NASA CONTRACTOR REPORT

NASA CR-1391

TECH LIBRARY KAFB, NM

0060480

c. 1

LOAN COPY: RETURN TO
AFWL (WLIL-2)
KIRTLAND AFB, N MEX

NASA CR-1391

RIGID ROTOR DYNAMICS

by Edgar J. Gunter, Jr., and P. De Choudhury

Prepared by
UNIVERSITY OF VIRGINIA
Charlottesville, Va.
for Lewis Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • AUGUST 1969

TECH LIBRARY KAFB, NM

0060480

NASA CR-1391

RIGID ROTOR DYNAMICS

By Edgar J. Gunter, Jr., and P. De Choudhury

Distribution of this report is provided in the interest of information exchange. Responsibility for the contents resides in the author or organization that prepared it.

Prepared under Grant No. NGR 47-005-050 by
Department of Mechanical Engineering
UNIVERSITY OF VIRGINIA
Charlottesville, Va.

for Lewis Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

For sale by the Clearinghouse for Federal Scientific and Technical Information
Springfield, Virginia 22151 – CFSTI price $3.00

ABSTRACT

This report analyzes the dynamic motion of an unbalanced rigid body rotor in general linearized bearings and the analysis is applicable to fluid film as well as rolling element bearings for small bearing displacements. The complete nonlinear dynamical equations of motion, including rotor acceleration, are derived by Lagrange's equations to include the influence of damped, flexibly mounted bearing supports. The dynamical equations of motion are linearized by assuming constant angular shaft velocity and shaft displacements, which are small in comparison to the rotor characteristic length. Computer programs to analyze the rotor steady state motion due to unbalance and the stability and complete transient response are presented. As an example, these computer programs are applied to evaluate the characteristics of a NASA experimental hybrid gas bearing rotor system.

FOREWORD

The research described herein, which was conducted at the University of Virginia, was performed under NASA Research Grant NGR 47-005-050 with Mr. William J. Anderson, Fluid System Components Division, NASA-Lewis Research Center, as Technical Manager.

CONTENTS

LIST OF FIGURES

NOMENCLATURE

A	rotor amplification factor
B_{ix}	bearing housing horizontal damping
B_{iy}	bearing housing vertical damping
C_{ixx}	C_{ix} = bearing damping coefficient in x-direction for the i^{th} bearing
C_{ixy}	D_{iy} = cross coupled bearing damping coefficient for force in x-direction from y-displacement
C_{iyx}	D_{ix} = cross coupled bearing damping coefficient for force in y-direction from x-displacement
C_{iyy}	C_{iy} = bearing damping coefficient in y-direction for the i^{th} bearing
C_z	axial thrust bearing damping coefficient
D	total dissipation energy
f_{ix}	bearing housing horizontal stiffness
f_{iy}	bearing housing vertical stiffness
G_{ix}	bearing housing angular stiffness in the x-direction
G_{iy}	bearing housing angular stiffness in the y-direction
h_1, h_2	axial location of unbalance masses from first bearing
I_P, I_T	rotor polar moments of inertia taken about C. G.
$I_{1,2}$	transverse moment of inertia of bearing housings one and two at point O_{b_1}, O_{b_2}
K_{ixx}	K_{ix} = bearing stiffness in x-direction for the i^{th} bearing, here, i = 1, 2
K_{ixy}	R_{iy} = cross coupling stiffness coefficient for force in x-direction from y-displacement
K_{iyx}	R_{ix} = cross coupling stiffness coefficient for force in y-direction from x-displacement
K_{iyy}	K_{iy} = bearing stiffness in y-direction for the i^{th} bearing, here, i = 1, 2
K_z	axial thrust bearing stiffness coefficient
L	rotor length between bearing spans
L_1	distance from first bearing to mass center of rotor
L_2	distance from second bearing to mass center of rotor
M_{ix}	bearing angular stiffness in x-direction

M_{iy}	bearing angular stiffness in y-direction
m_1, m_2	mass of bearing housing 1 and 2
N_{ix}	bearing angular damping in x-direction
N_{iy}	bearing angular damping in y-direction
P_{ix}	cross coupling angular damping coefficient in y-direction due to rotation in x-direction
P_{iy}	cross coupling angular damping coefficient in x-direction due to rotation in y-direction
Q_{ix}	cross coupling angular stiffness producing moment in y-direction due to rotation in x-direction
Q_{iy}	cross coupling angular stiffness producing moment in x-direction due to rotation in y-direction
R_1, R_2	radial distance of unbalance masses from rotor centerline
T	total kinetic energy of system
T_R	kinetic energy of rotation of balanced rotor
T_T	kinetic energy of translation
T_U	kinetic energy of unbalance masses
t	time
$u_{1,2}$	foundation or bearing housing motion in horizontal direction
V	total potential energy of system
$v_{1,2}$	foundation or bearing housing motion in vertical direction
x_m, y_m, z_m	displacement of rotor mass center
x_1, y_1	displacement of rotor at number 1 bearing location
x_2, y_2	displacement of rotor at number 2 bearing location
α_1	shaft angular displacement in x-z plane, $(x_2 - x_1)/L$
α_2	shaft angular displacement in y-z plane, $(y_2 - y_1)/L$
α_3	shaft angular displacement about spin axis
$\beta_{1,2}$	angular displacement of bearing housings in y-z plane
$\gamma_{1,2}$	angular displacement of bearing housings in x-z plane
$\vec{\delta}_{b1}$	position vector of first bearing center 0_{b1}

$\vec{\delta}_{j1}$	position vector of first journal center O_{j_1} relative to bearing center O_{b_1}
$\delta M_1, \delta M_2$	rotor unbalance masses; $\frac{\delta Mi}{M} << 1$
μ_{ix}	bearing housing angular damping in x-direction
μ_{iy}	bearing housing angular damping in y-direction
ρ_1, ρ_2	axial distance of unbalance masses from rotor mass center
Φ	angular phase displacement between two unbalance masses
Ω	angular velocity vector
ω	rotor angular velocity

PART I

INTRODUCTION

1.01 Statement of the Problem

The purpose of this investigation has been to derive the equations of motion of a rigid body rotor with an exciting force caused by unbalance situated along different locations and different planes. The gyroscopic effects of the rotor on the system have also been taken into account. The derived equations of motion include the bearing and support characteristics.

Computer programs have been developed to investigate the steady state and transient behavior of the rotor-bearing system which enables a parametric study of the system to be made.

The dynamical equations presented may be applied to any arbitrary rigid body rotor system, regardless of the type of bearings used, whether it be fluid film or rolling element bearing.

1.02 Description of Rotor Coordinate System

Figure 1 represents an arbitrary rotor system mounted in bearings on damped elastic supports. In order to express the dynamical equations of motion for the system, the total number of degrees of freedom must be determined. The required number of dynamical equations necessary will be determined by the degrees of freedom minus the equations of constraint. The constraint relations will be discussed in section 1.03.

The rigid body rotor has six degrees of freedom and requires six generalized coordinates to completely specify its motion. The proper choice of the coordinate system is important in order to express the dynamical equations in their simplest form. Two types of coordinate systems that may be employed are the Eulerian coordinate system and the Cartesian coordinate system. A detailed discussion of these coordinate systems and their equations of transformation are given in appendix A. Both coordinate systems consist of the Cartesian displacement of the rotor mass center and three angular displacements. The Eulerian coordinate system, which is commonly used to represent gyroscope systems, is given by

ψ	local spin angle
φ	precession angle
θ	nutation angle
x_m, y_m, z_m	Cartesian components of rotor mass center

In the Cartesian system, the generalized coordinates are given by

α_1	rotor angular displacement in y-z plane
α_2	rotor angular displacement in x-z plane
α_3	rotor spin angle
x_m, y_m, z_m	Cartesian components of the rotor mass center

The equations of motion in either coordinate system are, in general, highly nonlinear. The Cartesian coordinate system has the advantage that if small rotor displacements and constant angular velocity are assumed, the dynamical equations become linearized. This

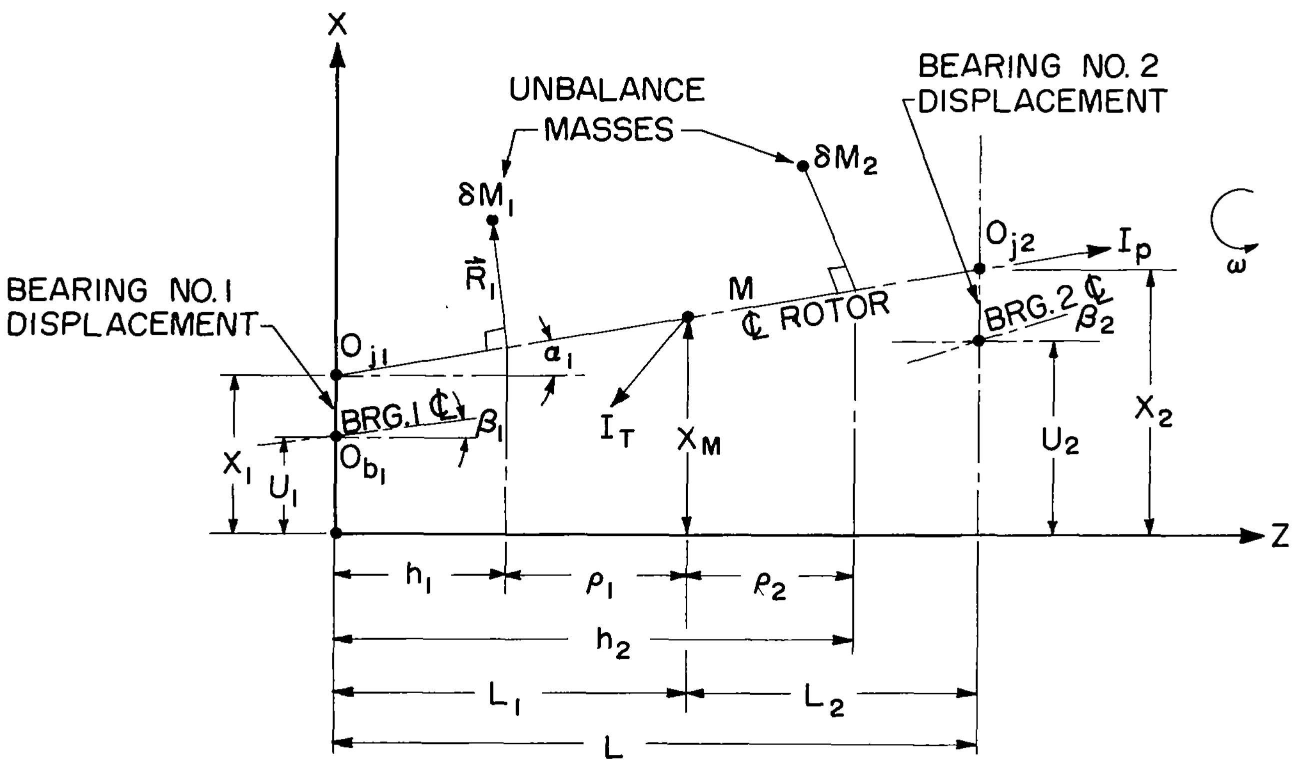


Figure 1. - Schematic side view of rotor system.

linearization process is not possible with the Eulerian representation. The Eulerian equations possess certain advantages in the investigation of asymptotic solutions for rotor precession rate and also are useful in the analysis of unstable forced backward rotor precession.

Figure 2 represents a schematic side view of the rotor system in the Cartesian x-z plane. The angular displacements α_1 and α_2 are given by

$$\alpha_1 = \frac{x_2 - x_1}{L} \tag{1.1}$$

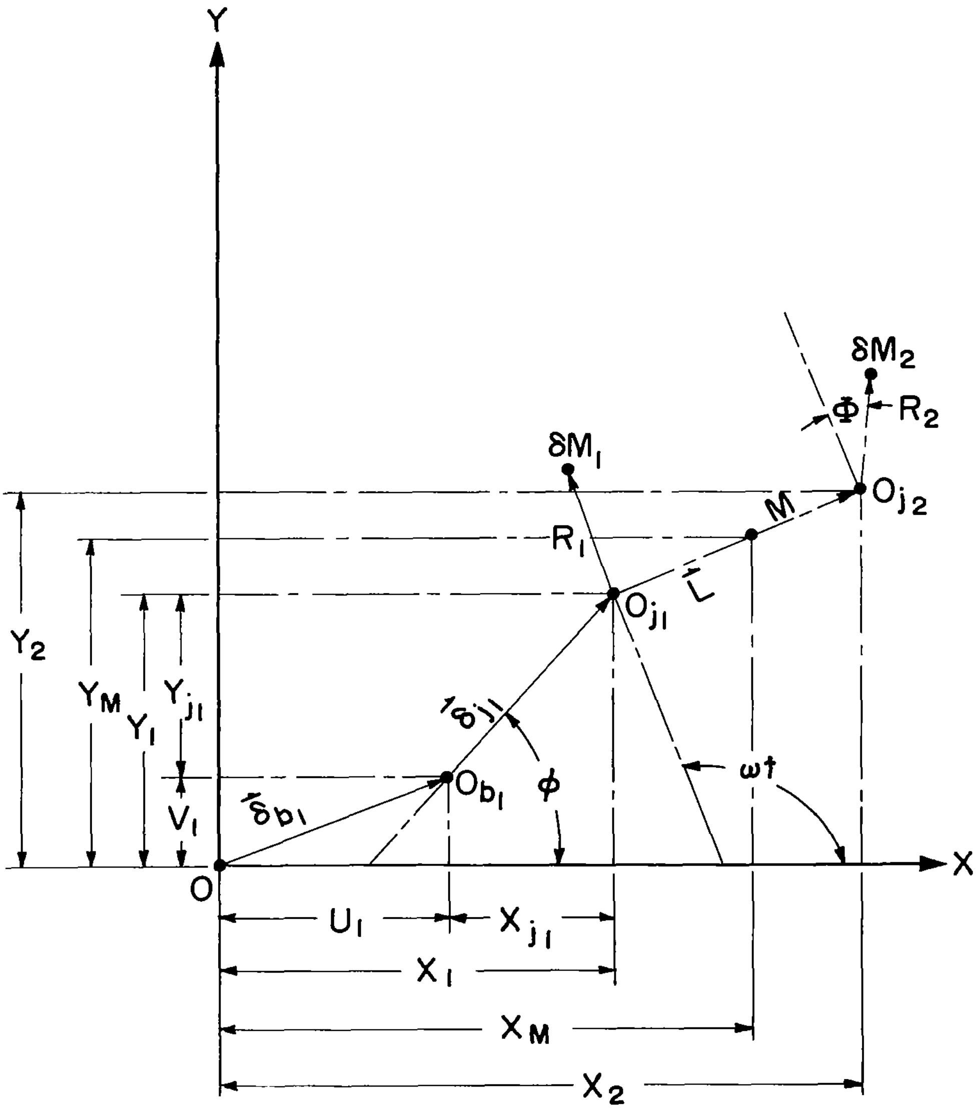


Figure 2. - Schematic axial view of rotor displacement at first bearing.

$$\alpha_2 = \frac{y_2 - y_1}{L} \tag{1.2}$$

where

x_1, y_1 absolute displacement at number 1 bearing

x_2, y_2 absolute displacement at number 2 bearing

The rotor displacements may be represented as follows:

$$\left.\begin{aligned} x_1 &= x_2 - \alpha_1 L = x_m - \alpha_1 L_1 \\ y_1 &= y_2 - \alpha_2 L = y_m - \alpha_2 L_1 \end{aligned}\right\} \tag{1.3}$$

$$\left.\begin{aligned} x_2 &= x_1 + \alpha_1 L = x_m + \alpha_1 L_2 \\ y_2 &= y_1 + \alpha_2 L = y_m + \alpha_2 L_2 \end{aligned}\right\} \tag{1.4}$$

Solving for the displacement at the rotor mass center

$$x_m = \frac{L_1 x_2 + L_2 x_1}{L} \tag{1.5}$$

$$y_m = \frac{L_1 y_2 + L_2 y_1}{L} \tag{1.6}$$

Figure 3 represents a schematic axial view of the rotor displacements at the first bearing. The absolute displacements x_1 and y_1 of the first journal center (point O_{j_1}) are given by

$$x_1, y_1 = (\vec{\delta}_{b_1} + \vec{\delta}_{j_1}) \cdot (\vec{n}_x, \vec{n}_y) \tag{1.7}$$

where

δ_{b_1} position vector of first bearing center O_{b_1}

δ_{j_1} position vector of first journal center O_{j_1} relative to the bearing center O_{b_1}

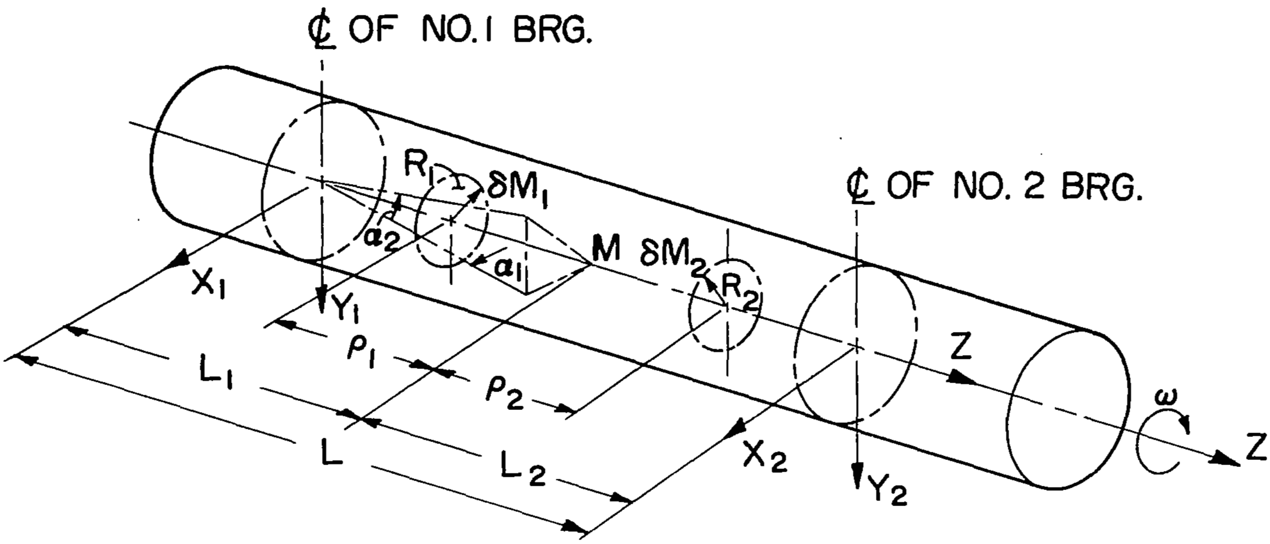


Figure 3. - Isometric view of rotor coordinate system.

The final equations of motion will be expressed in terms of the coordinates x_1, y_1, x_2, y_2, z_m, and ωt.

1.03 Bearing Coordinate System

Four degrees of freedom are required to represent the motion of each bearing housing since the bearing housing does not spin and axial bearing motion is assumed to be negligible. Therefore, the six degrees of freedom reduce to four.

The four coordinates used to represent the bearing motion are

- u, v — horizontal and vertical displacement of the midpoint of the bearing centerline O_b
- γ — angular displacement of the bearing center in the x-z plane
- β — angular displacement of the bearing center in the y-z plane

The number of degrees of freedom for two bearings is therefore eight. The total number of degrees of freedom for a rigid rotor is six. If, however, we assume that the rotor is moving with constant angular velocity, then the number of degrees of freedom for the system reduces to 13. This assumption of constant angular velocity uncouples the equation of motion of the rotor in the axial direction from the rest of the system equations for small displacements. Hence, the axial motion can be investigated independently from the remaining equations of motion.

Summarizing, the twelve generalized coordinates are the horizontal and vertical displacements of the rotor axis at the first and second bearings along with rotor angular motions which can also be expressed in terms of the horizontal and vertical displacement of the rotor at the two bearings. The two bearing housings are assumed to be flexible and as such, have x and y displacements along with angular motion in y-z and x-z planes denoted by β and γ, respectively.

In addition to the bearing stiffness and damping coefficients acting in x- and y-directions, cross coupled bearing characteristics are assumed to be present. Bearing housings have, in addition to the horizontal and vertical stiffness and damping, the angular stiffness, damping, and cross coupled stiffness and damping.

The generalized rotor coordinates are shown in figure 3 and those of the bearing housing in figure 4.

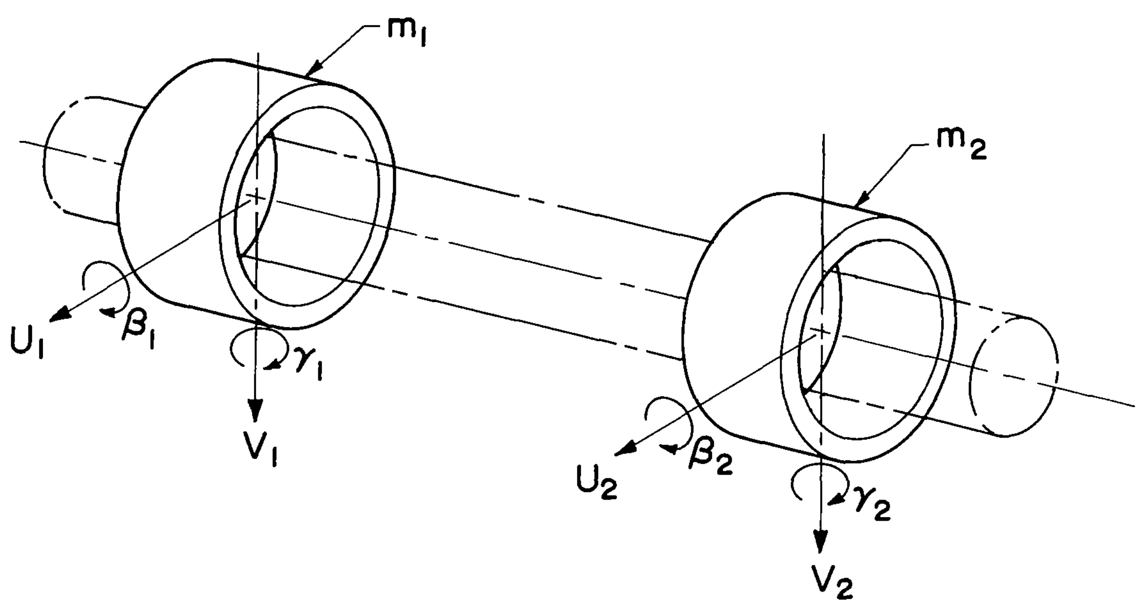


Figure 4. - Isometric view of bearing coordinate system.

PART II

ROTOR EQUATIONS OF MOTION

2.01 System Kinetic Energy

The equations of motion are derived from Lagrange's equation, which states that

$$\frac{d}{dt}\left[\frac{\partial T}{\partial \mathring{q}_r}\right] - \frac{\partial T}{\partial q_r} + \frac{\partial V}{\partial q_r} + \frac{\partial D}{\partial \mathring{q}_r} = Fq_r \qquad (2.1)$$

where

$$T = \text{Total kinetic energy of system}$$

$$= T_T + T_R + T_U \qquad (2.2)$$

where

$$T_T = \text{Kinetic energy of translation}$$

$$= \frac{1}{2} M_i \bar{v}_i \cdot \bar{v}_i \qquad (2.3)$$

$$T_R = \text{Kinetic energy of rotation of balanced rotor}$$

$$= \frac{1}{2} \omega_i \omega_j I_{ij}$$

$$= \frac{1}{2}\left[\omega_1^2 I_{11} + \omega_2^2 I_{22} + \omega_3^2 I_{33} + 2\omega_1\omega_2 I_{12} + 2\omega_1\omega_3 I_{13} + 2\omega_2\omega_3 I_{23}\right] \qquad (2.4)$$

and

$$T_U = \text{Kinetic energy of the unbalance masses}$$

$$= \frac{1}{2}\sum_{i=1}^{2} \delta M_i \vec{v}_i \cdot \vec{v}_i \qquad (2.5)$$

(See appendix A for the derivation of the kinetic energy expressions.) If the set of axes chosen are principal axes, the product of inertia terms are zero, and the kinetic energy of rotation reduces to:

$$T_R = \frac{1}{2}\left[\omega_1^2 I_{11} + \omega_2^2 I_{22} + \omega_3^2 I_{33}\right] \tag{2.6}$$

For a balanced axisymmetric rotor $I_{11} = I_{22} = I_T$, the rotor transverse moment of inertia; and $I_{33} = I_p$ the polar moment of inertia.

The total rotor kinetic energy of the assumed rotor-bearing system is given by:

$$\begin{aligned} T = {} & \frac{1}{2} M\left[\dot{x}_m^2 + \dot{y}_m^2 + \dot{z}_m^2\right] + \frac{1}{2}\left[(\dot{\alpha}_2^2 + \dot{\alpha}_1^2 \cos^2 \alpha_2) I_T + (\dot{\alpha}_3 + \dot{\alpha}_1 \sin \alpha_2)^2 I_p\right] \\ & + \frac{\delta M_1}{2}\Big\{(\dot{x}_m + \rho_1\dot{\alpha}_1 - R_1\dot{\alpha}_3 \sin \alpha_3)^2 + (\dot{y}_m + \rho_1\dot{\alpha}_2 + R_1\dot{\alpha}_3 \cos \alpha_3)^2 \\ & + \left[\dot{z}_m - R_1(\dot{\alpha}_2 \sin \alpha_3 + \dot{\alpha}_1 \cos \alpha_3)\right]^2\Big\} \\ & + \frac{\delta M_2}{2}\Big(\left[\dot{x}_m + \rho_2\dot{\alpha}_1 - \dot{\alpha}_3 R_2 \sin(\alpha_3 + \Phi)\right]^2 \\ & + \left[\dot{y}_m + \rho_2\dot{\alpha}_2 + R_2\dot{\alpha}_3 \cos(\alpha_3 + \Phi)\right]^2 \\ & + \Big\{\dot{z}_m - R_2\left[\dot{\alpha}_2 \sin(\alpha_3 + \Phi) + \dot{\alpha}_1 \cos(\alpha_3 + \Phi)\right]\Big\}^2\Big) \end{aligned} \tag{2.7}$$

The kinetic energy of translation and rotation of the bearing housing is given by:

$$T_B = \frac{1}{2} m_1(\dot{u}_1^2 + \dot{u}_2^2) + \frac{1}{2} m_2(\dot{v}_1^2 + \dot{v}_2^2) + \frac{1}{2}\left[I_{ix}\dot{\beta}_1^2 + I_{2x}\dot{\beta}_2^2 + I_{iy}\dot{\gamma}_1^2 + I_{2y}\dot{\gamma}_2^2\right] \tag{2.8}$$

2.02 Potential Energy With N-Bearings

The potential energy with N-bearing locations due to the stiffness in the horizontal and vertical directions is given by:

$$V_1 = \frac{1}{2}\sum_{i=1}^{N} (K_{ix}x_i^2 + K_{iy}y_i^2) \tag{2.9}$$

The potential energy of the bearing due to cross coupled damping terms is given by:

$$v_2 = \sum_{i=1}^{N} (D_{ix}\dot{x}_i y_i + D_{iy} x_i \dot{y}_i) \tag{2.10}$$

The potential energy of the bearing due to angular stiffness and cross coupled damping coefficients is given by:

$$v_3 = \frac{1}{2}\sum_{i=1}^{N}\left[(M_{ix}(\alpha_2 - \beta_i)^2 + M_{iy}(\alpha_1 - \gamma_i)^2\right]$$
$$+ \sum_{i=1}^{N} \left[P_{ix}(\dot{\beta}_i - \dot{\alpha}_2)(\alpha_1 - \gamma_i) + P_{iy}(\dot{\alpha}_1 - \dot{\gamma}_i)(\beta_2 - \alpha_2)\right] \tag{2.11}$$

The potential energy of the bearing housing due to the horizontal and vertical stiffness and the angular stiffness is given by:

$$v_4 = \frac{1}{2}\sum_{i=1}^{N} \left(f_{ix}U_i^2 + f_{iy}V_i^2 + G_{ix}\beta_i^2 + G_{iy}\gamma_i^2\right) \tag{2.12}$$

and that due to thrust bearing is

$$V_5 = \frac{1}{2} K_z z_M^2$$

The total potential energy is then

$$V = V_1 + V_2 + V_3 + V_4 + V_5 = \frac{1}{2}\sum_{i=1}^{N} \Big[K_{ix}x_i^2 + K_{iy}y_i^2 + 2\, D_{ix}\dot{x}_i y_i + 2\, D_{iy}x_i\dot{y}_i$$
$$+ M_{ix}(\alpha_2 - \beta_i)^2 + M_{iy}(\alpha_1 - \gamma_i)^2 + 2\, P_{ix}(\dot{\beta}_i - \dot{\alpha}_2)(\alpha_1 - \gamma_i)$$
$$+ 2\, P_{iy}(\dot{\alpha}_1 - \dot{\gamma}_1)(\beta_2 - \alpha_2) + f_{ix}U_i^2 + f_{iy}V_i^2 + G_{ix}\beta_i^2 + G_{iy}\gamma_i^2\Big] + \frac{1}{2} K_z z_m^2 \tag{2.13}$$

The coordinates x_i and y_i are related to the coordinates x_m, y_m, α_1, and α_2 by the following relations:

$$x_i = x_m + \alpha_1 L_i \tag{2.14}$$

$$y_i = y_m + \alpha_\beta L_i \tag{2.15}$$

where L_i = distance from the rotor mass center to the centerline of the bearings.

If $N = 2$, i.e., there are only two bearings, then in terms of the above generalized coordinates, expressions (2.14) and (2.15) reduce to

$$V = \frac{1}{2}\Big[f_{1x}U_1^2 + f_{2x}U_2^2 + f_{1y}V_1^2 + f_{2y}V_2^2 + G_{1x}\beta_1^2 + G_{2x}\beta_2^2$$
$$+ G_{1y}\gamma_1^2 + G_{2y}\gamma_2^2 + K_{1x}(x_m - L_1\alpha_1 - U_1)^2 + K_{2x}(x_m + L_2\alpha_1 - U_2)^2$$
$$+ K_{1y}(y_m - L_1\alpha_2 - V_1)^2 + K_{2y}(y_m + L_2\alpha_2 - V_2)^2 + M_{1x}(\alpha_2 - \beta_1)^2$$
$$+ M_{2x}(\alpha_2 - \beta_2)^2 + M_{1y}(\alpha_1 - \gamma_1)^2 + M_{2y}(\alpha_1 - \gamma_2)^2\Big] \qquad (2.16)$$

$$+ D_{1y}(\dot{y}_m - L_1\dot{\alpha}_2 - \dot{V}_1)(x_m - L_1\alpha_1 - U_1) + D_{1x}(\dot{x}_m - L_1\dot{\alpha}_1 - \dot{U}_1)(y_m - L_1\alpha_2 - V_1)$$
$$+ D_{2y}(\dot{y}_m + L_2\dot{\alpha}_2 - \dot{V}_2)(x_m + L_2\alpha_1 - U_2) + D_{2x}(\dot{x}_m + L_2\dot{\alpha}_1 - \dot{U}_2)(y_m + L_2\alpha_2 - V_2)$$
$$+ P_{1x}(\dot{\beta}_1 - \dot{\alpha}_2)(\alpha_1 - \gamma_1) + P_{2x}(\dot{\beta}_2 - \dot{\alpha}_2)(\alpha_1 - \gamma_2) + P_{1y}(\beta_1 - \alpha_2)(\dot{\alpha}_1 - \dot{\gamma}_1)$$
$$+ P_{2y}(\beta_2 - \alpha_2)(\dot{\alpha}_1 - \dot{\gamma}_2) + \frac{1}{2}K_z z_m^2 \qquad (2.17)$$

2.03 Dissipation Energy With N-Bearings

The dissipation energy with N-bearing locations due to damping in x- and y-directions is given by

$$D_1 = \frac{1}{2}\sum_{i=1}^{N}(C_{ix}\dot{x}_i^2 + C_{iy}\dot{y}_i^2) \qquad (2.18)$$

The dissipation energy due to cross coupled stiffness terms is

$$D_2 = \sum_{i=1}^{N}(R_{ix}y_i\dot{x}_i + R_{iy}\dot{y}_i x_i) \qquad (2.19)$$

The dissipation energy due to angular damping and cross coupled angular damping is given by

$$D_3 = \frac{1}{2}\sum_{i=1}^{N}\left\{N_{ix}(\dot{\alpha}_2 - \dot{\beta}_i)^2 + N_{iy}(\dot{\alpha}_1 - \dot{\beta}_i)^2\right\}$$

$$+ \sum_{i=1}^{N}\left\{Q_{ix}(\dot{\alpha}_1 - \dot{\gamma}_i)(\beta_i - \alpha_2) + Q_{iy}(\dot{\beta}_i - \dot{\alpha}_2)(\alpha_1 - \gamma_i)\right\} \quad (2.20)$$

The dissipation energy of the bearing housing due to the horizontal and vertical damping and that due to angular damping is given by:

$$D_4 = \frac{1}{2}\sum(B_{ix}\dot{U}_i^2 + B_{iy}\dot{V}_i^2 + \mu_{ix}\dot{\beta}_i^2 + \mu_{iy}\dot{\gamma}_i^2) \quad (2.21)$$

and that due to the thrust bearing is

$$D_5 = \frac{1}{2} C_z \dot{z}_m^2 \quad (2.22)$$

The total dissipation energy is then given by:

$$D = D_1 + D_2 + D_3 + D_4 + D_5$$

$$= \frac{1}{2}\sum_{i=1}^{N}\Big\{C_{ix}\dot{x}_i^2 + C_{iy}\dot{y}_i^2 + 2R_{ix}x_i\dot{y}_i + 2R_{iy}\dot{x}_i y_i$$

$$+ N_{ix}(\dot{\alpha}_2 - \dot{\beta}_i)^2 + N_{iy}(\dot{\alpha}_1 - \dot{\beta}_i)^2 + 2Q_{ix}(\dot{\alpha}_1 - \dot{\gamma}_1)(\beta_i - \alpha_2)$$

$$+ 2Q_{iy}(\dot{\beta}_i - \dot{\alpha}_2)(\alpha_1 - \gamma_i) + B_{ix}\dot{U}_i^2 + B_{iy}\dot{V}_i^2 + \mu_{ix}\dot{\beta}_i^2 + \mu_{iy}\dot{\gamma}_i^2\Big\} + \frac{1}{2} C_z z_m^2 \quad (2.23)$$

Considering only two bearings and using equations (2.14) and (2.15), the expression for the total dissipation energy becomes:

$$D = \frac{1}{2}\Big[B_{ix}\dot{U}_i^2 + B_{2x}\dot{U}_2^2 + B_{iy}\dot{v}_1^2 + B_{2y}\dot{V}_2^2 + \mu_{ix}\dot{\beta}_1^2 + \mu_{2x}\beta_2^2$$

$$+ \mu_{1y}\dot{\gamma}_1^2 + \mu_{2y}\dot{\gamma}_2^2 + C_{1x}(\dot{x}_m - L_1\dot{\alpha}_1 - \dot{U}_1)^2 + C_{2x}(\dot{x}_m + L_2\dot{\alpha}_1 - \dot{U}_2)^2$$

$$+ C_{iy}(\dot{y}_m - L_1\dot{\alpha}_2 - \dot{V}_1)^2 + C_{2y}(\dot{y}_m + L_2\dot{\alpha}_2 - \dot{V}_2)^2 + N_{ix}(\dot{\alpha}_2 - \dot{\beta}_1)^2$$

$$+ N_{2x}(\dot{\alpha}_2 - \dot{\beta}_2)^2 + N_{1y}(\dot{\alpha}_1 - \dot{\gamma}_1)^2 + N_{2y}(\dot{\alpha}_1 - \dot{\gamma}_2)^2\Big]$$

$$+ R_{ix}(\dot{y}_m - L_1\dot{\alpha}_2 - \dot{V}_1)(x_m - L_1\alpha_1 - U_1) + R_{2x}(\dot{y}_m + L_2\dot{\alpha}_2 - \dot{V}_2)(x_m + L_2\alpha_1 - U_2)$$

$$+ R_{iy}(\dot{x}_m - L_1\dot{\alpha}_1 - \dot{U}_1)(y_m - L_1\alpha_2 - V_1) + R_{2y}(\dot{x}_m + L_2\dot{\alpha}_1 - \dot{U}_2)(y_m + L_2\alpha_2 - V_2)$$

$$+ Q_{ix}(\beta_1 - \alpha_2)(\dot{\alpha}_1 - \dot{\gamma}_1) + Q_{2x}(\beta_2 - \alpha_2)(\dot{\alpha}_1 - \dot{\gamma}_2)$$

$$+ Q_{iy}(\alpha_1 - \gamma_1)(\dot{\beta}_1 - \dot{\alpha}_2) + Q_{2y}(\alpha_1 - \gamma_2)(\dot{\beta}_2 - \dot{\alpha}_2) + \frac{1}{2}C_z\dot{z}_m^2 \qquad (2.24)$$

2.04 Nonlinear Equations of Motion

The equations of motion are obtained by applying Lagrange's equation (2.1) to the twelve generalized coordinates considered here.

These equations of motion are nonlinear and are as follows:

$$U_1 : \; m_1\ddot{U}_1 + f_{ix}U_1 - K_{ix}(x_1 - U_1) - R_{iy}(y_1 - V_1) - D_{iy}(\dot{y}_1 - \dot{V}_1)$$

$$+ B_{ix}\dot{U}_1 - C_{1x}(\dot{x}_1 - \dot{U}_1) = 0 \qquad (2.25)$$

$$V_1: \; m_1\ddot{V}_1 + f_{iy}V_1 - K_{iy}(y_1 - V_1) - R_{ix}(x_1 - U_1) - D_{ix}(\dot{x}_1 - \dot{U}_1)$$

$$+ B_{iy}\dot{V}_1 - C_{1y}(\dot{y}_1 - \dot{V}_1) = 0 \qquad (2.26)$$

$$U_2: \; m_2\ddot{U}_2 + f_{2x}U_2 - K_{2x}(x_2 - U_2) - R_{2y}(y_2 - V_2) - D_{2y}(\dot{y}_2 - \dot{V}_2)$$

$$+ B_{2x}\dot{U}_2 - C_{2x}(\dot{x}_2 - \dot{U}_2) = 0 \qquad (2.27)$$

$$V_2:\ m_2\ddot{V}_2 + f_{2y}V_2 - K_{2y}(y_2 - V_2) - R_{2x}(x_2 - U_2) - D_{2x}(\dot{x}_2 - \dot{U}_2)$$

$$+ B_{2y}\dot{V}_2 - C_{2y}(\dot{y}_2 - \dot{V}_2) = 0 \qquad (2.28$$

$$\beta_1:\ I_{1x}\ddot{\beta}_1 + G_{ix}\beta_1 + M_{1x}(\beta_1 - \alpha_2) + Q_{1y}(\alpha_1 - \gamma_1) + P_{1y}(\dot{\alpha}_1 - \dot{\gamma}_1)$$

$$+ \mu_{ix}\dot{\beta}_1 + N_{1x}(\dot{\beta}_1 - \dot{\alpha}_2) = 0 \qquad (2.29)$$

$$\gamma_1:\ I_{1y}\ddot{\gamma}_1 + G_{1y}\gamma_1 + M_{1y}(\gamma_1 - \alpha_1) + Q_{1x}(\alpha_2 - \beta_1) + P_{1x}(\dot{\alpha}_2 - \dot{\beta}_1)$$

$$+ \mu_{1y}\dot{\gamma}_1 + N_{1y}(\dot{\gamma}_1 - \dot{\alpha}_1) = 0 \qquad (2.30)$$

$$\beta_2:\ I_{2x}\ddot{\beta}_2 + G_{2x}\beta_2 + M_{2x}(\beta_2 - \alpha_2) + Q_{2y}(\alpha_1 - \gamma_2) + P_{2y}(\dot{\alpha}_1 - \dot{\gamma}_2)$$

$$+ \mu_{2x}\dot{\beta}_2 + N_{2x}(\dot{\beta}_2 - \dot{\alpha}_2) = 0 \qquad (2.31)$$

$$\gamma_2:\ I_{2y}\ddot{\gamma}_2 + G_{2y}\gamma_2 + M_{2y}(\gamma_2 - \alpha_1) + Q_{2x}(\alpha_2 - \beta_2) + P_{2x}(\dot{\alpha}_2 - \dot{\beta}_2)$$

$$+ \mu_{2y}\dot{\gamma}_2 + N_{2y}(\dot{\gamma}_2 - \dot{\alpha}_1) = 0 \qquad (2.32)$$

$$X_M:\ (M + \delta M_1 + \delta M_2)\ddot{x}_m + (\delta M_1\rho_1 + \delta M_2\rho_\rho)\ddot{\alpha}_1 - (\delta M_1R_1 \sin\alpha_3 + \delta M_2R_2 \sin(\alpha_3 + \Phi))\ddot{\alpha}_3$$

$$- (\delta M_1R_1 \sin\alpha_3 + \delta M_2R_2 \sin(\alpha_3 + \Phi))\ddot{\alpha}_3$$

$$+ K_{ix}(x_m - L_1\alpha_1 - U_1) + K_{2x}(x_m + L_2\alpha_1 - U_2)$$

$$+ D_{1y}(\dot{y}_m - L_1\dot{\alpha}_2 - \dot{V}_1) + D_{2y}(\dot{y}_m + L_2\dot{\alpha}_2 - \dot{V}_2)$$

$$+ C_{1x}(\dot{x}_m - L_1\dot{\alpha}_1 - \dot{U}_1) + C_{2x}(\dot{x}_m + L_2\dot{\alpha}_1 - \dot{U}_2)$$

$$+ R_{1y}(y_m - L_1\alpha_2 - V_1) + R_{2y}(y_m + L_2\alpha_2 - V_2)$$

$$= \left[\delta M_1R_1 \cos\alpha_3 + \delta M_2R_2 \cos(\alpha_3 + \Phi)\right]\dot{\alpha}_3^2 \qquad (2.33)$$

$$
\begin{aligned}
Y_m:\ & (M + \delta M_1 + \delta M_2)\ddot{Y}_m + (\delta M_1\rho_1 + \delta M_2\rho_2)\ddot{\alpha}_2 + \left[\delta M_1 R_1 \cos\alpha_3 + \delta M_2 R_2 \cos(\alpha_3 + \Phi)\right]\ddot{\alpha}_3 \\
& + K_{1y}(y_m - L_1\alpha_2 - V_1) + K_{2y}(y_m + L_2\alpha_2 - V_2) \\
& + D_{1x}(\dot{x}_m - L_1\dot{\alpha}_1 - \dot{U}_1) + D_{2x}(\dot{x}_m + L_2\dot{\alpha}_1 - \dot{U}_2) \\
& + C_{1y}(\dot{y}_m - L_1\dot{\alpha}_2 - \dot{V}_1) + C_{2y}(\dot{y}_m + L_2\dot{\alpha}_2 - \dot{V}_2) \\
& + R_{1x}(x_m - L_1\alpha_1 - U_1) + R_{2x}(x_m + L_2\alpha_1 - U_2) \\
& = \left[\delta M_1 R_1 \sin\alpha_3 + \delta M_2 R_2 \sin(\alpha_3 + \Phi)\right]\dot{\alpha}_3^2 \qquad (2.34)
\end{aligned}
$$

$$
\begin{aligned}
Z_m:\ & (M + \delta M_1 + \delta M_2)\,\ddot{z}_m - \left[\delta M_1 R_1 \sin\alpha_3 + \delta M_2 R_2 \sin(\alpha_3 + \Phi)\right]\ddot{\alpha}_2 \\
& - \left[\delta M_1 R_1 \cos\alpha_3 + \delta M_2 R_2 \cos(\alpha_3 + \Phi)\right]\ddot{\alpha}_1 + K_z z_m + C_z \dot{z}_m \\
& = \left[\delta M_1 R_1 \cos\alpha_3 + \delta M_2 R_2 \cos(\alpha_3 + \Phi)\right]\dot{\alpha}_2\dot{\alpha}_3 \\
& - \left[\delta M_1 R_1 \sin\alpha_3 + \delta M_2 R_2 \sin(\alpha_3 + \Phi)\right]\dot{\alpha}_1\dot{\alpha}_3 \qquad (2.35)
\end{aligned}
$$

$$
\begin{aligned}
\alpha_1:\ & I_T\left[\ddot{\alpha}_1 \cos^2\alpha_2 - \dot{\alpha}_1\dot{\alpha}_2 \sin 2\alpha_2\right] + I_p\left[\dot{\alpha}_2\dot{\alpha}_3 \cos\alpha_2 + \dot{\alpha}_1\dot{\alpha}_2 \sin\alpha_2 + \ddot{\alpha}_3 \sin\alpha_2 + \ddot{\alpha}_1 \sin^2\alpha_2 + \dot{\alpha}_1\dot{\alpha}_2 \sin\alpha_2 \cos\alpha_2\right] \\
& + \delta M_1\rho_1\left[\ddot{x}_m + \rho_1\ddot{\alpha}_1 - \ddot{\alpha}_3 R_1 \sin\alpha_3 - \dot{\alpha}_3^2 R_1 \cos\alpha_3\right] \\
& + \delta M_2\rho_2\left[\ddot{x}_m + \rho_2\ddot{\alpha}_1 - \ddot{\alpha}_3 R_2 \sin(\alpha_3 + \Phi) - \dot{\alpha}_3^2 R_2 \cos(\alpha_3 + \Phi)\right] \\
& + \delta M_1 R_1 \sin\alpha_3\dot{\alpha}_3\left[\dot{z}_m - R_1(\dot{\alpha}_2 \sin\alpha_3 + \dot{\alpha}_1 \cos\alpha_3)\right] \\
& - \delta M_1 R_1 \cos\alpha_3\left[\ddot{z}_m - R_1\ddot{\alpha}_2 \sin\alpha_3 - R_1\dot{\alpha}_2\dot{\alpha}_3 \cos\alpha_3 - R_1\ddot{\alpha}_1 \cos\alpha_3 + R_1\dot{\alpha}_1\dot{\alpha}_3 \sin\alpha_3\right] \\
& + \delta M_2\, R_2\dot{\alpha}_3 \sin(\alpha_3 + \Phi)\left\{\dot{z}_m - R_2\left[(\dot{\alpha}_2 \sin(\alpha_3 + \Phi) - \dot{\alpha}_1 \cos(\alpha_2 + \Phi)\right]\right\} \\
& - \delta M_2 R_2 \cos(\alpha_3 + \Phi)\left[\ddot{z}_m - R_2\ddot{\alpha}_2 \sin(\alpha_3 + \Phi) - R_2\dot{\alpha}_2\dot{\alpha}_3 \cos(\alpha_3 + \Phi) - R_2\ddot{\alpha}_1 \cos(\alpha_3 + \Phi) + R_2\dot{\alpha}_1\dot{\alpha}_3 \sin(\alpha_3 + \Phi)\right] \\
& - K_{1x}L_1(x_m - L_1\alpha_1 - U_1) + K_{2x}L_2(x_m + L_2\alpha_1 - U_2) - D_{1y}L_1(\dot{y}_m - L_1\dot{\alpha}_2 - \dot{V}_1) \\
& + D_{2y}L_2(\dot{y}_m + L_2\dot{\alpha}_2 - \dot{V}_2) + M_{1y}(\alpha_1 - \gamma_1) + M_{2y}(\alpha_1 - \gamma_2) + P_{1x}(\dot{\beta}_1 - \dot{\alpha}_2) + P_{2x}(\dot{\beta}_2 - \dot{\alpha}_2) \\
& - C_{1x}L_1(\dot{x}_m - L_1\dot{\alpha}_1 - \dot{U}_1) + C_{2x}L_2(\dot{x}_m + L_2\dot{\alpha}_1 - \dot{U}_2) - R_{1y}L_1(y_m - L_1\alpha_2 - V_1) + R_{2y}L_2(y_m + L_2\alpha_2 - V_2) \\
& + N_{1y}(\dot{\alpha}_1 - \dot{\gamma}_1) + N_{2y}(\dot{\alpha}_1 - \dot{\gamma}_2) + Q_{1x}(\beta_1 - \alpha_2) + Q_{2x}(\beta_2 - \alpha_2) = 0 \qquad (2.36)
\end{aligned}
$$

$$\alpha_2:\ I_T(\ddot{\alpha}_2 + \dot{\alpha}_1^2 \sin\alpha_2 \cos\alpha_2) - I_p\dot{\alpha}_1 \cos\alpha_2(\dot{\alpha}_3 + \dot{\alpha}_3 \sin\alpha_2)$$

$$+ \delta M_1\rho_1\left[\ddot{Y}_m + \ddot{\alpha}_2\rho_1 + \ddot{\alpha}_3 R_1 \cos\alpha_3 - \dot{\alpha}_3^2 R_1 \sin\alpha_3\right]$$

$$+ \delta M_2\rho_2\left[\ddot{Y}_m + \ddot{\alpha}_2\rho_2 + \ddot{\alpha}_3 R_2 \cos(\alpha_3 + \Phi) - \dot{\alpha}_3^2 R_2 \sin(\alpha_3 + \Phi)\right]$$

$$- \delta M_1 R_1 \cos\alpha_3 \dot{\alpha}_3\left[\dot{z}_m - R_1(\dot{\alpha}_2 \sin\alpha_3 + \dot{\alpha}_1 \cos\alpha_3)\right]$$

$$- \delta M_1 R_1 \sin\alpha_3\left[\ddot{z}_m - R_1(\ddot{\alpha}_2 \sin\alpha_3 + \dot{\alpha}_2\dot{\alpha}_3 \cos\alpha_3 + \ddot{\alpha}_1 \cos\alpha_3 - \dot{\alpha}_1\dot{\alpha}_3 \sin\alpha_3)\right]$$

$$- \delta M_2 R_2\dot{\alpha}_3 \cos(\alpha_3 + \Phi)\left\{\dot{z}_m - R_2\left[\dot{\alpha}_2 \sin(\alpha_3 + \Phi) + \dot{\alpha}_1 \cos(\alpha_3 + \Phi)\right]\right\}$$

$$- \delta M_2 R_2 \sin(\alpha_3 + \Phi)\left[\ddot{z}_m - R_2\ddot{\alpha}_2 \sin(\alpha_3 + \Phi) - R_2\dot{\alpha}_2\dot{\alpha}_3 \cos(\alpha_3 + \Phi)\right.$$

$$\left. - R_2\ddot{\alpha}_1 \cos(\alpha_3 + \Phi) + R_2\dot{\alpha}_1\dot{\alpha}_3 \sin(\alpha_3 + \Phi)\right]$$

$$- K_{1y}L_1(y_m - L_1\alpha_2 - V_1) + K_{2y}L_2(y_m + L_2\alpha_2 - V_2)$$

$$- D_{1x}L_1(\dot{x}_m - L_1\dot{\alpha}_1 - \dot{U}_1) + D_{2x}L_2(x_m + L_2\dot{\alpha}_1 - \dot{U}_2)$$

$$+ M_{1x}(\alpha_2 - \beta_1) + M_{2x}(\alpha_2 - \beta_2) + P_{1y}(\dot{\gamma} - \dot{\alpha}_1) + P_{2y}(\dot{\gamma}_2 - \dot{\alpha}_1)$$

$$- C_{1y}L_1(\dot{y}_m - L_1\dot{\alpha}_2 - \dot{V}_1) + C_{2y}L_2(\dot{y}_m + L_2\dot{\alpha}_2 - \dot{V}_2)$$

$$+ R_{1x}L_1(x_m - L_1\alpha_1 - U_1) + R_{2x}L_2(x_m + L_2\alpha_1 - U_2) + N_{1x}(\dot{\alpha}_2 - \dot{\beta}_1)$$

$$+ N_{2x}(\dot{\alpha}_2 - \dot{\beta}_2) + Q_{1y}(\gamma_1 - \alpha_1) + Q_{2y}(\gamma_2 - \alpha_1) = 0 \qquad (2.37)$$

$$\alpha_3:\ I_p \frac{d}{dt}(\dot{\alpha}_3 + \dot{\alpha}_1 \sin\alpha_2) - \delta M_1 R_1 \frac{d}{dt}\left[\sin\alpha_3(\dot{x}_m + \rho_1\dot{\alpha}_1 - R_1\dot{\alpha}_3 \sin\alpha_3)\right]$$

$$+ \delta M_1 R_1 \frac{d}{dt}\left[\cos\alpha_3(\dot{Y}_m + \rho_1\dot{\alpha}_2 + R_1\dot{\alpha}_3 \cos\alpha_3)\right]$$

$$- \delta M_2 R_2 \frac{d}{dt}\left\{\sin(\alpha_3 + \Phi)\left[\dot{x}_m + \rho_2\dot{\alpha}_1 - R_2\dot{\alpha}_3 \sin(\alpha_3 + \Phi)\right]\right\}$$

$$- \delta M_2 R_2 \frac{d}{dt}\left\{\cos(\alpha_3 + \Phi)\left[\dot{y}_m + \rho_2\dot{\alpha}_2 + R_2\dot{\alpha}_3 \cos(\alpha_3 + \Phi)\right]\right\}$$

$$- \delta M_1 R_1\dot{\alpha}_3 \cos\alpha_3\left[(\dot{x}_m + \rho_1\dot{\alpha}_1 - R_1\dot{\alpha}_3 \sin\alpha_3)\right]$$

$$- \delta M_1 R_1\dot{\alpha}_3 \sin\alpha_3\left[\dot{y}_m + \rho_1\dot{\alpha}_2 + R_1\dot{\alpha}_3 \cos\alpha_3\right]$$

$$- \delta M_2 R_2\dot{\alpha}_3 \sin(\alpha_3 + \Phi)\left[\dot{x}_m + \rho_2\dot{\alpha}_1 - R_2\dot{\alpha}_3 \sin(\alpha_3 + \Phi)\right]$$

$$- \delta M_2 R_2\dot{\alpha}_3 \sin(\alpha_3 + \Phi)\left[\dot{y}_m + \rho_2\dot{\alpha}_2 + R_2\dot{\alpha}_3 \cos(\alpha_3 + \Phi)\right] = Q\alpha_3 \qquad (2.38)$$

The general rotor equations of motion (2.25) to (2.38) including rotor acceleration are highly nonlinear and represent a difficult problem to solve. These equations may be simplified considerably if we assume constant rotor angular velocity and small bearing displacements.

Hence,

$$\mathring{\alpha}_3 = \omega = \text{constant}, \qquad \alpha_3 = \omega t$$

$$\sin \alpha_1 \approx \alpha_1 = \frac{X_2 - X_1}{L} << 1$$

and

$$\sin \alpha_2 \approx \alpha_2 = \frac{Y_2 - Y_1}{L} << 1$$

The unbalance masses are considered small in comparison to the rotor mass. Hence,

$$\frac{\delta M_1}{M} \quad \text{or} \quad \frac{\delta M_2}{M} << 1$$

If it is further assumed that the bearing housing is rigid, then equations (2.25) to (2.32) do not enter into the system equations. The resulting five linearized equations of motion are as follows:

$$X_m: \ \frac{M}{L}(L_1\ddot{x}_2 + L_2\ddot{x}_1) + K_{1x}x_1 + K_{2x}x_2 + C_{1x}\dot{x}_1 + C_{2x}\dot{x}_2$$
$$+ R_{1y}y_1 + R_{2y}y_2 + D_{1y}\dot{y}_1 + D_{2y}\dot{y}_1$$
$$= \delta M_1\omega^2 R_1 \cos \omega t + \delta M_2\omega^2 R_2 \cos(\omega t + \Phi) \tag{2.39}$$

$$Y_m: \ \frac{M}{L}(L_1\ddot{y}_2 + L_2\ddot{y}_2) + K_{1y}y_1 + K_{2y}y_2 + C_{1y}\dot{y}_1 + C_{2y}\dot{y}_2$$
$$+ R_{1x}x_1 + R_{2x}x_2 + D_{1x}\dot{x}_1 + D_{2x}\dot{x}_2$$
$$= \delta M_1\omega^2 R_1 \sin \omega t + \delta M_2\omega^2 R_2 \sin(\omega t + \Phi) \tag{2.40}$$

$$Z_m: \ M\ddot{Z}_m + C_z\dot{Z}_m + K_z Z_m = 0 \tag{2.41}$$

$$\alpha_1: \; I_T \frac{(\ddot{x}_2 - \ddot{x}_1)}{L} + I_p\omega \frac{(\dot{y}_2 - \dot{y}_1)}{L} + D_{2y}L_2\dot{y}_2 - D_{1y}L_1\dot{y}_1$$

$$+ C_{2x}L_2\dot{x}_2 - C_{1x}L_1\dot{x}_1 + K_{2x}L_2x_2 - K_{1x}L_1x_1 + R_{2y}L_2y_2 - R_{1y}L_1y_1$$

$$= \delta M_1\rho_1\omega^2R_1 \cos \omega t + \delta M_2\rho_2\omega^2R_2 \cos(\omega t + \Phi) \qquad (2.42)$$

$$\alpha_2: \; I_T \frac{(\ddot{y}_2 - \ddot{y}_1)}{L} - I_p\omega \frac{(\dot{x}_2 - \dot{x}_1)}{L} + D_{2x}L_2\dot{x}_2 - D_{1x}L_1\dot{x}_1$$

$$+ C_{2y}L_2\dot{y}_2 - C_{1y}L_1\dot{y}_1 + K_{2y}L_2y_2 - K_{1y}L_1y_1 + R_{2x}L_2x_2 - R_{1x}L_1x_1$$

$$= \delta M_1\rho_1\omega^2R_1 \sin \omega t + \delta M_2\rho_2\omega^2R_2 \sin(\omega t + \Phi) \qquad (2.43)$$

Due to the assumption of constant rotor angular velocity, equation (2.38) identically reduces to zero. It may be further observed that equation (2.41) is uncoupled from the rest of the system equations; hence, it may be solved independently.

If we now substitute

$$I_T = MR_T^2 \qquad I_p = MR_p^2$$

where R_T and R_P are the transverse and polar radii of gyration of the rotor respectively, and

$$\frac{L_1}{L} = \ell_1 \qquad \frac{L_2}{L} = \ell_2$$

$$\frac{D_{1y}}{M} = \bar{D}_{1y} \qquad \frac{D_{2y}}{M} = \bar{D}_{2y} \qquad \frac{D_{1x}}{M} = \bar{D}_{1x} \qquad \frac{D_{2x}}{M} = \bar{D}_{2x}$$

$$\frac{R_{1y}}{M} = \bar{R}_{1y} \qquad \frac{R_{2y}}{M} = \bar{R}_{2y} \qquad \frac{R_{1x}}{M} = \bar{R}_{1x} \qquad \frac{R_{2x}}{M} = \bar{R}_{2x}$$

$$\frac{C_{1y}}{M} = \bar{C}_{1y} \qquad \frac{C_{2y}}{M} = \bar{C}_{2y} \qquad \frac{C_{1x}}{M} = \bar{C}_{1x} \qquad \frac{C_{2x}}{M} = \bar{C}_{2x}$$

$$\frac{K_{1y}}{M} = \bar{K}_{1y} \qquad \frac{K_{2y}}{M} = \bar{K}_{2y} \qquad \frac{K_{1x}}{M} = \bar{K}_{1x} \qquad \frac{K_{2x}}{M} = \bar{K}_{2x}$$

$$\left(\frac{R_T}{L}\right)^2 = \bar{R}_T \qquad \left(\frac{R_p}{M}\right)^2 = \bar{R}_p$$

The equations (2.39), (2.40), (2.42), and (2.43) reduce to

$$x_m:\ \ell_1\ddot{x}_2 + \ell_2\ddot{x}_1 + \bar{D}_{1y}\dot{y}_1 + \bar{D}_{2y}\dot{y}_2 + \bar{C}_{1x}\dot{x}_1 + \bar{C}_{2x}\dot{x}_2 + \bar{K}_{1x}x_1 + \bar{K}_{2x}x_2 + \bar{R}_{1y}y_1 + \bar{R}_{2y}y_2$$

$$= \frac{1}{M}\left[\delta M_1\omega^2 R_1 \cos \omega t + \delta M_2\omega^2 R_2 \cos(\omega t + \Phi)\right] \qquad (2.44)$$

$$y_m:\ \ell_1\ddot{y}_2 + \ell_2\ddot{y}_1 + \bar{D}_{1x}\dot{x}_1 + \bar{D}_{2x}\dot{x}_2 + \bar{C}_{1y}\dot{y}_1 + \bar{C}_{2y}\dot{y}_2 + \bar{K}_{1y}y_1 + \bar{K}_{2y}y_2 + \bar{R}_{1x}x_1 + \bar{R}_{2x}x_2$$

$$= \frac{1}{M}\left[\delta M_1\omega^2 R_1 \sin \omega t + \delta M_2\omega^2 R_2 \sin(\omega t + \Phi)\right] \qquad (2.45)$$

$$\alpha_1:\ \bar{R}_T(\ddot{x}_2 - \ddot{x}_1) + \bar{R}_p\omega(\dot{y}_2 - \dot{y}_1) + \bar{D}_{2y}\ell_2\dot{y}_2 - \bar{D}_{1y}\ell_1\dot{y}_1 + \bar{C}_{2x}\ell_2\dot{x}_2 - \bar{C}_{1x}\ell_1\dot{x}_1$$

$$+ \bar{K}_{2x}\ell_2 x_2 - \bar{K}_{1x}\ell_1 x_1 + \bar{R}_{2y}\ell_2 y_2 - \bar{R}_{1y}\ell_1 y_1$$

$$= \frac{1}{ML}\left[\delta M_1\omega^2 R_1\rho_1 \cos \omega t + \delta M_2\omega^2\rho_2 R_2 \cos(\omega t + \Phi)\right] \qquad (2.46)$$

$$\alpha_2:\ \bar{R}_T(\ddot{y}_2 - \ddot{y}_1) - \bar{R}_p\omega(\dot{x}_2 - \dot{x}_1) + \bar{D}_{2x}\ell_2\dot{x}_2 - \bar{D}_{1x}\ell_1\dot{x}_1 + \bar{C}_{2y}\ell_2\dot{y}_2 - \bar{C}_{1y}\ell_1\dot{y}_1$$

$$+ \bar{K}_{2y}\ell_2 y_2 - \bar{K}_{1y}\ell_1 y_1 + \bar{R}_{2x}\ell_2 x_2 - \bar{R}_{1x}\ell_1 x_1$$

$$= \frac{1}{ML}\left[\delta M_1\omega^2\rho_1 R_1 \sin \omega t + \delta M_2\omega^2\rho_2 R_2 \sin(\omega t + \Phi)\right] \qquad (2.47)$$

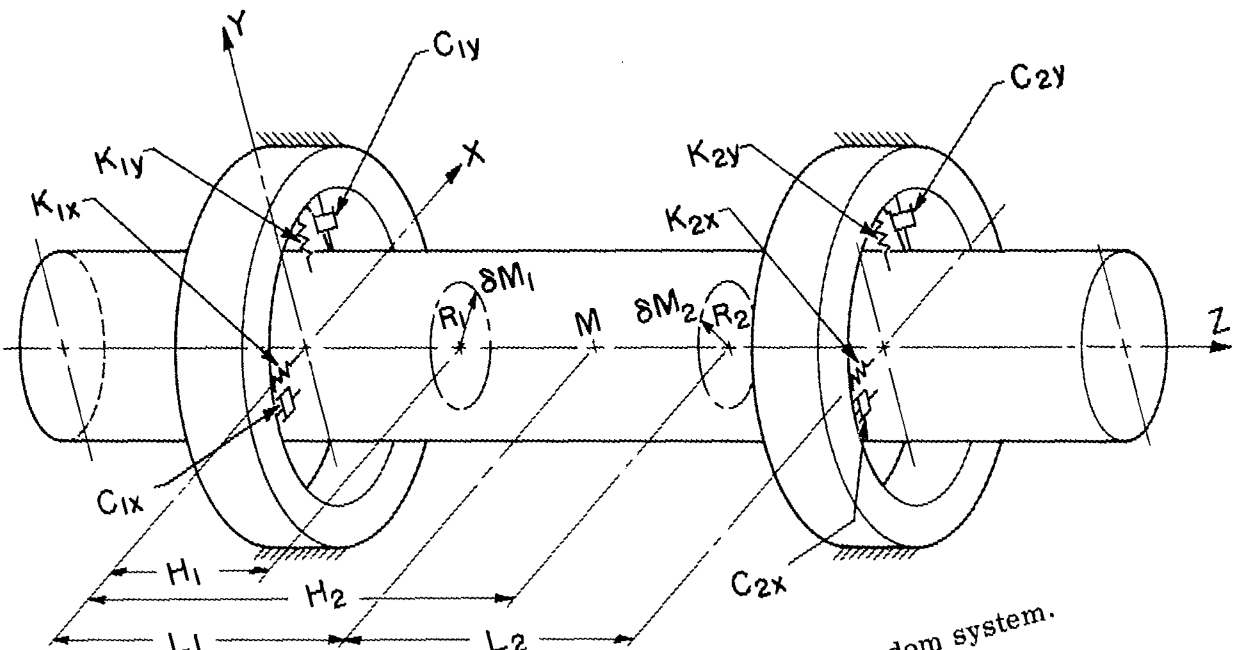


Figure 5. - Schematic diagram of four degree freedom system.

PART III

STEADY STATE SOLUTION OF THE EQUATIONS OF MOTION - FOUR DEGREES OF FREEDOM SYSTEM

3.01

The complete solution of the linearized equations of motion (2.44) to (2.47) will be the sum of the general solution of the homogeneous equation (i.e., the general solution of the equation with right-hand side zero) and the particular solution of the complete differential equations. This particular solution describes, on the other hand, the forced vibrations caused by the rotor unbalance.

