td>CLQR</td>
<td>$C_{Lq,R}$</td>
<td>lift pitching-rate stability derivative about reference point, $\partial C_L/\partial q_c V$</td>
</tr>
<tr>
<td>CMADR</td>
<td>$C_{m\alpha,R}$</td>
<td>moment $\alpha$-rate stability derivative about reference point, $\partial C_{m,R}/\partial \alpha_c V$</td>
</tr>
<tr>
<td>CMQR</td>
<td>$C_{mq,R}$</td>
<td>moment pitching-rate stability derivative about reference point, $\partial C_{m,R}/\partial q_c V$</td>
</tr>
<tr>
<td>DELCD</td>
<td>$\Delta C_D$</td>
<td>constant increments in coefficients about center of mass which are used for parametric studies</td>
</tr>
<tr>
<td>DELCDA</td>
<td>$\Delta C_{D_\alpha}$</td>
<td></td>
</tr>
<tr>
<td>DELCL</td>
<td>$\Delta C_L$</td>
<td></td>
</tr>
<tr>
<td>DELCLA</td>
<td>$\Delta C_{L_\alpha}$</td>
<td></td>
</tr>
<tr>
<td>DELCM</td>
<td>$\Delta C_m$</td>
<td></td>
</tr>
<tr>
<td>DELCMA</td>
<td>$\Delta C_{m_\alpha}$</td>
<td></td>
</tr>
<tr>
<td>FORTRAN variable name</td>
<td>Mathematical symbol</td>
<td>Definition</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------</td>
<td>------------</td>
</tr>
<tr>
<td>CDU</td>
<td>$C_{Du}$</td>
<td>rate of change of drag coefficient with velocity, $\frac{\partial C_D}{\partial \frac{u}{V}}$</td>
</tr>
<tr>
<td>CLU</td>
<td>$C_{Lu}$</td>
<td>rate of change of lift coefficient with velocity, $\frac{\partial C_L}{\partial \frac{u}{V}}$</td>
</tr>
<tr>
<td>CMUR</td>
<td>$C_{mu,R}$</td>
<td>rate of change of moment about reference point with velocity, $\frac{\partial C_{m,R}}{\partial \frac{u}{V}}$</td>
</tr>
<tr>
<td>S</td>
<td>$S$</td>
<td>reference area, $(\text{Volume of balloon})^{2/3}$</td>
</tr>
<tr>
<td>CBAR</td>
<td>$\bar{c}$</td>
<td>reference length, balloon body length used here</td>
</tr>
<tr>
<td>YYOI</td>
<td>$I_y$</td>
<td>pitching inertia about balloon center of mass (including aerodynamic apparent inertia)</td>
</tr>
<tr>
<td>TMASS</td>
<td>$m_T$</td>
<td>mass of balloon structure and contained gas</td>
</tr>
<tr>
<td>AXMASS</td>
<td>$m_{x,a}$</td>
<td>aerodynamic apparent mass in body-reference X-axis direction, $\alpha_t = 0$</td>
</tr>
<tr>
<td>AZMASS</td>
<td>$m_{z,a}$</td>
<td>aerodynamic apparent mass in body-reference Z-axis direction, $\alpha_t = 0$</td>
</tr>
<tr>
<td>WTS</td>
<td>$W_S$</td>
<td>structural weight of balloon</td>
</tr>
<tr>
<td>BUOY</td>
<td>$B$</td>
<td>net buoyancy force</td>
</tr>
<tr>
<td>BHR</td>
<td>$h_{br}$</td>
<td>component of distance from reference point to center of buoyancy, positive for center of buoyancy below reference point (see fig. 2)</td>
</tr>
<tr>
<td>BLR</td>
<td>$l_{br}$</td>
<td>component of distance from reference point to center of buoyancy, positive for center of buoyancy forward of reference point (see fig. 2)</td>
</tr>
<tr>
<td>FORTRAN variable name</td>
<td>Mathematical symbol</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------</td>
<td>------------</td>
</tr>
<tr>
<td>SHR</td>
<td>$h_{sr}$</td>
<td>component of distance from reference point to center of mass of balloon structure, positive for center of mass below reference point (see fig. 2)</td>
</tr>
<tr>
<td>SLR</td>
<td>$l_{sr}$</td>
<td>component of distance from reference point to center of mass of balloon structure, positive for center of mass aft of reference point (see fig. 2)</td>
</tr>
<tr>
<td>CGH</td>
<td>$h_{cg}$</td>
<td>component of distance from reference point to center of mass, positive for center of mass below reference point (see fig. 2)</td>
</tr>
<tr>
<td>CGL</td>
<td>$l_{cg}$</td>
<td>component of distance from reference point to center of mass, positive for center of mass forward of reference point (see fig. 2)</td>
</tr>
<tr>
<td>TLR</td>
<td>$l_{tr}$</td>
<td>component of distance from reference point to attachment point of tether line, positive for attachment point forward of reference point (see fig. 2)</td>
</tr>
<tr>
<td>TTR</td>
<td>$t_{tr}$</td>
<td>component of distance from reference point to attachment point of tether line, positive for attachment point below reference point (see fig. 2)</td>
</tr>
<tr>
<td>CLC</td>
<td>$l_c$</td>
<td>length of tether cable</td>
</tr>
<tr>
<td>CDIAM</td>
<td>$d_c$</td>
<td>diameter of tether cable</td>
</tr>
<tr>
<td>CDC</td>
<td>$C_{Dc}$</td>
<td>drag coefficient of tether cable based on diameter, i.e., drag of cable per unit length is $C_{Dc}d_c\rho V^2/2$</td>
</tr>
<tr>
<td>WC</td>
<td>$w_c$</td>
<td>weight per unit length of tether cable</td>
</tr>
<tr>
<td>RHO</td>
<td>$\rho$</td>
<td>ambient air density</td>
</tr>
<tr>
<td>VMIN</td>
<td>$V_{min}$</td>
<td>minimum wind velocity</td>
</tr>
</tbody>
</table>
FORTRAN variable name | Mathematical symbol | Definition
--- | --- | ---
DELV | $\Delta V$ | wind-velocity increment
NVEL | | number of velocity calculations

Limitations and Diagnostic Messages

The following comments are given to indicate some of the factors that are not treated in the program and to indicate some potential troublesome factors in program operation:

1. The balloon must lift the tether cable off the ground. No diagnostic messages are given, but the listing of tether conditions will indicate zero cable angle $\Gamma_{MO}$ as the constraint of $0 \leq \Gamma_{MO} \leq \pi/2$ is applied in the program.

2. The balloon must be able to trim. If the trim angle has not converged to a tolerance of $\text{ERR} \times 10^{-6}$ in $\text{ITCMAX} \times 100$ Newton iterations of the trim equation, the message
   
   ITERATION FOR TRIM DID NOT CONVERGE IN _______ ITERATIONS,
   
   $\text{ALPHA} = _____, \text{TLPHA} = _____.$

   is written on tape 11. The value of TLPHA is used for subsequent calculations.

3. The $z$-component of the cable tension acting on the bridle must be directed down for trim. If this condition is not satisfied, $\Gamma_{M1}$ is erroneous. However, the balloon would normally not be lifting the tether cable off the ground when this limitation would apply.

4. The bridle is treated as rigid and no consideration is given to the possibility that the bridle lines may go slack.

5. Cable drag and weight cannot be zero or negative, as these conditions lead to overflows or to a negative number to a real power, both fatal errors. This condition also indicates that the zero wind velocity limit cannot be reached.

6. The balloon drag must be positive. If there is a tendency for trim angle to diverge with velocity, care must be exercised in fitting $C_D$ versus $\alpha$ such that $C_D$ is always positive.

7. The minimum velocity that can be treated is about 0.5 m/sec. Loss of significant figures in some of the tether springs and tether conditions may occur at very low velocities. For example, the vertical spring $k_{zz} \rightarrow \infty$ as $V \rightarrow 0$ such that the eigenvalue problem becomes poorly conditioned.
(8) The conditioning of the eigenvalue matrix EIDET is checked. If the Turing's condition number of EIDET exceeds $10^6$, indicating an estimated loss of 6 or more of 14 significant figures, the diagnostic message

CONDITION NUMBER OF EIGENVALUE MATRIX = ________.

is written on tape 11 and calculations proceed.

(9) The conditioning of the mass matrix A is checked. The mass matrix here is normally well conditioned. If Turing's condition number of A exceeds $10^4$, the diagnostic message

CONDITION NUMBER OF A-MATRIX = ________.

is written on tape 11 and calculations proceed.

(10) The density $\rho$ is input and is considered as a constant both for the cable and for the balloon; thus, the altitude range may be restricted.

(11) The program transfers the stability derivatives from the reference point to the center of mass. However, the center of mass must be computed for input consistent with the structural weight center of mass, the included gas of the balloon, and the aerodynamic apparent masses. The aerodynamic apparent inertia must also be transferred external to this program for input for shifts in center of mass.

(12) It may be noted that the computer running time is closely related to the error tolerances EPS in the iteration and integration procedures used in the tether routines. Here, EPS is generally set to $10^{-8}$.

Listing of Input Data Cards for Sample Case

**LONGITUDINAL STABILITY OF TETHERED BALLOON - LRC BALLOON-REFERENCE CONFIGURATION**

$LONGDATA VMIN=1., NVEL=51, DELV=1.$
Listing of Longitudinal Stability Program

OVERLAYS(STATLTY,0,0)
PROGRAM STABILITY[INPUT=1,OUTPUT=1,TAPE5=INPUT,TAPE7,TAPE8=1,
+ TAPE11=1,TAPE30=1,TAPE31=1,TAPE32=1,TAPE33=1]
C******************************************************************************
C* PROGRAM A2864.1 - LONGITUDINAL STABILITY OF TETHERED BALLOON
C*
C* PROGRAM READS IDENTIFICATION CARD AND NAMELIST FROM INPUT FILE, AND
C* WRITES ONLY THE ID ARRAY FOR EACH CASE ON THE OUTPUT FILE
C* ALL FILES ARE BCD AND ARE SET TO MINIMUM BUFFER SIZE, EXCEPT TAPE7
C* WHICH IS BINARY AND USES STANDARD BUFFER SIZE
C* FILE ASSIGNMENTS ARE - TAPE7=PLOTTING PROGRAM INPUT, TAPE8=EIGEN-
C* VECTORS, TAPE11=EIGENVALUES, TAPE30=AERODYNAMIC COEFFICIENTS,
C* TAPE31=TETHER SPRINGS, TAPE32=TETHER CONDITIONS, AND TAPE33=
C* UNCOUPLED ROOTS
C*
C******************************************************************************
COMPLEX IROW/IRWO(300)/ICGl/ICGL(300)
DIMENSION A(3,3),B(3,3),C(3,3),SYMBOL(11),ID(10)
DIMENSION EIDET(6,6),SAVE(6,7),ROORI(6),ROOTG(6),INDEX(6)
+ ,IRON(6),P(6),PIPV(3),INDEX(3,2)
COMPLEX EICOEF(3,31),CROT,CATSQ,COET
DATA SYMBOL/110HCIRCLE SQUARE DIAMOND TRIANGLE RT TRNG1
+ QUADRANT DUG HOUSE FAN LNG DMND HOUSE
DATA RADEG,DELV,NVEL,VMIN/.01745329943269,.5,10/.5/.5/ SI UNITS
108 FORMAT(1HI///1X10A10///)
107 FORMAT(12X8G13.5)
106 FORMAT(1X* VELUCITY=**G13.5,2XA10)
105 FORMAT(150X EIGENVECTORS*/14X COMPLEX ROOT-REAL,IMAG*4X
+ *XTHETA,M/DEG-REAL,IMAG*4X
+ *THETA,DEG-REAL,IMAG*4X)
104 FORMAT(18X,A10,6G16.6)
103 FORMAT(/2X7G16.6)
102 FORMAT(/2X CONDITION NUMBER OF EIGENVALUE MATRIX=*E10.2/)
101 FORMAT(1X CONDITION NUMBER OF A-MATRIX=*E10.2/)
100 FORMAT(1X VELOCITY,REAL/RECIENT [1]I],I=1,NPOS)/* SYMBOL[1]IMAG[REGT[1]
+ ]I],I=1,NPOS)/*)
   A1= A(1X10A10) $ A10= 6*(6A10)
C INITIALIZATION SECTION - READ IDENTIFICATION CARD, CALL DAYTIM FOR
C DATE AND TIME, AND WRITE ID ARRAY ON BCD TAPES 8,11,30,31,32,33,
C AND BINARY TAPE 7 WITH RECOUT. DO NON-VELOCITY-DEPENDENT
C CALCULATIONS WITH A CALL TO INICOEF
C SEE SUBROUTINE WRITEUP FOR DESCRIPTION OF RECOUT
C NMP=3 $ NTWC=NMP+NMP $ NPL1=NTWO+1
REWIND 30 $ REWIND 31 $ REWIND 32 $ REWIND 33
REWIND 7 $ REWIND 8 $ REWIND 11
1 READ A10(ID1).I=1.B $ IF(EOF,5,1999,2)
2 CALL DAYTIM(ID1) $ PRINT A11.ID $ WRITE(11,10810)
WRITE(30,10810)ID $ WRITE(31,10810)ID $ WRITE(32,10810)ID
WRITE(8,10510) $ WRITE(33,10810)ID
CALL RECOUT(17,2,0,1,NV,10,1,1)
CALL INICOEF(A,B,C,NMP,VEL,VMIN,DELV,NVEL) $ WRITE(11,100)
CALL RECOUT(17,1,0,NVEL)
C 90-LOOP IS VELOCITY VARIATION LOOP
C DO 90 IV=1,NVEL $ VEL=VMIN+(IV-1.)*DELV
C
SET UP COEFFICIENT MATRICES FOR QUADRATIC STABILITY DETERMINANT
WITH CALL TO ENTRY VCOEF OF SUBROUTINE I~ICOEF
CALL VCOEF(A,B,C,NMP,VEL,VMIN,DELV,NVEL)
EXPAND QUADRATIC N X N STABILITY DETERMINANT INTO 2N X 2N STANDARD
EIGENVALUE FORM AND CHECK CONDITIONING OF MASS MATRIX A
CALL QUADEA(A,B,C, 3, 6,NMP,10,EIDET,CNOA)
IF(CNOA.GT.1.E+4)PRINT(11,101)CNOA
EIGENVALUES FOR 2N SYSTEM AND CHECK CONDITIONING OF 2N X 2N MATRIX
WITH CALL TO MATRIX INVERSE AND TURING CONDITION NUMBER
CALL REIG(EIDET,NTWO,NTWO,O,ROUTI,ROOTI,EIVEC, 6,INDEX,IRUN,P,
NPL,NPL1,SAVE) CALL MATRIX(10,NTWO,NTWO,O,EIDET, 6,DETEI,KB,CNOEI)
IF(CNOEI.GT.1.E+6)WRITE(ll,102)CNOEI
RECT SORTING - SORT COMPLEX ROOTS IN ORDER OF INCREASING MAGNITUDE OF
FREQUENCY AND DETERMINE THE NUMBER OF COMPLEX ROOTS WITH POSITIVE
VALUE OF FREQUENCY (IMAGINARY PART)
NEGR=1 $ DO 50 NRT=1,NTWO $ NI=NTWO-NRT $ DO 48 J=1,NI
IF(RCOT(J)-ROOTI(J+1))46,46,46
46 TRI=RCOT(J) $ TRK=ROUTR(J) $ ROUTI(J)=ROOTI(J+1)
ROOTR(J)=ROUTR(J+1) $ ROOTI(J+1)=TRI $ ROOTR(J+1)=TRR
43 CONTINUE
50 CONTINUE $ DO 52 NR=1,NTWO
IF(ROUTI(NR).LT.-1.E-12)NEGR=NEGR+1
52 CONTINUE
WRITE ROOTS ON TAPE 11
IK1=IV/10 $ IK2=11 $ IF(IV.EQ.10*IK1) IK2=1+MOD(IK1-1,10)
WRITE(11,103)VEL,(ROOTI(N),N=NEGR,NTWO)
WRITE(11,104)SYMBOL[IK2], (ROOTI(N),N=NEGR,NTWO)
WRITE RESULTS ON BINARY TAPE 7 FOR INPUT TO PLOTTING PROGRAMS
CALL RECOUT(7,1,0,VEL,IK2,NEGR,NTWO)
CALL RECOUT(7,2,0,ROOTI,NEGR,NTWO)
CALL RECOUT(7,2,0,ROOTR,NEGR,NTWO)
SETUP COEFFICIENT MATRICES FOR EIGENVECTOR (MODAL RATIOS) BY
DIVIDING BY THETA AND CALLING CXINV - RESULTS ON TAPE 8
WRITE(8,106)VEL,SYMBOL(IIK2)
DO 70 NE=NEGR,NTWO $ CRN=CMPLX(ROOTR(NE),ROOTI(NE))
CRSC=CRN*CRN $ DO 60 IC=1,2 $ DO 60 IR=1,3
60 ECCFIC,IC)=-A(IC,IR)*CRS+3(I,IR)*GR0T(C1C,IR)
DO 64 I=1,2
64 EIC(CFIC,31)=-RADE(EICFIC(1,3)
CALL CXINV(EICFIC,2,EICOEF,1,3),COET,IPIV,INDX,3,ISC)
EICFIC(3,31)=-A(1,0)
70 WRITE(8,107)CRN,(EICOF(1,3),I=1,3)
CONTINUE $ GO TO 1
999 ENDFIE 7 $ REWIND 7 $ ENDFIE 8 $ REWIND 8 $ REWIND 11
REWIND 30 $ REWIND 31 $ REWIND 32 $ REWIND 33
END PROGRAM STABILITY
SUBROUTINE INICOEF(A,B,C,NMAX,VEL,VMIN,DELV,NVEL)

SUBROUTINE calculates coefficient matrices for quadratic stability

EQUIVALENCE(EQURT(1),UNCRT(1))
DIMENSION A(NMAX,1),B(NMAX,1),C(NMAX,1),EQURT(1)
COMPLEX UNCRT(6),CRAD,CSQRT

INPUT parameters are read from the input file with a namelist read
of the namelist LONGDTA and are written on tape 11 with a namelist
write statement

NAMELIST/LONGDTA/COINS,CLAD,CLQR,CMADR,CMQR,DELCD,DELCLA,DELCL,
+ DELCMA,DELCDMA,CQ,CLU,CMUR,S,CBAR,YYOI,TMASS,AAXMSS,
+ AZMASS,WTS,BUOY,BHR,BLR,SLR,CGH,CGL,TTR,CLC,COIAM,COG,WC,
+ RMC,VMIN,DELV,NVEL
COMMON/LONGDLC/COINS,DELCD,DELCLA,DELCM

PARAMETERS FOR ERC BALLOON - REFERENCE CONFIGURATION - IN SI UNITS

DATA COINS,DELCD,DELCLA,DELCM,DELCDMA,DELCMRA,DU,CLU,CMUR/
+ .010,9.0
DATA DEGRAD/57.295795130823/
DATA CMADR,CMQR,CMUR,CLAD,CLQR/-0.264,0.189,0.089,0.685/
DATA S,CBAR,YYOI/7.04,7.64,1.17/
DATA TMASS,AAXMSS,AZMASS,WTS,BUOY/14.2,5.11,23.9,108.190.0/
DATA BHR,BLR,SLR,TTR/0.2,19.6,3.44,3.82/
DATA SHR,CGH,CGL/36.109,1.10/
DATA CLC,COIAM,COG,WC/61.0141,1.17,3.43/
DATA RHO/1.225/

VELOCITY INDEPENDENT CALCULATIONS

WRITE(30,101) WRITE(31,102) WRITE(32,103) WRITE(33,104)
READ LONGDTA WRITE(11,105)
SL=SLR+CGL SL=SLR+CGL
TL=TLR-CGL TL=TLR-CGL
CBAR=5.0*CBAR R=1.0*YYOI RS=i=RHO*S*CBAR2*RI
DELMA=.5*CBAR2*RHO*CMUR
A(1,2)=A(1,3)=A(2,1)=A(2,3)=A(3,1)=A(3,3)=1.
A11,11=A1(2,2)=A(3,3)=1. RETURN

ENTRY POINT VCQEF FOR VELOCITY-DEPENDENT CALCULATIONS

ENTRY VCOEF
Q=.5*RHO*VEL*VEL

TRIM ANGLE OF ATTACK AND AERODYNAMIC COEFFICIENTS ABOUT THE CENTER

OF MASS

CALL TRIMIS,CBAR,WTS,BUOY,BL,BH,SH,SL,SL,TL,TT,CGH,CGL,Q,
+ CL,CM,CO,CLA,CMA,CDA,ALPHA,XOC,ZOC,SINA,COSA)
ALPHA=DEGRAD*ALPHA
CDA=CLA+DELCOA & CLA=CLA+DELCLA & CMA=CMA+DELCM
TRANSFER DYNAMIC STABILITY DERIVATIVES FROM REFERENCE POINT (MOMENT CENTER) TO CENTER OF MASS AND WRITE AERODYNAMIC COEFFICIENTS ON TAPE 30

CALL DERTRN(XOC,ZOC,CD,CDU,CL,CLA,CLU,CLAD,CLQR,CM,CMATA,CMADR,CMQR, 
+ CMH,CLQ,CLMD,CMQ,CMU)
WRITE(30,100)VEL,ALPHA,D,CO,CL,CM,CD,CLA,CLQ,CMAD,CMQ

CALCULATE EQUILIBRIUM CABLE CONDITIONS AND WRITE RESULTS ON TAPE 32

DRAG=CD*Q*S & BLIFT=CL*Q*S & CDENAG=CD*CDIAM*Q
T1=QRT(DRAG+BLIFT+WS+BUOY)**2
COSG1=DRAG/T1 & GAMI=ACOS(COSG1) & SING1=SIGN(GAMI)
GAMI=DEGRAD*GAMI & GAMOD=DEGRAD*GAMOD
COSG0=COS(GAMI) & SING0=SIGN(GAMI)
WRITE(32,100)VEL,X1,CL,CL,CL,CL

CALCULATE CABLE SPRINGS FROM DERIVATIVES OF NEUMARK AND TRANSFER TO STABILITY AXES - WRITE RESULTS ON TAPE 31

SM1=(BL*BUOY+SL*WTSI*SINA+(BS*BUOY+SH*WTS)*COSA
AMZ=AMASS*COSA*AMASS*SINA+SINA
AMZ=AMASS*COSA*AMASS*SINA+SINA
RMX=1./TMASS+AMX & RMZ=1./TMASS+AMZ+OELMA & ROVSMX=VEL+ROVSMX
ROVSMZ=VEL+ROVSMZ

CALCULATE COEFFICIENT MATRICES A, B, AND C

A(3,2)=CD-ROVSCI*CMAD
B(1,1)=ROVSMX*(12.0D+CD+CDU) & B(1,2)=ROVSMX*(CDA-CL) & B(1,3)=0.0
B(2,2)=ROVSMX*(12.0CL+CLU) & B(2,2)=ROVSMX*(CLAD+CD)
B(2,3)=CDAR2*ROVSMZ*(CLAD+CLU) & B(3,1)=+ROVSCI(C3.0+CM+CMU)
B(3,2)=ROVSCI*CMAD & B(3,3)=CDAR2*ROVSCI*CMAD+CMQ
C(1,1)=SKXX+RMX & C(1,2)=SKXX+RMX & C(1,3)=SKX+RMX+VEL+ROVSMX*CD
C(1,2)=SKXX+RMX & C(1,2)=SKXX+RMX & C(1,3)=SKX+RMX+VEL+ROVSMX*CD
C(1,3)=SKXX+RMX & C(1,2)=SKXX+RMX & C(1,3)=SKX+RMX+VEL+ROVSMX*CD
C(2,1)=SKXX+RMX & C(2,2)=SKXX+RMX & C(2,3)=SKX+RMX+VEL+ROVSMX*CD
C(2,3)=SKXX+RMX & C(2,3)=SKXX+RMX & C(2,3)=SKX+RMX+VEL+ROVSMX*CD
C(3,1)=SKX+RMX & C(3,2)=SKX+RMX & C(3,3)=RI*(SM1+SKTT)-VEL+ROVSCI*CMAD
CALCULATE UNCOUPLED ROOTS BY FACTORING DIAGONAL QUADRATIC TERMS AND
WRITE RESULTS ON TAPE 33

DO 1 M=1,3 $ CRAD=+.25*B(M,M)+B(M,M)-C(M,M) $ CRAD=CSQRT(CRAD)
M2=2*M $ M1=M2-1 $ UNCRT(M1)=-.5*B(M,M)+CRAD
1 UNCRT(M2)=-.5*B(M,M)-CRAD
WRITE(33,105)VEL,EQUQT(I),I=1,11,2,EQUQT(I),I=2,12,2 $ RETURN

100 FORMAT(1X11G11.4)
101 FORMAT(1X20X* AERODYNAMIC COEFFICIENTS/* VELOCITY*5X*ALPHAD*9X*C)*
+ 9X*CL*9X*CM*8X*CD*8X*CA*8X*CMAD*8X*CMQ*1

102 FORMAT(1X20X*TETHER SPRINGS/* VELOCITY*7X*SKXX*7X*SKXXZ*7X*SKXST*7X*
+ *SKZ*7X*SKZT*7X*SKTT*5X*SKTT,T1=)
103 FORMAT(1X20X*TETHER CONDITIONS/* VELOCITY*9X*19X*Z1*7X*GAMDO*9X*
+ *T0*7X*GAM1*9X*T1*3X*CA DRAG*)
104 FORMAT(1X20X*UNCOUPLED ROOTS/* VELOC*8X*RLX1*8X*RLX2*8X*RLZ1*8X*
+ *RLZ2*8X*RL1*8X*RL2*8X*RLZ*8X*RLZ2*8X+18X*IMX1*8X*IMZ1*8X*IMZ2*8X*
+ *4IT1*8X*IMT2*)
105 FORMAT(1X7G12.4/14X6G12.4)
END

SUBROUTINE TRIM(IS,ICBAR,WTS,BUOY,BL,BH,SL,TL,TT,CGH,CGI,Q,+
+ CL,CM,CD,CLA,CMA,CDA,ALPHA,~,G,SA,CA)

SUBROUTINE COMPUTES THE STATIC TRIM ANGLE-OF-ATTACK ALPHA USING
NEWTGN ITERATION OF THE TRIM EQUATION
THE ALPHA DEPENDENT DERIVATIVES CD, CDA, CL, CLA, CM, AND CMA ARE
ALSO TRANSFERRED TO THE CENTER OF MASS AND RETURNED
IF CONVERGENCE IS NOT OBTAINED IN ITCMAX ITERATIONS, MESSAGE IS
WRITTEN ON TAPE 11

CG=CMCM/CDINS+DELCD+DELCL+DELCM
ERR=1.E-6 $ TLPHA=.05 $ QS=0.5 $ ITCMAX=100 $ ITC=0
D=BUOY*BL+WTS*SL $ E=BUOY*Bl+WTS*SH
1 ALPHA=TLPHA $ CL=FCCL(ALPHA,CLA) $ CD=FCTC(ALPHA,CDA)
CMR=FCMR(ALPHA,CMAR) $ CA=COSV(ALPHA) $ SA=SIN(ALPHA)
A=TL*CA+TT*SA $ B=TT*CA-RL*SA
F=(CG+CA+CM+CM+CM)**CAPAR $ G=(CG+CA+CLSG+CM/CA)*CAPAR
CM=CMR-FCM+CC $ CM=CMAR-F*(CLA+CD)-F*(CL-CDA)
CL=CL+DELCL $ CM=CM+DELCM $ C=BUOY-WTS+QS+CL
CD=CD+DELCD+CDINS
FUN=A*C-B*CD+CM*CBAR)*QS-D*CA+E*SA
DFUN=B*C+D*SA+E*CA+4*(CLA+CD)-B*CD+CM*CBAR)*QS
TLPHA=ALPHA-FUN/DFUN $ ITC=ITC+1 $ IF(11<ITG,ITCMAX=1)GO TO 2
IF(11.4 $ ALPHA=GT1*14 $ TLPHA=14 $ ITC=ITC+1)GO TO 2
2 WRITE(11,3)ITC,ALPHA,TLPHA
3 FORMAT(1X* ITERATION FOR TRIM DID NOT CONVERGE IN*16* ITERATIONS.+
+ ALPHA=*GIZ.6* TALPHA=*G12.6/1
4 RETURN
END

SUBROUTINE TRIM
FUNCTION FCD(A, CDA)
C
FCD IS A FUNCTION TO BE WRITTEN BY THE USER TO CALCULATE CD AND CDA
AS FUNCTIONS OF ANGLE OF ATTACK FOR THE CONFIGURATION TO BE ANALYZED
C
CURVE FIT OF CD AND CDA VS ANGLE OF ATTACK IN RADIANS FOR LRC
BALLON - REFERENCE CONFIGURATION
C
X=A-.023 $ X5=X**5 $ CDA=1117.2*X5
FCD=.0487+186.2*X*X5 $ RETURN $ END

FUNCTION FCL(A, CLA)
C
FCL IS A FUNCTION TO BE WRITTEN BY THE USER TO CALCULATE CL AND CLA
AS FUNCTIONS OF ANGLE OF ATTACK FOR THE CONFIGURATION TO BE ANALYZED
C
CURVE FIT OF CL AND CLA VS ANGLE OF ATTACK IN RADIANS FOR LRC
BALLON - REFERENCE CONFIGURATION
C
X=A-.023 $ X2=X*X $ CLA=.82-X2*(15.069-557.0*X2)
FCL=X*(.82-X2*(5.023-111.4*X2)) $ RETURN $ END

FUNCTION FCMR(A, CMAR)
C
FCMR IS A FUNCTION TO BE WRITTEN BY THE USER TO CALCULATE CMR AND CMAR
AS FUNCTIONS OF ANGLE OF ATTACK FOR THE CONFIGURATION TO BE ANALYZED
C
CURVE FIT OF CM AND CMAR ABOUT REFERENCE POINT VS ANGLE OF ATTACK IN
RADIANS FOR LRC BALLON - REFERENCE CONFIGURATION
C
CMAR=.1435 $ FCMR=-.0106+.1435*A $ RETURN $ END

SUBROUTINE DERTRN(X, Z, CD, CDU, CL, CLA, CLU, CLAD, CLQR, CM, CMA, CMADR,
+ CMQR, CMUR, CLQ, CMAD, CMQ, CMU)
C
TRANSFERS CLQ, CMAD, CMQ, AND CMU FROM REFERENCE POINT (MOMENT CENTER)
TO CENTER OF MASS LOCATED X/C FORWARD AND Z/C DOWN FROM REFERENCE
C
POINT
C
CLQ=CLQR+2.*X*(CL*Z+2.*CL+CLU))
CMU=CMUR+Z*CDU-X*CLU $ CMAD=CMADR-X*CLAD
CMQ=CMQR-X*ICLQR-2.*CM+2.*Z*(2.*CM+CMU) $ RETURN $ END DERTRN
SUBROUTINE TETHERIC(DRAG, WC, CL1, T1, GAMMA1, TO, GAMMAO, X1, Z1)

SUBROUTINE IS BASED ON THE ANALYSIS IN -

NEUMARK, S.- EQUILIBRIUM CONFIGURATIONS OF FLYING CABLES OF
CAPTIVE BALLOONS, AND CABLE DERIVATIVES FOR STABILITY

THE TETHER PARAMETERS CD, WC, CL1, T1, AND GAMMA1 ARE INPUT
THE TETHER PARAMETERS TO, GAMMAO, X1, AND Z1 ARE OUTPUT

EXTERNAL FLAM, FSIG $ COMMON/PQP/P, Q

P = .5*WC/CD, Q = SQRT(1+P*P) $ EPS = 1.E-8

CALL ROMBERG(RLAM1, Q, GAMMA1, FLAM, EPS) $ TAU1 = TAU(GAMMA1)

RLAM0 = RLAM1 - CD, TAU1*CL1/T1 $ CALL NEWINT(RLAM0, FLAM, 1, GAMMA0)

CALL ROMBERG(DSIG, GAMMA0, GAMMA1, FSIG, EPS)

X1 = T1*DSIG/(CD, TAU1) $ TAU0 = TAU(GAMMA0)

TO = T1*TAU0/TAU1 $ Z1 = (T1-TO)/WC $ RETURN $ END

FUNCTION FLAMIT) $ COMMON/PQP/P, Q $ CT = COS(T)

FLAM = ((Q+P-CT)/(Q-P+CT)**(P/Q)/(1-CT*CT+2*P*CT)) $ RETURN $ END

FUNCTION FSIGIT) $ COMMON/PQP/P, Q $ CT = COS(T)

FSIG = ((Q+P-CT)/(Q-P+CT)**(P/Q)/(1-CT*CT+2*P*CT)*CT $ RETURN $ END

FUNCTION TAUIT) $ COMMON/PQP/P, Q $ CT = COS(T)

TAU = ((Q+P-CT)/(Q-P+CT)**(P/Q) $ RETURN $ END

SUBROUTINE NEWINT(C, F, XO, X) $ EPS = 1.E-8 $ XT = XO $ I = 0

SUBROUTINE COMPUTES THE UPPER LIMIT X OF THE DEFINITE INTEGRAL
FROM 0 TO X OF THE FUNCTION F FOR WHICH THE VALUE OF THE
INTEGRAL C IS KNOWN

NEWTON ITERATION IS USED WITH THE VALUE OF THE INTEGRAL DETERMINED
BY SUBROUTINE ROMBERG

I = I+1 $ P12 = 1.570796326 $ IF (XT, GT, PI2) XT = PI2 $ IF (XT, LT, 0.) XT = 0.