Equations (2.44) to (2.47) will be satisfied if we assume a harmonic solution of the following form:

$$\left.\begin{aligned} x_1 &= x[1] \cos \omega t + x[2] \sin \omega t \\ x_2 &= x[3] \cos \omega t + x[4] \sin \omega t \\ y_1 &= x[5] \cos \omega t + x[6] \sin \omega t \\ y_2 &= x[7] \cos \omega t + x[8] \sin \omega t \end{aligned}\right\} \quad (3.1)$$

Substituting these in equations (2.44) to (2.47) we obtain the resulting equations in the following matrix form after equating the coefficients of $\cos \omega t$ and $\sin \omega t$ in the four equations.

$$
\begin{bmatrix}
\bar{K}_{1x} - \ell_2\omega^2 & \bar{C}_{1x}\omega & \bar{K}_{2x} - \ell_1\omega^2 & \bar{C}_{2x}\omega & \bar{R}_{1y} & \bar{D}_{1y}\omega & \bar{R}_{2y} & \bar{D}_{2y}\omega \\
C_{1x}\omega & \bar{K}_{1x} - \ell_2\omega^2 & -\bar{C}_{2x}\omega & \bar{K}_{2x} - \ell_1\omega^2 & -\bar{D}_{1y}\omega & \bar{R}_{1y} & -\bar{D}_{2y}\omega & \bar{R}_{2y} \\
\bar{R}_{1x} & \bar{D}_{1x}\omega & \bar{R}_{2x} & \bar{D}_{2x}\omega & \bar{K}_{1y} - \ell_2\omega^2 & \bar{C}_{1y}\omega & \bar{K}_{2y} - \ell_1\omega^2 & \bar{C}_{2y}\omega \\
-\bar{D}_{1x}\omega & \bar{R}_{1x} & -\bar{D}_{2x}\omega & \bar{R}_{2x} & -\bar{C}_{1y}\omega & \bar{K}_{1y} - \ell_2\omega^2 & -\bar{C}_{2y}\omega & \bar{K}_{2y} - \ell_1\omega^2 \\
\bar{R}_T\omega^2 - \bar{K}_{1x}\ell_1 & -\bar{C}_{1x}\ell_1\omega & -\bar{R}_T\omega^2 + \bar{K}_{2x}\ell_2 & \bar{C}_{2x}\ell_2\omega & -\bar{R}_{1y}\ell_1 & -\bar{R}_p\omega^2 - \bar{D}_{1y}\ell_1\omega & \bar{R}_{2y}\ell_2 & \bar{R}_p\omega^2 + \bar{D}_{2y}\ell_2\omega \\
\bar{C}_{1x}\ell_1\omega & \bar{R}_T\omega^2 - \bar{K}_{1x}\ell_1 & -\bar{C}_{2x}\ell_2\omega & -\bar{R}_T\omega^2 + \bar{K}_{2x}\ell_2 & \bar{R}_p\omega^2 + \bar{D}_{1y}\ell_1\omega & -\bar{R}_{1y}\ell_1 & -\bar{R}_p\omega^2 - \bar{D}_{2y}\ell_2\omega & \bar{R}_{2y}\ell_2 \\
-\bar{R}_{1x}\ell_1 & \bar{R}_p\omega^2 - \bar{D}_{1x}\ell_1\omega & \bar{R}_{2x}\ell_2 & -\bar{R}_p\omega^2 + \bar{D}_{2x}\ell_2\omega & \bar{R}_T\omega^2 - K_{1y}\ell_1 & -\bar{C}_{1y}\ell_1\omega & -\bar{R}_T\omega^2 + \bar{K}_{2y}\ell_2 & \bar{C}_{2y}\ell_2\omega \\
-\bar{R}_p\omega^2 + \bar{D}_{1x}\ell_1\omega & -\bar{R}_{1x}\ell_1\omega & \bar{R}_1\omega^2 - \bar{D}_{2x}\ell_2\omega & \bar{R}_{2x}\ell_2 & \bar{C}_{1y}\ell_1\omega & \bar{R}_T\omega^2 - \bar{K}_{1y}\ell_1 & -\bar{C}_{2y}\ell_2\omega & -\bar{R}_T\omega^2 + \bar{K}_{2y}\ell_2
\end{bmatrix}
\begin{bmatrix}
x[1] \\ x[2] \\ x[3] \\ x[4] \\ x[5] \\ x[6] \\ x[7] \\ x[8]
\end{bmatrix}
=
\begin{bmatrix}
\frac{1}{M}\left[\delta M_1\omega^2 R_1 + \delta M_2\omega^2 R_2\cos\Phi\right] \\
-\frac{\delta M_2}{M}\omega^2 R_2\sin\Phi \\
\frac{\delta M_2}{M}\omega^2 R_2\sin\Phi \\
\frac{\delta M_1}{M}\omega^2 R_1 + \frac{\delta M_2}{M}\omega^2 R_2\cos\Phi \\
\frac{\delta M_1}{ML}\rho_1\omega^2 R_1 \quad \frac{\delta M_2}{ML}\omega^2\rho_2 R_2\cos\Phi \\
\frac{-\delta M_2}{ML}\omega^2\rho_2 R_2\sin\Phi \\
\frac{\delta M_2}{ML}\omega^2 R_2\rho_2\sin\Phi \\
\frac{\delta M_1}{ML}\omega^2\rho_1 R_1 + \frac{\delta M_2}{ML}\omega^2\rho_2 R_2\cos\Phi
\end{bmatrix}
\tag{3.2}
$$

The solution of the above algebraic simultaneous equations will yield x[1], x[2], . . . , x[8]. The amplitudes of x_1, x_2, y_1, and y_2 are

$$
\left.
\begin{aligned}
|x_1| &= \sqrt{x[1]^2 + x[2]^2} \\
|x_2| &= \sqrt{x[3]^2 + x[4]^2} \\
|y_1| &= \sqrt{x[5]^2 + x[6]^2} \\
|y_2| &= \sqrt{x[7]^2 + x[8]^2}
\end{aligned}
\right\}
\tag{3.3}
$$

The angular amplitudes can be obtained from

$$\alpha_1 = \frac{x_2 - x_1}{L}$$

$$= \frac{1}{L}\left[(x[3] - x[1])\cos \omega t + (x[4] - x[2]) \sin \omega t\right]$$

$$\therefore |\alpha_1| = \frac{1}{L}\sqrt{(x[3] - x[2])^2 + (x[4] - x[2])^2} \qquad (3.4)$$

Similarly,

$$|\alpha_2| = \frac{1}{L}\sqrt{(x[7] - x[5])^2 + (x[8] - x[6])^2} \qquad (3.5)$$

The steady state solution is therefore given by

$$\left.\begin{aligned} x_1 &= |x_1| \cos(\omega t - \psi_1) \\ x_2 &= |x_2| \cos(\omega t - \psi_2) \\ y_1 &= |y_1| \sin(\omega t - \psi_3) \\ y_2 &= |y_2| \sin(\omega t - \psi_4) \end{aligned}\right\} \qquad (3.6)$$

where,

$$\left.\begin{aligned} \psi_1 &= \arctan\left[\frac{x[2]}{x[1]}\right] \\ \psi_2 &= \arctan\left[\frac{x[4]}{x[3]}\right] \\ \psi_3 &= \arctan\left[-\frac{x[5]}{x[6]}\right] \\ \psi_4 &= \arctan\left[-\frac{x[7]}{x[8]}\right] \end{aligned}\right\} \qquad (3.7)$$

Derivation of Phase Angles

The resultant exciting force due to the two planes of unbalance can be resolved into two components.

<u>x-component</u>

$$= \delta M_1 \omega^2 R_1 \cos \omega t + \delta M_2 \omega^2 R_2 \cos(\omega t + \Phi)$$

$$= \sqrt{(\delta M_1 R_1)^2 + (\delta M_2 R_2)^2 + 2\delta M_1 R_1 \delta M_2 R_2 \cos \Phi}\, \omega^2 \cos(\omega t + \omega)$$

$$= M e_u \omega^2 \cos(\omega t + \psi) \qquad (3.8)$$

where

$$\psi = \arctan\left[\frac{\delta M_2 R_2 \sin \Phi}{\delta M_1 R_1 + \delta M_2 R_2 \cos \Phi}\right] \qquad (3.9)$$

and

$$e_u = \frac{1}{M} \sqrt{(\delta M_1 R_1)^2 + (\delta M_2 R_2)^2 + 2\delta M_1 R_1 \delta M_2 R_2 \cos \Phi} \qquad (3.10)$$

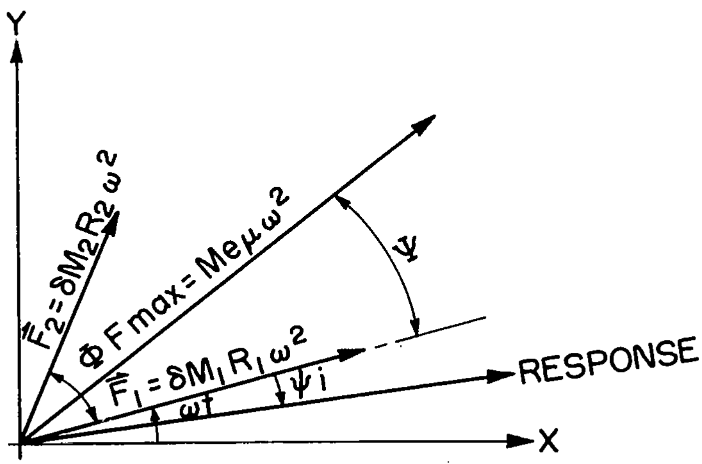


Figure 6. - Diagram of unbalance force and rotor response phase relation.

Similarly,

<u>y-component</u>

$$= Me_u\omega^2 \sin(\omega t + \psi)$$

From equation (3.6) it is observed that the response lags the angular velocity by an angle ψ_i (i = 1, 2, 3, 4). If the response vector is superimposed on the excitation vectors as shown in figure 6, it can be observed that the response lags the resultant excitation due to unbalance by an angle $(\psi + \psi_i)$. If, however, in computation, this angle turns out to be negative, then the response instead of lagging will lead the excitation.

The resultant phase angles of the cylindrical responses with respect to the unbalance is given by

$$\left.\begin{aligned} \psi_{x1} &= \psi + \psi_1 \\ \psi_{x2} &= \psi + \psi_2 \\ \psi_{y1} &= \psi + \psi_3 \\ \psi_{y2} &= \psi + \psi_4 \end{aligned}\right\} \tag{3.11}$$

The resultant moment about the x and y axes due to rotor unbalance will have to be calculated in order to compute the phase difference between the conical response and the excitation moments.

The moment due to unbalance about y axis is given by

$$M_y = \delta M_1 R_1 \rho_1 \omega^2 \cos \omega t + \delta M_2 R_2 \rho_2 \omega^2 \cos(\omega t + \Phi)$$

$$= \sqrt{(\rho_1 R_1 \delta M_1)^2 + (\rho_2 R_2 \delta M_2)^2 + 2\rho_1 \rho_2 R_1 R_2 \delta M_1 \delta M_2 \cos \Phi} \cdot \omega^2 \cos(\omega t + \psi_t) \tag{3.12}$$

and the moment due to unbalance about x axis is

$$M_x = -\sqrt{(\rho_1 R_1 \delta M_1)^2 + (\rho_2 R_2 \delta M_2)^2 + 2\rho_1 \rho_2 R_1 R_2 \delta M_1 \delta M_2 \cos \Phi} \cdot \omega^2 \sin(\omega t + \psi_t) \tag{3.13}$$

where

$$\psi_t = \arctan\left[\frac{\rho_2 R_2 \delta M_2 \sin\Phi}{\rho_1 R_1 \delta M_1 + \rho_2 R_2 \delta M_2 \cos\Phi}\right] \tag{3.14}$$

Now,

and

$$\left.\begin{aligned} \alpha_1 &= |\alpha| \cos(\omega t - \psi_5) \\ \alpha_2 &= |\alpha| \sin(\omega t - \psi_6) \end{aligned}\right\} \tag{3.15}$$

Hence the phase lags between the conical responses α_1 and α_2, with respect to the exciting moment, is given by

$$\left.\begin{aligned} \psi_{\alpha 1} &= \psi_t + \psi_5 \\ \psi_{\alpha 2} &= \psi_t + \psi_6 \end{aligned}\right\} \tag{3.16}$$

3.03 Calculation of Force Transmitted to Bearings and of Phase Angles Between Transmitted Force and Excitation

Force transmitted in the first bearing in x-direction is given by

$$\begin{aligned} F_{x1} &= C_{1x}\dot{x}_1 + K_{1x}x_1 + D_{1y}\dot{y}_1 + R_{1y}y_1 \\ &= \cos\omega t\left[x[2]\omega C_{1x} + x[1]K_{1x} + x[6]\omega D_{1y} + x[5]R_{1y}\right] \\ &\quad + \sin\omega t\left[-x[1]\omega C_{1x} + x[2]K_{1x} - x[5]\omega D_{1y} + x[6]R_{1y}\right] \\ &= A_{x1}\cos\omega t + B_{x1}\sin\omega t \end{aligned}$$

where

$$A_{x1} = x[2]\omega C_{1x} + x[1]K_{1x} + x[6]\omega D_{1y} + x[5]R_{1y}$$

$$B_{x1} = -x[1]\omega C_{1x} + x[2]K_{1x} - x[5]\omega D_{1y} + x[6]R_{1y}$$

$$\therefore |F_{x1}| = \sqrt{A_{x1}^2 + B_{x1}^2}$$

and

$$F_{x1} = |F_{x1}| \cos(\omega t - \psi_7)$$

where

$$\psi_7 = \arctan\left[\frac{B_{x1}}{A_{x1}}\right]$$

Similar expressions for the force transmitted in x- and y-directions in the first and second bearings can be obtained from

$$F_{y1} = C_{1y}\dot{y}_1 + K_{1y}y_1 + D_{1x}\dot{x}_1 + R_{1x}x_1 = |F_{y1}|\sin(\omega t + \psi_8)$$

$$F_{x2} = C_{2x}\dot{x}_2 + K_{2x}x_2 + D_{2y}\dot{y}_2 + R_{2y}y_2 = |F_{s2}|\cos(\omega t - \psi_9)$$

$$F_{y2} = C_{2y}\dot{y}_2 + K_{2y}y_2 + D_{2x}\dot{x}_2 + R_{2x}x_2 = |F_{y2}|\sin(\omega t + \psi_{10})$$

where

$$|F_{y1}| = \sqrt{A_{y1}^2 + B_{y1}^2}$$

$$|F_{x2}| = \sqrt{A_{x2}^2 + B_{x2}^2}$$

$$|F_{y2}| = \sqrt{A_{y2}^2 + B_{y2}^2}$$

$$A_{y1} = x[6]\omega C_{1y} + x[5]K_{1y} + x[2]\omega D_{1x} + x[1]R_{1x}$$

$$B_{y1} = -x[5]\omega C_{1y} + x[6]K_{1y} - x[1]\omega D_{1x} + x[2]R_{1x}$$

$$A_{x2} = x[4]\omega C_{2x} + x[3]K_{2x} + x[8]\omega D_{2y} + x[7]R_{2y}$$

$$B_{x2} = -x[3]\omega C_{2x} + x[4]K_{2x} - x[7]\omega D_{2y} + x[8]R_{2y}$$

$$A_{y2} = x[8]\omega C_{2y} + x[7]K_{2y} + x[4]\omega D_{2x} + x[3]R_{2x}$$

$$B_{y2} = -x[7]\omega C_{2y} + x[8]K_{2y} - x[3]\omega D_{2x} + x[4]R_{2x}$$

and

$$\psi_8 = \arctan \left[\frac{A_{y1}}{B_{y1}} \right]$$

$$\psi_9 = \arctan \left[\frac{B_{x2}}{A_{x2}} \right]$$

$$\psi_{10} = \arctan \left[\frac{A_{y2}}{B_{y2}} \right]$$

PART IV

ROTOR DYNAMIC UNBALANCE ANALYSIS

4.01 Computer Programs for Rotor Steady State Solution

Three versions of the steady state unbalance response computer programs were developed. These programs are ROTOR4, ROTOR4P, and ROTOR4M. The first version, ROTOR4, produces tables of the various bearing amplitudes, forces, and phase angles. The second version will plot any of these quantities as a function of speed. The third version prints out only the rotor maximum amplitude at any particular shaft location specified.

Detailed description of each of these programs is given in sections 4.02 and 4.03.

4.02 Computer Programs ROTOR4 and ROTOR4P

A computer program was written to obtain the rotor steady state behavior. The equations of motion (2.44) to (2.47) were solved for certain increments in speed. It is to be noted that the general rigid body system requires six degrees of freedom. However, with the assumption of constant angular velocity of the rotor and the rotor axial equation of motion being uncoupled from the rest of the system equations, reduces the system to one with four degrees of freedom. The equations considered are linearized, based on the assumption that the rotor amplitudes are small, and the terms such as $\delta M_1 \ddot{x}_m$, $\delta M_1 \ddot{y}_m$ are small in comparison to $M\ddot{x}_m$, $M\ddot{y}_m$, etc. This program evaluates the rotor behavior due to certain unbalance along different location and at different planes of the rotor.

In addition to computing the rotor amplitudes, their phase lag or lead with respect to the excitation and the amount of force transmitted, etc., this program also has provision for computing the amplitudes and their phase angles as functions of unbalance force at any arbitrary location along the rotor length.

The program requires the following to be read as input data:

Card 1

1. WO - Initial speed rps
2. DW - Increment in speed, rps
3. WM - Final speed, rps

Card 2

1. L - Length between the bearings, in.
2. L1 - Distance from first bearing to mass center, in.
3. L2 - Distance from second bearing to mass center, in.
4. W - Rotor weight, lb
5. IP - Polar moment of inertia of the rotor, lb-in. -sec^2
6. IT - Transverse moment of inertia of the rotor about mass center, lb-in. -sec^2

Card 3

1. WM1 - First unbalance weight, lb
2. WM2 - Second unbalance weight, lb
3. H1 - Distance from first bearing to first unbalance, in.
4. H2 - Distance from first bearing to second unbalance, in.
5. PHI - Phase angles between unbalance planes, deg
6. R1 - Radius of first unbalance location, in.
7. R2 - Radius of second unbalance location, in.

Card 4

1. N - Number of places other than the bearing locations where displacements are to be measured
2. LZ1 - Distance from first bearing to first probe, in.
3. LZ2 - Distance from first bearing to second probe, in.

Card 5

1. K1X - First bearing stiffness in x-direction, lb/in.
2. K2X - Second bearing stiffness in x-direction, lb/in.
3. K1Y - First bearing stiffness in y-direction, lb/in.
4. K2Y - Second bearing stiffness in y-direction, lb/in.

Card 6

1. C1X - First bearing damping coefficient in x-direction, lb-sec/in.
2. C2X - Second bearing damping coefficient in x-direction, lb-sec/in.
3. C1Y - First bearing damping coefficient in y-direction, lb-sec/in.
4. C2Y - Second bearing damping coefficient in y-direction, lb-sec/in.

Card 7

1. D1X - Cross coupling damping coefficient, lb-sec/in.
2. D2X - Cross coupling damping coefficient, lb-sec/in.
3. D1Y - Cross coupling damping coefficient, lb-sec/in.
4. D2Y - Cross coupling damping coefficient, lb-sec/in.

Card 8

1. R1X - Cross coupling stiffness, lb/in.
2. R2X - Cross coupling stiffness, lb/in.
3. R1Y - Cross coupling stiffness, lb/in.
4. R2Y - Cross coupling stiffness, lb/in.

The data cards are in free field format. A comma should separate all data entries. A comma is needed after the last data entry.

The heading printouts of the input data are as follows:

Line	
1	L, L1, L2, H1
2	H2, W, WM1, WM2
3	K1X, K2X, K1Y, K2Y
4	C1X, C2X, C1Y, C2Y
5	R1X, R2X, R1Y, R2Y
6	D1X, D2X, D1Y, D2Y
7	IP, IT, R1, R2
8	PHI

The output data are printed out as follows:

Column	
1	Speed, rps
2	Displacement at bearing 1 in x- or y-direction
3	Displacement at bearing 2 in x- or y-direction
4	Phase angle of displacements at first bearing with respect to excitation
5	Phase angle of displacements at second bearing with respect to excitation
6	Angular displacements in x-z or y-z plane
7	Phase angles of angular displacements with respect to exciting moment
8	Force transmitted to bearing 1 in x- or y-direction
9	Force transmitted to bearing 2 in x- or y-direction
10	Phase angles of the force transmitted to bearing 1 with respect to exciting force
11	Phase angles of the force transmitted to bearing 2 with respect to exciting force

The printout of the output at any arbitrary location is as follows:

Column	
1	LZ - The probe location along the rotor from first bearing
2	XL - Amplitude in the x-direction at arbitrary location
3	YL - Amplitude in the y-direction at arbitrary location
4	PXL - Phase angle of the amplitude in x-direction at arbitrary location with respect to excitation

Column

5	PYL - Phase angle of the amplitude in the y-direction at arbitrary location with respect to excitation
6	SPEED, rps

In addition to the above tables to be printed out, a plotter procedure is included , which plots out the amplitudes, force transmitted, phase angles of the amplitudes, and transmitted forces at different rotor speeds.

The computer program ROTOR4P, which plots up the different variables with speed, must be provided with the following input cards in addition to the eight input data cards of ROTOR4:

Card 9

1. WP - Number of cards to be read to plot

Card 10

1. A - Case number
2. GK - Always to be set equal to 1

Card 11

1. B - Grid type (described in detail in table I)
2. }
3. } C, D, E, F
4. }
5. }
6. YMIN - Minimum value of variable along y-axis
7. DY - Magnitude of variable along y-axis per inch of the total 6 inches along y-axis
8. QQ - If '0' (zero), then program scales according to first line drawn on graph; on the other hand, if this is '1', then the values of YMIN and DY must be provided

Card number 11 must be punched with proper input values and, as is obvious, can be more than one, depending on the number WP of card number 9.

A listing of the above computer program is given in appendix B along with sample output tables.

TABLE I. - EXPLANATION OF VARIABLES B, C, D, E, F

IN INPUT CARD 11 AND EXPLANATION OF SYMBOLS

If B = (a)	1	2	3	4	5	6	7	Meaning of symbols
C	x_1	ψ_{x1}	α_1	$\psi_{\alpha 1}$	F_{x1}	PUBFX1	Arbitrary amplitudes x and y	Δ
D	y_1	ψ_{y1}	α_2	$\psi_{\alpha 2}$	F_{y1}	PUBFY1	Phase angle of amplitude at arbitrary location	$\otimes$
E	x_2	ψ_{x2}	--	----	F_{x2}	PUBFX2	---------------	+
F	y_2	ψ_{y2}	--	----	F_{y2}	PUBFY2	---------------	*

[a]C, D, E, F should be either '1' or '0' of 1; the corresponding variables are to be plotted for particular value of B. On the other hand, if it is zero, no plot is made.

4.03 Computer Program to Evaluate Maximum Rotor Amplitude and Forces for a Four Degree of Freedom System (ROTOR4M)

This computer program evaluates design data for the four degree of freedom system that simulates a rigid body rotor on general anisotropic bearings. Two planes of unbalance are considered in this program and the response caused by this rotor unbalance is evaluated. The program uses an iterative procedure to find the maximum amplitudes and then the corresponding critical speeds, phase angles, etc. are determined. This program calculates the amplitudes at each increment of rotor speed. When a peak amplitude is found by the iterative process, the corresponding speed is recorded, and all other parameters are computed for this critical speed. A procedure is incorporated into the program to obtain the maximum amplitudes and other needed design parameters for arbitrary location along the rotor centerline.

The procedure which evaluates the maximum amplitude by iterative procedure is FINDMAX. A flow chart of this procedure is shown in figure 7. The program requires the following to be read as input data:

Card 1

1. SPEC - Allowable percent error on speed

Card 2

1. WO - Initial speed, rps
2. DW - Increment in speed, rps
3. WM - Final speed, rps

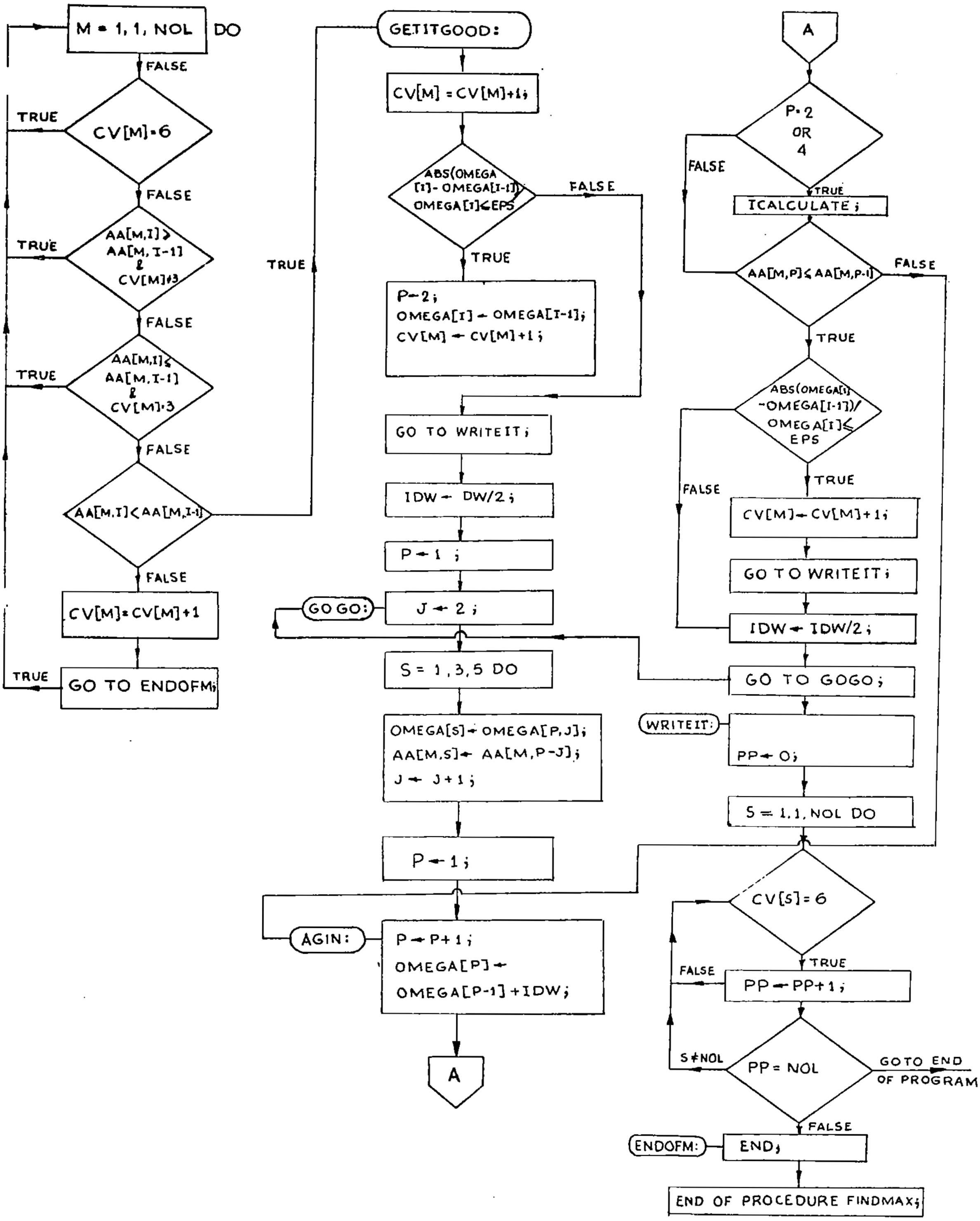


Figure 7. - Flow chart of procedure Findmax used in ROTOR4M.

Card 3

1. L - Length between bearings, in.
2. L1 - Distance from first bearing to mass center, in.
3. L2 - Distance from second bearing to mass center, in.
4. W - Rotor weight, lb.
5. IP - Polar moment of inertia, lb-in. -sec^2
6. IT - Transverse moment of inertia of rotor about mass center, lb-in. sec^2

Card 4

1. WM1 - First unbalance weight, lb
2. WM2 - Second unbalance weight, lb
3. H1 - Distance from first bearing to first unbalance, in.
4. H2 - Distance from first bearing to second unbalance, in.
5. PHI - Phase angles between unbalance planes, deg
6. R1 - Radius of first unbalance location, in.
7. R2 - Radius of second unbalance location, in.

Card 5

1. P - Number of places other than the bearing locations where displacements are to be measured
2. LZ1 - Distance from first bearing to first probe, in.
3. LZ2 - Distance from first bearing to second probe, in.

Card 6

1. K1X - First bearing stiffness in x-direction, lb/in.
2. K2X - Second bearing stiffness in x-direction, lb/in.
3. K1Y - First bearing stiffness in y-direction, lb/in.
4. K2Y - Second bearing stiffness in y-direction, lb/in.

Card 7

1. C1X - First bearing damping coefficient in x-direction, lb-sec/in.
2. C2X - Second bearing damping coefficient in x-direction, lb-sec/in.
3. C1Y - First bearing damping coefficient in y-direction, lb-sec/in.
4. C2Y - Second bearing damping coefficient in y-direction, lb-sec/in.

Card 8

1. D1X - Cross coupling damping coefficient, lb-sec/in.
2. D2X - Cross coupling damping coefficient, lb-sec/in.
3. D1Y - Cross coupling damping coefficient, lb-sec/in.
4. D2Y - Cross coupling damping coefficient, lb-sec/in.

Card 9

1. R1X - Cross coupling stiffness, lb/in.
2. R2X - Cross coupling stiffness, lb/in.
3. R1Y - Cross coupling stiffness, lb/in.
4. R2Y - Cross coupling stiffness, lb/in.

Card 10

1. CONTROL - Identifier controlling the symmetry of bearings. If CONTROL = 0, we are dealing with symmetric case

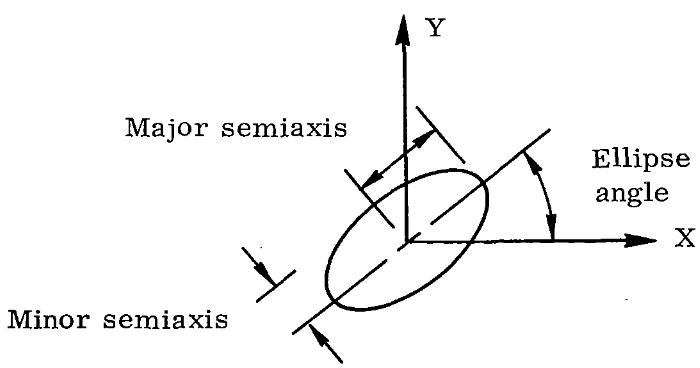
The data cards are in free field format. A comma should separate all data entries. A comma is needed after the last data entry.

The output data is as follows:

Column

1	Speed, rps
2	Coordinate (i.e., bearing 1 or 2 or any arbitrary location)
3	Amplitude, in.
4	Phase angle of the amplitude WRT unbalance
5	Major semi axis/amplitude of coordinate, DIM
6	Minor semi axis/amplitude of coordinate, DIM
7	Ellipse angle of major semi axis with x-axis
8	Bearing location of maximum force transmitted
9	Maximum force transmitted
10	Phase angle of maximum force WRT unbalance force
11	Percent cylindrical mode

Columns 5, 6, and 7 give quantities needed to plot the elliptical orbit motion at the specified location (i.e., bearing 1 or 2 or any arbitrary location). The following is an example of an orbit motion plot:



Column 11 indicates the percentage of the motion that is of a cylindrical mode type, as opposed to the conical mode.

A listing of the above computer program is given in appendix C along with sample output tables.

4.04 Application of Four Degree of Freedom Unbalance Response Computer Programs

As an example of the computer programs ROTOR4 and ROTOR4P, the following rotor is considered. The rotor and bearing characteristics are as follows:

W	=	Rotor weight, 110 lb
WM1	=	WM2 = Rotor unbalance, 0.2 lb
L	=	Distance between bearing centerlines, 30 in.
L1	=	Distance from first bearing to mass center, 15 in.
L2	=	Distance from second bearing to mass center, 15 in.
H1	=	Distance from first bearing to first unbalance, 0
H2	=	Distance from first bearing to second unbalance, 0
R1	=	Radius of first unbalance location, 2 in.
R2	=	Radius of second unbalance location, 2 in.
IP	=	Polar moment of inertia, 0.57 lb-in.-sec^2
IT	=	Transverse moment of inertia, 21.6 lb-in.-sec^2
PHI	=	Phase angle between unbalance planes, 0
K1X	=	20×10^4 lb/in.
K2X	=	1.5×10^4 lb/in.
K1Y	=	1.6×10^4 lb/in.
K2Y	=	1.2×10^4 lb/in.
R1X	=	0 lb/in.
R2X	=	0 lb/in.
R1Y	=	0 lb/in.
R2Y	=	0 lb/in.
C1X	=	7.0 lb-sec/in.
C2X	=	7.0 lb-sec/in.
C1Y	=	7.0 lb-sec/in.
C2Y	=	7.0 lb-sec/in.
D1X	=	0 lb-sec/in.
D2X	=	0 lb-sec/in.
D1Y	=	0 lb-sec/in.
D2Y	=	0 lb-sec/in.

The rotor performance was calculated for a speed range of 40 to 138 rps. Table B-I (appendix B) represents the rotor input characteristics and the rotor x or horizontal response for both bearings. The definition of the various rotor amplitudes, phase angles, and forces transmitted is given in section 4.02. Table B-II represents the rotor response in the y or vertical direction. Table B-III lists the rotor displacements and phase angles at arbitrary positions along the shaft corresponding to the number of places selected on input data card 4.

As an example of ROTOR4P the data presented in tables B-I and B-II (appendix B) were plotted up by use of the plotter routine. These plotting procedures are designed to automatically scale the figures in 6- by 8-inch graphs. There are eight basic graphs that may be obtained depending on the card input data. All or only several of these curves may be plotted as desired.

The first plot, figure 8, represents the horizontal and vertical motion at both bearings. Since the bearing stiffnesses in the s- and y-directions are all slightly different, we obtain 8 distinct peaks or critical speeds. For each generalized coordinate or displacement, we obtain two distinct peaks. The magnitudes of these peaks are directly influenced by the location and magnitude of the rotor unbalance and damping. For example, the first peak or critical speed in this particular example corresponds to a "cylindrical" resonance and the second corresponds to a "conical" resonance. The relative phase angle Φ and plane of the unbalances will determine the magnitude of excitation of each mode. For example, if the unbalances were situated at the rotor mass center of a symmetric rotor, the conical mode would not be excited.

Figure 9 represents the phase angles between the radial unbalance force and the displacement vector. Note that in the single degree of freedom model, the phase angle become 90° at the critical speed and goes to 180° above the critical. In the four degree of freedom rotor, the phase angle may vary from 0 to 360°. Figure 9 shows that at low speeds the two bearings are in phase and at higher speeds the two bearings are 180° out of phase. This also shows that the critical speeds need not occur with the response lagging the excitation by an angle of 90° and increasing thereafter continuously up to 180° with increase in speed. The phase lag may decrease after the first critical is reached, as can be noted in the response at bearing number 1, and then continuously increase with increase of speed.

Figure 10 shows the plot of angular amplitudes α_1 and α_2 with speed. On examining figure 8 along with figure 10, it will be noticed that the higher criticals observed in the response plot of figure 8 are, in fact, the conical criticals. This conclusion can be deduced from the phase angle-speed plot also. As shown in figure 9, the relative phase difference between the rotor response at two bearings in 180° at higher speeds, hence, the occurrence of conical criticals at these speeds.

Figure 11 shows the phase lag of conical responses with moment excitation in x-z and y-z planes. The angular responses lag the momental excitation by about 180° at low speeds, and this phase lag continuously increases to 300° with increase in speed.

Figure 12 shows the force transmitted to the bearings in horizontal and vertical directions at different speeds with the specified bearing characteristics of the system.

The occurrence of the maximum force transmitted is at the critical speeds, as can be observed when compared with figure 8.

Figure 13 shows the phase lag of the transmitted forces with respect to the excitation force with increase of speed. It can be observed that at low speed the forces transmitted in the two bearings are in phase, but as the speed increases the relative phase lag between the forces transmitted in two bearings increases. At very high speed they are out of phase with respect to each other. The trend of these phase angles, in the four degree of freedom system will be compared with a single degree of freedom system later. This leads to a very interesting result.

Figures 14 and 16 are plots of the amplitudes of the rotor at ±15 inches from the first bearing. Figures 15 and 17 show the corresponding phase angles.

Figure 18 shows the plot of amplitude ratio against frequency ratio for a single degree of freedom system. Figure 19 is a plot of the corresponding phase angle with respect to excitation. It is well known that for a single degree of freedom system the response lags the excitation by 90° at critical speeds, which increases rapidly at low damping coefficient and the response is out of phase with respect to excitation at high speeds. However, the same conclusions cannot be reached in case of rotor-bearing systems where there is more than one degree of freedom. Figure 9, which is a plot of the phase angles of response at bearings number 1 and 2, shows that in the first bearing the phase angle gradually increases as the speed increases. At the critical speeds, the phase shift is not necessarily 90°, but may be more or less than 90°, nor does this continually increase and reach a value of 180° at very high speed, as is observed in a single degree of freedom system. In the particular case considered, the phase angle at bearing number 1 reaches a maximum value, then decreases with increase of speed, and finally approaches a constant value of 180° at high speed, whereas the phase angle at bearing number 2 continually increases with increase of speed, and at high speed is lagging the excitation by 360°. This is quite an interesting and unexpected result and was not observed in the single degree system mathematical model. In reference 3 a plot of the phase angles of a six degree of freedom rotor-bearing system shows that the response may lag the excitation by 540°. Figure 9 may further be utilized to obtain the relative phase angle between the responses at two bearings. One important conclusion that can be drawn from this is that at low speed the responses at the two bearings are in phase with each other, but they are out of phase at high speed.

Figure 20 shows the plot of transmissibility against frequency ratio for a single degree of freedom system. It can be observed that below a frequency ratio of 1.41 the transmissibility increases with decrease of damping ratio, whereas above 1.42 the transmissibility increases with increase of damping ratio. Hence, if the operating frequency ratio is below 1.41, it is advisable to have a higher damping, and for a frequency ratio greater than 1.41, the damping ratio should be low in order that the transmissibility remain at a low value. A damping ratio value of 4.00 keeps the transmissibility almost constant in the entire frequency ratio range.

Figure 21 is a plot of force transmitted/impressed force against frequency ratio for a single degree of freedom system. Here also, as in the case of transmissibility, the force transmitted increases with decrease of damping for frequency ratio below 1.41, and above this the force transmitted increases with increase of damping ratio. This plot provides a method for choosing a value of damping such that the force transmitted remains at an optimum value in the entire frequency range. Figures 22 and 23 show the effect of damping on the system. Compared with figures 8 and 12, these show that the amplitude and the force transmitted to the bearings are considerably reduced. Figures 24 and 25 show that with a damping of 30 pound-sec per inch, the amplitude and the force transmitted are reduced further and the response increases with an increase in angular velocity. The interesting feature of the addition of this extra amount of damping is that the resonance of the system does not occur any more at the two angular velocities observed previously.

Figures 26 through 30 show plots of the output obtained from the ROTOR4M computer program. The plots shown are for symmetrical bearings; i.e., the assumed stiffness in the x- and y-directions for both the bearings are identical.

Figure 26 shows the cylindrical and conical critical speeds of the NASA gas bearing rotor for various values of stiffness. Since the bearing characteristics are symmetric, one cylindrical and one conical critical are obtained at a given stiffness. This shows that the system is susceptible to instability at the lower critical due to the conical mode and at the higher critical due to the cylindrical mode.

Figure 27 shows the plot of rotor amplification factor 'A' against bearing stiffness for different values of damping. For a particular damping value, the amplitifcation factor increases with increasing stiffness, and for a constant value of stiffness the amplification factor decreases with increasing damping.

Figure 28 is a cross plot of the amplification factor against damping coefficient for various values of stiffness. The same conclusions as observed in figure 24 apply in this case.

Figure 29 shows the plot of rotor phase angle at the cylindrical critical speeds for various values of bearing stiffness. For a particular value of damping coefficient the rotor phase angle decreases with increase of stiffness, and there is a decrease of phase angle with decrease of damping for a particular value of stiffness. This is shown in figure 30.

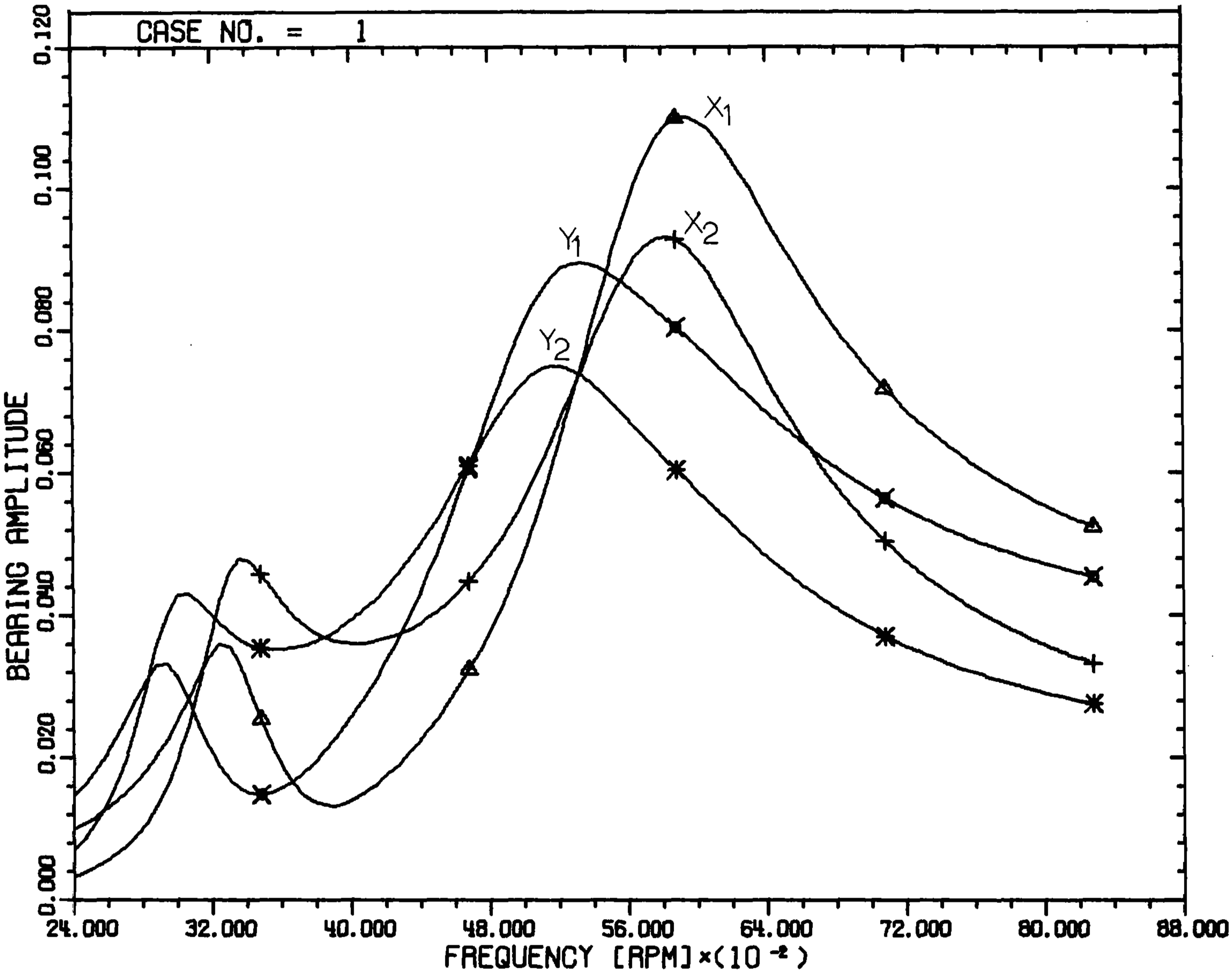


Figure 8. - Bearing horizontal and vertical amplitude against frequency.

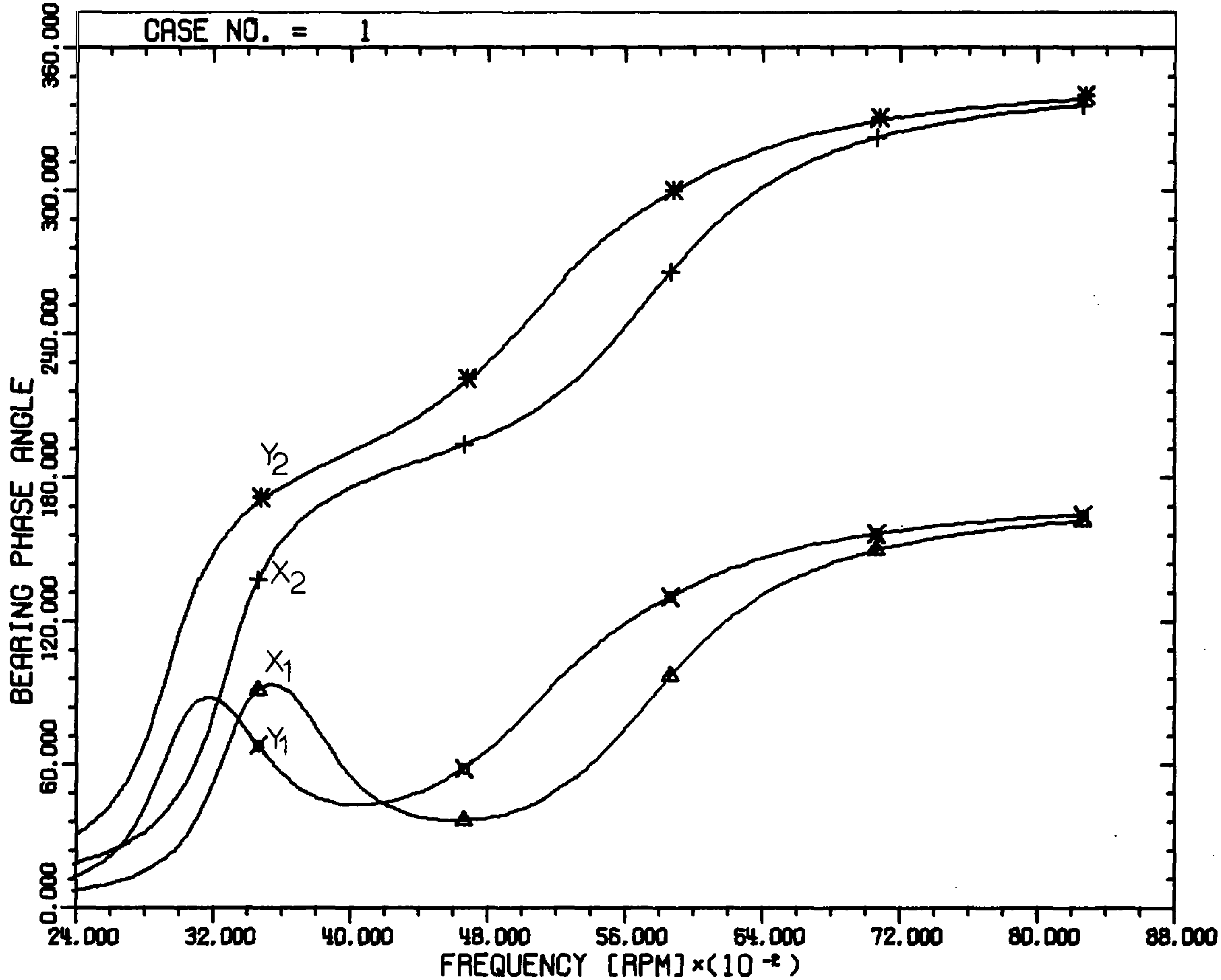


Figure 9. - Bearing horizontal and vertical phase angle against frequency.

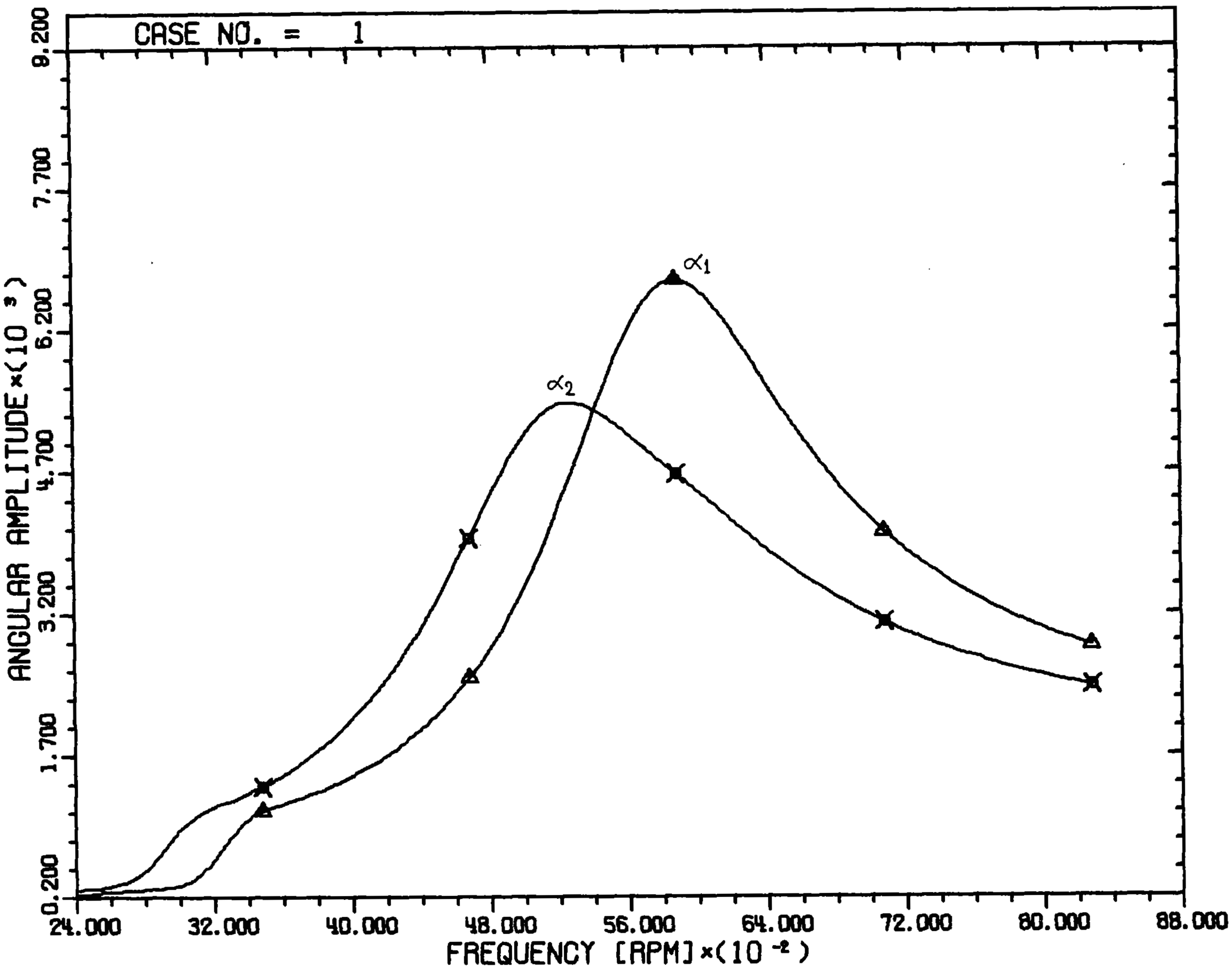


Figure 10. - Angular amplitude against frequency.

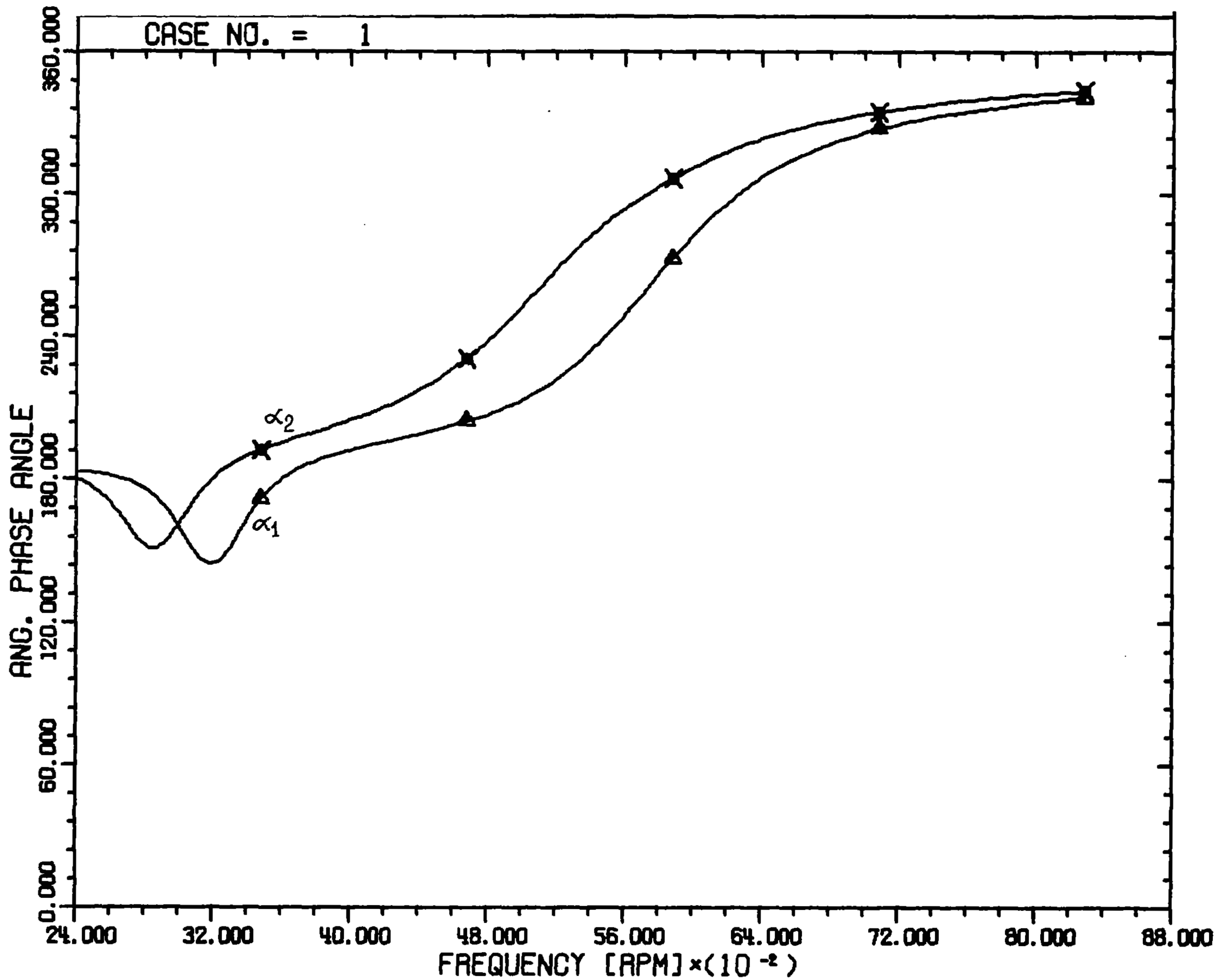


Figure 11. - Angular amplitude phase angle against frequency.

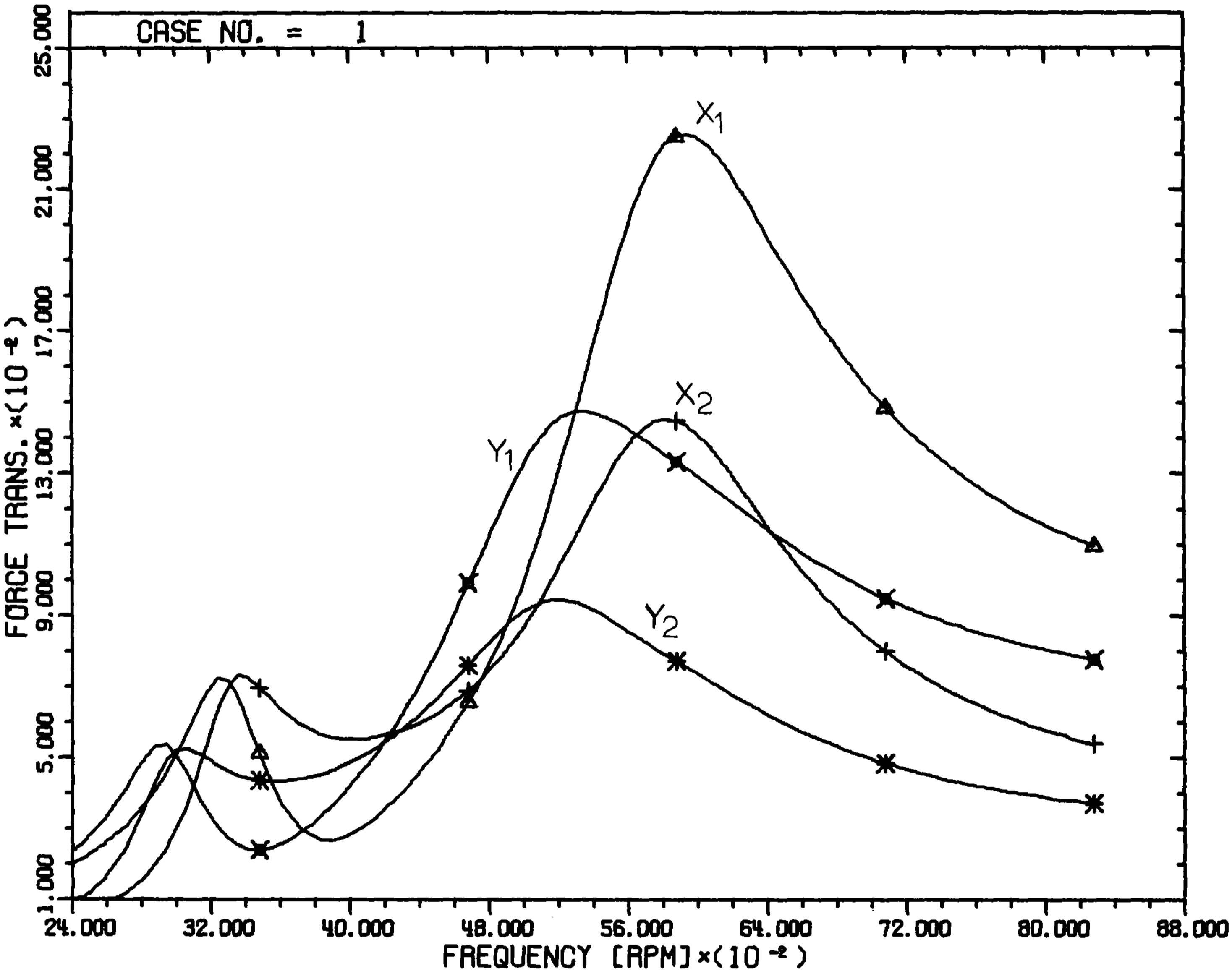


Figure 12. - Force transmitted against frequency.

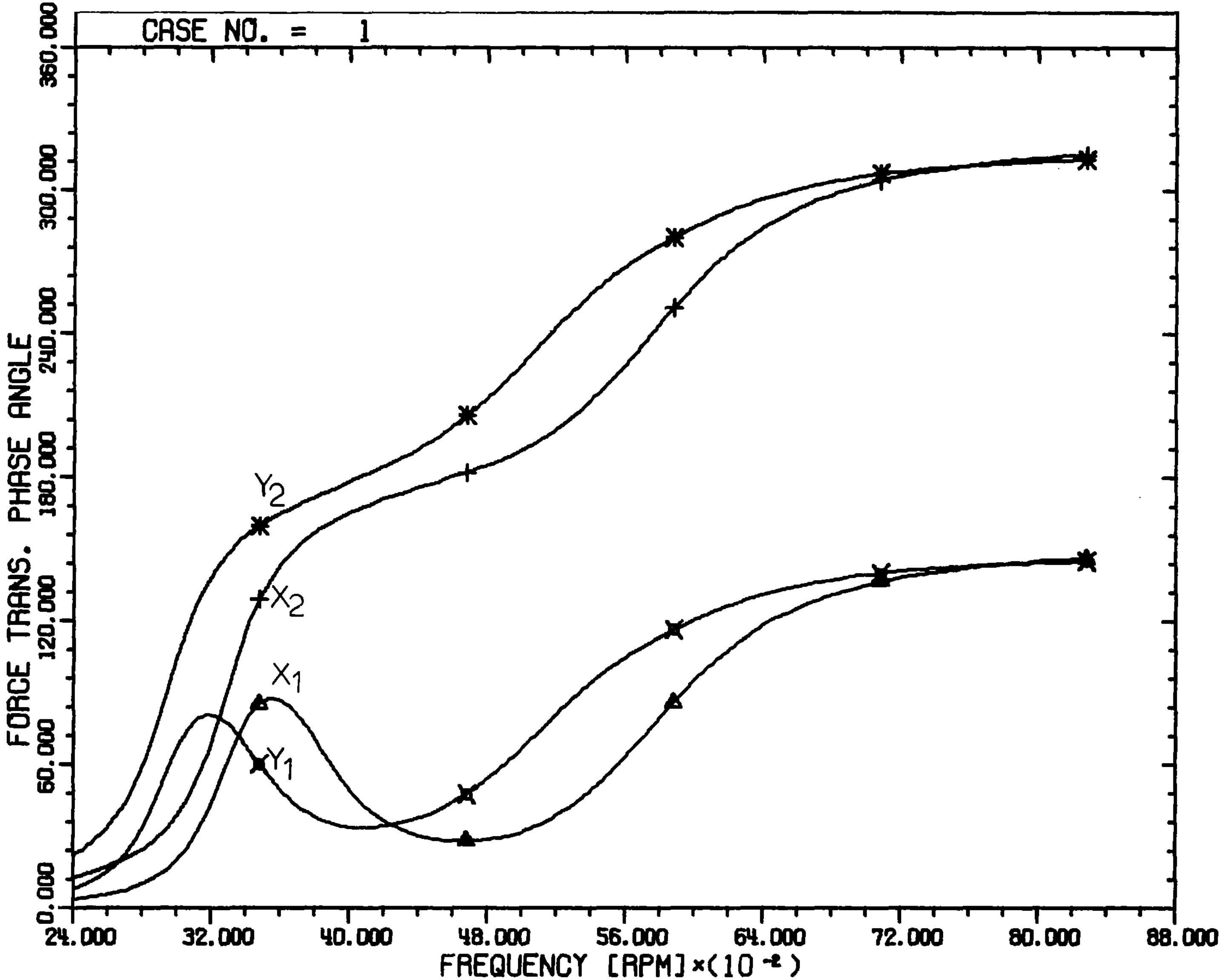


Figure 13. - Force transmitted phase angle against frequency.

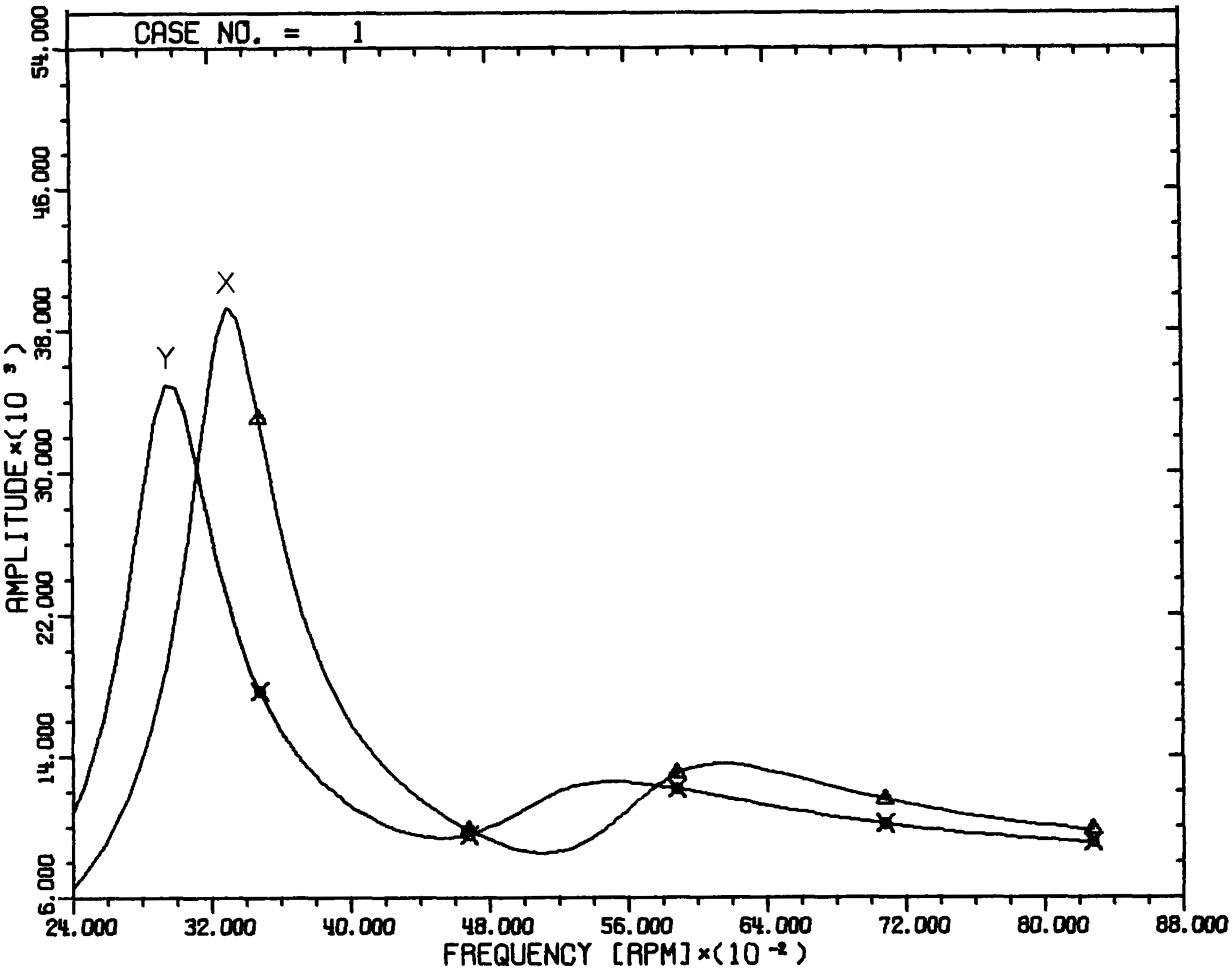


Figure 14. - Amplitude at +15-inch location from first bearing against frequency.

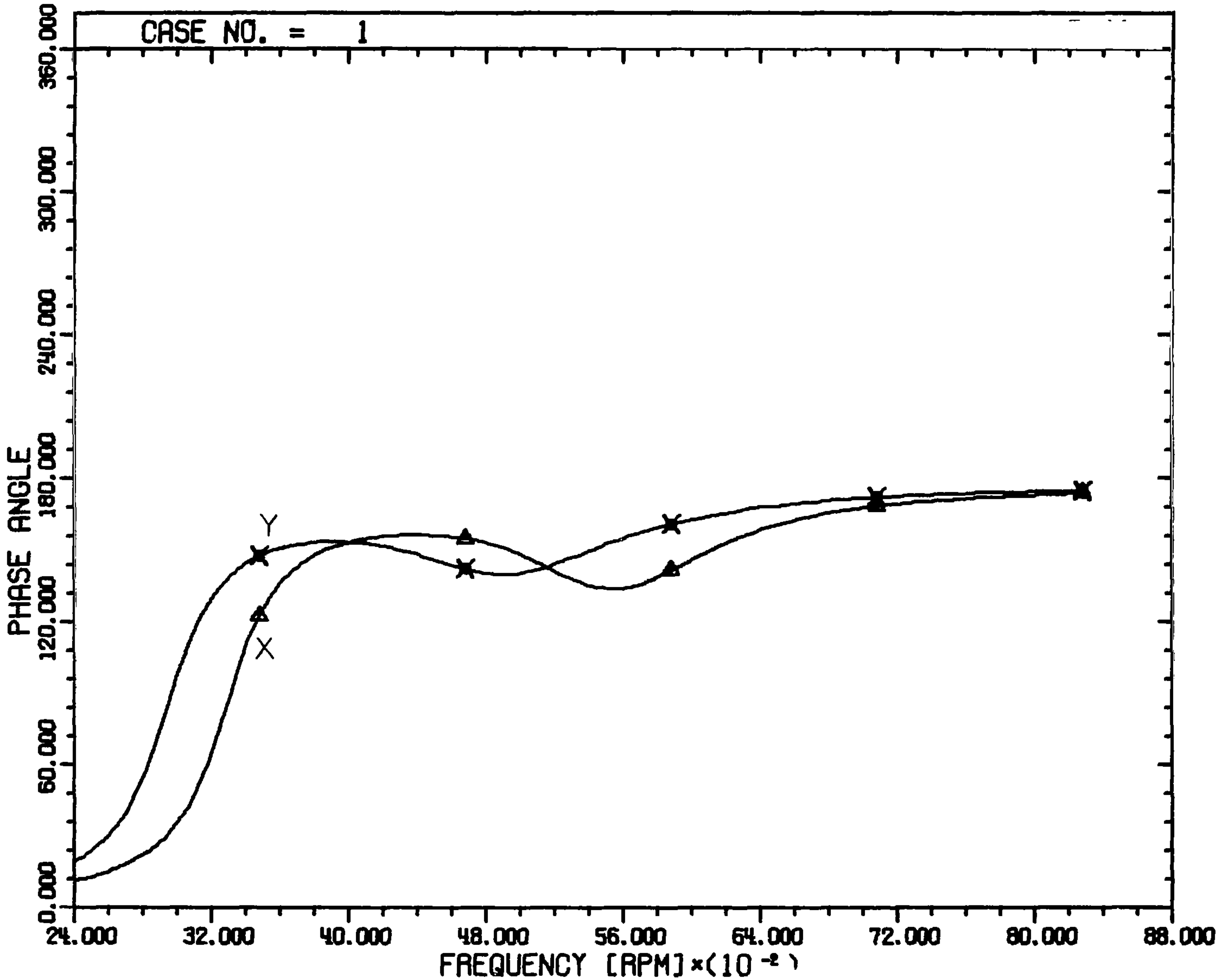


Figure 15. - Phase angle of amplitude at +15-inch location from first bearing against frequency.

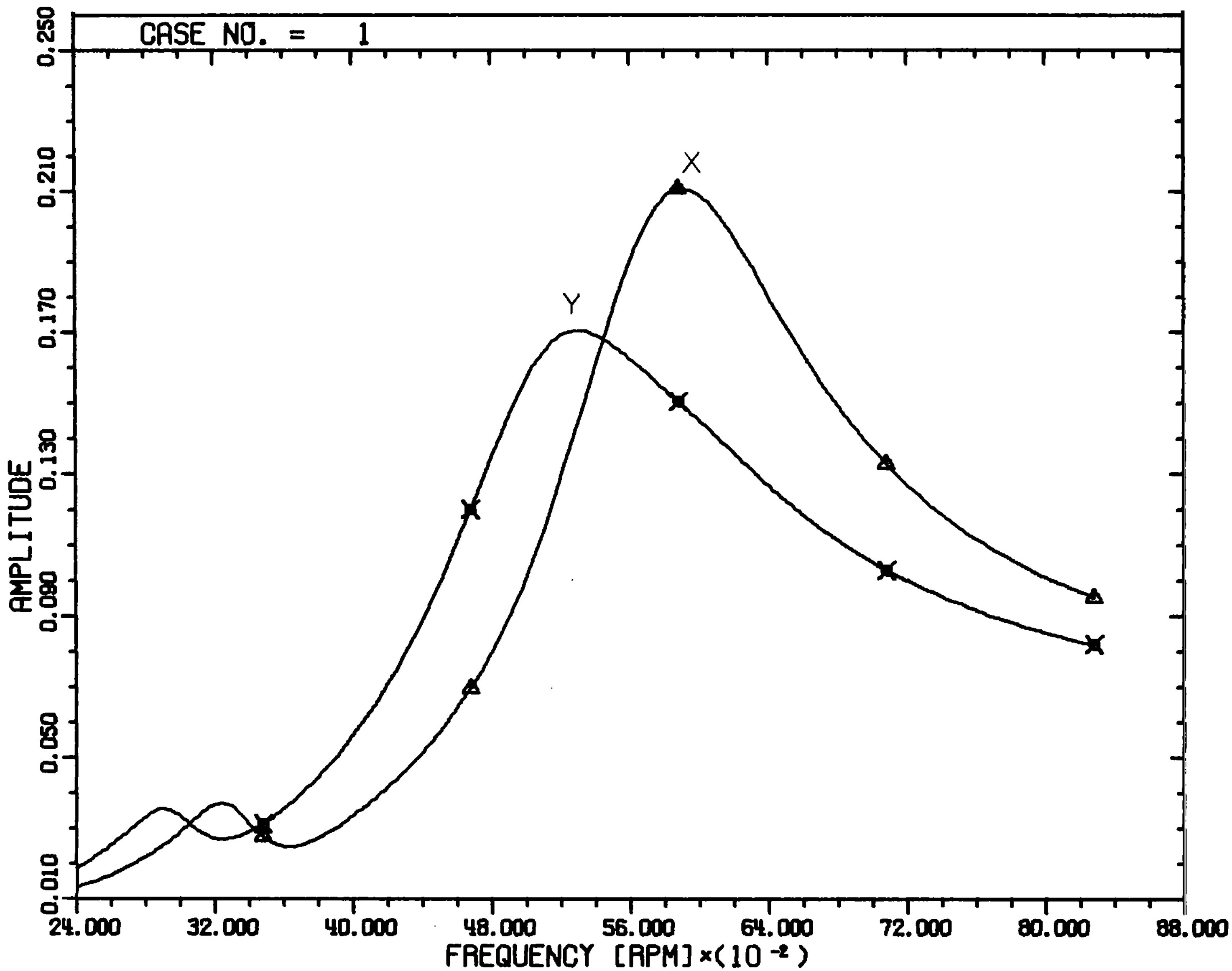


Figure 16. - Amplitude at -15-inch location from first bearing against frequency.

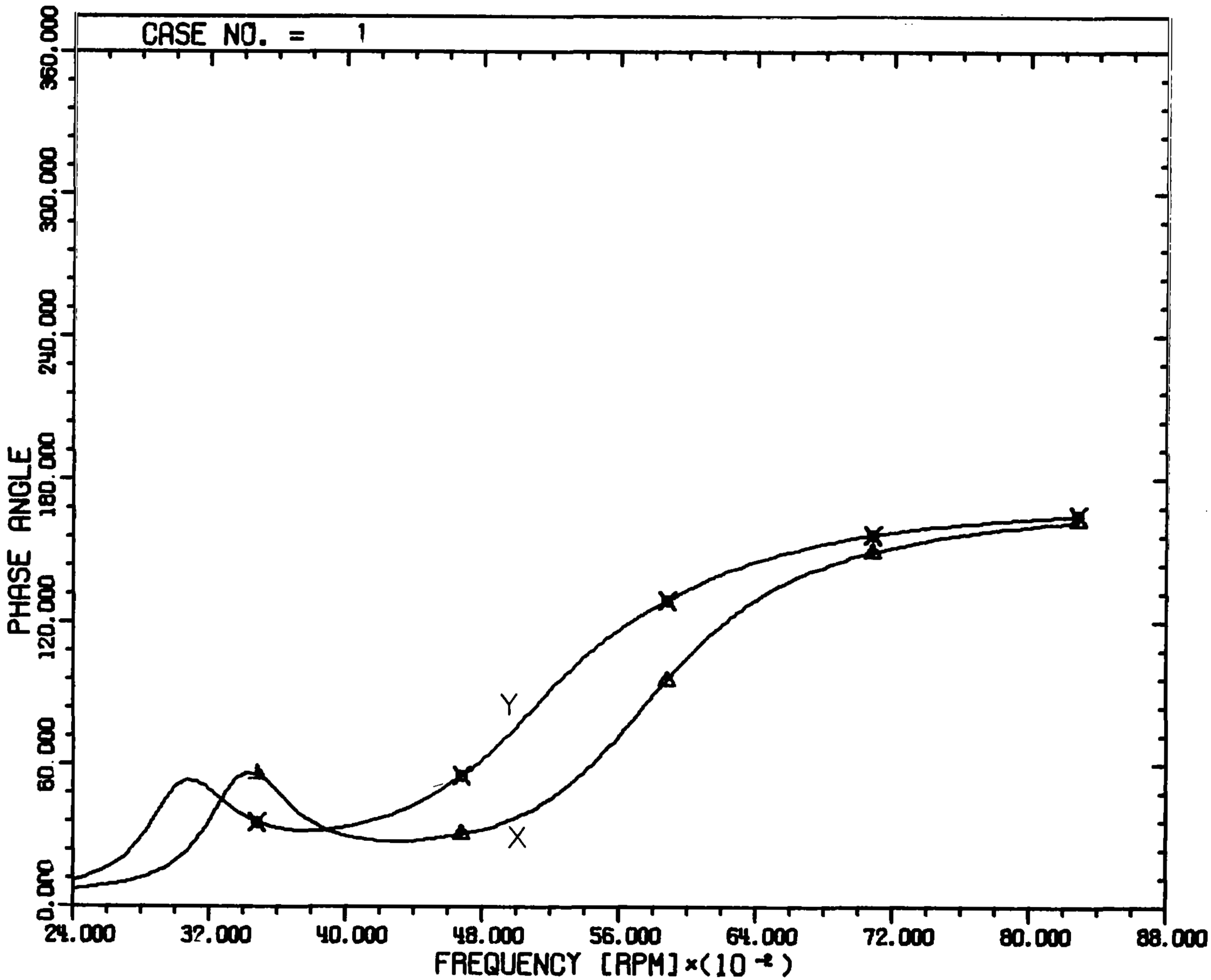


Figure 17. - Phase angle of amplitude at -15-inch location from first bearing against frequency.

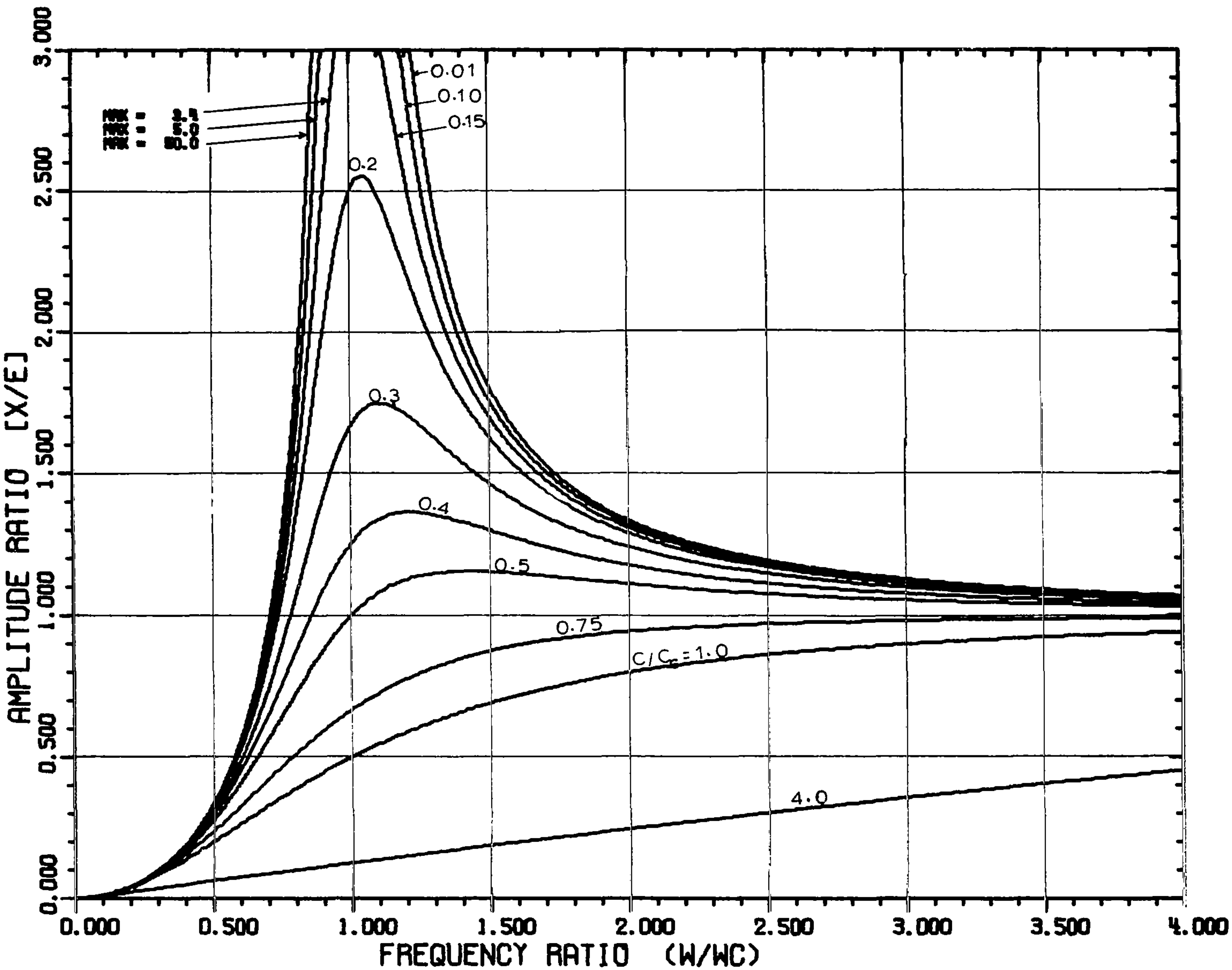


Figure 18. - Steady state response for inertial excitation for a single degree freedom system.

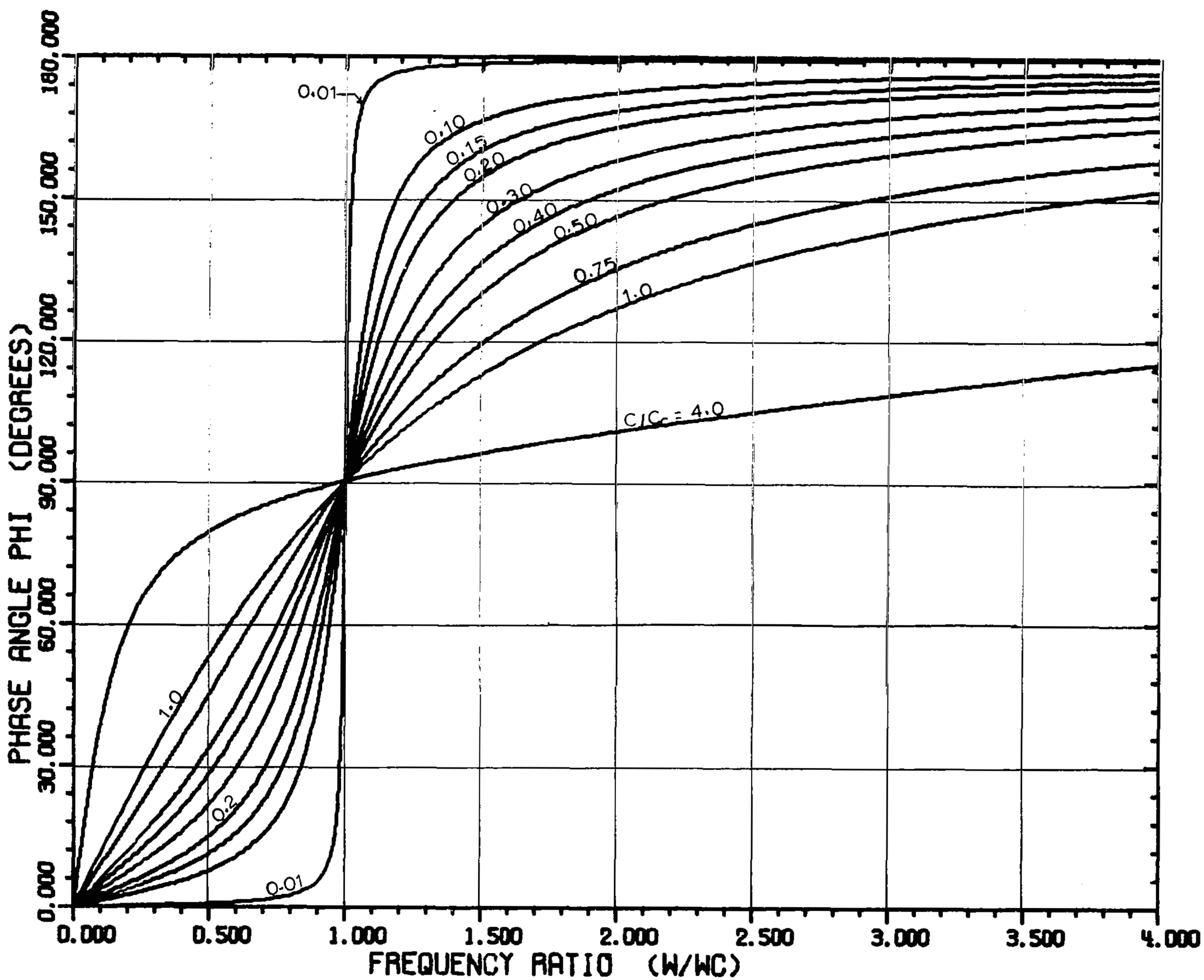


Figure 19. - Variation of phase angle for inertial excitation for a single degree freedom system.

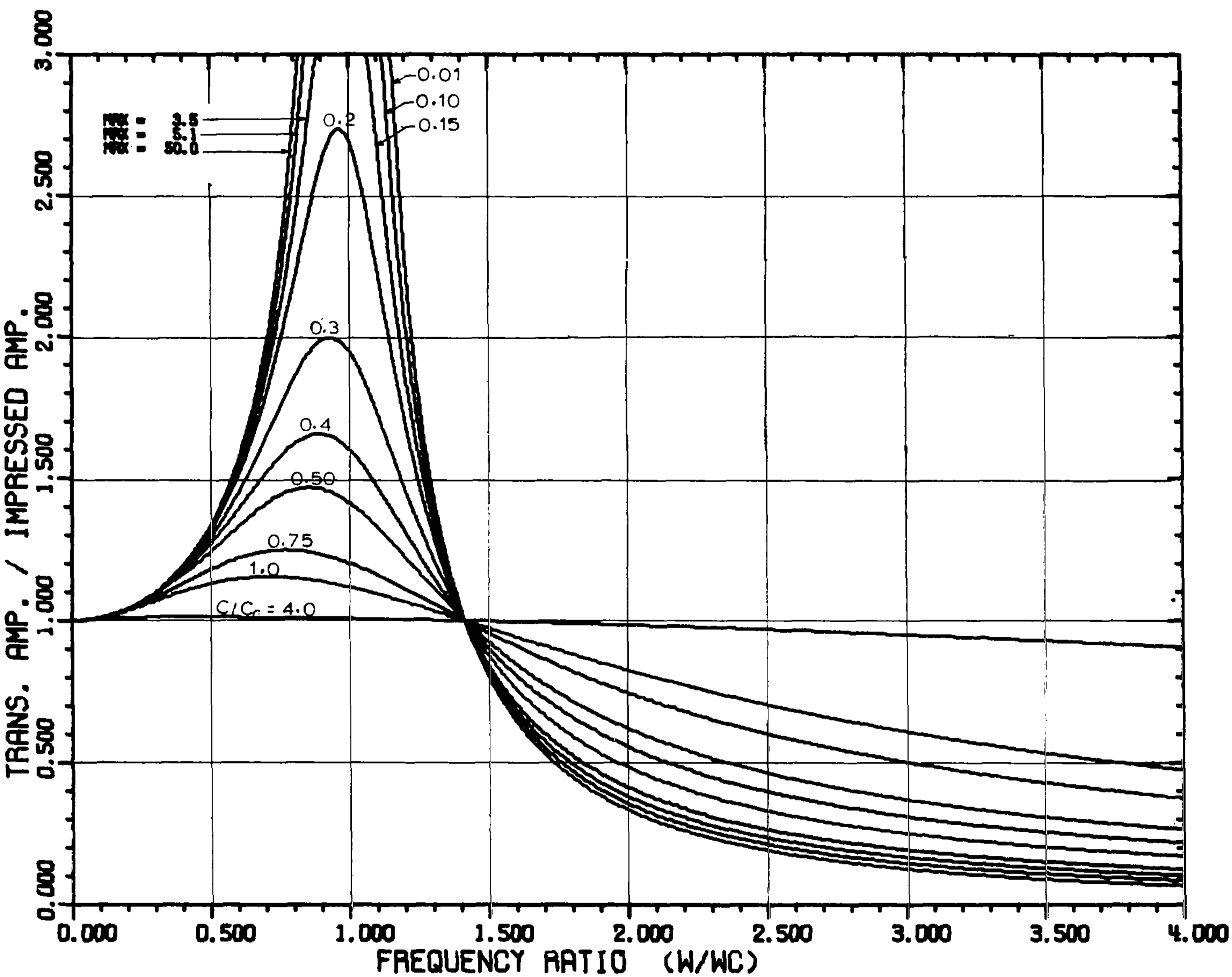


Figure 20. - Transmissibility for a single degree freedom system.

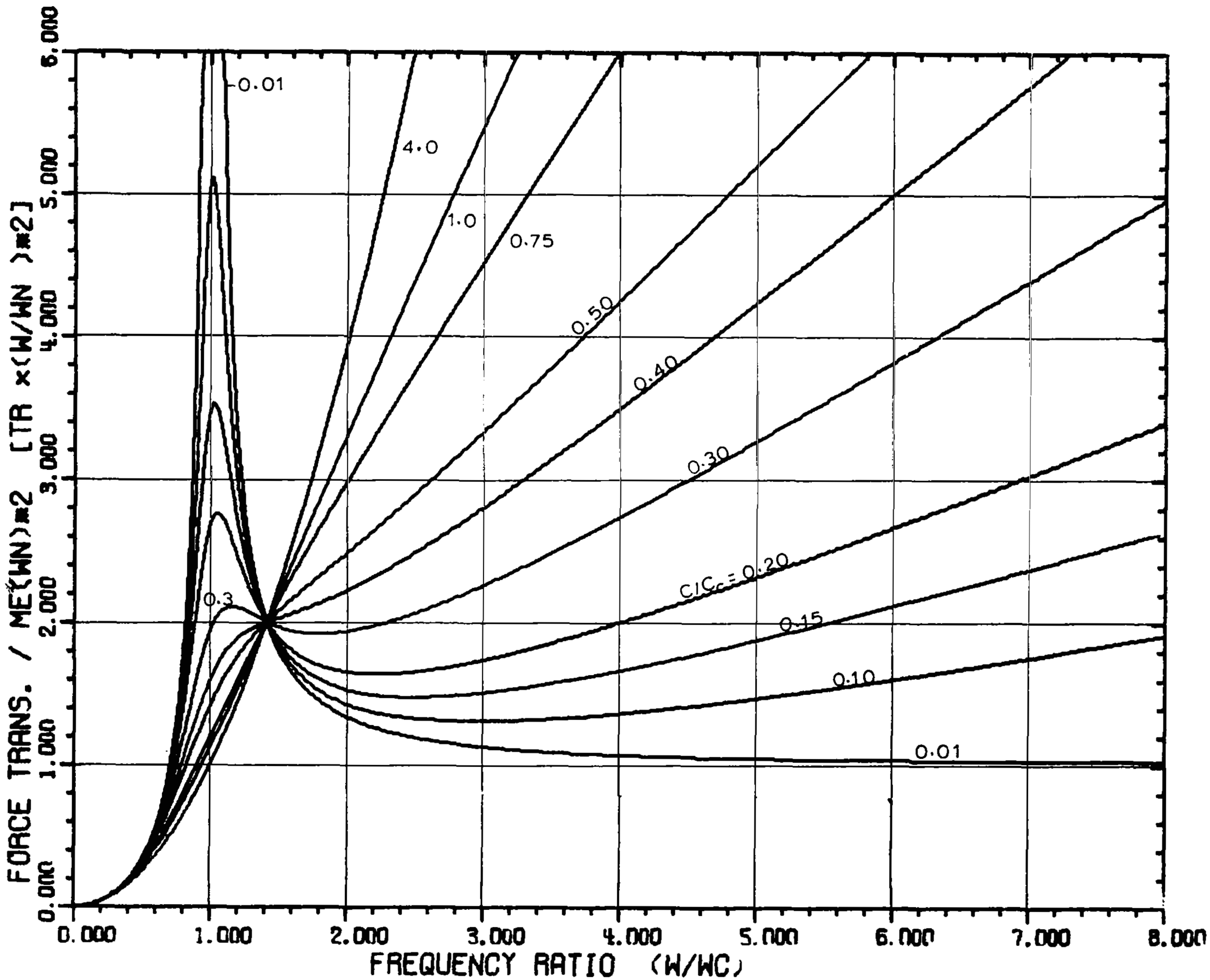


Figure 21. - Force transmitted against frequency ratio for a single degree freedom system.

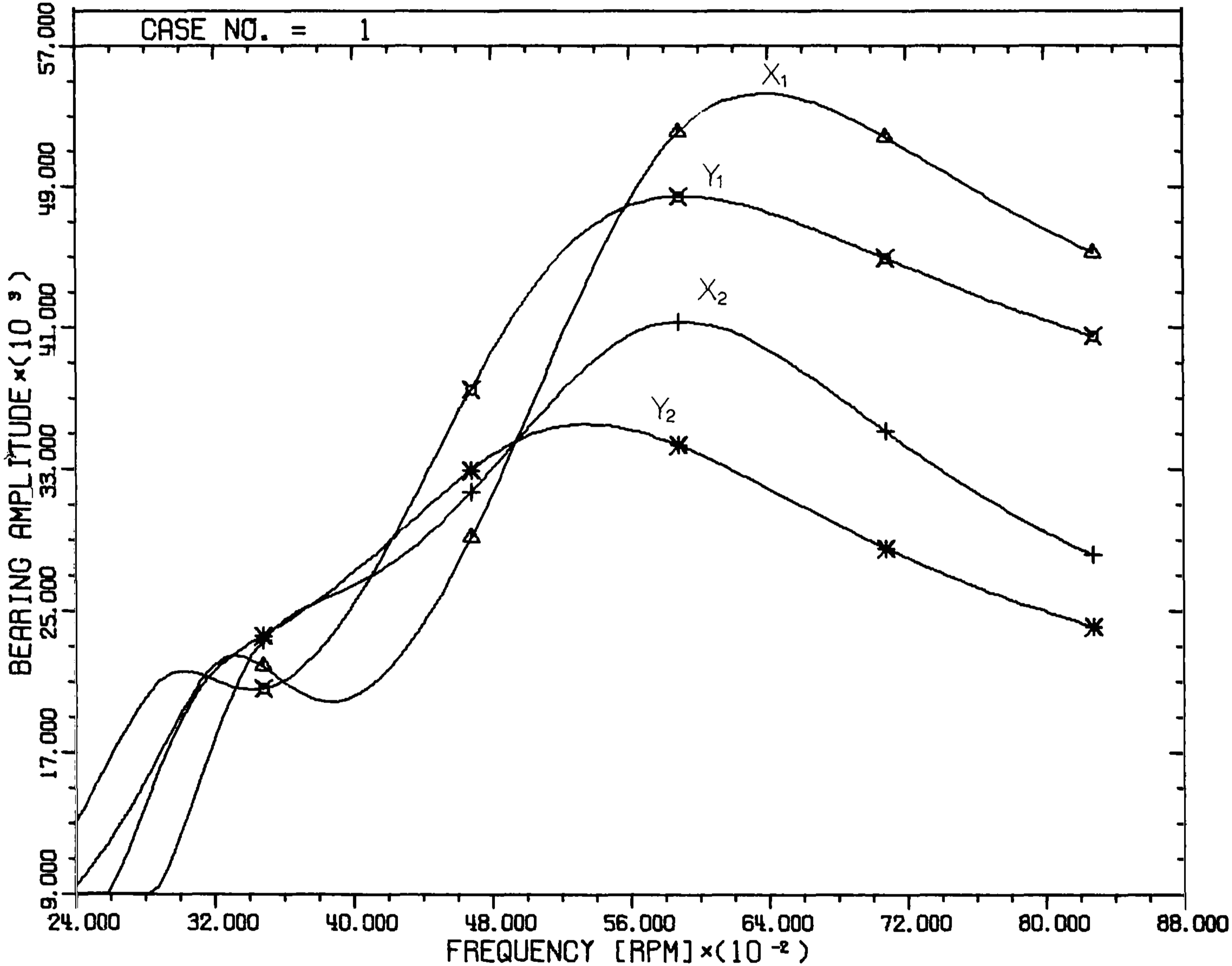


Figure 22. - Bearing amplitude against frequency (four degree freedom system) at damping coefficient 15 pound-seconds per inch.

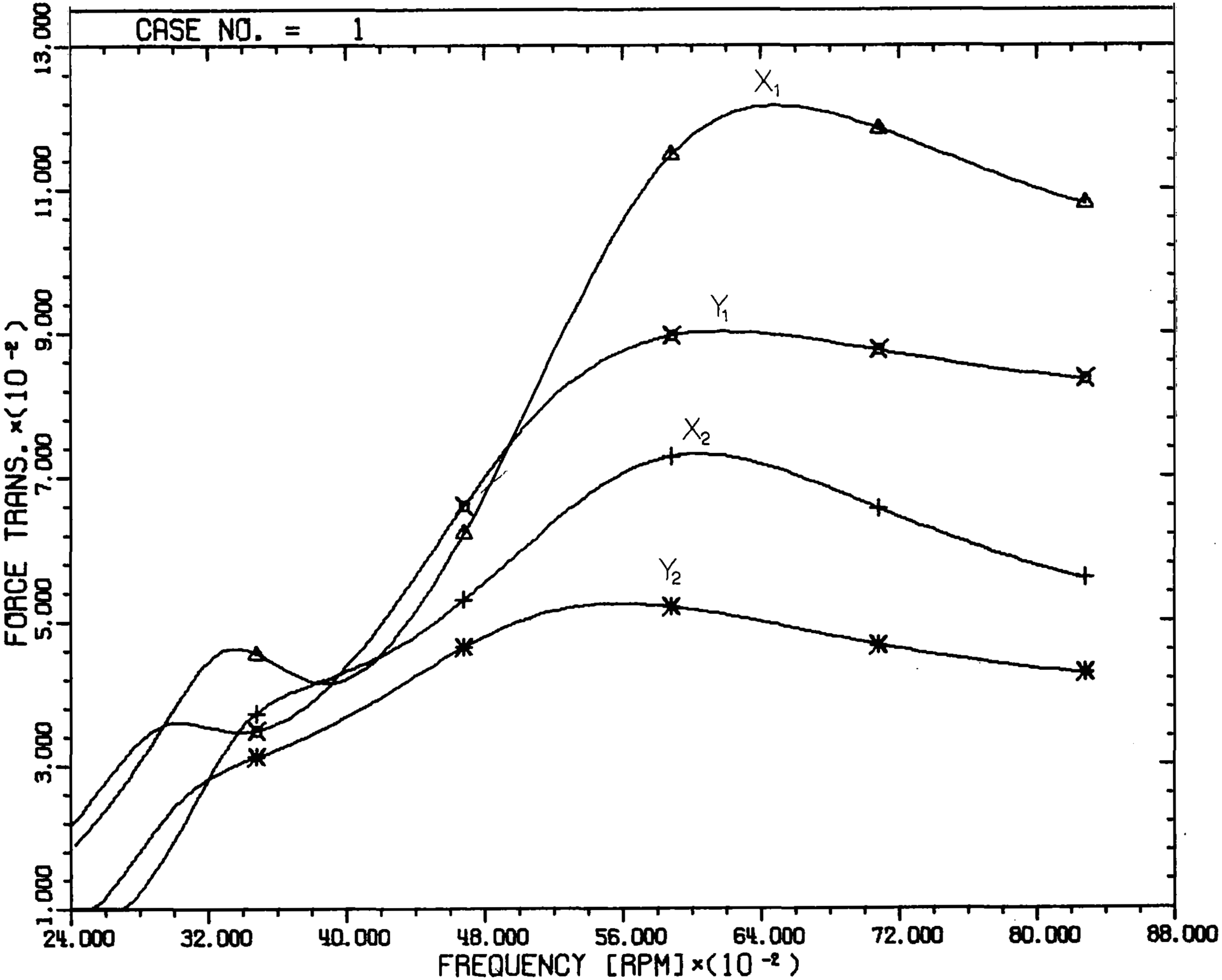


Figure 23. - Force transmitted against frequency (four degree freedom system) at damping coefficient 15 pound-seconds per inch.

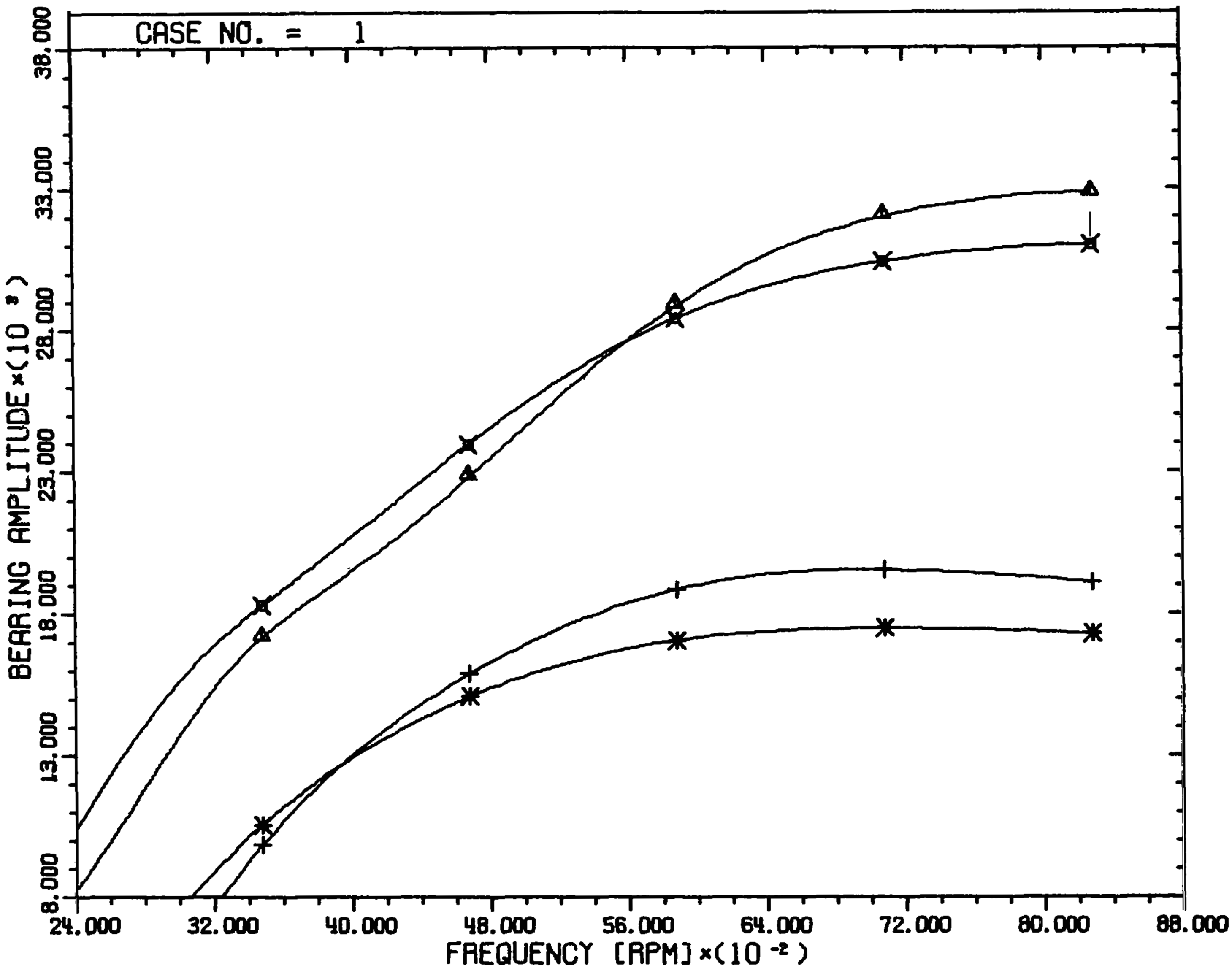


Figure 24. - Bearing amplitude against frequency (four degree freedom system) at damping coefficient 30 pound-seconds per inch.

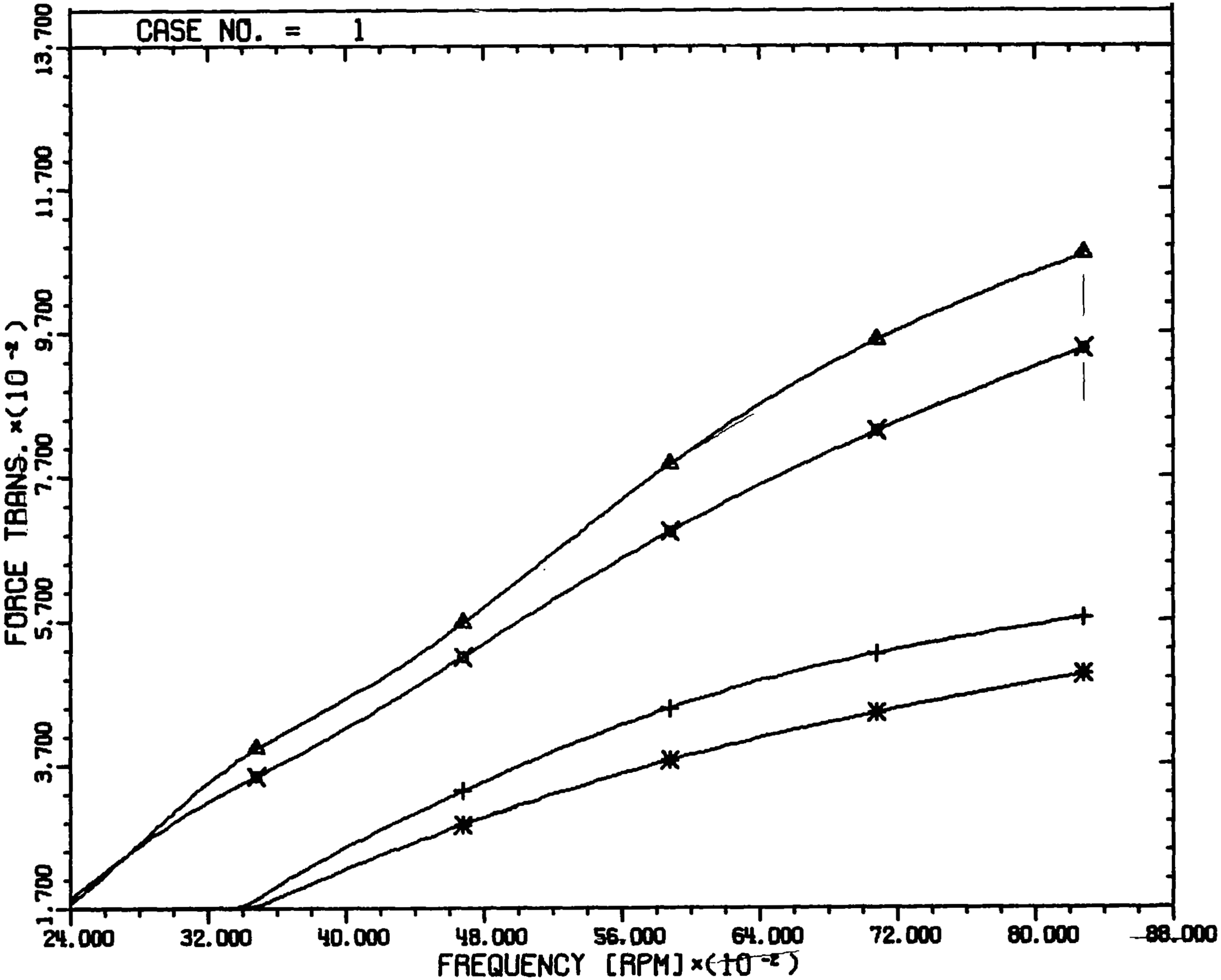


Figure 25. - Force transmitted against frequency (four degree freedom system) at damping coefficient 30 pound-seconds per inch.

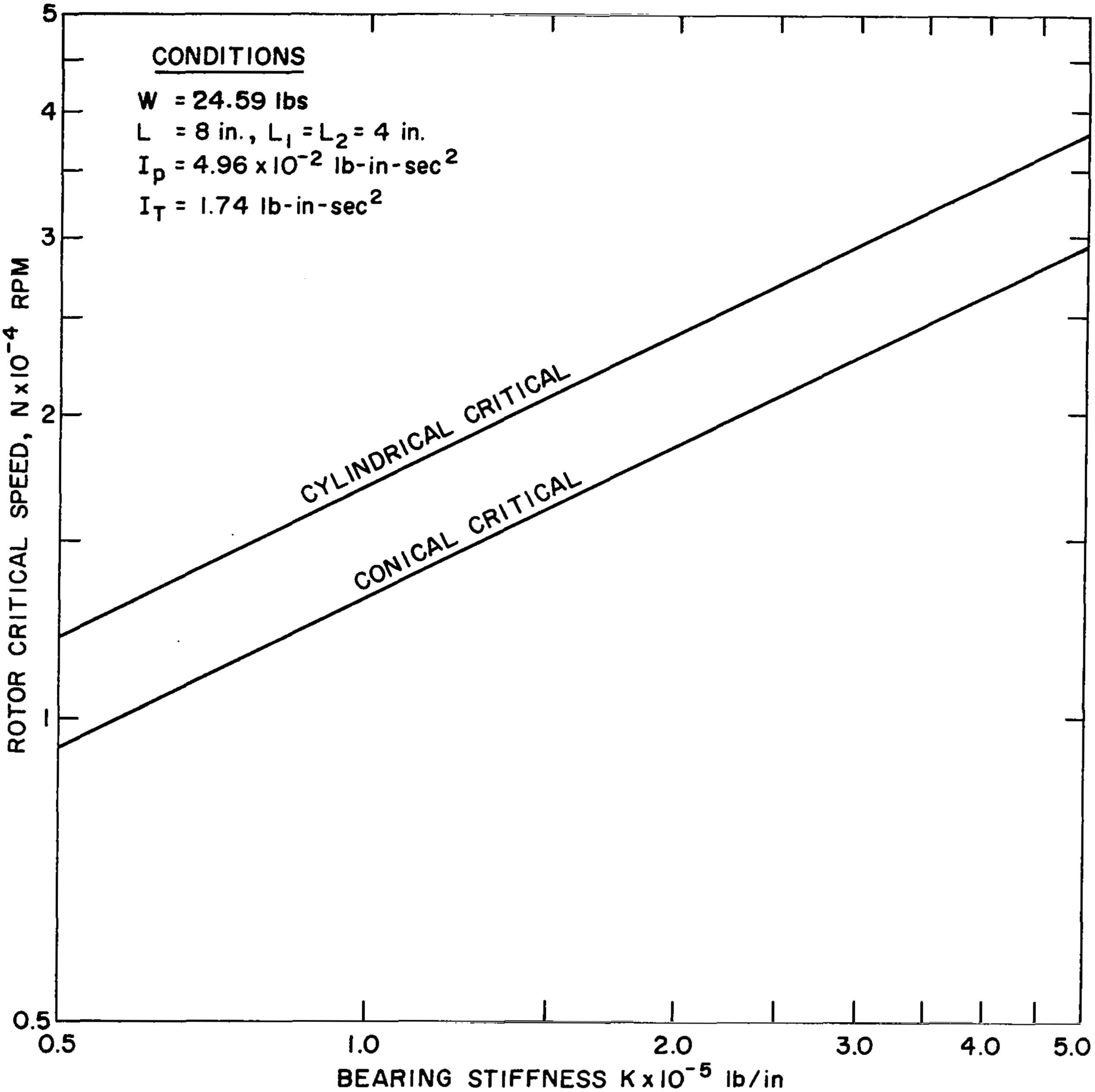


Figure 26. - NASA gas bearing rotor critical speeds for various values of bearing stiffness.

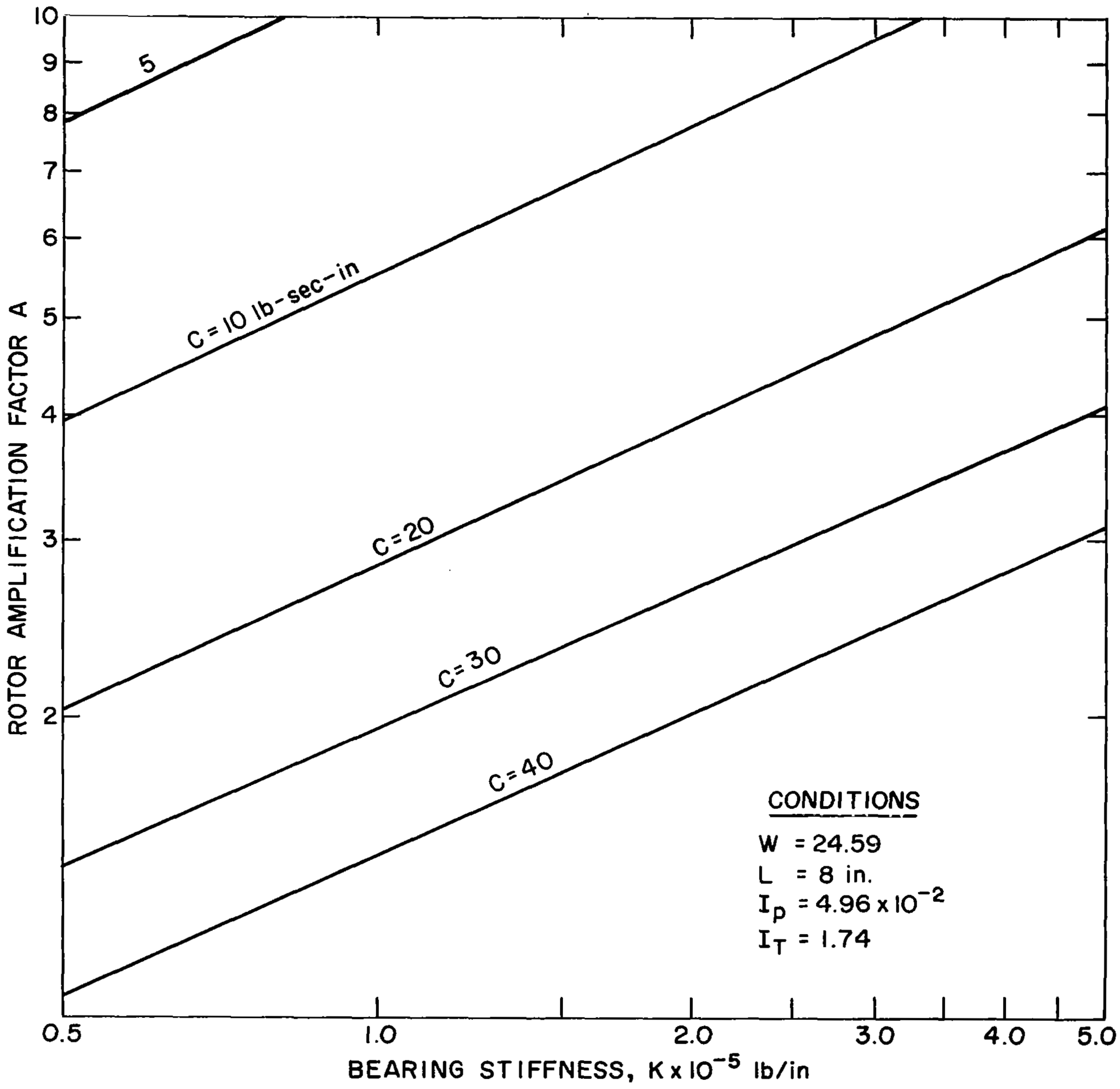


Figure 27. - Cylindrical critical speed rotor amplification factor against bearing stiffness.

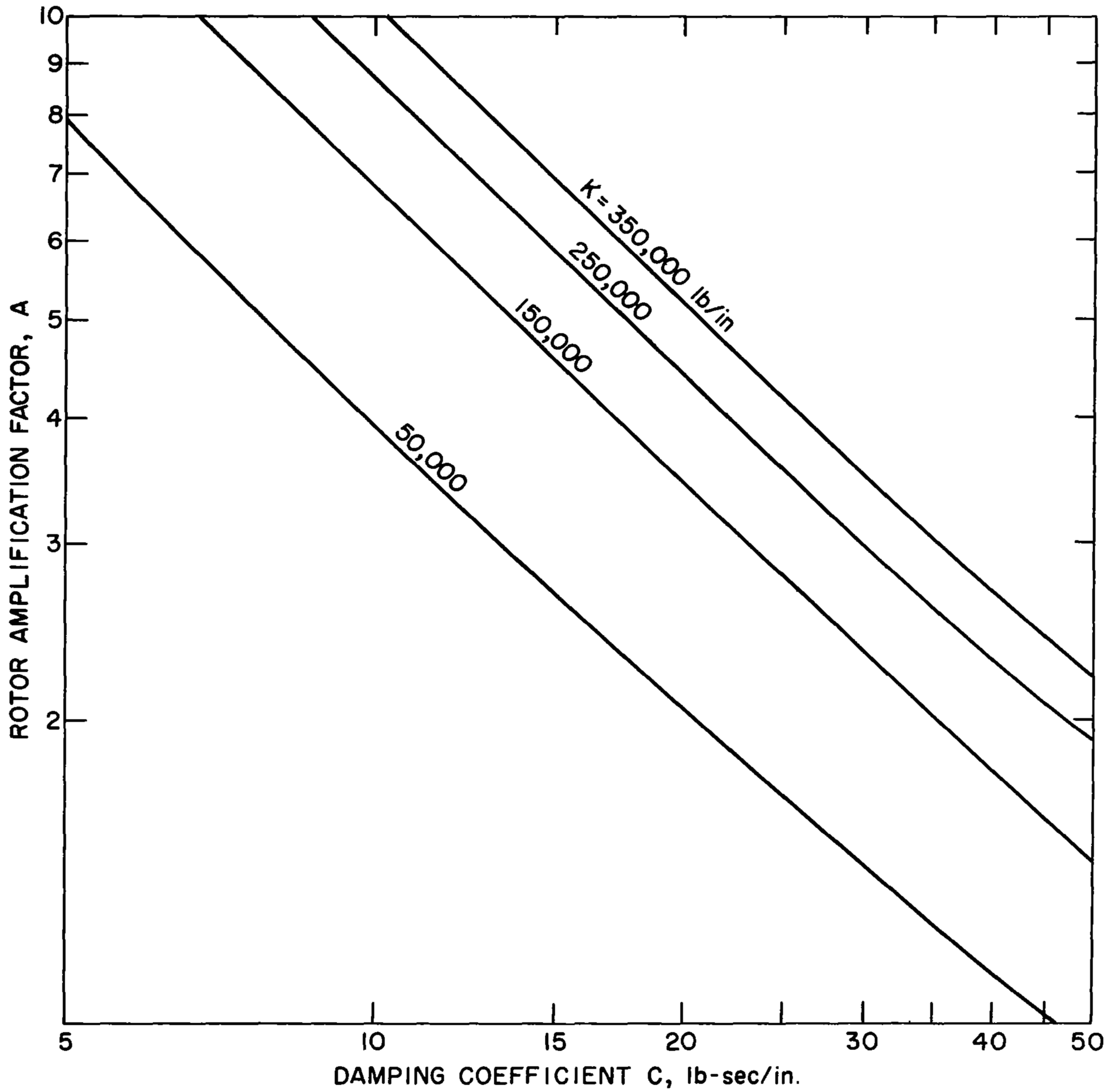


Figure 28. - Cylindrical critical speed amplification factor against damping coefficient.

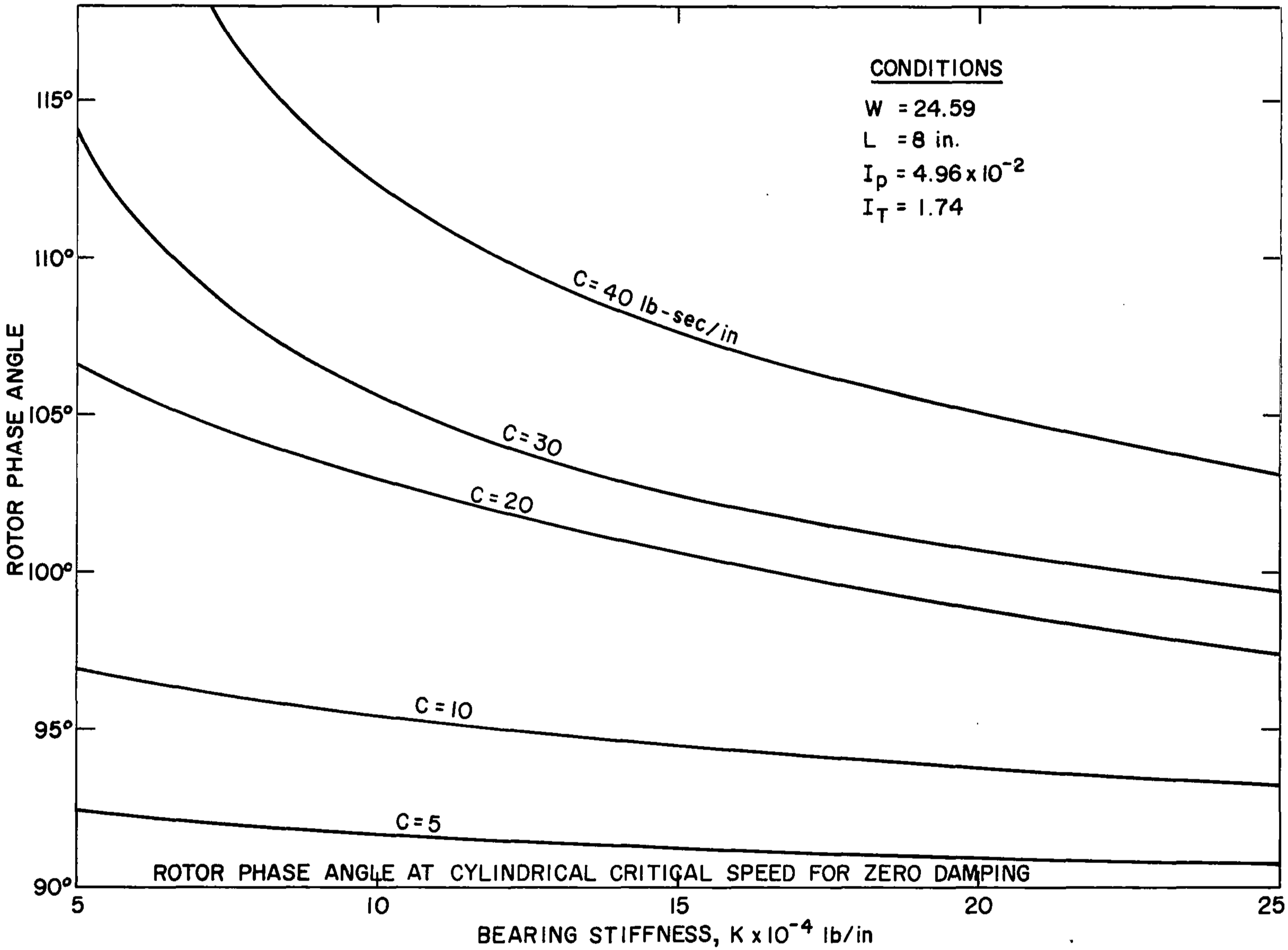


Figure 29. - Rotor phase angle at cylindrical critical speed against bearing stiffness.

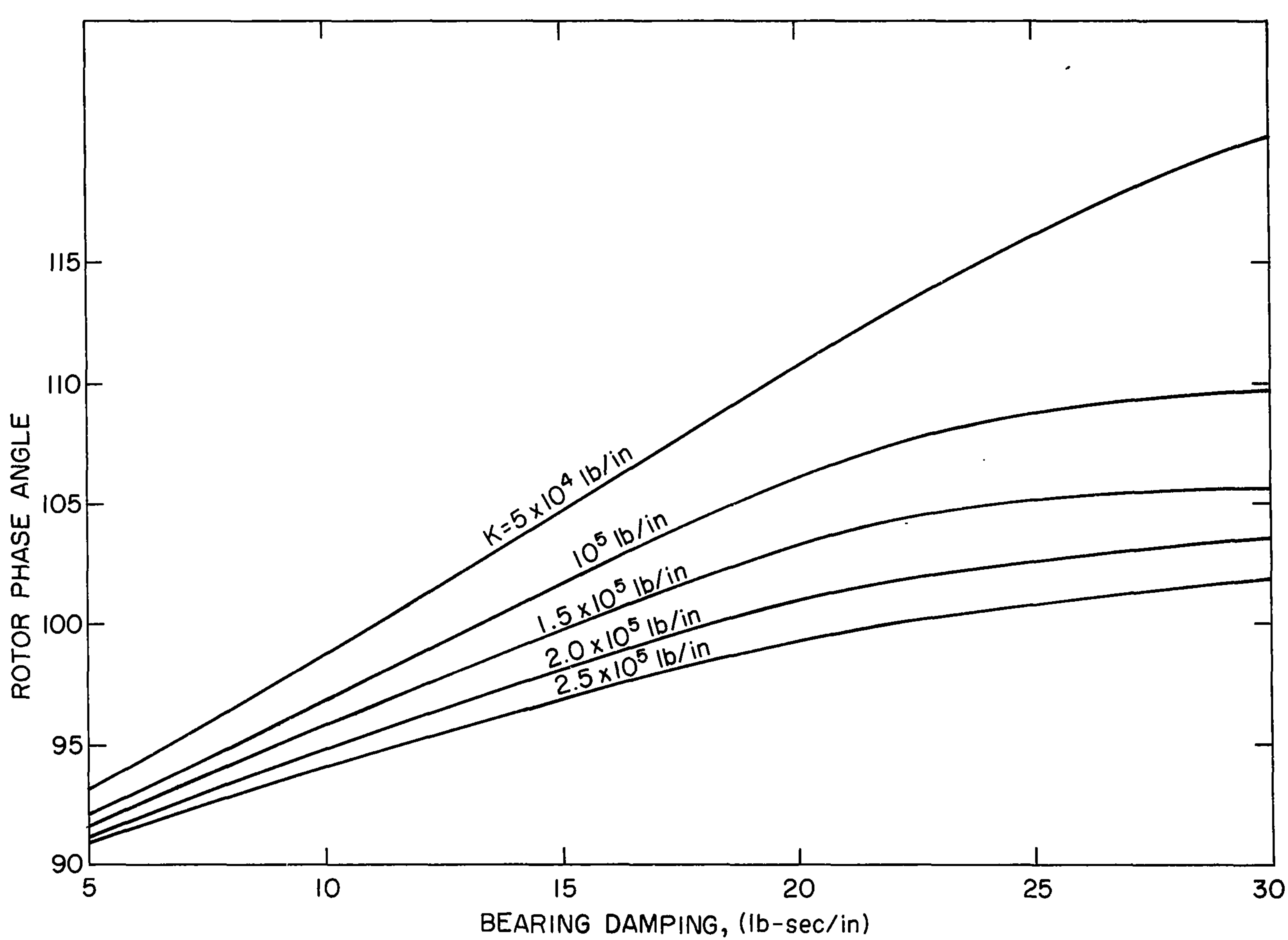


Figure 30. - Rotor phase angle at cylindrical critical speed against bearing damping.

PART V

STABILITY AND GENERAL TRANSIENT ANALYSIS

5.01 Stability Analysis of the System

In part III of this report a method to obtain the steady state solution of the system was shown. This, however, is insufficient to predict completely operation of the system. The stability characteristics of the system must also be known.

The homogeneous equations of motion are solved to get the time dependent transient solution. The homogeneous equations of motion for the general four degree of freedom system are:

$$\ell_1\ddot{x}_2 + \ell_2\ddot{x}_1 + \bar{K}_{1x}x_1 + \bar{K}_{2x}x_2 + \bar{R}_{1y}y_1 + \bar{R}_{2y}y_2 + \bar{C}_{1x}\dot{x}_1 + \bar{C}_{2x}\dot{x}_2 + \bar{D}_{1y}\dot{y}_1 + \bar{D}_{2y}\dot{y}_2 = 0 \tag{5.1}$$

$$\ell_1\ddot{y}_2 + \ell_2\ddot{y}_1 + \bar{K}_{1y}y_1 + \bar{K}_{2y}y_2 + \bar{R}_{1x}x_1 + \bar{R}_{2x}x_2 + \bar{C}_{1y}\dot{y}_1 + \bar{C}_{2y}\dot{y}_2 + \bar{D}_{1x}\dot{x}_1 + \bar{D}_{2x}\dot{x}_2 = 0 \tag{5.2}$$

$$\bar{R}_T(\ddot{x}_2 - \ddot{x}_1) + \bar{R}_p\omega(\dot{y}_2 - \dot{y}_1) + \bar{K}_{2x}\ell_2x_2 - \bar{K}_{1x}\ell_1x_1 + \bar{R}_{2y}\ell_2y_2 - \bar{R}_{1y}\ell_1y_1 + \bar{C}_{2x}\ell_2\dot{x}_2 - \bar{C}_{1x}\ell_1\dot{x}_1 + \bar{D}_{2y}\ell_2\dot{y}_2 - \bar{D}_{1y}\ell_1\dot{y}_1 = 0 \tag{5.3}$$

$$\bar{R}_T(\ddot{y}_2 - \ddot{y}_1) - \bar{R}_p\omega(\dot{x}_2 - \dot{x}_1) + \bar{K}_{2y}\ell_2y_2 - \bar{K}_{1y}\ell_1y_1 + \bar{R}_{2x}\ell_2x_2 - \bar{R}_{1x}\ell_1x_1 + \bar{C}_{2y}\ell_2\dot{y}_2 - \bar{C}_{1y}\ell_1\dot{y}_1 + \bar{D}_{2x}\ell_2\dot{x}_2 - \bar{D}_{1x}\ell_1\dot{x}_1 = 0 \tag{5.4}$$

To obtain the characteristic equation, we assume a solution of the form:

$$x_1 = A_1e^{\lambda t} \qquad x_2 = A_2e^{\lambda t} \qquad y_1 = A_3e^{\lambda t} \qquad y_2 = A_4e^{\lambda t} \tag{5.5}$$

If the above solution (5.5) is substituted into equations (5.1) to (5.4), equations are obtained which can be written in the following matrix form:

$$\begin{bmatrix} \lambda^2\ell_2 + \bar{K}_{1x} + \lambda\bar{C}_{1x} & \lambda^2\ell_1 + \bar{K}_{2x} + \lambda\bar{C}_{2x} & \lambda\bar{D}_{1y} + \bar{R}_{1y} & \lambda\bar{D}_{2y} + \bar{R}_{2y} \\ \lambda\bar{D}_{1x} + \bar{R}_{1x} & \lambda\bar{D}_{2x} + \bar{R}_{2x} & \lambda^2\ell_2 + \bar{K}_{1y} + \lambda\bar{C}_{1y} & \lambda^2\ell_1 + \lambda\bar{C}_{1y} + \bar{K}_{2y} \\ -\lambda^2\bar{R}_T - \lambda\bar{C}_{1x}\ell_1 - \bar{K}_{1x}\ell_1 & \lambda^2\bar{R}_T + \lambda\bar{C}_{2x}\ell_2 + \bar{K}_{2x}\ell_2 & -\lambda(\bar{R}_p\omega + \bar{D}_{1y}\ell_1) - \bar{R}_{1y}\ell_1 & \lambda(\bar{R}_p\omega + \bar{D}_{2y}\ell_2) + \bar{R}_{2y}\ell_2 \\ \lambda(\bar{R}_p\omega - \bar{D}_{1x}\ell_1) - \bar{R}_{1x}\ell_1 & -\lambda(\bar{R}_p\omega - \bar{D}_{2x}\ell_2) + R_{2x}\ell_2 & -\lambda^2\bar{R}_T - \lambda\bar{C}_{1y}\ell_1 - \bar{K}_{1y}\ell_1 & \lambda^2\bar{R}_T + \lambda\bar{C}_{2x}\ell_2 + \bar{K}_{2y}\ell_2 \end{bmatrix} \begin{bmatrix} A_1 \\ A_2 \\ A_3 \\ A_4 \end{bmatrix} = 0 \qquad (5.6)$$

In order to have a nontrivial solution the determinant of the coefficients A_1, A_2, A_3, and A_4 must vanish. This will give us the characteristic equation, which is of the form

$$A_0\lambda^8 + A_1\lambda^7 + A_2\lambda^6 + A_3\lambda^5 + A_4\lambda^4 + A_5\lambda^3 + A_6\lambda^2 + A_7\lambda + A_8 = 0 \qquad (5.7)$$

Knowing A_0, A_1, . . ., A_8, the above equation can be solved to get the roots of the equation, λ. λ is usually complex, and is of the form

$$\lambda = P + is$$

Here, P may be positive or negative or it may be zero. This, in fact, predicts the decay or growth rate of the motion of the system. If P is negative, then the system is stable, and the motion decays with time. If P is positive, then as is obvious, the motion grows with time and never comes to a stable situation. On the other hand, if P is zero then the system is said to be on the threshold of stability. The values of the parameters controlling the system keep it stable. If the parameters are changed, the system either becomes permanently stable or unstable.