IF (I, LT, 2) GO TO 1 $ IF (ABS(X-XT), GT, EPS) GO TO 1

X = XT $ RETURN $ END SUBROUTINE NEWINT
SUBROUTINE ROMBERG(SUM, A, B, FUN, EPS) $ DIMENSION Q(20)
*
SUBROUTINE FOR ROMBERG QUADRATURE - SEE WRITEUP FOR DESCRIPTION *
*
SUM=0 $ IF(A.EQ.B) RETURN
H=B-A $ FA=FUN(A) $ FB=FUN(B) $ FM=AMAX1(ABS(FA),ABS(FB))
T=.5*H*(FA+FB) $ NX=1 $ DO 5 N=1,19 $ H=.5*T $ SUM=0 $ I=0
1 $ I=I+1 $ FX=FUN(A+H*(I+I-1)) $ IF(ABS(FX).GT.FM) FM=ABS(FX)
SUM=SUM+FX $ IF(I.LT.NX) GO TO 1
T=.5*T+H $ SUM=SUM+T
Q=N+0.666666666666667*(T+H*SUM)
IF(N.LT.2) GO TO 4 $ F=4. $ DO 2 J=2,N $ I=N+1-J $ F=6.*F
2 Q(I)=Q(I+1)+(Q(I+1)-Q(I))/F-1. $ IF(N.LT.3) GO TO 3
X=ABS(Q(I)-QX2)+ABS(QX2-QX1) $ IF(X.EQ.EPS*FM*ABS(B-A)) GO TO 6
3 QX1=QX2
4 QX2=QX2
5 NX=NX+NX
6 SUM=Q(1) $ RETURN
END SUBROUTINE

SUBROUTINE QUAD(A, B, C, NMAX, NMAX2, N, IOP, EIGDET, CNO)
*
SEE SUBROUTINE WRITEUP FOR DESCRIPTION *
*
COMMON/IROW,ICOL/IROW(300),ICOL(300)
DIMENSION KR(7),A(NMAX,1),B(NMAX,1),C(NMAX,1),EIGDET(NMAX2,1)
AN=V $ AINV=0. $ RINDF=0$1777.000000000001777
NS1=NMAX2+NMAX2-NSQ $ NS2=NS1-NSQ $ NS3=NS2-NSQ $ NS3P1=NS3+1
DO 1 J=1,N $ DO 1 I=1,N $ IE=I+N*J-N $ EIGDET(NS3+IE)=A(I,J)
1 EIGDET(NS1+IE)=EIGDET(NS2+IE)=EIGDET(NS3+IE)=C(I,J)
2 A=AN+A(I,J)*A(I,J) $ CALL MASCNTR(KR,EIGDET(NS3P1),DET,C)
IF(IOP.NE.10)GO TO 2 $ CNO=SQRT(AN.AINV/NSQ)
3 AINV=AINV+EIGDET(NS3+J)*EIGDET(NS3+J) $ CNO=SQRT(AN.AINV/NSQ)
4 DO 5 J=1,N $ DO 5 I=1,N $ IE=I+N*J-N $ EIGDET(NS2+IE)=0.
5 EIGDET(I,J)=-EIGDET(NS2+IE) $ DO 6 J=1,N $ DO 6 I=1,N $ IE=I+N*J-N
6 EIGDET(I,J)=EIGDET(NS1+IE) $ DO 7 J=1,N $ DO 7 I=1,N
7 EIGDET(I,J)=EIGDET(I,J)+0. $ DO 8 J=1,N
8 EIGDET(I,J)=EIGDET(I,J) $ IF(NSQ.EQ.10)RETURN
2 N2P1=N+N+1 $ DO 9 I=1$N2P1 $ DO 9 J=1,N $ EIGDET(I,J)=RINDF
9 EIGDET(I,J)=RINDF $ RETURN
END SUBROUTINE QUAD

SUBROUTINE MATRIX(A,N,N,K,A,K,B,K,C,K)
*
SEE SUBROUTINE WRITEUP FOR DESCRIPTION *
*
COMMON/IROW,ICOL/IROW(300),ICOL(300) $ DIMENSION A(1),B(1),KR(7)
$ KR(1)=I $ IF(I.NE.10)GO TO 2 $ S=0 $ DO 1 L=1,M $ DO 1 J=1,M
1 S=S+A(L+J)*KA-KA $ KA=KA $ T=0
2 S=0 $ KA=0 $ KR(4)=K $ KR(5)=KA $ KR(6)=KB $ KR(7)=KC
CALL MASCNTR(KR,A,B,C) $ IF(I.NE.10)GO TO 4 $ DO 3 L=1,M $ DO 3 J=1,M
3 T=TA(L+J)*KA-KA $ A(L+J)*KA-KA $ C=SQR(S*T)/M
4 RETURN
END SUBROUTINE MATRIX
SUBROUTINE REIG (A,M,NVAL,NVEC,ROOTR,ROOTI,VEC,MAX,IDX,IRN,P,NP, 
1 SAVE) 
C PROGRAM TO CALL QR TRANSFORMATION, VARIABLE DIMENSION 
C MAXIMUM ITER IS 50 
C 
C THIS VERSION OF REIG HAS BEEN ALTERED SUCH THAT EIGENVECTORS ARE 
C NOT CALCULATED. SUBROUTINE VECTOR HAS BEEN DELETED AND VEC DOES 
C NOT HAVE TO BE DIMENSIONED 
C 
C SEE SUBROUTINE WRITEUP FOR DESCRIPTION 
C 
DIMENSION A(MAX,MAX),ROOTR(MAX),ROOTI(MAX),VEC(MAX,MAX),IDX(MAX), 
1 IRN(MAX),P(MAX),SAVE(MAX,NP) 
REAL IDX,IRN 
N = M 
C SAVE ORIGINAL MATRIX, RESTORE AT 200 
DO 5 I=1,M 
DO 5 J=1,M 
5 SAVE(I,J) = A(I,J) 
C REDUCE MATRIX TO HESSENBERG FORM 
CALL HESSEN (A,M,MAX) 
ZERCL = 0.0 
JJ = 1 
177 XNN = 0.0 
XN2 = 0.0 
AA = 0.0 
B = 0.0 
C = 0.0 
DD = 0.0 
R = 0.0 
SIG = 0.0 
ITER=0 
IF IN-Z) 13,14,12 
13 ROOTR(1) = refl 
ROOTI(1) = 0.0 
GO TO 200 
14 JJ= -1 
12 X = (A(N-1,N-1) - A(N,N))**2 
S = 4.0*A(N,N-1)*A(N-1,N) 
ITER = ITER + 1 
IF (X .EQ. 0.0) GO TO 15 
IF (ABS(S/X) .GT. 1.0E-8) GO TO 15 
IF (ABS(A(N-1,N-1)) - ABS(A(N,N))) 32,32,31 
31 E = A(N-1,N-1) 
G = A(N,N) 
GO TO 33 
32 G = A(N-1,N-1) 
E = A(N,N) 
33 F = C*0 
H = 0.0 
GO TO 24 
15 S = X + S 
X = A(N-1,N-1) + A(N,N) 
SQ = SQRT(ABS(S)) 
IF (SQ) 18,19,19 
18 F = SQ 
H = C*0 
IF (X) 21,21,22
21 E = (X-SQ)/2.0
G = (X+SQ)/2.0
GO TO 24
22 G = (X-SQ)/2.0
E = (X+SQ)/2.0
GO TO 24
18 F = SQ/2.0
E = SQ/G
H = -F
24 IF (JJ .LT. 0) GO TO 28
D = 1.0E-10*(ABS(E) + F)
IF (ABS(A(N-1,N-2)) .GT. D) GO TO 26
28 ROOTR(N) = E
ROUTI(N) = F
ROOTR(N-1) = G
ROUTI(N-1) = H
N = N-2
IF (JJ) 200,177,177
26 IF (ABS(A(N,N-1)) .GT. 1.0E-10*ABS(A(N,N))) GO TO 50
29 ROOTR(N) = A(N,N)
ROUTI(N) = 0.0
N = N-1
GO TO 177
50 IF (ABS(A(N,N-1)) - 1.0E-6) 63,63,62
62 IF (ABS(A(N-1,N-2)) - 1.0E-6) 63,63,700
63 VQ = ABS(A(N,N-1) - ABS(A(N-1,N-2))
IF (ITER-15) 53,164,64
164 IF (VQ) 165,165,166
165 R = A(N-1,N-2)**2
SIG = 2.0*A(N-1,N-2)
GO TO 60
166 R = A(N,N-1)**2
SIG = 2.0*A(N,N-1)
GO TO 60
64 IF (VQ) 29,29,28
700 IF (ITER .GT. 50) GO TO 63
IF (ITER .GT. 5) GO TO 53
Z1 = ((E-AA)**2+(F-B)**2)/(E*E+F*F)
Z2 = (G-C)**2+(H-DD)**2)/(G*G+H*H)
IF (Z1-0.25) 51,51,52
51 IF (Z2-0.25) 53,53,54
53 R = E*E-F*H
SIG = E*G
GO TO 60
54 R = E*F
SIG = E*E
GO TO 60
52 IF (Z2-0.25) 55,55,601
55 R = G*G
SIG = G*G
GO TO 60
601 R = 0.0
SIG = 0.0
60 XNN = A(N,N-1)
XN2 = A(N-1,N-2)
CALL QT(A,N,R,SIG,D,MAX)
AA = E
B = F
C = G
DD = H
GO TO 12
C  RESTORE MATRIX
230 DO 210 J=1,M
    DO 210 I=1,M
    A(I,J) = SAVE(I,J)
C    TEST FOR COMPLEX ROOTS
N = 0
DO 225 I=1,M
IF (ROOTRI(I) .EQ. 0.) GO TO 212
IF (ABS(ROOTRI(I)/ROOTRI(I)) .GE. 1.E-12) GO TO 215
212 IF (ABS(ROOTRI(I)) .LT. 1.E-12) GO TO 218
C    INDEX FOR COMPLEX (END OF ARRAYS)
215 NC = NC-1
    JM = JC
    GO TO 220
C    INDEX FOR REAL ROOTS SAME, N = NO. REAL ROOTS
218 N = N+1
    JM = N
220 IRN(JM) = ROOTRI(I)
225 IDX(JM) = ROOTRI(I)
C    REAL ROOTS IN DESCENDING ORDER BY MAGNITUDE
DO 240 I=1,M
    IF (I .GE. N) GO TO 235
    K = I+1
    DO 230 J=K,N
        IF (ABS(IRN(I)) .GE. ABS(IRN(J))) GO TO 230
        SIG = IRN(J)
        IRN(J) = IRN(I)
        IRN(I) = SIG
    230 CONTINUE
235 ROOTRI(I) = IRN(I)
    ROOTRI(I) = IDX(I)
C    STORE ZERO IN VECTOR
C    STORE ZERO IN VECTOR DELETED FROM THIS VERSION
C    DO 240 J=1,M
C    240 VEC(J,I) = 0.0
    240 CONTINUE
    IF (NVEC .LT. N) N = NVEC
    IF (N .LE. 0) GO TO 250
    DO 245 I = 1,N
        K = N+1-I
        IF (ABS(ROOTRI(K)/ROOTRI(I)) .LT. 1.E-14) GO TO 248
    245 ROOTRI(K) = 0.
    248 CONTINUE
C    CALL ROUTINE FOR N VECTORS
C    CALL ROUTINE FOR N VECTORS DELETED FROM THIS VERSION
C    CALL VECTOR (IDX,IRN,ROOTRI,A,VEC,M,SAVE,P,NP,MAX,N)
250 RETURN
END
SUBROUTINE QRT(A,N,R,SIG,D,MAX)

DIMENSION A(MAX,MAX),PSI(2),G(3)

N1 = N - 1
IA = N - 2
IP = IA
IF(N-3) 1C1,10,60
60 DO 12 J = 3,N1
J1 = N - J
IF(ABS(A(J1+1,J1+1))<10,10,11
11 DEN = A(J1+1,J1+1)*(A(J1+1,J1+1)-SIG)+A(J1+1,J1+2)*A(J1+2,J1+1)+R
IF(DEN) 61,12,61
61 IF(ABS(A(J1+1,J1+1))<ABS(A(J1+1,J1+1))+ABS(A(J1+2,J1+2)))DEN)-D) 10,10,11
12 IP=J1
10 DO 14 J=1,IP
J1=IP-J+1
IF(ABS(A(J1+1,J1+1))<13,13,14
14 IQ=J1
13 DO 100 I=IP,N1
IF(1-IP) 16,15,16
15 G(1)=A(IP,IP)*A(IP,IP)-SIG)+A(IP+1,IP)*A(IP+1,IP)+R
G(2)=A(IP+1,IP)*A(IP,IP)+A(IP+1,IP)-SIG)
G(3)=A(IP+1,IP)+A(IP+1,IP)+1
A(IP+2,IP)=0.0
GO TO 19
16 G(1)=A(I+1,1-1)
G(2)=A(I+1,1-1)
IF(I-1A) 17,17,18
17 G(3)=A(I+2,1-1)
GO TO 19
18 G13=0.0
19 XK = SIGN(SORT(G(1)**2 + G(2)**2 + G(3)**2), G(1))
22 IF(XK) 23,24,23
23 AL=G(1)/XK+1.0
PSI(1)=G(2)/(G(1)*XK)
PSI(2)=G(3)/(G(1)*XK)
GO TO 25
24 AL=2.0
PSI(1)=0.0
PSI(2)=0.0
25 IF(I-10) 26,27,26
26 IF(I-1P) 29,28,29
28 A(I-1-1)=-A(I-1-1)
GO TO 27
29 A(I-1-1)=-XK
27 DO 30 J=I,N
IF(I-1A) 31,31,32
31 C=PSI(2)*A(I+2,J)
GO TO 33
32 C=0.0
33 E=AL*(A(I,J)+PSI(1)*A(I+1,J)+C)
A(I,J)=A(I,J)-E
A(I+1,J)=A(I+1,J)+PSI(1)*E
IF(I-1A) 34,34,30
34 A(I+2,J)=A(I+2,J)+PSI(2)*E
30 CONTINUE
IF(I-1A) 35,35,36
35 L=I+2
GO TO 37
36 L=N
37 DO 40 J=10,L
40 CONTINUE

SUBROUTINE HESSEN(A,M,MAX)
C SUBROUTINE TO PUT MATRIX IN UPPER HESSENBERG FORM.
DIMENSION A(MAX,MAX), B(99)
DOUBLE PRECISION SUM
 IF (M - 2) 30,30,32
32 DO 40 LC = 3, M
 N = M - LC + 3
 N1 = N - 1
 N2 = N - 2
 N1 = N1
 DIV = ABS(A(N, N-1))
 DD 2 J = 1, N2
 IF (ABS(A(N, J)) - DIV) 2, 2, 1
1  N1 = J
 DIV = ABS(A(N, J))
2 CONTINUE
 IF (DIV) 3, 40, 3
3 IF (N1 - N1) 4, 7, 4
4 DO 5 J = 1, N
 DIV = A(J, N1)
 A(J, N1) = A(J, N1)
5 A(J, N1) = DIV
 DD 6 J = 1, M
 DIV = A(N1, J)
 A(N1, J) = A(N1, J)
6 A(N1, J) = DIV
7 DD 26 K = 1, N1
26 B(K) = A(N, K) / A(N, N-1)
 DD 45 J = 1, M
 SUM = 0.0
 IF (J - N1) 46, 43, 43
46 IF (A(J)) 41, 43, 41
41 A(N, J) = 0.0
 DD 42 K = 1, N1
 A(K, J) = A(K, J) - A(K, N1) * B(J)
42 SUM = SUM + A(K, J) * B(J)
 GO TO 45
43 DO 44 K = 1, N1
44 SUM = SUM + A(K, J) * B(K)
45 A(N1, J) = SUM
40 CONTINUE
30 RETURN
END

IF(I-1A) 38, 38, 39
38 C=PSI(2)*A(J, I+2)
 GO TO 41
39 C=0.0
41 E=AL*(A(J, I)*PSI(I)*A(J, I+1)+C)
 A(J, I)=A(J, I)-E
 A(J, I+1)=A(J, I+1)-PSI(I)*E
 IF(I-1A) 42, 42, 40
42 A(J, I+2)=A(J, I+2)-PSI(I)*E
40 CONTINUE
 IF(I-N+3) 43, 43, 100
43 E=AL*PSI(2*I+1)*AL(I+1)+C)
 A(I+1,N)=E
 A(I+3, l+l)=PSI(I)*E
 A(I+3, l+2)=A(I+3, I+2)-PSI(I)*E
100 CONTINUE
101 RETURN
ENC

*QRST0062 *
*QRST0063 *
*QRST0064 *
*QRST0065 *
*QRST0066 *
*QRST0067 *
*QRST0068 *
*QRST0069 *
*QRST0070 *
*QRST0071 *
*QRST0072 *
*QRST0073 *
*QRST0074 *
*QRST0075 *
*QRST0076 *
*QRST0077 *
*QRST0078 *
*QRST0079
SUBROUTINE CXINV(A,N,B,M,DET,PIV,INDX,MAX,ISCALE)

COMPLEX MATRIX INVERSION WITH SOLUTION OF LINEAR EQUATIONS

CAVM = CABS(A(MAX)), CAVA = CABS(A(I,J))
CAM = CABS(CTERM), CAPV = CABS(PIV)

COMPLEX A(MAX,N), B(MAX,M), SWAP, DET, PIV, PIVI, CO, CI
DIMENSION (PIV(N)), INDX(MAX,2)

CONSTANTS, INITIALIZATION

CO = (0.0,0.0)
CI = (1.0,0.0)
ISCALE = 0
RL = 10.0**100
RS = 1.0/RL
DET = CI
CAM = 1.0
DO 20 J=1,N
20 IPIV(J) = 0
DO 500 I=1,N

SEARCH FOR PIVOT ELEMENT

CAVM = 0.0
DO 105 J=1,N
IF (IPIV(J) .EQ. 1) GO TO 105
DO 100 K=1,N
IF (IPIV(K) - 1) .LT. 50,100,750
CONTINUE
CAVM = CABS(A(J,K))
IF (CAVM .GE. CAVA) GO TO 100
ICOL = K
CAVM = CAVA
100 CONTINUE
105 CONTINUE
IF (CAVM .EQ. 0.0) GO TO 720
IPIV(ICOL) = IPIV(ICOL) + 1

INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL

IF (IROW .EQ. ICOL) GO TO 230
DET = -DET
DO 200 L=1,N
SWAP = A(IROW,L)
A(IROW,L) = A(ICOL,L)
A(ICOL,L) = SWAP
200 CONTINUE
IF (IM .LE. 0) GO TO 230
DO 220 L=1,M
SWAP = B(IROW,L)
B(IROW,L) = B(ICOL,L)
B(ICOL,L) = SWAP
220 CONTINUE

INDEX(I,1) = IROW
INDEX(I,2) = ICOL
PIV = A(ICOL,ICOL)
CAPV = CABS(PIV)
IF (CAPV .EQ. 0.0) GO TO 720
C SCALE DETERMINANT
C
PIV = PIV
CADM = CABS(DET)
IF (CADM .LT. RL) GO TO 260
DET = DET/RL
CADM = CABS(DET)
ISCALE = ISCALE + 1
IF (CADM .LT. RL) GO TO 290
DET = DET/RL
ISCALE = ISCALE + 1
GO TO 290
260 CONTINUE
IF (CADM .GT. RS) GO TO 290
DET = DET*RL
CADM = CABS(DET)
ISCALE = ISCALE - 1
IF (CADM .GT. RS) GO TO 290
DET = DET/RL
ISCALE = ISCALE - 1
290 CONTINUE
CAPV = CABS(PIV)
IF (CAPV .LT. RL) GO TO 320
PIV = PIV/RL
CAPV = CABS(PIV)
ISCALE = ISCALE + 1
IF (CAPV .LT. RL) GO TO 340
PIV = PIV/RL
ISCALE = ISCALE + 1
GO TO 340
320 CONTINUE
IF (CAPV .GT. RS) GO TO 340
PIV = PIV*RL
CAPV = CABS(PIV)
ISCALE = ISCALE - 1
IF (CAPV .GT. RS) GO TO 340
PIV = PIV*RL
ISCALE = ISCALE - 1
340 CONTINUE
DET = DET * PIV
C CIVIDE PIVOT ROW BY PIVOT ELEMENT
C
A(IICCL,ICOL) = C1
DO 350 L=1,N
350 A(IICCL,L) = A(IICCL,L)/PIV
IF (M .LE. 0) GO TO 380
DO 370 L=1,M
370 B(IICCL,L) = B(IICCL,L)/PIV
C REDUCE NON-PIVOT ROWS
C
DO 380 L=1,N
IF (L .EQ. ICOL) GO TO 380
SWAP = A(L1,ICOL)
A(L1,ICOL) = C0
DO 400 L=1,N
400 A(L1,L) = A(L1,L) - A(ICOL,L)*SWAP
   IF (M <= 0) GO TO 500
   DO 450 L=1,M
   450 B(L1,L) = B(L1,L) - B(ICOL,L)*SWAP
   500 CONTINUE
C INTERCHANGE COLUMNS
C
DO 700 I=1,N
   L = N+1-I
   IF (INDEX(L,1).EQ. INDEX(L,2)) GO TO 700
   IROW = INDEX(L,1)
   ICOL = INDEX(L,2)
   DO 690 K=1,N
   SWAP = A(K, IROW)
   A(K, IROW) = A(K, ICOL)
   A(K, ICOL) = SWAP
690 CONTINUE
700 CONTINUE
GO TO 750
720 DET = CO
ISCALE = 0
750 RETURN
END
Printing of Files Containing Calculated Results
for Sample Case

A printing of each BCD file for the sample case is given. The headings essentially give the quantities printed using the FORTRAN variable names as previously given. The files and their contents are as follows:

<table>
<thead>
<tr>
<th>File name</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>OUTPUT</td>
<td>case identification, date, and time only</td>
</tr>
<tr>
<td>TAPE11</td>
<td>printing of input data NAMELIST and the NPOS dimensional (1/sec) characteristic roots with $\text{Im}(\lambda) \geq 0$ ($\text{NPOS} = \text{NTWO} - \text{NEGR} + 1$); velocity is printed followed by $\text{Re}(\lambda)$, SYMBOL, and $\text{Im}(\lambda)$</td>
</tr>
<tr>
<td>TAPE8</td>
<td>each characteristic root (with $\text{Im}(\lambda) \geq 0$) and the corresponding eigenvector is printed for every value of velocity</td>
</tr>
<tr>
<td>TAPE30</td>
<td>velocity, trim angle of attack, and the aerodynamic coefficients about the center of mass</td>
</tr>
<tr>
<td>TAPE31</td>
<td>dimensional tether derivatives or spring constants about the center of mass</td>
</tr>
<tr>
<td>TAPE32</td>
<td>dimensional balloon position and tether conditions (note that $\text{CAB \ DRAG} = n$)</td>
</tr>
<tr>
<td>TAPE33</td>
<td>the uncoupled roots of the x-, z-, and $\theta$-equations are calculated by factoring the diagonal quadratic factors of the stability determinant and are associated with the FORTRAN variable name UNCRT; both roots are printed (even if $\text{Im}(\lambda) &lt; 0$); the headings $\text{RLX1}$ and $\text{IMX1}$ denote the real and imaginary parts of the uncoupled roots associated with the x-equation, etc., where the letters $X$, $Z$, and $T$ in the headings indicate x-, z-, and $\theta$-equations, respectively</td>
</tr>
</tbody>
</table>
Listing of OUTPUT for sample case (identification card, and date and time only).

LONGITUDINAL STABILITY OF TETHERED BALLOON - LPC BALLOON-REFERENCE CONFIGURATION 11/20/72 09:56:50.

Listing of tape 11 for sample case (principal file for characteristic roots).

LONGOTA
CODINS = 0.1E+01,
CLAD = 0.89E-01,
CLOR = 0.685E+00,
CMQR = -0.25E-01,
CMQR = -0.189E+00,
CELCC = 0.0,
DELCOA = 0.0,
DELCL = 0.0,
DELCLA = 0.0,
DELCM = 0.0,
DELCOA = 0.0,
COU = 0.0,
CLU = 0.0,
CMUR = 0.0,
S = 0.754E+01,
CHAP = 0.764E+01,
YV01 = 0.117E+03,
TMSS = 0.142E+02,
ARMSS = 0.511E+01,
AZMSS = 0.245E+02,
WTS = 0.105E+03,
BUCY = 0.191E+03,
BHR = 0.0,
BLP = 0.215E+01,
SHR = 0.38E+00,
SLR = 0.66E+00,
CGM = 0.105E+00,
GOL = 0.11E+01,
TGR = 0.344E+01,
FIR = 0.362E+01,
CLC = 0.61E+02,
GOL = 0.142E+01,
COC = 0.117E+01,
MC = 0.343E+00,
PHG = 0.122E+01,
WMX = 0.1E+01,
DELV = 0.1E+01,
NEWL = 51,
RENQ
<p>| 1.00000 | -1.269213E-02 | -4.11192E-02 | 94.1124 |
| 2.00000 | -2.519945E-02 | -7.99310E-02 | 23.7948 |
| 3.00000 | -3.815506E-02 | -1.149889 | 10.8043 |
| 4.00000 | -5.281444E-02 | -1.45607 | 6.29106 |
| 5.00000 | -7.054334E-02 | -1.70315 | 4.23316 |
| 6.00000 | -9.24617E-02 | -1.84626 | 3.13920 |
| 7.00000 | -1.194622 | -1.82504 | 2.50597 |
| 8.00000 | -1.524797 | -1.71934 | 2.13306 |
| 9.00000 | -1.926966 | -1.76669 | 1.92381 |
| 10.0000 | -2.419910 | -2.06734 | 1.77962 |
| 11.0000 | -3.374131 | -2.59846 | 1.72567 |
| 12.0000 | -3.834708 | -3.36016 | 1.68768 |
| 13.0000 | -4.235635 | -4.30800 | 1.63994 |
| 14.0000 | -4.804029 | -5.57332 | 1.72591 |
| 15.0000 | -7.178237 | -6.67256 | 1.81445 |
| 16.0000 | -6.945949 | -7.58069 | 1.92471 |
| 17.0000 | -6.953798 | -8.35324 | 2.04656 |
| 18.0000 | -7.083373 | -9.04329 | 2.17151 |
| 19.0000 | -7.228525 | -9.98825 | 2.30225 |
| 20.0000 | -7.543117 | -2.62989 | 2.30669 |
| 21.0000 | -7.816099 | -1.08745 | 2.57437 |
| 22.0000 | -9.012151 | -1.04652 | 2.71495 |
| 23.0000 | -9.444514 | -1.25057 | 2.89814 |
| 24.0000 | -9.782829 | -1.25992 | 3.00367 |
| 25.0000 | -9.127651 | -1.31075 | 3.15128 |
| 26.0000 | -9.482432 | -1.35623 | 3.30075 |</p>
<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th>X1</th>
<th>Y1</th>
<th>Z1</th>
<th>X2</th>
<th>Y2</th>
<th>Z2</th>
</tr>
</thead>
<tbody>
<tr>
<td>27.0000</td>
<td>-984363</td>
<td>-5.09577</td>
<td>7.007234E-02</td>
<td>-1.41944</td>
<td>3.45187</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28.0000</td>
<td>-1.02099</td>
<td>-5.30202</td>
<td>0.0</td>
<td>8.252775E-02</td>
<td>-1.47346</td>
<td>3.60444</td>
<td></td>
<td></td>
</tr>
<tr>
<td>29.0000</td>
<td>-1.05803</td>
<td>-5.50688</td>
<td>0.0</td>
<td>9.436669E-02</td>
<td>-1.52734</td>
<td>3.75828</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30.0000</td>
<td>-1.09540</td>
<td>-5.71054</td>
<td>0.0</td>
<td>1.05860</td>
<td>-1.58112</td>
<td>3.91325</td>
<td></td>
<td></td>
</tr>
<tr>
<td>31.0000</td>
<td>-1.13304</td>
<td>-5.91315</td>
<td>0.0</td>
<td>1.16553</td>
<td>-1.63484</td>
<td>4.06919</td>
<td></td>
<td></td>
</tr>
<tr>
<td>32.0000</td>
<td>-1.17089</td>
<td>-6.11485</td>
<td>0.0</td>
<td>1.27040</td>
<td>-1.68850</td>
<td>4.22598</td>
<td></td>
<td></td>
</tr>
<tr>
<td>33.0000</td>
<td>-1.20892</td>
<td>-6.31574</td>
<td>0.0</td>
<td>1.37167</td>
<td>-1.74213</td>
<td>4.38350</td>
<td></td>
<td></td>
</tr>
<tr>
<td>34.0000</td>
<td>-1.24709</td>
<td>-6.51593</td>
<td>0.0</td>
<td>1.47037</td>
<td>-1.79574</td>
<td>4.54166</td>
<td></td>
<td></td>
</tr>
<tr>
<td>35.0000</td>
<td>-1.28538</td>
<td>-6.71545</td>
<td>0.0</td>
<td>1.56584</td>
<td>-1.84933</td>
<td>4.70035</td>
<td></td>
<td></td>
</tr>
<tr>
<td>36.0000</td>
<td>-1.32376</td>
<td>-6.91450</td>
<td>0.0</td>
<td>1.65928</td>
<td>-1.90291</td>
<td>4.85950</td>
<td></td>
<td></td>
</tr>
<tr>
<td>37.0000</td>
<td>-1.36222</td>
<td>-7.11301</td>
<td>0.0</td>
<td>1.75064</td>
<td>-1.95649</td>
<td>5.01904</td>
<td></td>
<td></td>
</tr>
<tr>
<td>38.0000</td>
<td>-1.40074</td>
<td>-7.31108</td>
<td>0.0</td>
<td>1.84016</td>
<td>-2.01007</td>
<td>5.17891</td>
<td></td>
<td></td>
</tr>
<tr>
<td>39.0000</td>
<td>-1.43931</td>
<td>-7.50876</td>
<td>0.0</td>
<td>1.92802</td>
<td>-2.06364</td>
<td>5.33904</td>
<td></td>
<td></td>
</tr>
<tr>
<td>40.0000</td>
<td>-1.47792</td>
<td>-7.70609</td>
<td>0.0</td>
<td>2.01438</td>
<td>-2.11722</td>
<td>5.49939</td>
<td></td>
<td></td>
</tr>
<tr>
<td>41.0000</td>
<td>-1.51656</td>
<td>-7.90310</td>
<td>0.0</td>
<td>2.09939</td>
<td>-2.17080</td>
<td>5.65962</td>
<td></td>
<td></td>
</tr>
<tr>
<td>42.0000</td>
<td>-1.55523</td>
<td>-8.09983</td>
<td>0.0</td>
<td>2.18317</td>
<td>-2.22438</td>
<td>5.82055</td>
<td></td>
<td></td>
</tr>
<tr>
<td>43.0000</td>
<td>-1.59391</td>
<td>-8.29630</td>
<td>0.0</td>
<td>2.26584</td>
<td>-2.27797</td>
<td>5.98136</td>
<td></td>
<td></td>
</tr>
<tr>
<td>44.0000</td>
<td>-1.63261</td>
<td>-8.49254</td>
<td>0.0</td>
<td>2.34750</td>
<td>-2.33156</td>
<td>6.14221</td>
<td></td>
<td></td>
</tr>
<tr>
<td>45.0000</td>
<td>-1.67132</td>
<td>-8.68857</td>
<td>0.0</td>
<td>2.42822</td>
<td>-2.38515</td>
<td>6.30311</td>
<td></td>
<td></td>
</tr>
<tr>
<td>46.0000</td>
<td>-1.71003</td>
<td>-8.88442</td>
<td>0.0</td>
<td>2.50810</td>
<td>-2.43874</td>
<td>6.46404</td>
<td></td>
<td></td>
</tr>
<tr>
<td>47.0000</td>
<td>-1.74875</td>
<td>-9.08009</td>
<td>0.0</td>
<td>2.58720</td>
<td>-2.49234</td>
<td>6.62497</td>
<td></td>
<td></td>
</tr>
<tr>
<td>48.0000</td>
<td>-1.78747</td>
<td>-9.27561</td>
<td>0.0</td>
<td>2.66550</td>
<td>-2.54594</td>
<td>6.78590</td>
<td></td>
<td></td>
</tr>
<tr>
<td>49.0000</td>
<td>-1.82619</td>
<td>-9.47098</td>
<td>0.0</td>
<td>2.74330</td>
<td>-2.59954</td>
<td>6.94680</td>
<td></td>
<td></td>
</tr>
<tr>
<td>50.0000</td>
<td>-1.86491</td>
<td>-9.66623</td>
<td>0.0</td>
<td>2.82040</td>
<td>-2.65315</td>
<td>7.10767</td>
<td></td>
<td></td>
</tr>
<tr>
<td>51.0000</td>
<td>-1.90362</td>
<td>-9.86136</td>
<td>0.0</td>
<td>2.89794</td>
<td>-2.70675</td>
<td>7.26849</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Listing of tape B for sample case (modal ratios or eigenvectors and characteristic roots).

**LONGITUDINAL STABILITY OF TETHERED BALLOON - LGT BALLON-REFERENCE CONFIGURATION 11/20/72 CS.56.5C.**

### EIGENVECTORS

<table>
<thead>
<tr>
<th>VELOCITY</th>
<th>( \text{REAL, IMAG} )</th>
<th>( \text{REAL, IMAG} )</th>
<th>( \text{REAL, IMAG} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2650</td>
<td>-1.5448, -9.14418</td>
<td>6.2704E-02, -1.2075E-03</td>
<td>1.0000, 0.0000</td>
</tr>
<tr>
<td>-1.5448</td>
<td>6.2704E-02, -1.2075E-03</td>
<td>1.0000, 0.0000</td>
<td></td>
</tr>
<tr>
<td>-1.2650</td>
<td>1.0000, 0.0000</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### VELOCITY 

| 2.0000   | -1.3579, -7.72946E-03     | -7.72946E-03, -1.3579    | 1.0000, 0.0000           |
| -1.3579  | -7.72946E-03, -1.3579     | 1.0000, 0.0000           |                           |
| -2.0000  | 1.0000, 0.0000            |                           |                           |

### VELOCITY 

| 3.0000   | -7.72946E-03, -1.3579     | -1.3579, -7.72946E-03    | 1.0000, 0.0000           |
| -7.72946E-03 | -1.3579, -7.72946E-03    | 1.0000, 0.0000           |                           |
| -3.0000  | 1.0000, 0.0000            |                           |                           |

### VELOCITY 

| 4.0000   | -2.78004E-02, -1.8075E-03 | -1.8075E-03, -2.78004E-02 | 1.0000, 0.0000           |
| -2.78004E-02 | -1.8075E-03, -2.78004E-02 | 1.0000, 0.0000           |                           |
| -4.0000  | 1.0000, 0.0000            |                           |                           |

### VELOCITY 

| 5.0000   | -3.3943, -2.5285E-03      | -2.5285E-03, -3.3943     | 1.0000, 0.0000           |
| -3.3943  | -2.5285E-03, -3.3943      | 1.0000, 0.0000           |                           |
| -5.0000  | 1.0000, 0.0000            |                           |                           |

### VELOCITY 

| 6.0000   | -6.3115, -4.5764E-03      | -4.5764E-03, -6.3115     | 1.0000, 0.0000           |
| -6.3115  | -4.5764E-03, -6.3115      | 1.0000, 0.0000           |                           |
| -6.0000  | 1.0000, 0.0000            |                           |                           |

### VELOCITY 

| 7.0000   | -1.1337, -5.6181E-03      | -5.6181E-03, -1.1337     | 1.0000, 0.0000           |
| -1.1337  | -5.6181E-03, -1.1337      | 1.0000, 0.0000           |                           |
| -7.0000  | 1.0000, 0.0000            |                           |                           |

### VELOCITY 

| 8.0000   | -3.3943, -2.5285E-03      | -2.5285E-03, -3.3943     | 1.0000, 0.0000           |
| -3.3943  | -2.5285E-03, -3.3943      | 1.0000, 0.0000           |                           |
| -8.0000  | 1.0000, 0.0000            |                           |                           |

### VELOCITY 

| 9.0000   | -6.3115, -4.5764E-03      | -4.5764E-03, -6.3115     | 1.0000, 0.0000           |
| -6.3115  | -4.5764E-03, -6.3115      | 1.0000, 0.0000           |                           |
| -9.0000  | 1.0000, 0.0000            |                           |                           |

### VELOCITY 

| 10.0000  | -1.1337, -5.6181E-03      | -5.6181E-03, -1.1337     | 1.0000, 0.0000           |
| -1.1337  | -5.6181E-03, -1.1337      | 1.0000, 0.0000           |                           |
| -10.0000 | 1.0000, 0.0000            |                           |                           |

### VELOCITY 

| 11.0000  | -3.3943, -2.5285E-03      | -2.5285E-03, -3.3943     | 1.0000, 0.0000           |
| -3.3943  | -2.5285E-03, -3.3943      | 1.0000, 0.0000           |                           |
| -11.0000 | 1.0000, 0.0000            |                           |                           |

### VELOCITY 

<p>| 12.0000  | -6.3115, -4.5764E-03      | -4.5764E-03, -6.3115     | 1.0000, 0.0000           |
| -6.3115  | -4.5764E-03, -6.3115      | 1.0000, 0.0000           |                           |
| -12.0000 | 1.0000, 0.0000            |                           |                           |</p>
<table>
<thead>
<tr>
<th>VELOCITY</th>
<th>13.000</th>
<th>14.000</th>
<th>15.000</th>
<th>16.000</th>
<th>17.000</th>
<th>18.000</th>
<th>19.000</th>
<th>20.000</th>
<th>21.000</th>
<th>22.000</th>
<th>23.000</th>
<th>24.000</th>
<th>25.000</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1.2356</td>
<td>-1.6903</td>
<td>-1.2172</td>
<td>-1.6958</td>
<td>-1.7034</td>
<td>-1.7262</td>
<td>-1.9125</td>
<td>-1.8703</td>
<td>-1.9089</td>
<td>-1.8649</td>
<td>-1.7352</td>
<td>-1.8050</td>
<td>-1.7173</td>
<td></td>
</tr>
<tr>
<td>-4.3055</td>
<td>-1.3517</td>
<td>-3.5015</td>
<td>-3.5007</td>
<td>-1.1244</td>
<td>-1.0144</td>
<td>-0.9843</td>
<td>-1.0135</td>
<td>-2.0452</td>
<td>-2.2922</td>
<td>-1.9056</td>
<td>-2.2922</td>
<td>-2.8672</td>
<td></td>
</tr>
<tr>
<td>-1.4899</td>
<td>-1.3383</td>
<td>-2.6991</td>
<td>-2.6991</td>
<td>-1.0148</td>
<td>-2.0452</td>
<td>-1.0385</td>
<td>-1.0791</td>
<td>-3.5015</td>
<td>-3.5007</td>
<td>-1.1244</td>
<td>-1.0135</td>
<td>-0.9843</td>
<td></td>
</tr>
<tr>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td></td>
</tr>
</tbody>
</table>

**Values:**
- VELOCITY: Velocity increments
- Values represent the change in velocity over time.
<table>
<thead>
<tr>
<th>VELOCITY</th>
<th>26.000</th>
<th>3.0000</th>
<th>5.78345E-03</th>
<th>2.972522E-02</th>
<th>2.93792</th>
<th>3.66463E-02</th>
<th>1.0000</th>
<th>0.0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-1.656</td>
<td>7.1767</td>
<td>5.78343E-03</td>
<td>2.972522E-02</td>
<td>2.93792</td>
<td>3.66463E-02</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>1.758</td>
<td>3.0000</td>
<td>5.78345E-03</td>
<td>2.972522E-02</td>
<td>2.93792</td>
<td>3.66463E-02</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>-1.087</td>
<td>7.1767</td>
<td>5.78343E-03</td>
<td>2.972522E-02</td>
<td>2.93792</td>
<td>3.66463E-02</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>1.758</td>
<td>3.0000</td>
<td>5.78345E-03</td>
<td>2.972522E-02</td>
<td>2.93792</td>
<td>3.66463E-02</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>-1.087</td>
<td>7.1767</td>
<td>5.78343E-03</td>
<td>2.972522E-02</td>
<td>2.93792</td>
<td>3.66463E-02</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>1.758</td>
<td>3.0000</td>
<td>5.78345E-03</td>
<td>2.972522E-02</td>
<td>2.93792</td>
<td>3.66463E-02</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>-1.087</td>
<td>7.1767</td>
<td>5.78343E-03</td>
<td>2.972522E-02</td>
<td>2.93792</td>
<td>3.66463E-02</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>1.758</td>
<td>3.0000</td>
<td>5.78345E-03</td>
<td>2.972522E-02</td>
<td>2.93792</td>
<td>3.66463E-02</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>-1.087</td>
<td>7.1767</td>
<td>5.78343E-03</td>
<td>2.972522E-02</td>
<td>2.93792</td>
<td>3.66463E-02</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>1.758</td>
<td>3.0000</td>
<td>5.78345E-03</td>
<td>2.972522E-02</td>
<td>2.93792</td>
<td>3.66463E-02</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>-1.087</td>
<td>7.1767</td>
<td>5.78343E-03</td>
<td>2.972522E-02</td>
<td>2.93792</td>
<td>3.66463E-02</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>1.758</td>
<td>3.0000</td>
<td>5.78345E-03</td>
<td>2.972522E-02</td>
<td>2.93792</td>
<td>3.66463E-02</td>
<td>1.0000</td>
<td>0.0</td>
</tr>
</tbody>
</table>

**Note:** The table above contains numerical data, possibly related to velocities or similar measurements. The placeholders for values are indicated by `0.` and the columns are labeled with placeholders for calculations or measurements. The values are represented in scientific notation.
<table>
<thead>
<tr>
<th>VEL.X</th>
<th>VEL.Y</th>
<th>VEL.Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1.6326</td>
<td>-0.71423</td>
<td>-0.88844</td>
</tr>
<tr>
<td>-0.7875</td>
<td>-0.69724</td>
<td>-1.7100</td>
</tr>
<tr>
<td>-2.3316</td>
<td>6.1</td>
<td>-1.6713</td>
</tr>
<tr>
<td>-2.7068</td>
<td>-0.68714</td>
<td>-1.5552</td>
</tr>
<tr>
<td>-2.4923</td>
<td>-0.70085</td>
<td>-0.23069</td>
</tr>
<tr>
<td>-0.28204</td>
<td>2.1327</td>
<td>-5.3390</td>
</tr>
<tr>
<td>-2.7433</td>
<td>2.9943</td>
<td>-5.8208</td>
</tr>
</tbody>
</table>

**Note:** The table above represents velocity components in a specific coordinate system.
### Listing of tape 30 for sample case (trim angle of attack and aerodynamic coefficients)

**Coningutational Stability of Yawed Ballon - LRL Ballon-Reference Configuration 11/20/72 05:56:00.**

<table>
<thead>
<tr>
<th>VELOCITY</th>
<th>ALPHA</th>
<th>CO</th>
<th>CL</th>
<th>CM</th>
<th>DDA</th>
<th>CLA</th>
<th>CMA</th>
<th>CLQ</th>
<th>CMAD</th>
<th>CMO</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000</td>
<td>5.194</td>
<td>5.193</td>
<td>5.476</td>
<td>5.900</td>
<td>5.210</td>
<td>5.476</td>
<td>5.900</td>
<td>5.210</td>
<td>5.476</td>
<td>5.900</td>
</tr>
<tr>
<td>2.000</td>
<td>5.193</td>
<td>5.194</td>
<td>5.476</td>
<td>5.900</td>
<td>5.210</td>
<td>5.476</td>
<td>5.900</td>
<td>5.210</td>
<td>5.476</td>
<td>5.900</td>
</tr>
<tr>
<td>3.000</td>
<td>5.192</td>
<td>5.193</td>
<td>5.476</td>
<td>5.900</td>
<td>5.210</td>
<td>5.476</td>
<td>5.900</td>
<td>5.210</td>
<td>5.476</td>
<td>5.900</td>
</tr>
<tr>
<td>4.000</td>
<td>5.191</td>
<td>5.192</td>
<td>5.476</td>
<td>5.900</td>
<td>5.210</td>
<td>5.476</td>
<td>5.900</td>
<td>5.210</td>
<td>5.476</td>
<td>5.900</td>
</tr>
<tr>
<td>5.000</td>
<td>5.190</td>
<td>5.191</td>
<td>5.476</td>
<td>5.900</td>
<td>5.210</td>
<td>5.476</td>
<td>5.900</td>
<td>5.210</td>
<td>5.476</td>
<td>5.900</td>
</tr>
</tbody>
</table>

...
<table>
<thead>
<tr>
<th>VELOCITY</th>
<th>SKXX</th>
<th>SKX2</th>
<th>SKX3</th>
<th>SKX4</th>
<th>SKX5</th>
<th>SKX6</th>
<th>SKX7</th>
<th>SKX8</th>
<th>SKX9</th>
<th>SKX10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000</td>
<td>4.483</td>
<td>16.17</td>
<td>1.273</td>
<td>0.280</td>
<td>20.57</td>
<td>1.455</td>
<td>0.273</td>
<td>0.273</td>
<td>0.273</td>
<td>0.273</td>
</tr>
<tr>
<td>2.000</td>
<td>4.483</td>
<td>16.17</td>
<td>1.273</td>
<td>0.280</td>
<td>20.57</td>
<td>1.455</td>
<td>0.273</td>
<td>0.273</td>
<td>0.273</td>
<td>0.273</td>
</tr>
<tr>
<td>3.000</td>
<td>4.483</td>
<td>16.17</td>
<td>1.273</td>
<td>0.280</td>
<td>20.57</td>
<td>1.455</td>
<td>0.273</td>
<td>0.273</td>
<td>0.273</td>
<td>0.273</td>
</tr>
<tr>
<td>4.000</td>
<td>4.483</td>
<td>16.17</td>
<td>1.273</td>
<td>0.280</td>
<td>20.57</td>
<td>1.455</td>
<td>0.273</td>
<td>0.273</td>
<td>0.273</td>
<td>0.273</td>
</tr>
</tbody>
</table>

**Listing of tape 31 for sample case (tether spring constants)**

**LIGNTUDINAL STABILITY OF TETHERED BALLOON - LRC BALLOON-REFERENCE CONFIGURATION 11/20/77 09:56:50.**

**TETHER SPRINGS**
### Listing of tape 32 for sample case (equilibrium tether conditions).

**LONGITUDINAL STABILITY OF TETHERED BALLOON - LFG BALLOON-REFERENCE CONFIGURATION 1120/72 05.56.50.**

<table>
<thead>
<tr>
<th>VELOCITY</th>
<th>1</th>
<th>1A</th>
<th>1B</th>
<th>1C</th>
<th>1D</th>
<th>4A</th>
<th>4B</th>
<th>4C</th>
<th>4D</th>
<th>7A</th>
<th>7B</th>
<th>7C</th>
<th>7D</th>
<th>10A</th>
<th>10B</th>
<th>10C</th>
<th>10D</th>
<th>11A</th>
<th>11B</th>
<th>11C</th>
<th>11D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000</td>
<td>.9513</td>
<td>.6170</td>
<td>.5040</td>
<td>.4961</td>
<td>.4962</td>
<td>.4963</td>
<td>.4964</td>
<td>1.0104</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Listing of tape 33 for sample case (uncoupled characteristic roots).

LONGITUDINAL STABILITY OF TETHERED BALLOON - LRC BALLOON-REFERENCE CONFIGURATION 11/20/72 09.56.50.

<table>
<thead>
<tr>
<th>VEL</th>
<th>RL1</th>
<th>RL2</th>
<th>RL3</th>
<th>RL4</th>
<th>RL5</th>
<th>RL6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000</td>
<td>-3.015E-02</td>
<td>-3.015E-02</td>
<td>-3.199E-02</td>
<td>-3.199E-02</td>
<td>-1.169</td>
<td>-1.169</td>
</tr>
<tr>
<td>2.000</td>
<td>-2.5943E-02</td>
<td>-2.5943E-02</td>
<td>-8.5662E-02</td>
<td>-8.5662E-02</td>
<td>-2.337</td>
<td>-2.337</td>
</tr>
<tr>
<td>3.000</td>
<td>-3.8935E-02</td>
<td>-3.8935E-02</td>
<td>-4.3199E-02</td>
<td>-4.3199E-02</td>
<td>-6.6639</td>
<td>-6.6639</td>
</tr>
<tr>
<td>8.000</td>
<td>-1.1037</td>
<td>-1.1037</td>
<td>-12.755</td>
<td>-12.755</td>
<td>-42.32</td>
<td>-42.32</td>
</tr>
<tr>
<td>9.000</td>
<td>-1.297</td>
<td>-1.297</td>
<td>-14.049</td>
<td>-14.049</td>
<td>-49.3</td>
<td>-49.3</td>
</tr>
<tr>
<td>10.00</td>
<td>-1.427</td>
<td>-1.427</td>
<td>-15.34</td>
<td>-15.34</td>
<td>-56.28</td>
<td>-56.28</td>
</tr>
<tr>
<td>11.00</td>
<td>-1.557</td>
<td>-1.557</td>
<td>-16.63</td>
<td>-16.63</td>
<td>-63.26</td>
<td>-63.26</td>
</tr>
<tr>
<td>12.00</td>
<td>-1.687</td>
<td>-1.687</td>
<td>-18.01</td>
<td>-18.01</td>
<td>-70.24</td>
<td>-70.24</td>
</tr>
<tr>
<td>13.00</td>
<td>-1.817</td>
<td>-1.817</td>
<td>-19.39</td>
<td>-19.39</td>
<td>-77.22</td>
<td>-77.22</td>
</tr>
<tr>
<td>14.00</td>
<td>-1.947</td>
<td>-1.947</td>
<td>-20.77</td>
<td>-20.77</td>
<td>-84.2</td>
<td>-84.2</td>
</tr>
<tr>
<td>15.00</td>
<td>-2.077</td>
<td>-2.077</td>
<td>-22.15</td>
<td>-22.15</td>
<td>-91.18</td>
<td>-91.18</td>
</tr>
<tr>
<td>16.00</td>
<td>-2.207</td>
<td>-2.207</td>
<td>-23.53</td>
<td>-23.53</td>
<td>-98.15</td>
<td>-98.15</td>
</tr>
<tr>
<td>17.00</td>
<td>-2.337</td>
<td>-2.337</td>
<td>-24.91</td>
<td>-24.91</td>
<td>-98.15</td>
<td>-98.15</td>
</tr>
<tr>
<td>18.00</td>
<td>-2.467</td>
<td>-2.467</td>
<td>-26.29</td>
<td>-26.29</td>
<td>-98.15</td>
<td>-98.15</td>
</tr>
<tr>
<td>19.00</td>
<td>-2.597</td>
<td>-2.597</td>
<td>-27.67</td>
<td>-27.67</td>
<td>-98.15</td>
<td>-98.15</td>
</tr>
<tr>
<td>20.00</td>
<td>-2.727</td>
<td>-2.727</td>
<td>-29.05</td>
<td>-29.05</td>
<td>-98.15</td>
<td>-98.15</td>
</tr>
<tr>
<td>21.00</td>
<td>-2.857</td>
<td>-2.857</td>
<td>-30.43</td>
<td>-30.43</td>
<td>-98.15</td>
<td>-98.15</td>
</tr>
<tr>
<td>22.00</td>
<td>-2.986</td>
<td>-2.986</td>
<td>-31.81</td>
<td>-31.81</td>
<td>-98.15</td>
<td>-98.15</td>
</tr>
<tr>
<td>23.00</td>
<td>-3.116</td>
<td>-3.116</td>
<td>-33.19</td>
<td>-33.19</td>
<td>-98.15</td>
<td>-98.15</td>
</tr>
<tr>
<td>24.00</td>
<td>-3.246</td>
<td>-3.246</td>
<td>-34.57</td>
<td>-34.57</td>
<td>-98.15</td>
<td>-98.15</td>
</tr>
<tr>
<td>25.00</td>
<td>-3.376</td>
<td>-3.376</td>
<td>-36.05</td>
<td>-36.05</td>
<td>-98.15</td>
<td>-98.15</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>26.00</td>
<td>-0.3376</td>
<td>-0.3376</td>
<td>-1.147</td>
<td>-1.147</td>
<td>-1.567</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.629</td>
<td>-2.629</td>
<td>.9318</td>
<td>-0.9316</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>27.00</td>
<td>-0.3306</td>
<td>-0.3306</td>
<td>-1.192</td>
<td>-1.192</td>
<td>-1.608</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.755</td>
<td>-2.755</td>
<td>.9662</td>
<td>-0.9662</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>28.00</td>
<td>-0.3036</td>
<td>-0.3036</td>
<td>-1.236</td>
<td>-1.236</td>
<td>-1.652</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.881</td>
<td>-2.881</td>
<td>.9619</td>
<td>-0.9619</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>29.00</td>
<td>-0.3766</td>
<td>-0.3766</td>
<td>-1.261</td>
<td>-1.261</td>
<td>-1.689</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.008</td>
<td>-3.008</td>
<td>.9789</td>
<td>-0.9789</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>30.00</td>
<td>-0.3896</td>
<td>-0.3896</td>
<td>-1.325</td>
<td>-1.325</td>
<td>-1.749</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.136</td>
<td>-3.136</td>
<td>.9989</td>
<td>-0.9989</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>31.00</td>
<td>-0.3026</td>
<td>-0.3026</td>
<td>-1.370</td>
<td>-1.370</td>
<td>-1.801</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.285</td>
<td>-3.285</td>
<td>1.015</td>
<td>-1.015</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>32.00</td>
<td>-0.3156</td>
<td>-0.3156</td>
<td>-1.414</td>
<td>-1.414</td>
<td>-1.825</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.354</td>
<td>-3.354</td>
<td>1.036</td>
<td>-1.036</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>33.00</td>
<td>-0.3266</td>
<td>-0.3266</td>
<td>-1.458</td>
<td>-1.458</td>
<td>-1.841</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.524</td>
<td>-3.524</td>
<td>1.057</td>
<td>-1.057</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>34.00</td>
<td>-0.3416</td>
<td>-0.3416</td>
<td>-1.503</td>
<td>-1.503</td>
<td>-1.809</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.654</td>
<td>-3.654</td>
<td>1.078</td>
<td>-1.078</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>35.00</td>
<td>-0.3946</td>
<td>-0.3946</td>
<td>-1.547</td>
<td>-1.547</td>
<td>-2.027</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.785</td>
<td>-3.785</td>
<td>1.100</td>
<td>-1.100</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>36.00</td>
<td>-0.4476</td>
<td>-0.4476</td>
<td>-1.592</td>
<td>-1.592</td>
<td>-2.067</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.916</td>
<td>-3.916</td>
<td>1.123</td>
<td>-1.123</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>37.00</td>
<td>-0.4906</td>
<td>-0.4906</td>
<td>-1.636</td>
<td>-1.636</td>
<td>-2.119</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.047</td>
<td>-4.047</td>
<td>1.147</td>
<td>-1.147</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>38.00</td>
<td>-0.4526</td>
<td>-0.4526</td>
<td>-1.681</td>
<td>-1.681</td>
<td>-2.111</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.178</td>
<td>-4.178</td>
<td>1.170</td>
<td>-1.170</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>39.00</td>
<td>-0.5066</td>
<td>-0.5066</td>
<td>-1.725</td>
<td>-1.725</td>
<td>-2.274</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.310</td>
<td>-4.310</td>
<td>1.195</td>
<td>-1.195</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>40.00</td>
<td>-0.5196</td>
<td>-0.5196</td>
<td>-1.765</td>
<td>-1.765</td>
<td>-2.338</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.441</td>
<td>-4.441</td>
<td>1.219</td>
<td>-1.219</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>41.00</td>
<td>-0.5326</td>
<td>-0.5326</td>
<td>-1.814</td>
<td>-1.814</td>
<td>-2.423</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.573</td>
<td>-4.573</td>
<td>1.245</td>
<td>-1.245</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>42.00</td>
<td>-0.5555</td>
<td>-0.5555</td>
<td>-1.868</td>
<td>-1.868</td>
<td>-2.534</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.705</td>
<td>-4.705</td>
<td>1.270</td>
<td>-1.270</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>43.00</td>
<td>-0.5585</td>
<td>-0.5585</td>
<td>-1.902</td>
<td>-1.902</td>
<td>-2.534</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.836</td>
<td>-4.836</td>
<td>1.296</td>
<td>-1.296</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>44.00</td>
<td>-0.5715</td>
<td>-0.5715</td>
<td>-1.947</td>
<td>-1.947</td>
<td>-2.609</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.968</td>
<td>-4.968</td>
<td>1.322</td>
<td>-1.322</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>45.00</td>
<td>-0.5945</td>
<td>-0.5945</td>
<td>-1.991</td>
<td>-1.991</td>
<td>-2.667</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5.100</td>
<td>-5.100</td>
<td>1.348</td>
<td>-1.348</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>46.00</td>
<td>-0.5575</td>
<td>-0.5575</td>
<td>-2.036</td>
<td>-2.036</td>
<td>-2.735</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5.231</td>
<td>-5.231</td>
<td>1.375</td>
<td>-1.375</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>47.00</td>
<td>-0.6105</td>
<td>-0.6105</td>
<td>-2.080</td>
<td>-2.080</td>
<td>-2.802</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5.363</td>
<td>-5.363</td>
<td>1.401</td>
<td>-1.401</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>48.00</td>
<td>-0.6235</td>
<td>-0.6235</td>
<td>-2.124</td>
<td>-2.124</td>
<td>-2.871</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5.495</td>
<td>-5.495</td>
<td>1.428</td>
<td>-1.428</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>49.00</td>
<td>-0.5369</td>
<td>-0.5369</td>
<td>-2.169</td>
<td>-2.169</td>
<td>-2.939</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5.626</td>
<td>-5.626</td>
<td>1.456</td>
<td>-1.456</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>50.00</td>
<td>-0.6495</td>
<td>-0.6495</td>
<td>-2.213</td>
<td>-2.213</td>
<td>-3.006</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5.758</td>
<td>-5.758</td>
<td>1.483</td>
<td>-1.483</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>51.00</td>
<td>-0.6825</td>
<td>-0.6825</td>
<td>-2.258</td>
<td>-2.258</td>
<td>-3.077</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5.889</td>
<td>-5.889</td>
<td>1.510</td>
<td>-1.510</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>
LATERAL STABILITY PROGRAM

The main program for the lateral stability program is the same as for the main program for the longitudinal stability program with the exception of the formats for labeling. In this case, however, rows 1, 2, and 3 of the matrices A, B, and C correspond to the coefficients of the y-, φ-, and ψ-equations, respectively. The general organization of subroutine INICOEF with entry point VCOEF is also similar to the organization of the longitudinal program. The only lateral tether spring is in the y-direction of the earth-axis system at the tether point (ref. 2) which is calculated in function subroutine YSUBY. The related springs about the balloon center of mass are then calculated from the y-spring. The FORTRAN variable names for the tether springs and their definitions are:

<table>
<thead>
<tr>
<th>FORTRAN variable name</th>
<th>Mathematical symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SKPP</td>
<td>k_φφ</td>
<td>spring constant for roll displacement</td>
</tr>
<tr>
<td>SKPS</td>
<td>k_φψ</td>
<td>rolling moment due to yawing displacement</td>
</tr>
<tr>
<td>SKPY</td>
<td>k_φγ</td>
<td>rolling moment due to y-displacement</td>
</tr>
<tr>
<td>SKSP</td>
<td>k_ψφ</td>
<td>yawing moment due to rolling displacement</td>
</tr>
<tr>
<td>SKSY</td>
<td>k_ψγ</td>
<td>yawing moment due to y-displacement</td>
</tr>
<tr>
<td>SKSS</td>
<td>k_ψψ</td>
<td>spring constant for yaw displacement</td>
</tr>
<tr>
<td>SKYP</td>
<td>k_yφ</td>
<td>y-force due to roll displacement</td>
</tr>
<tr>
<td>SKYS</td>
<td>k_yψ</td>
<td>y-force due to yaw displacement</td>
</tr>
<tr>
<td>SKYY</td>
<td>k_yy</td>
<td>spring constant for y-displacement</td>
</tr>
</tbody>
</table>

Trim or equilibrium conditions are calculated by the same procedures and subroutines as used in the longitudinal program. The lateral stability derivatives are defined about the reference point by user-written function subprograms and are transferred to the center of mass. The variable names for the derivatives about the center of mass are as follows:
<table>
<thead>
<tr>
<th>FORTRAN variable name</th>
<th>Mathematical symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLB</td>
<td>( C_{l\beta} )</td>
<td>( \frac{\partial C_l}{\partial \beta} )</td>
</tr>
<tr>
<td>CLBD</td>
<td>( C_{l\beta} )</td>
<td>( \frac{\partial C_l}{\partial \frac{r_c}{2V}} )</td>
</tr>
<tr>
<td>CLP</td>
<td>( C_{l_p} )</td>
<td>( \frac{\partial C_l}{\partial \frac{p_c}{2V}} )</td>
</tr>
<tr>
<td>CLR</td>
<td>( C_{l_r} )</td>
<td>( \frac{\partial C_l}{\partial \frac{r_c}{2V}} )</td>
</tr>
<tr>
<td>CNB</td>
<td>( C_{n\beta} )</td>
<td>( \frac{\partial C_n}{\partial \beta} )</td>
</tr>
<tr>
<td>CNBD</td>
<td>( C_{n\beta} )</td>
<td>( \frac{\partial C_n}{\partial \frac{r_c}{2V}} )</td>
</tr>
<tr>
<td>CNP</td>
<td>( C_{n_p} )</td>
<td>( \frac{\partial C_n}{\partial \frac{p_c}{2V}} )</td>
</tr>
<tr>
<td>CNR</td>
<td>( C_{n_r} )</td>
<td>( \frac{\partial C_n}{\partial \frac{r_c}{2V}} )</td>
</tr>
<tr>
<td>CYB</td>
<td>( C_{Y\beta} )</td>
<td>( \frac{\partial C_Y}{\partial \beta} )</td>
</tr>
<tr>
<td>CYBD</td>
<td>( C_{Y\beta} )</td>
<td>( \frac{\partial C_Y}{\partial \frac{r_c}{2V}} )</td>
</tr>
<tr>
<td>CYR</td>
<td>( C_{Y_r} )</td>
<td>( \frac{\partial C_Y}{\partial \frac{r_c}{2V}} )</td>
</tr>
<tr>
<td>CYP</td>
<td>( C_{Y_p} )</td>
<td>( \frac{\partial C_Y}{\partial \frac{p_c}{2V}} )</td>
</tr>
</tbody>
</table>

The limitations and diagnostics for the lateral program are essentially those given for the longitudinal program. In addition, the body reference axis for \( \alpha_t = \theta = 0 \) is assumed to be the principal axis for which \( I_{xz} = 0 \) (where \( I_{xz} \) is product of inertia).
Lateral Equations of Motion

The equations of lateral motion written about the center of mass are (see ref. 9):

\[ \ddot{y} - \frac{\rho V S c}{2m_y} C_{Y_\beta} \dot{y} + \frac{k_y y}{m_y} y - \frac{\rho V S c}{4m_y} C_{Y_P} \dot{\phi} + \left( \frac{k_y \phi}{m_y} - \frac{\rho V^2 S c L}{2m_y} \right) \phi + \frac{\rho V^2 S c}{4m_y} \left( C_{Y_\beta} - C_{Y_P} \right) \dot{\psi} + \frac{\rho V^2 S \left( C_{Y_\beta} + C_D \right)}{2m_y} + \frac{k_y \psi}{m_y} \psi = 0 \]  \hspace{1cm} (19)

Rolling moment

\[ -\frac{\rho S c^2}{4I_x} C_{l_\beta} \ddot{y} - \frac{\rho V S c}{2I_x} C_{l_\beta} \dot{y} + \frac{k_{\phi y}}{I_x} y + \frac{k_{\phi \phi}}{I_x} \phi + \frac{\rho V^2 S c}{4I_x} C_{l_P} \phi + \frac{\rho V S c}{4I_x} \left( C_{l_\beta} - C_{l_P} \right) \dot{\psi} + \frac{k_{\phi \psi}}{I_x} \psi + \frac{\rho V^2 S \left( c C_{l_\beta} - h k_2 C_D \right)}{2I_x} \psi = 0 \]  \hspace{1cm} (20)

Yawing moment

\[ -\frac{\rho S c^2}{4I_z} C_{n_\beta} \ddot{y} - \frac{\rho V S c}{2I_z} C_{n_\beta} \dot{y} + \frac{k_{\psi y}}{I_z} y + \frac{k_{\psi \psi}}{I_z} \psi + \frac{\rho V S c^2}{4I_z} C_{n_P} \phi + \frac{\rho V^2 S \left( c C_{n_\beta} - h k_1 C_D \right)}{2I_z} \psi + \frac{k_{\psi \psi}}{I_z} \psi + \frac{\rho V^2 S \left( c C_{n_\beta} + h k_1 C_D \right)}{2I_z} \psi = 0 \]  \hspace{1cm} (21)

The definition of \( M_{S_1} \) is given by equation (6), \( M_{S_2} \) by equation (9), and

\[ m_y = m_T + m_{y, a} - \frac{\rho S c}{4} C_{Y_\beta} \]

\[ I_x = I_{xx} \cos^2 \alpha_t + I_{zz} \sin^2 \alpha_t \]
\[ I_z = I_{zz} \cos^2 \alpha_t + I_{xx} \sin^2 \alpha_t \]
\[ I_{xz} = -\frac{I_{zz} - I_{xx}}{2} \sin 2\alpha_t = (I_{xx} - I_{zz}) \sin \alpha_t \cos \alpha_t \]

The tether derivatives about the center of gravity are

\[ k_{y\phi} = -h_{k2}k_{yy} \]  \hspace{1cm} (22a)

\[ k_{y\psi} = h_{k1}k_{yy} \]  \hspace{1cm} (22b)

\[ k_{\phi y} = k_{y\phi} \]  \hspace{1cm} (22c)

\[ k_{\phi \phi} = h_{k2}^2k_{yy} \]  \hspace{1cm} (22d)

\[ k_{\phi \psi} = k_{\psi \phi} \]  \hspace{1cm} (22e)

\[ k_{\psi y} = k_{y\psi} \]  \hspace{1cm} (22f)

\[ k_{\psi \phi} = -h_{k1}h_{k2}k_{yy} \]  \hspace{1cm} (22g)

\[ k_{\psi \psi} = h_{k1}^2k_{yy} \]  \hspace{1cm} (22h)

and \( h_{k1} \) and \( h_{k2} \) are given by equations (8).

The spring constant \( k_{yy} \) is based on the analysis of reference 2 and is given by

\[ k_{yy} = \frac{n (\tau_1 \sin^2 \gamma_1 + 2\bar{p} \cos \gamma_1)^{1/2}}{\int_{\gamma_0}^{\gamma_1} \frac{\tau(\gamma)}{\sin^2 \gamma + 2\bar{p} \cos \gamma} \, d\gamma} \]  \hspace{1cm} (23)

The above expression is evaluated by function subprogram YSUBY which calls subroutine ROMBERG to evaluate the integral numerically.

The lateral stability derivatives are also given about the reference point and are transferred to the center of mass by the following relations:
\[ C_{Y_T} = C_{Y_T,R} \cdot \frac{2x_t}{c} \cdot C_{Y_{\beta}} \]
\[ C_{Y_P} = C_{Y_P,R} + \frac{2z_t}{c} \cdot C_{Y_{\beta}} \]
\[ C_{n_{\beta}} = C_{n_{\beta,R}} - \frac{x_t}{c} \cdot C_{Y_{\beta}} \]
\[ C_{n_{\beta'}} = C_{n_{\beta',R}} - \frac{x_t}{c} \cdot C_{Y_{\beta}} \]
\[ C_{n_{r}} = C_{n_{r,R}} - \frac{x_t}{c} \left( C_{Y_{r,R}} + 2C_{n_{\beta}} \right) \]
\[ C_{n_{p}} = C_{n_{p,R}} + \frac{x_t}{c} \cdot C_{Y_{p,R}} + \frac{2z_t}{c} \cdot C_{n_{\beta}} \]
\[ C_{l_{\beta}} = C_{l_{\beta,R}} + \frac{z_t}{c} \cdot C_{Y_{\beta}} \]
\[ C_{l_{\beta'}} = C_{l_{\beta',R}} + \frac{z_t}{c} \cdot C_{Y_{\beta}} \]
\[ C_{l_{p}} = C_{l_{p,R}} + \frac{z_t}{c} \left( C_{Y_{p,R}} + 2C_{l_{\beta}} \right) \]
\[ C_{l_{T}} = C_{l_{T,R}} + \frac{z_t}{c} \cdot C_{Y_{T,R}} - \frac{2x_t}{c} \cdot C_{l_{\beta}} \]

**Input Required for Lateral Stability Program**

The user-written function subprograms FCD, FCL, and FCMM for the longitudinal static aerodynamic coefficients as described for the longitudinal program are also required for the lateral stability program as the lateral program also calculates longitudinal trim conditions. In addition, the 12 lateral stability derivatives for the configuration are described as user-written function subroutines with trim angle of attack as a formal parameter. Each function is written about the reference point and is transferred to the center of mass by DERTRMN. It might also be noted that the definitions of the lateral derivatives are conventional (ref. 11) except that they are based on the reference length \( \bar{c} \) and the reference area \( S \).
For each case, one card of 80 characters of case identification is read in an 8A10 format, and a namelist called LATDATA is read. The FORTRAN variable names, their equivalent mathematical symbols, and their definitions are listed in the NAMELIST which is also the order for printing. All variables are preset in the program with DATA statements to values for the reference configuration of the LRC balloon and only changes need to be read with the NAMELIST.

<table>
<thead>
<tr>
<th>FORTRAN variable name</th>
<th>Mathematical symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDINS</td>
<td>$C_{D_{ins}}$</td>
<td>constant increment of $C_D$ (allows for $C_D$ of instrument package of balloon)</td>
</tr>
<tr>
<td>DELCD</td>
<td>$\Delta C_D$</td>
<td>constant increments in coefficients about center of mass which are used for parametric studies</td>
</tr>
<tr>
<td>DELCL</td>
<td>$\Delta C_L$</td>
<td>constant increments in coefficients about center of mass which are used for parametric studies</td>
</tr>
<tr>
<td>DELCM</td>
<td>$\Delta C_m$</td>
<td>constant increments in lateral stability derivatives about center of mass which are used for parametric studies</td>
</tr>
<tr>
<td>DELCLB</td>
<td>$\Delta C_{L\beta}$</td>
<td>constant increments in lateral stability derivatives about center of mass which are used for parametric studies</td>
</tr>
<tr>
<td>DELCLBD</td>
<td>$\Delta C_{L\beta}$</td>
<td>constant increments in lateral stability derivatives about center of mass which are used for parametric studies</td>
</tr>
<tr>
<td>DELCLP</td>
<td>$\Delta C_{Lp}$</td>
<td>constant increments in lateral stability derivatives about center of mass which are used for parametric studies</td>
</tr>
<tr>
<td>DELCLR</td>
<td>$\Delta C_{Lr}$</td>
<td>constant increments in lateral stability derivatives about center of mass which are used for parametric studies</td>
</tr>
<tr>
<td>DELCNP</td>
<td>$\Delta C_{Np}$</td>
<td>constant increments in lateral stability derivatives about center of mass which are used for parametric studies</td>
</tr>
<tr>
<td>DELCNR</td>
<td>$\Delta C_{Nr}$</td>
<td>constant increments in lateral stability derivatives about center of mass which are used for parametric studies</td>
</tr>
<tr>
<td>DELCYB</td>
<td>$\Delta C_{Y\beta}$</td>
<td>constant increments in lateral stability derivatives about center of mass which are used for parametric studies</td>
</tr>
<tr>
<td>DELCYBD</td>
<td>$\Delta C_{Y\beta}$</td>
<td>constant increments in lateral stability derivatives about center of mass which are used for parametric studies</td>
</tr>
<tr>
<td>DELCYP</td>
<td>$\Delta C_{Yp}$</td>
<td>constant increments in lateral stability derivatives about center of mass which are used for parametric studies</td>
</tr>
<tr>
<td>DELCYR</td>
<td>$\Delta C_{Yr}$</td>
<td>constant increments in lateral stability derivatives about center of mass which are used for parametric studies</td>
</tr>
<tr>
<td>FORTRAN variable name</td>
<td>Mathematical symbol</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------</td>
<td>------------</td>
</tr>
<tr>
<td>RATIOKY</td>
<td></td>
<td>factor multiplying calculated $y$-spring, $k_{yy}$ ($SKYY$); used for parametric studies</td>
</tr>
<tr>
<td>S</td>
<td>$S$</td>
<td>reference area, $(\text{Volume of balloon})^{2/3}$</td>
</tr>
<tr>
<td>CBAR</td>
<td>$\bar{c}$</td>
<td>reference length, balloon body length used here</td>
</tr>
<tr>
<td>XXOI</td>
<td>$I_{xx}$</td>
<td>rolling inertia about axis through balloon center of mass parallel to body reference $X$-axis including aerodynamic apparent inertia, $\alpha_t = 0$</td>
</tr>
<tr>
<td>ZZOI</td>
<td>$I_{zz}$</td>
<td>yawing inertia about axis through balloon center of mass parallel to body reference $Z$-axis including aerodynamic apparent inertia, $\alpha_t = 0$</td>
</tr>
<tr>
<td>TMASS</td>
<td>$m_T$</td>
<td>mass of balloon structure and contained gas</td>
</tr>
<tr>
<td>AYMASS</td>
<td>$m_{y,a}$</td>
<td>aerodynamic apparent mass in body-reference $y$-axis direction, $\alpha_t = 0$</td>
</tr>
<tr>
<td>WTS</td>
<td>$W_s$</td>
<td>structural weight of balloon</td>
</tr>
<tr>
<td>BUOY</td>
<td>$B$</td>
<td>net buoyancy force</td>
</tr>
<tr>
<td>BHR</td>
<td>$h_{br}$</td>
<td>component of distance from reference point to center of buoyancy, positive for center of buoyancy below reference point (see fig. 2)</td>
</tr>
<tr>
<td>BLR</td>
<td>$l_{br}$</td>
<td>component of distance from reference point to center of buoyancy, positive for center of buoyancy forward of reference point (see fig. 2)</td>
</tr>
<tr>
<td>SHR</td>
<td>$h_{sr}$</td>
<td>component of distance from reference point to center of mass of balloon structure, positive for center of mass below reference point (see fig. 2)</td>
</tr>
<tr>
<td>FORTRAN variable name</td>
<td>Mathematical symbol</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------</td>
<td>------------</td>
</tr>
<tr>
<td>SLR</td>
<td>( l_{sr} )</td>
<td>component of distance from reference point to center of mass gravity of balloon structure, positive for center of mass aft of reference point (see fig. 2)</td>
</tr>
<tr>
<td>CGH</td>
<td>( h_{cg} )</td>
<td>component of distance from reference point to center of mass, positive for center of mass below reference point (see fig. 2)</td>
</tr>
<tr>
<td>CGL</td>
<td>( l_{cg} )</td>
<td>component of distance from reference point to center of mass, positive for center of mass forward of reference point (see fig. 2)</td>
</tr>
<tr>
<td>TLR</td>
<td>( l_{tr} )</td>
<td>component of distance from reference point to attachment point of tether line, positive for attachment point forward of reference point (see fig. 2)</td>
</tr>
<tr>
<td>TTR</td>
<td>( t_{tr} )</td>
<td>component of distance from reference point to attachment point of tether line, positive for attachment point below reference point (see fig. 2)</td>
</tr>
<tr>
<td>CLC</td>
<td>( l_c )</td>
<td>length of tether cable</td>
</tr>
<tr>
<td>CDIAM</td>
<td>( d_c )</td>
<td>diameter of tether cable</td>
</tr>
<tr>
<td>CDC</td>
<td>( C_{Dc} )</td>
<td>drag coefficient of tether cable based on diameter, i.e., drag of cable per unit length is ( C_{Dc} d_c \rho V^2/2 )</td>
</tr>
<tr>
<td>WC</td>
<td>( w_c )</td>
<td>weight per unit length of tether cable</td>
</tr>
<tr>
<td>RHO</td>
<td>( \rho )</td>
<td>ambient air density</td>
</tr>
<tr>
<td>VMIN</td>
<td>( V_{min} )</td>
<td>minimum wind velocity</td>
</tr>
<tr>
<td>DELV</td>
<td>( \Delta V )</td>
<td>wind-velocity increment</td>
</tr>
<tr>
<td>NVEL</td>
<td></td>
<td>number of velocity calculations</td>
</tr>
<tr>
<td>COLUMN NUMBER</td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0000000011111111112222222222333333333444444455555555566666666677777777778</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1234567890123456789012345678901234567890123456789012345678901234567890</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