The expansion of the characteristic determinant, as shown in equation (5.6) into the form of (5.7) even for this four degree of freedom system, by the usual procedure to expand a determinant, is formidable. This is much easier to do numerically on a digital computer. This has been done in the computer program ROTSTAB to compute numerically $A_0, A_1, \ldots, A_8$, and has been used to find the roots of the characteristic determinant. The real part of the root, as has already been observed, gives the decay or growth rate and the imaginary parts, the natural frequencies.

The computer program ROTSTAB is described in detail in 5.02.

Besides the above method of determining the stability criterion, the Routh Hurwitz criteria can be used to determine the stability of a system.

If the characteristic equation is given in the form

$$\sum_{K=0}^{N} A_{N-K}\lambda^{K} = 0 \qquad (5.8)$$

Then the Routh Hurwitz criterion is given by the following determinant:

$$D = \begin{array}{|cccccccc|} \multicolumn{1}{c}{D_0} & D_1 & D_2 & D_3 & D_4 & & & \\ A_1 & A_0 & 0 & 0 & 0 & 0 & \text{-----} & \\ A_3 & A_2 & A_1 & A_0 & 0 & 0 & \text{-----} & \\ A_5 & A_4 & A_3 & A_2 & A_1 & A_0 & \text{-----} & \\ A_7 & A_6 & A_5 & A_4 & A_3 & A_2 & A_1 & \text{--} \\ A_9 & A_8 & A_7 & A_6 & A_5 & \text{---} & \text{-----} & \end{array} \qquad (5.9)$$

The condition for stability is that the determinant D_N must be positive.

Thus,

$$\left.\begin{aligned} D_0 &= A_1 > 0 \\ D_1 &= A_1A_2 - A_0A_3 > 0 \\ D_2 &= A_3D_1 - A_1(A_1A_4 - A_0A_5) > 0 \\ D_3 &> 0, \text{ for stability} \end{aligned}\right\} \tag{5.10}$$

For systems larger than fourth order, the Routh-Hurwitz determinant method becomes cumbersome and unwieldy to use. It is often preferable in such cases to use the original Routh method as given below.

Consider the following array:

$$\left.\begin{matrix} A_0 & A_2 & A_4 & A_6 & A_8 \\ A_1 & A_3 & A_5 & A_7 & -- \\ C_1 & C_2 & C_3 & C_4 & -- \\ D_1 & D_2 & D_3 & -- & -- \\ E_1 & E_2 & E_3 & -- & -- \\ F_1 & F_2 & -- & -- & -- \\ G_1 & G_2 & -- & -- & -- \\ H_1 & -- & -- & -- & -- \end{matrix}\right\} \tag{5.11}$$

where

$$\left.\begin{aligned} C_1 &= A_2 - A_0A_3/A_1 \\ C_2 &= A_4 - A_0A_5/A_1 \\ C_3 &= A_6 - A_0A_7/A_1 \\ D_1 &= A_3 - A_1C_2/C_1 \\ D_2 &= A_5 - A_1C_3/C_1 \\ &\vdots \\ D_2 &= A_5 - A_1C_3/C_1 \end{aligned}\right\} \tag{5.12}$$

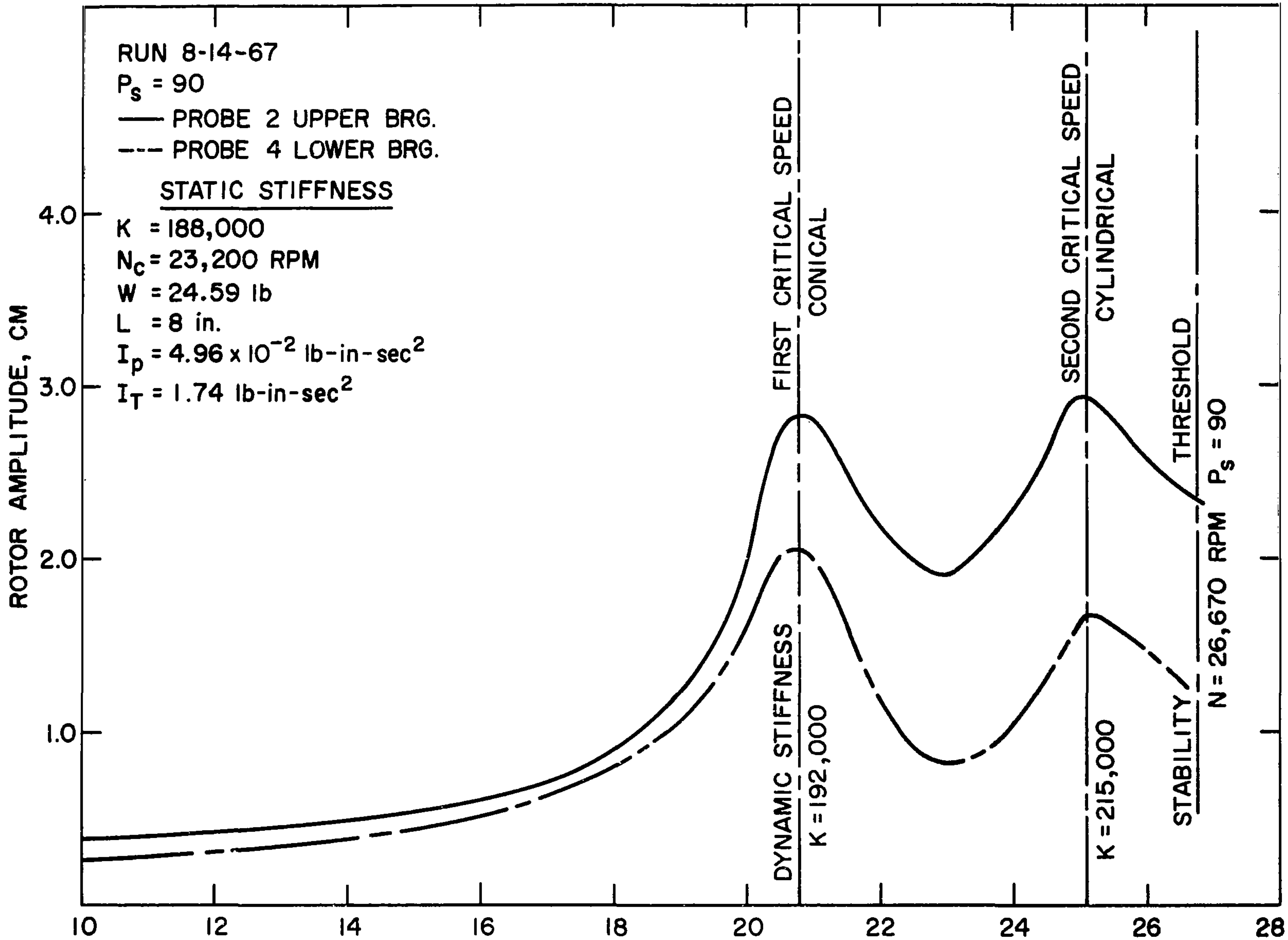


Figure 31. - Rotor motion of NASA gas bearing in system.

The necessary and sufficient condition for stability is that all of the coefficients of the first column of the array (5.11) must be positive.

This criterion of stability determination has been included in the program ROTSTAB.

5.02 Description of Computer Program ROTSTAB - Transient Solution of System

Figure 31 shows the experimental plot of the rotor amplitude against the rotor revolutions per minute of the NASA Lewis rotor. It was observed that the system goes unstable at 26 700 rpm. The stiffness values that have been indicated at the occurrence of the rotor critical speed have been obtained experimentally.

The numerical values of the stiffness and damping factors obtained experimentally are fed in as input data with different values of the cross coupling stiffness term in the ROTSTAB computer program to determine the attitude angle at which the rotor will become unstable. The output obtained at a running speed of 27 000 rpm is shown in appendix D. The calculated critical speeds follow quite closely those given by experimental results. The system is found to become unstable at a cross coupling stiffness of 127 000 pounds per inch. Hence the attitude angle will be given by

$$\Phi = \tan^{-1}\left(\frac{K}{R}\right) = \tan^{-1}\left(\frac{220\ 000}{127\ 000}\right) = 60^{\circ}$$

The whirl ratio at the threshold of stability is 0.73511, which corresponds closely with the experimental results.

As is obvious from the results obtained, this analysis provides the information required to change bearing characteristics to make an unstable system stable.

5.03 Special Case - Symmetric Bearing and Rotor

If we assume

$$\bar{K}_{1x} = \bar{K}_{2x} = \frac{\bar{K}_x}{2} \qquad \bar{C}_{1x} = \bar{C}_{2x} = \frac{\bar{C}_x}{2}$$

$$\bar{K}_{1y} = \bar{K}_{2y} = \frac{\bar{K}_y}{2} \qquad \bar{C}_{1y} = \bar{C}_{2y} = \frac{\bar{C}_y}{2}$$

$$\bar{R}_{1x} = \bar{R}_{2x} = \frac{\bar{R}_x}{2} \qquad \bar{D}_{1x} = \bar{D}_{2x} = \frac{\bar{D}_x}{2}$$

$$\bar{R}_{1y} = \bar{R}_{2y} = \frac{\bar{R}_y}{2} \qquad \bar{R}_{1y} = \bar{R}_{2y} = \frac{\bar{R}_y}{2}$$

and

$$\frac{L_1}{L} = \frac{L_2}{L} = \rho = 0.5$$

then the equations of motion are simplified considerably. From the above assumption of bearing symmetry we observe $x_1 = x_2 = x$ and $y_1 = y_2 = y$.

Equations (5.1) and (5.2) then reduce to:

$$\ddot{x} + \bar{K}_x x + \bar{R}_y y + \bar{C}_x \dot{x} + \bar{D}_y \dot{y} = 0 \tag{5.13}$$

$$\ddot{y} + \bar{K}_y y + \bar{R}_x x + \bar{C}_y \dot{y} + \bar{D}_x \dot{x} = 0 \tag{5.14}$$

Equations (2.42) and (2.43) reduce to:

$$I_T \ddot{\alpha}_1 + I_p \omega \dot{\alpha}_2 + \frac{K_x}{2} L^2 \alpha_1 + \frac{D_y}{2} L^2 \dot{\alpha}_2 + \frac{C_x}{2} L^2 \dot{\alpha}_1 + \frac{R_y}{2} L^2 \alpha_2 = 0 \tag{5.15}$$

$$I_T \ddot{\alpha}_2 - I_p \omega \dot{\alpha}_1 + \frac{K_y}{2} L^2 \alpha_2 + \frac{D_x}{2} L^2 \dot{\alpha}_1 + \frac{C_y}{2} L^2 \dot{\alpha}_2 + \frac{R_x}{2} L^2 \alpha_1 = 0 \tag{5.16}$$

It is to be noted that the two pairs of equations (5.13), (5.14) and (5.15), (5.16) are uncoupled. The first pair represents only the cylindrical mode and the second pair, the conical mode in a given system.

We now assume solutions of the form:

$$x = A_1 e^{\lambda t} \qquad y = A_2 e^{\lambda t} \qquad \alpha_1 = A_3 e^{\lambda t} \qquad \alpha_2 = A_4 e^{\lambda t} \tag{5.17}$$

Substituting (5.17) into equations (5.13) and (5.14), we obtain the characteristic equation for the cylindrical mode which is

$$\lambda^4 + \lambda^3 \left[\bar{C}_y + C_x\right] + \lambda^2 \left[K_y + K_x + C_x C_y - D_x D_y\right] + \lambda \left[K_y C_x + K_x C_y - R_y D_x - R_x D_y\right] + \left[K_x K_y - R_x R_y\right] = 0 \tag{5.18}$$

Substituting equation (5.17) in equations (5.15) and (5.16) and after some algebraic manipulations, the characteristic equation for conical mode is obtained:

$$R_T^2\lambda^4 + R_T(C_x + C_y)\lambda^3 + \lambda^2\left[R_T(K_x + K_y) + C_xC_y + (R_p\omega + D_y)(R_p\omega - D_x)\right]$$

$$+ \lambda\left[K_xC_y + K_yC_x + R_y(R_p\omega - D_x) - R_x(R_p\omega + D_y)\right] + (K_xK_y - R_xR_y) = 0 \qquad (5.19)$$

where

$$R_T = \left(\frac{2K_T}{L}\right)^2$$

$$R_P = \left(\frac{2K_P}{L}\right)^2$$

and

$$I_T = MK_T^2$$

$$I_P = MK_P^2$$

$$\bar{K}_x = \frac{2K_x}{M}$$

$$\bar{C}_x = \frac{2C_x}{M}$$

etc.

5.04 Computer Program to Find Stability of Symmetric System (STABIL4)

This program uses equations (5.18) and (5.19) for the stability analysis of a symmetric bearing system. The cylindrical and conical modes are evaluated separately. The real part of the roots gives the damping or growth rate and the imaginary part, the natural frequency of the system. If the real part of the root is negative, then the system is stable; if positive, it is unstable, and if zero, the system is neutrally stable.

The input data to the program is as follows:

Card 1

1. N - Highest power of the polynomial (in this case, always 4)

Card 2

1. K_x - Stiffness in x-direction, lb/in.
2. K_y - Stiffness in y-direction, lb/in.

Card 3

1. C_x - Damping coefficient in x-direction, lb-sec/in.
2. C_y - Damping coefficient in y-direction, lb-sec/in.

Card 4

1. R_x - Cross coupling stiffness in x-direction, lb/in.
2. R_y - Cross coupling stiffness in y-direction, lb/in.

Card 5

1. D_x - Cross coupling damping coefficient in x-direction, lb-sec/in.
2. D_y - Cross coupling damping coefficient in y-direction, lb-sec/in.

Card 6

1. L - Length between bearings, in.

Card 7

1. W - Weight of the rotor, lb

Card 8

1. I_P - Polar moment of inertia of the rotor, lb-sec/in.^{2}
2. I_T - Transverse moment of inertia of the rotor, lb-sec/in.^{2}

Card 9

1. OMEGA - Angular speed of the rotor, rps

For the first case, card 1 should be included, and for each additional cases cards 2 though 9 must be punched with proper data.

The data cards are in free field format. A comma should separate all data entries. A comma is required after the last data entry.

The output data is as follows:

Cylindrical mode:

1. The coefficients of the polynomial in ascending power
2. Column 1 - Real part of the roots

   Column 2 - Imaginary part of the roots (cylindrical natural frequencies)

Conical mode:

1. The coefficients of the polynomial in ascending power
2. Column 1 - Real part of the roots

   Column 2 - Imaginary part of the roots (conical natural frequencies)

The heading printout is as follows:

Line 1 - K_x, K_y, R_x, R_y

Line 2 - C_x, C_y, D_x, D_y

Line 3 - I_P, I_T, L, W

Line 4 - Speed, rps

PART VI

CONCLUSIONS AND SCOPE

1. The equations of motion that have been presented here consider 13 degrees of freedom, taking into account the axial movement of the system and the eight degrees of freedom for the two bearing housings. Equations (2. 25) to (2. 38) represent the generalized system equations of motion. The steady state analysis in this report assumes a constant angular speed and rigid housing. For preliminary design analysis of a rigid body rotor bearing system, the curves shown in figures 23 to 30 (pp. 55 to 62) are useful in finding the critical speeds for certain bearing characteristics.

Figures 7 to 16 can then be used for investigating the amplitudes, phase angles, and force transmitted for the speed range in which the rotor is expected to operate.

2. The analysis does not consider any particular type of bearing, but the equations of motion can be applied to any type of rotor-bearing system. In order to investigate the steady state and transient behavior of the rotor, the Reynold's equation must be included and solved to obtain the pressure distribution and the radial and tangential forces in order to find the bearing characteristics. These can be utilized to solve the steady state and transient equations for the system.

3. The assumption of a rigid bearing housing can be discarded, retaining the assumptions of small amplitude and constant rotor speed. This results in twleve coupled linearized second order equations. The axial motion equation, being uncoupled from the rest of the system equation, can be solved independently. These twelve equations can be used to investigate the effect of the flexible housing on the entire system.

4. The twelve linearized equations of motion can be further investigated in order to find the threshold of stability by applying Routh's criteria. By varying the various bearing parameters, the threshold of stability can be obtained and the optimum bearing characteristics for stable operation of the system determined.

5. The nonlinear equations of motion can be further analyzed to obtain the time-transient solution by numerical integration. Being time consuming, this may be applied only in particular critical situations. This orbital analysis will further supplement the threshold of stability analysis as indicated previously. The possibility of obtaining time-transient solution by numerical integration may further be extended to observe the effects of shock loading on the system.

6. Figures 20 and 21 show the transmissibility and force transmitted against frequency ratio curves for a single degree of freedom system. These curves are useful

for finding the optimum damping values if the rotor is expected to operate over a wide speed range. These curves give an approximate idea of the rotor-bearing behavior if, by making simplifying assumptions, the system is reduced to a single degree freedom. In order to get more accurate data for the opimization of damping values, the program ROTOR4P could be extended to plot similar curves.

7. The plots of phase angle between response and excitation show that they may exceed 180°. This differs from the results of simplified analyses. These phase angle plots may be used to predict whether a system will go unstable in a cylindrical or conical mode.

8. The derived equations of motion can be utilized to investigate further the effect of gyroscopic forces on the system.

9. The analysis and design data presented in this report are applicable to a general RIGID-BODY rotor bearing system. However, they can be extended to a flexible rotor-bearing system as indicated by Poritsky in his simplified analysis in reference 7.

REFERENCES

1. Goldstein, Herbert: Classical Mechanics, Addison-Wesley Publishing Company, Inc., April 1965.
2. Gunter, Edgar J., Jr.: Dynamic Stability of Rotor-Bearing Systems. NASA SP-113, 1966.
3. Gunter, Edgar J., Jr., Choudhury, P. De; Kirk, R. G.: Dynamics of Rotors Supported on Tilting Pad Gas Bearings - System Resonance Frequencies. Proposed paper for ASLE Lubrication Conf., Houston, Texas, Oct. 1969.
4. Routh, E.J.: Advanced Rigid Dynamics, Macmillan and Co., 1892.
5. Tondl, A.: Some Problems of Rotor Dynamics, Chapman & Hall, London, 1965.
6. Timoshenko, S.: Vibration Problems in Engineering, D. Van Nostrand Co., 1965.
7. Poritsky, H.: Rotor-Bearing Dynamics Design Technology. Part II - Rotor Stability Theory. Air Force Aeropropulsion Laboratory. TR-65-45, Part II, May 1965.

APPENDIX A

A-01 DERIVATION OF KINETIC ENERGY OF ROTATION OF ROTOR

Consider x, y and z the fixed reference frame. If we assume that the rotor undergoes small angular displacements α_1 in the x-z plane and α_2 in the y-z plane, then in order to arrive at the kinetic energy of rotation of the rigid rotor, it is necessary to express the resultant angular velocity fixed in the body. Let $\vec{n}_x$, $\vec{n}_y$, and $\vec{n}_z$ be the unit vectors in the direction of the fixed reference frame as shown in figure 32. In this figure the final configuration of the rotor is shown. To arrive at the expression for

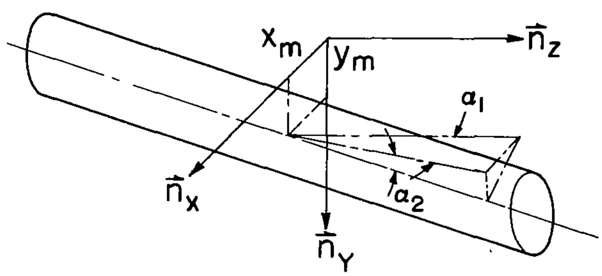


Figure 32. - Fixed reference frame.

angular velocity vector with reference to the axes fixed in the body, consider three angular rotations, one at a time.

Figure 33(a) shows the first rotation α_1 in the x-z plane and figure 33(b) shows the second rotation α_2 in the y-z plane. Then the rotor is rotated about its axis by an angle α_3.

The angular velocity vector with respect to the body axes will have three components along $\vec{n}''_y$, $\vec{n}''_x$, and $\vec{n}''_z$. Now

$$\vec{\Omega} = -\dot{\alpha}_2 \vec{n}'_x + \dot{\alpha}_1 \vec{n}_y + \dot{\alpha}_3 \vec{n}''_z \tag{A-1}$$

From figure 33(c)

$$\vec{n}'_x + \vec{n}''_x \cos \alpha_3 - \vec{n}''_y \sin \alpha_3 \tag{A-2}$$

From figure 33(b)

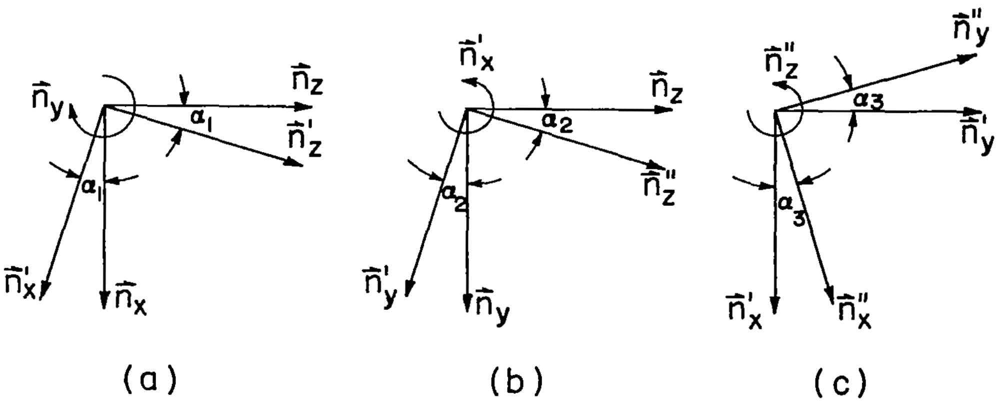


Figure 33. - Rotations from fixed reference frame.

$$\vec{n}_y = \vec{n}'_y \cos \alpha_2 + \vec{n}''_z \sin \alpha_2 \tag{A-3}$$

From fig. 33(c)

$$\vec{n}'_y = \vec{n}''_y \cos \alpha_3 + \vec{n}''_x \sin \alpha_3 \tag{A-4}$$

Substituting equation (A-4) in (A-3) we obtain:

$$\vec{n}_y = \vec{n}''_x \sin \alpha_3 \cos \alpha_2 + \vec{n}''_y \cos \alpha_2 \cos \alpha_3 + \vec{n}''_z \sin \alpha_2 \tag{A-5}$$

Substituting (A-2) and (A-5) in equation (A-1) we obtain the angular velocity vector fixed with the body

$$\Omega = \vec{n}''_x\left[-\dot{\alpha}_2 \cos \alpha_3 + \dot{\alpha}_1 \sin \alpha_3 \cos \alpha_2\right] + \vec{n}''_y\left[\dot{\alpha}_2 \sin \alpha_3 + \dot{\alpha}_1 \cos \alpha_3 \cos \alpha_2\right] + \vec{n}''_z\left[\dot{\alpha}_1 \sin \alpha_2 \cos \alpha_2 + \dot{\alpha}_3\right] \tag{A-6}$$

Since α_1, α_2 and α_3 are assumed small, equation (A-6) can be written as

$$\vec{\Omega} = -\dot{\alpha}_2 \vec{n}''_x + \dot{\alpha}_1 \cos \alpha_2 \vec{n}''_y + (\dot{\alpha}_1 \sin \alpha_2 + \dot{\alpha}_3) \vec{n}''_z \tag{A-7}$$

The kinetic energy of rotation of the rotor is given by:

$$T_R = \frac{1}{2} I''_x \omega''^2_x + \frac{1}{2} I''_y \omega''^2_y + \frac{1}{2} I''_z \omega''^2_z \tag{A-8}$$

From (A-7)

$$\omega_x'' = -\dot{\alpha}_2$$

$$\omega_y'' = \dot{\alpha}_1 \cos \alpha_2$$

$$\omega_z'' = \dot{\alpha}_1 \sin \alpha_2 + \dot{\alpha}_3$$

For a rotor

$$I_x'' + I_y'' = I_T = \text{Transverse moment of inertia}$$

$$I_z'' = I_P = \text{Polar moment of inertia of the rotor}$$

$$\therefore \; T_R = \frac{1}{2} I_T \left(\omega_x''^2 + \omega_y''^2\right) + \frac{1}{2} I_P \, \omega_z''^2$$

$$= \frac{1}{2} I_T \left(\dot{\alpha}_2^2 + \dot{\alpha}_1^2 \cos^2 \alpha_2\right) + \frac{1}{2} I_P \left(\dot{\alpha}_1 \sin \alpha_2 + \dot{\alpha}_3\right)^2 \qquad \text{(A-9)}$$

A. 02 Derivation of Kinetic Energy of Unbalance Masses

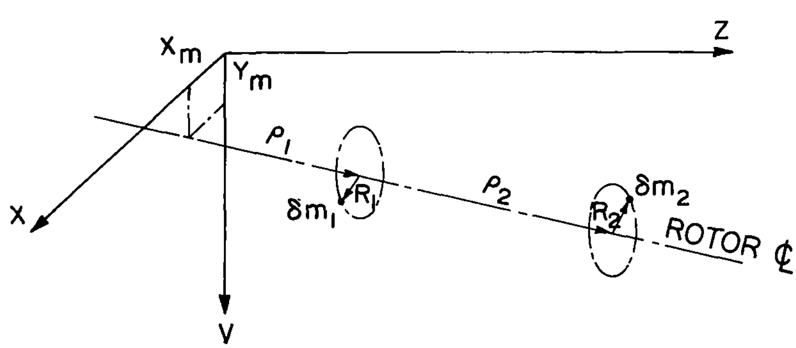


Figure 34. - Location of rotor unbalance masses.

The position vector of the first unbalance mass is given by

$$\vec{P}^{\delta M_1/0} = \rho_1 \vec{n}_z'' + R_1 \vec{e}_1$$

$$= \rho_1 \vec{n}_z'' + R_1(\cos \alpha_3 \vec{n}_x'' + \sin \alpha_3 \vec{n}_y'') \qquad \text{(A-10)}$$

The velocity of δM_1 is given by

$$^{R}\vec{V}^{\delta M_1/0} = \frac{^{R'}\delta \vec{P}}{\delta t} + {}^{R}\omega^{R'} \times \vec{P}^{\delta M_1/0} \tag{A-11}$$

From equation (A-7)

$$^{R}\omega^{R'} = -\dot{\alpha}_2\, \vec{n}_x'' + \dot{\alpha}_1 \cos \alpha_2\, \vec{n}_y'' + (\dot{\alpha}_1 \sin \alpha_2 + \dot{\alpha}_3)\, \vec{n}_z''$$

$$\simeq -\alpha_2\, \vec{n}_x'' + \alpha_1\, \vec{n}_y'' + \alpha_3\, \vec{n}_z'' \tag{A-12}$$

$$\therefore\ {}^{R}\omega^{R'} x \vec{P} = \begin{vmatrix} \vec{n}_x'' & \vec{n}_y'' & \vec{n}_z'' \\ -\dot{\alpha}_2 & \dot{\alpha}_1 & \dot{\alpha}_3 \\ R_1 \cos \alpha_3 & R_1 \sin \alpha_3 & \rho_1 \end{vmatrix}$$

$$= \vec{n}_x''(\dot{\alpha}_1\rho_1 - \dot{\alpha}_3 R_1 \sin \alpha_3) + \vec{n}_y''(\dot{\alpha}_2\rho_1 + \dot{\alpha}_3 R_1 \cos \alpha_3)$$

$$+ \vec{n}_z''(-\alpha_2 R_1 \sin \alpha_3 - \alpha_1 R_1 \cos \alpha_3) \tag{A-13}$$

The total velocity of δM_1 is, therefore, from equation (A-11),

$$^{R}\vec{V}^{\delta M_1/0} = (\dot{x}_m + \rho_1\dot{\alpha}_1 - \dot{\alpha}_3 R_1 \sin \alpha_3)\vec{n}_x'' + (\dot{y}_m + \rho_1\dot{\alpha}_2 + R_1\dot{\alpha}_3 \cos \alpha_3)\vec{n}_y''$$

$$+ (\dot{z}_m - \dot{\alpha}_2 R_1 \sin \alpha_3 - \dot{\alpha}_1 R_1 \cos \alpha_3)\, n_z'' \tag{A-14}$$

Assuming that the second unbalance mass is displaced from the first by a phase angle Φ, then the velocity of the second unbalance mass is given by:

$$^{R}\vec{V}^{\delta M_2/0} = \left[\dot{x}_m + \rho_2\dot{\alpha}_1 - \dot{\alpha}_3 R_2 \sin(\alpha_3 + \Phi)\right] \vec{n}_x'' + \left[\dot{y}_m + \rho_2\dot{\alpha}_2 + \dot{\alpha}_3 R_2 \cos(\alpha_3 + \Phi)\right] \vec{n}_y''$$

$$+ \left\{\dot{z}_m - R_2\left[\dot{\alpha}_2 \sin(\alpha_3 + \Phi) + \dot{\alpha}_1 \cos(\alpha_3 + \Phi)\right]\right\} \vec{n}_z'' \tag{A-15}$$

The kinetic energy of the unbalance masses is then given by:

$$T_U = \frac{1}{2}\,\delta M_1\,\vec{V}^{\delta M_1}\cdot\vec{V}^{\delta M_1} + \frac{1}{2}\,\delta M_2\,\vec{V}^{\delta M_2}\cdot\vec{V}^{\delta M_2}$$

$$= \frac{1}{2}\,\delta M_1 \Big[(\dot{x}_m + \rho_1\dot{\alpha}_1 - \alpha_3 R_1 \sin\alpha_3)^2 + (\dot{y}_m + \rho_1\dot{\alpha}_2 + R_1\dot{\alpha}_3 \cos\alpha_3)^2$$

$$+ (\dot{z}_m - \dot{\alpha}_2 R_1 \sin\alpha_3 - \dot{\alpha}_1 R_1 \cos\alpha_3)^2\Big]$$

$$+ \frac{1}{2}\,\delta M_2 \Big(\big[\dot{x}_m + \rho_2\dot{\alpha}_1 - \dot{\alpha}_3 R_2 \sin(\alpha_3 + \Phi)\big]^2 + \big[\dot{y}_m + \rho_2\dot{\alpha}_2 + \alpha_3 R_2 \cos(\alpha_3 + \Phi)\big]^2$$

$$+ \Big\{\dot{z}_m - R_2\big[\dot{\alpha}_2 \sin(\alpha_3 + \Phi) + \dot{\alpha}_1 \cos(\alpha_3 + \Phi)\big]\Big\}^2\Big) \qquad \text{(A-16)}$$

The kinetic energy of unbalance can be written in more general form with unbalance masses as:

$$T_U = \frac{1}{2}\sum_{i=1}^{N} \delta M_i \Big\{\big[\dot{x}_m + \rho_i\dot{\alpha}_1 - \dot{\alpha}_3 R_i \sin(\alpha_3 + \Phi_i)\big]^2 + \big[\dot{Y}_m + \rho_i\dot{\alpha}_2 + R_i\dot{\alpha}_3 \cos(\alpha_3 + \Phi_i)\big]^2\Big\}$$

$$+ \big[\dot{z}_m - \dot{\alpha}_2 R_i \sin(\alpha_3 + \Phi_i) - \dot{\alpha}_1 R_i \cos(\alpha_3 + \Phi)\big]^2 \qquad \text{(A-17)}$$

The phase angles Φ_i's are measured with respect to the first unbalance mass; hence, $\Phi_1 = 0$.