LATERAL STABILITY OF TETHERED BALLOON - LRC BALLOON - REFERENCE CONFIGURATION
$LATDATA VMIN=1., NVEL=51, DELV=1.$
Listing of Lateral Stability Program

OVERLAY(STBLTY2,0,0)
PROGRAM STBLTY2,INPUT=1,OUTPUT=1,TAPE5=INPUT,TAPE7,TAPE8=1,+
 + TAPE11=1,TAPE30=1,TAPE31=1,TAPE32=1,TAPE33=1)
C**********************************************************************
C
C PROGRAM A2864.2 - LATERAL STABILITY OF TETHERED BALLOON
C
C THE FOLLOWING SUBROUTINES ARE CALLED BY THIS PROGRAM, BUT ARE LISTED
C ONLY WITH THE LONGITUDINAL PROGRAM - ROMBERG, QUADET, MATRIX,
C REIG, HESSEN, AND QRT.
C
C PROGRAM REACS IDENTIFICATION CARD AND NAMELIST FROM INPUT FILE, AND
C WRITES ONLY THE ID ARRAY FOR EACH CASE ON THE OUTPUT FILE
C ALL FILES ARE BCD AND ARE SET TO MINIMUM BUFFER SIZE, EXCEPT TAPE7
C WHICH IS BINARY AND USES STANDARD BUFFER SIZE
C FILE ASSIGNMENTS ARE - TAPE7=PLOTTING PROGRAM INPUT, TAPE8=EIGEN-
C VECTORS, TAPE11=EIGENVALUES, TAPE30=AERODYNAMIC COEFFICIENTS,
C TAPE31=TETHER SPRINGS, TAPE32=TETHER CONDITIONS, AND TAPE33=
C UNCOUPLED ROOTS
C
C**********************************************************************
C
CMCOMMON/IR0M/(300)/IC0L/ICGl/(300)
DIMENSION AI3,3I,BI3,3I,CI3,3I,SYMBOLll11,IDI101
+IRUN(I),P(I),INDX(i),INDEX(i)
CMPPLEX EICOEP(3,3),CROT,CRTSQ,CDET
DATA SYMBOL/110nCIRCLE SQUARE DIAMOND TRIANGLE RT TRNGL
+QUADRANT DOG HCUSE LING DMND HOUSE
DATA RADEG,DELV9,NVEL,VMIN/.017453292Sl9943296,..51 51 UNITS
106 FORMAT(lH1111X10A10)
105 FORMAT(14X*EIGENVECTORS.1114X*COMPLEX
+*V/PSI, M/DEG-REAL,IMAG*SX
104 FORMAT(lX,A10,6G16.6)
103 FORMAT(2X,A10,6G16.6)
102 FORMAT(14X*CONDITION NUMBER OF EIGENVALUE MATRIX=*E10.2/)
101 FORMAT(14X*CONDITION NUMBER OF A-MATRIX=*E10.2/)
100 FORMAT(14X*VELOCITY,REAL(RGG=ROOT(I),I=1,NPOS)* SYMBCL,1IMAGE(root
+11,11,NPOS)/)
110 FORMAT(1H1111X10A10) 
A10= 8H(X10A10)
C
C INITIALIZATION SECTION - READ IDENTIFICATION CARD, CALL DAYTIM FOR
C DATE AND TIME, AND WRITE ID ARRAY ON BCD TAPES 8,11,30,31,32,33,
C AND BINARY TAPE 7 WITH RECOUT. DO NON-VELOCITY-DEPENDENT
C CALCULATIONS WITH A CALL TO INICOEF
C SEE SUBROUTINE WRITEUP FOR DESCRIPTION OF RECOUT
C
NMP=3 $ NTNC=NMP$NMP + NPL1=NTWO+1
REWIND 30 $ REWIND 31 $ REWIND 32 $ REWIND 33
REWIND 7 $ REWIND 8 $ REWIND 11
1 READ A10,1IC(1),1=1,8 $ IFIEOF,51999,2
2 CALL DAYTIM(1IC(9)) $ PRINT All,IO $ WRITE(31,108)10 $ WRITE(30,108)IO $ WRITE(31,108)IO $ WRITE(32,108)IO $ WRITE(33,108)IO
CALL RECOUT(7,2,0,1 l,D10,11)
CALL INICOEF(A,B,C,NMP,VEL,VMIN,DELV,NVEL) $ WRITE(11,100)
CALL RECOUT(7,1,0,NVEL)
C
C 90-LCOP IS VELOCITY VARIATION LOOP
C
C DO 90 IV=1,NVEL $ VEL=VMIN+(IV-1.)*DELV
SET UP COEFFICIENT MATRICES FOR QUADRATIC STABILITY DETERMINANT
WITH CALL TO ENTRY VCGEF OF SUBROUTINE INICOEF

CALL VCGEF(A,B,C,NMP,VEL,VMIN,DELV,NVEL)

EXPAND QUADRATIC N X N STABILITY DETERMINANT INTO 2N X 2N STANDARD
EIGENVALUE FORM AND CHECK CONDITIONING OF MASS MATRIX A

CALL QUADET(A,B,C,3,6,NMP,10,EIDET,CNOA)
IFI(CNOA.GT.1.E+4)WRITE(11,101)CNOA

EIGENVALUES FOR 2N SYSTEM AND CHECK CONDITIONING OF 2N X 2N MATRIX
WITH CALL TO MATRIX FOR INVERSE AND TURING CONDITION NUMBER

CALL REIG(EIDET,NTWO,NTWO,0,ROOTR,KOUTI,EVEC,6,INDEX,IRUN,P,+NPIL,SAVE)
CALL MATRIX(10,NTWO,NTWO,0,EIDET,6,DETEI,KB,CNOE)
IFI(CNOE.GT.1.E+6)WRITE(11,102)CNOE

ROOT SORTING - SCRT COMPLEX ROOTS IN ORDER OF INCREASING MAGNITUDE OF
FREQUENCY AND DETERMINE THE NUMBER OF COMPLEX ROOTS WITH POSITIVE
VALUE OF FREQUENCY (IMAGINARY PART)

NEGR=1 $ DO 50 NRT=1,NTWO $ NI=NTWO-NRT $ DO 48 J=1,NI
IFI(RCOT(J-J)-ROOT(J+I))48,46,46
46 TRI=ROOT(J-J) $ TRR=ROOT(J+I) $ ROOT(J-J)=ROOT(J+J+I)
ROOT(J+J)=ROOT(J+J-I) $ ROOT(J+J+I)=TRI $ ROOT(J+J-I)=TRR
48 CONTINUE
50 CONTINUE $ DO 52 NR=1,NTWO
IFI(RCUT(NR)+LT.-1.E-12)NEGR=NEGR+1
52 CONTINUE

WRITE ROOTS ON TAPE 11

IK1=IV/10 $ IK2=11 $ IF(IV.EQ.10*IK1)IK2=1+MOD(IK1-1,10)
WRITE(11,103)VEL,(ROOTR(N),N=NEGR,NTWO)
WRITE(11,104)SYMBOL(IK2),(ROOTI(N),N=NEGR,NTWO)

WRITE RESULTS IN BINARY TAPE 7 FOR INPUT TO PLOTTING PROGRAMS

CALL RECOUT(7,1,0,VEL,IK2,NEGR,NTWO)
CALL RECOUT(7,2,0,ROOTR,NEGR,NTWO,1)
CALL RECOUT(7,2,0,ROOTI,NEGR,NTWO,1)

SETUP COEFFICIENT MATRICES FOR EIGENVECTOR (MODAL RATIOS) BY
CIVICING BY PSI AND CALLING CIXIV - RESULTS ON TAPE 8

WRITE(8,106)VEL,SYMBOL(IK2)
DO 7C NE=NEGR,NTWO $ CROT=CMPLX(KOUT(N),ROOTI(N))
CRTS=CROT*CROT $ DO 60 IC=1,2 $ DO 60 IR=1,3
60 EICOEF(1,IR)=$AIIC,IR)*CRTS+BIIC,IR)*CROT+C(1C,IR)
DO 64 I=1,2
64 EICOEF(I,3)=RAOEG*EICOEF(I,3)
CALL CIXIV(EICOEF,2,EICOEF(1,3),1,COET,IP IV,INDX,3,ISC)
EICOEF(3,3)=(1.,0.)
70 WRITE(8,107)CROT,(EICOEF(I,3),I=1,3)

CONTINUE $ GO TO 1
999 ENDFILE 7 $ REWIND 7 $ ENDFILE 8 $ REWIND 8 $ REWIND 11
REWIND 30 $ REWIND 31 $ REWIND 32 $ REWIND 33
END PROGRAM STBLTV2

63
SUBROUTINE INICOEFS(A,B,C,NMAX,VEL,VMIN,DELV,NVEL)

SUBROUTINE CALCULATES COEFFICIENT MATRICES FOR QUADRATIC STABILITY
DETERMINANT
EQUIVALENCE(EQURT(1),UNCRT(1))
DIMENSION A(NMAX,1),B(INMAX,1),C(NMAX,1),EQURT(1)
COMPLEX UNCRT(6),CRAO,CSQRT

INPUT PARAMETERS ARE READ FROM THE INPUT FILE WITH A NAMELIST READ
OF THE NAMELIST LATDATA AND ARE WRITTEN ON TAPE 11 WITH A NAMELIST
WRITE STATEMENT

NAMESLIST/LATDATA/CDINS,DELCD,DELC,M,DELCM,DELCB,DELCLBD,DELCLP,
+ DELCLR,DELCNB,DELCNN,DELCL,DELCYB,DELCYD,DELCYP,DELCYR,
+ RATICKY,S,CBAR,XXOI,ZZOI,MPASS,AYPASS,WS,ADY,ADR,BLR,SLR,
+ CGH,CGL,TLR,TTR,CLC,CDIAM,CCE,MC,RHO,VMIN,DELV,NVEL
COMMON/LONGDEL/CDINS,DELCD,DELC,M,DELCM,DELCB,DELCLBD,DELCLP

PARAMETERS FOR LRC BALLOON - REFERENCE CONFIGURATION - IN SI UNITS

DATA CDINS,DELCD,DELCL,DELC,M,DELCM,DELCB,DELCLBD,DELCLP,
+ DELCLR,DELCNB,DELCNN,DELCL,DELCYB,DELCYD,DELCYP,DELCYR,
+ RATICKY,S,CBAR,XXOI,ZZOI,MPASS,AYPASS,WS,ADY,ADR,BLR,SLR,
+ CGH,CGL,TLR,TTR,CLC,CDIAM,CCE,MC,RHO,VMIN,DELV,NVEL

DATA DEGRAD/57.2/57.2/57.2/57.2/57.2/57.2,
DATA S,CBAR,XXOI,ZZOI/7.04/7.04/7.04/7.04/7.04/7.04,
DATA TMASS,AYMASS,kTS,BUOY/14.2/14.2/14.2/14.2/14.2/14.2,
DATA BHR,BLR,SLR,TLR,TTR/0.0/0.0/0.0/0.0/0.0/0.0,
DATA ShR,CGH,CGL/0.38/0.38/0.38/0.38/0.38/0.38,

DATA CLC,CDIAM,CCE,MC/61.01/61.01/61.01/61.01/61.01/61.01,
DATA RHO/1.23/1.23/1.23/1.23/1.23/1.23

SUBROUTINE CALCULATES COEFFICIENT MATRICES FOR QUADRATIC STABILITY
DETERMINANT
EQUIVALENCE(EQURT(1),UNCRT(1))
DIMENSION A(NMAX,1),B(INMAX,1),C(NMAX,1),EQURT(1)
COMPLEX UNCRT(6),CRAO,CSQRT

INPUT PARAMETERS ARE READ FROM THE INPUT FILE WITH A NAMELIST READ
OF THE NAMELIST LATDATA AND ARE WRITTEN ON TAPE 11 WITH A NAMELIST
WRITE STATEMENT

NAMESLIST/LATDATA/CDINS,DELCD,DELC,M,DELCM,DELCB,DELCLBD,DELCLP,
+ DELCLR,DELCNB,DELCNN,DELCL,DELCYB,DELCYD,DELCYP,DELCYR,
+ RATICKY,S,CBAR,XXOI,ZZOI,MPASS,AYPASS,WS,ADY,ADR,BLR,SLR,
+ CGH,CGL,TLR,TTR,CLC,CDIAM,CCE,MC,RHO,VMIN,DELV,NVEL
COMMON/LONGDEL/CDINS,DELCD,DELC,M,DELCM,DELCB,DELCLBD,DELCLP

PARAMETERS FOR LRC BALLOON - REFERENCE CONFIGURATION - IN SI UNITS

DATA CDINS,DELCD,DELCL,DELC,M,DELCM,DELCB,DELCLBD,DELCLP,
+ DELCLR,DELCNB,DELCNN,DELCL,DELCYB,DELCYD,DELCYP,DELCYR,
+ RATICKY,S,CBAR,XXOI,ZZOI,MPASS,AYPASS,WS,ADY,ADR,BLR,SLR,
+ CGH,CGL,TLR,TTR,CLC,CDIAM,CCE,MC,RHO,VMIN,DELV,NVEL

DATA DEGRAD/57.2/57.2/57.2/57.2/57.2/57.2,
DATA S,CBAR,XXOI,ZZOI/7.04/7.04/7.04/7.04/7.04/7.04,
DATA TMASS,AYMASS,kTS,BUOY/14.2/14.2/14.2/14.2/14.2/14.2,
DATA BHR,BLR,SLR,TLR,TTR/0.0/0.0/0.0/0.0/0.0/0.0,
DATA ShR,CGH,CGL/0.38/0.38/0.38/0.38/0.38/0.38,

DATA CLC,CDIAM,CCE,MC/61.01/61.01/61.01/61.01/61.01/61.01,
DATA RHO/1.23/1.23/1.23/1.23/1.23/1.23

VELOCITY INDEPENDENT CALCULATIONS

WRITE(30,101) $ WRITE(31,102) $ WRITE(32,103) $ WRITE(33,104)
READ LATDATA $ WRITE(11,1) LATDATA
SL=SLR+CGH $ SL=SHR-CGH $ SL=BLR-CGL
TL=TLR-CGL $ TT=TTR-CGH $ BH=CGH-BHR
CBAR2=.5*CBAR $ ROCS2=RHO*5.5*CBAR2 $ ROCS2=ROCS2
A(1,1)=A(2,2)=A(3,3)=1. $ A(1,2)=A(1,3)=0. $ RETURN

ENTRY POINT VCOEF FOR VELOCITY-DEPENDENT CALCULATIONS
ENTRY VCOEF
Q=.5*RHO*VEL*VEL $ VCON=.5*VEL*ROCS2

TRIM ANGLE OF ATTACK AND AERODYNAMIC COEFFICIENTS ABOUT THE CENTER
OF MASS
CALL TRIM2S,CEB,WS,BUOY,BL,BH,SL,SH,TL,TT,CGH,CGL,Q,
+ CL,CM,CD,CLA,CM,CD,CLC,CD,CUM,ALPHA,XOC,ZOC,SINA,COSA)
ALPHA=DEGRAD*ALPHA

TRANSFER DYNAMIC STABILITY DERIVATIVES FROM REFERENCE POINT (MOMENT
CENTER) TO CENTER OF MASS
CALL DERTRNXOC,ZOC,ALPHA,CLC,CLD,CLB,CLBD,CLP,CLR,CLN,CLNBD,CLNP,CLYB,CLYR,CLYR
+ CYBC,CRY,CRY)
INCREMENT LATERAL STABILITY DERIVATIVES FOR TREND STUDIES AND WRITE
AERODYNAMIC COEFFICIENTS ON TAPE 30

CLB=CLB+DELCLB  CLD=CLD+DELCLD  CLP=CLP+DELCLP
CNB=CNB+DELCNB  CNBD=CNBD+DELCNBD  CNP=CNP+DELCNP
CYB=CYB+DELCYB  CYBD=CYBD+DELCYBD  CYP=CYP+DELCYP
CLR=CLR+DELCJR  CNR=CNR+DELCNR  CWR=CR=+DELCWR
WRITE(30,106) VEL,ALPHAD,CD,CL,CLB,CLR,CNP,CNR,
+ CYB, CYBD, CYP, CYR

CALCULATE EQUILIBRIUM CABLE CONDITIONS AND WRITE RESULTS ON TAPE 32

DRAG=CD*Q*S  BLIFT=CL*Q*S  CD=CD*CDIARQ*Q
Tl=SQR(DRAG+DRAG+(BLIFT-WTS+BUOY)**2)
COSG1=DRAG/Tl  GAM1=ACOS(COSG1)  TSG1=T1*SIN(GAM1)
CALL TETHER(CDAR,WC,CLC,T1,GAM1,TO,GAMD,X1,Z1)
GAM1=DEGRAD*GAM1  GAMD=DEGRAD*GAM0
WRITE(32,100) VEL,X1,Z1,GAM1D,T1,CDPAG

CALCULATE WEIGHT-BOUyANCY MOMENT TERM AND MASS TERMS INCLUDING
APPARENT MASS ROTATION TO STABILITY AXES

HK1=TL*COSA+TT*SINA  HK2=TT*COSA-TL*SINA
HK2=TT*COSA-TL*SINA  HK2=TT*COSA-TL*SINA
SM1=BL*BUOY+SL*WTS-SINA*(BH+BUOY+SH)*SINA
SKYS=SKSS=HK1*SKYS  SM2=BL*BUOY+SL*WTS-SINA*(BH+BUOY+SH)*SINA

CALCULATE CABLE SPRINGS FROM DERIVATIVES OF NEUMARK AND TRANSFER TO
STABILITY AXES - WRITE RESULTS ON TAPE 31

SKYY=RATIOXY*YSLBY(CDRAG,GMGC,GAM1)
SKYS=SKSS=HK1=SKYY  SKYP=-HK2=SKYP  SKSS=HK1=SKYS
WRITE(31,100) VEL,SKYY,SKYP,SKYS,SKPP,SKPS,SKSS

CALCULATE COEFFICIENT MATRICES A , B, AND C

A(2,1)=-RIX*ROSCS4*CLB  A(2,3)=-X1I*RIX
A(3,1)=-RIZ*ROSCS4*CNBD  A(3,2)=-X1I*RIZ
B(1,1)=-VCON*CBAR*RYMT*CYB  B(1,2)=-VCON*RYMT*CLB
B(1,3)=-VCON*RYMT*(CYR-CYR0)  B(1,4)=-VCON*RYMT*CLR*CLBD
B(2,1)=VCON*CBAR*RIX*CLP  B(2,2)=-VCON*CBAR*RIX*(CLR*CLBD)
B(3,1)=-2.*VCGN*RIZ*CNBD  B(3,2)=-VCON*CBAR*RIZ*CNBD
B(3,2)=-VCON*CBAR*RIZ*CNBD  B(3,3)=VCON*CBAR*RIZ*I1,1
C(1,1)=SKKY=RMYT  C(1,2)=RYMT=(SKYD-Q*SL)
C(1,3)=RYMT=(SKYAC*ICBD)  C(1,4)=RIX*SKPY
C(2,2)=RIX*SKPP+HK1*TSGL*SM1
C(2,3)=RIX*SKPP+HK1*TSGL*SM1  C(3,1)=SKYD=RIX
C(3,2)=RIX*SKPP+HK1*TSGL*SM2  C(3,3)=SKY=Q*S*(CBAR*CNBD+HK1*CD1)

CALCULATE UNCOUPLED ROOTS BY FACTORING DIAGONAL QUADRATIC TERMS AND
WRITE RESULTS ON TAPE 33

DO 1 M=1,3  CRAD=.25*BIM,M+3(M,3)-CIM,M
CRAD=SQRT(CRAD)
M2=2*M  M1=M2-1  UNCFT(M1)=-.5*BIM,M+CRAD
1 UNCFT(M2)=-.5*BIM,M-CRAD
WRITE(33,105) VEL,EQURT(11),11,11,2, Equrt(11),12,12,2  RETURN
format(2/11g11.4)  
format(2/1x,g12.4)  
format(2/1x,g12.4)  
format(2/1x,g12.4)  
format(2/1x,g12.4)

end subroutine inicoef

subroutine trimis, cbar, wts, buoy, bl, bh, sl, sh, tl, tt, cg, cl, cm, q, cl, cm, cd, cla, cma, cda, alpha, f, g, sa, ca

C SUBROUTINE COMPUTES THE STATIC TRIM ANGLE-OF-ATTACK ALPHA USING NEWTON ITERATION OF THE TRIM EQUATION. THE ALPHA DEPENDENT DERIVATIVES CD, CDA, CL, CLA, CM, AND CMA ARE ALSO TRANSFERRED TO THE CENTER OF MASS AND RETURNED IF CONVERGENCE IS NOT OBTAINED IN ITMAX ITERATIONS, MESSAGE IS WRITTEN ON TAPE 11

C 

cbu=cm/longcl/cclins, delcd, delcl, delcm

err=1.e-6  $ talpha=.05  $ o=0.5  $ itcmax=100  $ itc=d

alpha=tlpha*05  $ o=s=0.5  $ itcmax=100  $ itc=d

do=buoy*bl+wts*sl  $ e=buoy*bl+wts*sh

if alpha=talpha-talpha > err11,4

write(11,311talpha,alphatalpha=guide)

1 return

end subroutine trim
FUNCTION FCCIA,CDA)
   C  
   CFCD IS A FUNCTION TO BE WRITTEN BY THE USER TO CALCULATE CD AND CDA.
   C AS FUNCTIONS OF ANGLE OF ATTACK FOR THE CONFIGURATION TO BE ANALYZED.
   C  
   C CURVE FIT OF CD AND CDA VS ANGLE OF ATTACK IN RADIANS FOR LRC
   C BALLCON - REFERENCE CONFIGURATION
   C  
   X=A-.023 $ X5=X**5 $ CDA=1117.2*X5
   FCD=.047418+2*X*X $ RETURN $ END

FUNCTION FCLIA,CLA)
   C  
   C FCL IS A FUNCTION TO BE WRITTEN BY THE USER TO CALCULATE CL AND CLA
   C AS FUNCTIONS OF ANGLE OF ATTACK FOR THE CONFIGURATION TO BE ANALYZED
   C  
   C CURVE FIT CF CL AND CLA VS ANGLE OF ATTACK IN RADIANS FOR LRC
   C BALLCON - REFERENCE CONFIGURATION
   C  
   X=A+.23 $ X2=X*X $ CLA=.82-X2*(15.069-557.0*X2)
   FCL=X*1.82-X2*(15.023-111.4*X2) $ RETURN $ END

FUNCTION FCMR(A,CMAR)
   C  
   CFCMR IS A FUNCTION TO BE WRITTEN BY THE USER TO CALCULATE CMR AND CMAR.
   C AS FUNCTIONS OF ANGLE OF ATTACK FOR THE CONFIGURATION TO BE ANALYZED
   C  
   C CURVE FIT OF CM AND CMAR ABOUT REFERENCE POINT VS ANGLE OF ATTACK IN
   C RADIANS FOR LRC BALLON - REFERENCE CONFIGURATION
   C  
   CMAR=.1435 $ FCMR=-.0106+.1435*A $ RETURN $ END

FUNCTION FCLBO(A)
   C  
   CFUNCTIONS FCLBO, ETC, ARE FUNCTIONS TO BE WRITTEN BY THE USER TO
   C CALCULATE THE LATERAL STABILITY DERIVATIVES AS FUNCTIONS OF ANGLE
   C OF ATTACK FOR THE CONFIGURATION TO BE ANALYZED. FUNCTIONS ARE
   C REFERENCED ONLY BY SUBROUTINE DERTRN.
   C  
   CFUNCTIONS FCLBO, ETC, GIVE THE LATERAL STABILITY DERIVATIVES VS ANGLE
   C OF ATTACK IN RADIANS FOR THE LRC BALLOON - REFERENCE CONFIGURATION.
   C  
   FCLBO=-.1435*SIN(A) $ RETURN $ END

FUNCTION FCLBCO(A)
   FCLBO=0. $ RETURN $ END

FUNCTION FCLPO(A)
   FCLPO=-.0237 $ RETURN $ END

FUNCTION FCLRO(A)
   FCLRO=-.175*SIN(A) $ RETURN $ END
FUNCTION FCNB(l, A)
FCNB = -0.1435 $ RETURN $ END

FUNCTION FCNBDO(l, A)
FCNBDO = 0.26 $ RETURN $ END

FUNCTION FCNBPO(l, A)
FCNBPO = -0.0641 * SIN(2 * A) $ RETURN $ END

FUNCTION FCNRO(l, A)
FCNRO = -0.189 $ RETURN $ END

FUNCTION FCYB(l, A)
FCYB = -0.82 $ RETURN $ END

FUNCTION FCYBDO(l, A)
FCYBDO = -0.089 $ RETURN $ END

FUNCTION FCYBPO(l, A)
FCYBPO = -0.494 * SIN(A) $ RETURN $ END

FUNCTION FCYRC(l, A)
FCYRC = 0.685 $ RETURN $ END

SUBROUTINE DERTRN(x, z, a, clb, clbdo, clp, clr, cnb, cnbdo, cnbpo, cnr, cyb, + cybdo, cyr, cybpo)
C TRANSFERS ALL 12 LATERAL DERIVATIVES FROM REFERENCE POINT (MOMENT CENTER) TO CENTER OF MASS LOCATED X/C FORWARD AND Z/C DOWN FROM REFERENCE POINT
C
C CYB = FCYB(l, A) $ CYBDO = FCYBDO(l, A) $ CYBPO = FCYBPO(l, A) $ CYRO = FCYRO(l, A)
C CYP = CYB * Z $ CYR = CYRO - Z * CYB $ CNB = FCNB(l, A) - X * CYB
C CNBDO = FCNBDO(l, A) - X * CYBDO $ CNBPO = FCNBPO(l, A) - X * CYBPO + Z * CNB
C CNR = FCNRO(l, A) - X * (CYRO + Z * CNB) $ CLB = FCLBDO(l, A) + Z * CYB
C CLBDO = FCLBDO(l, A) + Z * CYBDO $ CLP = FCLBPO(l, A) + Z * (CYBPO + 2 * CLB)
C CLR = FCLRRC(l, A) + Z * CYRO - 2 * X * CLB $ RETURN
END SUBROUTINE DERTRN
SUBROUTINE TETHER(CDRAG, WC, CL1, T1, GAMMA1, TO, GAMMAO, X1, Z1)

C SUBROUTINE IS BASED ON THE ANALYSIS IN -
C NEUMARK, S.-EQUILIBRIUM CONFIGURATIONS OF FLYING CABLES OF
C CAPTIVE BALLOONS, AND CABLE DERIVATIVES FOR STABILITY
C THE TETHER PARAMETERS CDRAG, WC, CL1, T1, AND GAMMA1 ARE INPUT
C THE TETHER PARAMETERS TO, GAMMAO, X1, AND Z1 ARE OUTPUT

C EXTERNAL FLAM, FSIG, $ COMMON/PC/P, Q
P=5*WC/CDRAG $ C=SQR(T(1+P*P)) $ EPS=1.E-8
CALL RCMBERG(RAM1, 0., GAMMA1, FLAM, EPS) $ TAUl=TAU(GAMMA1)
RAMO=RAM1-CDRAG*TAU1*CL1/T1 $ CALL NEWINT(RAMO, FLAM, 1., GAMMAO)
CALL RCMBERG(DSIG, GAMMAO, GAMMA1, FSIG, EPS)
X1=T1*DSIG/ICCRAG*TAU1) $ TAUQ=TAU(GAMMAO)
TO=T1*TAUO/TAU1 $ Z1=(T1-TO)/WC $ RETURN $ END

FUNCTION FLAMITI $ COMMON/PC/P, Q $ CT=COSIT
FLAM=I(QP-CT)/(Q-P-CT)**(P/Q)/(1-CT*CT+2*P*CT) $ RETURN $ END

FUNCTION FSIGITI $ COMMON/PC/P, Q $ CT=COSIT
FSIG=I(QP-CT)/(Q-P-CT)**(P/Q)/(1-CT*CT+2*P*CT)*CT $ RETURN $ END

FUNCTION TAUITI $ COMMON/PC/P, Q $ CT=COSIT
TAU=I(QP-CT)/(Q-P-CT)**(P/Q) $ RETURN $ END

FUNCTION YSUBY(CDRAG, GAMMAO, GAMMA1) $ COMMON/PC/P, Q
EXTERNAL FTHE $ EPS=1.E-8 $ CT=COS(GAMMA1)
CALL RCMBERG(DTHE, GAMMAO, GAMMA1, FTHE, EPS)
YSUBY=CDRAG*SQR(TAU(GAMMA1)*(1-CT*CT+2*P*CT))/DTHE $ RETURN $ END

FUNCTION FTHEITI $ COMMON/PC/P, Q $ CT=COSIT
FTHE=SQR(T((QP-CT)/(Q-P-CT)**(P/Q)/(1-CT*CT+2*P*CT)) $ RETURN
END SUBPROGRAM FTHE

SUBROUTINE NEWINT(FC, XO, XI) $ EPS=1.E-8 $ XT=XO $ I=0
C SUBROUTINE COMPUTES THE UPPER LIMIT X OF THE DEFINITE INTEGRAL
C FROM 0 TO X OF THE FUNCTION F FOR WHICH THE VALUE OF THE
C INTEGRAL C IS KNOWN
C NEWTON ITERATION IS USED WITH THE VALUE OF THE INTEGRAL DETERMINED
C BY SUBROUTINE RCMBERG

I=XT $ CALL RCMBERG(S, 0., X, F, EPS) $ XT=X+(C-S)/F(X) $ I=I+1
PI2=1.570796326 $ IF(XT.GT.PI2) XT=PI2 $ IF(XT.LT.0.) XT=0.
IF(I.LT.2) GO TO 1 $ IF(ABS1X-XT1.GT.EPS) GO TO 1
X=XT $ RETURN $ END

69
Printing of Files Containing Calculated Results
for Sample Case

A printing of each BCD file for the sample case is given. The headings essentially give the quantities printed using the FORTRAN variable names previously given. The lateral program uses the same files as does the longitudinal program. The information written on tapes 30 and 32 by the lateral program is identical to that written by the longitudinal program. Tapes 8, 11, 31, and 33 are also closely paralleled to the corresponding longitudinal files. Note that the headings PSI and PHI are used to denote $\psi$ and $\phi$. Also, the letters P, S, and Y are used to denote $\phi$, $\psi$, and $y$ in the headings on tapes 31 and 33.
Listing of OUTPUT for sample case (identification card, and date and time only).

LATERAL STABILITY OF TETHERED BALLOON - LCT BALLON - REFERENCE CONFIGURATION 11/17/72 00.33.33.

Listing of tape 11 for sample case (principal file for characteristic roots).

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDLYS</td>
<td>0.1E-01</td>
</tr>
<tr>
<td>CRCD</td>
<td>0.0</td>
</tr>
<tr>
<td>CRCL</td>
<td>0.3</td>
</tr>
<tr>
<td>CRCH</td>
<td>0.0</td>
</tr>
<tr>
<td>CRCP</td>
<td>0.0</td>
</tr>
<tr>
<td>CRCLL</td>
<td>0.0</td>
</tr>
<tr>
<td>CRCC</td>
<td>0.0</td>
</tr>
<tr>
<td>CRCLB</td>
<td>0.0</td>
</tr>
<tr>
<td>CRCLC</td>
<td>0.0</td>
</tr>
<tr>
<td>CRCLD</td>
<td>0.0</td>
</tr>
<tr>
<td>CRCLV</td>
<td>0.0</td>
</tr>
<tr>
<td>CRCLRO</td>
<td>0.0</td>
</tr>
<tr>
<td>CRCLRP</td>
<td>0.0</td>
</tr>
<tr>
<td>CRCLYM</td>
<td>0.0</td>
</tr>
<tr>
<td>CRCLYR</td>
<td>0.0</td>
</tr>
<tr>
<td>YATDSY</td>
<td>0.1E+01</td>
</tr>
<tr>
<td>S</td>
<td>0.704E+01</td>
</tr>
<tr>
<td>CRAS</td>
<td>0.764E+01</td>
</tr>
<tr>
<td>XXAS</td>
<td>0.1E+00</td>
</tr>
<tr>
<td>LCXI</td>
<td>0.1E+00</td>
</tr>
<tr>
<td>TWAS</td>
<td>0.1E+00</td>
</tr>
<tr>
<td>AMAS</td>
<td>0.235E+00</td>
</tr>
<tr>
<td>WTS</td>
<td>0.108E+00</td>
</tr>
<tr>
<td>BUOY</td>
<td>0.151E+00</td>
</tr>
<tr>
<td>BMP</td>
<td>0.0</td>
</tr>
<tr>
<td>DLX</td>
<td>0.219E+01</td>
</tr>
<tr>
<td>SMX</td>
<td>0.386E+00</td>
</tr>
<tr>
<td>SLX</td>
<td>0.608E+00</td>
</tr>
<tr>
<td>CGH</td>
<td>0.109E+00</td>
</tr>
<tr>
<td>CGL</td>
<td>0.118E+00</td>
</tr>
<tr>
<td>TLR</td>
<td>0.344E+00</td>
</tr>
<tr>
<td>TRA</td>
<td>0.362E+00</td>
</tr>
<tr>
<td>CLC</td>
<td>0.612E+00</td>
</tr>
<tr>
<td>CDOM</td>
<td>0.141E+00</td>
</tr>
<tr>
<td>CMC</td>
<td>0.117E+00</td>
</tr>
<tr>
<td>WC</td>
<td>0.343E+00</td>
</tr>
<tr>
<td>WMC</td>
<td>0.322E+00</td>
</tr>
<tr>
<td>VRM</td>
<td>0.1E+00</td>
</tr>
<tr>
<td>DLY</td>
<td>0.1E+00</td>
</tr>
<tr>
<td>NVEL</td>
<td>51</td>
</tr>
<tr>
<td>KIND</td>
<td></td>
</tr>
</tbody>
</table>

71
<table>
<thead>
<tr>
<th>VELOCITY: REAL(ROO1(I)); I=1, NPOS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYMBOL: I=1, NPOS)</td>
</tr>
<tr>
<td>-------------------------------------</td>
</tr>
<tr>
<td>1.00000</td>
</tr>
<tr>
<td>2.00000</td>
</tr>
<tr>
<td>3.00000</td>
</tr>
<tr>
<td>5.00000</td>
</tr>
<tr>
<td>6.00000</td>
</tr>
<tr>
<td>7.00000</td>
</tr>
<tr>
<td>8.00000</td>
</tr>
<tr>
<td>9.00000</td>
</tr>
<tr>
<td>10.0000</td>
</tr>
<tr>
<td>11.0000</td>
</tr>
<tr>
<td>12.0000</td>
</tr>
<tr>
<td>13.0000</td>
</tr>
<tr>
<td>14.0000</td>
</tr>
<tr>
<td>16.0000</td>
</tr>
<tr>
<td>17.0000</td>
</tr>
<tr>
<td>18.0000</td>
</tr>
<tr>
<td>19.0000</td>
</tr>
<tr>
<td>20.0000</td>
</tr>
<tr>
<td>21.0000</td>
</tr>
<tr>
<td>22.0000</td>
</tr>
<tr>
<td>23.0000</td>
</tr>
<tr>
<td>24.0000</td>
</tr>
<tr>
<td>25.0000</td>
</tr>
<tr>
<td>No.</td>
</tr>
<tr>
<td>-----</td>
</tr>
<tr>
<td>26</td>
</tr>
<tr>
<td>27</td>
</tr>
<tr>
<td>28</td>
</tr>
<tr>
<td>29</td>
</tr>
<tr>
<td>30</td>
</tr>
<tr>
<td>31</td>
</tr>
<tr>
<td>32</td>
</tr>
<tr>
<td>33</td>
</tr>
<tr>
<td>34</td>
</tr>
<tr>
<td>35</td>
</tr>
<tr>
<td>36</td>
</tr>
<tr>
<td>37</td>
</tr>
<tr>
<td>38</td>
</tr>
<tr>
<td>39</td>
</tr>
<tr>
<td>40</td>
</tr>
<tr>
<td>41</td>
</tr>
<tr>
<td>42</td>
</tr>
<tr>
<td>43</td>
</tr>
<tr>
<td>44</td>
</tr>
<tr>
<td>45</td>
</tr>
<tr>
<td>46</td>
</tr>
<tr>
<td>47</td>
</tr>
<tr>
<td>48</td>
</tr>
<tr>
<td>49</td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>51</td>
</tr>
</tbody>
</table>

**DIAMOND**

26. COO  | -1.56885 | -0.46325 | -1.01715 | -2.43158 | 8.37510 |

27. COO  | -1.62394 | -0.71444 | -1.03540 | -2.52777 | 8.58932 |

28. COO  | -1.68191 | -0.96642 | -1.05317 | -2.62417 | 8.79954 |

29. COO  | -1.73056 | -1.21798 | -1.07354 | -2.72057 | 9.01798 |

30. COO  | -1.79527 | -1.46953 | -1.09361 | -2.81696 | 9.23866 |

31. COO  | -1.85204 | -1.72106 | -1.11395 | -2.91335 | 9.46142 |

32. COO  | -1.90888 | -1.97257 | -1.13446 | -3.00972 | 9.68613 |

33. COO  | -1.96778 | -2.22406 | -1.15500 | -3.10608 | 9.91265 |

34. COO  | -2.02272 | -2.47552 | -1.17565 | -3.20243 | 10.1499 |

35. COO  | -2.07973 | -2.72696 | -1.19641 | -3.30896 | 10.3705 |

36. COO  | -2.13677 | -2.97338 | -1.21735 | -3.40552 | 10.5919 |

37. COO  | -2.19378 | -3.22997 | -1.23847 | -3.50217 | 10.8135 |

38. COO  | -2.25082 | -3.48115 | -1.25974 | -3.60003 | 11.0361 |


40. COO  | -2.36549 | -3.98382 | -1.30256 | -3.80290 | 11.4849 |

41. COO  | -2.42265 | -4.23512 | -1.32416 | -3.90596 | 11.7117 |

42. COO  | -2.47994 | -4.48646 | -1.34602 | -4.00917 | 11.9393 |

43. COO  | -2.53726 | -4.73777 | -1.36817 | -4.11254 | 12.1678 |

44. COO  | -2.59461 | -5.03809 | -1.39069 | -4.21687 | 12.3973 |

45. COO  | -2.65199 | -5.33842 | -1.41350 | -4.32130 | 12.6277 |

46. COO  | -2.70940 | -5.63875 | -1.43645 | -4.42674 | 12.8591 |

47. COO  | -2.76683 | -5.93909 | -1.45960 | -4.53231 | 13.0910 |

48. COO  | -2.82429 | -6.23944 | -1.48290 | -4.63885 | 13.3240 |

49. COO  | -2.88177 | -6.53983 | -1.50645 | -4.74554 | 13.5579 |

50. COO  | -2.93926 | -6.84022 | -1.53029 | -4.85231 | 13.7932 |

51. COO  | -2.99680 | -7.14060 | -1.55440 | -4.95916 | 14.0296 |
<table>
<thead>
<tr>
<th>VEL CITY</th>
<th>1.0000</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.12518</td>
<td>6.89261E-02 -3.3062E-02 2.39396E-03 1.39689E-04 4.22723E-05 1.0000 0</td>
</tr>
<tr>
<td>-2.0134E-02 0.2659 1.2634E-02 -2.0749E-02 7.7967E-04 -6.7554E-04 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>-8.4267E-02 4.7841 6.0391E-04 -6.1516E-04 -1.1250 1.6837E-03 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>VELOCITY</td>
<td>2.0000</td>
</tr>
<tr>
<td>-7.6065</td>
<td>0. 8.8602E-03 0. 8.4694E-04 0. 1.0000 0</td>
</tr>
<tr>
<td>-3.7439</td>
<td>0. -1.1113E-02 0. 6.298E-04 0. 1.0000 0</td>
</tr>
<tr>
<td>-5.8667E-03 2.5723 1.2497E-03 -9.9529E-02 7.6120E-04 -1.1912E-03 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>-1.1648 4.8173 6.4392E-04 -1.4925E-03 -1.2531 3.9962E-03 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>VELOCITY</td>
<td>3.0000</td>
</tr>
<tr>
<td>-3.6081</td>
<td>0. 0. -1.176 0. 2.5667E-03 0. 1.0000 0</td>
</tr>
<tr>
<td>-1.0604</td>
<td>0. -1.0031E-02 0. -6.0913E-04 0. 1.0000 0</td>
</tr>
<tr>
<td>-2.4768E-02 0.2697 1.1034E-02 -1.5539 8.5946E-04 -1.7714E-03 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>-1.2576 4.8684 7.0220E-04 -1.9173E-03 -1.2891 6.0435E-03 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>VELOCITY</td>
<td>4.0000</td>
</tr>
<tr>
<td>-4.3606</td>
<td>0. 0. 2.655 0. 6.3978E-03 0. 1.0000 0</td>
</tr>
<tr>
<td>-2.0485E-02 0.2563E-02 -2.4150E-02 -2.0809 9.5285E-04 -2.4002E-03 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>-3.4558 4.5997 7.6273E-04 -2.0901E-03 -1.1259 5.9989E-03 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>VELOCITY</td>
<td>5.0000</td>
</tr>
<tr>
<td>-6.5439</td>
<td>0. 0. 1.2058 0. 1.8978E-03 0. 1.0000 0</td>
</tr>
<tr>
<td>-9.1984E-03 2.9378 3.8241E-02 -2.5873 1.6125E-03 -3.0329E-03 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>-4.6387 5.0105 9.1079E-04 -3.3426E-03 -1.3085 1.6116E-02 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>VELOCITY</td>
<td>6.0000</td>
</tr>
<tr>
<td>-4.773</td>
<td>0. 0. -1.9584 0. -2.9476 0. 1.0000 0</td>
</tr>
<tr>
<td>-1.4438</td>
<td>0. -9.6244E-03 0. 1.8867E-03 0. 1.0000 0</td>
</tr>
<tr>
<td>-1.6813E-02 3.0418 3.6727E-02 -3.0684 1.3542E-03 -3.6035E-03 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>-5.5278 5.0074 8.3551E-04 -4.1039E-03 -1.3400 1.2055E-02 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>VELOCITY</td>
<td>7.0000</td>
</tr>
<tr>
<td>-3.5288</td>
<td>0. 0. -1.8365 0. -2.9872E-02 0. 1.0000 0</td>
</tr>
<tr>
<td>-1.6496</td>
<td>0. -9.5221E-03 0. 2.0549E-03 0. 1.0000 0</td>
</tr>
<tr>
<td>-2.4826E-02 0.3139 6.6773E-02 -3.5321 1.5671E-03 -4.8382E-03 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>-6.4555 5.1935 8.8289E-04 -4.8831E-03 -1.3427 1.5381E-02 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>VELOCITY</td>
<td>8.0000</td>
</tr>
<tr>
<td>-5.7411</td>
<td>0. 0. -1.2025 0. -1.6999E-02 0. 1.0000 0</td>
</tr>
<tr>
<td>-1.9450</td>
<td>0. -9.4676E-03 0. 2.4456E-03 0. 1.0000 0</td>
</tr>
<tr>
<td>-3.2786E-02 0.32315 8.8384E-02 -3.9517 1.7680E-03 -9.0545E-03 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>-7.1247 5.2986 7.6699E-04 -5.6838E-03 -1.4443 1.5423E-02 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>VELOCITY</td>
<td>9.0000</td>
</tr>
<tr>
<td>-6.2756</td>
<td>0. 0. -0.9793 0. -1.3657E-02 0. 1.0000 0</td>
</tr>
<tr>
<td>-2.1993</td>
<td>0. -9.4261E-03 0. 2.3849E-03 0. 1.0000 0</td>
</tr>
<tr>
<td>-4.0172E-02 0.3209 9.1062E-02 -3.3561 2.0360E-03 -5.7630E-03 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>-6.8092 5.4126 7.0823E-04 -6.4474E-03 -1.5406 1.6840E-02 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>VELOCITY</td>
<td>10.000 CIRCLE</td>
</tr>
<tr>
<td>-6.4131</td>
<td>0. 0. -8.8639 0. -1.2023E-02 0. 1.0000 0</td>
</tr>
<tr>
<td>-1.4431</td>
<td>0. 9.9777E-03 0. 3.2591E-03 0. 1.0000 0</td>
</tr>
<tr>
<td>-9.7081E-02 0.34077 -1.1166 -4.7394 2.2433E-03 -6.5695E-03 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>-8.9981 5.5351 5.9419E-04 -7.2989E-03 -1.3452 1.6055E-02 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>VELOCITY</td>
<td>11.000</td>
</tr>
<tr>
<td>-7.3557</td>
<td>0. 0. -7.9835 0. -1.1094E-02 0. 1.0000 0</td>
</tr>
<tr>
<td>-2.6767</td>
<td>0. 9.2776E-03 0. 3.6661E-03 0. 1.0000 0</td>
</tr>
<tr>
<td>-3.3907E-02 0.3415 -1.9130 -5.1930 2.4676E-03 -7.3340E-03 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>-9.9408 5.6660 4.8806E-04 -7.9495E-03 -1.1524 1.9110E-02 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>VELOCITY</td>
<td>12.000</td>
</tr>
<tr>
<td>-7.8971</td>
<td>0. 0. -7.9573 0. -1.0519E-02 0. 1.0000 0</td>
</tr>
<tr>
<td>-2.9455</td>
<td>0. 9.3632E-03 0. 6.0616E-03 0. 1.0000 0</td>
</tr>
<tr>
<td>-6.9103E-02 0.35722 -1.3445 -5.4556 2.2442E-03 -6.2530E-03 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>-1.0887 5.8050 2.7434E-03 -8.4499E-03 -1.1475 1.5990E-02 1.0000 0</td>
<td></td>
</tr>
<tr>
<td>VELOCITY</td>
<td>13.000</td>
</tr>
<tr>
<td>----------</td>
<td>-------</td>
</tr>
<tr>
<td>VELOCITY</td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>-------</td>
</tr>
<tr>
<td>26.000</td>
<td>1.5689</td>
</tr>
<tr>
<td>27.000</td>
<td>1.6253</td>
</tr>
<tr>
<td>28.000</td>
<td>1.6619</td>
</tr>
<tr>
<td>29.000</td>
<td>1.7336</td>
</tr>
<tr>
<td>30.000</td>
<td>1.8393</td>
</tr>
<tr>
<td>31.000</td>
<td>1.9520</td>
</tr>
<tr>
<td>32.000</td>
<td>2.0227</td>
</tr>
<tr>
<td>33.000</td>
<td>2.1079</td>
</tr>
<tr>
<td>34.000</td>
<td>2.1939</td>
</tr>
<tr>
<td>35.000</td>
<td>2.2799</td>
</tr>
<tr>
<td>36.000</td>
<td>2.3666</td>
</tr>
<tr>
<td>37.000</td>
<td>2.4536</td>
</tr>
<tr>
<td>38.000</td>
<td>2.5415</td>
</tr>
</tbody>
</table>

**Note:** The table above lists the velocity values with corresponding coordinates. Each line represents a set of coordinates, with the first two values indicating the x and y coordinates, respectively.
<table>
<thead>
<tr>
<th>VELOCITY</th>
<th>TRIANGLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>39.000</td>
<td>0.000</td>
</tr>
<tr>
<td>40.000</td>
<td>0.000</td>
</tr>
<tr>
<td>41.000</td>
<td>0.000</td>
</tr>
<tr>
<td>42.000</td>
<td>0.000</td>
</tr>
<tr>
<td>43.000</td>
<td>0.000</td>
</tr>
<tr>
<td>44.000</td>
<td>0.000</td>
</tr>
<tr>
<td>45.000</td>
<td>0.000</td>
</tr>
<tr>
<td>46.000</td>
<td>0.000</td>
</tr>
<tr>
<td>47.000</td>
<td>0.000</td>
</tr>
<tr>
<td>48.000</td>
<td>0.000</td>
</tr>
<tr>
<td>49.000</td>
<td>0.000</td>
</tr>
<tr>
<td>50.000</td>
<td>0.000</td>
</tr>
<tr>
<td>51.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>

### Details

- **Velocity Values:**
  - 39.000
  - 40.000
  - 41.000
  - 42.000
  - 43.000
  - 44.000
  - 45.000
  - 46.000
  - 47.000
  - 48.000
  - 49.000
  - 50.000
  - 51.000

- **Triangle Values:**
  - 0.000

### Additional Information

- The data contains various numerical values with columns likely representing different parameters or measurements.
**Listing of tape 30 for sample case (trim angle of attack and aerodynamic coefficients).**

LATERAL STABILITY OF TERTIARY BALLOON - LRC BALLOON - REFERENCE CONFIGURATION 11/17/72 00.33.53.

### Aerodynamic Coefficients

<table>
<thead>
<tr>
<th>Velocity</th>
<th>Alpha</th>
<th>CD</th>
<th>CL</th>
<th>CMB</th>
<th>CLB</th>
<th>CLD</th>
<th>CLP</th>
<th>CLR</th>
<th>CNB</th>
<th>CNBD</th>
<th>CNP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000</td>
<td>8.767</td>
<td>5.959E-02</td>
<td>9.706E-02</td>
<td>1.5439E-02</td>
<td>6.9808E-04</td>
<td>2.4048E-02</td>
<td>2.8041E-02</td>
<td>2.5033E-02</td>
<td>3.885E-02</td>
<td>2.9795E-02</td>
<td>.9219</td>
</tr>
<tr>
<td>2.000</td>
<td>8.550</td>
<td>5.945E-02</td>
<td>9.6972E-02</td>
<td>1.5351E-02</td>
<td>6.9475E-04</td>
<td>2.4126E-02</td>
<td>2.7026E-02</td>
<td>2.5010E-02</td>
<td>3.8861E-02</td>
<td>2.9096E-02</td>
<td>.9220</td>
</tr>
<tr>
<td>3.000</td>
<td>8.250</td>
<td>5.928E-02</td>
<td>9.3205E-02</td>
<td>1.5228E-02</td>
<td>5.9215E-04</td>
<td>2.3964E-02</td>
<td>2.5621E-02</td>
<td>2.4980E-02</td>
<td>3.8864E-02</td>
<td>2.8125E-02</td>
<td>.9220</td>
</tr>
<tr>
<td>4.000</td>
<td>7.923</td>
<td>5.913E-02</td>
<td>9.106E-02</td>
<td>1.5094E-02</td>
<td>5.0871E-04</td>
<td>2.3917E-02</td>
<td>2.4037E-02</td>
<td>2.4952E-02</td>
<td>3.8867E-02</td>
<td>2.7062E-02</td>
<td>.9221</td>
</tr>
<tr>
<td>5.000</td>
<td>7.609</td>
<td>5.902E-02</td>
<td>8.5161E-02</td>
<td>1.4946E-02</td>
<td>4.3807E-04</td>
<td>2.3875E-02</td>
<td>2.2612E-02</td>
<td>2.4928E-02</td>
<td>3.8869E-02</td>
<td>2.6037E-02</td>
<td>.9221</td>
</tr>
<tr>
<td>6.000</td>
<td>7.327</td>
<td>5.894E-02</td>
<td>8.1624E-02</td>
<td>1.4848E-02</td>
<td>3.7488E-04</td>
<td>2.3860E-02</td>
<td>2.1292E-02</td>
<td>2.4909E-02</td>
<td>3.8871E-02</td>
<td>2.5119E-02</td>
<td>.9221</td>
</tr>
<tr>
<td>7.000</td>
<td>7.086</td>
<td>5.889E-02</td>
<td>7.8599E-02</td>
<td>1.474E-02</td>
<td>3.2067E-04</td>
<td>2.3813E-02</td>
<td>2.0160E-02</td>
<td>2.4896E-02</td>
<td>3.8873E-02</td>
<td>2.4329E-02</td>
<td>.9221</td>
</tr>
<tr>
<td>8.000</td>
<td>6.843</td>
<td>5.885E-02</td>
<td>7.6016E-02</td>
<td>1.466E-02</td>
<td>2.7522E-04</td>
<td>2.3792E-02</td>
<td>1.9210E-02</td>
<td>2.4886E-02</td>
<td>3.8874E-02</td>
<td>2.3665E-02</td>
<td>.9221</td>
</tr>
<tr>
<td>10.00</td>
<td>6.577</td>
<td>5.881E-02</td>
<td>7.2110E-02</td>
<td>1.4536E-02</td>
<td>2.0632E-04</td>
<td>2.376E-02</td>
<td>1.7771E-02</td>
<td>2.4874E-02</td>
<td>3.8875E-02</td>
<td>2.2658E-02</td>
<td>.9221</td>
</tr>
<tr>
<td>11.00</td>
<td>6.463</td>
<td>5.879E-02</td>
<td>7.0640E-02</td>
<td>1.4488E-02</td>
<td>1.8051E-04</td>
<td>2.3754E-02</td>
<td>1.7231E-02</td>
<td>2.4871E-02</td>
<td>3.8876E-02</td>
<td>2.2280E-02</td>
<td>.9221</td>
</tr>
<tr>
<td>12.00</td>
<td>6.367</td>
<td>5.878E-02</td>
<td>6.9415E-02</td>
<td>1.4448E-02</td>
<td>1.5907E-04</td>
<td>2.3746E-02</td>
<td>1.6783E-02</td>
<td>2.4868E-02</td>
<td>3.8876E-02</td>
<td>2.1965E-02</td>
<td>.9221</td>
</tr>
<tr>
<td>13.00</td>
<td>6.287</td>
<td>5.877E-02</td>
<td>6.8389E-02</td>
<td>1.4414E-02</td>
<td>1.4116E-04</td>
<td>2.3740E-02</td>
<td>1.6409E-02</td>
<td>2.4866E-02</td>
<td>3.8876E-02</td>
<td>2.1703E-02</td>
<td>.9221</td>
</tr>
<tr>
<td>14.00</td>
<td>6.220</td>
<td>5.877E-02</td>
<td>6.7526E-02</td>
<td>1.4386E-02</td>
<td>1.2612E-04</td>
<td>2.3735E-02</td>
<td>1.6094E-02</td>
<td>2.4865E-02</td>
<td>3.8876E-02</td>
<td>2.1482E-02</td>
<td>.9221</td>
</tr>
<tr>
<td>Time</td>
<td>Value 1</td>
<td>Value 2</td>
<td>Value 3</td>
<td>Value 4</td>
<td>Value 5</td>
<td>Value 6</td>
<td>Value 7</td>
<td>Value 8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15.00</td>
<td>6.164</td>
<td>5.876E-02</td>
<td>6.675E-02</td>
<td>1.4362E-02</td>
<td>1.1340E-04</td>
<td>2.3731E-02</td>
<td>1.5828E-02</td>
<td>3.876E-02</td>
<td>2.1295E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16.00</td>
<td>5.876E-02</td>
<td>6.617E-02</td>
<td>1.4342E-02</td>
<td>1.0257E-04</td>
<td>2.3728E-02</td>
<td>1.5602E-02</td>
<td>2.4863E-02</td>
<td>3.875E-02</td>
<td>2.1136E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17.00</td>
<td>5.876E-02</td>
<td>6.566E-02</td>
<td>1.4325E-02</td>
<td>9.336E-05</td>
<td>2.3725E-02</td>
<td>1.5408E-02</td>
<td>2.4662E-02</td>
<td>3.877E-02</td>
<td>2.1000E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18.00</td>
<td>5.875E-02</td>
<td>6.517E-02</td>
<td>1.4310E-02</td>
<td>8.532E-05</td>
<td>2.3728E-02</td>
<td>1.5241E-02</td>
<td>2.4862E-02</td>
<td>3.877E-02</td>
<td>2.083E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19.00</td>
<td>5.875E-02</td>
<td>6.477E-02</td>
<td>1.4297E-02</td>
<td>7.840E-05</td>
<td>2.3720E-02</td>
<td>1.5078E-02</td>
<td>2.4862E-02</td>
<td>3.877E-02</td>
<td>2.071E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20.00</td>
<td>5.875E-02</td>
<td>6.442E-02</td>
<td>1.4286E-02</td>
<td>7.237E-05</td>
<td>2.3719E-02</td>
<td>1.4917E-02</td>
<td>2.4861E-02</td>
<td>3.877E-02</td>
<td>2.063E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21.00</td>
<td>5.875E-02</td>
<td>6.412E-02</td>
<td>1.4275E-02</td>
<td>6.710E-05</td>
<td>2.3717E-02</td>
<td>1.4764E-02</td>
<td>2.4861E-02</td>
<td>3.877E-02</td>
<td>2.061E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22.00</td>
<td>5.875E-02</td>
<td>6.385E-02</td>
<td>1.4267E-02</td>
<td>6.245E-05</td>
<td>2.3716E-02</td>
<td>1.4613E-02</td>
<td>2.4860E-02</td>
<td>3.877E-02</td>
<td>2.061E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>23.00</td>
<td>5.875E-02</td>
<td>6.360E-02</td>
<td>1.4259E-02</td>
<td>5.835E-05</td>
<td>2.3715E-02</td>
<td>1.4469E-02</td>
<td>2.4860E-02</td>
<td>3.877E-02</td>
<td>2.061E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>24.00</td>
<td>5.875E-02</td>
<td>6.336E-02</td>
<td>1.4253E-02</td>
<td>5.470E-05</td>
<td>2.3714E-02</td>
<td>1.4326E-02</td>
<td>2.4860E-02</td>
<td>3.877E-02</td>
<td>2.061E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>25.00</td>
<td>5.875E-02</td>
<td>6.317E-02</td>
<td>1.4247E-02</td>
<td>5.143E-05</td>
<td>2.3713E-02</td>
<td>1.4193E-02</td>
<td>2.4860E-02</td>
<td>3.877E-02</td>
<td>2.061E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26.00</td>
<td>5.875E-02</td>
<td>6.304E-02</td>
<td>1.4241E-02</td>
<td>4.854E-05</td>
<td>2.3712E-02</td>
<td>1.4067E-02</td>
<td>2.4860E-02</td>
<td>3.877E-02</td>
<td>2.061E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>27.00</td>
<td>5.875E-02</td>
<td>6.290E-02</td>
<td>1.4236E-02</td>
<td>4.593E-05</td>
<td>2.3711E-02</td>
<td>1.4042E-02</td>
<td>2.4860E-02</td>
<td>3.877E-02</td>
<td>2.061E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>28.00</td>
<td>5.875E-02</td>
<td>6.276E-02</td>
<td>1.4232E-02</td>
<td>4.357E-05</td>
<td>2.3711E-02</td>
<td>1.4007E-02</td>
<td>2.4860E-02</td>
<td>3.877E-02</td>
<td>2.061E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>29.00</td>
<td>5.875E-02</td>
<td>6.263E-02</td>
<td>1.4228E-02</td>
<td>4.143E-05</td>
<td>2.3710E-02</td>
<td>1.3972E-02</td>
<td>2.4860E-02</td>
<td>3.877E-02</td>
<td>2.061E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30.00</td>
<td>5.875E-02</td>
<td>6.250E-02</td>
<td>1.4224E-02</td>
<td>3.951E-05</td>
<td>2.3709E-02</td>
<td>1.3937E-02</td>
<td>2.4860E-02</td>
<td>3.877E-02</td>
<td>2.061E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>31.00</td>
<td>5.875E-02</td>
<td>6.237E-02</td>
<td>1.4220E-02</td>
<td>3.775E-05</td>
<td>2.3708E-02</td>
<td>1.3903E-02</td>
<td>2.4859E-02</td>
<td>3.877E-02</td>
<td>2.061E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>32.00</td>
<td>5.875E-02</td>
<td>6.224E-02</td>
<td>1.4216E-02</td>
<td>3.614E-05</td>
<td>2.3707E-02</td>
<td>1.3870E-02</td>
<td>2.4859E-02</td>
<td>3.877E-02</td>
<td>2.061E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>33.00</td>
<td>5.875E-02</td>
<td>6.211E-02</td>
<td>1.4212E-02</td>
<td>3.468E-05</td>
<td>2.3706E-02</td>
<td>1.3838E-02</td>
<td>2.4859E-02</td>
<td>3.877E-02</td>
<td>2.061E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Time (s)</td>
<td>Value</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34.00</td>
<td>5.807</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35.00</td>
<td>5.802</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36.00</td>
<td>5.797</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37.00</td>
<td>5.792</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>38.00</td>
<td>5.788</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39.00</td>
<td>5.784</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40.00</td>
<td>5.780</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>41.00</td>
<td>5.777</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42.00</td>
<td>5.773</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>43.00</td>
<td>5.770</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>44.00</td>
<td>5.766</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>45.00</td>
<td>5.765</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>46.00</td>
<td>5.762</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>47.00</td>
<td>5.760</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>48.00</td>
<td>5.758</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>49.00</td>
<td>5.756</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50.00</td>
<td>5.754</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>51.00</td>
<td>5.752</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Listing of tape 31 for sample case (tether spring constants).

LATERAL STABILITY OF TETHERED BALLCON - LKC BALLCON - REFERENCE CONFIGURATION 11/17/72 00.33.53.

<table>
<thead>
<tr>
<th>VELOCITY</th>
<th>SKYY</th>
<th>SKYP</th>
<th>SKYS</th>
<th>SKPP</th>
<th>SKPS</th>
<th>SKSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000</td>
<td>1.171</td>
<td>-3.379</td>
<td>3.372</td>
<td>12.64</td>
<td>-11.15</td>
<td>9.705</td>
</tr>
<tr>
<td>2.000</td>
<td>1.193</td>
<td>-3.962</td>
<td>3.418</td>
<td>13.16</td>
<td>-11.35</td>
<td>9.795</td>
</tr>
<tr>
<td>3.000</td>
<td>1.227</td>
<td>-4.559</td>
<td>3.496</td>
<td>13.67</td>
<td>-11.67</td>
<td>9.875</td>
</tr>
<tr>
<td>4.000</td>
<td>1.276</td>
<td>-5.170</td>
<td>3.510</td>
<td>14.16</td>
<td>-12.10</td>
<td>10.21</td>
</tr>
<tr>
<td>5.000</td>
<td>1.339</td>
<td>-5.792</td>
<td>3.535</td>
<td>14.66</td>
<td>-12.64</td>
<td>10.59</td>
</tr>
<tr>
<td>6.000</td>
<td>1.419</td>
<td>-6.419</td>
<td>3.565</td>
<td>15.20</td>
<td>-13.19</td>
<td>10.98</td>
</tr>
<tr>
<td>7.000</td>
<td>1.516</td>
<td>-7.061</td>
<td>4.215</td>
<td>17.47</td>
<td>-14.31</td>
<td>11.72</td>
</tr>
<tr>
<td>8.000</td>
<td>1.632</td>
<td>-7.721</td>
<td>4.518</td>
<td>18.91</td>
<td>-15.39</td>
<td>12.51</td>
</tr>
<tr>
<td>9.000</td>
<td>1.766</td>
<td>-8.392</td>
<td>4.876</td>
<td>20.38</td>
<td>-16.64</td>
<td>13.45</td>
</tr>
<tr>
<td>10.000</td>
<td>1.923</td>
<td>-9.087</td>
<td>5.237</td>
<td>22.47</td>
<td>-18.08</td>
<td>14.54</td>
</tr>
<tr>
<td>11.000</td>
<td>2.097</td>
<td>-9.811</td>
<td>5.733</td>
<td>24.59</td>
<td>-19.70</td>
<td>15.70</td>
</tr>
<tr>
<td>12.000</td>
<td>2.291</td>
<td>-10.561</td>
<td>6.271</td>
<td>26.93</td>
<td>-21.50</td>
<td>17.16</td>
</tr>
<tr>
<td>13.000</td>
<td>2.503</td>
<td>-11.332</td>
<td>6.840</td>
<td>29.49</td>
<td>-23.48</td>
<td>18.69</td>
</tr>
<tr>
<td>14.000</td>
<td>2.734</td>
<td>-12.124</td>
<td>7.460</td>
<td>32.27</td>
<td>-25.63</td>
<td>20.15</td>
</tr>
<tr>
<td>15.000</td>
<td>2.983</td>
<td>-12.945</td>
<td>8.129</td>
<td>35.27</td>
<td>-27.95</td>
<td>21.55</td>
</tr>
<tr>
<td>16.000</td>
<td>3.250</td>
<td>-13.807</td>
<td>8.847</td>
<td>38.47</td>
<td>-30.44</td>
<td>22.95</td>
</tr>
<tr>
<td>17.000</td>
<td>3.535</td>
<td>-14.701</td>
<td>9.613</td>
<td>41.89</td>
<td>-33.09</td>
<td>24.40</td>
</tr>
<tr>
<td>18.000</td>
<td>3.837</td>
<td>-15.636</td>
<td>10.43</td>
<td>44.52</td>
<td>-35.91</td>
<td>25.83</td>
</tr>
<tr>
<td>19.000</td>
<td>4.156</td>
<td>-16.611</td>
<td>11.29</td>
<td>47.30</td>
<td>-38.89</td>
<td>27.23</td>
</tr>
<tr>
<td>20.000</td>
<td>4.493</td>
<td>-17.628</td>
<td>12.19</td>
<td>50.24</td>
<td>-42.04</td>
<td>28.65</td>
</tr>
<tr>
<td>21.000</td>
<td>4.848</td>
<td>-18.671</td>
<td>13.15</td>
<td>53.39</td>
<td>-45.34</td>
<td>30.10</td>
</tr>
<tr>
<td>22.000</td>
<td>5.219</td>
<td>-19.749</td>
<td>14.15</td>
<td>56.73</td>
<td>-48.83</td>
<td>31.63</td>
</tr>
<tr>
<td>23.000</td>
<td>5.602</td>
<td>-20.865</td>
<td>15.20</td>
<td>60.34</td>
<td>-52.43</td>
<td>33.19</td>
</tr>
<tr>
<td>24.000</td>
<td>6.013</td>
<td>-21.999</td>
<td>16.29</td>
<td>64.12</td>
<td>-56.22</td>
<td>34.77</td>
</tr>
<tr>
<td>25.000</td>
<td>6.436</td>
<td>-23.163</td>
<td>17.43</td>
<td>68.11</td>
<td>-60.16</td>
<td>36.37</td>
</tr>
<tr>
<td>26.000</td>
<td>6.866</td>
<td>-24.357</td>
<td>18.62</td>
<td>72.32</td>
<td>-64.26</td>
<td>38.00</td>
</tr>
<tr>
<td>27.000</td>
<td>7.313</td>
<td>-25.574</td>
<td>19.65</td>
<td>76.81</td>
<td>-68.53</td>
<td>39.66</td>
</tr>
<tr>
<td>28.000</td>
<td>7.807</td>
<td>-26.816</td>
<td>21.13</td>
<td>81.58</td>
<td>-73.05</td>
<td>41.35</td>
</tr>
<tr>
<td>29.000</td>
<td>8.298</td>
<td>-28.090</td>
<td>22.45</td>
<td>86.67</td>
<td>-77.74</td>
<td>43.07</td>
</tr>
<tr>
<td>30.000</td>
<td>8.806</td>
<td>-30.408</td>
<td>23.92</td>
<td>92.01</td>
<td>-82.63</td>
<td>44.82</td>
</tr>
<tr>
<td>31.000</td>
<td>9.331</td>
<td>-32.857</td>
<td>25.59</td>
<td>97.60</td>
<td>-87.71</td>
<td>46.60</td>
</tr>
<tr>
<td>32.000</td>
<td>9.874</td>
<td>-35.344</td>
<td>27.20</td>
<td>103.42</td>
<td>-93.00</td>
<td>48.40</td>
</tr>
<tr>
<td>33.000</td>
<td>10.436</td>
<td>-37.962</td>
<td>28.86</td>
<td>109.47</td>
<td>-98.51</td>
<td>50.23</td>
</tr>
<tr>
<td>34.000</td>
<td>11.011</td>
<td>-39.709</td>
<td>30.60</td>
<td>115.79</td>
<td>-104.26</td>
<td>52.09</td>
</tr>
<tr>
<td>35.000</td>
<td>11.601</td>
<td>-41.586</td>
<td>32.41</td>
<td>122.24</td>
<td>-110.20</td>
<td>53.97</td>
</tr>
<tr>
<td>36.000</td>
<td>12.211</td>
<td>-43.592</td>
<td>34.31</td>
<td>128.89</td>
<td>-116.30</td>
<td>55.87</td>
</tr>
<tr>
<td>37.000</td>
<td>12.842</td>
<td>-45.728</td>
<td>36.31</td>
<td>135.74</td>
<td>-122.55</td>
<td>57.79</td>
</tr>
<tr>
<td>38.000</td>
<td>13.494</td>
<td>-47.986</td>
<td>38.45</td>
<td>142.80</td>
<td>-129.00</td>
<td>59.73</td>
</tr>
<tr>
<td>39.000</td>
<td>14.154</td>
<td>-49.369</td>
<td>38.83</td>
<td>149.06</td>
<td>-135.60</td>
<td>61.70</td>
</tr>
<tr>
<td>40.000</td>
<td>14.838</td>
<td>-51.912</td>
<td>40.36</td>
<td>155.62</td>
<td>-142.45</td>
<td>63.70</td>
</tr>
<tr>
<td>41.000</td>
<td>15.520</td>
<td>-54.666</td>
<td>41.94</td>
<td>162.50</td>
<td>-149.50</td>
<td>65.71</td>
</tr>
<tr>
<td>42.000</td>
<td>16.240</td>
<td>-57.633</td>
<td>43.67</td>
<td>169.64</td>
<td>-156.75</td>
<td>67.73</td>
</tr>
<tr>
<td>43.000</td>
<td>16.977</td>
<td>-60.833</td>
<td>45.54</td>
<td>176.98</td>
<td>-164.20</td>
<td>69.77</td>
</tr>
<tr>
<td>44.000</td>
<td>17.727</td>
<td>-64.265</td>
<td>47.55</td>
<td>184.55</td>
<td>-171.85</td>
<td>71.82</td>
</tr>
<tr>
<td>45.000</td>
<td>18.497</td>
<td>-67.839</td>
<td>49.69</td>
<td>192.37</td>
<td>-179.60</td>
<td>73.89</td>
</tr>
<tr>
<td>46.000</td>
<td>19.286</td>
<td>-71.560</td>
<td>52.32</td>
<td>200.32</td>
<td>-187.54</td>
<td>75.98</td>
</tr>
<tr>
<td>47.000</td>
<td>20.090</td>
<td>-75.430</td>
<td>54.16</td>
<td>208.48</td>
<td>-195.70</td>
<td>78.10</td>
</tr>
<tr>
<td>48.000</td>
<td>20.898</td>
<td>-79.442</td>
<td>56.10</td>
<td>216.84</td>
<td>-204.06</td>
<td>80.24</td>
</tr>
<tr>
<td>49.000</td>
<td>21.711</td>
<td>-83.600</td>
<td>58.33</td>
<td>225.36</td>
<td>-212.54</td>
<td>82.40</td>
</tr>
<tr>
<td>50.000</td>
<td>22.536</td>
<td>-87.908</td>
<td>60.92</td>
<td>234.07</td>
<td>-221.17</td>
<td>84.58</td>
</tr>
<tr>
<td>51.000</td>
<td>23.373</td>
<td>-92.363</td>
<td>63.26</td>
<td>243.01</td>
<td>-230.00</td>
<td>86.77</td>
</tr>
</tbody>
</table>