APPENDIX B

LISTING AND SAMPLE OUTPUT OF COMPUTER PROGRAM ROTOR4P

PLOT PACKAGE

```
COMMENT         THE DRC CALCOMP PACKAGE HAS BEEN INSERTED AT THIS POINT;
ALPHA FILE OUT  PLOTTER 2 ( 1, 376, SAVE 1 );
FILE OUT SPO 11(1,10);
PROCEDURE       SYMBOL(XO, YO, HGT, BCD, THETA, N);
VALUE           XO, YO, HGT, THETA, N;
INTEGER         N;   ALPHA ARRAY BCD[0];
REAL            XO, YO, HGT, THETA;        FORWARD;
INTEGER ARRAY   PLOTARREYA[0: 502],PLOTARREYA1, PLOTARREYA2[0:8];
INTEGER ARRAY   SYMBOLARREYA[0:112], SYMBOLARREYB[-15:63];
ALPHA ARRAY     PLOTTERBCDEF[0:3];
PROCEDURE      PLOT(X,Y,IC)  ;
VALUE IC;          REAL X,Y  ;  INTEGER IC   ;
         BEGIN
PROCEDURE TALK;
BEGIN
                ARRAY MESS[0:10]; INTEGER ILK;
     FILL MESS[*] WITH "OPERATOR, SET PLOTTER TAPE TO LOW DENSITY AND PU
RGE.   ←";
     WRITE(SPO,7,MESS[*]);
END;
DEFINE          M = 498 # ;  COMMENT  M+2 MUST BE A MULTIPLE OF 4;
COMMENT        UPPER BOUND FOR PLOTARREYA MUST BE AT LEAST M+1;
COMMENT        BUFFER SIZE MUST BE AT LEAST 3×(M+2)/4+1;
LABEL           AUS, FINSH, UN, L1, FIRST;
DEFINE          A  =  PLOTARREYA #,
                A1 =  PLOTARREYA1 #,
                A2 =  PLOTARREYA2 #;
DEFINE          BLKNO = PLOTTERBCDEE #;
PROCEDURE       PACK (A, N);   VALUE  N;
        ARRAY A[0];           INTEGER  N;
          BEGIN
INTEGER         I,J;
                I ← -1;  J ← 0;    WHILE  J ≤ N  DO
          BEGIN
                A[I←I+1] ← A[J+1].[12:12] & A[J][1:13:35]; J ← J + 1;
                A[I←I+1] ← A[J+1].[12:24] & A[J][1:25:23]; J ← J + 1;
                A[I←I+1] ← A[J+1] & A[J][1:37:11];  J ← J + 2
          END
          END   OF PACK;
OWN BOOLEAN     FIXED, NUTAPE, BOOL, PEN;
OWN INTEGER     I, NPX, NPY, RA, T, BUF;
OWN REAL        LENGTH, RECORD;
INTEGER          J,K,JJ,NX,NY,DX,DY,IX,IY,NR,NT,NC,II1,II2,NA,JJ1  ;
INTEGER         SNPY, SNPX, ISEE;
DEFINE         TAPECK =
               LENGTH ← LENGTH + RECORD; NUTAPE ← LENGTH > 900   #;
DEFINE         NUBUFF =
                BEGIN A[I] ← "34←    ";    PACK (A, M+1);
                WRITE (PLOTTER, BUF, A[*]);
                TAPECK; STARTPLOT  END   # ;
DEFINE         STARTPLOT =
                A[0] ← "444444"; A[1] ← "444433"; A[2] ← "333332";
                I ← 3  #  ;

                IF IC > 3 OR IC <-5  THEN GO TO AUS;
                ISEE ← IC;
```

```
                IF IC < -3  THEN  IC ← -3;
                IF NOT FIXED THEN
         BEGIN      LABEL  DUMMY;
FILL A1[*] WITH     OCT50500000000,OCT50600000000,OCT50700000000  ,
                    OCT60500000000,OCT60600000000,OCT60700000000  ,
                    OCT70500000000,OCT70600000000,OCT70700000000  ;
FILL A2[*] WITH     OCT50500,OCT50600,OCT50700,OCT60500,OCT60600,
                    OCT60700,OCT70500,OCT70600,OCT70700  ;
               NPX ← NPY ←  0;  BA ← IF ISEE = -4 THEN 0 ELSE 1;
               LENGTH ← 0; NUTAPE ← FALSE; RECORD ←(6×(M+1)/200+0.75)/12;
                BUF ← 3 × (M+2) / 4 + 1;  BOOL ← PEN ← TRUE;
                FOR I ← 1 STEP 1 UNTIL 5 DO
          BEGIN
                WRITE (PLOTTER, BUF, A[*]);
                IF TIME(1) - T < .017 × BUF + 1 AND I > 2   THEN
          BEGIN
                CLOSE(PLOTTER,SAVE);    I ← 0;
TALK;
          END;
                T ← TIME(1);
          END;
                GO TO FINSH;
         END ;
FIRST:          FIXED ← TRUE;
                IF IC = 0  THEN
          BEGIN
                X ← NPX/100;  Y ← NPY/100;  GO TO AUS
          END;
                IF BOOL THEN T← "006006";
               IF ABS(IC) = 2 THEN
         BEGIN
               IF NOT PEN THEN GO TO ON  ;
                IF BOOL THEN T← "0007006" ELSE T ← T + 1;
         END
               ELSE IF ABS(IC) = 3 THEN
         BEGIN
               IF PEN THEN GO TO ON   ;
                IF BOOL THEN T←  "005006" ELSE T ← T - 1;
         END
               ELSE GO TO ON  ;
               PEN ← NOT PEN  ;
                A[I] ← IF BOOL THEN T + "660660" ELSE T + "000660";
               BOOL ← TRUE  ;
               I ← I + 1  ;
               JJ ← IF ABS(IC) = 2  THEN  8  ELSE  2;
               FOR K ← 1 STEP 1 UNTIL JJ  DO
         BEGIN
               IF I > M  THEN  NUBUFF;
                A[I] ← "666666";  I ← I + 1;
         END ;
                T ← "006006";
                I ← I - 1;
ON:            NX ← 100.0 × X  ;  NY ← 100.0 × Y  ;
               DX ← NX - NPX  ;  DY ← NY - NPY  ;
               NPX ← NX  ;  NPY ← NY  ;
               IF DX ≥ 0 THEN
```

```
            IF DX = 0 THEN IX ← 3 ELSE IX ← 6 ELSE IX ← 0  ;
            IF DY ≥ 0 THEN
            IF DY = 0 THEN IY ← IX + 1 ELSE IY ← IX + 2 ELSE IY ← IX ;
            IF ABS(DX) ≥ ABS(DY) THEN
        BEGIN
            NR ← ABS(DY)  ;  NC ← NT ← ABS(DX)  ;
            IX ← IX + 1  ;
        END
            ELSE
        BEGIN
            NR ← ABS(DX)  ;  NC ← NT ← ABS(DY)  ;
            IX ← IY - IX + 3  ;
        END ;
            NA ← NT DIV 2  ;
L1:         IF NC ≠ 0 THEN
        BEGIN
            NA ← NA + NR  ;
            IF NA ≥ NT THEN
        BEGIN
            IF BOOL THEN T ← T + A1[IY]
            ELSE T ← T + A2[IY]  ;
            NA ← NA - NT  ;
        END
            ELSE
        BEGIN
            IF BOOL THEN T ← T + A1[IX]
            ELSE T ← T + A2[IX]  ;
        END ;
            BOOL ← NOT BOOL  ;
            IF BOOL THEN
        BEGIN
            A[I] ← T  ;  I ← I +1  ;
            T ← "006006";
            IF I > M  THEN  NUBUFF;
        END ;
            NC ← NC - 1  ;
            GO TO L1  ;
        END;
            IF NUTAPE  AND  ABS(IC) = 3  THEN
        BEGIN
            NUTAPE  ← FALSE;  LENGTH ← 0;
            SNPX ← NPX;  SNPY ← NPY;
     PLOT(0,0,-1);
             LOCK (PLOTTER, SAVE);   BA ← BA -2;
            PLOT (0,0,-1);  PLOT (SNPX/100, SNPY/100,  1);
        END;
            IF IC < 0 THEN
        BEGIN
            IF BOOL THEN I ← I - 1  ELSE  A[I] ← T + "000660";
             I ← I + 1 ;     NUBUFF ;
            BOOL ← TRUE  ;
            NPX ← NPY ← 0  ;
             IF ISEE > -5  THEN
         BEGIN
FINSH:
             JJ←BA; NA←0;
```

```
                FOR K ← 0, 2, 4  DO
          BEGIN
                J ← JJ MOD 10;    JJ ← JJ DIV 10;
                NA ← NA + ((4 + J MOD 4) + (4 + J DIV 4) × 64) × 64 * K
          END;
                STARTPLOT;  A[60] ← A[2];  A[2] ← A[2] - 1;  A[3] ← NA;
                A[4] ← "133333";  A[5] ← "334444";  A[59] ← A[1];
                FOR JJ ← 6 STEP 1 UNTIL 58  DO A[JJ]← A[0];
                BA ← BA + 1;  I ← 61;   NUBUFF
          END;
                IF ISEE = -4 AND FIXED  THEN
          BEGIN
                BLKNO[0] ← "START ";  BLKNO[1] ← "OF BLO";
                BLKNO[2] ← "CK 000";  JJ ← BA - 1;
                FOR K  ← 0,1,2   DO
          BEGIN
                J  ← JJ MOD 10;  JJ ← JJ DIV 10;
                BLKNO[2] ← BLKNO[2] + J × 64 * K
          END;
                SYMBOL (0.1,1.4,0.07, BLKNO, 270, 18);
                PLOT (1,0,-5)
          END ;
                IF NOT FIXED THEN  GO TO FIRST;
          END;

AUS:      END  OF PLOT;

PROCEDURE       SYMBOL(XO, YO, HGT, BCD, THETA, N) ;
VALUE           XO, YO, HGT, THETA, N ;
INTEGER         N ;
REAL            XO, YO, HGT, THETA ;
ALPHA ARRAY     BCD[0] ;
           BEGIN
INTEGER         BINX, AC, W, OSC, AINX, I, MOVE;
REAL            XA, YA, X, Y, XN, YN, OSTS;
OWN  BOOLEAN    FIXED;  BOOLEAN  LP, M7;
DEFINE          A = SYMBOLARREYA #, B = SYMBULARREYB #;
LABEL           Y1, EL, EXIT, LOADB;
                IF NOT FIXED THEN
               FILL A[*] WITH OCT103041463717060, OCT110000000000000,
               OCT103020271600000,OCT400001454637170,OCT605000000000000.
               OCT011030414334143,OCT445463717060000,OCT070343333730204,
               OCT000000000000000,OCT011030414334040,OCT747000000000000,
               OCT031434434130100,OCT106173746000000,OCT060747212000000,
               OCT344341301001031,OCT434454637170605,OCT140000000000000,
               OCT011030414637170,OCT604133344000000,OCT111514041412024,
               OCT232313534440400,OCT313313114144040,OCT040000000000000,
               OCT000000000000000,OCT101121201070222,OCT345463717060000,
               OCT111222211170141,OCT525241400000000,OCT024406000000000,
               OCT014170064402000,OCT212523034300000,OCT000343463717060,
               OCT343400000000000,OCT040737464534040,OCT030414334000000,
               OCT424130100106173,OCT746450000000000,OCT000737464130000,
               OCT470704340400400,OCT470704340400000,OCT433343413010010,
               OCT617374645000000,OCT000704444740000,OCT103020271737000,
               OCT102021111000000,OCT301017370000000,OCT362717060540314,
               OCT220100103140000,OCT301215370000000,OCT460442000000000,
```

```
        OCT44041513040000O,OCT01452305410000O,OCT01103041470000O,
        OCT00070347254000O,OCT07004000000000,OCT00072347400000,
        OCT00074047000000,OCT10304146377036 4,OCT77037170601100O,
        OCT00073746453404O,OCT10010617374641 3,OCT01070224000000,
        OCT00073746453404 3,OCT44340000000000,OCT02324334140516 4,
        OCT62627200000000,OCT01452305412303 4,OCT32325210000000,
        OCT03430000000000,OCT10323517000000,OCT10212212112170 2,
        OCT42515142400000,OCT01417042044600O,OCT00000000000000,
        OCT00470000000000,OCT01103041433414O,OCT50617374600000,
        OCT20270747000000,OCT07011030414700O,OCT07204700000000,
        OCT07002440470000O,OCT00477007400000O,OCT07254725200000,
        OCT07472414342400 4,OCT00000000000000,OCT10212212112100O,
        OCT00477016060717 1,OCT67041313040410O,OCT02463545053513O,
        OCT34300000000000,OCT02427004440000O,OCT10303717000000,
        OCT14167036340000O,OCT00404404000424 2,OCT20000000000000,
        OCT00442204402200O,OCT00400044044422O,OCT00400444002200O,
        OCT24344341301001O,OCT31424220000000,OCT24422002242200O,
        OCT20224422042200O,OCT22242200000000,OCT24014124220000O,
        OCT24024224202200O,OCT44331304131100 1,OCT13140313322000O,
        OCT10365004641070 2,OCT22200000000000,OCT02422220242200O,
        OCT00440440220000O,OCT00442224202204 4,OCT02202422200000 ;
         IF NOT FIXED  THEN
         FILL B[*] WITH OCT30157,OCT12156,OCT14155,OCT22153,
        OCT32151,OCT14150,OCT12147,OCT06146,OCT14145,OCT14144,
        OCT26142,OCT14141,OCT16140,OCT14137,OCT20135,OCT22000,
        OCT12002,OCT22003,OCT32005,OCT20007,OCT22011,OCT30013,
        OCT12015,OCT40016,OCT30021,OCT34023,OCT22025,OCT32030,
        OCT26032,OCT06034,OCT14035,OCT12036,OCT24037,OCT30041,
        OCT24043,OCT16045,OCT16046,OCT14047,OCT30050,OCT14052,
        OCT14053,OCT12054,OCT10055,OCT32056,OCT10060,OCT06061,
        OCT12062,OCT12063,OCT12064,OCT14065,OCT06066,OCT12067,
        OCT10070,OCT34071,OCT16073,OCT30074,OCT24076,OCT26100,
        OCT26102,OCT04104,OCT10105, OCT30106,OCT14110,OCT00111,
        OCT04112,OCT30113,OCT10115,OCT14116,OCT06117,OCT12120,
        OCT12121,OCT12122,OCT20123,OCT14125,OCT34126,OCT22130,
        OCT12132,OCT10133,OCT12134 ;
         FIXED ← TRUE;
        XA ← (HGT/7) × COS(0.01745330754 × THETA)  ;
        YA ← (HGT/7) × SIN(0.01745330754 × THETA)  ;
        IF N≥0 THEN
BEGIN
        X ← X0 ;  Y ← Y0
END  ELSE
BEGIN
         IF N < -99 THEN
 BEGIN
         BINX ← -(N+100);
         X← X0 - 3×XA + 3.5×YA;    Y ← Y0 - 3×XA - 3.5×YA;
 END    ELSE
 BEGIN
        XA ← 7 ×XA / 4;  YA ← 7 × YA / 4;   BINX ← N;
        X ← X0 - 2×XA + 2×YA ;    Y ← Y0 - 2×XA - 2×YA;
 END;       PLOT (X0,Y0,3);      GO TO LOADB;
 END;
        FOR AC ← 1 STEP 1 UNTIL N DO
BEGIN
```

```
                IF AC MOD 6 =1 THEN W← BCD[(AC-1) DIV 6].[12:36];
                BINX ← W.[12:6];      W.[12:30] ← W.[18:30];
LOADB:          LP ← TRUE;          M7 ← FALSE;
                OSC ← B[BINX].[33:6];     AINX ← B[BINX].[39:9] - 1;
                FOR I ← 1 STEP 1 UNTIL OSC  DO
            BEGIN
                IF I MOD 15 = 1 THEN OSTS ← A[AINX ← AINX+1];
                MOVE ← OSTS.[3:3];  OSTS.[3:42] ← OSTS.[6:42];
                IF NOT BOOLEAN(I) THEN IF M7 THEN GO TO EL ELSE GO TO Y1;
                M7 ← MOVE = 7;     LP ← LP OR M7;
               XN ← XA × MOVE;  YN ← YA × MOVE;  GO TO EL;
Y1:            XN ← XN - YA × MOVE + X;  YN ← YN + XA × MOVE + Y;
                 PLOT(XN, YN, 2 + REAL (LP));  LP ← FALSE;
EL:        END  I LOOP;
               IF N < 0 THEN
           BEGIN
               PLOT (X0, Y0, 3);  GO TO EXIT
           END;
               X ← X +6×XA ;   Y ← Y + 6×YA
           END ;
EXIT:      END  OF SYMBOL;

PROCEDURE       CONVERT(X, N, ALF1, ALF2);
VALUE           X, N;                INTEGER   N;
ALPHA           ALF1, ALF2;          REAL      X;
            BEGIN
INTEGER         A1, A2, INT, DF;
ALPHA STREAM PROCEDURE   ALF (P);
            BEGIN
                DI ← LOC ALF;        SI ← P;
                DS ← 2 LIT "00";     DS ← 6 DEC
            END;
                X ← X + 0.5 × SIGN(X) / 10 * N;
                A1 ← IF X ≥ 0 THEN "     0"   ELSE  "    -0";
                IF (INT ← ENTIER(ABS(X))) > 99999  THEN
            BEGIN
                ALF1 ← "ILLEGA";     ALF2 ← "L NO. "
            END       ELSE
            BEGIN
                DF ← A2 ← ALF(INT);
                FOR DF ← DF.[12:30] WHILE DF>0 DO A1.[12:30]←A1.[18:30];
                DF ← ENTIER ((ABS (X) - INT) × @5);
                ALF2 ← ".00000" + ALF (DF);   ALF1 ← A2 + A1
            END
            END  OF CONVERT;

PROCEDURE       NUMBER (X, Y, HGT, FLT, THETA, N);
VALUE           X, Y, HGT, FLT, THETA, N;     INTEGER   N;
REAL            X, Y, HGT, FLT, THETA;
            BEGIN
DEFINE          BCD = PLOTTERBCDEE#;
                CONVERT (FLT, N, BCD[0], BCD[1]);  IF N < 1 THEN N ← -1;
                IF BCD[0] = "ILLEGA"  THEN  N ← 4;
                SYMBOL (X, Y, HGT, BCD, THETA, N+7)
            END  OF NUMBER;
```

```
PROCEDURE       AXIS(X,Y,BCD,NC,SIZE,THETA,YMIN,DY) ;
VALUE           X,Y,NC,SIZE,THETA,YMIN,DY ;
REAL            X, Y, SIZE, THETA, YMIN, DY ;  INTEGER NC ;
ALPHA ARRAY     BCD[0] ;
          BEGIN
REAL            SGN,TH,CTH,STH,XB,YB,XA,YA,XC,YC,CHAR,ABSV,EXPP,ADY,TNC  ;
REAL             XD, YD, DD;           BOOLEAN   FINE, FLIP;
INTEGER         N, I, NT, NAC ;        ALPHA ARRAY ABCD[0:1] ;
LABEL            L90, L91, L92, L50;
                SGN ← IF NC = 0 THEN 1 ELSE SIGN(NC);
                 FINE ← SIZE < 0;   SIZE ← ABS(SIZE); NAC ← ABS(NC);
                 IF FLIP ← BCD[0] < 0 THEN SGN ← -SGN;
                TH ← THETA × 0.017455  ;   N ← SIZE  ;   CTH ← COS(TH)  ;
                STH ← SIN(TH)  ;   XB ← X  ;   YB ← Y  ;
                XA ← X - 0.1 × SGN × STH  ;   YA ← Y + 0.1 × SGN × CTH  ;
                 I ← IF ABS(DY) <10 THEN ABS(DY)×@9 ELSE ABS(DY);
                 FOR I ← I/10 WHILE I >10 DO;
                 DD ← IF I=8 OR I=4 THEN  4 ELSE  5;
                PLOT(XA, YA, 3) ;
                FOR I ← 1 STEP 1 UNTIL N DO
          BEGIN
                PLOT(XB, YB, 2) ;   XC ← XB + CTH ;   YC ← YB + STH ;
                 IF FINE THEN FOR NT ← 2 STEP 1 UNTIL DD   DO
           BEGIN
                 XB ← XB + CTH/DD;   YB ← YB + STH/DD;   PLOT(XB,YB,2);
                 XD ← XB - SGN × STH / 20;     YD ← YB + SGN × CTH / 20;
                 PLOT (XD, YD, 2);   PLOT (XB, YB, 2)
           END;
                PLOT(XC, YC, 2) ;   XA ← XA + CTH ;   YA ← YA + STH;
                PLOT(XA, YA, 2) ;   XB ← XC ;   YB ← YC
          END ;
                IF NC=0 THEN GO TO L50;
                 ABSV ← ABS(DY);   EXPP ← 0;
                 IF ABSV < 100.1 AND ABSV > 0.00999
                 AND ABS(YMIN) < 10000  THEN GO TO L92;
L90:             IF ABS(YMIN) < 10 * (EXPP+3) AND ABSV < 0.9999 THEN
           BEGIN
                 ABSV ← ABSV × 10;   EXPP ← EXPP - 1; GO TO L90
           END;
L91:             IF ABSV > 10.0001 THEN
           BEGIN
                 ABSV ← ABSV / 10;   EXPP ← EXPP + 1; GO TO L91
           END;
L92:             ADY ← DY × 10 * (-EXPP);
                ABSV ← YMIN × 10*(-EXPP) + N × ADY  ;
                 XC ← (IF FLIP THEN -SGN/10 ELSE SGN/5) - 0.05;
                XA • XB - XC × STH - 0.53 × CTH  ;
                YA ← YB + XC × CTH - 0.53 × STH  ;   N ← N + 1  ;
                FOR I • 1 STEP 1 UNTIL N DO
          BEGIN
                NUMBER(XA, YA, .1, ABSV, THETA, 3) ;   ABSV ← ABSV - ADY ;
                XA ← XA - CTH ;   YA ← YA - STH
          END ;
                TNC ← NAC + 7 ;
                XC ← SIZE × 0.5 - 0.06 × TNC  ;
                 YC ← (IF FLIP THEN SGN × 0.3 ELSE -SGN × 0.4) + 0.07;
```

```
                XA ← X + XC × CTH + YC × STH  ;
                YA ← Y + XC × STH - YC × CTH  ;
                SYMBOL(XA, YA, .14, BCD, THETA, NAC) ;
                IF EXPP = 0 THEN GO TO L50  ;
                XC ● (TNC - 6) × 0.12  ;
                XA ← XA + XC × CTH  ;
                YA ← YA + XC × STH  ;
                ABCD[0] ← "(10  )"  ;
                SYMBOL(XA, YA, .14, ABCD, THETA, 6) ;
                XA ← XA + 0.18 × CTH - 0.07 × STH  ;
                YA ← YA + 0.18 × STH + 0.07 × CTH  ;
                NUMBER (XA, YA, 0.07,-EXPP,THETA, 0);
                XA ← XA - 0.24 × CTH;
                YA ← YA - 0.24 × STH;
                SYMBOL (XA, YA, 0.08, ABCD, THETA+45, -13);
L50:       END OF AXIS;

PROCEDURE       LYNE(X, Y, N, K) ;
VALUE           N, K ;
INTEGER         N,K  ;
ARRAY           X,Y[0]  ;
           BEGIN
INTEGER          I, I3, NF, NL;
REAL             PX, PY;
                 I3 ← 3;  NF ← NL ← (N-1)×K + 1;
                 PLOT (PX,PY,0);
                 IF (PX-X[1])*2 + (PY-Y[1])*2 < (PX-X[NL])*2 +(PY-Y[NL])*2
                 THEN  NF ← 1  ELSE
            BEGIN
                 NL ← 1;  K ← -K
            END;
                 FOR I ← NF STEP K UNTIL NL DO
            BEGIN
                 PLOT (X[I], Y[I], I3);  I3 ← 2
           END
           END OF LYNE;
PROCEDURE       SCALFS (X,N,XMIN,DX,K)  ;
VALUE           N,XMIN,DX,K  ;
REAL            XMIN,DX  ;
INTEGER         N,K  ;
REAL ARRAY      X[0]  ;
           BEGIN
INTEGER         I,NP  ;
                NP ← N×K  ;
                FOR I ← 1 STEP K UNTIL NP DO
                X[I] ← (X[I] -XMIN)/DX  ;
           END OF SCALES;

PROCEDURE       DXDY(YMAX,YMIN,TDY)  ;
VALUE YMAX;            REAL YMAX,YMIN,TDY  ;
           BEGIN
REAL            ADY,K1,V  ;
INTEGER         K  ;
LABEL            FIN, CHUZ;
DEFINE          BCD  ■  PLOTTERBCDEE  #  ;
                 V ← IF BCD[0] = TDY AND TDY ≠ 0 THEN TDY ELSE 1;
```

```
            ADY ← YMAX - YMIN   ;
            K1 ← 0;   TDY ← 1;   IF   ADY = 0   THEN
        BEGIN
            FILL BCD[*] WITH  "DATA E", "RROR: ",  "YMIN=Y", "MAX    ";
            SYMBOL (1, 2, 0.07, BCD, -90, 21);    GO TO FIN             ;
        END;
             IF YMIN ≠ 0  THEN
         BEGIN
             K ← ENTIER (2.0001 - LN(ADY/V) / LN(10));
             V  ←  1 + SIGN(YMIN) / @4;
             YMIN ← ENTIER (' × YMIN × 10 * K) / 10* K;
             ADY  ← YMAX - YMIN
         END;
             WHILE ADY < 10  DO
         BEGIN
             ADY ← 10 × ADY;      K1 ← K1 - 1
         END;
CHUZ:        FOR TDY ← 10, 15, 20, 25, 40, 50, 80  DO
             IF ADY < 1.001 × TDY THEN GO TO FIN;
             ADY ← ADY / 10;  K1 ← K1 + 1; GO  TO CHUZ;
FIN:         TDY ← TDY × 10 * K1
         END   OF DXDY;

PROCEDURE     SCALE(X,N,S,YMIN,DY,K)   ;
VALUE S,N,K   ;  INTEGER N,K   :   REAL S,YMIN,DY   ;
REAL ARRAY   X[0]  ;
        BEGIN
REAL        YMAX  ;
INTEGER     I,NP  ;
            NP ← N × K   ;
            YMAX ← X[1]  ;
            YMIN ← X[1]  ;
            FOR I ← 2 STEP K UNTIL NP DO
        BEGIN
            IF YMAX < X[I] THEN
            YMAX ← X[I]  ,
            IF X[I] < YMIN THEN
            YMIN ← X[I]  ;
        END ;
             IF S = 0  THEN S ← 1;
             PLOTTERBCDEE[0] ← DY ← 10 / S;
             DXDY (DY × (YMAX - YMIN) + YMIN, YMIN, DY);
            DY ← DY/10.0  ;
            FOR I ← 1 STEP K UNTIL NP DO
            X[I] ← (X[I] - YMIN)/DY  ;
        END OF SCALE;

PROCEDURE     DASHLINE(X,Y,N1,K)   ;
VALUE N1,K  ;  INTEGER N1,K   ;
REAL ARRAY X,Y[0]  ;
        BEGIN
INTEGER     I,NP,M,N  ;
REAL        PEN,XN,YN,ADX,ADY,DX,DY,DLTX,DLTY   ;
LABEL       L1  ;
            PEN ← 2  ;
            M ← 20  ;
```

```
                NP ← N1 × K  ;
                XN ← X[1]  ;  YN ← Y[1]  ;
                PLOT(XN,YN,3)  ;
                FOR I ← 2 STEP K UNTIL NP DO
          BEGIN
                DX ← X[I] - XN  ;
                DY ← Y[I] - YN  ;
                ADX ← ABS(DX)  ;  ADY ← ABS(DY)  ;
                IF ADX > ADY THEN
          BEGIN
                DLTX ← SIGN(DX) × 0.01  ;
                DLTY ← 0.01 × DY/ADX  ;
                N ← ADX × 100.0  ;
          END
                ELSE
          BEGIN
                DLTY ← SIGN(DY) × 0.01  ;
             DLTX ← 0.01 × DX/ADY  ;
                N ← ADY × 100.0
          END ;
L1:             IF M ≤ N THEN
          BEGIN
                PLOT(XN ← XN + M × DLTX,YN ← YN + M × DLTY,PEN)  ;
                N ← N - M  ;
                IF PEN = 3 THEN
          BEGIN
                M ← 20  ;
                PEN ← 2  ;
          END
                ELSE
          BEGIN
                M ← 10  ;
                 PEN ← 3;
          END ;
                 GO TO L1 ;
          END
                ELSE
          BEGIN
                PLOT(XN ← XN + N × DLTX,YN ← YN + N × DLTY,PEN)  ;
                M ← M - N  ;
          END ;
          END ;
                PLOT(XN,YN,2)  ;
          END OF DASHLINE;

PROCEDURE       NAMELINE(X,Y,N,K,A,T,DASH)  ;
VALUE N,K,T  ;  INTEGER N,K,T  ;
REAL ARRAY X,Y[0]  ;
ALPHA ARRAY A[0]  ;
BOOLEAN DASH  ;
          BEGIN
INTEGER         T1,N1,I,J,NP  ;
REAL            TH,XM,YM,MX,DX,DY,S,YL  ;
REAL ARRAY      X1,Y1[0:N DIV 2 + 2]  ;
                N1 ← N DIV 2  ;
                NP ← N × K  ;
```

```
      I ← (N1 - 1) × K + 1   ;
      IF DASH THEN
      DASHLINE(X,Y,I,K)
      ELSE
      LYNE(X,Y,I,K)   ;
      DX ← X[I + K] - X[I - K]   ;
      DY ← Y[I + K] - Y[I - K]   ;
      IF DX = 0 THEN
      TH ← 1.5707963
      ELSE
      TH ← ARCTAN(DY/DX)   ;
      IF T ≥ 0 THEN
BEGIN
      T1 ← T DIV 2   ;
       S ← SIGN (1@-6    + (DX × Y[I] - DY × X[I]
                   + X[I-K] × Y[I+K] - X[I+K] × Y[I-K]));
       S ← IF S = 0 THEN 0.14 ELSE 0.05 + S × 0.09;
      XM ← S × SIN(TH) - 0.0857 × T1 × COS(TH) + X[I]   ;
      YM ← -S × COS(TH)-0.0857 × T1 × SIN(TH) + Y[I]   ;
      TH ← 57.2959125 × TH   ;
      SYMBOL(XM,YM,0.10,A,TH,T)   ;
END
      ELSE
      SYMBOL(X[I],Y[I],0.10,A,TH,T)   ;
      J ← 0   ;
      FOR I ← I      STEP K UNTIL NP DO
BEGIN
      J ← J + 1   ;
      X1[J] ← X[I]   ;   Y1[J] ← Y[I]   ;
END ;
      IF DASH THEN
      DASHLINE(X1,Y1,J,K)
      ELSE
      LYNE(X1,Y1,J,K)   ;
 END OF NAMELINE;

                    THERE ARE    595 CARDS IN THE DECK

  E 10, 1968.  TOTAL ELAPSED TIME IS    63 SECONDS.

   PROCESSOR TIME IS    10 SECONDS.  I/O
```

```
REAL WW ,WO  , DW , WM , L , L1 , L2 , W , IP , IT , WM1 , WM2 , H1 ,
H2 , PH , R1 , R2 , K1X , K2X , K1Y , K2Y , C1X , C2X , C1Y , C2Y ,
 D1X , D2X , D1Y , D2Y , R1X , R2X , R1Y , R2Y , G , PI , M, DM1 , DM2 ,
 RPP , RTT , RP , RT , L11 , L22 , RO1 , RO2 , PHI , EPS , RAD ,
 K1XX , K2XX , K1YY , K2YY , C1XX , C2XX , C1YY , C2YY , R1XX , R2XX ,
 R1YY , R2YY , D1XX , D2XX , D1YY , D2YY , PN1 , PO1 , SI , PN2 , PD2 ,
 SIT , BB , CC , DD , EE , PX1 , PX2 , PY1 , PY2 , PA1 , PA2   ,
 PFX , PFY , E ;
INTEGER  P , N , I , K1 , J   ,  K  ;
REAL ARRAY    OMEGA , S , SS , XX1 , XX2 , YY1 , YY2 , SIX1 , SIX2 ,RPM,
 SIY1 , SIY2 , SIA1 , SIA2 , ALFA1 , ALFA2 , FX1 , FX2 , PFX2 , FY1 ,
 FY2 , PFY2 , PURFX1 , PURFX2 , PUBFY1 , PUBFY2[0:200] ,LZ[0:4]  ,
  XL , YL , PXL , PYL[0:200,0:4] ,A[0:8,0:8]  , C , X[0:8]  ,
PFX1 , PFY1[0:200]   ;
LABEL   LOU , FINIS , E1 , E2 , LOU1  ;
BOOLEAN  RSW   ;
REAL XMIN,DX,YMIN,DY  ; INTEGER WP,T,  B,  D,  F,QQ ,RN,GK  ;
ALPHA ARRAY ALPHA2,ALPHA4,ALPHA5,ALPHA6,ALPHA7,ALB11,ALB13,ALB21,ALB23,
ALB31,ALB33,ALB41,ALB43,ALB51,ALB53,ALB61,ALB63,ALB71,ALB73,ALB81,ALB83
[0:8];

PROCEDURE INTERP(X0,XF,H,Y,X,BETA,EPSILON,M,INW,LLABEL);VALUE X0,XF,H,X,
BETA,EPSILON,M;INTEGER M:REAL X0,XF,H,X,EPSILON,INW;BOOLEAN BETA;ARRAY Y
[0];LABEL LLABEL;BEGIN INTEGER N,I,J,K;REAL U1,U,TEMP;LABEL NFOR,L1,NBAK
,L2,EOP;IF(X<(X0-2×H))OR(X>(XF+2×H))THEN GO TO LLABEL;N←ABS((XF-X0)/H);I
F M>N THEN M←N;U1←(XF+X0)/2.0;K←1;IF X<(U1-5×H)THEN GO TO NFOR ELSE IF(X
>U1+5×H)OR BETA THEN GO TO NBAK;NFOR:J←M-1;U←(X-X0)/H;INW←Y[0]:U1←1.0;L1
:Y[0]←Y[1]-Y[0];IF ABS(Y[0])<EPSILON THEN GO TO EOP;U1←U1×(U-K+1.0)/K;IN
W←INW+U1×Y[0];IF K=M THEN GO TO EOP;FOR I←1STEP 1UNTIL J DO Y[I]←Y[I+1]-
Y[I];J←J-1:K←K+1:GO TO L1;NBAK:U←(X-XF)/H;J←N-M+1:INW←Y[N];U1←1.0;L2:Y[N
]←Y[N]-Y[N-1];IF ABS(Y[N])<EPSILON THEN GO TO EOP;U1←U1×(U+K-1.0)/K;INW←
INW+Y[N]×U1;IF K=M THEN GO TO EOP:FOR I←N-1STEP-1UNTIL J DO Y[I]←Y[I]-Y[
I-1];J←J+1:K←K+1;GO TO L2;EOP:END;

PROCEDURE GETTUN(RY,RX,RN,X,Y,N,GK);REAL ARRAY X,Y,RX,RY[0];INTEGER N,
RN,GK;  BEGIN  REAL INW,X1,W; INTEGER I,J,K ;  REAL ARRAY OY[0:10]:
LABEL LLABEL, EXU ;
K ← 2;  FOR J ← 1 STEP 5 UNTIL N-5 DO BEGIN  K ← K-1 ;
FOR I ← 0 STEP 1 UNTIL 5 DO OY[I] ← Y[J+I]:
W ← (X[J+5]-X[J])/5;
FOR X1 ← X[J] + W/GK STEP W/GK UNTIL X[J+5] DO BEGIN
```

```
INTERP(X[J],X[J+5], W .DY,X1,FALSE,@-5,5,INW,LLABEL);
RY[K] ← INW ; RX[K] ← X1; K ← K+1; END; END; RN ← K-1: GO TO EXO;
LLABEL : WRITE(LP,<"OUTSIDE RANGE">); EXO: END OF GETTUM ;

PROCEDURE ALINE (X,Y,N,YMIN,DY,K,Q,QQ); VALUE N; INTEGER N,K,Q,QQ;
REAL YMIN,DY; REAL ARRAY X,Y[0];
BEGIN INTEGER I,J,RN ; REAL ARRAY R,RX,RY[0:405];
ALPHA ARRAY BCD[0:2];
IF QQ = 1 THEN SCALES(Y,N,YMIN,DY,K) ELSE SCALE(Y,N,6,YMIN,DY,K);
IF GK ≠ 1 THEN BEGIN
FOR I ← 1 STEP 1 UNTIL N DO R[I] ← X[I] ; GETTUM(RY,RX ,RN,R,Y,N,GK);
IF QQ = 1 THEN FOR I ← 1 STEP 1 UNTIL RN DO BEGIN IF RY[I] > 6 THEN RY[I
] ← 6 ELSE IF RY[I] < 0 THEN RY[I] ← 0 ; END;LYNE(RX,RY,RN,K);
FOR I ← 1 STEP 1 UNTIL 5 DO BEGIN J ← ENTIER(I×(RN/5)); IF RY[J] < 5.8
AND RY[J] > .2 THEN SYMBOL(RX[J],RY[J],.12,BCD,0,Q); END;END ELSE BEGIN
IF QQ = 1 THEN FOR I ← 1 STEP 1 UNTIL N DO BEGIN IF Y[I] > 6 THEN Y[I
] ← 6 ELSE IF Y[I] < 0 THEN Y[I] ← 0 ; END;LYNE( X, Y, N,K);
FOR I ← 1 STEP 1 UNTIL 5 DO BEGIN J ← ENTIER(I×( N/5)); IF Y[J] < 5.8
AND Y[J] > .2 THEN SYMBOL(X[J],Y[J],.12,BCD,0,Q); END;
END;
END OF ALINE ;

PROCEDURE AGRID(NY,NT,F1,F2,F3,F4,ALBN1,ALBN3);
INTEGER NY,NT ;
REAL F1,F2,F3,F4 ;
ALPHA ARRAY ALBN1,ALBN3[0];
BEGIN REAL XOT;
AXIS(0,0,ALBN1,NY,-6,90,YMIN,DY); AXIS(0,0,ALPHA2,-15,-8,0,XMIN,DX);
AXIS(8,0,ALPHA2,0,-6,90,YMIN,DY); AXIS(8,6,ALPHA2,0,-8,180,XMIN,DX);
PLOT(0,6,2); PLOT(0,6.5,1); PLOT(8,6.5,1); PLOT(8,6,1); PLOT(8,6.25,3);
PLOT(0,6.25,2);PLOT(.5,6.05,3);
SYMBOL(.5,6.05,.14,ALPHA4,0,10 ); NUMBER(1.46,6.05,.14,F1,0,0);
XOT ← 4-(NT×.12)/2 ; SYMBOL(XOT,6.30,.14,ALBN3,0,NT);
IF B = 7 THEN BEGIN
SYMBOL(4,6.05,.14,ALPHA7,0,25); NUMBER(6.64,6.05,.14,F4,0,2 ); END;
END OF AGRID ;

PROCEDURE GOPLOTTER (WW): REAL ARRAY WW [0]:
BEGIN LABEL GETOUT, GOAGIN,DRAW1,DRAW2,DRAW3,DRAW4,DRAW5,DRAW6,
DRAW7 : INTEGER A,B,C ;
READ(CR,/,WP);IF WP = 0 THEN GO TO GETOUT; PLOT(2,0,-4);PLOT(2,1.5,-5)
; FILL ALPHA2[*] WITH "FREQUE","NCY (R","PM) "; FILL ALPHA4[*] WITH
"CASE N","O. = " ;
READ (CR,/,A,GK);
SCALE (WW,I,8,XMIN,DX,1): FOR T ← 1 STEP 1 UNTIL WP DO BEGIN
READ (CR,/,B,C,D,E,F,YMIN,DY,QQ);
IF B = 1 THEN GO TO DRAW1 ELSE IF B = 2 THEN GO TO DRAW2 ELSE
IF B = 3 THEN GO TO DRAW3 ELSE IF B = 4 THEN GO TO DRAW4 ELSE
IF B = 5 THEN GO TO DRAW5 ELSE IF B = 6 THEN GO TO DRAW6 ELSE
IF B = 7 THEN GO TO DRAW7 ;
DRAW1 : BEGIN
                    FILL ALB11[*] WITH "BEARIN","G AMPL","ITUDE ";
                    FILL ALB13[*] WITH "BEARIN","G AMPL","ITUDE ",
" VS. ","FREQUE","NCY ";
IF C = 1 THEN BEGIN
 ALINE(WW,XX1, I, YMIN,DY,1,-9 ,QQ); QQ ← 1 : END;
```

```
IF D = 1 THEN BEGIN
 ALINE(WW,YY1, I, YMIN,DY,1,-11,QQ); QQ ← 1 ; END;
IF E = 1 THEN BEGIN
 ALINE(WW,XX2, I, YMIN,DY,1,-13,QQ); QQ ← 1 ; END;
IF F = 1 THEN ALINE(WW,YY2, I,YMIN,DY,1,-15,QQ);
AGRID(17,33,A,0,0,0,ALB11,ALB13); PLOT(10,0,-4); PLOT(2,0,-5); END;
GO TO GOAGIN ;
DRAW2 : BEGIN
FILL ALB21[*] WITH "BEARIN","G PHAS","E ANGL","E     ";
FILL ALB23[*] WITH "BEARIN","G PHAS","E ANGL","E VS F","REQUEN","CY      "
;
IF C = 1 THEN BEGIN
ALINE(WW,SIX1,I,YMIN,DY,1,-9 ,QQ);QQ ← 1; END;
IF D = 1 THEN BEGIN
ALINE(WW,SIY1,I,YMIN,DY,1,-11,QQ);  QQ ← 1 ;  END;
 IF E = 1 THEN BEGIN
ALINE(WW,SIX2,I,YMIN,DY,1,-13,QQ);  QQ ← 1 ; END;
IF F = 1 THEN ALINE(WW,SIY2, I,YMIN,DY,1,-15,QQ);
AGRID(19,32,A,0,0,0,ALB21,ALB23);  PLOT(10,0,-4); PLOT(2,0,-5); END;
GO TO GOAGIN ;
DRAW3 : BEGIN
FILL ALB31[*] WITH "ANGULA","R AMPL","ITUDE ";
FILL ALB33[*] WITH "ANGULA","R AMPL","ITUDE ","VS. FR","EQUENC","Y       "
;
IF C = 1 THEN BEGIN
 ALINE(WW,ALFA1, I, YMIN,DY,1,-9 ,QQ);  QQ ← 1 ;  END;
IF D = 1 THEN ALINE(WW,ALFA2, I,YMIN,DY,1,-11,QQ);
AGRID(17,31,A,0,0,0,ALB31,ALB33); PLOT(10,0,-4); PLOT(2,0,-5); END;
GO TO GOAGIN ;    DRAW4 : BEGIN
FILL ALB41[*] WITH "ANG. P","HASE A","NGLE  ";
FILL ALB43[*] WITH "ANG. P","HASE A","NGLE V","S. FRE","QUENCY";
IF C = 1 THEN BEGIN
 ALINE(WW,SIA1 , I, YMIN,DY,1,-9 ,QQ);  QQ ← 1 ;  END;
IF D = 1 THEN ALINE(WW,SIA2, I,YMIN,DY,1,-11,QQ);
AGRID(16,30,A,0,0,0,ALB41,ALB43); PLOT(10,0,-4); PLOT(2,0,-5); END;
GO TO GOAGIN ;
DRAW5 : BEGIN
FILL ALB51[*] WITH "FORCE ","TRANS.";
FILL ALB53[*] WITH "FORCE ","TRANS."," VS. F","REQUEN","CY    ";
IF C = 1 THEN BEGIN
 ALINE(WW,FX1,I,YMIN,DY,1,-9 ,QQ); QQ ← 1 ; END;
IF D = 1 THEN BEGIN
 ALINE(WW,FY1,I,YMIN,DY,1,-11,QQ); QQ ← 1 ; END;
IF E = 1 THEN BEGIN
 ALINE(WW,FX2,I,YMIN,DY,1,-13,QQ); QQ ← 1 ; END;
IF F = 1 THEN ALINE(WW,FY2, I,YMIN,DY,1,-15,QQ);
AGRID(12,26,A,0,0,0,ALB51,ALB53); PLOT(10,0,-4); PLOT(2,0,-5); END;
GO TO GOAGIN ;
DRAW6 : BEGIN
FILL ALB61[*] WITH "FORCE ","TRANS."," PHASE"," ANGLE";
FILL ALB63[*] WITH "FORCE ","TRANS."," PHASE"," ANGLE"," VS. F",
"REQUEN","CY     " ;
IF C = 1 THEN BEGIN
 ALINE( WW,PUBFX1,I,YMIN,DY,1,-9 ,QQ);  QQ ← 1 ;  END;
IF D = 1 THEN BEGIN
 ALINE( WW,PUBFY1,I,YMIN,DY,1,-11,QQ);  QQ ← 1 ;  END;
```

```
IF E = 1 THEN BEGIN
 ALINE( WW,PUBFX2,I,YMIN,DY,1,-13,QQ);   QQ ← 1 ;   END;
IF F = 1 THEN ALINE(WW,PUBFY2, I,YMIN,DY,1,-15,QQ);
AGRID(24,33,A,0,0,0,ALB61,ALB63); PLOT(10,0,-4); PLOT(2,0,-5); END;
GO TO GOAGIN ;   DRAW7 :   BEGIN REAL DYRE; DYRE ← DY ;
FILL ALPHA7[*] WITH "LENGTH"," FROM ","1ST BE","ARING ","=      ";
IF C = 1 THEN BEGIN  REAL ARRAY  NX,NY[0:100] ;
FILL ALB71[*] WITH "AMPLIT","UDE    ";
FILL ALB73[*] WITH "AMPLIT","UDE VS",". FREQ","UENCY ";
FOR E ← 1 STEP 1 UNTIL P DO BEGIN
FOR F ← 1 STEP 1 UNTIL I DO BEGIN NX[F] ← XL[F,E]; NY[F] ← YL[F,E];END;
ALINE(WW,NX, I,YMIN,DY,1,-9 ,QQ);
QQ ← 1;
ALINE(WW,NY, I,YMIN,DY,1,-11,QQ);
AGRID(9,23,A,0,0,LZ[E],ALB71,ALB73); IF DYRE ≠ DY THEN QQ ← 0 ;
PLOT(12,0,-5); END; PLOT(1,0,-4);   GO TO GOAGIN ;   END;
IF D = 1 THEN BEGIN  REAL ARRAY NPX,NPY[0:100];REAL DYRE ; DYRE ← DY ;
FILL ALB81[*] WITH "PHASE ","ANGLE ";
FILL ALB83[*] WITH "PHASE ","ANGLE ","VS. FR","EQUENC","Y      ";
FOR E ← 1 STEP 1 UNTIL P DO BEGIN
FOR F ← 1 STEP 1 UNTIL I DO BEGIN NPX[F] ← PXL[F,E] × RAD;
NPY[F] ← PYL[F,E] × RAD ; END;
ALINE(WW,NPX, I,YMIN,DY,1,-9 ,QQ);
QQ ← 1;
ALINE(WW,NPY, I,YMIN,DY,1,-11,QQ);
AGRID(11,25,A,0,0,LZ[E],ALB81,ALB83);IF DYRE ≠ DY THEN QQ ← 0 ;
PLOT(12,0,-5); END;      PLOT(1,0,-4); END; END;
GOAGIN :
END;
GETOUT:   END OF GOPLOTTER;
     COMMENT   THIS  PROGRAM EVALUATES DESIGN DATA FOR A FOUR DEGREE
 FREEDOM SYSTEM THAT SIMULATES A ROTOR ON GENERAL ANISOTROPIC BRGS.
     THE EQUATIONS SOLVED HAD BEEN LINEARIZED . NO ASSUMPTIONS WERE
MADE ON THE BEARING CHARACTERISTICS . THE CROSS COUPLING TERMS  ARE
KEPT WITH PROPER SUBSCRIPTS AS USED IN THE DERIVATION OF THE EQNS.
     THE  PROGRAM REQUIRES THE FOLLOWING TO BE READ AS INPUT DATA:
CARD 1
     1. WO- INITIAL SPEED (RPS)
     2. DW- INCREMENT IN SPEED (RPS)
     3.WM- FINAL SPEED (RPS)
CARD 2
     1. L- LENGTH BETN BRGS (INCH)
     2.L1- DIST FROM 1ST BRG TO MASS CENTER (INCH)
     3.L2- DIST FROM 2ND BRG TO MASS CENTER (INCH)
     4.W- ROTOR WEIGHT (LBS)
     5. IP- POLAR M.I. (LB-IN-SEC2)
     6. IT-TRANSVERSE M.I.  OF ROTOR ABOUT MASS CENTER (LB-IN-SEC2)
CARD 3
     1. WM1-FIRST UNBALANCE WEIGHT (LBS)
     2. WM2- SECOND UNBALANCE WEIGHT (LBS)
     3. H1- DIST FROM 1ST BRG TO 1ST UNBALANCE (INCH)
     4. H2- DIST FROM 1ST BRG TO 2ND UNBALANCE (INCH)
     5. PHI- PHASE ANGLES BETN UNBALANCE PLANES
     6. R1- RADIUS OF 1ST UNBALANCE LOCATION
     7. R2- RADIUS OF 2ND UNBALANCE LOCATION
CARD 4
```

```
     1. N- NO. OF PLACES OTHER THAN THE BRG LOCATIONS WHERE
DISPLACEMENTS ARE TO BE MEASURED
     2.LZ1- DIST FROM 1ST BRG TO 1ST PROBE (INCH)
     3. LZ2- DIST FROM 1ST BRG TO 2ND PROBE  (INCH)
CARD 5
     1. K1X- 1ST BRG STIFFNESS IN X DIRECTION (LB/IN)
     2. K2X- 2ND BRG STIFFNESS IN X DIRECTION (LB/IN)
     3. K1Y- 1ST BRG STIFFNESS IN Y DIRECTION (LB/IN)
     4. K2Y- 2ND BRG STIFFNESS IN Y DIRECTION(LB/IN)
CARD 6
     1. C1X-1ST BRG DAMPING COEFF IN X DIRECTION(LB.SEC/IN)
     2. C2X- 2ND BRG DAMPING COEFF IN X DIRECTION (LB.SEC/IN)
     3.C1Y-1ST BRG DAMPING COEFF IN Y DIRECTION (LB.SEC/IN)
     4. C2Y- 2ND BRG DAMPING COEFF IN Y DIRECTION (LB.SEC/IN)
CARD 7
     1. D1X- CROSS COUPLING DAMPING COEFF  (LB.SEC/IN)
     2. D2X- CROSS COUPLING DAMPING COEFF  (LB.SEC/IN)
     3. D1Y- CROSS COUPLING DAMPING COEFF  (LB.SEC/IN)
     4. D2Y- CROSS COUPLING DAMPING COEFF  (LB.SEC/IN)
CARD 8
     1. R1X- CROSS COUPLING STIFFNESS (LB/IN)
     2. R2X- CROSS COUPLING STIFFNESS (LB/IN)
     3. R1Y- CROSS COUPLING STIFFNESS (LB/IN)
     4. R2Y- CROSS COUPLING STIFFNESS (LB/IN)
    THE OUTPUT DATA ARE AS FOLLOWS :
COL 1     WW-SPEED (RPS)
COL2      X1 OR Y1 - DISP AT BRG1 IN X OR Y DIRECTION
COL3      X2 OR Y2- DISP AT BRG2 IN X OR Y DIRECTION
COL4      SIX1 OR SIY1-PHASE ANGLE OF X1 OR Y1 WRT UNBALANCE
COL5      SIX2 OR SIY2-PHASE ANGLE OF X2 OR Y2 WRT UNBALANCE
COL6      ALFA1 OR ALFA2-ANGULAR DISPLACEMENTS
COL 7     SIA1 OR SIA2- PHASE ANGLE OF ALFA1 OR ALFA2
COL 8     FX1 OR FY1-FORCE TRANSMITTED TO BRG 1 IN X OR Y DIRECTION
COL 9     FX2 OR FY2- FORCE TRANSMITTED TO BRG2 IN X OR Y DIRECTION
COL 10    PUBFX1 OR PUBFY1- PHASE ANGLE OF FX1 OR FY1
COL 11    PUBFX2 OR PUBFY2- PHASE ANGLE OF FX2 OR FY2
     THE HEADINC PRINT OUT OF THE INPUT DATA ARE AS FOLLOWS :
LINE1     L,L1,L2,H1
LINE2     H2,W,WM1,WM2
LINE3     K1X,K2X,K1Y,K2Y
LINE4     C1X,C2X,C1Y,C2Y
LINE5     R1X,R2X,R1Y,R2Y
LINE6     D1X,D2X,D1Y,D2Y
LINE7     IP,IT,R1,R2
LINE8     PHI      ;
FORMAT  HEAD1 (6(2(59("*")),/),
                              24("*"), X40 , X31 ,23("*"),/,
24("*"), X1 , "DESIGN DATA FOR A SINGLE MASS ROTOR WITH FLEXIBLE SUPPORT
 AND DAMPING" ,X1 , 23 ("*"),/,  6(2(59("*")), /))  ;
FORMAT  HEAD2(2(2(59("*")),/),
X5 , "L=",E11.4 , "INCH" , X12 , "L1=" , E11.4 ,"INCH" , X12 , "L2=" ,
 E11.4 , "INCH" , X12 , "H1=" , E11.4 , "INCH" , / ,
X4 , "H2=" , E11.4 , "INCH" , X13 , "W=" , E11.4 , "LB" , X13 , "WM1=" ,
 E11.4 , "LB" , X13 , "WM2=" , E11.4 ,"LB" , / ,
 X3 , "K1X=" , E11.4 , "LB/IN" , X10 , "K2X=" , E11.4 ,"LB/IN" , X10 ,
 "K1Y=" , E11.4 , "LB/IN" , X10 , "K2Y=" , E11.4 , "LB/IN" , / ,
```

```
X3 , "C1X=" , E11.4 , "LB.SEC/IN" , X6 , "C2X=" , E11.4 , "LB.SEC/IN" ,
X6 , "C1Y=" , E11.4 , "LB.SEC/IN" ,X6 , "C2Y=" , E11.4 ,"LB.SEC/IN" ,/,
X3 ,"R1X=" , E11.4 , "LB/IN" , X10 , "R2X=" , E11.4 , "LB/IN" , X10 ,
"R1Y=" , E11.4 , "LB/IN" , X10 ,"R2Y=" , E11.4 , "LB/IN" , / ,
X3 , "D1X=" , E11.4 , "LB.SEC/IN" , X6 , "D2X=" , E11.4 , "LB.SEC/IN" ,
X6 , "D1Y=" , E11.4 , "LB.SEC/IN" , X6 , "D2Y=" , E11.4 , "LB.SEC/IN",/
,X4,"IP=" , E11.4 , "LB-IN-SEC2" , X6 , "IT=" , E11.4 , "LB-IN-SEC2" ,
X6 , "R1=" , E11.4 , "INCH" , X12 , "R2=" , E11.4,"INCH" ,/ ,
X30 , "PHI=" , E11.4 , "DEGREES" , / ,
                                          2(2(59("*")),/))    ;
FORMAT OUT1 (X6 , "SPEED" , X6 , "X1" , X9 , "X2" , X9 , "SIX1" , X3 ,
"SIX2" , X3 ,
"ALFA1" , X7 , "SIA1" , X4 , "FX1" , X9 , "FX2" , X6 ,"PUBFX1" , X2 ,
"PUBFX2" , // )  ;
FORMAT OUT2( X6 , I6 , X1 , E11.4 , X1 , E11.4 , X1 , F6.1 ,
X1 , F6.1 , X1 , E11.4 , X1 , F6.1 , X1 , E11.4 , X1 , E11.4 , X1 ,
F6.1 , X1 , F6.1 )  ;
FORMAT OUT3 (X6 , "SPEED" , X6 , "Y1" , X9 , "Y2" , X9 ,"SIY1" , X3 ,
"SIY2" , X3 ,  "ALFA2" , X7 , "SIA2" , X4 , "FY1" , X9 , "FY2" , X6 ,
"PUBFY1",X2 , "PUBFY2",// ) ;
FORMAT OUT4( X8 , "LZ" , X20 , "XL" , X18 , "YL" , X16 , "PXL" , X18 ,
"PYL" , X9 , "SPEED" ,// )  ;
FORMAT  OUT5 ( X6 , F6.1 , X13 , F11.4 , X9 , E11.4 , X10 , F7.2 , X14 ,
F7.2 , X6 , F7.2 )  ;
REAL  PROCEDURE  ANGLE(PN,PD)  ;
VALUE  PN , PD ;        REAL  PN, PD ;
BEGIN
REAL  B , PI  ;
LABEL  L1 , L2 , L3 , L4   ;
PI ← 3.14159  ;
IF  PN> 0 AND  PD=0  THEN  GO  TO L1  ;
IF  PN<0 AND  PD=0  THEN  GO  TO L2  ;
IF  PN=0  AND  PD=0  THEN  GO  TO  L3 ;
B ← ARCTAN(ABS(PN/PD)) ;
IF  PN<0  AND  PD>0  THEN  B ← 2× PI - B  ;
IF  PN >0  AND PD<0  THEN B← PI - B  ;
IF PN<0  AND  PD<0  THEN  B← PI+B   ;
GO  TO  L4  ;
L1:  B←PI/2  ;
GO  TO  L4  ;
L2:   B← (3×PI) /2  ;
 GO  TO  L4  ;
L3:  B←0  ;
L4:   ANGLE←B  ;
END  OF  PROCEDURE  ;
PROCEDURE  FORCE(C , K,D,R,C1,S1,C2,S2,WW,F,PFX,PFY)   ;
VALUE  C , K , D , R , C1, S1 , C2 , S2 , WW ;
REAL   C , K , D , R ,   C1 , S1 , C2 , S2 , WW , F , PFX , PFY   ;
COMMENT  THIS  PROCEDURE  CALCULATES  THE  FORCE OR  MOMENT
 PRODUCED  BY  THE  REACTIONS  WHERE
C= DAMPING COEFF    D= CROSS COUPLING  DAMPING
K= STIFFNESS  COEFF  R= CROSS  COUPLING  STIFFNESS
THE  FORCE  CALCULATED  IS  IN THE  DIRECTION  OF  X1  WHERE
X1= C1 COS(WWT) + S1 SIN(WWT) ,WHERE  WW=ROTOR  SPEED  IN  RAD/SEC
DIRECTION  NORMAL  TO  X1  IS  X2  WHERE
X2= C2 COS (WWT) + S2 SIN(WWT)
```

```
F=F COS(WWT-PH)=A COS(WWT) + B SIN(WWT)  ;
BEGIN
REAL  A, B  ;
A ← C × WW × S1 + K × C1 + D × WW × S2 + R× C2  ;
B ←-WW × C × C1 + K × S1 - WW × D × C2  + R × S2  ;
F ← SQRT ( A× A + B× B )  ;
PFX ← ANGLE ( B , A) ;    PFY ← ANGLE (-A , B )  ;
END  OF  PROCEDURE  FORCE  ;
PROCEDURE  ARBITRARYDISPLACEMENT (LZ , L , X , XL , YL , PXL , PYL )  ;
VALUE  LZ , L  ;
REAL  LZ , L , XL , YL , PXL , PYL  ;
REAL  ARRAY  X[O]  ;
BEGIN
COMMENT  THIS  PROCEDURE  CALCULATES  THE  X AND  Y  DISPLACEMENTS AT
ANY  POINT  MEASURED  FROM  THE  FIRST  BRG .  XL  IS  SHAFT ABSOLUTE
X DISPLACEMENT AND  PXL IS THE PHASE  ANGLE  ;
REAL  AX , BX , AY , BY , Z  ;
Z ← LZ/L  ;
AX ← Z× X[3] + (1 - Z ) × X[1]  ;
BX ← Z× X[4] + (1 - Z ) × X[2] ;
 AY ← Z × X[7] + ( 1 - Z ) × X[5] ;
 BY ← Z × X[8] + ( 1 - Z ) × X[6]  ;
 XL ← SQRT ( AX× AX + BX× BX ) ;
YL ← SQRT ( AY × AY + BY × BY )  ;
PXL ← ANGLE (BX , AX ) ;
PYL ← ANGLE ( - AY , BY )   ;
END  OF  PROCEDURE  ARBITRARYDISPLACEMENT  ;
PROCEDURE SOLVE(N,A,C,RSW,E,K1,EPS,X,E1,E2);VALUE N,RSW,E,K1,EPS;INTEGER
 N,K1;REAL E,EPS;BOOLEAN RSW;REAL ARRAY A[0,0],C,X[0];LABEL E1,E2;BEGIN
INTEGER I,J,K,J1,K2,I,REAL BIG,TEMP,DIAG,NORM,Q;OWN INTEGER ARRAY F[0:N]
;REAL ARRAY D[0:N];OWN REAL ARRAY B[0:N,0:N];LABEL S1,S2,S3,S4,S5,S6,REP
,S7,S8,S9,IT1,S10,S11,S12,S13,S14,S15,EXIT;S1:IF RSW THEN GO TO REP;FOR
I←1STEP 1UNTIL N DO FOR J←1STEP 1UNTIL N DO B[I,J]←A[I,J];S2:FOR I←1STEP
 1UNTIL N DO BEGIN L←I-1;FOR J←I STEP 1UNTIL N DO BEGIN Q←0;FOR K←1STEP
1UNTIL L DO Q←B[J,K]×B[K,I]+Q;B[J,I]←B[J,I]-Q END;BIG←0;K2←I;S3:FOR K←I
STEP 1UNTIL N DO BEGIN IF ABS(B[K,I])>BIG THEN BEGIN BIG←ABS(B[K,I]);K2←
K END END;S4:IF BIG≤EPS THEN GO TO E1;F[I]←K2;IF K2≠I THEN S5:FOR K←1STE
P 1UNTIL N DO BEGIN TEMP←A[K2,K];A[K2,K]←A[I,K];A[I,K]←TEMP;TEMP←B[K2,K]
;B[K2,K]←B[I,K];B[I,K]←TEMP;END;DIAG←B[I,I];S6:FOR J←I+1STEP 1UNTIL N DO
 BEGIN Q←0;FOR K←1STEP 1UNTIL L DO Q←B[I,K]×B[K,J]+Q;B[I,J]←(B[I,J]-Q)/D
IAG END END;REP:FOR I←1STEP 1UNTIL N DO BEGIN TEMP←C[F[I]];C[F[I]]←C[I];
D[I]←C[I]←TEMP END;FOR I←1STEP 1UNTIL N DO BEGIN L←I-1;Q←0;S7:FOR K←1STE
P 1UNTIL L DO Q←B[I,K]×D[K]+Q;D[I]←(D[I]-Q)/B[I,I]END;S8:FOR I←N STEP-1U
NTIL 1DO BEGIN Q←0;FOR K←I+1STEP 1UNTIL N DO Q←B[I,K]×X[K]+Q;X[I]←D[I]-Q
 END;S9:IF E=0THEN GO TO EXIT;J1←0;IT1:IF J1>K1 THEN GO TO E2;NORM←0;FOR
 I←1STEP 1UNTIL N DO BEGIN Q←0;L←I-1;S10:FOR K←1STEP 1UNTIL N DO Q←A[I,K
]×X[K]+Q;D[I]←C[I]-Q;S11:NORM←ABS(D[I])+NORM;Q←0;S12:FOR K←1STEP 1UNTIL
L DO Q←B[I,K]×D[K]+Q;D[I]←(D[I]-Q)/B[I,I]END;FOR I←N STEP-1UNTIL 1DO BEG
IN Q←0;S13:FOR K←I+1STEP 1UNTIL N DO Q←B[I,K]×D[K]+Q;X[I]←X[I]+D[I]-Q EN
D;S14:J1←J1+1;S15:IF N×E<NORM THEN GO TO IT1;EXIT:END;
READ (CR , / , WO , DW , WM ) ;
LDO:  READ (CR , / , L , L1 , L2 , W , IP , IT ) [FINIS]  ;
READ (CR , / , WM1 , WM2 , H1 , H2 , PH , R1 , R2 ) ;
READ (CR , / , P , FOR J←1 STEP 1 UNTIL P DO  [LZ[J]] )  ;
READ (CR , / , K1X , K2X , K1Y , K2Y ) ;
READ (CR , / , C1X , C2X , C1Y , C2Y ) ;
```

```
READ (CR , / , D1X , D2X , D1Y , D2Y ) ;
READ (CR , / , R1X , R2X , R1Y , R2Y )          ;
G ← 32.2 × 12 ;
PI ← 3.14159265  ;
M← W/G ;    DM1 ← WM1 / G  ;  DM2 ← WM2/G  :
RPP ← IP / M ;  RTT← IT / M  ;
RP ← RPP/ (L× L )  ;   RT ← RTT / ( L× L )  ;
L11 ← L1 / L  ; L22 ← L2 / L  ;
RO1 ← H1 - L1  ;  RO2 ← H2 - L1  :
PHI  ← (PH × PI ) / 180  ;
N ← 8 ;  RSW ← FALSE  ;  EPS ← 4.0 @-10  ;K1 ← 2   ;      E←1.0@-6   ;
RAD ← 57.29578  ;
K1XX ← K1X / M   ;            K2XX ← K2X /M  ;
K1YY ← K1Y /M  ;              K2YY ← K2Y /M  ;
C1XX ← C1X / M  ;             C2XX ← C2X / M  ;
C1YY ← C1Y / M  ;             C2YY ← C2Y  / M   ;
R1XX ← R1X / M  ;             R2XX ←  R2X / M  ;
R1YY ← R1Y / M ;              R2YY ← R2Y / M  ;
D1XX ← D1X / M  ;             D2XX ← D2X / M  ;
D1YY ← D1Y / M  ;             D2YY ← D2Y / M  ;
PN1 ← DM2 × R2 × SIN (PHI )  ;
PD1 ← DM1 × R1 + DM2 × R2 × COS (PHI )  ;
SI ← ANGLE (PN1 , PD1 )  ;
PN2 ← RO2 × R2 × DM2 × SIN (PHI ) ;
PD2 ← RO1 × R1 × DM1 + RO2 × R2 × DM2 × COS (PHI )  ;
SIT ← ANGLE (PN2 , PD2 ) ;
WRITE (LP[3])  ;
WRITE (LP , HEAD1) ;
WRITE (LP[PAGE]) ;
WRITE (LP , HEAD2 , L , L1 , L2 , H1 , H2 , W , WM1 , WM2 , K1X , K2X ,
 K1Y , K2Y , C1X , C2X , C1Y , C2Y , R1X , R2X , R1Y , R2Y , D1X , D2X ,
D1Y ,D2Y , IP , IT , R1 , R2 , PH  )  ;
I← 0 ;
FOR  WW ←WO  STEP  DW  UNTIL  WM  DO
BEGIN
I ←I + 1  ;
OMEGA[I] ← WW  ;
     RPM[I]←OMEGA[I]×60;
S[ I ] ← 2 × PI × OMEGA [I ]  ;
SS [ I ] ← S [ I ] × S [ I ]   ;
BEGIN
REAL  XXXX  ;
A[1,1] ← K1XX - L22 × SS [I]  ;
A[1,2] ← C1XX × S[I] ;
A[1,3] ← K2XX - L11 × SS[I]  ;
A[1,4] ← C2XX × S[I]  ;
A[1,5] ← R1YY  ;
A[1,6] ← D1YY × S[I]  ;
A[1,7] ← R2YY  ;
A[1,8] ← D2YY × S [I]  ;
A[2,1] ← - C1XX × S[I]  ;
A[2,2] ← K1XX - L22 × SS[I]  ;
A[2,3] ← - C2XX × S[I] ;
A[2,4] ← K2XX - L11 × SS[I]   ;
A[2,5] ← - D1YY × S[I]  ;
A[2,6] ← R1YY  ;
```

```
A[2,7] ← - D2YY × S[I] ;
A[2,8] ← R2YY ;
A[3,1] ← R1XX ;
A[3,2] ← D1XX × S[I] ;
A[3,3] ← R2XX ;
A[3,4 ] ← D2XX × S[I] ;
A[3,5] ← K1YY - L22 × SS[I] ;
A[3,6] ← C1YY × S[I] ;
 A[3,7] ← K2YY - L11 × SS[I] ;
A[3,8] ← C2YY × S[I] ;
A[4,1] ← - D1XX × S[I] ,
A[4,2] ← R1XX ;
A[4,3] ← - D2XX × S[I] ;
A[4,4] ← R2XX ;
A[4,5] ← - C1YY × S[I] ;
A[4,6] ← K1YY - L22 ×SS[I] ;
A[4,7] ← - C2YY × S[I] ;
A[4,8] ← K2YY - L11 × SS[I] ;
A[5,1] ← RT × SS[I] - K1XX × L11 ;
A[5,2] ← - C1XX × L11 × S[I] ;
A[5,3] ← - RT × SS[I] + K2XX × L22 ;
A[5,4] ← C2XX × L22 × S[I] ;
A[5,5] ← -R1YY × L11 ;
A[5,6] ← - RP × SS[I] - L11 × S[I] ×D1YY ;
A[5,7] ← R2YY × L22 ;
A[5,8] ← RP × SS[I] + D2YY × L22 × S[I] ;
A[6,1] ← C1XX × L11 × S[I] ;
A[6,2] ← RT × SS[I] - K1XX × L11 ;
A[6,3] ← - C2XX × L22 × S[I] ;
A[6,4] ← - RT × SS[I] + K2XX × L22 ;
A[6,5] ← RP × SS[I] + D1YY × L11 × S[I] ;
A[6,6] ← -R1YY × L11 ;
A[6,7] ← - RP × SS[I] - D2YY × L22 × S[I] ;
A[6,8] ← R2YY × L22 ;
A[7,1] ← - R1XX × L11 ;
A[7,2] ← RP × SS[I] - D1XX × L11 × S[I] ;
A[7,3] ← R2XX × L22 ;
A[7,4] ← - RP × SS[I] + D2XX × L22 × S[I] ;
A[7,5] ← RT × SS[I] - K1YY × L11 ;
A[7,6] ← - C1YY × L11 × S[I] ;
A[7,7] ← -RT × SS[I] + K2YY × L22 ;
A[7,8] ← C2YY × L22 × S[I] ;
A[8,1] ← - RP × SS[I] + D1XX × L11 ×S[I] ;
A[8,2] ← -R1XX × L11 ;
A[8,3] ← RP × SS[I] - D2XX × L22 × S[I] ;
A[8,4] ← R2XX × L22 ;
A[8,5] ← C1YY × L11 × S[I] ;
A[8,6] ← RT × SS[I] - K1YY × L11 ;
A[8,7] ← - C2YY × L22 × S[I] ;
A[8,8] ← - RT × SS[I] + K2YY × L22 ;
C[1] ← (DM1 × SS[I] × R1 ) / M + ( DM2 × SS[I] × R2 × COS(PHI))/ M ;
C[2] ← - ( DM2 × SS[I] × R2 × SIN (PHI)) / M ;
C[3] ←   ( DM2 × SS[I] × R2 × SIN (PHI)) / M ;
C[4] ← (DM1 × SS[I] × R1 ) / M + ( DM2 × SS[I] × R2 × COS(PHI))/ M ;
C[5] ← (DM1 × RO1 × SS[I] × R1 + DM2 × RO2 × SS[I]×R2 × COS(PHI)) /
 (M× L ) ;
```

```
C[6] ←- (DM2 × SS[I] × RO2 × R2 × SIN ( PHI )) / (M× L)  ;
C[7] ←  (DM2 × SS[I] × RO2 × R2 × SIN ( PHI )) / (M× L)  ;
C[8] ← (DM1 × RO1 × SS[I] × R1 + DM2 × RO2 × SS[I]×R2 × COS(PHI)) /
 (M× L )  ;
END ;
SOLVE  (N , A , C , RSW , E , K1 , EPS , X , E1 , E2 )  ;
BB ← X[1] × X[1] + X[2]× X[2]   ;
XX1[I] ← SQRT (BB)  ;
CC ← X[3] × X[3] + X[4] × X[4]  ;
XX2[I] ← SQRT ( CC )   ;
DD ← X[5] × X[5] + X[6] × X[6]    ;
YY1[I] ← SQRT ( DD )  ;
EE ← X[7] × X[7] + X[8] × X[8] ;
YY2[I] ← SQRT ( EE )   ;
COMMENT  THE FOLLOWING CALCULATES  THE PHASE ANGLES BETWEEN
DISPLACEMENT AND UNBALANCE   ;
PX1 ← ANGLE (X[2] , X[1] )  ;          PX2 ← ANGLE (X[4] , X[3] )  ;
PY1 ← ANGLE (- X[5] , X[6] )  ;        PY2 ← ANGLE (-X[7] , X[8] )  ;
SIX1[I] ← (SI + PX1) × RAD ;        SIX2 [I] ← (SI + PX2) × RAD ;
SIY1[I] ← (SI + PY1) × RAD ;        SIY2 [I] ← (SI + PY2) × RAD ;
COMMENT  THE  FOLLOWING  CALCULATES  THE  PHASE  ANGLES  OF  ALFA1
 ALFA2  AND  UNBALANCE  MOMENT  ;
PA1 ← ANGLE ( X[4] - X[2] , X[3] - X[1]  )  ;
SIA1[I] ←(SIT + PA1)× RAD;
PA2 ← ANGLE (X[5] - X[7] , X[8] - X[6]  )  ;
SIA2[I] ←(SIT + PA2)× RAD ;
ALFA1[I] ← SQRT (( X[3] - X[1])* 2 + ( X[4] - X[2] ) * 2 ) / L  ;
ALFA2[I] ← SQRT (( X[7] - X[5] ) * 2 + ( X[8] - X[6] ) * 2 ) / L  ;
IF  P=0  THEN  GO  TO  LOO1  ;
FOR  J←1  STEP  1  UNTIL  P  DO
BEGIN
ARBITRARYDISPLACEMENT ( LZ[J] , L , X , XL[I,J] , YL[I,J] , PXL[I,J] ,
 PYL[I,J]  )  ;
END  ;
LOO1:   FORCE(C1X , K1X ,D1Y , R1Y , X[1] , X[2] , X[5], X[6] , S[I] ,
 FX1[I] , PFX1[I] , PFY )  ;
FORCE (C2X , K2X , D2Y , R2Y , X[3] , X[4] , X[7] , X[8] , S[I] ,
FX2[I] , PFX2[I] , PFY )  ;
FORCE ( C1Y , K1Y , D1X , R1X , X[5] , X[6] , X[1] , X[2],S[I] ,
FY1[I] , PFX  , PFY1[I] )  ;
FORCE ( C2Y , K2Y , D2X , R2X , X[7] , X[8] , X[3] , X[4] , S[I] ,
 FY2[I] , PFX , PFY2[I] )  ;
COMMENT  THE FOLLOWING  CALCULATES  THE  RELATIVE  PHASE  ANGLES
BETWEEN FX1 , FX2 , FY1 , FY2 ,  WRT  THE  UNBALANCE  FORCE  ;
PUBFX1[I] ←(SI + PFX1[I])× RAD ;
PUBFX2[I] ←(SI + PFX2[I])× RAD ;
PUBFY1[I] ← (SI + PFY1[I])× RAD ;
PUBFY2[I] ←(SI+ PFY2 [I])× RAD;
END  ;
WRITE (LP , OUT1 )  ;
FOR J←1  STEP 1 UNTIL I  DO
BEGIN
WRITE ( LP , OUT2 , OMEGA[J] , XX1[J] , XX2[J] , SIX1[J]      ,
SIX2[J]      , ALFA1[J]  , SIA1[J]     , FX1[J] , FX2[J] ,
PUBFX1[J]       , PUBFX2[J]       )  ;
END  ;
```

```
        WRITE (LP[PAGE] )  ;
        WRITE (LP , OUT3 )  ;
        FOR  J ←1 STEP  1 UNTIL  I  DO
        BEGIN
        WRITE (LP , OUT2 , OMEGA[J] , YY1[J] , YY2[J] , SIY1[J]
        SIY2[J]        , ALFA2[J] ,SIA2[J]        ,FY1[J] , FY2[J] ,
        PUBFY1[J]        , PUBFY2[J]        )  ;
        END ;
        WRITE (LP [PAGE] )  ;
        IF  P=0  THEN  GO  TO  LDO  ELSE
        BEGIN
        WRITE (LP , OUT4 )  ;
        FOR J←1  STEP  1 UNTIL  P  DO
        FOR  K←1  STEP  1  UNTIL I  DO
        BEGIN
        WRITE (LP , OUT5 , LZ[J] ,  XL[K,J] ,     YL[K,J]  ,  PXL[K,J] × RAD ,
        PYL[K,J] × RAD , OMEGA[K] )  ;
        END  ;
        END  ;
             GOPLOTTER(RPM);
        GO  TO  LDO  ;
        E2:   WRITE (LP , < "ACCURACY  NOT  OBTAINED " > )  ;
        GO  TO  LDO  ;
        E1:  WRITE (LP , < " SINGULARITY  OR  ILL  CONDITIONED  MATRIX " > ) ;
        GO  TO  LDO  ;
        FINIS:   END.
      ARCTAN IS SEGMENT NUMBER 0065,PRT ADDRESS IS 0116
      COS IS SEGMENT NUMBER 0066,PRT ADDRESS IS 0074
      EXP IS SEGMENT NUMBER 0067,PRT ADDRESS IS 0071
      LN IS SEGMENT NUMBER 0068,PRT ADDRESS IS 0070
      SIN IS SEGMENT NUMBER 0069,PRT ADDRESS IS 0075
      SQRT IS SEGMENT NUMBER 0070,PRT ADDRESS IS 0445
      OUTPUT(W) IS SEGMENT NUMBER 0071,PRT ADDRESS IS 0363
      BLOCK CONTROL IS SEGMENT NUMBER 0072,PRT ADDRESS IS 0005
      INPUT(W) IS SEGMENT NUMBER 0073,PRT ADDRESS IS 0370
      X TO THE I IS SEGMENT NUMBER 0074,PRT ADDRESS IS 0072
      GO TO SOLVER  IS SEGMENT NUMBER 0075,PRT ADDRESS IS 0064
      ALGOL WRITE   IS SEGMENT NUMBER 0076,PRT ADDRESS IS 0014
      ALGOL READ    IS SEGMENT NUMBER 0077,PRT ADDRESS IS 0015
      ALGOL SELECT  IS SEGMENT NUMBER 0078,PRT ADDRESS IS 0016
COMPILATION TIME =  137 SECONDS.
NUMBER OF ERRORS DETECTED = 000. LAST ERROR ON CARD #
NUMBER OF SEQUENCE ERRORS COUNTED =   0.
NUMBER OF SLOW WARNINGS =   0.
PRT SIZE= 330;    TOTAL SEGMENT SIZE= 3463 WORDS.
DISK STORAGE REQ.=  173 SEGS.;    NO. SEGS.=  79.
ESTIMATED CORE STORAGE REQUIREMENT =  6394 WORDS.
```

TABLE B-I. - INPUT CHARACTERISTICS AND ROTOR HORIZONTAL RESPONSE FOR A SINGLE MASS ROTOR WITH FLEXIBLE SUPPORT AND DAMPING

L= 3.0000@+01INCH	L1= 1.5000@+01INCH	L2= 1.5000@+01INCH	H1= 0.0000@+00INCH
H2= 0.0000@+00INCH	W= 1.1000@+02LB	WM1= 2.0000@-01LB	WM2= 2.0000@-01LB
K1X= 2.0000@+04LB/IN	K2X= 1.5000@+04LB/IN	K1Y= 1.6000@+04LB/IN	K2Y= 1.2000@+04LB/IN
C1X= 7.0000@+00LB.SEC/IN	C2X= 7.0000@+00LB.SEC/IN	C1Y= 7.0000@+00LB.SEC/IN	C2Y= 7.0000@+00LB.SEC/IN
R1X= 0.0000@+00LB/IN	R2X= 0.0000@+00LB/IN	R1Y= 0.0000@+00LB/IN	R2Y= 0.0000@+00LB/IN
D1X= 0.0000@+00LB.SEC/IN	D2X= 0.0000@+00LB.SEC/IN	D1Y= 0.0000@+00LB.SEC/IN	D2Y= 0.0000@+00LB.SEC/IN
IP= 5.7000@-01LB-IN-SEC2	IT= 2.1600@+01LB-IN-SEC2	R1= 2.0000@+00INCH	R2= 2.0000@+00INCH
	PHI= 0.0000@+00DEGREES		

SPEED	X1	X2	SIX1	SIX2	ALFA1	SIA1	FX1	FX2	PUBFX1	PUBFX2
40	9.9043@-03	3.2572@-03	8.5	19.5	2.2451@-04	183.2	1.9885@+02	4.9193@+01	3.5	12.8
41	1.0746@-02	3.8320@-03	9.2	20.8	2.3449@-04	182.9	2.1579@+02	5.7894@+01	4.0	13.9
42	1.1677@-02	4.5179@-03	9.9	22.2	2.4422@-04	182.4	2.3454@+02	6.8280@+01	4.6	15.2
43	1.2715@-02	5.3410@-03	10.8	23.8	2.5357@-04	181.7	2.5543@+02	8.0750@+01	5.4	16.7
44	1.3878@-02	6.3358@-03	11.9	25.7	2.6243@-04	180.8	2.7885@+02	9.5824@+01	6.3	18.3
45	1.5192@-02	7.5472@-03	13.1	27.8	2.7070@-04	179.5	3.0533@+02	1.1419@+02	7.5	20.3
46	1.6688@-02	9.0356@-03	14.7	30.3	2.7836@-04	177.7	3.3547@+02	1.3676@+02	8.9	22.6
47	1.8404@-02	1.0882@-02	16.6	33.3	2.8570@-04	175.2	3.7004@+02	1.6477@+02	10.7	25.4
48	2.0383@-02	1.3193@-02	19.0	36.8	2.9376@-04	171.6	4.0992@+02	1.9984@+02	12.9	28.8
49	2.2672@-02	1.6103@-02	22.0	41.3	3.0545@-04	166.6	4.5607@+02	2.4402@+02	15.9	33.1
50	2.5306@-02	1.9765@-02	26.0	46.8	3.2683@-04	160.2	5.0918@+02	2.9965@+02	19.7	38.5
51	2.8266@-02	2.4323@-02	31.2	53.9	3.6777@-04	153.1	5.6887@+02	3.6891@+02	24.8	45.4
52	3.1388@-02	2.9819@-02	38.0	62.8	4.4005@-04	147.1	6.3184@+02	4.5245@+02	31.5	54.1
53	3.4213@-02	3.5971@-02	46.8	73.9	5.5113@-04	144.4	6.8888@+02	5.4605@+02	40.1	65.0
54	3.5910@-02	4.1900@-02	57.3	87.1	6.9447@-04	146.0	7.2324@+02	6.3633@+02	50.5	78.1
55	3.5605@-02	4.6233@-02	68.6	101.6	8.4479@-04	151.4	7.1729@+02	7.0246@+02	61.7	92.4
56	3.3157@-02	4.8032@-02	79.1	115.7	9.7148@-04	158.5	6.6815@+02	7.3013@+02	72.1	106.4
57	2.9387@-02	4.7573@-02	87.4	128.3	1.0612@-03	165.5	5.9234@+02	7.2349@+02	80.2	118.8
58	2.5348@-02	4.5857@-02	92.6	138.6	1.1206@-03	171.4	5.1106@+02	6.9774@+02	85.3	128.9
59	2.1703@-02	4.3763@-02	94.8	146.8	1.1626@-03	176.2	4.3770@+02	6.6620@+02	87.4	137.0
60	1.8713@-02	4.1771@-02	94.3	153.4	1.1973@-03	179.9	3.7750@+02	6.3619@+02	86.8	143.4
61	1.6423@-02	4.0070@-02	91.5	158.6	1.2306@-03	182.8	3.3141@+02	6.1059@+02	83.9	148.5
62	1.4799@-02	3.8705@-02	86.8	162.9	1.2657@-03	185.2	2.9872@+02	5.9009@+02	79.1	152.6
63	1.3782@-02	3.7659@-02	80.8	166.5	1.3040@-03	187.1	2.7827@+02	5.7445@+02	72.9	156.1
64	1.3301@-02	3.6900@-02	74.2	169.6	1.3463@-03	188.7	2.6864@+02	5.6316@+02	66.2	159.0
65	1.3278@-02	3.6392@-02	67.6	172.3	1.3930@-03	190.2	2.6826@+02	5.5570@+02	59.4	161.5
66	1.3631@-02	3.6103@-02	61.5	174.7	1.4441@-03	191.5	2.7548@+02	5.5159@+02	53.2	163.7
67	1.4285@-02	3.6008@-02	56.2	176.8	1.5000@-03	192.7	2.8879@+02	5.5044@+02	47.8	165.7
68	1.5177@-02	3.6086@-02	51.7	178.8	1.5608@-03	193.8	3.0691@+02	5.5194@+02	43.2	167.5
69	1.6260@-02	3.6323@-02	48.1	180.6	1.6267@-03	194.8	3.2892@+02	5.5588@+02	39.5	169.2
70	1.7502@-02	3.6707@-02	45.2	182.3	1.6980@-03	195.8	3.5417@+02	5.6208@+02	36.5	170.7
71	1.8884@-02	3.7230@-02	43.0	184.0	1.7750@-03	196.9	3.8226@+02	5.7043@+02	34.1	172.2
72	2.0395@-02	3.7890@-02	41.2	185.6	1.8582@-03	197.9	4.1298@+02	5.8088@+02	32.2	173.6
73	2.2031@-02	3.8684@-02	39.9	187.1	1.9479@-03	198.9	4.4626@+02	5.9340@+02	30.8	175.0
74	2.3795@-02	3.9613@-02	38.9	188.6	2.0448@-03	199.9	4.8216@+02	6.0802@+02	29.7	176.4
75	2.5694@-02	4.0682@-02	38.3	190.2	2.1497@-03	201.0	5.2082@+02	6.2481@+02	28.9	177.8
76	2.7739@-02	4.1896@-02	37.9	191.7	2.2633@-03	202.1	5.6247@+02	6.4386@+02	28.4	179.1
77	2.9945@-02	4.3265@-02	37.8	193.3	2.3866@-03	203.2	6.0743@+02	6.6532@+02	28.2	180.5
78	3.2332@-02	4.4800@-02	37.9	194.9	2.5209@-03	204.5	6.5609@+02	6.8936@+02	28.1	182.0
79	3.4923@-02	4.6515@-02	38.2	196.5	2.6674@-03	205.8	7.0892@+02	7.1619@+02	28.3	183.5

80	3.7742e-02	4.8423e-02	38.7	198.3	2.8275e-03	207.2	7.6644e+02	7.4607e+02	28.7	185.1
81	4.0818e-02	5.0542e-02	39.4	200.1	3.0028e-03	208.8	8.2922e+02	7.7921e+02	29.3	186.8
82	4.4178e-02	5.2883e-02	40.5	202.2	3.1945e-03	210.5	8.9780e+02	8.1586e+02	30.2	188.6
83	4.7843e-02	5.5458e-02	41.7	204.3	3.4039e-03	212.4	9.7266e+02	8.5615e+02	31.4	190.7
84	5.1829e-02	5.8267e-02	43.3	206.7	3.6315e-03	214.5	1.0541e+03	9.0012e+02	32.9	192.9
85	5.6138e-02	6.1303e-02	45.3	209.4	3.8771e-03	217.0	1.1422e+03	9.4767e+02	34.7	195.4
86	6.0759e-02	6.4544e-02	47.5	212.3	4.1397e-03	219.7	1.2367e+03	9.9847e+02	36.8	198.1
87	6.5658e-02	6.7954e-02	50.2	215.5	4.4170e-03	222.7	1.3370e+03	1.0520e+03	39.3	201.1
88	7.0783e-02	7.1480e-02	53.2	219.0	4.7056e-03	226.0	1.4419e+03	1.1073e+03	42.2	204.5
89	7.6061e-02	7.5054e-02	56.6	222.8	5.0007e-03	229.7	1.5501e+03	1.1635e+03	45.5	208.1
90	8.1394e-02	7.8588e-02	60.3	226.9	5.2964e-03	233.7	1.6595e+03	1.2192e+03	49.1	212.1
91	8.6664e-02	8.1983e-02	64.3	231.3	5.5853e-03	238.0	1.7676e+03	1.2728e+03	53.0	216.4
92	9.1732e-02	9.5124e-02	68.7	236.0	5.8591e-03	242.6	1.8718e+03	1.3225e+03	57.3	220.9
93	9.6445e-02	.7889e-02	73.3	240.9	6.1086e-03	247.4	1.9688e+03	1.3665e+03	61.7	225.7
94	1.0065e-01	.0160e-02	78.1	246.0	6.3249e-03	252.4	2.0555e+03	1.4028e+03	66.4	230.6
95	1.0419e-01	.1836e-02	83.1	251.3	6.4995e-03	257.6	2.1288e+03	1.4300e+03	71.3	235.7
96	1.0696e-01	[illegible].2841e-02	88.1	256.6	6.6263e-03	262.8	2.1864e+03	1.4467e+03	76.2	240.8
97	1.0889e-01	[illegible].3145e-02	93.2	261.8	6.7018e-03	267.9	2.2268e+03	1.4526e+03	81.1	246.0
98	1.0995e-01	9.2758e-02	98.1	267.0	6.7257e-03	273.0	2.2496e+03	1.4477e+03	86.0	251.0
99	1.1019e-01	9.1737e-02	102.9	272.1	6.7010e-03	278.0	2.2555e+03	1.4329e+03	90.7	255.9
100	1.0969e-01	9.0170e-02	107.5	276.9	6.6335e-03	282.7	2.2462e+03	1.4095e+03	95.1	260.5
101	1.0856e-01	8.8162e-02	111.9	281.5	6.5303e-03	287.2	2.2240e+03	1.3792e+03	99.4	265.0
102	1.0691e-01	8.5828e-02	116.0	285.8	6.3994e-03	291.4	2.1914e+03	1.3438e+03	103.4	269.1
103	1.0489e-01	8.3272e-02	119.8	289.8	6.2483e-03	295.4	2.1510e+03	1.3046e+03	107.1	273.0
104	1.0260e-01	8.0589e-02	123.4	293.5	6.0840e-03	299.0	2.1050e+03	1.2638e+03	110.5	276.5
105	1.0014e-01	7.7855e-02	126.6	296.9	5.9124e-03	302.4	2.0556e+03	1.2219e+03	113.6	279.8
106	9.7598e-02	7.5130e-02	129.6	300.1	5.7381e-03	305.5	2.0043e+03	1.1801e+03	116.5	282.8
107	9.5027e-02	7.2458e-02	132.3	303.0	5.5646e-03	308.3	1.9524e+03	1.1391e+03	119.1	285.6
108	9.2476e-02	6.9870e-02	134.9	305.7	5.3945e-03	310.9	1.9010e+03	1.0994e+03	121.5	288.1
109	8.9981e-02	6.7387e-02	137.2	308.1	5.2296e-03	313.3	1.8506e+03	1.0612e+03	123.7	290.4
110	8.7564e-02	6.5020e-02	139.3	310.4	5.0712e-03	315.5	1.8018e+03	1.0248e+03	125.7	292.5
111	8.5238e-02	6.2775e-02	141.2	312.5	4.9197e-03	317.5	1.7548e+03	9.9024e+02	127.5	294.5
112	8.3014e-02	6.0653e-02	143.0	314.4	4.7757e-03	319.4	1.7099e+03	9.5759e+02	129.2	296.2
113	8.0895e-02	5.8652e-02	144.6	316.2	4.6392e-03	321.1	1.6671e+03	9.2681e+02	130.7	297.8
114	7.8882e-02	5.6769e-02	146.2	317.8	4.5100e-03	322.7	1.6265e+03	8.9785e+02	132.1	299.3
115	7.6972e-02	5.4998e-02	147.6	319.3	4.3881e-03	324.1	1.5879e+03	8.7061e+02	133.4	300.7
116	7.5165e-02	5.3334e-02	148.9	320.8	4.2729e-03	325.5	1.5514e+03	8.4502e+02	134.5	302.0
117	7.3454e-02	5.1770e-02	150.1	322.1	4.1644e-03	326.8	1.5169e+03	8.2098e+02	135.6	303.1
118	7.1836e-02	5.0301e-02	151.2	323.3	4.0620e-03	327.9	1.4843e+03	7.9840e+02	136.6	304.2
119	7.0306e-02	4.8919e-02	152.2	324.5	3.9654e-03	329.0	1.4535e+03	7.7717e+02	137.5	305.2
120	6.8859e-02	4.7619e-02	153.2	325.5	3.8743e-03	330.0	1.4243e+03	7.5721e+02	138.4	306.1
121	6.7490e-02	4.6396e-02	154.1	326.5	3.7883e-03	331.0	1.3968e+03	7.3844e+02	139.2	307.0
122	6.6194e-02	4.5243e-02	154.9	327.5	3.7070e-03	331.9	1.3707e+03	7.2075e+02	139.9	307.8
123	6.4967e-02	4.4156e-02	155.7	328.4	3.6302e-03	332.7	1.3460e+03	7.0409e+02	140.6	308.5
124	6.3803e-02	4.3130e-02	156.4	329.2	3.5576e-03	333.5	1.3227e+03	6.8838e+02	141.2	309.2
125	6.2700e-02	4.2160e-02	157.1	330.0	3.4888e-03	334.3	1.3005e+03	6.7355e+02	141.8	309.9
126	6.1653e-02	4.1244e-02	157.8	330.8	3.4237e-03	335.0	1.2795e+03	6.5953e+02	142.3	310.5
127	6.0659e-02	4.0377e-02	158.4	331.5	3.3619e-03	335.6	1.2596e+03	6.4628e+02	142.8	311.0
128	5.9713e-02	3.9555e-02	159.0	332.1	3.3032e-03	336.3	1.2407e+03	6.3374e+02	143.3	311.6
129	5.8814e-02	3.8776e-02	159.6	332.8	3.2475e-03	336.9	1.2227e+03	6.2185e+02	143.7	312.0
130	5.7958e-02	3.8036e-02	160.1	333.4	3.1945e-03	337.4	1.2056e+03	6.1058e+02	144.1	312.5
131	5.7142e-02	3.7333e-02	160.6	334.0	3.1441e-03	338.0	1.1893e+03	5.9989e+02	144.5	312.9
132	5.6363e-02	3.6665e-02	161.0	334.5	3.0961e-03	338.5	1.1738e+03	5.8973e+02	144.9	313.3
133	5.5621e-02	3.6029e-02	161.5	335.0	3.0503e-03	339.0	1.1590e+03	5.8007e+02	145.2	313.7
134	5.4911e-02	3.5423e-02	161.9	335.5	3.0067e-03	339.4	1.1449e+03	5.7088e+02	145.5	314.1
135	5.4233e-02	3.4845e-02	162.3	336.0	2.9650e-03	339.9	1.1315e+03	5.6213e+02	145.8	314.4
136	5.3585e-02	3.4293e-02	162.7	336.5	2.9251e-03	340.3	1.1186e+03	5.5379e+02	146.1	314.7
137	5.2964e-02	3.3767e-02	163.1	336.9	2.8870e-03	340.7	1.1063e+03	5.4584e+02	146.3	315.0
138	5.2369e-02	3.3263e-02	163.4	337.3	2.8505e-03	341.1	1.0945e+03	5.3825e+02	146.6	315.3

TABLE B-II. - INPUT CHARACTERISTICS AND ROTOR VERTICAL RESPONSE FOR A SINGLE MASS ROTOR WITH FLEXIBLE SUPPORT AND DAMPING

L= 3.0000@+01INCH	L1= 1.5000@+01INCH	L2= 1.5000@+01INCH	H1= 0.0000@+00INCH
H2= 0.0000@+00INCH	W= 1.1000@+02LB	WM1= 2.0000@-01LB	WM2= 2.0000@-01LB
K1X= 2.0000@+04LB/IN	K2X= 1.5000@+04LB/IN	K1Y= 1.6000@+04LB/IN	K2Y= 1.2000@+04LB/IN
C1X= 7.0000@+00LB.SEC/IN	C2X= 7.0000@+00LB.SEC/IN	C1Y= 7.0000@+00LB.SEC/IN	C2Y= 7.0000@+00LB.SEC/IN
R1X= 0.0000@+00LB/IN	R2X= 0.0000@+00LB/IN	R1Y= 0.0000@+00LB/IN	R2Y= 0.0000@+00LB/IN
D1X= 0.0000@+00LB.SEC/IN	D2X= 0.0000@+00LB.SEC/IN	D1Y= 0.0000@+00LB.SEC/IN	D2Y= 0.0000@+00LB.SEC/IN
IP= 5.7000@-01LB-IN-SEC2	IT= 2.1600@+01LB-IN-SEC2	R1= 2.0000@+00INCH	R2= 2.0000@+00INCH
	PHI= 0.0000@+00DEGREES		