81
<table>
<thead>
<tr>
<th>VELLCITY</th>
<th>X1</th>
<th>Z1</th>
<th>GAM1</th>
<th>TJ</th>
<th>GAM1</th>
<th>T1</th>
<th>CAB CRAG</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000</td>
<td>4.953</td>
<td>61.00</td>
<td>69.19</td>
<td>61.51</td>
<td>89.62</td>
<td>82.43</td>
<td>1.0104E-02</td>
</tr>
<tr>
<td>2.000</td>
<td>1.943</td>
<td>60.96</td>
<td>56.82</td>
<td>62.77</td>
<td>89.30</td>
<td>83.68</td>
<td>4.0614E-02</td>
</tr>
<tr>
<td>3.000</td>
<td>4.236</td>
<td>60.83</td>
<td>65.08</td>
<td>64.78</td>
<td>88.46</td>
<td>85.65</td>
<td>9.0940E-02</td>
</tr>
<tr>
<td>4.000</td>
<td>7.218</td>
<td>60.51</td>
<td>73.28</td>
<td>67.49</td>
<td>87.35</td>
<td>88.24</td>
<td>-0.167</td>
</tr>
<tr>
<td>5.000</td>
<td>10.69</td>
<td>59.92</td>
<td>72.78</td>
<td>70.85</td>
<td>86.01</td>
<td>91.40</td>
<td>0.2526</td>
</tr>
<tr>
<td>6.000</td>
<td>14.44</td>
<td>59.02</td>
<td>66.38</td>
<td>74.37</td>
<td>94.48</td>
<td>95.11</td>
<td>-0.3638</td>
</tr>
</tbody>
</table>

LATERAL STABILITY OF TETHERED BALLOON - LRC BALLOON - REFERENCE CONFIGURATION 11/17/72 00.33.53.
### Listing of tape 33 for sample case (uncoupled characteristic roots)

**LATERAL STABILITY OF TETHERED BALLOON - LAG BALLOON - REFERENCE CONFIGURATION  11/17/72 00.33.53.**

<table>
<thead>
<tr>
<th>VEL</th>
<th>RLY1</th>
<th>RLY2</th>
<th>RLP1</th>
<th>RLP2</th>
<th>RLS1</th>
<th>RLS2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000</td>
<td>-4.4683E-02</td>
<td>-4.4683E-02</td>
<td>-7.7458E-02</td>
<td>-7.7458E-02</td>
<td>-1.252</td>
<td>-1.252</td>
</tr>
<tr>
<td></td>
<td>1.162</td>
<td>-1.162</td>
<td>4.368</td>
<td>-4.368</td>
<td>2.2103</td>
<td>-2.2103</td>
</tr>
<tr>
<td>2.000</td>
<td>-8.9366E-02</td>
<td>-8.9366E-02</td>
<td>-1.1580</td>
<td>-1.1580</td>
<td>-1.677</td>
<td>-3.127</td>
</tr>
<tr>
<td></td>
<td>-1.149</td>
<td>-1.149</td>
<td>4.411</td>
<td>-4.411</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3.000</td>
<td>-1.1340</td>
<td>-1.1340</td>
<td>-2.263</td>
<td>-2.263</td>
<td>-8.515E-02</td>
<td>-2.6647</td>
</tr>
<tr>
<td></td>
<td>-1.113</td>
<td>-1.113</td>
<td>4.474</td>
<td>-4.474</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4.000</td>
<td>-1.1787</td>
<td>-1.1787</td>
<td>-3.183</td>
<td>-3.183</td>
<td>-5.6692E-02</td>
<td>-9.9465</td>
</tr>
<tr>
<td></td>
<td>1.7479E-02</td>
<td>-1.7479E-02</td>
<td>4.550</td>
<td>-4.550</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5.000</td>
<td>-9.6676E-02</td>
<td>-3.502</td>
<td>-4.018</td>
<td>-4.018</td>
<td>-4.074E-02</td>
<td>-1.206</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>4.635</td>
<td>-4.635</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6.000</td>
<td>-7.8326E-02</td>
<td>-4.579</td>
<td>-4.894</td>
<td>-4.894</td>
<td>-3.0034E-02</td>
<td>-1.464</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>4.725</td>
<td>-4.725</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7.000</td>
<td>-8.835E-02</td>
<td>-5.567</td>
<td>-5.716</td>
<td>-5.716</td>
<td>-2.0202E-02</td>
<td>-1.719</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>4.819</td>
<td>-4.819</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8.000</td>
<td>-4.3315E-02</td>
<td>-6.516</td>
<td>-6.572</td>
<td>-6.572</td>
<td>-1.5657E-02</td>
<td>-1.473</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>4.916</td>
<td>-4.916</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9.000</td>
<td>-6.0235E-02</td>
<td>-7.443</td>
<td>-7.430</td>
<td>-7.430</td>
<td>-1.0340E-02</td>
<td>-2.226</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>5.321</td>
<td>-5.321</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10.00</td>
<td>-5.8170E-02</td>
<td>-8.355</td>
<td>-6.228</td>
<td>-6.228</td>
<td>-5.7074E-03</td>
<td>-2.477</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>5.129</td>
<td>-5.129</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11.00</td>
<td>-5.7268E-02</td>
<td>-9.258</td>
<td>-6.948</td>
<td>-6.948</td>
<td>-1.7686E-03</td>
<td>-2.729</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>5.243</td>
<td>-5.243</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>12.00</td>
<td>-5.7026E-02</td>
<td>-1.015</td>
<td>-1.000</td>
<td>-1.000</td>
<td>1.856E-03</td>
<td>-2.979</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>5.361</td>
<td>-5.361</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>13.00</td>
<td>-5.7285E-02</td>
<td>-1.104</td>
<td>-1.086</td>
<td>-1.086</td>
<td>5.1838E-03</td>
<td>-3.230</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>5.480</td>
<td>-5.480</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>14.00</td>
<td>-5.7914E-02</td>
<td>-1.193</td>
<td>-1.172</td>
<td>-1.172</td>
<td>8.2816E-03</td>
<td>-3.400</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>5.613</td>
<td>-5.613</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>15.00</td>
<td>-5.8827E-02</td>
<td>-1.282</td>
<td>-1.258</td>
<td>-1.258</td>
<td>1.1199E-02</td>
<td>-3.731</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>5.747</td>
<td>-5.747</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16.00</td>
<td>-5.9963E-02</td>
<td>-1.370</td>
<td>-1.344</td>
<td>-1.344</td>
<td>1.3972E-02</td>
<td>-3.981</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>5.886</td>
<td>-5.886</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>17.00</td>
<td>-6.1276E-02</td>
<td>-1.458</td>
<td>-1.429</td>
<td>-1.429</td>
<td>1.6629E-02</td>
<td>-4.231</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>6.029</td>
<td>-6.029</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>18.00</td>
<td>-6.2732E-02</td>
<td>-1.546</td>
<td>-1.515</td>
<td>-1.515</td>
<td>1.9189E-02</td>
<td>-4.481</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>6.176</td>
<td>-6.176</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>19.00</td>
<td>-6.4305E-02</td>
<td>-1.634</td>
<td>-1.600</td>
<td>-1.600</td>
<td>2.1669E-02</td>
<td>-4.731</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>6.320</td>
<td>-6.320</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20.00</td>
<td>-6.5576E-02</td>
<td>-1.721</td>
<td>-1.686</td>
<td>-1.686</td>
<td>2.4083E-02</td>
<td>-4.980</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>6.463</td>
<td>-6.463</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>21.00</td>
<td>-6.7729E-02</td>
<td>-1.809</td>
<td>-1.771</td>
<td>-1.771</td>
<td>2.6438E-02</td>
<td>-5.230</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>6.642</td>
<td>-6.642</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>22.00</td>
<td>-6.9152E-02</td>
<td>-1.898</td>
<td>-1.857</td>
<td>-1.857</td>
<td>2.8745E-02</td>
<td>-5.480</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>6.805</td>
<td>-6.805</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>23.00</td>
<td>-7.1435E-02</td>
<td>-1.984</td>
<td>-1.942</td>
<td>-1.942</td>
<td>3.1010E-02</td>
<td>-5.730</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>6.971</td>
<td>-6.971</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>24.00</td>
<td>-7.3370E-02</td>
<td>-2.071</td>
<td>-2.027</td>
<td>-2.027</td>
<td>3.3237E-02</td>
<td>-5.979</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td></td>
<td>25.00</td>
<td>-7.539E-02</td>
<td>-2.159</td>
<td>-2.113</td>
<td>-2.113</td>
<td>3.5432E-02</td>
</tr>
<tr>
<td></td>
<td>26.00</td>
<td>-7.756E-02</td>
<td>-2.246</td>
<td>-2.198</td>
<td>-2.198</td>
<td>3.7597E-02</td>
</tr>
<tr>
<td></td>
<td>27.00</td>
<td>-7.923E-02</td>
<td>-2.333</td>
<td>-2.283</td>
<td>-2.283</td>
<td>3.9738E-02</td>
</tr>
<tr>
<td></td>
<td>28.00</td>
<td>-8.159E-02</td>
<td>-2.421</td>
<td>-2.368</td>
<td>-2.368</td>
<td>4.1855E-02</td>
</tr>
<tr>
<td></td>
<td>29.00</td>
<td>-8.362E-02</td>
<td>-2.508</td>
<td>-2.454</td>
<td>-2.454</td>
<td>4.3952E-02</td>
</tr>
<tr>
<td></td>
<td>30.00</td>
<td>-8.571E-02</td>
<td>-2.595</td>
<td>-2.539</td>
<td>-2.539</td>
<td>4.6031E-02</td>
</tr>
<tr>
<td></td>
<td>31.00</td>
<td>-8.792E-02</td>
<td>-2.682</td>
<td>-2.624</td>
<td>-2.624</td>
<td>4.8092E-02</td>
</tr>
<tr>
<td></td>
<td>32.00</td>
<td>-9.010E-02</td>
<td>-2.770</td>
<td>-2.709</td>
<td>-2.709</td>
<td>5.0139E-02</td>
</tr>
<tr>
<td></td>
<td>33.00</td>
<td>-9.230E-02</td>
<td>-2.857</td>
<td>-2.794</td>
<td>-2.794</td>
<td>5.2171E-02</td>
</tr>
<tr>
<td></td>
<td>34.00</td>
<td>-9.452E-02</td>
<td>-2.944</td>
<td>-2.880</td>
<td>-2.880</td>
<td>5.4191E-02</td>
</tr>
<tr>
<td></td>
<td>35.00</td>
<td>-9.675E-02</td>
<td>-3.031</td>
<td>-2.965</td>
<td>-2.965</td>
<td>5.6200E-02</td>
</tr>
<tr>
<td></td>
<td>36.00</td>
<td>-9.900E-02</td>
<td>-3.118</td>
<td>-3.050</td>
<td>-3.050</td>
<td>5.8197E-02</td>
</tr>
<tr>
<td></td>
<td>37.00</td>
<td>-1.013E-01</td>
<td>-3.205</td>
<td>-3.135</td>
<td>-3.135</td>
<td>6.0185E-02</td>
</tr>
<tr>
<td></td>
<td>40.00</td>
<td>-1.081E-01</td>
<td>-3.467</td>
<td>-3.390</td>
<td>-3.390</td>
<td>6.6097E-02</td>
</tr>
<tr>
<td></td>
<td>41.00</td>
<td>-1.104E-01</td>
<td>-3.554</td>
<td>-3.475</td>
<td>-3.475</td>
<td>6.8052E-02</td>
</tr>
<tr>
<td></td>
<td>42.00</td>
<td>-1.127E-01</td>
<td>-3.641</td>
<td>-3.560</td>
<td>-3.560</td>
<td>7.0000E-02</td>
</tr>
<tr>
<td></td>
<td>43.00</td>
<td>-1.150E-01</td>
<td>-3.728</td>
<td>-3.645</td>
<td>-3.645</td>
<td>7.1942E-02</td>
</tr>
<tr>
<td></td>
<td>44.00</td>
<td>-1.174E-01</td>
<td>-3.815</td>
<td>-3.730</td>
<td>-3.730</td>
<td>7.3876E-02</td>
</tr>
<tr>
<td></td>
<td>45.00</td>
<td>-1.197E-01</td>
<td>-3.902</td>
<td>-3.815</td>
<td>-3.815</td>
<td>7.5801E-02</td>
</tr>
<tr>
<td></td>
<td>46.00</td>
<td>-1.221E-01</td>
<td>-3.989</td>
<td>-3.901</td>
<td>-3.901</td>
<td>7.7735E-02</td>
</tr>
<tr>
<td></td>
<td>47.00</td>
<td>-1.244E-01</td>
<td>-4.076</td>
<td>-3.986</td>
<td>-3.986</td>
<td>7.9656E-02</td>
</tr>
<tr>
<td></td>
<td>48.00</td>
<td>-1.268E-01</td>
<td>-4.163</td>
<td>-4.071</td>
<td>-4.071</td>
<td>8.1572E-02</td>
</tr>
<tr>
<td></td>
<td>49.00</td>
<td>-1.291E-01</td>
<td>-4.250</td>
<td>-4.156</td>
<td>-4.156</td>
<td>8.3484E-02</td>
</tr>
<tr>
<td></td>
<td>50.00</td>
<td>-1.315E-01</td>
<td>-4.337</td>
<td>-4.241</td>
<td>-4.241</td>
<td>8.5392E-02</td>
</tr>
<tr>
<td></td>
<td>51.00</td>
<td>-1.339E-01</td>
<td>-4.424</td>
<td>-4.326</td>
<td>-4.326</td>
<td>8.7286E-02</td>
</tr>
</tbody>
</table>
PROGRAM FOR PLOTTING FREQUENCIES AND DECAY RATES
VERSUS WIND VELOCITY

General Description

The dynamic characteristics of a tethered balloon may vary considerably with wind velocity. Plots of the frequencies $\text{Im}(\lambda)$ and of the decay rates $\text{Re}(\lambda)$ versus wind velocity are helpful in assessing trends. Program VPLOT is used to plot both the frequencies and the decay rates versus wind velocity for the longitudinal and lateral cases. A sample plot is given in figure 3.

The parameters for the scales of the plot are set with data statements within the program and must be changed internally if desired. The number of frequencies and number of decay rates off scale are counted and written on the plot (fig. 3). All of the data for the program are read from binary tape 7 written by program STABLTY. The file INPUT is thus deleted from the file assignments for the program because no data are read from it. The only printing from the program is the identification array and the number of velocity increments to be processed, both of which are read from tape 7. The principal plotting routines are described in the appendix. The version of the program given here requires 30-inch-wide paper for the plotter.

Definitions of Program Variables

Some of the principal FORTRAN variable names are given and defined in the following sections. The variables associated with scaling and axes are:

<table>
<thead>
<tr>
<th>FORTRAN variable name</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAXL, FAXL, VAXL</td>
<td>length of plot axes for decay rate, frequency, and velocity, respectively, in.</td>
</tr>
<tr>
<td>DLAB, FLAB, VLAB</td>
<td>arrays of labeling information for decay rate, frequency, and velocity axes, respectively</td>
</tr>
<tr>
<td>DDEL, FDEL, VDEL</td>
<td>scale factors for decay rate, frequency, and velocity (change in units per in. of plot)</td>
</tr>
<tr>
<td>DMIN, FMIN, VMIN</td>
<td>minimum values of decay rate, frequency, and velocity to be plotted</td>
</tr>
<tr>
<td>DMAX, FMAX</td>
<td>maximum values of decay rate and frequency to be plotted</td>
</tr>
</tbody>
</table>
Figure 3.- Example of plot from program VPLOT.
<table>
<thead>
<tr>
<th>FORTRAN variable name</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>DTICL, FTICL, VTICL</td>
<td>distance between large tick marks in inches on plot for the damping, frequency, and velocity axes, respectively</td>
</tr>
<tr>
<td>DZERO</td>
<td>distance in inches from zero decay rate level to plot origin as set by CALPLT</td>
</tr>
<tr>
<td>FFHH</td>
<td>vertical distance in inches from plot origin to end of frequency axis</td>
</tr>
<tr>
<td>FH</td>
<td>vertical distance in inches from plot origin to zero frequency level</td>
</tr>
<tr>
<td>HGT</td>
<td>height of lettering for labeling, in.</td>
</tr>
<tr>
<td>TICSND, TICSNF, TICSNV</td>
<td>number of small tick marks per inch on plot for the damping, frequency, and velocity axes, respectively</td>
</tr>
</tbody>
</table>

The variables read from tape 7 are:

<table>
<thead>
<tr>
<th>FORTRAN variable name</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>ten-word array containing case identification and date and time of processing of case by STABLY</td>
</tr>
<tr>
<td>IK2</td>
<td>index for cycling symbols</td>
</tr>
<tr>
<td>NEGR</td>
<td>one plus the number of complex roots with negative frequencies</td>
</tr>
<tr>
<td>NROOT</td>
<td>number of roots for plotting for a given velocity</td>
</tr>
<tr>
<td>NTWO</td>
<td>order of linear eigenvalue problem</td>
</tr>
<tr>
<td>NVEL</td>
<td>number of velocity increments</td>
</tr>
<tr>
<td>ROOTI</td>
<td>array containing imaginary portion of eigenvalues (modal frequencies for a given velocity)</td>
</tr>
<tr>
<td>FORTRAN variable name</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------------------</td>
<td>------------</td>
</tr>
<tr>
<td>ROOTR</td>
<td>array containing real portion of eigenvalues (decay rates) for a given velocity, in same sequence as ROOTI</td>
</tr>
<tr>
<td>VEL</td>
<td>velocity (first in velocity units, then scaled to in. on plot)</td>
</tr>
</tbody>
</table>

Other variables are:

<table>
<thead>
<tr>
<th>FORTRAN variable name</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISYM</td>
<td>index for plotting symbols</td>
</tr>
<tr>
<td>QLSYM</td>
<td>symbol quality parameter (+1. for high quality, -1. for low quality)</td>
</tr>
<tr>
<td>NDOFF, NFOFF</td>
<td>number of decay rates or frequencies off scale (encoded to Hollerith as DOFF and FOFF, respectively, for labeling plot)</td>
</tr>
<tr>
<td>RIN, RRN</td>
<td>real and imaginary parts of an eigenvalue (scaled to in. from plot origin as FREQ and DECAY, respectively)</td>
</tr>
</tbody>
</table>
Listing of Program

OVERLAY(VPLOT,0,0)
PROGRAM VPLOT(OUTPUT=1,TAPE7)

C******************************************************************************
C* PROGRAM A2864.3 - ROOTS VS VELOCITY PLOTTING PROGRAM *
C* PLOTS REAL(ROOT) VS VELOCITY AND IMAG(ROOT) VS VELOCITY *
C* SEE SUBROUTINE WRITEUPS FOR DESCRIPTION OF CALPLT,NUTATE,AXES, *
C* PNTPLT, OR DASHLN *
C* C******************************************************************************

DIMENSION ID(10),ROOTR(50),ROOTT(50),VLABS(15),DLAB(15),FLAB(15)

C SCALING PARAMETERS AND LABELS FOR AXES
C
DATA DLAB/50H DECAY RATE, 1/SEC / 
DATA FLS3/50H VELOCITY, M/SEC / 
DATA VLAB/50H VELOCITY, M/SEC / 
DATA VLAB/50H VELOCITY, M/SEC / 
DATA VLAB/50H VELOCITY, M/SEC / 
DATA VLAB/50H VELOCITY, M/SEC / 
DATA VLAB/50H VELOCITY, M/SEC / 
DATA VLAB/50H VELOCITY, M/SEC / 
DATA VLAB/50H VELOCITY, M/SEC / 
DATA VLAB/50H VELOCITY, M/SEC / 
C
CALL CALCOMP $ CALL CALPLT(3.3,3.-3) 
CALSYM=1. $ IF(CALSYM.GT.0.) CALL LEROY 
RDDEL=1./VUEL $ RDDEL=1./VUEL 
FMAX=VMAX*FDEL $ DMAX=DMIN*DAXL*DDDEL $ DZERO=-DMIN*RDDEL 
FH=(MAX-MIN)*RDDEL+1. 
C
READ IC ARRAY AND NUMBER OF VELOCITY POINTS FROM BINARY TAPE7 
SEE SUBROUTINE WRITEUP FOR DESCRIPTION OF RECIN 
1 CALL RECIN(7,2,IEOF,ID,1,10,1) $ IF(IEOF,7999,2 
2 CALL RECIN(7,1,IEOF,NVEL) 
PRINT 100,ID,NVEL $ NOFF=0 $ NOFF=0 
C
WRITE IC ARRAY BELOW HORIZONTAL AXIS OF PLOT AND DRAW X-Y AXES WITH 
WITH TIC-MARK GRIDS, NUMBERS AND LABELS 
C
CALL NUTATE(0.,-4.*HGT,50,HGT,ID,0.,100) 
CALL AXES(0.,0.,0.,VAXL,VMIN,VOEL,VTICL,TICSV7,VLAB,HGT,-50) 
CALL AXES(0.,0.,0.,VAXL,VMIN,VOEL,VTICL,TICSV7,VLAB,HGT,-50) 
CALL AXES(-5.,0.,90.,DAXL,DMIN,DEURL,TICSV7,DLAB,HGT,50) 
CALL AXES(-5.,0.,90.,DAXL,DMIN,DEURL,TICSV7,DLAB,HGT,50) 
C
DRAW LINE FOR ZERO DECAY RATE 
C
IF(DZERO.GT.0.)CALL DASHLN(0.,DZERO,VAXL,DZERO,VAXL) 
C
90-LOOP IS VELOCITY LOOP 
C
DO 90 IV=1,NVEL $ CALL RECIN(7,1,VEL,IK2,NEG,NTW) 
NUCT=NTW-NEG+1 
CALL RECIN(7,2,NROOT,ROOTR,1,NROOT,1) 
CALL RECIN(7,2,NROOT,ROOTT,1,NROOT,1) 
VEL=(VEL-VMIN)*VUEL 
90 CONTINUE
20-LOOP IS ROOT PLOTTING LOOP FOR VELOCITY VEL

DO 20 N=1, NROUT $ RR=ROOTR(N) $ RI=ROOTI(N)

SCALE FOR PLOTTING AND SET UP SYMBOL CYCLING

DECAY=(RR-OMIN)*RODEL $ FREQ=(RR-FMIN)*RFDEL*FH
IF(IK2, EQ, 11) GO TO 10 $ ISYM=QLSYM*(IK2+10)

PLOT SYMBOL IF DECAY RATE IS ON PLOT - IF OFF-SCALE, INCREMENT NDFF

- REPEAT FOR FREQUENCY, NFUff

IF(1K1+1, GE, FMIN, AND, 1K1, LE, FMAX)4, 5
4 CALL PNTPLT(VEL, DECAY, [SYM], 1) $ GO TO 6
5 NDFF=NDFF+1
6 IF(1K1, GE, DMIN, AND, 1K1, LE, DMAX)7, 8
7 CALL PNTPLT(VEL, FREQ, [SYM], 1) $ GO TO 20
8 NFUff=NFUff+1 $ GO TO 20

PLOT PLUS SIGN IF DECAY RATE IS ON PLOT - IF OFF-SCALE, INCREMENT NDFF

- REPEAT FOR FREQUENCY, NFUff

10 IF(1K1, GE, DMIN, AND, 1K1, LE, DMAX)12, 13
12 CALL NCTATE(VAXL, DECAY, 07, 3, 0, -1) $ GO TO 14
13 NDFF=NDFF+1
14 IF(1K1, GE, FMIN, AND, 1K1, LE, FMAX)15, 16
15 CALL NCTATE(VEL, FREQ, 07, 3, C, -1) $ GO TO 20
16 NFUff=NFUff+1
20 CONTINUE
90 CONTINUE

ENCODE NUMBER OF POINTS OFF PLOTT AND WRITE ON PLOT

ENCODE(10, 101, NDFF, NDFF) $ ENCODE(10, 101, NFUff, NFUff)
CALL NCTATE(VAXL, HAXL, HGT, DCFF, 0, 10) $ FFHH=FHH+FAXL
CALL NCTATE(VAXL, FFHH, FGT, FOFF, 0, 10)

SHIFT ORIGIN AND CHECK FOR NEXT CASE

CALL CALPLT(VAXL+6.1, 0, -3) $ GO TO 1

999 CALL CALPLT(0, 0, 999) $ REWIND 7
101 FORMAT(14, 6HPT OFF)
100 FORMAT(/2X10A10/# NVEL==14)
END PROGRAM VPlot
PROGRAM FOR PLOTTING ROOTS IN ROOT-LOCUS FORM
WITH WIND VELOCITY AS A PARAMETER

General Description

One form of plotting characteristic roots often used in parametric stability investigations is the plotting of $\text{Im}(\lambda)$ versus $\text{Re}(\lambda)$; that is, frequency versus decay rate. Curves are formed by the roots as a parameter is varied. Root-locus diagrams are often used because the qualitative variation of the roots with the parameter can be sketched with no computation, providing the parameter enters the stability determinant in a simple manner. Wind velocity is used as the parameter and enters the stability determinant in a complicated fashion. Thus, the root-locus plots generated by program RTLOCUS is used only as a form of plotting the calculated results in order to assist in interpreting such trends as the splitting of a complex pair into real roots. One feature of this type of plot is that radial lines from the origin form lines of constant damping ratio. A sample plot is given in figure 4.

The organization and operation of program RTLOCUS is quite similar to that of program VPLOT. The definitions of the FORTRAN variables are also essentially the same. The program uses the same tape 7 as VPLOT and is normally executed in series with VPLOT. The version of RTLOCUS given here requires 30-inch-wide paper.
Figure 4.- Example of plot from program RTLOCUS.
Listing of Program

OVERPLAY(RTOCUS,0,0)
PROGRAM RTOCULSP(OUTPUT=1,TAPE7)

C******************************************************************************
C* PROGRAM A2664.4 - ROUT LOCUS PLOTTING PROGRAM
C* PLOTS IMAG(ROTT) VS REAL(ROTT) WITH VELOCITY AS A PARAMETER
C* SEE SUBROUTINE WRITEUPS FOR DESCRIPTION OF CALPLT,NOTATE,AXES, OP
C* PRTPLT
C*
C******************************************************************************
DIMENSION ID(10),POOR(50),ROOTI(50),DLAB(5),FLAB(5)

C SCALING PARAMETERS AND LABELS FOR AXES
C
DATA CLAB/50H DECAY RATE, 1/SEC /
DATA DAXL,DMIN,DEDEL,DTCIL,TICSNO/10.,-6.,1.,1.,10. /
DATA FAXL,FMIN,FDEL,FTICL,TICSNF/10.,0.,1.,1.,10. /
DATA FLAB/50H FREQUENCY, RAD/SEC /
DATA HGT,FF/250.0./
C
C INITIALIZE PLOTTING ROUTINES, SET PLOT ORIGIN, AND CALCULATE SCALING
C PARAMETERS - CALL TO LFROY SLCWS PLOTTER FOR USING INK
C
CALL CALCUMP $ CALL CALPLT(1.5*HGT,-3)
QLSYM=1. $ IF QLSYM.GT.0.1 CALL LEROY
RDDEI=1./DEDEL $ RFDEL=1./FDEL
FMAX=FMIN*FAXL*FDEL $ DMAX=DMIN*DAXL*DEDEL $ DZERO=-DMIN*DEDEL
C
C READ IC ARRAY AND NUMBEP OF VELOCITY POINTS FROM BINARY TAPE7
C SEE SUBROUTINE WRITEUP FOR DESCRIPTION OF RECIN
C
1 CALL RECINI7,2,IEOF,ID,1,10,11 $ IF(IEOF,7)999,2
2 CALL RECINI7,1,IEOF,NVEL)
PRINT 100,ID,NVEL $ NOFF=0
C
C WRITE IC ARRAY BELOW HORIZONTAL AXIS OF PLOT AND DRAW X-Y AXES
C WITH TIC-MARK GRUIS, NUMBERS AND LABELS
C
CALL NOTATEID,-4.*HGT,5*HGT,10,0,100)
CALL AXESIO.,0.,0.,DAXL,DMIN,DEDEL,DTCIL,TICSNO,DLAB,HGT,-50)
CALL AXESIDZERO,0.,90.,FAXL,FMIN,FDEL,FTICL,TICSNF,FLAB,HGT,50)
C
C 90-LOOP IS VELOCITY LOOP
C
DO 90 IV=1,NVEL $ CALL RECINI7,1,4,VEL,IK2,NEGR,NTWO)
NROOT=NTWO-NEGR+1
CALL RECINI7,2,NROOT,ROOTR,1,NROUT,1)
CALL RECINI7,2,NROOT,ROOTI,1,NROUT,1)
C
C 20-LOOP IS ROOT PLOTTING LOOP FOR VELOCITY VEL
C
DO 20 N=1,NROOT $ RRN=ROOTR(N) $ RIN=ROOTI(N)
C
SCALE FOR PLOTTING AND SET UP SYMBOL CYCLING
C
DECAY=(RRN-DMIN)*DEDEL $ FREQ=(RIN-FMIN)*FDEL+FHL
IF(IK2.EQ.11)GO TO 10 $ ISYM=QLSYM*(IK2+10)
C
93
IF ON PLOT, PLOT SYMBOL - COUNT OFF SCALE POINTS

IF((RRN.GE.DMIN.AND.RRN.LE.DMAX) .AND.
1 (RIN.GE.FMIN.AND.RIN.LE.FMAX))8,9
8 CALL FNTPLT(DECAY,FREQ,ISYM,1) $ GO TO 20
9 NOFF=NOFF+1 $ GO TO 20

IF ON PLOT, PLOT PLUS SIGN - COUNT OFF SCALE POINTS

10 IF((RRN.GE.DMIN.AND.RRN.LE.DMAX) .AND.
1 (RIN.GE.FMIN.AND.RIN.LE.FMAX))14,15
14 CALL NOTATE(DECAY,FREQ,.07,3,0.,-1) $ GO TO 20
15 NOFF=NOFF+1
20 CONTINUE

CONTINUE

ENCODE NUMBER OF POINTS OFF PLOT AND WRITE ON PLOT

ENCODE(10,101,POFF)NOFF
CALL NOTATE(CAXL,FAXL,HGT,POFF,.0,.10)

SHIFT ORIGIN AND CHECK FOR NEXT CASE

CALL CALPLT(15,.0,.3) $ GO TO 1
959 CALL CALPLT(.0,.0,.999) $ REWIND 7
101 FORMAT(14,6HPT OFF)
100 FORMAT(/2X10A10/* NVEL=*I4)
END PROGRAM RTLOCLS

SUBROUTINE NOTATE(X,Y,HT,BCD,THETA,N)
CALL CFTRAN(X,Y,HT,BCD,THETA,N) $ RETURN
END SUBROUTINE NOTATE
PROGRAM FOR PLOTTING LONGITUDINAL MODES OF MOTION

As a means of illustrating the longitudinal modes of motion, the outline of the tethered balloon and the center of mass are displaced in proportion to the eigenvector and are drawn for a sequence of time intervals by program CALBALM. The balloon is viewed from the side. It is displaced proportional to the pitch-angle amplitude, which is an input to the program, and is selected initially such that the balloon remains on the plot. The center-of-mass position is shown as a plotted point for each time sample, and the balloon outline can be deleted for some time samples. Shifting of the plotting frame between time intervals is also optional. A sample plot is given in figure 5. It might be noted that this is not a transient response problem in the usual sense. In general, all modes of a dynamic system participate in transient motions to a degree depending upon the initial conditions or excitation. The purpose of program CALBALM is to illustrate the character of a single mode.

Program CALBALM is highly specialized as it is based on the shape of a particular balloon given in feet by DATA statements within the program (and the same is true for program CALBLM2 subsequently described). However, it can be modified for other purposes with minor reprogramming efforts. For example, a slightly modified version has been used for making computer-generated movies of the modes of motion.

Figure 5.- Example of plot from program CALBALM.
**Definition of Program Variables**

The principal FORTRAN variables for the program are given and defined except for the variables required for input data which are described subsequently. Note that the coordinate y refers to the plotting coordinate direction which is actually -z for the stability-axis system.

<table>
<thead>
<tr>
<th>FORTRAN variable name</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>AXESL</td>
<td>length of X- and Y-axes of plotting frame for a given time or times, in.</td>
</tr>
<tr>
<td>DAY</td>
<td>two-word array containing date and time of processing of case</td>
</tr>
<tr>
<td>DT</td>
<td>time increment between frames, 0.0625 * TIME (where TIME is time scale factor)</td>
</tr>
<tr>
<td>HGT</td>
<td>height of title (identification and date and time) written underneath initial frame</td>
</tr>
<tr>
<td>NB</td>
<td>number of points used to describe balloon half profile</td>
</tr>
<tr>
<td>NF</td>
<td>number of points in array used to describe fins</td>
</tr>
<tr>
<td>NLB</td>
<td>number of points in array used to describe load band</td>
</tr>
<tr>
<td>NT</td>
<td>number of points in array used to describe tether bridle</td>
</tr>
<tr>
<td>XAXE, YAXE</td>
<td>arrays containing coordinates for drawing axes for equilibrium position of balloon</td>
</tr>
<tr>
<td>XBAL, YBAL, XBTM, YBTM</td>
<td>x- and y-coordinates of balloon profile, given as station coordinates in feet from nose with balloon facing to the left (rotated and translated coordinates including motion are stored in XBTM and YBTM for plotting)</td>
</tr>
<tr>
<td>XCG, YCG</td>
<td>x- and y-coordinates of balloon center of mass in coordinate system consistent with XBAL and YBAL</td>
</tr>
</tbody>
</table>
FORTRAN variable name | Definition
--- | ---
XFIN, YFIN, XFTM, YFTM | fin coordinate array in coordinate system consistent with XBAL and YBAL (rotated and translated coordinates are stored in XFTM and YFTM)
XLBAND, YLBAND, XLBDTM, YLBDTM | load band coordinate array in coordinate system consistent with XBAL and YBAL, including motion stored in XLBDTM and YLBDTM
XMIN, YMIN, XMAX, YMAX | minimum and maximum values of x and y for limits of plotting frame, in units relating to full-scale balloon (here ft)
XMG, YMG | x- and y-coordinates of center of mass, including motion scaled to inches on plot
XTET, YTET, XTTM, YTTM | tether bridle x- and y-arrays in coordinate system consistent with XBAL and YBAL, including motion stored in XTTM and YTTM

Description of Input Data

The input data consist of four cards per case. The program can process multiple cases.

Card 1 (8A10 format).- Eighty columns of identification information read into ID array.

Card 2 (8F10.0 format).- The eight 10-column fields contain the following variables in sequence:

ROOTR | real part of eigenvalue for mode (decay rate), 1/sec
ROOTI | imaginary part of eigenvalue for mode (frequency), 1/sec
XREAL | real part of x-component of eigenvector normalized by θ, m/deg
XIMAG | imaginary part of x-component of eigenvector normalized by θ, m/deg
ZREAL | real part of z-component of eigenvector normalized by θ, m/deg
ZIMAG imaginary part of z-component of eigenvector normalized by \( \theta \), m/deg
TREAL real part of \( \theta \)-component of eigenvector, normally 1.0
TIMAG imaginary part of \( \theta \)-component of eigenvector, normally 0.0

Card 3 (BF10.0 format). - The first four 10-column fields contain the following variables in sequence:

AMPL initial amplitude of pitching motion for mode, deg
TRIMA trim angle of attack, deg
TIME time scale factor, \( DT = \Delta t = 0.0625 \times TIME \), seconds between frames
XSHIFT distance between origins of frames in inches on plot if frames are shifted between plot time intervals

Card 4 (2014 format). - The first four 4-column fields contain the following integer variables in sequence:

NFRM number of time increments
ISHIFT frame origin is shifted between plot time intervals only if = 0
MODBAL parameter determining interval for drawing full balloon outline (center-of-mass point is plotted MODBAL times as often as balloon outline)
INIMOD initial time frame for plotting balloon outline (= 0, outline plotted first frame; = 1, second frame, etc.)

Listing of Input Data Cards for Sample Case

```
LRC BALLOON - REFERENCE CONFIGURATION - MODE 1A AT V=20 M/S, ROOT=-.763+0.*I
-.76353 7. 1.3920 0. 1.2549 5.980 15. 10.
```

88
Listing of Program

OVERLAY(CALBAM,0,0)
PROGRAM CALBAM(OUTPUT=1,INPUT=1,TAPE5=INPUT)

*****************************************************************************

* PROGRAM A2864.5 - LONGITUDINAL EIGENVECTOR PLOTTING PROGRAM *
* FLGTS C G PCSTION AND/OR BALCON OUTLINE FOR A SINGLE MODE OF *
* MOTION FOR SELECTED INCREMENTS OF TIME *
* SEE SUBROUTINE WRITEUPS FOR DESCRIPTION OF CALPLOT,NOTATE,LINE, OR *
* PNTPLT - LEROY SLOWS PLOTTER FOR DRAWING WITH INK *
*
*****************************************************************************

DIMENSION XBAL(461,XFIN(121,XLBAND(21,XTET(31,XAXEIS(11+1,CQA(81,XTMI141
+ XBTM(481,XFTM(141,XLBUTM(41,XTTM(11
+ YBTM(481,YFTM(141,YLBUTM(41,YTTM(11
DATA AI/6HI8AIO1,FI0/SHI8FI0.01/,AO/10HI/XI0AI0/JI
CATA 14,AO~/6HIZO1,3AI011
DATA HGT/.1L5/,AXESl/I0.1,ISHIFT/O1
DATA DEGRAO/.01745329294/9431
DATA FTPERM/3.20839895/ SI UNITS
C DATA FOR LRC BALCON IN FEET
C BALCON DATA IS FOR NOSE FACING LEFT - TURNED AROUND IN CALCULATIONS
C DATA NB,NF,NL8,NT/23,12,2,3/ DATA (XBAl(I1,I=1,231/0,...+1,07,2,33,5,671,1,33,1,672,1+
+ 2.33,2.67,3,3,33,4,4,67,5,33,6,67,8,9,33,10,51,251
DATA YBAl(I1,I=1,231/0,...+1,245,95,1,33,1,72,2,03,2,52,2,88,+
+ 3,10,3,42,3,63,7,73,7,87,3,95,4,06,4,14,4,18,4,13,3,99,3,78,+
+ 3,57,331
DATA XFIN/17.22,25.93,2*25,...2*17.22,2*25,93,2*25,1,25,93,1,7221
DATA YFIN/2.07,4.25,3,2,03,20,2*03,2,03,2*03,2*03,2*031
DATA XBTM(481,XFTM(141,XLBUTM(41,XTTM(11
+ YBTM(481,YFTM(141,YLBUTM(41,YTTM(11
DATA HBG/6H1B8H101,F10/8H18F10.01/,AO/10H1/X10A101/
DATA IN,AGM/6H(20141,7H113A101/
DATA ED/1.251,AXESL/10.1,ISHIFT/O1
DATA DEGRAD/.0174532922519433/
DATA FTPERM/3.280839895/ SI UNITS
C C REFLECT BALCON HALF-PROFILE, INITIALIZE PLOTTING ROUTINES, COMPUTE
C SCALING PARAMETERS, AND SET ORIGIN FOR INITIAL FRAME
C Nb1=NBI+1$ NB2=NB1+NBS$ DU 1 I=NB1,NB2$I=NB2+1-I$ XBAl(I1=XBAl(I)
1 YBAl(I11=-YBAl(I1
SCALE=IXMAX-XMIN1/AXESL$ RSC1=1.1/SCALE$ ZER=.75
XAXE(111=-10. $ XAXE(211=-10. $ XAXE(311=-XAXE(711=XMIN
XAXE(511=XAXE(611=0. $ XAXE(411=XAXE(811=SCALE
YAXE(111=YAXE(211=-YAXE(311=YAXE(711=YMIN
YAXE(411=YAXE(511=SCALE$ YAXE(511=-10. $ YAXE(611=10.
CALL CALCMMP $ CALL CALPLOT(ZER,ZER,-3) $ CALL LEROY
C C READ IDENTIFICATION ARRAY, CALL FOR DATE AND TIME, AND PRINT AND
C WRITE AT THE BOTTOM OF THE FIRST FRAME
C 10 REAC AI,1D$ IF(EOF,51999,11
11 CALL CAYTIM(DAY) $ PRINT AO,1D,DAY
CALL NOTATE(0,15,HGT,1D,0.01,01
C C READ EIGENVECTOR-MOTION DATA, TRIM PARAMETERS, AND FRAME DATA
C READ F10,ROOTR,AUti,XREAL,XIMAG,ZREAL,ZIMAG,TREAL,TIMAG
READ F10,AMPL,TRIMA,TIME,XSHIFT$ READ I4,NFFM,ISHIFT,MODBAL,INIMOD
INIMCD=INIMCD+MODBAL
99
CONVERT DISPLACEMENT TO FEET, SCALE TIME, AND COMPUTE MAGNITUDES AND PHASES OF MODAL COMPONENTS

\[
X_{\text{REAL}} = \text{FTPERM} \times X_{\text{REAL}} \quad X_{\text{IMAG}} = \text{FTPERM} \times X_{\text{IMAG}}
\]

\[
Z_{\text{REAL}} = \text{FTPERM} \times Z_{\text{REAL}} \quad Z_{\text{IMAG}} = \text{FTPERM} \times Z_{\text{IMAG}}
\]

\[
\Delta t = 0.25 \times \text{TIME}
\]

\[
\Delta t_w = C \times \text{ROOTI} \quad \Delta t_r = \Delta t \times \text{ROCTR}
\]

\[
X_{\text{ABS}} = \sqrt{X_{\text{REAL}}^2 + X_{\text{IMAG}}^2} \quad Z_{\text{ABS}} = \sqrt{Z_{\text{REAL}}^2 + Z_{\text{IMAG}}^2}
\]

\[
\phi_X = \arctan \left( \frac{X_{\text{IMAG}}}{X_{\text{REAL}}} \right) \quad \phi_Z = \arctan \left( \frac{Z_{\text{IMAG}}}{Z_{\text{REAL}}} \right)
\]

\[
C_{\text{PHI}} = \cos(\phi_X) \quad S_{\text{PHI}} = \sin(\phi_X)
\]

50-LOOP IS TIME-FRAME LOOP

DO 50 NTS = 1, NFRM \& N = NTS - 1

CALCULATE MOTION FOR TIME FRAME

\[
S_T = \sin(N \times \Delta t_w) \quad C_T = \cos(N \times \Delta t_w) \quad A_E = A_E \times \exp[N \times \Delta t_r] \quad T_H = \text{CGRAD} \times (T_E + T_R) \quad C_T = \cos(T_H) \quad S_T = \sin(T_H)
\]

IF (SHIFT.EQ.0) GO TO 18

IF C G POINT UNLY IS DRAWN FOR THIS TIME FRAME GO TO 40

IF (WCC(N+INIMOD, MDFBAL) .NE. 0) GO TO 40

20-LOOP PREPARES ARRAYS FOR BALLOON OUTLINE FOR PLOTTING

25-LOOP - FINS, 30-LOOP - LOAD BAND, AND 35-LOOP - TETHER BRIDLE

POINTS OFF-SCALE ARE SET TO OUTER LIMITS OF FRAME

18 DO 20 I = 1, N2B \& XTEM = XREAL(I) - XCG \& YTEM = YBALL(I) - YCG

YBTM(I) = YTEM \times CTH - XTEM \times STH - ZM \& XBTM(I) = -(XTEM \times CTH + YTEM \times STH) - XM

IF (XBTM(I) .LT. XMIN) XBTM(I) = XMIN \& IF (XBTM(I) .GT. XMAX) XBTM(I) = XMAX

IF (YBTM(I) .LT. YMIN) YBTM(I) = YMIN \& IF (YBTM(I) .GT. YMAX) YBTM(I) = YMAX

20 CONTINUE \& DO 25 I = 1, N2F \& XTEM = XFIN(I) - XCG \& YTEM = YFIN(I) - YCG

YFTM(I) = YTEM \times CTH - XTEM \times STH - ZM \& XFTM(I) = -(XTEM \times CTH + YTEM \times STH) - XM

IF (XFTM(I) .LT. XMIN) XFTM(I) = XMIN \& IF (XFTM(I) .GT. XMAX) XFTM(I) = XMAX

IF (YFTM(I) .LT. YMIN) YFTM(I) = YMIN \& IF (YFTM(I) .GT. YMAX) YFTM(I) = YMAX

25 CONTINUE \& DO 30 I = 1, N2L \& XTEM = XLBAND(I) - XCG \& YTEM = YLBAND(I) - YCG

YLBDM(I) = YTEM \times CTH - XTEM \times STH - ZM \& XLBDM(I) = -(XTEM \times CTH + YTEM \times STH) - XM

IF (XLBDM(I) .LT. XMIN) XLBDM(I) = XMIN \& IF (XLBDM(I) .GT. XMAX) XLBDM(I) = XMAX

IF (YLBDM(I) .LT. YMIN) YLBDM(I) = YMIN \& IF (YLBDM(I) .GT. YMAX) YLBDM(I) = YMAX

30 CONTINUE \& DO 35 I = 1, N2T \& XTEM = XTET(I) - XCG \& YTEM = YTET(I) - YCG

YTMM(I) = YTEM \times CTH - XTEM \times STH - ZM \& XTTM(I) = -(XTEM \times CTH + YTEM \times STH) - XM

IF (XTTM(I) .LT. XMIN) XTTM(I) = XMIN \& IF (XTTM(I) .GT. XMAX) XTTM(I) = XMAX

IF (YTTM(I) .LT. YMIN) YTTM(I) = YMIN \& IF (YTTM(I) .GT. YMAX) YTTM(I) = YMAX

35 CONTINUE

SET ADJUSTED MINIMUM AND SCALE FACTOR, AND PLOT BALLOON OUTLINE BY CONNECTING POINTS

\[
X_{\text{BTM}(N2B+1)} = \text{XTFM}(NF+1) = X_{\text{LBDM}(NL8+1)} = X_{\text{TTM}(NT+1)} = X_{\text{MIN}}
\]

\[
Y_{\text{BTM}(N2B+1)} = \text{YFMT}(NF+1) = Y_{\text{LBDM}(NL8+1)} = Y_{\text{TTM}(NT+1)} = Y_{\text{MIN}}
\]

\[
X_{\text{BTM}(N2B+2)} = \text{XTFM}(NF+2) = X_{\text{LBDM}(NL8+2)} = X_{\text{TTM}(NT+2)} = \text{SCALE}
\]

\[
Y_{\text{BTM}(N2B+2)} = \text{YFMT}(NF+2) = Y_{\text{LBDM}(NL8+2)} = Y_{\text{TTM}(NT+2)} = \text{SCALE}
\]

CALL LINE(XBTM, YBTM, N2B, 1, 0, 1, *1)

CALL LINE(XFTM, YFTM, NF, 1, 0, 1, *1)

CALL LINE(XLBDM, YLBDM, NL8, 1, 0, 1, *1)

CALL LINE(XTTM, YTTM, NT, 1, 0, 1, *1)
DRAW X AND Z AXES FOR EQUILIBRIUM BALLOON POSITION

CALL LINE(XAXE,YAXE,2,1,0,1,1,1)
CALL LINE(XAXE(5),YAXE(5),2,1,0,1,1,1)

PLOT C G POSITION FOR TIME FRAME

40 XMG=-(XMIN-XM)*RSCLSYMG=-(YMIN+YM)*RSCLS CALL PNTPLT(XMG, YMG, 11,1)

SHIFT PLOT ORIGIN IF ISHIFT IS 0

IF(ISHIFT.EQ.0)CALL CALPLT(XSHIFT,0,0,3)
50 CONTINUE

SHIFT PLOT ORIGIN AND CHECK FOR NEXT CASE

CALL CALPLT((AXESL+XSHIFT),0,0,-3) $ GO TO 10

999 CALL CALPLT(0,0,0,999)
END PROGRAM CALBALM
Program CALBLM2 plots the lateral modes of motion in similar fashion to that described for the longitudinal program. The organization of this program is basically the same as that for the longitudinal program. However, the additional complication of treating the rolling motion about the stability axis requires the z-coordinates of the components to be given in addition to the x- and y-coordinates. The apparent shape of the balloon must also be altered as a result of the trim pitch angle. Roll displacements are treated as linearized displacements and the hidden portion of the top fin is deleted in approximate fashion. The eigenvector is normalized by yaw angle $\psi$. An example of a plot generated by this program is given in figure 6.

![Example plot from program CALBLM2.](image)

Figure 6.- Example of plot from program CALBLM2.

The four input-data cards are essentially the same as for the longitudinal program with the following exceptions: For Card 2, the third through eighth 10-column fields contain

- **YREAL**: real part of y-component of eigenvector normalized by $\psi$, m/deg
- **YIMAG**: imaginary part of y-component of eigenvector normalized by $\psi$, m/deg
- **PREAL**: real part of $\phi$-component of eigenvector normalized by $\psi$, nondimensional
- **PIMAG**: imaginary part of $\phi$-component of eigenvector normalized by $\psi$, nondimensional
- **SREAL**: real part of $\psi$-component of eigenvector, normally 1.0
- **SIMAG**: imaginary part of $\psi$-component of eigenvector, normally 0.0

For Card 3, the variable AMPL refers to the initial yaw angle $\psi$, in degrees.
Listing of Input Data Cards for Sample Case

LRC BALLOON - REFERENCE CONFIGURATION - MODE 2 AT V=20 M/S, ROOT=-.089+.407*1
-.088919 .40736 -.23446 -.80488 .003732 -.015992 1. 0.
12. 5.980 30.848 2.5
9 0 1 0

103
Listing of Program

OVERLAY(CALBLM2,O,0)
PROGRAM CALBLM2(OUTPUT=1, INPUT=1, TAPE5=INPUT)
C***********************************************************************
C* PROGRAM A264.6 - LATERAL EIGENVECTOR PLOTTING PROGRAM
C* PLOTS C & POSITION AND/OR BALLOON OUTLINE FOR A SINGLE MODE OF
C* MOTION FOR SELECTED INCREMENTS OF TIME
C* SEE SUBROUTINE WRITEUPS FOR DESCRIPTION OF CALPLT, NOTATE, LINE, OR
C* PNTPLT - LEROY SLOWS PLOTTER FOR DRAWING WITH INK
C*
C***********************************************************************
DIMENSION XBL(46), XFIN(16), XLBAND(46), XETE(16), XA(8)
C
+ ,DAY(2), YBL(46), YFIN(16), YLBAND(46), YETE(16), Y(8)
+ ,IC(18), XSTM(48), XTM(24), XLBDM(49), XTM(10), XTRIMM(14)
+ ,IN(2), YBTM(49), YTM(24), YLBDM(49), YTM(10), XTRIMM(14)
DATA A1/6/H(6101)/, F10/B(H(6101), 0.01, AD/10(/X10A101)/
DATA A14, A0/6/H(2041), 7/H(13A101)/
DATA GT, /125/, AXESL/10, I, ISH/70 /
DATA DEGR, 0.174532925199433 /
DATA FTPRM/3.2B0839895/ SI UNITS
C DATA FOR LRC BALLOON IN FEET
C BALLOON DATA IS FOR NOSE FACING LEFT - TURNED AROUND IN CALCULATIONS
C
DATA NB, NF, NLB, NT/23, 16, 23, 6 /
DATA (XBL(11)), 1=1,23/, 0., 07, 2., 33, 5., 67, 1., 1, 33, 1., 67, 2. 
C + 2.33, 2.67, 3.33, 4.67, 5.33, 6.67, 8.9, 33, 10.51, 25, /
DATA (YBL(11)), 1=1,23/, 0., 12., 45, 0.1, 33, 1.72, 2.03, 2.52, 2.88, 
C + 3.18, 3.42, 3.63, 3.77, 3.87, 5.95, 4.06, 4.14, 4.13, 3.99, 3.79, 
C + 3.57, 3.3, /
DATA ZBLA/46*0.0/
DATA YFIN/40, 2.07, 4.25, 33, 2.08, 4.0, -2.07, -4.25, -33, -2.07 /
DATA ZFIN/-2.07, -4.25, -33, -2.07, 4.0, 2.07, 4.25, 33, 2.07, 4.0 /
DATA (YLBAND(I)), 1=1,23/, 0., 12., 45, 95, 1, 33, 1.72, 2.03, 2.52, 2.88, 
C + 3.18, 3.42, 3.63, 3.77, 3.87, 5.95, 4.06, 4.14, 4.13, 3.99, 3.79, 
C + 3.57, 3.3, /
DATA ZLBAND/46*2.333333333333/3
DATA XETE/2.64, 4.13, 2.64, 13.79, 4.13, 13.79 /
DATA YETE/2.93, 0., -2.93, 1.62, 0., -1.62 /
DATA (ZETE), 2.32, 12.51, 2*2.32, 12.51, 2.32 /
DATA XMC/15.42/, YMC/O.1, XCG/14.32/, YCG/O.1, ZCG/29/
DATA XMIN, XMAX, YMIN, YMAX, -60., 50., 50., 50., /
C REFLECT BALLOON HALF-PROFILE, INITIALIZE PLOTTING ROUTINES, 
C SCALING PARAMETERS, AND SET ORIGIN FOR INITIAL FRAME
C
NLB1=NLB+1 $ NLB2=NLB+NLB $ DO 101 I=NLB1, NLB2
IB=NLS2+1 $ I=NLB1, NLB2 $ DO 101 I=NLB1, NLB2
101 YBLAN(I)=YLBAND(I)
NTU=2*NB+NF+2*NLB+NT $ NT2=NT/2 $ NF=NF/4
NR1=NB+1 $ NB=NB $ DO 1 I=NB1, NB2 $ IB=NB2+1 $ I=NB1, NB2 $ DO 102 I=IB$I=XBAL(I)=XBAL(I)
1 YBLA(I)=YBAL(I)
SCALE=(XMAX-XMIN)/AXESL $ RSCS=1./SCALE $ ZER=1.25
XAKE(1)=10., $ XAKE(2)=10. $ XAKE(3)=XAKE(7)=XMIN
XAKE(5)=XAKE(6)=0. $ XAKE(4)=XAKE(8)=SCALE
YAKE(1)=YAKE(2)=0. $ YAKE(3)=YAKE(7)=YMIN
YAKE(5)=YAKE(6)=SCALE $ YAKE(5)=10. $ YAKE(6)=10.
CALL CALCGMP $ CALL CALPTIZER, ZER, -3 $ CALL LEROY

104
READ IDENTIFICATION ARRAY, CALL FOR DATE AND TIME, AND PRINT AND WRITE AT THE BOTTOM OF THE FIRST FRAME

10 READ AI, ID $ IF (EOF, 51999, 11)
11 CALL DATEIM(DAY) $ PRINT AD, ID, DAY
CALL NOTATE(0, 0, HGT, ID, 0, 80)

READ EIGENVECTOR-MOTION DATA, TRIM PARAMETERS, AND FRAME DATA

READ F10, ROOTR, ROOTI, YREAL, YIMAG, PREAL, PIMAG, SREAL, SIMAG
READ F10, AMPL, TRIMA, TIME, XSHIFTS READ I4, NFRM, ISHIFT, MODBAL, INMOD
INMOD = INMOD + MODBAL

CONVERT DISPLACEMENT TO FEET, SCALE TIME, AND COMPUTE MAGNITUDES AND PHASES OF MODAL COMPONENTS

DT = .C625*TIME $ AMFL = DEGRAC*AMPL
DTR = DT*ROOTR $ TH = DEGRAD*TRIMA $ CTH = COS(TH) $ STH = SIN(TH)
YABS = SQRT(YREAL**2 + YIMAG**2) $ PHY = ATAN2(YIMAG, YREAL)
YABS = YABS $ PABS = SQRT(REAL**2 + PIMAG**2) $ PHIP = ATAN2(PIMAG, PREAL)
CPY = COS(PHY) $ SPY = SIN(PHY) $ CPP = COS(PHIP) $ SPP = SIN(PHIP)

ROTATE BALLOON IN PITCH TO TRIM ANGLE

DO 15 I = 1, NMOD $ XTEM = XBAL(I) - XCG $ ZTEM = ZBAL(I) - ZCG
XTRIM(I) = XTEM * CTH + ZTEM * STH
15 ZTRIM(I) = XTEM * STH + ZTEM * CTH
NB21 = NB21 + 1
ZTRIM(NB21) = ZTRIM(NB21) + (ZTRIM(NB21) - ZTRIM(NB21)
+ (ZTRIM(NB21) - ZTRIM(NB21)) * (ZTRIM(NB21) - ZTRIM(NB21))
XTRIM(NB21) = XTRIM(NB21) + (XTRIM(NB21) - XTRIM(NB21))
XTRIM(NB21) = XTRIM(NB21) + 31 = XTRIM(NB21)

50-LOOP IS TIME-FRAME LOOP

DO 50 NTS = 1, NFRM $ N = NTS - 1
CALCULATE MOTION FOR TIME FRAME

ST = SIN(N*DT) $ CT = COS(N*DT) $ AET = AMPL*EXP(N*DTR) $ SM = AET*CT
YM = AET*YABS*CT*CPY - ST*SPY $ PM = -AET*PABS*CT*CPP - ST*SPPI
CS = COS(SM) $ SS = SIN(SM) $ IF(ISHIFT.EQ.0) GO TO 18

IF (CDONL = CDONL, MODBAL) NE.0) GO TO 40

IF (CDONL = CDONL + MODBAL, NE.0) GO TO 40

20-LOOP PREPARES ARRAYS FOR BALLOON OUTLINE FOR PLOTTING

25-LOOP - FINS, 30-LOOP - LOAD BAND, AND 35-LOOP - TETHER BRIDLE
POINTS OFF-SCALE ARE SET TO OUTER LIMITS OF FRAME
SET ADJUSTED MINIMUM AND SCALE FACTOR, AND PLOT BALLOON OUTLINE BY CONNECTING POINTS

18 DO 20 I = 1, NB2 $ XTEM = XTRIM(I) $ YTEM = YBAL(I) - ZTRIM(I) * PM
XBTM(I) = XTEM * CS + YTEM * SS $ YBTM(I) = XTEM * SS - YTEM * CS - YM
IF (XBTM(I).LT.XMIN) XBTM(I) = XMIN $ IF (XBTM(I).GT.XMAX) XBTM(I) = XMAX
IF (YBTM(I).LT.YMIN) YBTM(I) = YMIN $ IF (YBTM(I).GT.YMAX) YBTM(I) = YMAX
20 CONTINUE

XBTM(NB2+1) = XMIN $ XBTM(NB2+2) = SCALE
YBTM(NB2+1) = YMIN $ YBTM(NB2+2) = SCALE
CALL LINE(XBTM, YBTM, NB2, 1, 0, 1, 1)
DO 28 IF=1,4 $ K=1+([IF-1]*NF4
DO 25 J=1,NF4 $ I=J*K+2*([IF-1]-1 $ II=NB2+J*K-1
XTEM=XTRIM(I) $ YTEM=YBAL(I)-ZTRIM(I)*PM
XFTM(I)=XTEM*CS+YTEM*SS $ YFTM(I)=XTEM*SS-YTEM*CS-YM
IF(XFTM(I),LT,XMIN)XFTM(I)=XMIN $ IF(XFTM(I),GT,XMAX)XFTM(I)=XMAX
IF(YFTM(I),LT,YMIN)YFTM(I)=YMIN $ IF(YFTM(I),GT,YMAX)YFTM(I)=YMAX
25 CONTINUE
XFTM(I+1)=XMIN $ XFTM(I+2)=SCALE
YFTM(I+1)=YMIN $ YFTM(I+2)=SCALE $ KI=K+2*([IF-1]
28 CALL LINE(XFTM(KI),YFTM(KI),NF4,1,0,1,1)
DO 30 I=1,NLB2 $ II=I+NB2+NF
XTEM=XTRIM(I) $ YTEM=YBAL(I)-ZTRIM(I)*PM
XLBDM(I)=XTEM*CS+YTEM*SS $ YLBDM(I)=XTEM*SS-YTEM*CS-YM
IF(XLBDM(I),LT,XMIN)XLBDM(I)=XMIN
IF(YLBDM(I),LT,YMIN)YLBDM(I)=YMIN
IF(XLBDM(I),GT,XMAX)XLBDM(I)=XMAX
IF(YLBDM(I),GT,YMAX)YLBDM(I)=YMAX
30 CONTINUE
XLBDM(NLB2+1)=XMIN $ XLBDM(NLB2+2)=SCALE
YLBDM(NLB2+1)=YMIN $ YLBDM(NLB2+2)=SCALE
CALL LINE(XLBDM,YLBDM,NLB2,1,0,1,1)
DO 38 IT=1,2 $ K=1+([IT-1]*NT2
DO 35 J=1,NT2 $ I=J*K+2*([IT-1]-1 $ II=NB2+NF+NB2+J*K-1
XTEM=XTRIM(I) $ YTEM=YBAL(I)-ZTRIM(I)*PM
XFTM(I)=XTEM*CS+YTEM*SS $ YFTM(I)=XTEM*SS-YTEM*CS-YM
IF(XFTM(I),LT,XMIN)XFTM(I)=XMIN $ IF(XFTM(I),GT,XMAX)XFTM(I)=XMAX
IF(YFTM(I),LT,YMIN)YFTM(I)=YMIN $ IF(YFTM(I),GT,YMAX)YFTM(I)=YMAX
35 CONTINUE
XFTM(I+1)=XMIN $ XFTM(I+2)=SCALE
YFTM(I+1)=YMIN $ YFTM(I+2)=SCALE $ KI=K+2*([IT-1]
38 CALL LINE(XFTM(KI),YFTM(KI),NT2,1,0,1,1)
C C DRAW X AND Y AXES FOR EQUILIBRIUM BALLOON POSITION
C CALL LINE(XAXE,YAXE,2,1,0,1,1)
CALL LINE(XAXE(5),YAXE(5),2,1,0,1,1)
C C FLCT C G POSITION FOR TIME FRAME
C 40 XMG=-XMIN*RSCL $ YMG=-YMIN*YM*RSCL $ CALL PNTPLT(XMG,YMG,11,1)
C SHIFT PLOT ORIGIN IF ISHIFT IS 0
C IF(ISHIFT.EQ.0)CALL CALPLT(XSHIFT,0..,-3)
50 CONTINUE
C SHIFT PLOT ORIGIN AND CHECK FOR NEXT CASE
C CALL CALPLT((AXESL+XSHIFT),0..,-3) $ GO TO 10
C 999 CALL CALPLT(0..,0..,999)
END PROGRAM CALBLM2

Langley Research Center,
National Aeronautics and Space Administration,
APPENDIX

DESCRIPTIONS OF SELECTED SUBROUTINES

Basic Subroutines

Usage descriptions are given for several of the basic subroutines called by the programs of this report. Subroutines QUADET and ROMBERG were written by the authors. The versions of REIG and MATRIX given herein are modified versions of the LRC computer system library. The subprograms CXINV, DAYTIM, REcin, and REcout are LRC computer system library subroutines. Note that listings of QUADET, ROMBERG, REIG, MATRIX, and CXINV are given in the listing of the longitudinal program STABILITY. The listings of the REcin, REcout, and DAYTIM are not given, but the usage descriptions are given to facilitate replacement with equivalent routines if necessary.

In addition to the above subprograms, a FORTRAN subroutine to simulate the COMPASS subroutine MASCNT is described and listed. The subroutine MATINV which is called by MASCNT (simulator) is also described and listed.

The subroutines are described in the following order:

<table>
<thead>
<tr>
<th>Subroutine</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>QUADET</td>
<td>108</td>
</tr>
<tr>
<td>ROMBERG</td>
<td>111</td>
</tr>
<tr>
<td>REIG</td>
<td>113</td>
</tr>
<tr>
<td>MATRIX</td>
<td>115</td>
</tr>
<tr>
<td>CXINV</td>
<td>117</td>
</tr>
<tr>
<td>REcin</td>
<td>119</td>
</tr>
<tr>
<td>REcout</td>
<td>121</td>
</tr>
<tr>
<td>DAYTIM</td>
<td>124</td>
</tr>
<tr>
<td>MASCNT</td>
<td>126</td>
</tr>
<tr>
<td>MATINV</td>
<td>128</td>
</tr>
</tbody>
</table>
APPENDIX – Continued

Subroutine QUADET

LANGUAGE: FORTRAN and COMPASS

PURPOSE: To convert an \( n \times n \) matrix equation, \( [A] \lambda^2 + [B] \lambda + [C] = 0 \),
to a \( 2n \times 2n \) matrix equation of standard eigenvalue form,
\( [E] - \lambda[I] = 0 \).

USE: CALL QUADET (A, B, C, NMAX, NMAX2, N, IOP, EIGDET, CNO)

A A two-dimensional input array containing the coefficients of \( \lambda^2 \). The matrix is not destroyed.

B A two-dimensional input array containing the coefficients of \( \lambda \). The matrix is not destroyed.

C A two-dimensional input array containing constant coefficients. The matrix is not destroyed.

NMAX Column length (number of rows) of A as dimensioned in the calling program.

NMAX2 Column length (number of rows) of EIGDET as dimensioned in the calling program.

N The order of A, B, and C.

IOP An integer, 10 or 11 supplied by user to select option for calculating the condition number of A:
IOP = 10 Condition number of A is calculated.
IOP = 11 Condition number of A is not calculated.

EIGDET The two-dimensional eigenvalue output array.

CNO CNO is the Turing condition number of A. If 0.0 is returned from MASCNT for the determinant of A (singular A), CNO = -1.

RESTRICTIONS: Arrays A, B, C, and EIGDET are used with variable dimensions in the subroutine. The maximum size in the calling program must be A (NMAX, N), B (NMAX, N), C (NMAX, N), EIGDET (NMAX2, NMAX2); NMAX2 \( \geq 2 \times NMAX \). A must be nonsingular. Restricted to real arrays.
APPENDIX – Continued

METHOD:

The \( n \times n \) system \( [A] \lambda^2 + [B] \lambda + [C] = 0 \) is equivalent to the \( 2n \times 2n \) system (refs. (a) and (b)). \( \begin{bmatrix} \lambda \end{bmatrix} - \lambda \begin{bmatrix} I \end{bmatrix} = 0 \), where

\[
\begin{bmatrix}
-A^{-1}B & -A^{-1}C \\
\vdots & \vdots \\
I & 0
\end{bmatrix}
\]

QUADET performs these matrix manipulations within the storage allocated for EIGDET. The matrix solutions are obtained from a matrix routine written in COMPASS (ref. (c)). The arrays A, B, C, and EIGDET are treated as one-dimensional arrays. A, B, C are transferred into locations \((NMAX2**2-3*N*N)\) through \((NMAX2**2)\) of EIGDET. MASCNT is called and \(A^{-1}B\) is returned in the location of \(B\) and \(A^{-1}C\) is returned in the location of \(C\) in EIGDET. If \(IOP = 10\), \(A^{-1}\) is returned in \(A\) and Turing's condition number defined as

\[
C_A = \frac{\|A\| \cdot \|A^{-1}\|}{N}
\]

is calculated. The matrices \([A^{-1}B]\), \([A^{-1}C]\), \([I]\), and \([0]\) are then transferred to the upper left \(2n \times 2n\) portion of EIGDET and the remainder of EIGDET filled with indefinites \((\phi 1777 0000 0000 0000 1777)\) such that EIGDET is returned as

\[
\begin{bmatrix}
\begin{array}{ccc}
\vdots & \vdots \\
-A^{-1}B & -A^{-1}C \\
\vdots & \vdots \\
I & 0
\end{array}
\end{bmatrix}
\]

\[
\begin{array}{cc}
\text{INDEF.} \\
\text{INDEF.}
\end{array}
\]

\[
\begin{array}{c}
\text{NMAX2} \\
N \\
N
\end{array}
\]
The number of figures lost in the operations $A^{-1}B$ and $A^{-1}C$ can be estimated as $\log_{10}(CNO)$ where $CNO$ is condition number of $A$.


QUADET 3338
MASCNT 4318
Labeled COMMON 11308 - /IROW/IROW(300)
/ICOL/ICOL(300)

SUBPROGRAM USED: MASCNT
Subroutine ROMBERG

LANGUAGE: FORTRAN

PURPOSE: To integrate the function FUN(X) between the limits A and B.

USE: CALL ROMBERG (SUM, A, B, FUN, EPS)

SUM The computed value of the integral.
A The lower limit of integration.
B The upper limit of integration.
FUN The name of the integrand function.
EPS Relative error criterion.

An example of the usage follows. A segment of the program to evaluate \( \int_a^b \cos(t^2 + p) \, dt \) might be:

EXTERNAL FUN
COMMON PHI
PHI=.33
A=.1
B=.275
EPS=1.E-6
CALL ROMBERG (SUM, A, B, FUN, EPS)
.
.
END

The function subprogram would be:

FUNCTION FUN(T)
COMMON PHI
FUN = COS(T * T + PHI)
RETURN
END

RESTRICTIONS: A function subprogram with a single argument must be written by the user to evaluate the integrand FUN(X). Since ROMBERG requires that its integrand be a function of one argument only, any variable parameters of the integrand must be passed to the function subprogram through COMMON. The name of the function must appear in an EXTERNAL statement in the calling program. (See example under USE.)
This subroutine is taken with minor modifications from p. 199 of the reference. The method is described on pp. 166 to 170 of the reference. ROMBERG integration is a so-called automatic method in that the routine normally returns a value of the integral within the prescribed accuracy (see ACCURACY, below).

Normally, iteration proceeds until

\[ |\text{SUM}_i - \text{SUM}_{i-1}| < \text{EPS} \cdot |B-A| \cdot \max_{X \in [A,B]} |\text{FUN}(X)| \]

The subroutine is dimensioned such that if the accuracy criterion is not satisfied after 19 steps (262,144 integrand evaluations) the best estimate of the integral at that point is returned and no error message is given.


2548 locations
Subroutine REIG

FORTRAN

To find the eigenvalues of a real matrix.

CALL REIG (A, N, NVAL, NVEC, RTR, RTI, VEC, NMAX, INDEX, IRUN, P, NPLUS, SAVE)

A A two-dimensional array containing the input matrix.

N The order of A; 1 ≤ N ≤ NMAX.

NVAL The number of eigenvalues desired.

NVEC Dummy parameter; not used.

RTR A one-dimensional array in which the real parts of the eigenvalues are stored. The real eigenvalues are stored first, and are sorted by magnitude in decreasing order.

RTI A one-dimensional array in which the imaginary parts of the eigenvalues are stored.

VEC Dummy parameter; not used.

NMAX The maximum order of A as stated in the dimension statement of the calling program.

INDEX A one-dimensional array of temporary storage.

IRUN A one-dimensional array of temporary storage.

P A one-dimensional array of temporary storage.

NPLUS The order of A plus one; i.e., NPLUS = N + 1.

SAVE A two-dimensional array of temporary storage.

The following arrays must be dimensioned in the calling program as indicated: A (NMAX, NMAX), RTR (NMAX), RTI (NMAX), INDEX (NMAX), IRUN (NMAX), P (NMAX), SAVE (NMAX, NMAX + 1). N is limited to 100. The input matrix is not destroyed.
The original matrix is transformed to upper Hessenberg form. Then the eigenvalues are found using the QR transform of J. G. F. Francis (ref. (a)).

Accuracy depends on the conditioning of A (ref. (c)).


REIG 21678 locations including QRT and HESSEN.

The following subprograms are used by REIG:

QRT 5408 locations
HESSEN 4318 locations
Subroutine MATRIX (Modified)

FORTRAN AND COMPASS

Shortened version of CDC subroutine MATRIX (CDC publication No. 60135200) for a comprehensive group of matrix operations. This version deletes the real symmetric eigenvalue/eigenvector options to conserve storage. In addition, Turing's condition number is calculated for option 10.

CALL MATRIX (I, M, N, K, A, KA, B, KB, C, KC).

<table>
<thead>
<tr>
<th>I - option code</th>
<th>Matrix operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Transpose A into B (B ≠ A)</td>
</tr>
<tr>
<td>1</td>
<td>Move A into B</td>
</tr>
<tr>
<td>2</td>
<td>Symmetric product (A^T \cdot A = B), B packed</td>
</tr>
<tr>
<td>3</td>
<td>Deleted</td>
</tr>
<tr>
<td>4</td>
<td>Pack symmetric matrix A into B</td>
</tr>
<tr>
<td>5</td>
<td>Unpack symmetric matrix A into B</td>
</tr>
<tr>
<td>10</td>
<td>Solve (D X = E) with (D^{-1}) returned, (A) upon call = ([D \mid E]) (A) upon return = ([D^{-1} \mid X])</td>
</tr>
<tr>
<td>11</td>
<td>Same as 10 except calculation of (D^{-1}) deleted</td>
</tr>
<tr>
<td>20</td>
<td>Multiply (A \cdot B = C); (C = A) or (B) is permissible.</td>
</tr>
<tr>
<td>21</td>
<td>Add (A + B = C)</td>
</tr>
<tr>
<td>22</td>
<td>Subtract (A - B = C)</td>
</tr>
<tr>
<td>23</td>
<td>Transpose multiply (A^T \cdot B = C)</td>
</tr>
<tr>
<td>24</td>
<td>Scalar multiply (A \cdot B = C); (A = ) Scalar</td>
</tr>
</tbody>
</table>

M  Number of rows of the matrix A.

N  Number of columns of the matrix A.