SPEED	Y1	Y2	SIY1	SIY2	ALFA2	SIA2	FY1	FY2	PUBFY1	PUBFY2
40	1.4693@-02	7.0564@-03	14.1	30.4	2.7216@-04	180.0	2.3651@+02	8.5582@+01	7.8	22.0
41	1.6266@-02	8.5869@-03	15.8	33.4	2.8273@-04	178.1	2.6191@+02	1.0420@+02	9.4	24.8
42	1.8079@-02	1.0508@-02	18.0	36.9	2.9397@-04	175.4	2.9118@+02	1.2758@+02	11.5	28.2
43	2.0174@-02	1.2936@-02	20.9	41.3	3.0768@-04	171.6	3.2503@+02	1.5715@+02	14.1	32.3
44	2.2588@-02	1.6014@-02	24.5	46.7	3.2771@-04	166.6	3.6403@+02	1.9465@+02	17.6	37.5
45	2.5316@-02	1.9892@-02	29.3	53.4	3.6129@-04	160.7	4.0814@+02	2.4193@+02	22.2	44.1
46	2.8243@-02	2.4659@-02	35.6	62.0	4.1921@-04	154.9	4.5548@+02	3.0009@+02	28.4	52.4
47	3.1010@-02	3.0169@-02	43.7	72.7	5.1183@-04	151.2	5.0028@+02	3.6736@+02	36.3	62.9
48	3.2914@-02	3.5773@-02	53.5	85.6	6.3936@-04	151.3	5.3118@+02	4.3587@+02	46.0	75.7
49	3.3110@-02	4.0294@-02	64.4	100.1	7.8339@-04	155.3	5.3454@+02	4.9126@+02	56.7	89.9
50	3.1301@-02	4.2711@-02	74.6	114.5	9.1466@-04	161.6	5.0553@+02	5.2106@+02	66.8	104.1
51	2.8120@-02	4.3008@-02	82.6	127.5	1.0152@-03	168.2	4.5432@+02	5.2504@+02	74.6	116.9
52	2.4567@-02	4.1983@-02	87.5	138.4	1.0867@-03	174.1	3.9706@+02	5.1286@+02	79.4	127.6
53	2.1344@-02	4.0459@-02	89.4	147.1	1.1399@-03	178.9	3.4512@+02	4.9458@+02	81.1	136.1
54	1.8751@-02	3.8934@-02	88.5	154.0	1.1846@-03	182.7	3.0330@+02	4.7627@+02	80.1	142.8
55	1.6840@-02	3.7626@-02	85.4	159.7	1.2270@-03	185.8	2.7251@+02	4.6060@+02	76.8	148.3
56	1.5577@-02	3.6604@-02	80.8	164.3	1.2711@-03	188.2	2.5216@+02	4.4840@+02	72.0	152.7
57	1.4905@-02	3.5869@-02	75.1	168.2	1.3191@-03	190.3	2.4138@+02	4.3972@+02	66.2	156.4
58	1.4758@-02	3.5399@-02	69.2	171.6	1.3722@-03	192.1	2.3912@+02	4.3428@+02	60.1	159.6
59	1.5059@-02	3.5167@-02	63.6	174.5	1.4307@-03	193.6	2.4410@+02	4.3176@+02	54.4	162.3
60	1.5725@-02	3.5146@-02	58.6	177.2	1.4948@-03	195.1	2.5500@+02	4.3183@+02	49.3	164.8
61	1.6684@-02	3.5315@-02	54.5	179.6	1.5647@-03	196.5	2.7067@+02	4.3425@+02	45.0	167.0
62	1.7879@-02	3.5659@-02	51.2	181.9	1.6407@-03	197.9	2.9019@+02	4.3881@+02	41.5	169.1
63	1.9272@-02	3.6164@-02	48.7	184.1	1.7230@-03	199.3	3.1295@+02	4.4539@+02	38.9	171.1
64	2.0840@-02	3.6823@-02	46.8	186.2	1.8121@-03	200.6	3.3855@+02	4.5387@+02	36.8	173.0
65	2.2567@-02	3.7629@-02	45.5	188.2	1.9083@-03	202.0	3.6679@+02	4.6419@+02	35.3	174.8
66	2.4448@-02	3.8581@-02	44.6	190.3	2.0121@-03	203.5	3.9755@+02	4.7632@+02	34.3	176.7
67	2.6483@-02	3.9676@-02	44.2	192.3	2.1240@-03	205.0	4.3085@+02	4.9026@+02	33.8	178.5
68	2.8675@-02	4.0915@-02	44.1	194.4	2.2445@-03	206.6	4.6675@+02	5.0600@+02	33.5	180.4
69	3.1031@-02	4.2299@-02	44.4	196.5	2.3742@-03	208.3	5.0535@+02	5.2357@+02	33.7	182.4
70	3.3559@-02	4.3831@-02	45.0	198.7	2.5136@-03	210.1	5.4679@+02	5.4300@+02	34.1	184.4
71	3.6267@-02	4.5510@-02	45.8	201.0	2.6633@-03	212.0	5.9121@+02	5.6431@+02	34.8	186.5
72	3.9163@-02	4.7339@-02	46.9	203.5	2.8236@-03	214.1	6.3876@+02	5.8751@+02	35.7	188.7
73	4.2254@-02	4.9315@-02	48.3	206.0	2.9949@-03	216.3	6.8954@+02	6.1259@+02	37.0	191.0
74	4.5543@-02	5.1434@-02	50.0	208.7	3.1771@-03	218.7	7.4362@+02	6.3951@+02	38.5	193.5
75	4.9030@-02	5.3687@-02	52.0	211.6	3.3700@-03	221.3	8.0097@+02	6.6814@+02	40.3	196.2
76	5.2704@-02	5.6058@-02	54.2	214.7	3.5729@-03	224.1	8.6146@+02	6.9830@+02	42.4	199.1
77	5.6545@-02	5.8521@-02	56.7	218.0	3.7842@-03	227.2	9.2476@+02	7.2968@+02	44.8	202.2
78	6.0520@-02	6.1040@-02	59.6	221.5	4.0017@-03	230.5	9.9032@+02	7.6183@+02	47.5	205.6
79	6.4576@-02	6.3567@-02	62.7	225.3	4.2220@-03	234.1	1.0573@+03	7.9413@+02	50.5	209.1

80	6.8643@-02	6.6035@-02	66.2	229.3	4.4407@-03	237.9	1.1245@+03	8.2578@+02	53.8	212.9
81	7.2628@-02	6.8368@-02	69.9	233.5	4.6522@-03	242.0	1.1905@+03	8.5581@+02	57.3	217.0
82	7.6424@-02	7.0478@-02	73.9	238.0	4.8499@-03	246.3	1.2535@+03	8.8310@+02	61.2	221.3
83	7.9908@-02	7.2273@-02	78.1	242.7	5.0269@-03	250.8	1.3114@+03	9.0652@+02	65.2	225.7
84	8.2965@-02	7.3670@-02	82.4	247.4	5.1765@-03	255.4	1.3624@+03	9.2500@+02	69.4	230.3
85	8.5491@-02	7.4606@-02	86.9	252.3	5.2931@-03	260.1	1.4047@+03	9.3771@+02	73.8	235.0
86	8.7416@-02	7.5044@-02	91.4	257.1	5.3734@-03	264.8	1.4372@+03	9.4421@+02	78.1	239.6
87	8.8715@-02	7.4990@-02	95.8	261.8	5.4164@-03	269.4	1.4595@+03	9.4452@+02	82.4	244.1
88	8.9406@-02	7.4482@-02	100.1	266.4	5.4241@-03	273.9	1.4718@+03	9.3912@+02	86.5	248.5
89	8.9551@-02	7.3591@-02	104.2	270.8	5.4009@-03	278.2	1.4751@+03	9.2889@+02	90.5	252.7
90	8.9241@-02	7.2405@-02	108.1	274.9	5.3528@-03	282.2	1.4709@+03	9.1491@+02	94.2	256.7
91	8.8580@-02	7.1016@-02	111.8	278.8	5.2862@-03	286.0	1.4610@+03	8.9834@+02	97.7	260.3
92	8.7672@-02	6.9509@-02	115.2	282.4	5.2074@-03	289.5	1.4469@+03	8.8025@+02	101.0	263.8
93	8.6605@-02	6.7953@-02	118.3	285.8	5.1216@-03	292.8	1.4303@+03	8.6150@+02	104.0	267.0
94	8.5448@-02	6.6393@-02	121.2	288.9	5.0327@-03	295.9	1.4121@+03	8.4267@+02	106.8	269.9
95	8.4242@-02	6.4854@-02	124.0	291.9	4.9428@-03	298.7	1.3931@+03	8.2408@+02	109.4	272.7
96	8.3009@-02	6.3345@-02	126.6	294.8	4.8530@-03	301.5	1.3736@+03	8.0582@+02	111.8	275.4
97	8.1754@-02	6.1860@-02	129.1	297.5	4.7632@-03	304.1	1.3538@+03	7.8784@+02	114.1	277.9
98	8.0475@-02	6.0391@-02	131.4	300.1	4.6731@-03	306.6	1.3335@+03	7.7002@+02	116.3	280.3
99	7.9165@-02	5.8929@-02	133.6	302.6	4.5822@-03	308.9	1.3127@+03	7.5226@+02	118.4	282.6
100	7.7823@-02	5.7470@-02	135.8	305.0	4.4902@-03	311.2	1.2914@+03	7.3450@+02	120.4	284.9
101	7.6452@-02	5.6015@-02	137.8	307.3	4.3973@-03	313.4	1.2695@+03	7.1676@+02	122.3	287.0
102	7.5059@-02	5.4569@-02	139.7	309.4	4.3039@-03	315.4	1.2473@+03	6.9909@+02	124.1	288.9
103	7.3654@-02	5.3139@-02	141.5	311.5	4.2105@-03	317.3	1.2248@+03	6.8160@+02	125.7	290.8
104	7.2249@-02	5.1735@-02	143.2	313.4	4.1180@-03	319.1	1.2023@+03	6.6439@+02	127.3	292.5
105	7.0856@-02	5.0365@-02	144.8	315.2	4.0268@-03	320.8	1.1800@+03	6.4759@+02	128.7	294.2
106	6.9486@-02	4.9036@-02	146.3	316.9	3.9378@-03	322.4	1.1580@+03	6.3128@+02	130.1	295.7
107	6.8147@-02	4.7754@-02	147.7	318.5	3.8512@-03	323.9	1.1365@+03	6.1554@+02	131.3	297.1
108	6.6846@-02	4.6522@-02	149.0	319.9	3.7675@-03	325.3	1.1157@+03	6.0041@+02	132.5	298.3
109	6.5589@-02	4.5342@-02	150.2	321.3	3.6870@-03	326.6	1.0955@+03	5.8592@+02	133.5	299.5
110	6.4378@-02	4.4216@-02	151.3	322.6	3.6097@-03	327.8	1.0761@+03	5.7209@+02	134.5	300.6
111	6.3214@-02	4.3143@-02	152.3	323.8	3.5357@-03	328.9	1.0575@+03	5.5893@+02	135.4	301.7
112	6.2099@-02	4.2123@-02	153.3	324.9	3.4651@-03	329.9	1.0396@+03	5.4641@+02	136.2	302.6
113	6.1032@-02	4.1153@-02	154.2	326.0	3.3977@-03	330.9	1.0225@+03	5.3452@+02	137.0	303.5
114	6.0013@-02	4.0232@-02	155.1	327.0	3.3335@-03	331.8	1.0063@+03	5.2324@+02	137.7	304.3
115	5.9040@-02	3.9358@-02	155.9	327.9	3.2723@-03	332.7	9.9071@+02	5.1254@+02	138.3	305.0
116	5.8111@-02	3.8529@-02	156.6	328.7	3.2141@-03	333.5	9.7590@+02	5.0240@+02	138.9	305.7
117	5.7225@-02	3.7741@-02	157.3	329.6	3.1586@-03	334.2	9.6179@+02	4.9278@+02	139.5	306.4
118	5.6380@-02	3.6993@-02	158.0	330.3	3.1059@-03	334.9	9.4835@+02	4.8366@+02	140.0	307.0
119	5.5573@-02	3.6283@-02	158.6	331.1	3.0556@-03	335.6	9.3554@+02	4.7500@+02	140.5	307.5
120	5.4804@-02	3.5608@-02	159.2	331.8	3.0077@-03	336.2	9.2333@+02	4.6679@+02	140.9	308.0
121	5.4069@-02	3.4966@-02	159.7	332.4	2.9621@-03	336.8	9.1170@+02	4.5900@+02	141.3	308.5
122	5.3367@-02	3.4355@-02	160.2	333.0	2.9186@-03	337.4	9.0061@+02	4.5159@+02	141.7	308.9
123	5.2697@-02	3.3773@-02	160.7	333.6	2.8771@-03	337.9	8.9004@+02	4.4456@+02	142.0	309.4
124	5.2056@-02	3.3219@-02	161.2	334.2	2.8374@-03	338.5	8.7995@+02	4.3786@+02	142.4	309.7
125	5.1443@-02	3.2691@-02	161.6	334.7	2.7996@-03	338.9	8.7032@+02	4.3150@+02	142.7	310.1
126	5.0856@-02	3.2186@-02	162.1	335.2	2.7634@-03	339.4	8.6112@+02	4.2544@+02	142.9	310.4
127	5.0294@-02	3.1705@-02	162.5	335.7	2.7288@-03	339.9	8.5233@+02	4.1966@+02	143.2	310.8
128	4.9756@-02	3.1245@-02	162.8	336.2	2.6957@-03	340.3	8.4393@+02	4.1415@+02	143.5	311.1
129	4.9240@-02	3.0806@-02	163.2	336.6	2.6640@-03	340.7	8.3590@+02	4.0890@+02	143.7	311.3
130	4.8745@-02	3.0385@-02	163.5	337.1	2.6336@-03	341.1	8.2822@+02	4.0389@+02	143.9	311.6
131	4.8269@-02	2.9982@-02	163.9	337.5	2.6045@-03	341.4	8.2086@+02	3.9911@+02	144.1	311.8
132	4.7813@-02	2.9596@-02	164.2	337.9	2.5766@-03	341.8	8.1381@+02	3.9453@+02	144.3	312.0
133	4.7374@-02	2.9226@-02	164.5	338.2	2.5497@-03	342.1	8.0706@+02	3.9016@+02	144.4	312.2
134	4.6953@-02	2.8871@-02	164.8	338.6	2.5240@-03	342.4	8.0059@+02	3.8598@+02	144.6	312.4
135	4.6547@-02	2.8530@-02	165.1	338.9	2.4992@-03	342.8	7.9438@+02	3.8198@+02	144.7	312.6
136	4.6157@-02	2.8203@-02	165.4	339.3	2.4754@-03	343.1	7.8843@+02	3.7815@+02	144.9	312.8
137	4.5780@-02	2.7888@-02	165.6	339.6	2.4524@-03	343.3	7.8271@+02	3.7448@+02	145.0	312.9
138	4.5418@-02	2.7585@-02	165.9	339.9	2.4304@-03	343.6	7.7722@+02	3.7096@+02	145.1	313.1

TABLE B-III. - ROTOR DISPLACEMENTS AND PHASE ANGLES

[Positions on shaft correspond to number of places selected on input data card 4.]

L= 3.0000@+01INCH	L1= 1.5000@+01INCH	L2= 1.5000@+01INCH	H1= 0.0000@+00INCH
H2= 0.0000@+00INCH	W= 1.1000@+02LB	WM1= 2.0000@-01LB	WM2= 2.0000@-01LB
K1X= 2.0000@+04LB/IN	K2X= 1.5000@+04LB/IN	K1Y= 1.6000@+04LB/IN	K2Y= 1.2000@+04LB/IN
C1X= 7.0000@+00LB.SEC/IN	C2X= 7.0000@+00LB.SEC/IN	C1Y= 7.0000@+00LB.SEC/IN	C2Y= 7.0000@+00LB.SEC/IN
R1X= 0.0000@+00LB/IN	R2X= 0.0000@+00LB/IN	R1Y= 0.0000@+00LB/IN	R2Y= 0.0000@+00LB/IN
D1X= 0.0000@+00LB.SEC/IN	D2X= 0.0000@+00LB.SEC/IN	D1Y= 0.0000@+00LB.SEC/IN	D2Y= 0.0000@+00LB.SEC/IN
IP= 5.7000@-01LB-IN-SEC2	IT= 2.1600@+01LB-IN-SEC2	R1= 2.0000@+00INCH	R2= 2.0000@+00INCH
	PHI= 0.0000@+00DEGREES		

LZ	XL	YL	PXL	PYL	SPEED
15.0	6.5582@-03	1.0779@-02	11.21	19.35	40.00
15.0	7.2599@-03	1.2296@-02	12.20	21.87	41.00
15.0	8.0601@-03	1.4113@-02	13.34	24.97	42.00
15.0	8.9792@-03	1.6305@-02	14.65	28.82	43.00
15.0	1.0044@-02	1.8951@-02	16.18	33.68	44.00
15.0	1.1287@-02	2.2112@-02	17.99	39.89	45.00
15.0	1.2753@-02	2.5755@-02	20.15	47.87	46.00
15.0	1.4498@-02	2.9613@-02	22.76	57.98	47.00
15.0	1.6593@-02	3.3008@-02	25.98	70.27	48.00
15.0	1.9123@-02	3.4955@-02	30.01	84.03	49.00
15.0	2.2170@-02	3.4839@-02	35.13	97.76	50.00
15.0	2.5785@-02	3.2990@-02	41.68	110.01	51.00
15.0	2.9893@-02	3.0288@-02	50.08	120.04	52.00
15.0	3.4116@-02	2.7457@-02	60.66	127.90	53.00
15.0	3.7604@-02	2.4860@-02	73.35	133.97	54.00
15.0	3.9269@-02	2.2603@-02	87.29	138.66	55.00
15.0	3.8612@-02	2.0681@-02	100.92	142.32	56.00
15.0	3.6194@-02	1.9048@-02	112.86	145.20	57.00
15.0	3.3018@-02	1.7658@-02	122.56	147.46	58.00
15.0	2.9817@-02	1.6468@-02	130.15	149.24	59.00
15.0	2.6925@-02	1.5444@-02	136.02	150.62	60.00
15.0	2.4432@-02	1.4557@-02	140.58	151.67	61.00
15.0	2.2318@-02	1.3782@-02	144.15	152.45	62.00
15.0	2.0530@-02	1.3101@-02	146.98	153.00	63.00
15.0	1.9011@-02	1.2501@-02	149.23	153.33	64.00
15.0	1.7713@-02	1.1969@-02	151.03	153.46	65.00
15.0	1.6594@-02	1.1497@-02	152.48	153.42	66.00
15.0	1.5621@-02	1.1078@-02	153.65	153.20	67.00
15.0	1.4769@-02	1.0707@-02	154.56	152.81	68.00
15.0	1.4016@-02	1.0382@-02	155.28	152.26	69.00
15.0	1.3345@-02	1.0100@-02	155.81	151.54	70.00
15.0	1.2745@-02	9.8616@-03	156.19	150.67	71.00
15.0	1.2203@-02	9.6667@-03	156.41	149.66	72.00
15.0	1.1712@-02	9.5174@-03	156.50	148.52	73.00
15.0	1.1265@-02	9.4162@-03	156.45	147.27	74.00
15.0	1.0856@-02	9.3663@-03	156.26	145.95	75.00
15.0	1.0480@-02	9.3708@-03	155.94	144.60	76.00
15.0	1.0134@-02	9.4325@-03	155.47	143.29	77.00
15.0	9.8169@-03	9.5528@-03	154.84	142.09	78.00
15.0	9.5269@-03	9.7311@-03	154.04	141.06	79.00

15.0	9.2645@-03	9.9635@-03	153.05	140.29	80.00
15.0	9.0320@-03	1.0243@-02	151.84	139.84	81.00
15.0	8.8336@-03	1.0557@-02	150.42	139.75	82.00
15.0	8.6757@-03	1.0891@-02	148.75	140.04	83.00
15.0	8.5667@-03	1.1228@-02	146.86	140.71	84.00
15.0	8.5167@-03	1.1550@-02	144.79	141.72	85.00
15.0	8.5361@-03	1.1840@-02	142.60	143.02	86.00
15.0	8.6338@-03	1.2086@-02	140.40	144.52	87.00
15.0	8.8156@-03	1.2280@-02	138.32	146.16	88.00
15.0	9.0826@-03	1.2420@-02	136.51	147.87	89.00
15.0	9.4300@-03	1.2509@-02	135.09	149.58	90.00
15.0	9.8468@-03	1.2551@-02	134.16	151.24	91.00
15.0	1.0317@-02	1.2556@-02	133.78	152.83	92.00
15.0	1.0819@-02	1.2531@-02	133.96	154.33	93.00
15.0	1.1330@-02	1.2484@-02	134.68	155.74	94.00
15.0	1.1824@-02	1.2420@-02	135.87	157.05	95.00
15.0	1.2281@-02	1.2345@-02	137.46	158.28	96.00
15.0	1.2680@-02	1.2260@-02	139.35	159.44	97.00
15.0	1.3008@-02	1.2168@-02	141.44	160.53	98.00
15.0	1.3260@-02	1.2069@-02	143.65	161.56	99.00
15.0	1.3435@-02	1.1966@-02	145.89	162.53	100.00
15.0	1.3538@-02	1.1859@-02	148.09	163.44	101.00
15.0	1.3576@-02	1.1748@-02	150.22	164.29	102.00
15.0	1.3561@-02	1.1635@-02	152.23	165.09	103.00
15.0	1.3502@-02	1.1521@-02	154.12	165.83	104.00
15.0	1.3409@-02	1.1408@-02	155.86	166.52	105.00
15.0	1.3292@-02	1.1294@-02	157.46	167.16	106.00
15.0	1.3156@-02	1.1183@-02	158.93	167.74	107.00
15.0	1.3009@-02	1.1074@-02	160.27	168.29	108.00
15.0	1.2855@-02	1.0967@-02	161.48	168.79	109.00
15.0	1.2697@-02	1.0864@-02	162.58	169.25	110.00
15.0	1.2539@-02	1.0764@-02	163.59	169.68	111.00
15.0	1.2383@-02	1.0667@-02	164.50	170.08	112.00
15.0	1.2229@-02	1.0574@-02	165.33	170.44	113.00
15.0	1.2079@-02	1.0484@-02	166.08	170.78	114.00
15.0	1.1934@-02	1.0397@-02	166.77	171.10	115.00
15.0	1.1793@-02	1.0314@-02	167.40	171.39	116.00
15.0	1.1658@-02	1.0234@-02	167.98	171.67	117.00
15.0	1.1528@-02	1.0157@-02	168.51	171.92	118.00
15.0	1.1403@-02	1.0083@-02	169.00	172.17	119.00
15.0	1.1283@-02	1.0012@-02	169.44	172.39	120.00
15.0	1.1169@-02	9.9438@-03	169.86	172.60	121.00
15.0	1.1059@-02	9.8782@-03	170.24	172.80	122.00
15.0	1.0954@-02	9.8151@-03	170.60	172.99	123.00
15.0	1.0853@-02	9.7545@-03	170.93	173.16	124.00
15.0	1.0757@-02	9.6961@-03	171.23	173.33	125.00
15.0	1.0665@-02	9.6399@-03	171.52	173.49	126.00
15.0	1.0576@-02	9.5858@-03	171.79	173.64	127.00
15.0	1.0492@-02	9.5337@-03	172.04	173.78	128.00
15.0	1.0410@-02	9.4835@-03	172.27	173.91	129.00
15.0	1.0332@-C2	9.4350@-03	172.49	174.04	130.00
15.0	1.0258@-02	9.3883@-03	172.70	174.16	131.00
15.0	1.0186@-02	9.3432@-03	172.90	174.28	132.00
15.0	1.0117@-02	9.2996@-03	173.08	174.39	133.00
15.0	1.0051@-02	9.2576@-03	173.25	174.50	134.00
15.0	9.9867@-03	9.2169@-03	173.42	174.60	135.00
15.0	9.9253@-03	9.1776@-03	173.57	174.70	136.00
15.0	9.8662@-03	9.1395@-03	173.72	174.79	137.00
15.0	9.8092@-03	9.1027@-03	173.86	174.88	138.00

TABLE B-III. - Concluded. ROTOR DISPLACEMENTS AND PHASE ANGLES

[Positions on shaft correspond to number of places selected on input data card 4.]

L= 3.0000@+01INCH	L1= 1.5000@+01INCH	L2= 1.5000@+01INCH	H1= 0.0000@+00INCH
H2= 0.0000@+00INCH	W= 1.1000@+02LB	WM1= 2.0000@-01LB	WM2= 2.0000@-01LB
K1X= 2.0000@+04LB/IN	K2X= 1.5000@+04LB/IN	K1Y= 1.6000@+04LB/IN	K2Y= 1.2000@+04LB/IN
C1X= 7.0000@+00LB.SEC/IN	C2X= 7.0000@+00LB.SEC/IN	C1Y= 7.0000@+00LB.SEC/IN	C2Y= 7.0000@+00LB.SEC/IN
R1X= 0.0000@+00LB/IN	R2X= 0.0000@+00LB/IN	R1Y= 0.0000@+00LB/IN	R2Y= 0.0000@+00LB/IN
D1X= 0.0000@+00LB.SEC/IN	D2X= 0.0000@+00LB.SEC/IN	D1Y= 0.0000@+00LB.SEC/IN	D2Y= 0.0000@+00LB.SEC/IN
IP= 5.7000@-01LB-IN-SEC2	IT= 2.1600@+01LB-IN-SEC2	R1= 2.0000@+00INCH	R2= 2.0000@+00INCH
	PHI= 0.0000@+00DEGREES		

LZ	XL	YL	PXL	PYL	SPEED
-15.0	1.3261@-02	1.8680@-02	7.16	11.04	40.00
-15.0	1.4247@-02	2.0347@-02	7.60	12.19	41.00
-15.0	1.5316@-02	2.2212@-02	8.12	13.65	42.00
-15.0	1.6481@-02	2.4304@-02	8.71	15.53	43.00
-15.0	1.7757@-02	2.6638@-02	9.42	18.00	44.00
-15.0	1.9162@-02	2.9182@-02	10.26	21.28	45.00
-15.0	2.0718@-02	3.1797@-02	11.30	25.65	46.00
-15.0	2.2449@-02	3.4120@-02	12.58	31.28	47.00
-15.0	2.4381@-02	3.5511@-02	14.21	38.02	48.00
-15.0	2.6540@-02	3.5311@-02	16.29	44.95	49.00
-15.0	2.8938@-02	3.3500@-02	19.02	50.47	50.00
-15.0	3.1527@-02	3.0933@-02	22.66	53.21	51.00
-15.0	3.4116@-02	2.8654@-02	27.49	52.95	52.00
-15.0	3.6246@-02	2.7240@-02	33.69	50.50	53.00
-15.0	3.7176@-02	2.6770@-02	41.00	47.05	54.00
-15.0	3.6257@-02	2.7088@-02	48.33	43.49	55.00
-15.0	3.3654@-02	2.8011@-02	53.96	40.28	56.00
-15.0	3.0397@-02	2.9403@-02	56.54	37.60	57.00
-15.0	2.7566@-02	3.1163@-02	55.85	35.49	58.00
-15.0	2.5714@-02	3.3218@-02	52.72	33.94	59.00
-15.0	2.4908@-02	3.5518@-02	48.38	32.86	60.00
-15.0	2.4980@-02	3.8033@-02	43.90	32.18	61.00
-15.0	2.5707@-02	4.0751@-02	39.88	31.84	62.00
-15.0	2.6901@-02	4.3671@-02	36.56	31.78	63.00
-15.0	2.8427@-02	4.6797@-02	33.93	31.96	64.00
-15.0	3.0200@-02	5.0139@-02	31.92	32.35	65.00
-15.0	3.2168@-02	5.3713@-02	30.41	32.94	66.00
-15.0	3.4302@-02	5.7533@-02	29.31	33.71	67.00
-15.0	3.6590@-02	6.1619@-02	28.54	34.66	68.00
-15.0	3.9028@-02	6.5990@-02	28.04	35.79	69.00
-15.0	4.1618@-02	7.0666@-02	27.75	37.09	70.00
-15.0	4.4369@-02	7.5665@-02	27.65	38.58	71.00
-15.0	4.7294@-02	8.1005@-02	27.70	40.26	72.00
-15.0	5.0408@-02	8.6697@-02	27.90	42.14	73.00
-15.0	5.3732@-02	9.2747@-02	28.21	44.23	74.00
-15.0	5.7289@-02	9.9151@-02	28.65	46.56	75.00
-15.0	6.1108@-02	1.0589@-01	29.20	49.13	76.00
-15.0	6.5222@-02	1.1291@-01	29.86	51.96	77.00
-15.0	6.9671@-02	1.2017@-01	30.64	55.06	78.00

-15.0	7.4499@-02	1.2754@-01	31.55	58.44	79.00
-15.0	7.9755@-02	1.3490@-01	32.61	62.09	80.00
-15.0	8.5488@-02	1.4207@-01	33.84	66.01	81.00
-15.0	9.1747@-02	1.4884@-01	35.26	70.16	82.00
-15.0	9.8572@-02	1.5500@-01	36.92	74.52	83.00
-15.0	1.0599@-01	1.6031@-01	38.83	79.03	84.00
-15.0	1.1400@-01	1.6460@-01	41.04	83.63	85.00
-15.0	1.2257@-01	1.6774@-01	43.56	88.23	86.00
-15.0	1.3163@-01	1.6969@-01	46.42	92.76	87.00
-15.0	1.4109@-01	1.7052@-01	49.62	97.16	88.00
-15.0	1.5080@-01	1.7033@-01	53.16	101.37	89.00
-15.0	1.6057@-01	1.6931@-01	57.03	105.33	90.00
-15.0	1.7018@-01	1.6766@-01	61.22	109.05	91.00
-15.0	1.7936@-01	1.6558@-01	65.70	112.50	92.00
-15.0	1.8783@-01	1.6324@-01	70.43	115.72	93.00
-15.0	1.9528@-01	1.6076@-01	75.35	118.72	94.00
-15.0	2.0145@-01	1.5822@-01	80.42	121.54	95.00
-15.0	2.0613@-01	1.5565@-01	85.54	124.20	96.00
-15.0	2.0920@-01	1.5306@-01	90.66	126.73	97.00
-15.0	2.1063@-01	1.5044@-01	95.70	129.15	98.00
-15.0	2.1051@-01	1.4777@-01	100.58	131.46	99.00
-15.0	2.0901@-01	1.4506@-01	105.26	133.66	100.00
-15.0	2.0634@-01	1.4230@-01	109.69	135.75	101.00
-15.0	2.0274@-01	1.3952@-01	113.85	137.73	102.00
-15.0	1.9847@-01	1.3672@-01	117.73	139.60	103.00
-15.0	1.9372@-01	1.3393@-01	121.32	141.36	104.00
-15.0	1.8870@-01	1.3118@-01	124.62	143.00	105.00
-15.0	1.8355@-01	1.2848@-01	127.67	144.53	106.00
-15.0	1.7838@-01	1.2585@-01	130.46	145.96	107.00
-15.0	1.7329@-01	1.2330@-01	133.02	147.29	108.00
-15.0	1.6833@-01	1.2083@-01	135.37	148.53	109.00
-15.0	1.6354@-01	1.1847@-01	137.52	149.69	110.00
-15.0	1.5895@-01	1.1620@-01	139.50	150.77	111.00
-15.0	1.5457@-01	1.1403@-01	141.33	151.77	112.00
-15.0	1.5041@-01	1.1195@-01	143.00	152.71	113.00
-15.0	1.4646@-01	1.0997@-01	144.55	153.59	114.00
-15.0	1.4273@-01	1.0808@-01	145.98	154.41	115.00
-15.0	1.3920@-01	1.0628@-01	147.31	155.18	116.00
-15.0	1.3586@-01	1.0457@-01	148.54	155.91	117.00
-15.0	1.3271@-01	1.0293@-01	149.68	156.59	118.00
-15.0	1.2974@-01	1.0137@-01	150.74	157.23	119.00
-15.0	1.2693@-01	9.9887@-02	151.74	157.84	120.00
-15.0	1.2427@-01	9.8470@-02	152.67	158.42	121.00
-15.0	1.2176@-01	9.7117@-02	153.53	158.96	122.00
-15.0	1.1938@-01	9.5825@-02	154.35	159.47	123.00
-15.0	1.1713@-01	9.4591@-02	155.12	159.96	124.00
-15.0	1.1500@-01	9.3411@-02	155.84	160.42	125.00
-15.0	1.1297@-01	9.2283@-02	156.52	160.87	126.00
-15.0	1.1105@-01	9.1203@-02	157.16	161.29	127.00
-15.0	1.0923@-01	9.0169@-02	157.76	161.69	128.00
-15.0	1.0750@-01	8.9179@-02	158.34	162.07	129.00
-15.0	1.0585@-01	8.8229@-02	158.88	162.43	130.00
-15.0	1.0428@-01	8.7317@-02	159.39	162.78	131.00
-15.0	1.0278@-01	8.6442@-02	159.88	163.12	132.00
-15.0	1.0135@-01	8.5602@-02	160.35	163.44	133.00
-15.0	9.9987@-02	8.4795@-02	160.79	163.75	134.00
-15.0	9.8685@-02	8.4018@-02	161.22	164.04	135.00
-15.0	9.7439@-02	8.3271@-02	161.62	164.33	136.00
-15.0	9.6248@-02	8.2551@-02	162.01	164.60	137.00
-15.0	9.5106@-02	8.1858@-02	162.37	164.86	138.00

APPENDIX C

LISTING AND SAMPLE OUTPUT OF COMPUTER PROGRAM ROTOR4M

```
BEGIN
     COMMENT   THIS  PROGRAM EVALUATES DESIGN DATA FOR A FOUR DEGREE
 FREEDOM SYSTEM THAT SIMULATES A ROTOR ON GENERAL ANISOTROPIC BRGS.
     THE EQUATIONS SOLVED HAD BEEN LINEARIZED . NO ASSUMPTIONS WERE
MADE ON THE BEARING CHARACTERISTICS . THE CROSS COUPLING TERMS  ARE
KEPT WITH PROPER SUBSCRIPTS AS USED IN THE DERIVATION OF THE EQNS.
     THE  PROGRAM REQUIRES THE FOLLOWING TO BE READ AS INPUT DATA:
CARD 0
      SPEC-ALLOWABLE PERCENT ERROR  ON SPEED
CARD 1
     1. WO- INITIAL SPEED (RPS)
     2. DW- INCREMENT IN SPEED (RPS)
     3.WM- FINAL SPEED (RPS)
CARD 2
     1. L- LENGTH BETN BRGS (INCH)
     2.L1- DIST FROM 1ST BRG TO MASS CENTER (INCH)
     3.L2- DIST FROM 2ND BRG TO MASS CENTER (INCH)
     4.W- ROTOR WEIGHT (LBS)
     5. IP- POLAR M.I. (LB-IN-SEC2)
     6. IT-TRANSVERSE M.I.  OF ROTOR ABOUT MASS CENTER (LB-IN-SEC2)
CARD 3
     1. WM1-FIRST UNBALANCE WEIGHT (LBS)
     2. WM2- SECOND UNBALANCE WEIGHT (LBS)
     3. H1- DIST FROM 1ST BRG TO 1ST UNBALANCE (INCH)
     4. H2- DIST FROM 1ST BRG TO 2ND UNBALANCE (INCH)
     5. PHI- PHASE ANGLES BETN UNBALANCE PLANES
     6. R1- RADIUS OF 1ST UNBALANCE LOCATION
     7. R2- RADIUS OF 2ND UNBALANCE LOCATION
CARD 4
     1. P- NO. OF PLACES OTHER THAN THE BRG LOCATIONS WHERE
     DISPLACEMENTS ARE TO BE MEASURED
     2.LZ1- DIST FROM 1ST BRG TO 1ST PROBE (INCH)
     3. LZ2- DIST FROM 1ST BRG TO 2ND PROBE  (INCH)
CARD 5
     1. K1X- 1ST BRG STIFFNESS IN X DIRECTION (LB/IN)
     2. K2Y- 2ND BRG STIFFNESS IN X DIRECTION (LB/IN)
     3. K1Y- 1ST BRG STIFFNESS IN Y DIRECTION (LB/IN)
     4. K2Y- 2ND BRG STIFFNESS IN Y DIRECTION(LB/IN)
CARD 6
     1. C1X-1ST BRG DAMPING COEFF IN X DIRECTION(LB.SEC/IN)
     2. C2X- 2ND BRG DAMPING COEFF IN X DIRECTION (LB.SEC/IN)
     3.C1Y-1ST BRG DAMPING COEFF IN Y DIRECTION (LB.SEC/IN)
     4. C2Y- 2ND BRG DAMPING COEFF IN Y DIRECTION (LB.SEC/IN)
CARD 7
     1. D1X- CROSS COUPLING DAMPING COEFF  (LB.SEC/IN)
     2. D2X- CROSS COUPLING DAMPING COEFF  (LB.SEC/IN)
     3. D1Y- CROSS COUPLING DAMPING COEFF  (LB.SEC/IN)
     4. D2Y- CROSS COUPLING DAMPING COEFF  (LB.SEC/IN)
CARD 8
     1. R1X- CROSS COUPLING STIFFNESS (LB/IN)
     2. R2X- CROSS COUPLING STIFFNESS (LB/IN)
     3. R1Y- CROSS COUPLING STIFFNESS (LB/IN)
     4. R2Y- CROSS COUPLING STIFFNESS (LB/IN)
     THE HEADING PRINT OUT OF THE INPUT DATA ARE AS FOLLOWS :
CARD 9
```

```
     CONTROL-IDENTIFIER CONTROLLING THE SYMMETRY OF BEARING
LINE1     L,L1,L2,H1
LINE2     H2,N,WM1,WM2
LINE3     K1X,K2X,K1Y,K2Y
LINE4     C1X,C2X,C1Y,C2Y
LINE5     R1X,R2X,R1Y,R2Y
LINE6     D1X,D2X,D1Y,D2Y
LINE7     IP,IT,R1,R2
LINE8     PHI       ;

COMMENT
    THIS PROGRAM FINDS THE CRITICAL SPEED AND THE CORRESPONDING
AMPLITUDES ALONG WITH THE PHASE ANGLES , FORCE TRANSMITTED ETC.
    IN ORDER TO GET SENSIBLE RESULTS WHICH MAY KEEP THE ALLOWABLE
PERCENT ERROR WITHIN 1% ON THE CRITICAL SPEED THUS FOUND , THE DAMPING
CHARACTERISTICS OF THE BEARING SHOULD NOT BE TOO SMALL .
    THE OUTPUT DATA ARE AS FOLLOWS:
COL1:  SPEED (RPS)
COL2:  COORDINATE
COL3:  AMPLITUDE (IN)
COL4:  PHASE ANGLE OF THE AMPLITUDE WRT UNBALANCE
COL5:  MAJOR SEMI AXIS/AMPLITUDE OF COORDINATE (DIM)
COL6:  MINOR SEMI AXIS/AMPLITUDE OF COORDINATE (DIM)
COL7:  ELLIPSE ANGLE OF MAJOR SEMI AXIS WITH X AXIS
COL8:  BEARING LOCATION OF MAXIMUM FORCE TRANSMITTED
COL9:  MAX FORCE TRANSMITTED
COL10:  PHASE ANGLE OF MAX FORCE WRT UNBALANCE FORCE
COL11:  PERCENT  CYLINDRICAL MODE

     IF CONTROL=0 THEN WE ARE DEALING WITH A SYMMETRIC BEARING CASE ;

     REAL T, IOTIME, PTIME  ;
REAL WN ,WD  , DW , WM , L , L1 , L2 , W , IP , IT , WM1 , WM2 , H1 ,
H2 , PH , R1 , R2 , K1X , K2X , K1Y , K2Y , C1X , C2X , C1Y , C2Y ,
 D1X , D2X , D1Y , D2Y , R1X , R2X , R1Y , R2Y , G , PI , M, DM1 , DM2 ,
 RPP , RTT , RP , RT , L11 , L22 , R01 , R02 , PHI , EPS , RAD ,
 K1XX , K2XX , K1YY , K2YY , C1XX , C2XX , C1YY , C2YY , R1XX , R2XX ,
 R1YY , R2YY , D1XX , D2XX , D1YY , D2YY , PN1 , PD1 , SI , PN2 , PD2 ,
 SIT , BB , CC , DD , EE , PX1 , PX2 , PY1 , PY2 , PA1 , PA2   ,
 PFX , PFY , E , SPEC , AX , BX , AY , BY , THETA , MA , MI ,
PERCENT ;
INTEGER P,N,I,K1,J,K,IM,NOL,PP,CONTROL,LON;
REAL ARRAY    OMEGA , S , SS , XX1 , XX2 , YY1 , YY2 ,
 PFX2 , PFY2 [0:500] , LZ[0:4] ,
  XL , YL , PXL , PYL[0:500,0:4] ,A[0:8,0:8]  , C , X[0:8]  ,
PFX1 , PFY1[0:500], CV[0:12] , AA[0:12,0:500] , PA, FPA , AAX , ABX ,
AAY  , ABY  ,   FA  , COOR[0:12];
LABEL    LOO , FINIS , E1 , E2 , LOO1  , DOITAGIN ;
BOOLEAN  RSW   ;
FORMAT  HEAD1 (6(2(59("*")),/),
                                     24("*"), X40 , X31 ,23("*"),/,
24("*"), X1 , "DESIGN DATA FOR A SINGLE MASS ROTOR WITH FLEXIBLE SUPPORT
 AND DAMPING" ,X1 , 23 ("*"),/,  6(2(59("*")), /))  ;
FORMAT  HEAD2(2(2(59("*")),/),
X5 , "L=",E11.4 , "INCH" , X12 , "L1=" , E11.4 ,"INCH" , X12 , "L2=" ,
```

```
 E11.4 , "INCH" , X12 , "H1=" , E11.4 , "INCH" , / ,
X4 , "H2=" , E11.4 , "INCH" , X13 , "W=" , E11.4 , "LB" , X13 , "WM1=" ,
 E11.4 , "LB" , X13 , "WM2=" , E11.4 ,"LB" , / ,
 X3 , "K1X=" , E11.4 , "LB/IN" , X10 , "K2X=" , E11.4 ,"LB/IN" , X10 ,
 "K1Y=" , E11.4 , "LB/IN" , X10 , "K2Y=" , E11.4 , "LB/IN" , / ,
 X3 , "C1X=" , E11.4 , "LB.SEC/IN" , X6 , "C2X=" , E11.4 , "LB.SEC/IN" ,
 X6 , "C1Y=" , E11.4 , "LB.SEC/IN" ,X6 , "C2Y=" , E11.4 ,"LB.SEC/IN" ,/,
 X3 ,"R1X=" , E11.4 , "LB/IN" , X10 , "R2X=" , E11.4 , "LB/IN" , X10 ,
 "R1Y=" , E11.4 , "LB/IN" , X10 ,"R2Y=" , E11.4 , "LB/IN" , / ,
 X3 , "D1X=" , E11.4 , "LB.SEC/IN" , X6 , "D2X=" , E11.4 , "LB.SEC/IN" ,
 X6 , "D1Y=" , E11.4 , "LB.SEC/IN" , X6 , "D2Y=" , E11.4 , "LB.SEC/IN",/
,X4,"IP=" , E11.4 , "LB-IN-SEC2" , X6 , "IT=" , E11.4 , "LB-IN-SEC2" ,
 X6 , "R1=" , E11.4 , "INCH" , X12 , "R2=" , E11.4,"INCH" ,/ ,
 X30 , "PHI=" , E11.4 , "DEGREES" , / ,
                                        2(2(59("*")),/))    ;
FORMAT OUT1( X1 , "SPEED" , X3 , "COORDINATE" , X2 , "AMPLITUDE" , X2 ,
"PHASE" , X2 , "MAJOR SEMI" , X2 , "MINOR SEMI" , X2 , "ELLIPSE" , X2 ,
      "BEARING" , X3 , "BEARING" , X2 , "FORCE PHASE" , X4 , "PERCENT" ,
 / , "REV/SEC" , X16 , "(IN)" , X5 , "ANGLE" , X2,"AXIS (DIM)",X2,"AXIS
(DIM)",
X3,"ANGLE" , X3 , "LOCATION" , X3 ,"FORCE" , X6 , "ANGLE" , X5 ,
 "CYLINDRICAL" ) ;
FORMAT OUT2A ( F7.1 , X6 , "X" , I1 , X5 , E10.3 , X1 , F6.1 , X4 ,
 F5.2 , X7 , F5.2 , X5 , F6.1 , X6 , I2 , X4 , E10.3 , X2 , F6.1 , X7 ,
 F6.1 ) ;
FORMAT OUT2B ( F7.1 , X6 , "Y" , I1 , X5 , E10.3 , X1 , F6.1 , X4 ,
 F5.2 , X7 , F5.2 , X5 , F6.1 , X6 , I2 , X4 , E10.3 , X2 , F6.1 , X7 ,
 F6.1 ) ;
FORMAT OUT2C ( F7.1 , X3 , "X(" ,    F5.1 , ")" , X2 , E10.3 , X1 ,
 F6.1 , X4 , F5.2 , X7 , F5.2 , X5 , F6.1 , X6 , I2 , X4 , E10.3 ,
 X2 , F6.1 , X7 , F6.1 ) ;
FORMAT OUT2D ( F7.1 , X3 , "Y(" ,    F5.1 , ")" , X2 , E10.3 , X1 ,
 F6.1 , X4 , F5.2 , X7 , F5.2 , X5 , F6.1 , X6 , I2 , X4 , E10.3 ,
 X2 , F6.1 , X7 , F6.1 ) ;

REAL  PROCEDURE  ANGLE(PN,PD)  ;
VALUE  PN , PD ;       REAL  PN, PD ;
BEGIN
REAL  B , PI  ;
LABEL  L1 , L2 , L3 , L4   ;
PI ← 3.14159  ;
IF  PN> 0 AND  PD=0  THEN  GO  TO L1  ;
IF  PN<0 AND  PD=0  THEN  GO  TO L2  ;
IF  PN=0  AND  PD=0  THEN  GO  TO  L3 ;
B ← ARCTAN(ABS(PN/PD)) ;
IF  PN<0  AND  PD>0  THEN  B ← 2× PI - B  ;
IF  PN >0  AND PD<0  THEN B← PI - B  ;
IF PN<0  AND  PD<0  THEN  B← PI+B   ;
GO  TO  L4  ;
L1:  B←PI/2  ;
GO  TO  L4  ;
L2:   B← (3×PI) /2  ;
 GO  TO  L4  ;
L3:  B←0  ;
L4:   ANGLE←B  ;
END  OF  PROCEDURE  ;
```

```
PROCEDURE  FORCE(C , K,D,R,C1,S1,C2,S2,WW,F,PFX,PFY)   ;
VALUE  C , K , D , R , C1, S1 , C2 , S2 , WW ;
REAL   C , K , D , R ,   C1 , S1 , C2 , S2 , WW , F , PFX , PFY   ;
COMMENT  THIS  PROCEDURE  CALCULATES  THE  FORCE OR  MOMENT
 PRODUCED  BY  THE  REACTIONS  WHERE
C= DAMPING COEFF   D= CROSS COUPLING  DAMPING
K= STIFFNESS  COEFF  R= CROSS  COUPLING  STIFFNESS
THE  FORCE  CALCULATED  IS  IN THE  DIRECTION  OF  X1  WHERE
X1= C1 COS(WWT) + S1 SIN(WWT) ,WHERE  WW=ROTOR  SPEED  IN  RAD/SEC
DIRECTION  NORMAL  TO  X1  IS  X2  WHERE
X2= C2 COS (WWT) + S2 SIN(WWT)
F=F COS(WWT-PH)=A COS(WWT) + B SIN(WWT)  ;
BEGIN
REAL  A, B  ;
A ← C × WW × S1 + K × C1 + D × WW × S2 + R× C2  ;
B ←-WW × C × C1 + K × S1 - WW × D × C2  + R × S2  ;
F ← SQRT ( A× A + B× B )  ;
PFX ← ANGLE ( B , A) ;    PFY ← ANGLE (-A , B )  ;
END  OF  PROCEDURE  FORCE  ;

PROCEDURE  ARBITRARYDISPLACEMENT (LZ , L , X , XL , YL , PXL , PYL )  ;
VALUE  LZ , L  ;
REAL  LZ , L , XL , YL , PXL , PYL  ;
REAL  ARRAY  X[0]  ;
BEGIN
COMMENT  THIS  PROCEDURE  CALCULATES  THE  X AND  Y  DISPLACEMENTS AT
ANY  POINT  MEASURED  FROM  THE  FIRST  BRG .  XL  IS  SHAFT ABSOLUTE
X DISPLACEMENT AND  PXL IS THE PHASE  ANGLE  ;
REAL   Z   ;
Z ← LZ/L  ;
AX ← Z× X[3] + (1 - Z ) × X[1]  ;
BX ← Z× X[4] + (1 - Z ) × X[2] ;
 AY ← Z × X[7] + ( 1 - Z ) × X[5] ;
 BY ← Z × X[8] + ( 1 - Z ) × X[6]  ;
 XL ← SQRT ( AX× AX + BX× BX ) ;
YL ← SQRT ( AY × AY + BY × BY )  ;
PXL ← ANGLE (BX , AX ) ;
PYL ← ANGLE ( - AY , BY )  ;
END  OF  PROCEDURE  ARBITRARYDISPLACEMENT  ;

PROCEDURE PERCYL (A,B,C,D,PERCENT) ;
      VALUE A,B,C,D;
      REAL A,B,C,D,PERCENT ;
BEGIN
      REAL XX1,XX2,U ;
      XX1← SQRT(A×A+B×B) ;
      XX2←SQRT(C×C+D×D) ;
     U←SQRT((A+C)*2+(B+D)*2) ;
      IF XX1>XX2 THEN
      PERCENT←U/(2×XX1)×100
      ELSE
      PERCENT←U/(2×XX2)×100   ;
END OF PROCEDURE PERCYL ;

PROCEDURE ELLIPSE(A,B,C,D,MA,MI,THETA);
```

```
      VALUE A,B,C,D;
      REAL A,B,C,D,MA,MI,THETA;
BEGIN
      REAL U,V,W;
      LABEL  FIN  ;
      U←A×A+B×B+C×C+D×D;
      V←4×(A×D-B×C)*2;
      W ← SQRT(ABS(U×U-V)) /2    ;
      MA ← SQRT(ABS(U/2+W)) ;
      MI ← SQRT(ABS(U/2-W));
      IF (MA-MI)/MA≤.01 THEN
BEGIN
      THETA←0;    GO TO FIN ;
END
      ELSE
BEGIN
      U←2×(A×C+B×D);
      V←A×A+B×B-C×C-D×D;
      THETA←ANGLE(U,V)×90/3.14159;
END;
FIN:  END OF PROCEDURE ELLIPSE ;

PROCEDURE SOLVE(N,A,C,RSW,E,K1,EPS,X,E1,E2);VALUE N,RSW,E,K1,EPS;INTEGER
 N,K1;REAL E,EPS;BOOLEAN RSW;REAL ARRAY A[0,0],C,X[0];LABEL E1,E2;BEGIN
INTEGER I,J,K,J1,K2,L;REAL BIG,TEMP,DIAG,NORM,Q;OWN INTEGER ARRAY F[0:N]
;REAL ARRAY D[0:N];OWN REAL ARRAY B[0:N,0:N];LABEL S1,S2,S3,S4,S5,S6,REP
,S7,S8,S9,IT1,S10,S11,S12,S13,S14,S15,EXIT;S1:IF RSW THEN GO TO REP;FOR
I←1STEP 1UNTIL N DO FOR J←1STEP 1UNTIL N DO B[I,J]←A[I,J];S2:FOR I←1STEP
 1UNTIL N DO BEGIN L←I-1;FOR J←I STEP 1UNTIL N DO BEGIN Q←0;FOR K←1STEP
1UNTIL L DO Q←B[J,K]×B[K,I]+Q;B[J,I]←B[J,I]-Q END;BIG←0;K2←I;S3:FOR K←I
STEP 1UNTIL N DO BEGIN IF ABS(B[K,I])>BIG THEN BEGIN BIG←ABS(B[K.I]);K2←
K END END;S4:IF BIG≤EPS THEN GO TO E1;F[I]←K2:IF K2≠I THEN S5:FOR K←1STE
P 1UNTIL N DO BEGIN TEMP←A[K2,K];A[K2,K]←A[I,K];A[I,K]←TEMP;TEMP←B[K2,K]
;B[K2,K]←B[I,K];B[I,K]←TEMP;END;DIAG←B[I,I];S6:FOR J←I+1STEP 1UNTIL N DO
 BEGIN Q←0;FOR K←1STEP 1UNTIL L DO Q←B[I,K]×B[K,J]+Q;B[I,J]←(B[I,J]-Q)/D
IAG END END;REP:FOR I←1STEP 1UNTIL N DO BEGIN TEMP←C[F[I]];C[F[I]]←C[I];
D[I]←C[I]←TEMP END;FOR I←1STEP 1UNTIL N DO BEGIN L←I-1;Q←0;S7:FOR K←1STE
P 1UNTIL L DO Q←B[I,K]×D[K]+Q;D[I]←(D[I]-Q)/B[I,I]END;S8:FOR I←N STEP-1U
NTIL 1DO BEGIN Q←0;FOR K←I+1STEP 1UNTIL N DO Q←B[I.K]×X[K]+Q;X[I]←D[I]-Q
 END;S9:IF E=0THEN GO TO EXIT;J1←0;IT1:IF J1≥K1 THEN GO TO E2;NORM←0;FOR
 I←1STEP 1UNTIL N DO BEGIN Q←0;L←I-1;S10:FOR K←1STEP 1UNTIL N DO Q←A[I,K
]×X[K]+Q;D[I]←C[I]-Q;S11:NORM←ABS(D[I])+NORM;Q←0;S12:FOR K←1STEP 1UNTIL
L DO Q←B[I,K]×D[K]+Q;D[I]←(D[I]-Q)/B[I,I]END;FOR I←N STEP-1UNTIL 1DO BEG
IN Q←0;S13:FOR K←I+1STEP 1UNTIL N DO Q←B[I,K]×D[K]+Q;X[I]←X[I]+D[I]-Q EN
D;S14:J1←J1+1;S15:IF N×E<NORM THEN GO TO IT1;EXIT:END;

PROCEDURE ICALCULATE(I,OMEG,IM,CGO);
      VALUE OMEG,I,IM,CGO;
      REAL OMEG;
      INTEGER I,IM,CGO;
BEGIN
COMMENT  THIS PROCEDURE CALCULATES THE AMPLITUDES AT BRG. LOCATIONS
UNTIL THE MAX AMPLITUDE IS REACHED AT CERTAIN SPEED ;
      OMEGA[I]←OMEG;
S[ I ] ← 2 × PI × OMEGA [I ]   ;
SS [ I ] ← S [ I ] × S [ I ]    ;
```

```
BEGIN
REAL  XXXX  ;
A[1,1] ← K1XX - L22 × SS [I]  ;
A[1,2] ← C1XX × S[I] ;
A[1,3] ← K2XX - L11 × SS[I]  ;
A[1,4] ← C2XX × S[I]  ;
A[1,5] ← R1YY  ;
A[1,6] ← D1YY × S[I]  ;
A[1,7] ← R2YY  ;
A[1,8] ← D2YY × S [I]  ;
A[2,1] ← - C1XX × S[I]  ;
A[2,2] ← K1XX - L22 × SS[I]  ;
A[2,3] ← - C2XX × S[I] ;
A[2,4] ← K2XX - L11 × SS[I]  ;
A[2,5] ← - D1YY × S[I]  ;
A[2,6] ← R1YY  ;
A[2,7] ← - D2YY × S[I]  ;
A[2,8] ← R2YY  ;
A[3,1] ← R1XX  ;
A[3,2] ← D1XX × S[I]  ;
A[3,3] ← R2XX  ;
A[3,4 ] ← D2XX × S[I]  ;
A[3,5] ← K1YY - L22 × SS[I]  ;
A[3,6] ← C1YY × S[I]  ;
 A[3,7] ← K2YY - L11 × SS[I]  ;
A[3,8] ← C2YY × S[I]  ;
A[4,1] ← - D1XX × S[I]  ;
A[4,2] ← R1XX  ;
A[4,3] ← - D2XX × S[I]  ;
A[4,4] ← R2XX  ;
A[4,5] ← - C1YY × S[I]  ;
A[4,6] ← K1YY - L22 ×SS[I]  ;
A[4,7] ← - C2YY × S[I]  ;
A[4,8] ← K2YY - L11 × SS[I] ;
A[5,1] ← RT × SS[I] - K1XX × L11  ;
A[5,2] ← - C1XX × L11 × S[I]  ;
A[5,3] ← - RT × SS[I] + K2XX × L22  ;
A[5,4] ← C2XX × L22 × S[I]  ;
A[5,5] ← -R1YY × L11  ;
A[5,6] ← - RP × SS[I] - L11 × S[I] ×D1YY  ;
A[5,7] ← R2YY × L22  ;
A[5,8] ← RP × SS[I] + D2YY × L22 × S[I]  ;
A[6,1] ← C1XX × L11 × S[I]  ;
A[6,2] ← RT × SS[I] - K1XX × L11 ;
A[6,3] ← - C2XX × L22 × S[I]  ;
A[6,4] ← - RT × SS[I] + K2XX × L22  ;
A[6,5] ← RP × SS[I] + D1YY × L11 × S[I]  ;
A[6,6] ← -R1YY × L11  ;
A[6,7] ← - RP × SS[I] - D2YY × L22 × S[I]  ;
A[6,8] ← R2YY × L22  ;
A[7,1] ← - R1XX × L11  ;
A[7,2] ← RP × SS[I] - D1XX × L11 × S[I]  ;
A[7,3] ← R2XX × L22  ;
A[7,4] ← - RP  ×  SS[I] + D2XX × L22 × S[I]  ;
A[7,5] ← RT × SS[I] - K1YY × L11  ;
A[7,6] ← - C1YY × L11 × S[I]  ;
```

```
A[7,7] ← -RT × SS[I] + K2YY × L22 ;
A[7,8] ← C2YY × L22 × S[I] ;
A[8,1] ← - RP × SS[I] + D1XX × L11 ×S[I] ;
A[8,2] ← -R1XX × L11 ;
A[8,3] ← RP × SS[I] - D2XX × L22 × S[I] ;
A[8,4] ← R2XX × L22 ;
A[8,5] ← C1YY × L11 × S[I] ;
A[8,6] ← RT × SS[I] - K1YY × L11 ;
A[8,7] ← - C2YY × L22 × S[I] ;
A[8,8] ← - RT × SS[I] + K2YY × L22 ;
C[1] ← (DM1 × SS[I] × R1 ) / M + ( DM2 × SS[I] × R2 × COS(PHI))/ M ;
C[2] ← - ( DM2 × SS[I] × R2 × SIN (PHI)) / M ;
C[3] ← ( DM2 × SS[I] × R2 × SIN (PHI)) / M ;
C[4] ← (DM1 × SS[I] × R1 ) / M + ( DM2 × SS[I] × R2 × COS(PHI))/ M ;
C[5] ← (DM1 × RO1 × SS[I] × R1 + DM2 × RO2 × SS[I]×R2 × COS(PHI)) /
 (M× L ) ;
C[6] ←- (DM2 × SS[I] × RO2 × R2 × SIN ( PHI )) / (M× L) ;
C[7] ← (DM2 × SS[I] × RO2 × R2 × SIN ( PHI )) / (M× L) ;
C[8] ← (DM1 × RO1 × SS[I] × R1 + DM2 × RO2 × SS[I]×R2 × COS(PHI)) /
 (M× L ) ;
END ;
SOLVE (N , A , C , RSW , E , K1 , EPS , X , E1 , E2 ) ;
BB ← X[1] × X[1] + X[2]× X[2] ;
XX1[I] ← SQRT (BB) ;
CC ← X[3] × X[3] + X[4] × X[4] ;
XX2[I] ← SQRT ( CC ) ;
DD ← X[5] × X[5] + X[6] × X[6] ;
YY1[I] ← SQRT ( DD ) ;
EE ← X[7] × X[7] + X[8] × X[8] ;
YY2[I] ← SQRT ( EE ) ;
      IF IM>4 THEN
FOR J←1 STEP 1 UNTIL P DO
BEGIN
ARBITRARYDISPLACEMENT ( LZ[J] , L , X , XL[I,J] , YL[I,J] , PXL[I,J] ,
 PYL[I,J] ) ;
      AA[2×J+3,I]←XL[I,J];
      AA[2×J+4,I]←YL[I,J];
      IF CGO=1 THEN
BEGIN
      AAX[2×J+3]←AAX[2×J+4]←AX;
      ABX[2×J+3]←ABX[2×J+4]←BX;
      AAY[2×J+3]←AAY[2×J+4]←AY;
      ABY[2×J+3]←ABY[2×J+4]←BY;
      PA[2×J+3]←PXL[I,J]×RAD;
      PA[2×J+4]←PYL[I,J]×RAD;
END;
END;
      AA[1,I]←XX1[I];
      AA[2,I]←YY1[I];
      AA[3,I]←XX2[I];
      AA[4,I]←YY2[I];
END OF PROCEDURE ICALCULATE;

PROCEDURE HELPME ( H , Q ,IM , I ) ;
      VALUE H , Q , IM , I ;
      INTEGER H , Q , IM , I ;
```

```
BEGIN
      INTEGER  LOC , J ;
      IF H = 1 THEN
      PERCYL(X[1],X[2],X[3],X[4],PERCENT)
      ELSE
      PERCYL(X[5],X[6],X[7],X[8],PERCENT) ;
      IF IM=1 OR IM=2 THEN
          ELLIPSE ( X[1] , X[2] , X[5] , X[6] , MA , MI , THETA )
      ELSE
      IF IM=3 OR IM=4 THEN
          ELLIPSE( X[3] , X[4] , X[7], X[8] , MA , MI , THETA )
      ELSE
          ELLIPSE ( AAX[IM],ABX[IM],AAY[IM],ABY[IM],MA,MI,THETA);
IF IM ≤ 4 THEN
BEGIN
      LOC←COOR[IM] ;
COMMENT FOLLOWING PRINTS THE RESULTS IN X DIRECTION AT BEARING;
      IF H=1 THEN
      WRITE( LP , OUT2A , OMEGA[1] , COOR[IM] , AA[IM,I], PA[IM],MA/AA[IM
,I],MI/AA[IM,I],THETA,LOC,FA[IM],FPA[IM],PERCENT)
          ELSE
COMMENT FOLLOWING PRINTS THE RESULTS IN Y DIRECTION AT BEARING;
      WRITE( LP , OUT2B , OMEGA[I] , COOR[IM] , AA[IM,I], PA[IM],MA/AA[IM
,I],MI/AA[IM,I],THETA,LOC,FA[IM],FPA[IM],PERCENT);
END
      ELSE
BEGIN
      IF FA[Q] > FA[H] THEN
BEGIN
      LOC←COOR[Q] ;
COMMENT FOLLOWING PRINTS THE RESULTS IN X DIRECTION AT ARBITRARY LOC ;
      IF H=1 THEN
          WRITE ( LP , OUT2C , OMEGA[I] , COOR[IM] , AA[IM , I ] ,
      PA[IM] ,MA/AA[IM,I],MI/AA[IM,I],THETA,LOC,FA[Q],FPA[Q],PERCENT)
      ELSE
COMMENT FOLLOWING PRINTS THE RESULTS IN Y DIRECTION AT ARBITRARY LOC ;
          WRITE ( LP , OUT2D , OMEGA[I] , COOR[IM] , AA[IM , I ] ,
      PA[IM] ,MA/AA[IM,I],MI/AA[IM,I],THETA,LOC,FA[Q],FPA[Q],PERCENT) ;
END
      ELSE
BEGIN
      LOC←COOR[H] ;
COMMENT FOLLOWING PRINTS THE RESULTS IN X DIRECTION AT ARBITRARY LOC ;
          IF H=1 THEN
          WRITE ( LP , OUT2C , OMEGA[I] , COOR[IM] , AA[IM , I ] ,
      PA[IM] ,MA/AA[IM,I],MI/AA[IM,I],THETA,LOC,FA[H],FPA[H],PERCENT)
      ELSE
COMMENT FOLLOWING PRINTS THE RESULTS IN Y DIRECTION AT ARBITRARY LOC ;
          WRITE ( LP , OUT2D , OMEGA[I] , COOR[IM] , AA[IM , I ] ,
      PA[IM] ,MA/AA[IM,I],MI/AA[IM,I],THETA,LOC,FA[H],FPA[H],PERCENT) ;
END ;
END;
END OF PROCEDURE HELPME:

PROCEDURE CALCULATE (I,OMEG,IM);
      VALUE I,OMEG,IM;
```

```
      REAL OMEG;
      INTEGER I,IM;
BEGIN
      ICALCULATE(I,OMEG,IM,I);
PX1 ← ANGLE (X[2] , X[1] )  ;          PX2 ← ANGLF (X[4] , X[3] )  ;
PY1 ← ANGLE (- X[5] , X[6] )  ;        PY2 ← ANGLE (-X[7] , X[8] )  ;
      IF ENTIER(IM/2)=IM/2 THEN
BEGIN
COMMENT  THIS CALCULATES THE PHASE ANGLES AND FORCES AT CRITICAL
SPEED IN Y DIRECTION  ;
      PA[2]←(SI+PY1)×RAD;
      PA[4]←(SI+PY2)×RAD;
      FORCE(C1Y,K1Y,D1X,R1X,X[5],X[6],X[1],X[2],S[I],FA[2],PFX,PFY1[I]);
      FORCE(C2Y,K2Y,D2X,R2X,X[7],X[8],X[3],X[4],S[I],FA[4],PFX,PFY2[I]);
      FPA[2]←(SI+PFY1[I])×RAD;
      FPA[4]←(SI+PFY2[I])×RAD;
      HELPME(2,4,IM,I);
END  ELSE
BEGIN
COMMENT  THIS CALCULATES THE PHASF ANGLES AND FORCES AT CRITICAL
SPEED IN X DIRECTION  ;
      PA[1]←(SI+PX1)×RAD;
      PA[3]←(SI+PX2)×RAD;
      FORCE(C1X,K1X,D1Y,R1Y,X[1],X[2],X[5],X[6],S[I],FA[1],PFX1[I],PFY);
      FORCE(C2X,K2X,D2Y,R2Y,X[3],X[4],X[7],X[8],S[I],FA[3],PFX2[I],PFY);
      FPA[1]←(SI+PFX1[I])×RAD;
      FPA[3]←(SI+PFX2[I])×RAD;
      HELPME(1,3,IM,I);
END;
END OF PROCEDURE CALCULATE;

PROCEDURE FINDMAX ;
BEGIN
      REAL  IDW     ;
      INTEGER  P, J, S     ;
      LABEL  ENDOFM , GETITGOOD , WRITEIT , GOGO, AGIN  ;
      IF CONTROL=0 THEN LON←2 ELSE LON←1 ;
      FOR IM←1 STEP LON UNTIL NOL DO
BEGIN
      IF CV[IM] = 6 THEN GO TO ENDOFM ;
      IF AA[IM,I] ≥ AA[IM,I-1] AND CV[IM] ≠ 3 THEN GO TO ENDOFM ;
      IF AA[IM,I] ≤ AA[IM,I-1] AND CV[IM] = 3 THEN GO TO ENDOFM ;
      IF AA[IM,I] < AA[IM,I-1]  THEN GO TO GETITGOOD ;
      CV[IM] ← CV[IM] + 1 ;
      GO TO ENDOFM ;
GETITGOOD :
      CV[IM] ← CV[IM] +1 ;
      IF (OMEGA[I] - OMEGA[I-1]) / OMEGA[I] ≤ SPEC THEN
BEGIN
      P ← I ;
      CV[IM] ← CV[IM] + 1 ;
      GO TO WRITEIT ;
END ;
      IDW ← DW / 2 ;    P ← I ;
GOGO :
      J ← 2 ;    FOR S ← 1, 3, 5 DO
```

```
BEGIN
     OMEGA [S] ← OMEGA [P-J] ;
     AA[IM,S] ← AA[IM,P-J] ;
     J ← J-1  ;
END;
     P ← 2 ;  OMEGA [P] ← OMEGA [P-1] + IDW ;
     ICALCULATE (P,OMEGA[P],IM,0);
AGIN :
     P ← P + 1 ;  IF P = 4 THEN
BEGIN
     OMEGA[P] ← OMEGA[P-1] + IDW ;
     ICALCULATE (P,OMEGA[P],IM,0);
END;
     IF AA[IM,P] < AA[IM,P-1] THEN
BEGIN
     IF (OMEGA[P] - OMEGA[P-1]) / OMEGA[P] ≤ SPEC THEN
BEGIN
     CV[IM] ← CV[IM] + 1 ;
     GO TO WRITEIT ;
END ;
     IDW ← IDW / 2  ;   GO TO GOGO ;
END;
     GO TO AGIN ;
WRITEIT :
     CALCULATE (P-1, OMEGA [P-1], IM) ;
     PP←0 ;   FOR S←1 STEP LON UNTIL NOL DO
     IF  CV[S]=6 THEN PP←PP+LON  ;
     IF PP ≥ NOL THEN
BEGIN
     WRITE (LP[DBL] ,<//,"THE FOLLOWING VALUES ARE AT THE MAX. SPEED =",
     F7.1 , X1, "RPS"> , WM) ;
     FOR IM ← 1,2,3,4 DO      CALCULATE (1, WM, IM ) ;
     IM ← NOL ;
END ;
ENDDFM :
END;
END OF PROCEDURE FINDMAX;

     IOTIME ← TIME(3) ;
     PTIME ← TIME(2) ;
G ← 32.2 × 12 ;
WRITE (LP[3])  ;
WRITE (LP , HEAD1) ;
WRITE (LP[PAGE]) ;
READ ( CR , / , SPEC ) ;
READ (CR , / , WO , DW , WM ) ;
LDO:  READ (CR , / , L , L1 , L2 , W , IP , IT ) [FINIS]  ;
READ (CR , / , WM1 , WM2 , H1 , H2 , PH , R1 , R2 ) ;
READ (CR , / , P , FOR J←1 STEP 1 UNTIL P DO  [LZ[J]] )  ;
READ (CR , / , K1X , K2X , K1Y , K2Y ) ;
READ (CR , / , C1X , C2X , C1Y , C2Y ) ;
READ (CR , / , D1X , D2X , D1Y , D2Y ) ;
READ (CR , / , R1X , R2X , R1Y , R2Y )        ;
     READ ( CR , / , CONTROL ) ;
PI ← 3.14159265  ;
M← W/G ;   DM1 ← WM1 / G  ;  DM2 ← WM2/G  ;
```

```
RPP ← IP / M ;  RTT← IT / M  ;
RP ← RPP/ (L× L )  ;  RT ← RTT / ( L× L )  ;
L11 ← L1 / L  ; L22 ← L2 / L  ;
RO1 ← H1 - L1  ;  RO2 ← H2 - L1  ;
PHI  ← (PH × PI ) / 180  ;
N ← 8 ;  RSW ← FALSE  ;  EPS ← 4.0 @-10  ;K1 ← 2    ;      E←1.0@-5   ;
RAD ← 57.29578  ;
K1XX ← K1X / M    ;             K2XX ← K2X /M  ;
K1YY ← K1Y /M  ;                K2YY ← K2Y /M  ;
C1XX ← C1X / M  ;               C2XX ← C2X / M  ;
C1YY ← C1Y / M  ;               C2YY ← C2Y  / M     ;
R1XX ← R1X / M  ;               R2XX ←  R2X / M  ;
R1YY ← R1Y / M ;                R2YY ← R2Y / M  ;
D1XX ← D1X / M  ;               D2XX ← D2X / M   ;
D1YY ← D1Y / M  ;               D2YY ← D2Y / M   ;
PN1 ← DM2 × R2 × SIN (PHI )  ;
PD1 ← DM1 × R1 + DM2 × R2 × COS (PHI )  ;
SI ← ANGLE (PN1 , PD1 )  ;
PN2 ← RO2 × R2 × DM2 × SIN (PHI ) ;
PD2 ← RO1 × R1 × DM1 + RO2 × R2 × DM2 × COS (PHI )  ;
SIT ← ANGLE (PN2 , PD2 ) ;
      PP←0;
WRITE (LP , HEAD2 , L , L1 , L2 , H1 , H2 , W , WM1 , WM2 , K1X , K2X ,
 K1Y , K2Y , C1X , C2X , C1Y , C2Y , R1X , R2X , R1Y , R2Y , D1X , D2X ,
D1Y ,D2Y , IP , IT , R1 , R2 , PH  )  ;
      WRITE ( LP[DBL] ) ;
      WRITE ( LP , OUT1 ) ;
      WRITE ( LP[DBL] ) ;
      NOL ← 4 + 2×P ;
      I ← 5 ;
      OMEGA[I] ← 0 ;
      FOR IM ← 1 STEP 1 UNTIL NOL DO
BEGIN
      AA[IM,I] ← 0 ;
      CV[IM] ← 1 ;
END;
      COOR[1] ← COOR[2] ← 1 ;
      COOR[3] ← COOR[4] ← 2 ;
      FOR J ← 1 STEP 1 UNTIL P DO
      COOR[2×J+3] ← COOR[2×J+4] ← LZ[J] ;
DOITAGIN :
      I ← I + 1 ;
      OMEGA [I] ← OMEGA [I-1] + DW ;
      ICALCULATE (I, OMEGA[I], NOL, 0 );
      FINDMAX ;
      IF PP ≥ NOL THEN
BEGIN
      WRITE (LP[PAGE]);
      WRITE (LP,<"TOTAL PROCESSOR TIME = ", F6.2,X1,"MINUTES">,
     (TIME(2) - PTIME) / 3600 ) ;
      WRITE (LP[PAGE ] , <"TOTAL I-O TIME = ", F6.2, X1, "MINUTES" > ,
      (TIME(3)-IOTIME)/3600 ) ;
      GO TO LDO ;
END ;
      IF OMEGA[I] ≥ WM THEN
BEGIN
```

```
            WRITE (LP[DBL] ,<//,"THE FOLLOWING VALUES ARE AT THE MAX. SPEED =",
            F7.1 , X1, "RPS"> , WM) ;
            FOR IM ← 1,2,3,4 DO       CALCULATE (1, WM, IM ) ;
            WRITE (LP[PAGE] );
            WRITE (LP,<"TOTAL PROCESSOR TIME = ", F6.2,X1,"MINUTES">,
           (TIME(2) - PTIME) / 3600 ) ;
            WRITE (LP[PAGE ] , <"TOTAL I-O TIME = ", F6.2, X1, "MINUTES" > ,
            (TIME(3)-IOTIME)/3600 ) ;
            GO TO LDO ;
       END;
            GO TO DOITAGIN ;
       E2:   WRITE (LP , < "ACCURACY  NOT  OBTAINED " > )  ;
       GO  TO  LDO  ;
       E1:  WRITE (LP , < " SINGULARITY  OR  ILL  CONDITIONED  MATRIX " > ) ;
       GO  TO  LDO  ;
       FINIS :
       END .
     ARCTAN IS SEGMENT NUMBER 0027,PRT ADDRESS IS 0252
     COS IS SEGMENT NUMBER 0028,PRT ADDRESS IS 0265
     SIN IS SEGMENT NUMBER 0029,PRT ADDRESS IS 0266
     SQRT IS SEGMENT NUMBER 0030,PRT ADDRESS IS 0254
     OUTPUT(W) IS SEGMENT NUMBER 0031,PRT ADDRESS IS 0302
     BLOCK CONTROL IS SEGMENT NUMBER 0032,PRT ADDRESS IS 0005
     INPUT(W) IS SEGMENT NUMBER 0033,PRT ADDRESS IS 0321
     GO TO SOLVER  IS SEGMENT NUMBER 0034,PRT ADDRESS IS 0271
     ALGOL WRITE   IS SEGMENT NUMBER 0035,PRT ADDRESS IS 0014
     ALGOL READ    IS SEGMENT NUMBER 0036,PRT ADDRESS IS 0015
     ALGOL SELECT  IS SEGMENT NUMBER 0037,PRT ADDRESS IS 0016
COMPILATION TIME =  119 SECONDS.
NUMBER OF ERRORS DETECTED = 000. LAST ERROR ON CARD #
NUMBER OF SEQUENCE ERRORS COUNTED =    0.
NUMBER OF SLOW WARNINGS =    0.
PRT SIZE= 234;     TOTAL SEGMENT SIZE= 1984 WORDS.
DISK STORAGE REQ.=    91 SEGS.;     NO. SEGS.=  36.
ESTIMATED CORE STORAGE REQUIREMENT =  5029 WORDS.
```

TABLE C-I

L= 3.0000@+01INCH	L1= 1.5000@+01INCH	L2= 1.5000@+01INCH	H1= 0.0000@+00INCH
H2= 0.0000@+00INCH	W= 1.1000@+02LB	WM1= 2.0000@-01LB	WM2= 2.0000@-01LB
K1X= 2.0000@+04LB/IN	K2X= 1.5000@+04LB/IN	K1Y= 1.6000@+04LB/IN	K2Y= 1.2000@+04LB/IN
C1X= 7.0000@+00LB.SEC/IN	C2X= 7.0000@+00LB.SEC/IN	C1Y= 7.0000@+00LB.SEC/IN	C2Y= 7.0000@+00LB.SEC/IN
R1X= 0.0000@+00LB/IN	R2X= 0.0000@+00LB/IN	R1Y= 0.0000@+00LB/IN	R2Y= 0.0000@+00LB/IN
D1X= 0.0000@+00LB.SEC/IN	D2X= 0.0000@+00LB.SEC/IN	D1Y= 0.0000@+00LB.SEC/IN	D2Y= 0.0000@+00LB.SEC/IN
IP= 5.7000@-01LB-IN-SEC2	IT= 2.1600@+01LB-IN-SEC2	R1= 2.0000@+00INCH	R2= 2.0000@+00INCH
	PHI= 0.0000@+00DEGREES		

SPEED REV/SEC	COORDINATE	AMPLITUDE (IN)	PHASE ANGLE	MAJOR SEMI AXIS (DIM)	MINOR SEMI AXIS (DIM)	ELLIPSE ANGLE	BEARING LOCATION	BEARING FORCE	FORCE PHASE ANGLE	PERCENT CYLINDRICAL
48.8	Y1	3.325@-02	61.7	1.11	0.45	118.5	1	5.368@+02	54.0	88.1
51.3	Y2	4.284@-02	130.4	1.16	0.14	120.4	2	5.231@+02	119.8	75.5
50.0	Y(15.0)	3.484@-02	97.8	1.16	0.25	121.1	2	5.211@+02	104.1	81.6
48.8	Y(-15.0)	3.553@-02	43.3	1.09	0.60	117.7	1	5.368@+02	54.0	88.1
53.8	X1	3.565@-02	54.5	1.06	0.42	159.5	1	7.179@+02	47.8	91.0
56.3	X2	4.809@-02	119.1	1.17	0.45	145.7	2	7.312@+02	109.7	79.3
55.0	X(15.0)	3.927@-02	87.3	1.11	0.32	153.3	1	7.173@+02	61.7	84.9
53.8	X(-15.0)	3.710@-02	39.1	1.01	0.71	167.5	1	7.179@+02	47.8	91.0
86.3	Y2	7.508@-02	258.3	1.23	0.50	129.5	2	9.449@+02	240.7	13.6
88.8	Y1	8.956@-02	103.2	1.22	0.46	128.1	1	1.475@+03	89.5	13.8
91.3	Y(15.0)	1.256@-02	151.6	1.06	0.71	116.2	1	1.458@+03	98.6	14.2
88.8	Y(-15.0)	1.705@-01	100.3	1.24	0.47	129.7	1	1.475@+03	89.5	13.8
97.5	X2	9.304@-02	264.4	1.09	0.50	153.7	2	1.451@+03	248.5	11.7
100.0	X1	1.097@-01	107.5	1.08	0.58	153.2	1	2.246@+03	95.1	12.2
102.5	X(15.0)	1.357@-02	151.2	1.05	0.80	151.4	1	2.172@+03	105.3	12.8
97.5	X(-15.0)	2.101@-01	93.2	1.11	0.53	150.1	1	2.240@+03	83.6	11.7

HE FOLLOWING VALUES ARE AT THE MAX. SPEED = 140.0 RPS

140.0	X1	5.125@-02	164.1	1.00	0.87	172.1	1	1.073@+03	147.0	18.9
140.0	Y1	4.473@-02	166.3	1.15	1.00	172.1	1	7.669@+02	145.3	20.2
140.0	X2	3.232@-02	338.1	1.00	0.83	173.5	2	5.241@+02	315.8	18.9
140.0	Y2	2.701@-02	340.5	1.20	1.00	173.5	2	3.644@+02	313.4	20.2

APPENDIX D

LISTING AND SAMPLE OUTPUT OF COMPUTER PROGRAM ROTSTAB

```
     ROTSTAB PROGRAM CALCULATES THE GENERAL TRANSIENT MOTION OF
THE FOUR DEGREE OF FREEDOM RIGID BODY ROTOR.  A TOTAL OF 8 CROSS
COUPLED STIFFNESS AND DAMPING COEFFICIENTS MAY BE PRESCRIBED FOR
EACH BEARING.  THE ROTOR CHARACTERISTIC EQUATION IS EXPANDED TO
OBTAIN AN 8TH ORDER POLYNOMIAL EQUATION WHICH IS SOLVED TO DETER-
MINE ALL REAL AND IMAGINARY ROOTS.  THE IMAGINARY COMPONENT
REPRESENTS THE ROTOR NATURAL FREQUENCY OR WHIRL SPEED AND THE REAL
COMPONENT DETERMINES STABILITY.  THE ROUTH CRITERION MAY BE USED
TO DETERMINE THE PRESENCE OF A REAL POSITIVE ROOT WITHOUT SOLVING
THE COMPLETE CHARACTERISTIC EQUATION.  IN THE CASE OF A SYMMETRIC
ROTOR USE STABIL4 OR SET ORDER TO 6.  CONVERGENCE PROBLEMS MAY
OCCUR WITH DOUBLE REPEATED ROOTS.
BEGIN
COMMENT   INPUT DATA TO THE PROGRAM, ROTSTAB, WILL BE SOUGHT IN THE
FILE, "CR".  ALL DATA IN THIS FILE MUST BE IN FREE FIELD FORMAT.  THE
LAYOUT OF THE FILE WILL BE GIVEN BELOW.

                          -----------------
                          ROTSTAB INPUT DATA
                          -----------------

     <OPTION CARDS>             THESE CARDS ARE OPTIONAL AND ANY OR ALL
                          OF THEM MAY BE OMITTED.  IF MORE THAN ONE IS
                          PRESENT THEN THEY MUST OCCUR IN THE RELATIVE
                          ORDER DESCRIBED BELOW.

          <SIGFIG CARD>         IF THE STRING, "SIGFIG" IS THE FIRST
                          FIELD ON AN OPTION CARD THEN THE CARD MUST ALSO
                          CONTAIN A SECOND VALUE WHICH WILL BE USED AS
                          THE NUMBER OF SIGNIFICANT FIGURES OF AGREEMENT
                          REQUIRED IN THE CONVERGENCE TEST.  IN THE
                          ABSENCE OF THIS CARD, TEN SIGNIFICANT FIGURES
                          WILL BE REQUIRED.

          <ORDER CARD>          IF THE STRING, "ORDER", IS THE FIRST
                          FIELD ON A CARD THEN THE CARD MUST ALSO
                          CONTAIN A SECOND VALUE.  THIS VALUE WILL BE
                          USED AS THE ORDER OF THE POLYNOMIAL AND ANY
                          HIGHER ORDER COEFICIENTS WILL BE SET TO ZERO.

          <ROUTH CARD>          IF THE FIRST FIELD ON AN OPTION CARD IS
                          THE STRING, "ROUTH", THEN THE ROUTH CRITERION
                          WILL BE APPLIED IN ORDER TO DETERMINE THE
                          STABILITY OF THE ROTOR AND THE PROBLEM WILL
                          NOT BE SOLVED FURTHER.

     <BASIC DATA CARD>          THERE WILL BE ONE <BASIC DATA CARD> FOR
                          EACH RUN OF ROTSTAB.  THE FIELDS OF THIS CARD
                          WILL BE USED AS VALUES FOR THE FOLLOWING INPUT
                          DATA AND IN THE SAME ORDER AS THEY ARE
```

```
                         DESCRIBED BELOW.
    1. L- LENGTH BETN BRGS (INCH)
    2.L1- DIST FROM 1ST BRG TO MASS CENTER (INCH)
    3.L2- DIST FROM 2ND BRG TO MASS CENTER (INCH)
    4.W- ROTOR WEIGHT (LBS)
    5. IP- POLAR M.I. (LB-IN-SEC2)
    6. IT-TRANSVERSE M.I.  OF ROTOR ABOUT MASS CENTER (LB-IN-SEC2)

    <DATA SET>                THERE MAY BE AS MANY SETS OF DATA AS
                         DESIRED.  THE LAYOUT OF A SET OF DATA WILL BE
                         DESCRIBED BELOW.  WITH THE EXCEPTION OF THE
                         FIRST SET, EACH NEW SET OF DATA SHOULD FOLLOW
                         IMMEDIATELY AFTER THE LAST CARD OF THE
                         PRECEDING SET.

CARD 1
    1. WO- INITIAL SPEED (RPS)
    2. DW- INCREMENT IN SPEED (RPS)
    3.WM- FINAL SPEED (RPS)
CARD 2
    1. K1X- 1ST BRG STIFFNESS IN X DIRECTION (LB/IN)
    2. K2X- 2ND BRG STIFFNESS IN X DIRECTION (LB/IN)
    3. K1Y- 1ST BRG STIFFNESS IN Y DIRECTION (LB/IN)
    4. K2Y- 2ND BRG STIFFNESS IN Y DIRECTION(LB/IN)
CARD 3
    1. C1X-1ST BRG DAMPING COEFF IN X DIRECTION(LB.SEC/IN)
    2. C2X- 2ND BRG DAMPING COEFF IN X DIRECTION (LB.SEC/IN)
    3.C1Y-1ST BRG DAMPING COEFF IN Y DIRECTION (LB.SEC/IN)
    4. C2Y- 2ND BRG DAMPING COEFF IN Y DIRECTION (LB.SEC/IN)
CARD 4
    1. D1X- CROSS COUPLING DAMPING COEFF  (LB.SEC/IN)
    2. D2X- CROSS COUPLING DAMPING COEFF  (LB.SEC/IN)
    3. D1Y- CROSS COUPLING DAMPING COEFF  (LB.SEC/IN)
    4. D2Y- CROSS COUPLING DAMPING COEFF  (LB.SEC/IN)
CARD 5
    1. R1X- CROSS COUPLING STIFFNESS (LB/IN)
    2. R2X- CROSS COUPLING STIFFNESS (LB/IN)
    3. R1Y- CROSS COUPLING STIFFNESS (LB/IN)
    4. R2Y- CROSS COUPLING STIFFNESS (LB/IN)
POLY
    THIS IS AN OPTION CARD AND MAY BE OMITTED.  IF PRESENT, IT MUST
    CONTAIN THE STRING, "POLY", AS THE FIRST FIELD ON THE CARD.  THIS
    WILL CAUSE THE COEFICIENTS OF THE DETERMINANT POLYNOMIAL TO
    BE PRINTED.
MODE
    THIS IS AN OPTION CARD AND MAY BE OMITTED.  IF PRESENT THEN THE
    STRING, "MODE", SHOULD BE THE FIRST FIELD ON THE CARD.  THIS
    WILL CAUSE THE MODE SHAPE VECTORS TO BE PRINTED.

THIS IS THE END OF THE COMMENT TO ROTSTAB;
    FILE SECNDRY 18 "  POLY " "  MODE " (2,15);
    FILE PRIMARY 18 "PRIMARY" " OUTPUT" (2,15);
    FORMAT FMTSPD ("----  SPEED = ",I5," RPS  ----"  ),
         FMTPOLY ("THE COEFFICIENTS OF THE DETERMINANT POLYNOMIAL ",
              "( IN ASCENDING ORDER ) ARE:"//9E13.5//),
         FMTROOT ("THERE ARE ",I1," CHARACTERISTIC ROOTS, WITH REAL ",
```

```
            "AND IMAGINARY PARTS AS FOLLOWS:"//
            "REAL     ",*E14.5/"IMAG     ",*E14.5//),
       FMTWHRL ("THE WHIRL RATIOS ARE:"/),
       FMTFREQ ("THE NATURAL FREQUENCIES ( IN CPS ) ARE:"//
            X8,8E14.5//),
       FMTMODE ("THE MODE SHAPE VECTORS ARE AS FOLLOWS:"//),
       FM2 (X24,"------------   MODE ",I1," ( NATURAL FREQUENCY = ",
            E13.5," CPS )   ------------"//
            2("VECTOR OF ",A4," PARTS  ------    ",4E18.9/)),
       PLYECHO  ("THE COEFFIEIENTS OF THE DETERMINANT POLYNOMIAL ",
            "WILL BE GIVEN."),
       MODECHO  ("THE MODE SHAPE VECTORS WILL BE GIVEN."),
       FMTODD (X60,"ODD ORDER POLYNOMIAL"//),
       ERRFMT     (60("* ")/
                 "GETRANSIENTSOLUTION WAS UNSUCESSFUL IN DOING ITS ",
            "WORK.  THE HANGUP OCCURED WHILE COMPUTING VECTOR ",
            "NUMBER ",I1,"."//60(" *")),
       FMTECHO  (3(X16,A3,E11.4,"INCH")//X16,"W=",E11.4,"LB",
            2(X16,A3,E11.4,"LB-IN-SEC2")//
 X3 , "K1X=" , E11.4 , "LB/IN" , X10 , "K2X=" , E11.4 ,"LB/IN" , X10 ,
 "K1Y=" , E11.4 , "LB/IN" , X10 , "K2Y=" , E11.4 , "LB/IN" , / ,
 X3 , "C1X=" , E11.4 , "LB.SEC/IN" , X6 , "C2X=" , E11.4 , "LB.SEC/IN" ,
 X6 , "C1Y=" , E11.4 , "LB.SEC/IN" ,X6 , "C2Y=" , E11.4 ,"LB.SEC/IN" ,/,
 X3 ,"R1X=" , E11.4 , "LB/IN" , X10 , "R2X=" , E11.4 , "LB/IN" , X10 ,
 "R1Y=" , E11.4 , "LB/IN" , X10 ,"R2Y=" , E11.4 , "LB/IN" , / ,
 X3 , "D1X=" , E11.4 , "LB.SEC/IN" , X6 , "D2X=" , E11.4 , "LB.SEC/IN" ,
 X6 , "D1Y=" , E11.4 , "LB.SEC/IN" , X6 , "D2Y=" , E11.4 , "LB.SEC/IN",/
                                       2(2(59("*")),/))   ,
       ALLWHRL  (X8,8E14.5);
    SWITCH FORMAT SWITFMT←
          (X60,"UNSTABLE",X10,"RR = ",E13.5,"   ROW = ",I2//),
          (X62,"STABLE",A1//);
    REAL G,PI,W,M,DM1,WM1,DM2,WM2,RPP,IP,RTT,IT,L,L11,L1,L22,L2,W0,
          DW,WM,K1X,K2X,K1Y,K2Y,C1X,C2X,C1Y,C2Y,D1X,D2X,D1Y,D2Y,R1X,
          R2X,R1Y,R2Y,STRING,RAD,K1XX,K2XX,K1YY,K2YY,C1XX,C2XX,RT,RP,
          C1YY,C2YY,R1XX,R2XX,R1YY,R2YY,D1XX,D2XX,D1YY,D2YY,PI2,R,S,EPS;
    INTEGER I,J,K,ODI,ROW;  REAL RR;
    ARRAY A,B,C,AR,AI,BR,BI[0:4,0:4],CMTX,ICMTX[0:8,0:8],
          MUVEK,NUVEK[0:9],WHRARY[0:14],TO[1:3];
    BOOLEAN EOFBOOL,MODE,POLY,WHRLBOL,ROUTH,ORDER;
    LABEL ALAB,BLAB,PROCESS,EOF,EXIT,SLP;
    LIST LSTALL (FOR I←1 STEP 1 UNTIL M DO NUVEK[I]/S),
          LSTMODE (K,NUVEK[K]/PI2,"REAL",FOR I←1 STEP 1 UNTIL 4 DO
               CMTX[K,I],"IMAG",FOR I←1 STEP 1 UNTIL 4 DO ICMTX[K,I]),
          LSTFREQ (FOR I←1 STEP 1 UNTIL M DO NUVEK[I]/PI2),
          LSTROUT (M,M,FOR I←1 STEP 1 UNTIL M DO MUVEK[I],
               M,FOR I←1 STEP 1 UNTIL M DO NUVEK[I]),
          LS0 (RR,ROW),  LS1 (" "),
          LSTECHO ("L =",L,"L1=",L1,"L2=",L2,W,"IP=",IP,"IT=",IT,
               K1X,K2X,K1Y,K2Y,C1X,C2X,C1Y,C2Y,R1X,R2X,R1Y,R2Y,
               D1X,D2X,D1Y,D2Y);
    SWITCH LIST SWITLST ←   LS0 , LS1;
    STREAM PROCEDURE BLANK(BASE,SKI );
    VALUE SKI ;
    BEGIN DI←BASE; SKI (DI←DI+14); DS←14 LIT "   "; END;
PROCEDURE  ROUTHH(R, N , A , RR , STABLE , ROW ) ;
```

```
COMMENT  N=ORDER OF THE POLYNOMIAL.,
     THE COEFFICIENTS OF THE POLYNOMIAL A[I] ARE READ IN DESCENDING
     POWERS OF LAMDA.
                         A[0] CORRESPONDS TO THE HIGHEST POWER OF LAMDA;
VALUE N ;
REAL  ARRAY  A[0] ,R[0,0];
REAL  RR ;
INTEGER  N , ROW ;
BOOLEAN  STABLE  ;
BEGIN
INTEGER I , J , K ;
LABEL  FIN ;                 A[N+1] ← 0;
FOR  K←0  STEP 1 UNTIL  N DO
IF  A[K]≤0  THEN
BEGIN
STABLE←FALSE ;
ROW ← K ;
RR ← A[K] ;
GO TO FIN ;
END
ELSE
FOR  I←0 , 1 DO
FOR J ←0  STEP 1 UNTIL N/2 DO
R[I,J]← A[2× J + I ] ;
FOR I← 2 STEP  1 UNTIL  N-1 DO
FOR  J←0  STEP 1 UNTIL  N/2-1  DO
BEGIN
R[I,J]← R[I-2,J+1] - R[I-2,0] × R[I-1,J+1]/R[I-1,0] ;
IF  R[I,0]<0 THEN
BEGIN
STABLE ← FALSE ;
ROW←I ;
RR← R[I,0] ;
GO TO FIN;
END ;
END;
STABLE← TRUE ;
FIN: END OF PROCEDURE  ROUTH ;
     PROCEDURE TIMEANDATE(TZERO,FYLE,OPTION);
     VALUE OPTION; REAL OPTION; INTEGER ARRAY TZERO[*]; FILE FYLE;
     COMMENT THIS IS A UTILITY PROCEDURE WRITTEN BY R. TOMLIN,
          RLES. THE ACTION OF THE PROCEDURE DEPENDS ON THE
          RIGHTMOST 39 BITS OF THE PARAMETER OPTION. FOR
          CONVENIENCE, THIS 39 BIT PACKAGE WILL BE IDENTIFIED
          WITH THE STRING "NFFFDDD". HERE N IS THE OCTAL DIGI
          CONSISTING OF THE 3 LEFTMOST BITS OF THE PACKAGE,
          AND FFFDDD IS THE COLLECTION OF 6 CHARACTERS DEFINED
          BY THE REMAINING 36 BITS. N IS CALLED THE IDENT-
          -IFICATION DIGIT, AND FFF, DDD, AND FFFDDD ARE CALLED
          THE FILE, DATE, AND COMPOSITE OPTIONS, RESPECTIVELY.
               INITIALLY, "FFFDDD" IS COMPARED WITH THE STRING,
          "CENTER". IF THEY ARE EQUAL, THEN A CHECK IS MADE ON
          THE VALUE OF N. IF N=0, THEN FYLE IS ASSUMED TO BE
          A LINE PRINTER FILE. THE PRINTER IS DOUBLE SPACED AND
          THE DATE IS WRITTEN OUT, CENTERED ON THE LINE, WITH
          CARRIAGE CONTROL [DBL]. IF N DOES NOT EQUAL ZERO,
```

```
     THEN FYLE IS TAKEN TO BE AN ALPHA TAPE FILE, AND THE
     RLESMPT EQUIVALENT OF DOUBLE-SPACE, CENTERED-DATE, DOUBLE
     SPACE IS WRITTEN ON TAPE. IN EITHER CASE, THE PROCEDURE
     IS THEN EXITED.
          IF "FFFDDD" DOES NOT EQUAL "CENTER", THEN "FFF" IS
     COMPARED WITH "MPT". IF "FFF" AND "MPT" DO NOT AGREE, THE
     PROCEDURE ASSUMES THAT FYLE IS A LINE PRINTER FILE. FIRST
     THE PRINTER IS DOUBLE SPACED, AND THEN A LINE IS WRITTEN
     WHICH CONTAINS THE DATE, PLACED NEAR THE LEFT MARGIN. THE
     CARRIAGE CONTROL FOR THIS LINE IS [NO], AND THE FIRST
     CHARACTER OF THE LINE IS DETERMINED BY N. IF N=0, THEN
     THE FIRST CHARACTER IS A BLANK, OTHERWISE IT IS THE
     DIGIT, N.  NEXT, "DDD" IS COMPARED WITH "DAT". IF THEY
     AGREE, THE PRINTER IS DOUBLE SPACED AND THE PROCEDURE IS
     EXITED. IF "DDD" DIFFERS FROM "DAT", THEN IT IS ASSUMED
     THAT THE FIRST THREE ENTRIES OF TZERO HAVE BEEN INITIAL-
     IZED WITH READINGS FROM THE ELAPSED, PROCESSOR, AND I/O
     CLOCKS. THE REMAINDER OF THE LINE JUST WRITTEN IS THEN
     FILLED OUT (USING CARRIAGE CONTROL [DBL]) WITH THE
     AMOUNTS OF ELAPSED, PROCESSOR, AND I/O TIME WHICH HAVE
     PASSED SINCE THAT INITIALIZING. THE PROCEDURE IS
     THEN EXITED.
          IF "FFF"="MPT", THEN FYLE IS ASSUMED TO BE AN
     ALPHA TAPE FILE. THE REMAINING ACTION IS IDENTICAL TO
     THAT ABOVE EXCEPT THAT RLESMPT RECORDS WILL BE WRITTEN
     ON TAPE, INSTEAD OF LINE IMAGES BEING WRITTEN ON THE
     LINE PRINTER;
BEGIN STREAM PROCEDURE SEPARATEYYDDD(YYDDD,YY,DDD);
     BEGIN DI←YYDDD; DS←3 LIT"019"; DI←YY; SI←YYDDD;
          SI←SI+1; DS←4 OCT; DI←DDD; DS←3 OCT
     END OF SEPARATEYYDDD PROCEDURE;
     STREAM PROCEDURE TRANSFER(VEKIN,VEKOUT);
     BEGIN SI←VEKIN; DI←VEKOUT; DS←3 WDS END;
     ALPHA ALF;
     INTEGER MNTHNMBR,DAYNMBR,EXCESS,TYMZERO,YEAR,
          DAYOFMNTH,TUMNO,K,J;
     INTEGER ARRAY TNAUT,DELTA[1:3],DAYCOUNT[0:11],
          MNTHNAME[0:23];
     FORMAT FM0 (A1,A4,A5,I3,",",I5,".  "),
          FM1 (O,A1,A4,A5,I3,",",I5,".  "),
          FM2 (X22,"TOTAL ELAPSED TIME IS",I6," SECONDS",
               ".  PROCESSOR TIME IS",I6," SECONDS.  ",
               "I/O TIME IS",I6," SECONDS."),
          FM3 (O,X22,"TOTAL ELAPSED TIME IS",I6," SECONDS",
               ".  PROCESSOR TIME IS",I6," SECONDS.  ",
               "I/O TIME IS",I6," SECONDS."),
          FMCNTRE1 (X50,A5,A5,I3,",",I5,"."),
          FMCNTRE2 (O,X50,A5,A5,I3,",",I5,"."),
          FMSTAR (O);
     FILL MNTHNAME[*] WITH "   JA","NUARY",
          "  FEB","RUARY","     ","MARCH","     ","APRIL",
          "     "," MAY","     "," JUNE","     "," JULY",
          "    A","UGUST"," SEPT","EMBER","   OC","TOBER",
          "  NOV","EMBER","  DEC","EMBER";
     TRANSFER(TZERO,TNAUT); K←0;
     FOR J←31,28,31,30,31,30,31,31,30,31,30,31 DO
```

```
        BEGIN DAYCOUNT[K]←J; K←K+1 END;
        TYMZERO←TIME(0);
        SEPARATEYYDDD(TYMZERO,YEAR,DAYNMBR);
        IF YEAR MOD 4 =0 THEN DAYCOUNT[1]←29;
        EXCESS←DAYNMBR; MNTHNMBR←-1;
        FOR K←MNTHNMBR WHILE EXCESS>0 DO
        BEGIN EXCESS←EXCESS-DAYCOUNT[K+1];
             MNTHNMBR←MNTHNMBR+1
        END; TUMNO←2×MNTHNMBR;
        DAYOFMNTH←EXCESS+DAYCOUNT[MNTHNMBR];
        ALF←(IF OPTION.[9:3]=0 THEN " " ELSE OPTION.[9:3]);
        IF OPTION.[12:36]="CENTER" THEN
        BEGIN IF OPTION.[9:3]=0 THEN
             BEGIN WRITE(FYLE[DBL]);
                  WRITE(FYLE[DBL],FMCNTRE1,MNTHNAME[TUMNO],
                       MNTHNAME[TUMNO+1],DAYOFMNTH,YEAR)
             END
                  ELSE
             BEGIN WRITE(FYLE,FMSTAR,2);
                  WRITE(FYLE,FMCNTRE2,2,MNTHNAME[TUMNO],
                       MNTHNAME[TUMNO+1],DAYOFMNTH,YEAR)
             END
        END OF CENTER OPTIONS
             ELSE
        IF OPTION.[12:18]≠"MPT" THEN
        BEGIN WRITE(FYLE[DBL]);
             WRITE(FYLE[NO],FMO,ALF,MNTHNAME[TUMNO],
             MNTHNAME[TUMNO+1],DAYOFMNTH,YEAR);
             IF OPTION.[30:18]="DAT" THEN WRITE(FYLE[DBL])
                  ELSE
             BEGIN FOR J←1,2,3 DO DELTA[J]←(TIME(J)-TNAUT[J])/60;
                  WRITE(FYLE[DBL],FM2,DELTA[1],DELTA[2],DELTA[3])
             END
        END OF PRINTER OPTIONS
             ELSE
        BEGIN WRITE(FYLE,FMSTAR,2);
             WRITE(FYLE,FM1,0,ALF,MNTHNAME[TUMNO],
                  MNTHNAME[TUMNO+1],DAYOFMNTH,YEAR);
             IF OPTION.[30:18]="DAT" THEN WRITE(FYLE,FMSTAR,2)
                  ELSE
             BEGIN FOR J←1,2,3 DO DELTA[J]←(TIME(J)-TNAUT[J])/60;
                  WRITE(FYLE,FM3,2,DELTA[1],DELTA[2],DELTA[3])
             END
        END OF RLESMPT OPTIONS
   END OF TIMEANDATE PROCEDURE;
        PROCEDURE MLTPLYREALPOLY(M,N,A,B,C);
        VALUE M,N; INTEGER M,N; ARRAY A,B,C[0];
        BEGIN COMMENT   THIS PROCEDURE ASSUMES THAT THE
                  VECTORS A AND B CONTAIN THE COEFFICIENTS OF
                  POLYNOMIALS OF ORDER M AND ORDER N, RESPECTIVELY.
                  SPECIFICALLY, THE COEFFICIENTS OF THE KTH POWER
                  OF THE POLYNOMIAL VARIABLE ARE STORED IN A[K]
                  AND B[K], RESPECTIVELY.
                       MLTPLYREALPOLY COMPUTES THE COEFFICIENTS
                  OF THE POLYNOMIAL WHICH IS THE PRODUCT OF THE
                  GIVEN TWO, AND STORES THEM INTO THE VECTOR, C.
```

```
          AS WITH A AND B, THE COEFFICIENT OF THE KTH
          POWER IS STORED INTO C[K].
               THE ARITHMETIC IS SO ARRANGED THAT IF A
          CONTAINS THE COEFFICIENTS OF THE POLYNOMIAL
          OF LESSER DEGREE, THEN THE MOST EFFICIENT
          CONFIGURATION HAS BEEN REALIZED;
     REAL AP; INTEGER K,P,Q; P←M+N;
     FOR K←0 STEP 1 UNTIL P DO C[K]←0;
     FOR P←0 STEP 1 UNTIL M DO
     BEGIN AP←A[P];
          FOR Q←0 STEP 1 UNTIL N DO
          C[K←(P+Q)]←B[Q]×AP+C[K]
     END OF THE LOOP ON P
END OF THE MLTPLYREALPOLY PROCEDURE;
PROCEDURE GETDETPOLY(N,A,B,C,D);
VALUE N; INTEGER N; ARRAY A,B,C[0,0],D[0];
BEGIN COMMENT   CONSIDER THE N×N MATRIX, G, DEFINED BY
          G[I,J]=A[I,J]×T*2+B[I,J]×T+C[I,J].  IT IS CLEAR
          THAT THE DETERMINANT OF G IS A POLYNOMIAL OF
          DEGREE 2N IN THE PARAMETER,T.
               GETDETPOLY COMPUTES THE COEFFICIENTS OF THIS
          POLYNOMIAL AND STORES THEM INTO THE VECTOR, D.
          THE COEFFICIENT OF THE KTH POWER OF T IS
          STORED INTO D[K], THIS FOR K=0, 1,..., 2×N.
               THE ENTRIES OF A, B, AND C WHICH HAVE
          INDICES IN THE RANGE FROM ONE TO N ARE
          ASSUMED TO CONTAIN THE REQUIRED QUANTITIES.
          THOSE ENTRIES INVOLVING A ZERO INDEX ARE NOT
          REFERENCED BY GETDETPOLY;
     INTEGER TN,NM1,TNM1,KM1,K,P,Q,I,J; LABEL EXIT;
     ARRAY QUAD[0:2],DALT[0:2×N],
          DM[0:2×(N-1)],AM,BM,CM[0:N-1,0:N-1];
          TN←2×N; TNM1←2×(NM1←N-1);
     IF N=1 THEN
     BEGIN D[0]←C[1,1]; D[1]←B[1,1];
          D[2]←A[1,1]; GO TO EXIT
     END OF THE SPECIAL CASE WHEN N EQUALS ONE;
     FOR K←0 STEP 1 UNTIL TN DO D[K]←0;
     FOR K←1 STEP 1 UNTIL N DO
     IF A[K,1]≠0 OR B[K,1]≠0 OR C[K,1]≠0 THEN
     BEGIN KM1←K-1;
          FOR I←1 STEP 1 UNTIL KM1 DO
          FOR J←2 STEP 1 UNTIL N DO
          BEGIN AM[I,Q←(J-1)]←A[I,J];
               BM[I,Q]←B[I,J]; CM[I,Q]←C[I,J]
          END OF THE LOOP ON J;
          FOR I←K+1 STEP 1 UNTIL N DO
          BEGIN P←I-1; FOR J←2 STEP 1 UNTIL N DO
               BEGIN AM[P,Q←(J-1)]←A[I,J];
                    BM[P,Q]←B[I,J]; CM[P,Q]←C[I,J]
               END OF THE LOOP ON J
          END OF THE LOOP ON I;
          DM[0]←CM[1,1]; DM[1]←BM[1,1]; DM[2]←AM[1,1];
          IF N>2 THEN GETDETPOLY(NM1,AM,BM,CM,DM);
          IF BOOLEAN(K) THEN
          BEGIN QUAD[0]←C[K,1];
```

```
                QUAD[1]←B[K,1]; QUAD[2]←A[K,1]
            END OF EVEN PARITY CASE
                ELSE
            BEGIN QUAD[0]←-C[K,1];
                QUAD[1]←-B[K,1]; QUAD[2]←-A[K,1]
            END OF ODD PARITY CASE;
            MLTPLYREALPOLY(2,TNM1,QUAD,DM,DALT);
            FOR I←0 STEP 1 UNTIL TN DO D[I]←DALT[I]+D[I]
        END OF THE LOOP ON K;
      EXIT:
   END OF THE GETDETPOLY PROCEDURE;
        REAL PROCEDURE INRPROD(N,A,B);
        VALUE N; INTEGER N; ARRAY A,B[0];
        BEGIN COMMENT   THIS PROCEDURE COMPUTES THE
                INNER PRODUCT OF A AND B AND STORES IT
                INTO THE IDENTIFIER, INRPROD.
                    A AND B ARE ASSUMED TO HAVE INDICES
                IN THE RANGE 0 TO N.  A[0] AND B[0] ARE
                NOT REFERENCED BY THIS PROCEDURE;
            INTEGER K; REAL T; T←0;
            FOR K←1 STEP 1 UNTIL N DO T←A[K]×B[K]+T;
            INRPROD←T
        END OF THE INRPROD PROCEDURE;
   REAL PROCEDURE MODOFINRPROD(N,A,IA,B,IB);
   VALUE N; INTEGER N; ARRAY A,IA,B,IB[0];
   BEGIN COMMENT   THE MODULUS OF THE INNER PRODUCT OF
            THE COMPLEX VECTORS S AND T IS COMPUTED AND
            STORED INTO MODOFINRPROD.  FURTHER, THE REAL
            AND IMAGINARY PARTS OF <S,T>, ITSELF, ARE STORED
            INTO A[0] AND IA[0], RESPECTIVELY.
                S AND T ARE ASSUMED TO HAVE N ENTRIES,
            BEGINNING AT INDEX VALUE ONE.  THE REAL AND
            IMAGINARY PARTS OF S ARE, RESPECTIVELY, A AND
            IA.  THOSE OF T ARE B AND IB, RESPECTIVELY;
        INTEGER K; REAL RE,IM; RE←IM←0;
        FOR K←1 STEP 1 UNTIL N DO
        BEGIN RE←A[K]×B[K]+IA[K]×IB[K]+RE;
            IM←IA[K]×B[K]-IB[K]×A[K]+IM
        END OF THE SUMMATION LOOP;
        MODOFINRPROD←SQRT((A[0]←RE)*2+(IA[0]←IM)*2)
   END OF THE MODOFINRPROD PROCEDURE;
   REAL PROCEDURE MODSQOFINRPROD(N,A,IA,B,IB);
   VALUE N; INTEGER N; ARRAY A,IA,B,IB[0];
   BEGIN COMMENT   THE MODULUS SQUARED OF THE INNER PRODUCT OF
            THE COMPLEX VECTORS S AND T IS COMPUTED AND
            STORED INTO MODSQOFINRPROD.  FURTHER, THE REAL
            AND IMAGINARY PARTS OF <S,T>, ITSELF, ARE STORED
            INTO A[0] AND IA[0], RESPECTIVELY.
                S AND T ARE ASSUMED TO HAVE N ENTRIES,
            BEGINNING AT INDEX VALUE ONE.  THE REAL AND
            IMAGINARY PARTS OF S ARE, RESPECTIVELY, A AND
            IA.  THOSE OF T ARE B AND IB, RESPECTIVELY;
        INTEGER K; REAL RE,IM; RE←IM←0;
        FOR K←1 STEP 1 UNTIL N DO
        BEGIN RE←A[K]×B[K]+IA[K]×IB[K]+RE;
            IM←IA[K]×B[K]-IB[K]×A[K]+IM
```

```
                 END OF THE SUMMATION LOOP;
                 MODSQOFINRPROD←   (A[0]←RE)*2+(IA[0]←IM)*2
            END OF THE MODSQOFINRPROD PROCEDURE;
REAL PROCEDURE CMPLXINVERSE(N,A,IA);
VALUE N; INTEGER N; ARRAY A, IA[0,0];
COMMENT   THIS IS A MODIFICATION OF RODMANS PROCEDURE FOR
     INVERTING A COMPLEX MATRIX, S.  THE MATRIX, S, IS
     ASSUMED TO BE OF ORDER N, AND TO HAVE IJ-TH ENTRIES
     WHOSE REAL AND IMAGINARY PARTS ARE A[I,J] AND IA[I,J],
     RESPECTIVELY.  THE PROCEDURE IS EXITED WITH THE MODULUS
     OF THE DETERMINANT OF S STORED INTO CMPLXINVERSE;
COMMENT        THIS PROCEDURE INVERTS A MATRIX OF COMPLEX ELEMENTS.
               SEE CORRESPONDING TECHNICAL BULLETIN FOR DETAILS  ON USE
               OF THE PROCEDURE.

               R.D. RODMAN
               (PROFESSIONAL SERVICES DIVISIONAL GROUP),

               CARD SEQUENCE BEGINS WITH CINV0001,
               FIRST RELEASE 4/1/63 ;

          BEGIN
INTEGER        I, Z, K, L, K2, J, V, Y, W ;
REAL           BIG, T, EPS, TEMP, DIAG, IT ;
ARRAY          Q1[0:1,0:N], Q2[0:1,0:N] ;
INTEGER ARRAY F[0:N] ;    LABEL EXIT;
PROCEDURE      CIP(A, B, N) ;

VALUE          N ;
INTEGER        N ;
ARRAY          A, B[0,0] ;

          BEGIN
REAL           Q, IQ ;
INTEGER        I ;

               Q ← IQ ← 0 ;

               FOR I ← 1 STEP 1 UNTIL N DO
          BEGIN
               Q ← A[0,I] × B[0,I] - A[1,I] × B[1,I] + Q ;
               IQ ← A[1,I] × B[0,I] + A[0,I] × B[1,I] + IQ
          END ;
               A[0,0] ← Q ;  A[1,0] ← IQ
          END ;

               FOR I ← 1 STEP 1 UNTIL N DO
          BEGIN
               Z ← I-1 ;

               FOR K ← 1 STEP 1 UNTIL Z DO
          BEGIN
               Q1[0,K] ← A[K,I] ;  Q1[1,K] ← IA[K,I]
          END ;

               FOR K ← I STEP 1 UNTIL N DO
```

```
BEGIN

     FOR L ← 1 STEP 1 UNTIL Z DO
BEGIN
     Q2[0,L] ← A[K,L] ;  Q2[1,L] ← IA[K,L]
END ;
     CIP(Q1, Q2, Z) ;
     A[K,I] ← A[K,I] - Q1[0,0] ;  IA[K,I] ← IA[K,I] - Q1[1,0]
END ;
     BIG ← 0 ;  K2 ← I ;

     FOR K ← I STEP 1 UNTIL N DO
BEGIN
     T ● A[K,I]*2 + IA[K,I]*2 ;
     IF T > BIG THEN
BEGIN
     BIG ← T ;  K2 ← K
END
END ;
      IF BIG=0 THEN BEGIN CMPLXINVERSE←0; GO TO EXIT END;
     F[I] ← K2 ;
     IF K2 ≠ I THEN FOR K ← 1 STEP 1 UNTIL N DO
BEGIN
     TEMP ← A[I,K] ;  A[I,K] ← A[K2,K] ;  A[K2,K] ← TEMP ;
     TEMP ← IA[I,K] ;  IA[I,K] ← IA[K2,K] ;  IA[K2,K] ← TEMP
END ;
     DIAG ● 1/(A[I,I]*2 + IA[I,I]*2) ;

     FOR K ← 1 STEP 1 UNTIL Z DO
BEGIN
     Q1[0,K] ← A[I,K] ;  Q1[1,K] ← IA[I,K]
END ;

     FOR K ← I+1 STEP 1 UNTIL N DO
BEGIN

     FOR L ← 1 STEP 1 UNTIL Z DO
BEGIN
     Q2[0,L] ← A[L,K] ;  Q2[1,L] ● IA[L,K]
END ;
     CIP(Q1, Q2, Z) ;
     T ← A[I,K] - Q1[0,0] ;  IT ← IA[I,K] - Q1[1,0] ;
     A[I,K] ←(T×A[I,I] + IT×IA[I,I]) × DIAG ;
     IA[I,K] ← (IT×A[I,I] - T×IA[I,I]) × DIAG
END
END ;
      T←1; FOR K←1 STEP 1 UNTIL N DO T←(A[K,K]*2+IA[K,K]*2)×T;
           CMPLXINVERSE←SQRT(T);

     FOR I ← 1 STEP 1 UNTIL N DO
BEGIN
     DIAG ← 1/(A[I,I]*2 + IA[I,I]*2) ;   Z ← I-1 ;

     FOR J ← 1 STEP 1 UNTIL I DO
BEGIN
     IF I ≠ J THEN
```

```
BEGIN

     FOR K ← J STEP 1 UNTIL Z DO
BEGIN
     Q1[0,K-J+1] ← A[K,J] ;   Q1[1,K-J+1] ← IA[K,J] ;
     Q2[0,K-J+1] ← A[I,K] ;   Q2[1,K-J+1] ← IA[I,K]
END ;
     CIP(Q1, Q2, I-J) ;
     A[I,J] ← (-Q1[0,0]×A[I,I] - Q1[1,0]×IA[I,I]) × DIAG ;
     IA[I,J] ← (Q1[0,0]×IA[I,I]- Q1[1,0]×A[I,I]) ×DIAG
END
     ELSE
BEGIN
     A[I,I] ← A[I,I] × DIAG ;
     IA[I,I] ← -IA[I,I] × DIAG
END
END
END ;
     V ← N-1 ;

     FOR I ← V STEP -1 UNTIL 1 DO
BEGIN
     Z ← I+1 ;

     FOR J ← N STEP -1 UNTIL Z DO
BEGIN
     Y ← J-1 ;

     FOR K ← I+1 STEP 1 UNTIL Y DO
BEGIN
     Q1[0,W←K-I] ← A[K,J] ;   Q1[1,W] ← IA[K,J] ;
     Q2[0,W] ← A[I,K] ;   Q2[1,W] ← IA[I,K]
END ;
     CIP(Q1, Q2, Y-I) ;
     A[I,J] ← -A[I,J] - Q1[0,0] ;
     IA[I,J] ← -IA[I,J] - Q1[1,0]
END
END ;

     FOR I ← 1 STEP 1 UNTIL V DO
BEGIN

     FOR J ← 1 STEP 1 UNTIL N DO
BEGIN
     IF I ≥ J THEN
BEGIN

     FOR K ← I+1 STEP 1 UNTIL N DO
BEGIN
     Q1[0,K-I] ← A[I,K] ;   Q1[1,K-I] ← IA[I,K] ;
     Q2[0,K-I] ← A[K,J] ;   Q2[1,K-I] ← IA[K,J]
END ;
     CIP(Q1, Q2, N-I) ;
     A[I,J] ← A[I,J] + Q1[0,0] ;
     IA[I,J] ← IA[I,J] + Q1[1,0]
END
```

```
        ELSE
    BEGIN

        FOR K ← J STEP 1 UNTIL N DO
    BEGIN
        Q1[0,W←K-J+1] ← A[K,J] ;  Q1[1,W] ← IA[K,J] ;
        Q2[0,W] ← A[I,K] ;  Q2[1,W] ← IA[I,K]
    END ;
        CIP(Q1, Q2, N-J+1) ;
        A[I,J] ← Q1[0,0] ;  IA[I,J] ← Q1[1,0]
    END
    END
    END ;

        FOR J ← N STEP -1 UNTIL 1 DO
    BEGIN
        IF F[J] ≠ J THEN
    BEGIN
        K2 ← F[J] ;

        FOR K ← 1 STEP 1 UNTIL N DO
    BEGIN
        TEMP ← A[K,K2] ;  A[K,K2] ← A[K,J] ;  A[K,J] ← TEMP ;
        TEMP ← IA[K,K2] ;  IA[K,K2] ← IA[K,J] ;  IA[K,J] ← TEMP
    END
    END
        ELSE
    END;
EXIT:
    END;
     PROCEDURE FINDPOLYORDERANDNORMALIZE(N, AR, AI, P);
     VALUE N; INTEGER N,P; ARRAY AR,AI[0];
     BEGIN COMMENT   A POLYNOMIAL OF DEGREE LESS THAN OR
               EQUAL TO N, WHOSE K-TH POWER COEFFICIENT HAS
          REAL AND IMAGINARY PARTS AR[K] AND AI[K], FOR
               K = 0, ..., N, WILL BE EXAMINED BY THIS PROCEDURE.
                    THE COEFFICIENTS WILL BE ADJUSTED TO MAKE
               IT A MONIC POLYNOMIAL, I.E., THE COEFFICIENT
               OF THE HIGHEST POWER WILL BECOME A QUANTITY
               WITH MODULUS UNITY, AND THE TRUE ORDER (DEGREE)
               OF THE POLYNOMIAL WILL BE INSERTED INTO P;
          INTEGER K; REAL T;
          FOR P←N STEP -1 WHILE (T←AR[P]*2+AI[P]*2) = 0 DO;
          T←SQRT(T);
          FOR K←0 STEP 1 UNTIL P DO
          BEGIN AR[K]←AR[K]/T; AI[K]←AI[K]/T END
     END OF THE FINDPOLYORDERANDNORMALIZE PROCEDURE;
     PROCEDURE SCALECOEFFICIENTS(P,  AR,  AI,  SCALE); VALUE P;
          INTEGER P; ARRAY AR, AI[0]; REAL SCALE;
     BEGIN COMMENT   GIVEN HERE IS A POLYNOMIAL IN THE VARIABLE
          Z WHOSE COEFFICIENT FOR THE K-TH POWER OF Z HAS REAL
          AND IMAGINARY PARTS AR[K] AND AI[K], FOR K= 0, ..., P.
               THIS PROCEDURE SCALES THE COEFFICIENTS OF THE
          POLYNOMIAL, DEFINING IN THE PROCESS A NEW POLYNOMIAL
          IN THE VARIABLE ZPRIME, WHERE Z = SCALE × ZPRIME,
          SUCH THAT THE COEFFICIENT OF THE LOWEST ORDER TERM
```

```
        IN THE POLYNOMIAL HAS MODULUS UNITY;
             REAL A, R, I, T; INTEGER K,Q; LABEL L; K←0;
          L:    A←AR[K]*2+AI[K]*2;
             IF A=0 THEN BEGIN K←K+1; GO TO L END;
             SCALE←T←A*(1/(2×(P-K))); Q←K;
             FOR K←P-1 STEP -1 UNTIL Q DO
             BEGIN AR[K]←AR[K]/T; AI[K]←AI[K]/T;
                  T←T×SCALE
             END
    END OF THE SCALECOEFFICIENTS PROCEDURE;
PROCEDURE GETPOLYZEROS(N, AR, AI, EPSILON);
VALUE N, EPSILON; REAL EPSILON; INTEGER N; ARRAY AR, AI[0];
COMMENT     THIS PROCEDURE FINDS ZEROS OF A POLYNOMIAL
    OF ORDER N. THE COEFFICIENT OF THE HIGHEST POWER OF
    THE VARIABLE MUST BE UNITY.  ON ENTRY, AR[K] AND
    AI[K] FOR K=0, 1, - - -, N ARE THE REAL AND
    IMAGINARY PARTS OF THE COEFFICIENTS OF ASCENDING
    POWERS OF THE VARIABLE.  ON EXIT, AR AND AI[1, ..., N]
    CONTAIN THE ZEROS.  NEWTONS METHOD IS USED.
    ITERATION CONTINUES UNTIL THE SQUARE OF THE FRACTIONAL
    CHANGE IN THE ZERO DOES NOT EXCEED EPSILON.  AFTER THE
    FIRST ZERO IS FOUND, THE ORDER OF THE POLYNOMIAL IS
    REDUCED BY DIVISION.  ZEROS OBTAINED FROM THE REDUCED
    POLYNOMIAL ARE IMPROVED BY ITERATION WITH THE ORIGINAL
    POLYNOMIAL.  THEN THE ORDER OF THE REDUCED POLYNOMIAL
    IS FURTHER REDUCED.  ;
BEGIN REAL X, Y, FR, FI, GR, GI, U, V, W; INTEGER K, P, Q;
    ARRAY BR, BI[0:N], CR, CI, RR, RI, MF[1:N]; REAL T;
    LABEL AGAIN, GUESSZERO, ITERATE, REITERATE, EXIT;
    BOOLEAN ONCE; INTEGER NDIV2; NDIV2←N DIV 2;
    FOR K←0 STEP 1 UNTIL NDIV2 DO
    BEGIN T←AR[K]; AR[K]←AR[N-K]; AR[N-K]←T;
         T←AI[K]; AI[K]←AI[N-K]; AI[N-K]←T
    END OF THE SWITCH AROUND LOOP ON THE INTEGER K;
    N←N+1; FOR N←N-1 WHILE AR[N]=0 AND AI[N]=0 DO    ;
    IF N=1 THEN BEGIN AR[1]←-AR[1]; AI[1]←-AI[1]; GO TO EXIT END;
    BR[0]←1.0; BI[0]←0; ONCE←FALSE;
  AGAIN: FOR K←1 STEP 1 UNTIL N DO
    BEGIN BR[K]←AR[K]; BI[K]←AI[K] END;
    P←N;
  GUESSZERO: IF ONCE THEN BEGIN X←RR[P]; Y←RI[P] END
                    ELSE BEGIN X←1-BR[1]; Y←1-BI[1];
                               IF P=1 THEN
                               BEGIN X←X-1; Y←Y-1; GO TO ITERATE
                               END
                         END;
    Q←P;
    FOR K←1 STEP 1 UNTIL Q DO
    BEGIN CR[K]←BR[K]; CI[K]←BI[K] END;
  ITERATE: FR←1; FI←0;
    FOR K←1 STEP 1 UNTIL Q DO
    BEGIN U←X×FR-Y×FI+CR[K];
          V←X×FI+Y×FR+CI[K];
          FR←U; FI←V
    END;
    GR←Q; GI←0;
```

```
        FOR K←1 STEP 1 UNTIL Q-1 DO
        BEGIN U←X×GR-Y×GI+(Q-K)×CR[K];
               V←X×GI+Y×GR+(Q-K)×CI[K];
               GR←U; GI←V
        END;
        U←FR×GR+FI×GI; V←FI×GR-FR×GI; W←GR*2+GI*2;
        IF W=0 THEN W←1;
        U←U/W; V←V/W; W←U×U+V×V;
        U←X-U; V←Y-V; W←2.0×W/(U×U+V×V+X×X+Y×Y);
        X←U; Y←V;
        IF W>EPSILON THEN GO TO ITERATE;
    REITERATE: IF Q≠N THEN
                        BEGIN FOR K←1 STEP 1 UNTIL N DO
                             BEGIN CR[K]←AR[K]; CI[K]←AI[K] END;
                             Q←N; GO TO ITERATE
                        END;
        RR[P]←X; RI[P]←Y; MF[P]←FR*2+FI*2;
        IF P≠1 THEN
             BEGIN P←P-1;
                  FOR K←1 STEP 1 UNTIL P DO
                  BEGIN BR[K]←BR[K]+X×BR[K-1]-Y×BI[K-1];
                       BI[K]←BI[K]+X×BI[K-1]+Y×BR[K-1]
                  END;
                  GO TO GUESSZERO
             END;
        IF NOT ONCE THEN
        BEGIN ONCE←TRUE;
             FOR K←1 STEP 1 UNTIL N DO
             BEGIN U←RR[K]; V←RI[K]; W←MF[K];
                  FOR Q←K+1 STEP 1 UNTIL N DO
                  IF MF[Q]>W THEN
                  BEGIN RR[K]←RR[Q]; RI[K]←RI[Q]; MF[K]←MF[Q];
                      RR[Q]←U; RI[Q]←V; MF[Q]←W;
                       U←RR[K]; V←RI[K]; W←MF[K]
                  END
             END;
             GO TO AGAIN
        END;
        FOR K←1 STEP 1 UNTIL N DO
        BEGIN AR[K]←RR[K]; AI[K]←RI[K] END;
    EXIT:
  END OF PROCEDURE POLYZEROS;
        PROCEDURE UNSCALETHEROOTS(P, AR, AI, SCALE); VALUE P;
             INTEGER P; ARRAY AR, AI[0]; REAL SCALE;
             BEGIN COMMENT  THIS PROCEDURE IS USED IN CONJUNCTION
                  WITH THE SCALECOEFFICIENTS PROCEDURE.  IT UNSCALES
                  THE ROOTS OF THE POLYNOMIAL WHICH WAS SCALED;
                  INTEGER K;
                  FOR K←1 STEP 1 UNTIL P DO
                  BEGIN AR[K]←AR[K]×SCALE; AI[K]←AI[K]×SCALE END
        END OF THE UNSCALETHEROOTS PROCEDURE;
             PROCEDURE CMPLXLINTRAN(N,A,IA,X,IX,Y,IY);
             VALUE N; INTEGER N; ARRAY X,IX,Y,IY[0],A,IA[0,0];
             BEGIN COMMENT  THE INDEX UPPER BOUNDS FOR A,
                       IA,X,IX,Y, AND IY ARE ASSUMED TO BE EQUAL
                       TO N.  THE ENTRIES OF THESE ARRAYS WHICH
```

```
          CORRESPOND TO A ZERO INDEX ARE NOT
          REFERENCED BY THIS PROCEDURE.
               CONSIDER THE COMPLEX MATRIX, S, WHOSE
          IJTH ENTRY HAS REAL AND IMAGINARY PARTS
          A[I,J] AND IA[I,J], RESPECTIVELY, FOR I,J
          =1, ..., N.  FURTHER, LET T AND U DENOTE
          THE COMPLEX VECTORS WHOSE KTH ENTRIES HAVE
          REAL AND IMAGINARY PARTS X[K], IX[K] AND
          Y[K], IY[K], RESPECTIVELY, FOR
                    K=1, ..., N.
               WHERE S IS REGARDED AS A LINEAR
          TRANSFORMATION, THIS PROCEDURE COMPUTES
          THE IMAGE OF T UNDER S AND STORES IT
          INTO U, I.E., ST IS STORED INTO U;
     INTEGER K;
     PROCEDURE DOMULT(N,A,IA,B,IB);
     VALUE N; INTEGER N; ARRAY A,IA,B,IB[0];
     BEGIN COMMENT   DOMULT IS DESIGNED TO DO
               THE ROW-COLUMN MULTIPLICATIONS WHICH
               ARE NEEDED IN CMPLXLINTRAN;
          INTEGER K; REAL RE,IM; RE←IM←0;
          FOR K←1 STEP 1 UNTIL N DO
          BEGIN RE←A[K]×B[K]-IA[K]×IB[K]+RE;
               IM←A[K]×IB[K]+B[K]×IA[K]+IM
          END; A[0]←RE; IA[0]←IM
     END OF THE DOMULT PROCEDURE;
     FOR K←1 STEP 1 UNTIL N DO
     BEGIN DOMULT(N,X,IX,A[K,*],IA[K,*]);
          Y[K]←X[0]; IY[K]←IX[0]
     END OF THE LOOP ON K
END OF THE CMPLXLINTRAN PROCEDURE;
INTEGER PROCEDURE MOSTLD(N,A);
VALUE N; INTEGER N; ARRAY A[0,0];
BEGIN COMMENT   A IS ASSUMED TO HAVE INDEX UPPER
          BOUND EQUAL TO N, AND TO HAVE BEEN THE TARGET
          MATRIX IN A CALL OF COSQBUILDER OR CMPLX-
          COSQBUILDER.  THUS, THERE EXISTS AN ORDERED
          COLLECTION OF N VECTORS WHOSE "COSINE-SQUARED"
          MATRIX (SEE THE COMMENTS IN COSQBUILDER
          AND CMPLXCOSQBUILDER ) IS A.
               THIS PROCEDURE INSERTS INTO THE
          IDENTIFIER, MOSTLD, THE NUMBER OF THE VECTOR
          WHICH IS MOST LINEARLY DEPENDENT UPON ITS
          NEIGHBOURS.  THIS VECTOR IS DETERMINED BY
          FIRST SCANNING THE ABOVE-DIAGONAL PORTION
          OF A TO FIND A PAIR, (I,J), SUCH THAT A[I,J]
          IS AS LARGE AS ANY ENTRY IN THIS PORTION.
          THE VALUE OF THIS A[I,J] IS THEN STORED INTO
          MAXAIJ.
               FOR EACH PAIR (P,Q) SUCH THAT A[P,Q]
          EQUALS MAXAIJ, THE NORMS OF ROW P AND ROW
          Q OF A ARE COMPARED, AND THE LARGER NORM,
          TOGETHER WITH ITS ASSOCIATED INDEX, IS
          DISTINGUISHED.  IN THE CASE OF EQUAL NORMS,
          THE INDEX DISTINGUISHED WILL BE THE LARGER OF
          P AND Q.  THIS BEING THE CASE, THE VECTOR
```

```
            WHICH IS "MOST LINEARLY DEPENDENT" UPON ITS
            NEIGHBOURS IS GOTTEN BY CONSIDERING THESE
            PAIRS OF DISTINGUISHED NORMS AND INDICES.  THE
            VECTOR CORRESPONDING TO THE LARGEST SUCH
            NORM IS THE ONE CHOSEN, AND IN CASE OF A
            TIE, THE CANDIDATE HAVING THE LARGEST INDEX IS
            SELECTED.  IT IS THIS INDEX, THEN, WHICH IS
            STORED INTO THE IDENTIFIER, MOSTLD;
        INTEGER I,J,K,KMAX,INX; REAL MAXAIJ,NORM,MAXNORM;
        INTEGER PROCEDURE MAXINX(A,I,J,N,NORM);
        VALUE I,J,N; INTEGER I,J,N; REAL NORM; ARRAY A[0,0];
        BEGIN COMMENT   THIS PROCEDURE CONSIDERS THE
                SUMS OF THE ENTRIES IN ROWS I AND
                J OF THE MATRIX, A.  THE
                LARGER SUM IS STORED INTO NORM, AND
                THE ASSOCIATED INDEX IS STORED INTO
                MAXINX.
                    IN THE CASE OF A TIE, THE
                LARGER OF I AND J IS STORED INTO
                THE IDENTIFIER, MAXINX;
            INTEGER K; REAL NRMI,NRMJ;
            NRMI←NRMJ←0;
            FOR K←1 STEP 1 UNTIL N DO
            BEGIN NRMI←A[I,K]+NRMI;
                NRMJ←A[J,K]+NRMJ
            END OF THE LOOP ON K; NORM←NRMJ;
            IF NRMI>NRMJ THEN
            BEGIN NORM←NRMI; MAXINX←I END
                ELSE
            IF NRMI=NRMJ THEN
            MAXINX←(IF I>J THEN I ELSE J)
                ELSE MAXINX←J
        END OF THE MAXINX PROCEDURE;
        MAXAIJ←MAXNORM←KMAX←0;
        FOR I←1 STEP 1 UNTIL N DO
        FOR J←I+1 STEP 1 UNTIL N DO
        IF MAXAIJ≤NORM←A[I,J] THEN MAXAIJ←NORM;
        FOR I←1 STEP 1 UNTIL N DO
        FOR J←I+1 STEP 1 UNTIL N DO
        IF A[I,J]=MAXAIJ THEN
        BEGIN INX←MAXINX(A,I,J,N,NORM);
            IF NORM>MAXNORM THEN
            BEGIN MAXNORM←NORM; KMAX←INX END
                ELSE
            IF NORM=MAXNORM AND INX>KMAX THEN
                KMAX←INX
        END OF THE MAIN SCANNING PROCESS;
        MOSTLD←KMAX
    END OF THE MOSTLD PROCEDURE;
    PROCEDURE CMPLXCOSQBUILDER(N,A,B,C,ROWCOL);
    VALUE N,ROWCOL; INTEGER N; REAL ROWCOL; ARRAY A,B,C[0,0];
    BEGIN COMMENT   A, B, AND C ARE ASSUMED TO HAVE
            INDEX UPPER BOUND EQUAL TO N.  IT IS
            OF INTEREST TO CONSIDER THE COMPLEX
            MATRIX, S, WHOSE IJTH ENTRY HAS REAL
            AND IMAGINARY PARTS A[I,J] AND B[I,J],
```

```
                RESPECTIVELY, FOR I,J=1, ..., N.  S IS
                REGARDED AS INPUT TO THIS PROCEDURE.
                     IF ROWCOL="ROWS", THEN EACH ROW OF S
                WILL BE REGARDED AS A COMPLEX VECTOR WITH
                N ENTRIES.  OTHERWISE, THE COLUMNS OF S
                WILL BE SO REGARDED.  IN EITHER CASE, AN
                ORDERED SET OF N COMPLEX VECTORS HAS
                BEEN DISTINGUISHED.
                     FOR I,J=1, ..., N THE PROCEDURE STORES
                INTO C[I,J] THE "COSINE" OF THE ANGLE
                BETWEEN THE ITH AND JTH VECTORS IN THE
                DISTINGUISHED SET.  HERE THE MODULUS OF
                THE INNER PRODUCT OF TWO VECTORS IS TAKEN
                TO BE EQUAL TO THE PRODUCT OF THE NORMS
                OF THE TWO TIMES THE COSINE OF THE
                ANGLE BETWEEN THEM;
           INTEGER I,J,IM1; BOOLEAN ANYMORE;
           REAL T; LABEL TRANSPOSE,EXIT,DOIT;
           IF (ANYMORE←ROWCOL≠"ROWS") THEN
         TRANSPOSE:   FOR I←1 STEP 1 UNTIL N DO
           BEGIN IM1←I-1; FOR J←1 STEP 1 UNTIL IM1 DO
                BEGIN T←A[I,J]; A[I,J]←A[J,I];
                     A[J,I]←T; T←B[I,J];
                     B[I,J]←B[J,I]; B[J,I]←T
                END OF THE LOOP ON J
           END ELSE GO TO DOIT;
           IF NOT ANYMORE THEN GO TO EXIT;
         DOIT:   FOR I←1 STEP 1 UNTIL N DO
           BEGIN IM1←I-1; FOR J←1 STEP 1 UNTIL IM1 DO
                C[J,I]←MODSQOFINRPROD
                          (N,A[I,*],B[I,*],A[J,*],B[J,*]);
                C[I,I]←MODOFINRPROD
                          (N,A[I,*],B[I,*],A[I,*],B[I,*])
           END OF THE LOOP ON I;
           FOR I←1 STEP 1 UNTIL N DO
           FOR J←I+1 STEP 1 UNTIL N DO
           C[J,I]←C[I,J]←C[I,J]/(C[I,I]×C[J,J]);
           FOR I←1 STEP 1 UNTIL N DO C[I,I]←1;
           IF ANYMORE THEN
           BEGIN ANYMORE←FALSE; GO TO TRANSPOSE END;
         EXIT:
      END OF THE CMPLXCOSQBUILDER PROCEDURE;
      PROCEDURE TRANSPOSE(N,A);
      VALUE N; INTEGER N; ARRAY A[0,0];
      BEGIN COMMENT   A IS ASSUMED TO HAVE INDEX
                UPPER BOUNDS EQUAL TO N.  THIS PROCEDURE
                TRANSPOSES THE ROWS AND COLUMNS OF A;
           INTEGER I,J; REAL T;
           FOR I←0 STEP 1 UNTIL N DO
           FOR J←I+1 STEP 1 UNTIL N DO
           BEGIN T←A[I,J]; A[I,J]←A[J,I]; A[J,I]←T END
      END OF THE TRANSPOSE PROCEDURE;
REAL PROCEDURE CMPLXHOMOSOLVER(N,A,IA,B,IB,X,IX);
VALUE N; INTEGER N; ARRAY A,IA,B,IB[0,0],X,IX[0];
BEGIN COMMENT   ALTHOUGH THE ACTUAL INDEX LOWER BOUNDS
           ARE ZERO, THIS PROCEDURE ONLY REFERENCES ENTRIES
```

```
        IN THE ARRAYS A, IA, B, IB, X, AND IX WHICH
        CORRESPOND TO INDICES IN THE RANGE FROM ONE TO
        N.  THE MATRIX, B, IS USED FOR TEMPORARY STORAGE.
             CONSIDER THE COMPLEX MATRIX, U, SUCH THAT
        U[I,J] HAS REAL AND IMAGINARY PARTS, A[I,J]
        AND IA[I,J], RESPECTIVELY.  FURTHER, LET T DENOTE
        THE COMPLEX VECTOR DEFINED BY SAYING THAT
        T[K] HAS REAL AND IMAGINARY PARTS, X[K] AND
        IX[K], RESPECTIVELY.  ASSUMING THAT U IS
        SINGULAR AND OF RANK (N-1), CMPLXHOMOSOLVER
        ATTEMPTS TO FIND A NON-TRIVIAL SOLUTION TO THE
        EQUATION, UT=0.  SUCH A SOLUTION IS, OF COURSE,
        UNIQUE UP TO MULTIPLICATION BY A SCALAR.
             THE PROCEDURE BEGINS BY THROWING OUT THE ROW
        OF U WHICH IS MOST LINEARLY DEPENDENT UPON ITS
        NEIGHBOURS.  THIS LEAVES A SET OF (N-1) EQUATIONS
        IN N UNKNOWNS OF THE FORM VT=0, WHERE V IS
        THE MATRIX GOTTEN BY THROWING OUT A ROW OF U.
             NEXT, THE COLUMNS OF V ARE EXAMINED, AND AN
        (N-1) BY (N-1) MATRIX, W, IS FORMED BY REMOVING
        THE COLUMN OF V WHICH IS MOST LINEARLY DEPENDENT
        UPON ITS NEIGHBOURS.  LETTING THIS COLUMN BE
        DENOTED BY Y, THE PROCEDURE THEN SOLVES THE SYSTEM
        OF EQUATIONS, WZ=-Y, WHERE Z DENOTES THE VECTOR
        GOTTEN FROM T BY DELETING THE ENTRY WHICH CORRES-
        PONDS TO THE COLUMN, Y.
             FINALLY, THE VECTOR, T, IS FILLED WITH VALUES
        FROM Z, WHEREVER POSSIBLE, AND THE ENTRY CORRES-
        PONDING TO THE COLUMN, Y, IS GIVEN THE VALUE
        ONE.  THIS, THEN, IS THE SOLUTION TO THE
        EQUATION, UT=0.
             TO GIVE SOME INDICATION AS TO THE AMOUNT
        OF CANCELLATION INVOLVED IN COMPUTING THE DETERMI-
        NANT OF W, ABOVE, THE PROCEDURE COMPUTES THE PRODUCT
        OF THE MODULI OF THE NORMS OF THE ROWS OF W,
        AND OF THE COLUMNS OF W.  THE AVERAGE OF THESE
        TWO PRODUCTS IS THEN COMPUTED, AND THE DETERMINANT
        OF W, DIVIDED BY THIS AVERAGE, IS INSERTED
        INTO THE IDENTIFIER, CMPLXHOMOSOLVER.
             PROCEDURES REFERENCED BY CMPLXHOMOSOLVER
        ARE :  MODOFINRPROD, CMPLXCOSQBUILDER,
        MOSTLD, CMPLXLINTRAN, TRANSPOSE, AND CMPLXINVERSE;
INTEGER I,J,K,LDROW,LDCOL,NM1;
REAL NRMR,NRMC,NORM,MODET;
PROCEDURE MOVECTOR(N,A,B);
VALUE N; INTEGER N; ARRAY A,B[0];
BEGIN COMMENT   A AND B ARE ASSUMED TO HAVE INDEX
             UPPER BOUNDS EQUAL TO N.  THIS PROCEDURE
             COPIES A[K] INTO B[K] FOR K=0, ..., N;
        INTEGER K;
        FOR K←0 STEP 1 UNTIL N DO B[K]←A[K]
END OF THE MOVECTOR PROCEDURE;
LABEL EXIT;
COMMENT   HERE THE EXECUTABLE STATEMENTS BEGIN;
CMPLXHOMOSOLVER←1;
IF N = 1 THEN
```

```
BEGIN X[1]←1; IX[1]←0; GO TO EXIT END;
CMPLXCOSQBUILDER(N,A,IA,B,"ROWS"); LDROW←MOSTLD(N,B);
COMMENT   LDROW NOW EQUALS THE NUMBER OF THE ROW
     IN THE MATRIX, U, WHICH IS MOST LINEARLY
     DEPENDENT UPON ITS NEIGHBOURS;
MOVECTOR(N,A[LDROW,*],A[0,*]);
MOVECTOR(N,IA[LDROW,*],IA[0,*]);
FOR K←LDROW+1 STEP 1 UNTIL N DO
BEGIN MOVECTOR(N,A[K,*],A[K-1,*]);
     MOVECTOR(N,IA[K,*],IA[K-1,*])
END OF THE K LOOP;
FOR K←1 STEP 1 UNTIL N DO A[N,K]←IA[N,K]←0;
COMMENT   NOW THE LDROW-TH ROW OF U HAS BEEN
     COPIED INTO THE ZERO-TH ROW OF U, THE
     REMAINING ROWS HAVE BEEN SHUFFLED DOWN, AND
     THE N-TH ROW HAS BEEN FILLED WITH ZEROES;
CMPLXCOSQBUILDER(N,A,IA,B,"COLUMNS");
LDCOL←MOSTLD(N,B); NM1←N-1;
FOR I←1 STEP 1 UNTIL NM1 DO
BEGIN MOVECTOR(LDCOL,A[I,*],B[I,*]);
     MOVECTOR(LDCOL,IA[I,*],IB[I,*]);
     FOR J←LDCOL+1 STEP 1 UNTIL N DO
     BEGIN B[I,J-1]←A[I,J];
          IB[I,J-1]←IA[I,J]
     END OF THE J LOOP
END OF THE I LOOP;
COMMENT   NOW LOCOL IS THE NUMBER OF THE COLUMN
     OF THE MATRIX, U, WHICH IS MOST LINEARLY
     DEPENDENT UPON ITS NEIGHBOURS.  COLUMNS ONE
     THRU (LDCOL-1) OF U HAVE BEEN COPIED INTO
     THE CORRESPONDING COLUMNS OF THE MATRIX
     WITH REAL AND IMAGINARY PARTS B AND IB,
     RESPECTIVELY.  FURTHER, COLUMNS (LDCOL+1) THRU
     N OF U HAVE BEEN COPIED INTO COLUMNS
     LOCOL THRU (N-1) OF THE (B,IB) MATRIX;
NRMR←NRMC←1; TRANSPOSE(N,B); TRANSPOSE(N,IB);
FOR K←1 STEP 1 UNTIL NM1 DO
NRMC←NRMC×MODOFINRPROD
          (NM1,B[K,*],IB[K,*],B[K,*],IB[K,*]);
NORM←(SQRT(NRMC)+SQRT(NRMR))/2;
COMMENT   NOW THE PRODUCT OF THE SQUARES OF THE
     MODULI OF THE NORMS OF THE ROWS OF (B,IB)
     HAS BEEN STORED INTO NRMR.  THE CORRESPONDING
     PRODUCT FOR THE COLUMNS OF (B,IB) HAS BEEN
     STORED INTO NRMC.
          THE AVERAGE OF THE SQUARE ROOTS OF NRMR
     AND NRMC HAS BEEN STORED INTO NORM;
MODET←CMPLXINVERSE(NM1,B,IB);
FOR K←1 STEP 1 UNTIL NM1 DO
BEGIN B[0,K]←-A[K,LDCOL];
     IB[0,K]←-IA[K,LDCOL]
END OF COPYING OVER THE MOSTLD COLUMN;
CMPLXLINTRAN(NM1,B,IB,B[0,*],IB[0,*],X,IX);
FOR K←NM1 STEP -1 UNTIL LDCOL DO
BEGIN X[K+1]←X[K]; IX[K+1]←IX[K] END;
X[LDCOL]←1; IX[LDCOL]←0;
```

```
      CMPLXHOMOSOLVER←MODET/NORM;
      COMMENT   NOW THE SOLUTION TO UT=0 HAS BEEN
           STORED INTO T, AND THE DETERMINANT-OVER-NORM
           QUANTITY HAS BEEN INSERTED INTO THE IDENTIFIER,
           CMPLXHOMOSOLVER.  WHAT REMAINS TO BE DONE
           IS THE RESTORING OF ROWS ONE THRU N OF U;
      FOR K←NM1 STEP -1 UNTIL LDROW DO
      BEGIN MOVECTOR(N,A[K,*],A[K+1,*]);
           MOVECTOR(N,IA[K,*],IA[K+1,*])
      END;
      MOVECTOR(N,A[0,*],A[LDROW,*]);
      MOVECTOR(N,IA[0,*],IA[LDROW,*]);
    EXIT:
END OF THE CMPLXHOMOSOLVER PROCEDURE;
INTEGER PROCEDURE GETRANSIENTSOLUTION
(M,N,MUVEK,NUVEK,ALFA,BETA,GAMMA,A,IA,B,IB,CMTX,ICMTX,
     POLY,MODE,EPS); VALUE N,POLY,MODE,EPS; BOOLEAN POLY,MODE;
         INTEGER M,N; ARRAY MUVEK,NUVEK[0],
ALFA,BETA,GAMMA,A,IA,B,IB,CMTX,ICMTX[0,0];  REAL EPS;
BEGIN COMMENT   THE MATRICES, CMTX AND ICMTX, MUST
          HAVE 2N ROWS OF N ELEMENTS EACH, I.E., THEY
          MUST BE 2N×N MATRICES.  THE OTHER MATRICES
          MUST BE N×N, AND THE VECTORS, MUVEK AND NUVEK,
          MUST HAVE UPPER BOUNDS EQUAL TO 2×N.  THE
          MATRICES, A, IA, B, AND IB ARE USED FOR
          TEMPORARY STORAGE.
               WHERE G IS THE MATRIX OF DIFFERENTIAL
          OPERATORS DEFINED BY G[I,J] = ALFA[I,J]×D*2
          + BETA[I,J]×D + GAMMA[I,J], FOR I, J=1, ..., N
          ( D DENOTES DIFFERENTIATION WITH RESPECT TO
          TIME ). THIS PROCEDURE FINDS THE M INDEPENDENT
          SOLUTIONS TO THE EQUATION GQ=<NULL VECTOR>.
          OF COURSE, M≤2N, AND THE VALUE OF M IS
          ALWAYS STORED INTO THE PARAMETER, M, PRIOR
          TO EXIT.
               THE BASIC SOLUTIONS TO THE ABOVE EQUATION
          HAVE THE FORM C × EXP( LAMBDA × TIME ),
          WHERE C IS A VECTOR WITH N COMPLEX ENTRIES,
          AND LAMBDA IS A COMPLEX NUMBER WITH REAL AND
          IMAGINARY PARTS, MU AND NU, RESPECTIVELY.  THE
          PROCEDURE COMPUTES ALL SUCH VECTORS, C, AND
          STORES THEM AS ROWS ONE THRU M OF THE
          COMPLEX MATRIX WITH REAL AND IMAGINARY PARTS,
          CMTX AND ICMTX, RESPECTIVELY.  IN EACH CASE,
          THE CORRESPONDING LAMBDA IS STORED INTO
          THE CORRESPONDING POSITION OF THE COMPLEX
          VECTOR WHOSE REAL AND IMAGINARY PARTS ARE
          MUVEK AND NUVEK, RESPECTIVELY.
               IF THE PROCEDURE IS SUCCESSFUL IN DOING
          ITS WORK, THEN A ZERO IS INSERTED INTO THE
          IDENTIFIER, GETRANSIENTSOLUTION, PRIOR TO
          EXIT.  OTHERWISE, THE INDEX CORRESPONDING
          TO THE VECTOR, C, WHICH WAS BEING COMPUTED
          AT THE TIME OF THE HANGUP WILL BE INSERTED
          BEFORE EXITING.
               THIS PROCEDURE MAKES EXPLICIT CALLS ON
```

```
        GETDETPOLY, FINDPOLYORDERANDNORMALIZE,
        SCALECOEFFICIENTS, GETPOLYZEROS, UNSCALE-
        THEROOTS, AND CMPLXHOMOSOLVER;
    INTEGER I,J,K,P; REAL SCALE,REL,IML,AIJ,BIJ;
    LABEL EXIT,EOL; BOOLEAN STABLE;
    COMMENT   HERE BEGIN THE EXECUTABLE STATEMENTS;
    GETDETPOLY(N,ALFA,BETA,GAMMA,MUVEK);
    COMMENT   NOW MUVEK CONTAINS THE COEFFICIENTS
        OF THE DETERMINANT POLYNOMIAL;
        P←8;

    FOR I←8 STEP -1 UNTIL 0 DO
    IF MUVEK[I]=0 THEN P←I-1 ELSE GO TO EOL;
EOL:   IF BOOLEAN(P) THEN
    BEGIN WRITE(PRIMARY,FMTODD);
        IF POLY OR MODE THEN WRITE(SECNDRY,FMTODD);
    END          ELSE
    BEGIN  FOR K←0 STEP 1 UNTIL P DO
        NUVEK[K]←MUVEK[P-K];
        IF NUVEK[0] < 0 THEN
        FOR K←0 STEP 1 UNTIL P DO
        BEGIN NUVEK[K]←-NUVEK[K]; MUVEK[K]←-MUVEK[K]; END;
       ROUTHH(CMTX,P,NUVEK,RR,STABLE,ROW);
        WRITE(PRIMARY,SWITFMT[I←  REAL(STABLE)],SWITLST[I]);
        IF POLY OR MODE THEN
        WRITE(SECNDRY,SWITFMT[I],SWITLST[I]);
    END;
        IF ROUTH THEN GO TO EXIT;
        IF ORDER THEN
        FOR I←ODDI+1 STEP 1 UNTIL 8 DO MUVEK[I]←0;
    IF POLY THEN WRITE(SECNDRY,FMTPOLY,
        FOR K←0 STEP 1 UNTIL 8 DO MUVEK[K]);
    FOR K←0 STEP 1 UNTIL 2×N DO NUVEK[K]←0;
    FINDPOLYORDERANDNORMALIZE(2×N,MUVEK,NUVEK,P);
    SCALECOEFFICIENTS(P,MUVEK,NUVEK,SCALE);
    GETPOLYZEROS(P,MUVEK,NUVEK,EPS     );
    UNSCALETHEROOTS(P,MUVEK,NUVEK,SCALE);
    COMMENT   NOW MUVEK AND NUVEK CONTAIN THE
        REAL AND IMAGINARY PARTS, RESPECTIVELY, OF
        THE ROOTS OF THE DETERMINANT POLYNOMIAL.  THESE
        ROOTS ARE, OF COURSE, THE LAMBDAS MENTIONED
        IN THE MAIN COMMENT, ABOVE.
            NEXT, ALL SUCH ROOTS WITH MODULUS ZERO
        WILL BE THROWN OUT, AND THE M REMAINING
        ONES WILL BE SHUFFLED DOWN IN THE MUVEK-
        NUVEK PAIR;  M←P;
    FOR K←P STEP -1 UNTIL 1 DO
    IF MUVEK[K]*2+NUVEK[K]*2=0 THEN
    BEGIN M←M-1;
        FOR J←K+1 STEP 1 UNTIL P DO
        BEGIN MUVEK[J-1]←MUVEK[J];
            NUVEK[J-1]←NUVEK[J]
        END OF THE SHUFFLEDOWN
    END OF ZERO MODULUS CASE;
    COMMENT   NOW M IS PROPERLY SET UP, AND THE
        NON-ZERO ROOTS OF THE DETERMINANT POLYNOMIAL
```

```
                    ARE THE ONE THRU M-TH ENTRIES OF THE
                    MUVEK-NUVEK PAIR;
                IF NOT MODE THEN
                BEGIN GETRANSIENTSOLUTION←0; GO TO EXIT; END;
                FOR K←1 STEP 1 UNTIL M DO
                BEGIN COMMENT   EACH PASS THRU THIS LOOP CAUSES
                        THE "C-VECTOR" CORRESPONDING TO THE
                        LAMBDA WITH REAL AND IMAGINARY PARTS,
                        MUVEK[K] AND NUVEK[K], RESPECTIVELY,
                        TO BE STORED AS THE K-TH ROW OF
                        THE COMPLEX MATRIX WITH REAL AND IMAGINARY
                        PARTS, CMTX AND ICMTX, RESPECTIVELY;
                    REL←MUVEK[K]; IML←NUVEK[K];
                    FOR I←1 STEP 1 UNTIL N DO
                    FOR J←1 STEP 1 UNTIL N DO
                    BEGIN A[I,J]←(REL*2-IML*2)×(AIJ←ALFA[I,J])
                            +(BIJ←BETA[I,J])×REL+GAMMA[I,J];
                        IA[I,J]←(AIJ×REL×2+BIJ)×IML
                    END OF SETTING UP A AND IA;
                    IF CMPLXHOMOSOLVER(N,A,IA,B,IB,
                        CMTX[K,*],ICMTX[K,*]) ≤ 1.0@-11 THEN
                    BEGIN GETRANSIENTSOLUTION←K;
                        GO TO EXIT
                    END OF THE HANGUP CASE
                END OF THE LOOP ON K;
                GETRANSIENTSOLUTION←0;
              EXIT:
            END OF THE GETRANSIENTSOLUTION PROCEDURE;
    COMMENT   **********  EXECUTABLE  STATEMENTS  **********;
    FOR I←1,2,3 DO TO[I]←TIME(1);
    EOFBOOL ← FALSE;  G←32.17       ×12;  PI2←(2×(PI←ARCTAN(1)×4));
    READ(CR[NO],/,STRING);
    IF STRING = "SIGFIG" THEN
    BEGIN READ(CR,/,STRING,EPS); EPS←10*(-2×EPS); END ELSE EPS←1.0@-21;
    READ(CR[NO],/,STRING);
    IF (ORDER←STRING="ORDER")THEN READ(CR,/,STRING,ODI);
    READ(CR[NO],/,STRING);
    IF (ROUTH← STRING="ROUTH") THEN READ(CR);
    READ(CR,/,L,L1,L2,W,IP,IT); RAD ← 180/PI; POLY←MODE←FALSE;
        ALAB:   READ(CR,/,WO,DW,WM); READ(CR,/,K1X,K2X,K1Y,K2Y);
    READ(CR,/,C1X,C2X,C1Y,C2Y); READ(CR,/,D1X,D2X,D1Y,D2Y);
    READ(CR,/,R1X,R2X,R1Y,R2Y);          POLY←MODE←FALSE;
BLAB:   READ(CR[NO],/,STRING)[EOF];
    IF STRING="POLY" THEN BEGIN POLY←TRUE; READ(CR); GO TO BLAB; END;
    IF STRING="MODE" THEN BEGIN MODE←TRUE; READ(CR); GO TO BLAB; END;
    GO TO PROCESS;
EOF:   EOFBOOL← TRUE;
        PROCESS:   WRITE(PRIMARY,FMTECHO,LSTECHO);
    IF ROUTH THEN POLY←MODE←FALSE;
    IF POLY OR MODE THEN
    BEGIN WRITE(SECNDRY,FMTECHO,LSTECHO);
        IF POLY THEN
        BEGIN WRITE(PRIMARY,PLYECHO); WRITE(SECNDRY,PLYECHO); END;
        IF MODE THEN
        BEGIN WRITE(PRIMARY,MODECHO); WRITE(SECNDRY,MODECHO); END;
        WRITE(SECNDRY[DBL]);
```

```
END OF THE POLY MODE ECHO;
WRITE(PRIMARY[DBL]);
     M← W/G ;    DM1 ← WM1 / G  ;  DM2 ← WM2/G  ;
     RPP ← IP / M ;  RTT← IT / M  ;
     RP ● RPP/ (L× L )  ;  RT ← RTT / ( L× L )  ;
     L11 ← L1 / L  ; L22 ← L2 / L  ;
     K1XX ← K1X / M   ;            K2XX ← K2X /M  ;
     K1YY ← K1Y /M  ;              K2YY ← K2Y /M  ;
     C1XX ← C1X / M  ;             C2XX ← C2X / M  ;
     C1YY ← C1Y / M  ;             C2YY ● C2Y  / M    ;
     R1XX ← R1X / M  ;             R2XX ←  R2X / M  ;
     R1YY ← R1Y / M ;              R2YY ← R2Y / M  ;
     D1XX ● D1X / M  ;             D2XX ← D2X / M   ;
     D1YY ← D1Y / M  ;             D2YY ← D2Y / M   ;
FOR W ← WO STEP DW UNTIL WM DO
BEGIN WRITE(PRIMARY[NO],FMTSPD,W); J← O; I←1; WHRLBOL←FALSE;
     IF MODE OR POLY THEN WRITE(SECNDRY[NO],FMTSPD,W); S←W×PI2;
     FOR R←L22,L11,0,0,0,0,L22,L11,-RT,RT,0,0,0,0,-RT,RT  DO
     BEGIN IF J=4 THEN BEGIN J← 0;I←I+1; END;
          A[I,J←J+1]←R;
     END OF THE A MATRIX SETUP; J← 0; I←1;

     FOR R←C1XX,C2XX,D1YY,D2YY,D1XX,D2XX,C1YY,C2YY,
          -C1XX×L11,C2XX×L22,-(RP×S+D1YY×L11),
          RP×S+D2YY×L22,RP×S-D1XX×L11,-(RP×S-D2XX×L22),
          -C1YY×L11, C2YY×L22 DO
     BEGIN IF J=4 THEN BEGIN J← 0;I←I+1; END;
          B[I,J←J+1]←R;
     END OF THE B MATRIX SETUP; J← 0; I←1;

     FOR R←K1XX,K2XX,R1YY,R2YY,R1XX,R2XX,K1YY,K2YY,
          -K1XX×L11, K2XX×L22, -R1YY×L11,R2YY×L22,
          -R1XX×L11,R2XX×L22,-K1YY×L11,K2YY×L22 DO
     BEGIN IF J=4 THEN BEGIN J← 0;I←I+1; END;
     C[I,J←J+1]←R;
     END OF THE C MATRIX SETUP;
     IF I← GETRANSIENTSOLUTION
          (M,4,MUVEK,NUVEK,A,B,C,AR,AI,BR,BI,CMTX,ICMTX,
                    POLY,MODE,EPS ) ≠ 0 THEN
     BEGIN WRITE(PRIMARY,ERRFMT,I);
          WRITE(PRIMARY[PAGE]);
          IF MODE OR POLY THEN
          BEGIN WRITE(SECNDRY,ERRFMT,I); WRITE(SECNDRY[PAGE]); END;
          IF EOFBOOL THEN GO TO EXIT; GO TO ALAB;
     END OF THE ERROR QUIT;
     IF ROUTH THEN GO TO SLP;
     WRITE(PRIMARY,FMTROOT,LSTROOT);
     WRITE(WHRARY[*],ALLWHRL,LSTALL);
     WRITE(PRIMARY,FMTFREQ,LSTFREQ);
     FOR I←1 STEP 1 UNTIL M DO
     IF MUVEK[I] ≥ 0 THEN WHRLBOL← TRUE
                    ELSE
     BLANK(WHRARY[1],I-1);
     IF WHRLBOL THEN
     BEGIN WRITE(PRIMARY,FMTWHRL); WRITE(PRIMARY,15,WHRARY[*]) END;
     IF MODE OR POLY THEN
```

```
                BEGIN WRITE(SECNDRY,FMTROOT,LSTROOT);
                     WRITE(SECNDRY,FMTFREQ,LSTFREQ);
                     IF WHRLBOL THEN
                     BEGIN WRITE(SECNDRY,FMTWHRL);
                          WRITE(SECNDRY,15,WHRARY[*]); WRITE(SECNDRY[DBL]);
                     END OF THE WHRIL SECONDARY WRITE;
                     IF MODE THEN
                     BEGIN  WRITE(SECNDRY,FMTMODE);
                     FOR K←1 STEP 1 UNTIL M DO
                     WRITE(SECNDRY,FM2    ,LSTMODE); WRITE(SECNDRY[DBL]);
                     END;
                END OF THE POLY MODE PRINT OUT; WRITE(PRIMARY[DBL]);
         SLP:
           END OF THE SPEED LOOP;
           IF NOT EOFBOOL THEN
           BEGIN WRITE(PRIMARY[PAGE]);
                IF POLY OR MODE THEN
                WRITE(SECNDRY[PAGE]);
                GO TO ALAB;
           END OF THE PROCESSING;
        EXIT:   TIMEANDATE(TO,PRIMARY,"GIVTME");
           IF POLY OR MODE THEN TIMEANDATE(TO,SECNDRY,"GIVTME");
       END OF THE ROTSTAB PROGRAM.
     ARCTAN IS SEGMENT NUMBER 0030,PRT ADDRESS IS 0246
     EXP IS SEGMENT NUMBER 0031,PRT ADDRESS IS 0227
     LN IS SEGMENT NUMBER 0032,PRT ADDRESS IS 0226
     SQRT IS SEGMENT NUMBER 0033,PRT ADDRESS IS 0220
     OUTPUT(W) IS SEGMENT NUMBER 0034,PRT ADDRESS IS 0202
     BLOCK CONTROL IS SEGMENT NUMBER 0035,PRT ADDRESS IS 0005
     INPUT(W) IS SEGMENT NUMBER 0036,PRT ADDRESS IS 0250
     X TO THE I IS SEGMENT NUMBER 0037,PRT ADDRESS IS 0230
     GO TO SOLVER  IS SEGMENT NUMBER 0038,PRT ADDRESS IS 0265
     ALGOL WRITE   IS SEGMENT NUMBER 0039,PRT ADDRESS IS 0014
     ALGOL READ    IS SEGMENT NUMBER 0040,PRT ADDRESS IS 0015
     ALGOL SELECT  IS SEGMENT NUMBER 0041,PRT ADDRESS IS 0016
COMPILATION TIME =  296 SECONDS.
NUMBER OF ERRORS DETECTED = 000. LAST ERROR ON CARD #
NUMBER OF SEQUENCE ERRORS COUNTED =   0.
NUMBER OF SLOW WARNINGS =   0.
PRT SIZE= 189;    TOTAL SEGMENT SIZE= 2783 WORDS.
DISK STORAGE REQ.=  119 SEGS.;    NO. SEGS.=  42.
ESTIMATED CORE STORAGE REQUIREMENT = 32000 WORDS.
```