K  Unused, = 0, except for \(I = 10, 11, 20,\) and 23.
   For \(I = 10\) and 11, \(K = \) pivoting parameter
   \(K = 0\), full search each pass
   \(K = 1\), search Jth row on Jth pass
   \(K = 2\), no pivoting, use diagonal elements
   For \(I = 20\) and 23, \(K = \) number of columns of matrices B and C.

A  Matrix A.
APPENDIX — Continued

KA  Column size of matrix A.
B  Matrix B, or the determinant of D for \( I = 10 \).
KB  Column size of matrix B.
C  Matrix C, or Turing's condition number for \( I = 10 \).
KC  Column size of matrix C.

RESTRICTIONS:  For description of restrictions, see reference. However, it may be noted that maximum number of rows or columns is 300.

Maximum dimensions in calling program must be consistent with operations. For options 0 and 1, \( KA \geq M, KB \geq N \); for options 10 and 11, \( N \geq M, KA \geq M \); for options 21, 22, and 23, \( KA \geq M, KB \geq M \); for options 21 and 22, \( KC \geq M \); for option 23, \( KA \geq M \). Note that for options 10 and 11, matrix A must contain D and E of \( DX = E \) in adjacent columns.

Option 3 has been deleted for this version.

METHOD:  The CDC subroutine MATRIX is a FORTRAN subroutine that computes the eigenvalues and eigenvectors of a real symmetric matrix for option 3. For all other options, the subprogram only sets up the call to a COMPASS subroutine, MASCNT, to perform the matrix operations. In this version the lengthy FORTRAN eigenvalue/eigenvector section has been deleted to conserve storage, and the calculation of Turing's condition number has been added for the matrix-inversion option. See reference for further discussion of methods.

ACCURACY:  Applicability depends on matrix operation. For matrix inversion and simultaneous linear equations, the loss of significant figures can be estimated from \( \log_{10}(CNO) \) where CNO is Turing's condition number defined as:

\[
CNO = \frac{\| A \| \cdot \| A^{-1} \|}{N}
\]


STORAGE:  

<table>
<thead>
<tr>
<th>Subroutine</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>MATRIX (Mod.)</td>
<td>132_8</td>
</tr>
<tr>
<td>MASCNT</td>
<td>431_8</td>
</tr>
<tr>
<td>LABELED COMMON</td>
<td>1130_8</td>
</tr>
<tr>
<td></td>
<td>/IROW/IROW(300)</td>
</tr>
<tr>
<td></td>
<td>/ICOL/ICOL(300)</td>
</tr>
</tbody>
</table>

116
Subroutine CXINV

LANGUAGE: FORTRAN

PURPOSE: To solve the complex matrix equation \( AX = B \) where \( A \) is a square complex coefficient matrix and \( B \) is a complex matrix of constant vectors. The solution to a set of simultaneous equations, the matrix inverse, and the determinant may be obtained.

USE: CALL CXINV (A, N, B, M, DETERM, IPIVOT, INDEX, MAX, ISCALE)

A A two-dimensional complex array of the coefficients. On return to the calling program, \( A \) contains the matrix inverse.

N The order of \( A \); \( 1 \leq N \leq MAX \)

B A two-dimensional complex array of the constant vectors \( B \). On return to the calling program, \( B \) contains the \( X \) values.

M The number of column vectors in \( B \). If \( M = 0 \), there is no solution of the simultaneous equations; however, there must be an entry for \( B \) in the call statement.

DETERM Gives the complex value of the determinant by the formula:
\[
\text{DET}(A) = (10^{100})^{\text{ISCALE}}(\text{DETERM})
\]

IPIVOT A one-dimensional integer array of temporary storage.

INDEX A two-dimensional integer array of temporary storage.

MAX The maximum order of \( A \) as stated in the dimension statement of the calling program.

ISCALE A scale factor computed by the subroutine to keep the results of computation within the floating point word size of the computer.
The calling program must dimension arrays as indicated:
A (MAX, MAX), B (MAX, M), IPIVOT (MAX), INDEX (MAX, 2).
It must also type A, B, DETERM as COMPLEX.

The input matrices A and B are destroyed. On return to the
calling program the inverse of A is in A and X is in B.

Jordan's method is used to reduce the matrix A to the identity
matrix I through a succession of elementary transformations:
l_n, l_{n-1}, \ldots, l_1 * A = I. If these transformations are simul­
taneously applied to I and to the matrix B of constant vectors,
the results are A^{-1} and X where \ AX = B.

Each transformation is selected so that the largest element is
used in the pivotal position.

Total pivotal strategy is used to minimize the rounding errors;
however, the accuracy of the final results depends upon how
well-conditioned the original matrix is.

Fox, L.: An Introduction to Numerical Linear Algebra. Oxford

7218 locations

Library: CABS
APPENDIX – Continued

Subroutine RECIN

LANGUAGE: COMPASS

PURPOSE: To read binary records written by the subroutine RECOUT.

USE:

1. Type 1 – Individual elements (not arrays)
   CALL RECIN (LUN, IT, ICOUNT, L1, L2, ..., LN) where
   LUN = logical unit number
   IT = type = 1
   ICOUNT = location reserved by the user. RECIN will store the following information in this location:
   0 = end-of-file; nonzero = number of words actually in the logical record. If the end-of-file flag was written by a call to RECOUT with IEOF = 1, then end-of-file testing must be done by testing ICOUNT for 0. If the end-of-file was written by an END FILE statement, then testing for end-of-file must be done by the IF (EOF, LUN) statement.
   L1, L2, ..., LN = individual list elements.

2. Type 2 – Arrays
   CALL RECIN (LUN, IT, ICOUNT, ARRAY, IFIRST, ILAST, INC) where
   LUN = logical unit number
   IT = type = 2
   ICOUNT = 0 = end-of-file; nonzero = number of words actually in the logical record (See ICOUNT under type 1)
   ARRAY = array name
   IFIRST = first subscript
   ILAST = last subscript
   INC = increment

EXAMPLES:

1. CALL RECIN (1, 1, K, A, B, ARRAY(1), ARRAY(2))
   Read a record from logical unit 1 into A, B, ARRAY(1) and Array (2). Note that if the record contained only 3 words, K would equal 3 and ARRAY(2) would be unaltered.
APPENDIX – Continued

2. CALL RECIN (1, 2, K, ARRAY, 1, 39, 2)
   Read 20 words from logical unit 1 into
   ARRAY(1), ARRAY(3), ..., ARRAY(39).

   **RESTRICTIONS:**
   If RECIN is used on a file, the only other FORTRAN statements
   which may be used on that file are REWIND and IF (EOF, i).
   The buffer size must be at least 20018.
   RECIN must be used to read files written by RECOUT and only
   by RECOUT.

   **METHOD:**
   RECIN reads into a central memory buffer physical records
   written by RECOUT, then passes to the user the requested
   logical record via a list giving the elements of the desired
   logical record. RECIN is analogous to a FORTRAN binary
   READ statement.

   **ACCURACY:**
   Not applicable.

   **REFERENCE:**
   None.

   **STORAGE:**
   3158 locations

   **OTHER CODING INFORMATION:**
   Day file diagnostics and their meaning:
   (1) UNASSIGNED FILE MEDIUM FILE TAPE<nn> – No FET
   exists for this file. Every file has a file environment table
   that contains information describing the file to the system.
   This error would probably be caused by the file not being
   defined in the PROGRAM card or the user accidentally over­
   writing portions of his program.
   (2) BAD TYPE – The IT parameter was not 1 or 2.
   (3) UNCHECKED END FILE – The program attempted to read
   past EOF without testing for EOF.
   (4) READ/WRITE SEQUENCE ERROR – An attempt was made
   to read after writing.
Subroutine RECOUT

LANGUAGE: COMPASS

PURPOSE: To write short binary records on a disk or tape in an optimum manner to increase peripheral processor and central processor efficiency. These records are to be read by RECIN.

USE: RECOUT may be used for either tape or disk files.

1. Type 1 – Individual elements (not arrays)
   CALL RECOUT (LUN; IT; IEOF; L1, L2, . . . , LN) where
   
   LUN = logical unit number
   IT = type = 1
   IEOF = 1 if an end-of-file flag is desired, otherwise it must be zero. There are two methods by which the user may end his file. One method is to call RECOUT with IEOF = 1 when the last data record is written. This will cause an end-of-file flag (a short length record of less than 51210 CM words) to be written. RECIN is programed to sense this and will set ICOUNT = 0 when sensed. If this method is used, the user must set IEOF = 1 when outputting his last data record since RECOUT should not be called with an empty list. For all other calls to RECOUT, IEOF must be set = 0. The other method of ending the file is to use the END FILE statement. This is the most convenient way of ending the file.
   
   L1, L2, . . . , LN = individual list elements.

2. Type 2 – Arrays
   CALL RECOUT (LUN, IT, IEOF, ARRAY, IFIRST, ILAST, INC) where
   
   LUN = logical unit number
   IT = type = 2
   IEOF = 1 if an end-of-file desired
        = 0 no end-of-file
        See explanation under type 1.
   
   ARRAY = array name
APPENDIX – Continued

IFIRST = first subscript
ILAST = last subscript
INC = increment

EXAMPLES:

(1) CALL RECOUT (1, 1, 0, A, B, ARRAY(1), ARRAY(2))
    Write a record on logical unit 1 containing A, B, ARRAY(1),
    ARRAY(2).

(2) CALL RECOUT (1, 2, 0, ARRAY, 1, 20, 1)
    Write a record containing ARRAY(1) through ARRAY(20). This
    is equivalent to WRITE(1) (ARRAY(I), I = 1, 20).

RESTRICTIONS:

If RECOUT is used on a file, the only other FORTRAN state­
ments which may be used on that file areREWIND and END
FILE.

The buffer size must be at least 20018. A normal FORTRAN
buffer is this size.

FILES written with RECOUT must be read with RECIN.

If the list to be written in a logical record is larger than
51110 CM words, then RECOUT offers no advantage and should
not be used.

If the programer wishes to write a file containing multifiles
using RECOUT, then he must end each file by setting IEOF = 1
and not by using the END FILE statement. Consequently, he
should then test for end-of-file in RECIN by testing ICOUNT
for zero.

METHOD:

Under the CDC SCOPE 3.0 operating system each binary write
commanded by the FORTRAN statement WRITE (LUN) . . .
causes one or more physical records to be output to either a
disk or tape file. If the logical record size written by the
programer is small and the number of records processed is
large, then excessive usage of I/O routines and equipment
results. To decrease this I/O time, RECOUT blocks binary
data into an optimum record size (51210 CM words) in a central
memory buffer before transmitting it to the actual disk or tape
file.

ACCURACY: Not applicable.

REFERENCE: None.
APPENDIX – Continued

STORAGE: 3408 locations

OTHER CODING INFORMATION:

Day file diagnostics and their meaning:

(1) UNASSIGNED FILE MEDIUM FILE TAPE<nn> – No FET exists for the file. Every file has a file environment table that contains information describing the file to the system. This error would probably be caused by the file not being defined in the PROGRAM card or the user accidentally overwriting portions of his program.

(2) BAD TYPE – The IT parameter was not 1 or 2.

(3) BUFFER TOO SMALL – The buffer size was less than 20018.

(4) BAD PARAM COUNT – The number of parameters in the call was illegal.

(5) WRITE/READ SEQUENCE ERROR – A write request was made after a read request.
APPENDIX – Continued

Subroutine DAYTIM

LANGUAGE: COMPASS

PURPOSE: The purpose of this subroutine is to provide the current date and time of day to a central memory program.

USE: The subroutine may be called from a FORTRAN program using the following sequence:

CALL DAYTIM (RESULT, JUDATE)

RESULT = A single subscripted array dimensioned by two (2). The current date will be returned in RESULT(1) and the current time of day will be returned in RESULT(2).

JUDATE = An optional parameter. JUDATE need not be supplied. If it is, the Julian date will be returned in the memory cell named JUDATE. If JUDATE is not specified, then the Julian date will not be returned.

The date, the time, and the Julian date will be in display code in the following formats:

DATE: b MM/DD/YY

TIME: b HH.MM.SS

JULIAN DATE: bbb YY DDD bb

Each value may be printed using an A10 format in a FORTRAN program.
RESTRICTIONS: Whenever the 6000 system is deadstarted, the operators key in the date and the current time of day. It is the date and the current time of day based on the initial date and time of day keyed in by the operator that are returned to the calling program.

METHOD: The subroutine uses the macros supplied by CDC with Scope 3.0 to obtain the current date, time of day, and Julian date.

STORAGE: 318
APPENDIX – Continued

Subroutine MASCNT (Simulator)

LANGUAGE: FORTRAN

PURPOSE: Simulate the COMPASS language subroutine MASCNT with a FORTRAN subroutine for use with options 10 and 11 of FORTRAN subroutine MATRIX (options 10 and 11 are for matrix inverse and solutions of linear equations, $DX = E$).

USE:

CALL MASCNT (KR, A, B, C)

KR An integer array that contains seven of the formal parameters of subroutine MATRIX. The usage for this version of MASCNT is:

<table>
<thead>
<tr>
<th>MATRIX parameter</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>KR(1) I</td>
<td>Unused option parameter; I of 10 of MATRIX is assumed</td>
</tr>
<tr>
<td>KR(2) M</td>
<td>Number of rows of A</td>
</tr>
<tr>
<td>KR(3) N</td>
<td>Number of columns of A</td>
</tr>
<tr>
<td>KR(4) K</td>
<td>Unused pivoting parameter; full pivoting used by MATINV</td>
</tr>
<tr>
<td>KR(5) KA</td>
<td>Maximum number of rows of A as dimensioned in calling program</td>
</tr>
<tr>
<td>KR(6) KB</td>
<td>Unused</td>
</tr>
<tr>
<td>KR(7) KC</td>
<td>Unused</td>
</tr>
</tbody>
</table>

A A single two-dimensional array containing the two-dimensional matrix of coefficients and the two-dimensional matrix of column vectors. For $DX = E$, the first M columns of A must contain D and the adjacent (M + 1) to N columns must contain E. Upon return to the calling program, the locations of E contain the solution vectors, D is destroyed and the inverse of D is returned in D.

B The determinant of D.

C Unused.
APPENDIX - Continued

RESTRICTIONS: The array A must be dimensioned in the calling program as:
A (KA, MAX), MAX ≥ N. The original matrix D is destroyed.
Although this version of MASCNT can be used with MATRIX option 11, the inverse of D is always obtained. N ≤ 150.

METHOD: Subprogram sets up a call to FORTRAN subroutine MATINV to perform the matrix operations (see description of MATINV). Common blocks IROW and ICOL are used for the temporary storage required by MATINV.

ACCURACY: See description of subroutine MATINV.


STORAGE: MASCNT (Sim.) 668
MATINV 5428
Labeled COMMON 11308 - /IROW/IROW(300)
/ICOL/ICOL(300)

SUBPROGRAM USED: MATINV

SUBROUTINE MASCNT(KR,A,B,C)
C FORTRAN SIMULATOR FOR CDC MATRIX PACKAGE COMPASS LANGUAGE SUBROUTINE
C MATRIX OPTIONS 10 AND 11 ONLY
C
COMMON/IROW/IROW(300)/ICOL/ICOL(300)
DIMENSION KR(5), A(1)
NS=1+KR(2)*KR(5)
M=KR(3)-KR(2)
CALL MATINV(A,KR(2),A(NS),M,B,IROW,ICOL,KR(5),IS)
B=10**(100*IS)*B
RETURN
END SUBROUTINE MASCNT
APPENDIX – Continued

Subroutine MATINV

LANGUAGE: FORTRAN

PURPOSE: MATINV solves the matrix equation $AX = B$ where $A$ is a square coefficient matrix and $B$ is a matrix of constant vectors. The solution to a set of simultaneous equations, the matrix inverse, and the determinant may be obtained. If the user does not want the inverse, use SIMEQ for savings in time and storage. For the determinant only, use DETEV.

USE:

CALL MATINV (A, N, B, M, DETERM, IPIVOT, INDEX, NMAX, ISCALE)

A A two-dimensional array of the coefficients. On return to the calling program, $A^{-1}$ is stored in A.

N The order of $A; \ 1 \leq N \leq NMAX$

B A two-dimensional array of the constant vectors $B$. On return to calling program, $X$ is stored in $B$.

M The number of column vectors in $B$. $M = 0$ signals that the subroutine is used solely for inversion; however, in the call statement an entry corresponding to $B$ must still be present.

DETERM Gives the value of the determinant by the following formula:

$$\text{DET}(A) = (10^{100})^\text{ISCALE} \cdot \text{DETERM}$$

IPIVOT A one-dimensional array of temporary storage used by the routine.

INDEX A two-dimensional array of temporary storage used by the routine.

NMAX The maximum order of $A$ as stated in the dimension statement of the calling program.

ISCALE A scale factor computed by the subroutine to keep the results of computation within the floating point word size of the computer.
Arrays A, B, IPIVOT, and INDEX are dimensioned with variable dimensions in the subroutine. The maximum size of these arrays must be specified in a dimension statement of the calling program as: A (NMAX, NMAX), B (NMAX, M), IPIVOT (NMAX), INDEX (NMAX, 2). The original matrices, A and B, are destroyed. They must be saved by the user if there is further need for them.

The determinant is set to zero for a singular matrix.

Jordan's method is used to reduce a matrix A to the identity matrix I through a succession of elementary transformations: \( l_n l_{n-1} \cdots l_1 A = I \). If these transformations are simultaneously applied to I and to a matrix B of constant vectors, the results are \( A^{-1} \) and X where \( AX = B \). Each transformation is selected so that the largest element is used in the pivotal position.

Total pivotal strategy is used to minimize the rounding errors; however, the accuracy of the final results depends upon how well-conditioned the original matrix is.


5428 locations.
APPENDIX – Continued

SUBROUTINE MATINV(A,N,M,DETERM,PIVOT,INDEX,Amax,SCALE)

MATRIX INVERSION WITH ACCOMPANYING SOLUTION OF LINEAR EQUATIONS

DIMENSION PIVOT(N),AINMAX,M,BINMAX,M,INDEX(NMAX,2)
EQUIVALENCE (IRCW,IPROW), (ICOLUM,ICLUM), (AMAX, T, SWAP)

INITIALIZATION

5 ISCALE=0
6 RI=10.0**100
7 R2=1.0/R1
10 DETERM=1.0
15 DO 20 J=1,N
20 PIVOT(J)=0
30 DO 550 I=1,N

SEARCH FOR PIVOT ELEMENT

40 AMAX=0.0
45 DO 105 J=1,N
50 IF (PIVOT(J)-1) 60, 105, 60
60 DO 100 K=1,N
70 IF (PIVOT(K)-1) 80, 100, 740
80 IF (ABS(A(J,K))-ABS(A(I,I)))<EPS,100,100
85 IROW=J
90 ICOLUMN=K
95 AMAX=A(J,K)
100 CONTINUE
105 CONTINUE

IF (AMAX) 110,106,110
106 DETERM=0.0
107 ISCALE=0
108 GO TO 740
110 PIVOT(ICOLUMN)=PIVOT(ICOLUMN)+1

INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL

130 IF (IROW-ICOLUMN) 140, 260, 140
140 DETERM=-DETERM
150 DO 200 L=1,N
160 SWAP=A(IROW,L)
170 A(IROW,L)=A(ICOLUMN,L)
180 A(ICOLUMN,L)=SWAP
205 IF (L) 260, 260, 210
210 DO 250 L=1,M
220 SWAP=B(IROW,L)
230 B(IROW,L)=B(ICOLUMN,L)
250 B(ICOLUMN,L)=SWAP
260 INDEX(1,1)=IROW
270 INDEX(1,2)=ICOLUMN
310 PIVOT=A(ICOLUMN,ICOLUMN)

SCALE THE DETERMINANT

1000 PIVOT=PIVOT
1005 IF (ABS(DETERM)-R1) 1030, 1010, 1010
1010 DETERM=DETERM/R1
1015 ISCALE=ISCALE+1
1020 DETERM=DETERM/R1

130
APPENDIX - Continued

ISCALE = ISCALE + 1
GO TO 1060
1030 IF (ABS (DETERM) - R1) >= 1040, 1040, 1060
1040 DETERM = DETERM * R1
ISCALE = ISCALE - 1
IF (ABS (DETERM) - R1) >= 1050, 1050, 1060
1050 DETERM = DETERM * R1
ISCALE = ISCALE - 1
1060 IF (ABS (PIVOTI) - R1) >= 1070, 1070, 1080
1070 PIVCTI = PIVUTI / R1
ISCALE = ISCALE + 1
IF (ABS (PIVOTI) - R1) >= 1080, 1080, 1090
1090 PIVCTI = PIVUTI / R1
ISCALE = ISCALE - 1
1100 IF (ABS (PIVOTI) - R2) >= 1110, 1110, 1120
1110 PIVCTI = PIVUTI / R1
ISCALE = ISCALE - 1
1120 IF (ABS (PIVOTI) - R2) >= 1130, 1130, 1140
1130 PIVCTI = PIVUTI / R1
ISCALE = ISCALE - 1
320 IF (ABS (PIVOTI) - R2) >= 330, 330, 340
330 AICCLUD, 1CCLUD) = 1.0
340 DO 350 L = 1, N
350 AICCLUD, L) = A1CCLUD, L) / PIVOT
360 GO TO 370
370 AICCLUD, L) = AICCLUD, L) / PIVCT
C DIVIDE PIVOT ROW BY PIVOT ELEMENT

C
380 DO 550 L = 1, N
390 IF (LI - ICCLUD) >= 400, 400, 410
410 T = A(LI, 1CCLUD)
420 A(LI, 1CCLUD) = 0.0
430 DO 450 L = 1, N
450 A(LI, L) = A(L1, L) - A1CCLUD, L) * T
460 GO TO 500
500 B(LI, L) = B(LI, L) - B1CCLUD, L) * T
550 CONTINUE
C REDUCE NON-PIVOT ROWS

C
600 DO 710 I = 1, N
610 L = N + 1 - I
620 IF (INDEX(L, 1) - INDEX(L, 2)) >= 630, 630, 640
630 JCCLUD = INDEX(L, 1)
640 JCLUD = INDEX(L, 2)
650 DO 705 K = 1, N
660 SWAP = A(K, JCCLUD)
670 A(K, JCCLUD) = A1K, JCLUD)
700 A(K, JCCLUD) = SWAP
705 CONTINUE
710 CONTINUE
740 RETURN
END
Plotting Subroutines

Usage descriptions are given for the plotting subroutines called by the plotting programs of this report. These subroutines are LRC computer system library subroutines and listings are not given. They are described in the following order:

<table>
<thead>
<tr>
<th>Subroutine</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>CALCOMP</td>
<td>133</td>
</tr>
<tr>
<td>AXES</td>
<td>134</td>
</tr>
<tr>
<td>CALPLT</td>
<td>136</td>
</tr>
<tr>
<td>DASHLN</td>
<td>138</td>
</tr>
<tr>
<td>LEROY/BALLPT</td>
<td>139</td>
</tr>
<tr>
<td>LINE</td>
<td>140</td>
</tr>
<tr>
<td>NOTATE</td>
<td>141</td>
</tr>
<tr>
<td>PNTPLT</td>
<td>143</td>
</tr>
</tbody>
</table>
Subroutine CALCOMP

LANGUAGE: FORTRAN

PURPOSE: This is the normal mode processor. The necessary parameters and linkage are set up to output a tape for the CalComp Model 780/763 Electro-Mechanical Plotter.

USE: CALL CALCOMP

RESTRICTIONS: This call must be given before the first call to a plotting routine.

STORAGE: CALCOMP 54678 total for all subprograms used.

SUBPROGRAMS USED: PLOTSW, PLT763, BLCR, STRC, TAPWRI, ENCOD1, STRCALL, CREATEF, BOUNDCK, TRUNCL, and LOCATE

OTHER CODING INFORMATION: The following is a list of messages, the circumstances under which they will appear, and the action taken:

NO PLOTTING DEVICE SPECIFIED

This message is printed in the output file and the job is ended in subroutine PLOTSW. This condition occurs if there is no initialization CALL CALCOMP in the program prior to using subroutines which generate plotting output.

PLOTTING COMMENCED

This message is printed in the dayfile when the first pen movement is encountered as a result of a program call to the plotting subroutines.

THE LAST CALCOMP BLOCK ADDRESS WAS xxx DATA PLOTTED = yyyyyy

These messages are printed in the dayfile when the plotting is completed. xxx is the value of the last block address on the CalComp plotter tape. This block address is at the end of the last valid data. yyyyyy is the approximate number of data points. Plotting is completed either as a result of a CALL CALPLT(x,y,999) or when the job is ended due to an error recognized by the CalComp subroutines.
APPENDIX – Continued

Subroutine AXES

LANGUAGE: FORTRAN

PURPOSE: To draw a line with tick marks at specified intervals, annotate the value of the variable at tick marks, and provide an axis identification label.

USE: CALL AXES (X, Y, THETA, DIST, ORIGIN, DV, TMAJ, TMIN, BCD, HGT, N) where

X, Y are the coordinates in floating point inches of the starting point of the axis with reference to the plotting area origin as established by CALPLT.

THETA is the angle of rotation measured counterclockwise from the X-axis in floating point degrees. NOTE: Normally, THETA is 0° for an X-axis and 90° for a Y-axis.

DIST is the length of the axis in floating point inches. Should be a multiple of TMAJ.

ORIGIN is the functional value to be assigned to the origin (i.e., the value of the first scale), in floating point.

DV is the adjusted scale factor for the array to be plotted (change in value per in.). NOTE: Values of ORIGIN and DV which will produce a reasonable scale may be calculated using the subroutine ASCALE.

TMAJ is the distance in floating point inches for major tick marks (0.25 in. high). Numbers are placed on the axis at the major tick marks in accordance with the values of ORIGIN and DV. The numbers written along the axis are adjusted to be between 1000.00 and 0.01 in magnitude. Immediately after the last number on the axis is placed the caption, x10^exp, where exp is the required exponent.

If the values are integer multiples, the decimal point and decimal places are eliminated. A negative TMAJ will cause the actual value to be written instead of the adjusted value.
APPENDIX – Continued

TMIN is the divisions per inch in floating point for minor tick marks (0.125 in. high). To eliminate minor tick marks the following may be used:

\[ TMIN = 0. \]

BCD is the character label for the axis (see NOTATE routine).

HGT is the height of the full-size characters in the BCD title. Numbers at the tick marks will be \((0.75 \times HGT)\) high. HGT is in floating point inches.

If \( HGT = 0. \), all annotation will be eliminated.

N is an integer specifying the number of characters in BCD title. A negative \( N \) places the annotation on the clockwise side of the axis and a positive \( N \) places the annotation on the counterclockwise side of the axis. \( N = 0 \) is not allowed. If it is desired to have no label, then the BCD parameter should be \( 1HA \) and \( N = +1 \) or \( -1 \).

RESTRICTIONS: Only perpendicular axes are recommended.

STORAGE: 10168

SUBPROGRAMS USED: CALPLT, NOTATE, NUMBER, ROUND, SIN, COS
Subroutine CALPLT

**LANGUAGE:** FORTRAN

**PURPOSE:** To move the plotter pen to a new location with pen up or down and to signal the end of a job segment by incrementing the block address number.

**USE:**

- CALL CALPLT (X, Y, IPEN)
- **X, Y** The floating-point values for pen movement
- **IPEN = 2** Pen down
- **IPEN = 3** Pen up

Negative IPEN will assign \((X = 0, \ Y = 0)\) as the location of the pen after moving the \(X, Y\) (creating a new reference point) and will increase the block number by one. (The block number is the number that appears in the display at the top of the tape drive on the plotter and identifies the portion of the output tape that is being plotted. The block address 001 is written automatically as a result of the initialization processor call.) Each block address generally implies a separate page or plot.

- **IPEN = 999** Writes a terminating block address of 999 for peripheral handling of the plotter tape and all further processing is skipped. \(X\) and \(Y\) may be any values since they are ignored.

**RESTRICTIONS:** All \(X\)- and \(Y\)-coordinates must be expressed as floating-point values in inches (actual page dimensions) in deflection from the origin.

(A CALL TO CALPLT WITH EITHER NEGATIVE IPEN (USUALLY -3) OR A TERMINATING BLOCK ADDRESS (IPEN = 999) MUST BE GIVEN AS THE LAST PLOTTING INSTRUCTION BEFORE ENDING A PROGRAM WHICH USES ANY OF THE PLOTTER SUBROUTINES: THIS IS TO BE SURE THAT ALL PLOTTER INSTRUCTIONS ARE WRITTEN ON THE PLOTTER TAPE.)
The main subroutine in the CalComp software package is the CALPLT subroutine. All other special-purpose subroutines eventually call CALPLT either directly or indirectly. Subroutine CALPLT moves the pen in a straight line between the present pen position and another pen location to which the programmer wishes the pen to be moved.

In order to cause such instructions to be written, the programmer specifies the coordinates of the point to which the pen is to be moved and whether the pen is to be moved in a raised or lowered position. This movement is accomplished by the FORTRAN instruction

\[
\text{CALL CALPLT (X, Y, IPEN)}
\]

Also, the subroutine provides "sequence numbers" on the tape, making it possible to afford identification of job segments. The block address 001 is written on the first call to CALPLT. Thereafter, if the programmer defines a new origin or wishes to divide the job into several segments, he need only set the argument IPEN negative. The CALPLT routine then moves the pen to (X,Y); stores this location as (0,0), that is, a new origin; and increases the block address by one.

**STORAGE:**  CALPLT  2518

**SUBPROGRAMS USED:**  PLOTSW, STRCALL, and LOCATE
APPENDIX – Continued

Subroutine DASHLN

LANGUAGE: FORTRAN

PURPOSE: To draw a dashed line between two points.

USE: CALL DASHLN (X0, Y0, X1, Y1, D) where

X0, Y0, and X1, Y1 are coordinates in floating point inches of the end points of a line.

D is the length in floating point inches of each dash.

RESTRICTIONS: No matter what the slope of the line, the dash length will remain the requested length. The first dash of a line segment is set at one-half the requested dash length so that whenever line segments are connected, the dash at the meeting of the line segments will not be twice as large as the requested dash length. The last dash of a line segment will derive its length from that portion at the end of a line segment which is less than the requested dash length. The subroutine will draw a dash the length of the line segment if a dash length is requested that is equal to or larger than the line segment. If the end points of the line segment are the same, the subroutine will return to the calling program.

METHOD: No matter what the slope of the line, the dash length will remain the requested length. The first dash of a line segment is set at one-half the requested dash length so that whenever line segments are connected, the dash at the meeting of the line segments will not be twice as large as the requested dash length. The last dash of a line segment will derive its length from that portion at the end of a line segment which is less than the requested dash length. The subroutine will draw a dash the length of the line segment if a dash length is requested that is equal to or larger than the line segment. If the end points of the line segment are the same, the subroutine will return to the calling program.

STORAGE: 2678

SUBPROGRAMS USED: CALPLT, SQRT
Subroutine LEROY/BALLPT

**LANGUAGE:** FORTRAN

**PURPOSE:** The parameters necessary to accommodate plotting with the liquid ink pen are set up by CALL LEROY. Once set, this mode will remain in effect as long as CALCOMP is in use or until a call to BALLPT is given.

The parameters for plotting with the ballpoint pen are reset by CALL BALLPT. This mode is automatically in effect with CALCOMP unless there has been a call to LEROY.

**USE:**

CALL BALLPT
CALL LEROY

**RESTRICTIONS:** The CALL LEROY should be used only with CALCOMP. In addition to reducing the speed of the plotter for all plotting movements, the number of plot vectors in any annotation is considerably increased.

The CALL LEROY must be made prior to any plotting calls, but after the CALL CALCOMP.

**STORAGE:** LEROY/BALLPT 25g

**SUBPROGRAMS USED:** None.
Subroutine LINE

LANGUAGE: FORTRAN

PURPOSE: To draw a continuous line through and/or draw a symbol at each successive data point (stored in an array).

USE: CALL LINE (XARRAY, YARRAY, N, K, J, L, S) where

XARRAY and YARRAY are the names of arrays containing the X values and Y values, respectively, to be plotted. Values must be in floating point.

N is the number of points to be plotted.

K is the interleave factor of a mixed array (normally = 1).

J is positive for line and symbol plot, negative for symbol only plot. The magnitude specifies the alternate number of data points at which to plot a symbol.

= 0 for line plot.

= 1 for symbol for every data point.

= 2 for symbol for every other data point, etc.

L is an integer describing symbol to be used, see NOTATE routine for list.

S is the desired symbol height in floating point (see NOTATE routine).

RESTRICTIONS: LINE expects the adjusted minimums and scale factors as described in ASCALE since the routine automatically sets an origin and scales the data in the array.

STORAGE: 3528

SUBPROGRAMS USED: CALPLT, NOTATE, WHERE
APPENDIX – Continued

Subroutine NOTATE

**LANGUAGE:** FORTRAN

**PURPOSE:** To draw alphanumeric information for annotation and labeling.

**USE:**

CALL NOTATE (X, Y, HEIGHT, BCD, THETA, N) where

- **X,Y** are the floating point page coordinates of the first character. The coordinates of the lower left-hand corner of the characters are specified.
- **HEIGHT** specifies character size and spacing in floating point inches for a full-size character. The smallest possible character is 0.07 inch high. The width of a character will be \((4/7) \times \text{HEIGHT}\) and the space between characters is \((2/7) \times \text{HEIGHT}\).
- The ith character is plotted at:
  \[
  x_i = X + (i-1)(6/7)(\text{HEIGHT})(\cos \theta)
  \]
  \[
  y_i = Y + (i-1)(6/7)(\text{HEIGHT})(\sin \theta)
  \]
  \(1 \leq i \leq N\)
- **BCD** is the string of characters to be drawn and is usually written in the form: nHXXXX--- (the same way an alpha message is written using FORTRAN format statements). Instead of specifying alpha information as above, one may give the beginning storage location of an array containing alphanumeric information.
- **THETA** is the angle in floating point degrees at which the information is to be drawn. Zero degrees will print horizontally reading from left to right, 90° will print the line vertically reading from bottom to top, 180° will print the line horizontally reading from right to left (i.e., upside down), and 270° will print vertically reading from top to bottom.
- **N** is the number of characters, including blanks, in the label.
APPENDIX – Continued

METHOD: The character height is a variable entry parameter to the subroutine NOTATE. However, the width-to-height ratio is fixed at 4/7. This is because the characters are defined by a series of bi-octal offset pairs for a 4×7 matrix. The reference origin for the offset pairs which define each character is the lower left-hand corner of the matrix. The X and Y values which are entry parameters to NOTATE define the location of the lower left-hand corner of the first character to be plotted for this entry to NOTATE. Subsequent characters to be plotted are spaced from the previous character origin by 6/7 of the specified character height.

OTHER CODING INFORMATION: Only the alphanumeric option used in this report is described by this usage description.

STORAGE: 1252₈

SUBPROGRAMS USED: CALPLT, CNTRLN, DECODE, DECOD2, SIN, COS
Subroutine PNTPLT

CALL PNTPLT (A, B, NO, IS) where

A is the X coordinate for the centered symbol in floating point inches.

B is the Y coordinate for the centered symbol in floating point inches.

NO is an integer specifying the symbol to be used.

= 21 for a point ·

= 22 for a plus sign +

IS is an integer value specifying the size symbol to be used.

= 1 small

= 2 medium

= 3 large

(See fig. A1.)

A positive integer value for NO in the calling sequence will produce symbols of the same quality as in figure A1. A negative integer value will produce symbols of less quality but will result in a considerably faster computer run.
## APPENDIX – Concluded

<table>
<thead>
<tr>
<th>INTEGER BACK REFERENCE</th>
<th>SMALL</th>
<th>MEDIUM</th>
<th>LARGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>2</td>
<td>□</td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>3</td>
<td>◊</td>
<td>◊</td>
<td>◊</td>
</tr>
<tr>
<td>4</td>
<td>△</td>
<td>△</td>
<td>△</td>
</tr>
<tr>
<td>5</td>
<td>▲</td>
<td>▲</td>
<td>▲</td>
</tr>
<tr>
<td>6</td>
<td>◉</td>
<td>◉</td>
<td>◉</td>
</tr>
<tr>
<td>7</td>
<td>◊</td>
<td>◊</td>
<td>◊</td>
</tr>
<tr>
<td>8</td>
<td>◊</td>
<td>◊</td>
<td>◊</td>
</tr>
<tr>
<td>9</td>
<td>△</td>
<td>△</td>
<td>△</td>
</tr>
<tr>
<td>10</td>
<td>△</td>
<td>△</td>
<td>△</td>
</tr>
</tbody>
</table>

Figure A1. - NASA standard plot symbols.
REFERENCES