TABLE D-I

L =	8.0000INCH	L1=	4.0000INCH	L2=	4.0000INCH
W=	24.5900LB	IP=	0.0496LB-IN-SEC2	IT=	1.7400LB-IN-SEC2

K1X=	220000.00LB/IN	K2X=	220000.00LB/IN	K1Y=	220000.00LB/IN	K2Y=	220000.00LB/IN
C1X=	13.80LB.SEC/IN	C2X=	13.80LB.SEC/IN	C1Y=	13.80LB.SEC/IN	C2Y=	13.80LB.SEC/IN
R1X=	-19350.00LB/IN	R2X=	-19350.00LB/IN	R1Y=	19350.00LB/IN	R2Y=	19350.00LB/IN
D1X=	0.0000LB.SEC/IN	D2X=	0.0000LB.SEC/IN	D1Y=	0.0000LB.SEC/IN	D2Y=	0.0000LB.SEC/IN

THE COEFFIEIENTS OF THE DETERMINANT POLYNUMIAL WILL BE GIVEN.
THE MODE SHAPE VECTORS WILL BE GIVEN.

---- SPEED = 450 RPS ---- STABLE

THE COEFFICIENTS OF THE DETERMINANT POLYNOMIAL (IN ASCENDING ORDER) ARE:

1.44504@+26 3.64815@+22 1.15888@+20 2.13374@+16 3.34749@+13 4.05858@+094118158.95352 250.33871 0.18217

THERE ARE 8 CHARACTERISTIC ROOTS, WITH REAL AND IMAGINARY PARTS AS FOLLOWS:

REAL	-40.90499	-212.88812	-100.78335	-332.51083	-100.78335	-212.88812	-332.51083	-40.90499
IMAG	2049.99874	-1969.40064	-2621.84609	-2621.84608	2621.84609	1969.40064	2621.84608	-2049.99874

THE NATURAL FREQUENCIES (IN CPS) ARE:

326.27	-313.44	-417.28	-417.28	417.28	313.44	417.28	-326.27

THE NATURAL FREQUENCIES (IN RPM) ARE:

18806.4	19576.0	25036.8

THE MODE SHAPE VECTORS ARE AS FOLLOWS:

------------ MODE 1 (NATURAL FREQUENCY = 3.26267@+02 CPS) ------------

VECTOR OF REAL PARTS	------	-0.909223690	1.000000000	0.315218401	-0.827592651
VECTOR OF IMAG PARTS	------	-0.135858846	0.000000000	-0.109729704	-0.112157331

------------ MODE 2 (NATURAL FREQUENCY = -3.13440@+02 CPS) ------------

VECTOR OF REAL PARTS	------	-0.897487828	1.000000000	0.262098593	-0.861462049
VECTOR OF IMAG PARTS	------	-0.128358638	0.000000000	-0.021025111	-0.903715198

------------ MODE 3 (NATURAL FREQUENCY = -4.17280@+02 CPS) ------------

VECTOR OF REAL PARTS	------	2.241921886	1.000000000	-0.019012139	0.505164903
VECTOR OF IMAG PARTS	------	-0.052257765	0.000000000	0.832758901	0.140161136

------------ MODE 4 (NATURAL FREQUENCY = -4.17280@+02 CPS) ------------

VECTOR OF REAL PARTS	------	2.199579545	1.000000000	0.020780826	0.418771000
VECTOR OF IMAG PARTS	------	-0.905894399	0.000000000	-0.763201164	-0.084746540

------------ MODE 5 (NATURAL FREQUENCY = 4.17280@+02 CPS) ------------

VECTOR OF REAL PARTS	------	2.241921885	1.000000000	-0.019012140	0.505164902
VECTOR OF IMAG PARTS	------	0.052257767	0.000000000	-0.832758897	-0.140161136

------------ MODE 6 (NATURAL FREQUENCY = 3.13440@+02 CPS) ------------

VECTOR OF REAL PARTS	------	-0.897487828	1.000000000	0.262098593	-0.861462048
VECTOR OF IMAG PARTS	------	0.128358638	0.000000000	0.021025111	0.903715198

------------ MODE 7 (NATURAL FREQUENCY = 4.17280@+02 CPS) ------------

VECTOR OF REAL PARTS	------	2.199579543	1.000000000	0.020780822	0.418770999
VECTOR OF IMAG PARTS	------	0.905894399	0.000000000	0.763201165	0.084746540

------------ MODE 8 (NATURAL FREQUENCY = -3.26267@+02 CPS) ------------

VECTOR OF REAL PARTS	------	-0.909223690	1.000000000	0.315218401	-0.827592651
VECTOR OF IMAG PARTS	------	0.135858847	0.000000000	0.109729704	0.112157333

TABLE D-I. - Continued.

L =	8.0000INCH	L1=	4.0000INCH	L2=	4.0000INCH
W=	24.5900LB	IP=	0.0496LB-IN-SEC2	IT=	1.7400LB-IN-SEC2

K1X=	220000.00LB/IN	K2X=	220000.00LB/IN	K1Y=	220000.00LB/IN	K2Y=	220000.00LB/IN
C1X=	50.00LB.SEC/IN	C2X=	50.00LB.SEC/IN	C1Y=	50.00LB.SEC/IN	C2Y=	50.00LB.SEC/IN
R1X=	-80000.00LB/IN	R2X=	-80000.00LB/IN	R1Y=	80000.00LB/IN	R2Y=	80000.00LB/IN
D1X=	0.0000LB.SEC/IN	D2X=	0.0000LB.SEC/IN	D1Y=	0.0000LB.SEC/IN	D2Y=	0.0000LB.SEC/IN

---- SPEED = 450 RPS ---- STABLE

THERE ARE 8 CHARACTERISTIC ROOTS, WITH REAL AND IMAGINARY PARTS AS FOLLOWS:

REAL	-819.86122	-293.58024	-99.67901	-1276.32623	-1276.32623	-293.58024	-819.86122	-99.67901
IMAG	1951.15220	2555.95076	-2031.75030	2555.95076	2555.95076	2555.95076	-1951.15220	2031.75030

THE NATURAL FREQUENCIES (IN CPS) ARE:

310.54	406.79	-323.36	406.79	-406.79	-406.79	-310.54	323.36

THE NATURAL FREQUENCIES (IN RPM) ARE:

18632.1	19401.8	24407.5

L =	8.0000INCH	L1=	4.0000INCH	L2=	4.0000INCH
W=	24.5900LB	IP=	0.0496LB-IN-SEC2	IT=	1.7400LB-IN-SEC2

K1X=	220000.00LB/IN	K2X=	220000.00LB/IN	K1Y=	220000.00LB/IN	K2Y=	220000.00LB/IN
C1X=	50.00LB.SEC/IN	C2X=	50.00LB.SEC/IN	C1Y=	50.00LB.SEC/IN	C2Y=	50.00LB.SEC/IN
R1X=	-102500.00LB/IN	R2X=	-102500.00LB/IN	R1Y=	102500.00LB/IN	R2Y=	102500.00LB/IN
D1X=	0.0000LB.SEC/IN	D2X=	0.0000LB.SEC/IN	D1Y=	0.0000LB.SEC/IN	D2Y=	0.0000LB.SEC/IN

---- SPEED = 450 RPS ---- STABLE

THERE ARE 8 CHARACTERISTIC ROOTS, WITH REAL AND IMAGINARY PARTS AS FOLLOWS:

REAL	-919.07018	-162.31176	-1407.59470	-1407.59470	-0.47005	-919.07018	-162.31176	-0.47005
IMAG	1971.45864	-2584.39921	2584.39921	-2584.39921	2052.05674	-1971.45864	2584.39921	-2052.05674

THE NATURAL FREQUENCIES (IN CPS) ARE:

313.77	-411.32	411.32	-411.32	326.59	-313.77	411.32	-326.59

THE NATURAL FREQUENCIES (IN RPM) ARE:

18826.0	19595.7	24679.2

TABLE D-I. - Continued.

```
          L =      8.0000INCH                    L1=      4.0000INCH                  L2=      4.0000INCH

          W=     24.5900LB                  IP=      0.0496LB-IN-SEC2               IT=      1.7400LB-IN-SEC2

   K1X=  220000.00LB/IN       K2X=  220000.00LB/IN       K1Y=  220000.00LB/IN       K2Y=  220000.00LB/IN
   C1X=      50.00LB.SEC/IN   C2X=      50.00LB.SEC/IN   C1Y=      50.00LB.SEC/IN   C2Y=      50.00LB.SEC/IN
   R1X= -127000.00LB/IN       R2X= -127000.00LB/IN       R1Y=  127000.00LB/IN       R2Y=  127000.00LB/IN
   D1X=      0.0000LB.SEC/IN  D2X=      0.0000LB.SEC/IN  D1Y=      0.0000LB.SEC/IN  D2Y=      0.0000LB.SEC/IN


 ----  SPEED =   450 RPS  ----                          UNSTABLE          RR =  -5.10159@+14   ROW =  5

 THERE ARE 8 CHARACTERISTIC ROOTS, WITH REAL AND IMAGINARY PARTS AS FOLLOWS:

 REAL       -1545.62498   104.10965    104.10965   -24.28149  -1545.62498  -1023.64988  -1023.64988    -24.28149
 IMAG        2621.07962 -2078.47786   2078.47786 -2621.07962  -2621.07962   1997.87976  -1997.87976   2621.07962

 THE NATURAL FREQUENCIES ( IN CPS ) ARE:

                 417.16     -330.80      330.80     -417.16      -417.16       317.97      -317.97       417.16

 THE NATURAL FREQUENCIES ( IN RPM ) ARE:

                19078.3     19848.0     25029.5
 THE WHIRL RATIOS ARE:

                            -0.73511     0.73511
```

L = 8.0000INCH L1= 4.0000INCH L2= 4.0000INCH

W= 24.5900LB IP= 0.0496LB-IN-SEC2 IT= 1.7400LB-IN-SEC2

K1X= 220000.00LB/IN	K2X= 220000.00LB/IN	K1Y= 220000.00LB/IN	K2Y= 220000.00LB/IN
C1X= 50.00LB.SEC/IN	C2X= 50.00LB.SEC/IN	C1Y= 50.00LB.SEC/IN	C2Y= 50.00LB.SEC/IN
R1X= -154000.00LB/IN	R2X= -154000.00LB/IN	R1Y= 154000.00LB/IN	R2Y= 154000.00LB/IN
D1X= 0.0000LB.SEC/IN	D2X= 0.0000LB.SEC/IN	D1Y= 0.0000LB.SEC/IN	D2Y= 0.0000LB.SEC/IN

---- SPEED = 450 RPS ---- UNSTABLE RR = -6.17172@+15 ROW = 5

THERE ARE 8 CHARACTERISTIC ROOTS, WITH REAL AND IMAGINARY PARTS AS FOLLOWS:

REAL	-1691.44354	121.53707	-1134.41357	214.87334	-1691.44354	214.87334	121.53707	-1134.41357
IMAG	2667.05110	-2667.05110	-2031.25985	2111.85795	-2667.05110	-2111.85795	2667.05110	2031.25985

THE NATURAL FREQUENCIES (IN CPS) ARE:

424.47	-424.47	-323.29	336.11	-424.47	-336.11	424.47	323.29

THE NATURAL FREQUENCIES (IN RPM) ARE:

19397.1	20166.8	25468.5

THE WHIRL RATIOS ARE:

-0.94328		0.74692		-0.74692	0.94328	

TABLE D-I. - Concluded.

L =	8.0000INCH	L1=	4.0000INCH	L2=	4.0000INCH
W=	24.5900LB	IP=	0.0496LB-IN-SEC2	IT=	1.7400LB-IN-SEC2

K1X=	220000.00LB/IN	K2X=	220000.00LB/IN	K1Y=	220000.00LB/IN	K2Y=	220000.00LB/IN
C1X=	50.00LB.SEC/IN	C2X=	50.00LB.SEC/IN	C1Y=	50.00LB.SEC/IN	C2Y=	50.00LB.SEC/IN
R1X=	-184500.00LB/IN	R2X=	-184500.00LB/IN	R1Y=	184500.00LB/IN	R2Y=	184500.00LB/IN
D1X=	0.0000LB.SEC/IN	D2X=	0.0000LB.SEC/IN	D1Y=	0.0000LB.SEC/IN	D2Y=	0.0000LB.SEC/IN

---- SPEED = 450 RPS ---- UNSTABLE RR = -3.68123@+16 ROW = 5

THERE ARE 8 CHARACTERISTIC ROOTS, WITH REAL AND IMAGINARY PARTS AS FOLLOWS:

REAL	-1848.15248	334.20846	-1253.74869	334.20846	-1848.15248	-1253.74869	278.24602	278.24602
IMAG	2724.30349	2153.73570	-2073.13759	-2153.73570	-2724.30349	2073.13759	-2724.30349	2724.30349

THE NATURAL FREQUENCIES (IN CPS) ARE:

433.59	342.78	-329.95	-342.78	-433.59	329.95	-433.59	433.59

THE NATURAL FREQUENCIES (IN RPM) ARE:

19797.0	20566.7	26015.2

THE WHIRL RATIOS ARE:

	0.76173		-0.76173			-0.96353	0.96353

```
L =     8.0000INCH            L1=    4.0000INCH            L2=    4.0000INCH
W=    24.5900LB              IP=    0.0496LB-IN-SEC2       IT=    1.7400LB-IN-SEC2

K1X=  220000.00LB/IN      K2X=  220000.00LB/IN      K1Y=  220000.00LB/IN      K2Y=  220000.00LB/IN
C1X=     100.00LB.SEC/IN  C2X=     100.00LB.SEC/IN  C1Y=     100.00LB.SEC/IN  C2Y=     100.00LB.SEC/IN
R1X= -220000.00LB/IN      R2X= -220000.00LB/IN      R1Y=  220000.00LB/IN      R2Y=  220000.00LB/IN
D1X=      0.0000LB.SEC/IN D2X=      0.0000LB.SEC/IN D1Y=      0.0000LB.SEC/IN D2Y=      0.0000LB.SEC/IN


---- SPEED =   450 RPS  ----                  UNSTABLE        RR =  -2.75969@+21   ROW =  7

THERE ARE 8 CHARACTERISTIC ROOTS, WITH REAL AND IMAGINARY PARTS AS FOLLOWS:

REAL     -1894.16749  -2942.87936  -2942.87936  -196.93357  -196.93357    55.08703  -1894.16749     55.08703
IMAG      1997.33317  -2515.55893   2515.55893  2515.55893  -2515.55893  2077.93128  -1997.33317  -2077.93128

THE NATURAL FREQUENCIES ( IN CPS ) ARE:

             317.89      -400.36       400.36      400.36      -400.36      330.71      -317.89      -330.71

THE NATURAL FREQUENCIES ( IN RPM ) ARE:

            19073.1      19842.8      24021.8
THE WHIRL RATIOS ARE:

                                                                          0.73492                  -0.73492


   MARCH  5, 1968.  TOTAL ELAPSED TIME IS    68 SECONDS.  PROCESSOR TIME IS    32 SECONDS.  I/O TIME IS    60 SECONDS.
```

APPENDIX E

LISTING AND SAMPLE OUTPUT OF COMPUTER PROGRAM STABIL4

```
BEGIN
COMMENT  THIS PROGRAM IS FOR THE STABILITY ANALYSIS OF 2 DEGREE FREEDOM
SYSTEM. THE REAL PART GIVES DAMPING RATE AND THE IMAGINARY PART , THE
NATURAL FREQ OF THE SYSTEM . IF REAL PART OF THE ROOT IS NEGATIVE THEN
THE SYSTEM IS STÁBLE . IF REAL PART IS ZERO THEN THE SYSTEM IS NEUTRAL .
IF THE REAL PART IS POSITIVE THEN THE SYSTEM IS UNSTABLE.
PROCEDURE FUNCTION IS A FUNCTION GENERATOR .
PROCEDURE COEFFICIENT CALCULATES THE COEFFICIENTS OF DIFFERENT POWERS
OF LAMBDA .  THE HIGHEST ONE STARTING WITH C[0] .
THE INPUT DATA ARE AS FOLLOWS
CARD1
1. N- HIGHEST POWER OF THE POLYNOMIAL
CARD2
1. W- MASS OF THE ROTOR (LBS)
CARD3
1. KX- STIFFNESS COEFF IN X DIRECTION (LB/IN)
2.KY - STIFFNESS COEFF IN Y DIRECTION (LB/IN)
CARD5
1. CX- DAMPING COEFF IN X DIRECTION (LB.SEC/IN)
1. CY- DAMPING COEFF IN Y DIRECTION (LB.SEC/IN)
CARD5
1. RX- CROSS COUPLING STIFFNESS IN X DIRECTION (LB/IN)
2. RY- CROSS COUPLING STIFFNESS IN Y DIRECTION (LB/IN)
CARD6
1. DX- CROSS COUPLING DAMPING COEFF IN X DIRECTION (LB.SEC/IN)
2. DY- CROSS COUPLING DAMPING COEFF IN Y DIRECTION (LB.SEC/IN)
CARD 7
1.L-LENGTH OF SHAFT (IN.)
CARD 8
1.IP- POLAR MOMENT OF INERTIA (LB-SEC-IN2)
2.IT-TRANSVERSE MOMENT OF INERTIA (LB-SEC-IN2)
CARD 9
1.OMEGA-REV / SEC
COL1. REAL PART (DAMPING RATE)
COL2 IMAGINARY PART ( NATURAL FREQ) ;
REAL KX , KY , CX , CY , RX , RY , DX , DY , KXX , KYY , CXX , CYY ,
RXX , RYY , DXX , DYY , L , W , M , G , TP1 , TP2 , TP3 , TEP1 . TEP2 ,
 KXA , KYA , CXA , CYA , RXA , RYA , DXA , DYA ,IP , IT , KTT , KPP ,
 RT , RP , OMEG , OMEGA ;
INTEGER I , A , TMXM , TNRTS , N , S , CYC , REP ;
REAL ARRAY TRRT , TIRT . COEF[0:100] ;
BOOLEAN TSW1 , TSW2 , TSW3 , TSWR ;
LABEL AGAIN , FINIS ;
ARRAY TYME[1:3] ;
FORMAT  HEAD1(X35 , "STABILITY ANALYSIS OF 4-DEGREE FREEDOM SYSTEM",/,
X35 , 45("*") , // ) ;
FORMAT  HEAD2(1(2(59("*")),/),
 X3 , " KX=" , E11.4 , "LB/IN" , X10 , " KY=" . E11.4 ,"LB/IN" , X10 ,
 " RX=" , E11.4 , "LB/IN" , X10 , " RY=" , E11.4 , "LB/IN" , / ,
 X3 , " CX=" , E11.4 , "LB.SEC/IN" , X6 , " CY=" , E11.4 , "LB.SEC/IN" ,
 X6 , " DX=" , E11.4 , "LB.SEC/IN" ,X6 , " DY=" , E11.4 ,"LB.SEC/IN" ,/,
 X3 , "IP=" , E11.4 , "LB-SEC-IN2" , X7 , "IT=" , E11.4 . "LB-SEC-IN2" ;
 X10 , "L=" , E11.4 , "IN." . X13 , "W=" , E11.4 , "LBS." , / , X50 ,
 "SPEED=" , E11.4 , "RPS" , / ,
 1(2(59("*")),/));
```

```
FORMAT  HEAD3 ( X31 , "REAL" , X12, "IMAGINARY"         ) ;
FORMAT OUT1 ( X22 , E18.11 , X4 , E18.11 ) ;
PROCEDURE COEFFICIENT(KX , KY , CX , CY , RX , RY , DX , DY , L ,COEF);
VALUE KX , KY , CX , CY , RX • RY , DX , DY , L ;
REAL  KX , KY , CX , CY , RX , RY , DX , DY , L ;
REAL ARRAY COEF[0]        ;
BEGIN
COEF[0]← KY × KX - RX × RY ;
COEF[1]← KX × CY + KY × CX - RX × DY - RY × DX ;
COEF[2]← KX + KY + CX × CY - DX × DY ;
COEF[3]← CX + CY ;
COEF[4]←  1.0 ;
END OF PROCEDURE COEFFICIENT ;
PROCEDURE  FUNCTIUN ( REALE ,  IMAG , REVAL , IEVAL ) ;
VALUE  REALE , IMAG ;
REAL  REALE , IMAG , REVAL , IEVAL ;
BEGIN
REAL  RTOT , ITOT ;
REAL  ARRAY  RE , IM [0:100] ;
RE[1] ← REALE ;    IM[1] • IMAG ;
FOR  S←2  STEP  1  UNTIL  N  DO
BEGIN
RE[S] ← RE[S-1] × RE[1] - IM[S-1] × IM[1]  ;
IM[S]←  RE[S-1] × IM[1] + IM[S-1] × RE[1]  ;
END;
RTOT← COEF [0] ;
ITOT←0.0 ;
FOR  S←1  STEP  1  UNTIL N  DO
BEGIN
RTOT←  RTOT + RE[S] × COEF [S] ;
ITOT←  ITOT + IM[S] × COEF [S] ;
END ;
REVAL ← RTOT ;
IEVAL ← ITOT  ;
END  ;
PROCEDURE MULLER(P1,P2,P3,MXM,NRTS,EP1,EP2,SW1,SW2,SW3,SWR,RRT,IRT,OT1);
VALUE P1,P2,P3,MXM,NRTS,EP1,EP2,SW1,SW2,SW3,SWR;INTEGER MXM,NRTS;BOOLEAN
 SW1,SW2,SW3,SWR;REAL P1,P2,P3,EP1,EP2;REAL ARRAY RRT,IRT[0];FILE OT1;BE
GIN BOOLEAN BOOL;INTEGER C1,RTC,I,ITC;REAL RX1,RX2,RX3,IX1,IX2,IX3,RROOT
,IROOT,RDNR,IDNR,T1,IT1,FRROOT,FIROOT,RFX1,RFX2,RFX3,IFX1,IFX2,IFX3,RH,I
H,RLAM,ILAM,RDEL,IDEL,T2,IT2,T3,IT3,T4,IT4,RG,IG,RDEN,IDEN,RFUNC,IFUNC;L
ABEL M0,M1,M2,M3,M4,M9,M8,M6,M7,FIN1,FIN2,FIN3,M10,M12,M11,EXIT;SWITCH J
←M2,M3,M4,M7,M11;FORMAT OUT F2(X46,"REAL",X12,"IMAGINARY"/X37,E18.11,X4,
E18.11/X39,"THE FUNCTION EVALUATED AT THIS POINT IS"/X46,"REAL",X12,"IMA
GINARY"/X37,E18.11,X4,E18.11/X35,"THE MOD""IFIED FUNCTION EVALUATED AT T
HIS POINT IS"/X46,"REAL",X12,"IMAGINARY"/X37,E18.11,X4,E18.11),F4(///X29
,I3," ITERATIONS HAVE BEEN MADE.  THE VALUE OF ""THE ITERANT IS NOW"),F6
(//X37,"SUCCESSIVE ITERANTS MEET CONVERGENCE CRITERION"/X39,"AFTER",I3,"
 ITERATIONS.  THE ROOT FOUND IS"),F8(//X33,"THE FUNCTION VALUES OF THE L
AST ITERANT ARE"" SUFFICIENTLY"/X33,"NEAR ZERO. ",I4," ITERATIONS WERE M
A""DE.  THE ROOT FOUND IS"),F10(//X35,I3," ITERATIONS COMPLETED AND SUCC
ESSFUL CONVE""RGENCE"/X41,"HAS NOT OCCURRED.  THE LAST ITERANT IS"),F12(
//X40,"THE PREVIOUS ROOT FOUND WAS COMPLEX.  THE"/X40,"CONJUGATE OF THIS
 VALUE IS ALSO A ROOT.");PROCEDURE COMPLEX(A,IA,B,IB,K,C,IC);VALUE A,IA,
B,IB,K;INTEGER K;REAL A,IA,B,IB,C,IC;BEGIN REAL TEMP;LABEL MPY,DVD,SQT,E
XIT;SWITCH JUNCTION←MPY,DVD,SQT;GO TO JUNCTION[K];MPY:C←A×B-IA×IB;IC←A×I
```

```
B+IA×B;GO TO EXIT;DVD:IF B=0AND IB=0THEN BEGIN C←1;IC←0;GO TO EXIT END;T
EMP←B×B+IB×IB;C←(A×B+IA×IB)/TEMP;IC←(IA×B−A×IB)/TEMP;GO TO EXIT;SQT:IF(I
A=0)AND(A<0)THEN BEGIN C←0;IC←SQRT(−A)END ELSE IF IA=0THEN BEGIN C←SQRT(
A);IC←0END ELSE BEGIN TEMP←SQRT(A×A+IA×IA);C←SQRT((TEMP+A)/2);IC←IF(TEMP
−A)<0THEN 0ELSE SQRT((TEMP−A)/2)END;IF((B+C)×(B+C)+(IB+IC)×(IB+IC))<((B−
C)×(B−C)+(IB−IC)×(IB−IC))THEN BEGIN C←B−C;IC←IB−IC END ELSE BEGIN C←B+C;
IC←IB+IC END;EXIT:END;FOR I←1STEP 1UNTIL NRTS DO RRT[I]←IRT[I]←0;RTC←0;M
0:IX1←IX2←IX3←C1←IROOT←ITC←0;RROOT←P1;BOOL←FALSE;M1:C1←C1+1;RDNR←1;IDNR←
0;FOR I←1STEP 1UNTIL RTC DO BEGIN COMPLEX(RDNR,IDNR,RROOT−RRT[I],IROOT−I
RT[I],1,T1,IT1);RDNR←T1;IDNR←IT1 END;FUNCTION(RROOT,IROOT,T1,IT1);COMPLE
X(T1,IT1,RDNR,IDNR,2,FRROOT,FIROOT);GO TO J[C1];M2:RFX1←FRROOT;IFX1←FIRO
OT;RROOT←P2;GO TO M1;M3:RFX2←FRROOT;IFX2←FIROOT;RROOT←P3;GO TO M1;M4:RFX
3←FRROOT;IFX3←FIROOT;RX1←P1;RX2←P2;RX3←P3;RH←RX3−RX2;IH←IX3−IX2;COMPLEX(
RH,IH,RX2−RX1,IX2−IX1,2,RLAM,ILAM);RDEL←RLAM+1;IDEL←ILAM;M9:IF(RFX1=RFX2
)AND(RFX2=RFX3)AND(IFX1=IFX2)AND(IFX2=IFX3)THEN BEGIN RLAM←1;ILAM←0;GO T
O M8 END;COMPLEX(RFX1,IFX1,RLAM,ILAM,1,T1,IT1);COMPLEX(RFX2,IFX2,RDEL,ID
EL,1,T2,IT2);T1←T1−T2+RFX3;IT1←IT1−IT2+IFX3;COMPLEX(RDEL,IDEL,RLAM,ILAM,
1,T2,IT2);COMPLEX(T1,IT1,T2,IT2,1,T3,IT3);COMPLEX(RFX3,IFX3,T3,IT3,1,T1,
IT1);T1←−4×T1;IT1←−4×IT1;COMPLEX(RFX3,IFX3,RLAM+RDEL,ILAM+IDEL,1,T2,IT2)
;COMPLEX(RDEL×RDEL−IDEL×IDEL,2×RDEL×IDEL,RFX2,IFX2,1,T3,IT3);COMPLEX(RLA
M×RLAM−ILAM×ILAM,2×RLAM×ILAM,RFX1,IFX1,1,T4,IT4);RG←T4−T3+T2;IG←IT4−IT3+
IT2;IF SWR AND((RG×RG+T1)<0)THEN BEGIN RDEN←RG;IDEN←IG←0END ELSE COMPLEX
(RG×RG−IG×IG+T1,2×RG×IG+IT1,RG,IG,3,RDEN,IDEN);COMPLEX(−2×RFX3,−2×IFX3,R
DEL,IDEL,1,T1,IT1);COMPLEX(T1,IT1,RDEN,IDEN,2,RLAM,ILAM);M8:ITC←ITC+1;RX
1←RX2;RX2←RX3;RFX1←RFX2;RFX2←RFX3;IX1←IX2;IX2←IX3;IFX1←IFX2;IFX2←IFX3;CO
MPLEX(RLAM,ILAM,RH,IH,1,T1,IT1);RH←T1;IH←IT1;M6:RDEL←RLAM+1;IDEL←ILAM;RX
3←RX2+RH;IX3←IX2+IH;C1←3;RROOT←RX3;IROOT←IX3;GO TO M1;M7:RFX3←FRROOT;IFX
3←FIROOT;FUNCTION(RX3,IX3,RFUNC,IFUNC);COMPLEX(RFX3,IFX3,RFX2,IFX2,2,T1,
IT1);IF(T1×T1+IT1×IT1)>100THEN BEGIN RLAM←RLAM/2;RH←RH/2;ILAM←ILAM/2;IH←
IH/2;GO TO M6 END;IF SW1 THEN BEGIN WRITE(OT1,F4,ITC);WRITE(OT1,F2,RX3,I
X3,RFUNC,IFUNC,RFX3,IFX3)END;T1←RX3−RX2;IT1←IX3−IX2;COMPLEX(T1,IT1,RX2,I
X2,2,T2,IT2);IF SQRT(T2×T2+IT2×IT2)≤EP1 THEN GO TO FIN1;IF(SQRT(RFUNC×RF
UNC+IFUNC×IFUNC)≤EP2)AND(SQRT(RFX3×RFX3+IFX3×IFX3)≤EP2)THEN GO TO FIN2;I
F ITC≥MXM THEN GO TO FIN3 ELSE GO TO M9;FIN1:IF(NOT SW2)THEN GO TO M12 E
LSE WRITE(OT1,F6,ITC);GO TO M10;FIN2:IF(NOT SW2)THEN GO TO M12 ELSE WRIT
E(OT1,F8,ITC);GO TO M10;FIN3:BOOL←TRUE;IF(NOT SW2)THEN GO TO M12 ELSE WR
ITE(OT1,F10,ITC);M10:WRITE(OT1,F2,RX3,IX3,RFUNC,IFUNC,RFX3,IFX3);M12:RTC
←RTC+1;RRT[RTC]←RX3;IRT[RTC]←IX3;IF RTC≥NRTS THEN GO TO EXIT;IF(ABS(IX3)
>EP1)AND(SW3)AND(NOT BOOL)THEN BEGIN IX3←−IX3;FUNCTION(RX3,IX3,RFUNC,IFU
NC);RROOT←RX3;IROOT←IX3;C1←4;GO TO M1;M11:IF SW2 THEN BEGIN WRITE(OT1,F1
2);WRITE(OT1,F2,RX3,IX3,RFUNC,IFUNC,FRROOT,FIROOT)END;RTC←RTC+1;RRT[RTC]
←RX3;IRT[RTC]←IX3 END ELSE GO TO M0;IF RTC<NRTS THEN GO TO M0;EXIT:END;
SWITCH  FORMAT  FMTYME ← ( "DATE" , A21 ),
("TOTAL TIME", F15.2 , "   SECONDS") ,
("PROCESSOR TIME" , F11.2 , "   SECONDS") ,
("I/O TIME" , F17.2 , "   SECONDS") ;
FOR A←1 STEP 1 UNTIL 3 DO TYME[A] ← TIME(A) ;
WRITE (LP , HEAD1 ) ;
READ ( CR , / , N ) ;
AGAIN: READ (CR , /,KX , KY ) [FINIS] ;
READ ( CR , / , CX , CY ) ;
READ ( CR , / , RX , RY ) ;
READ ( CR , / , DX , DY ) ;
READ ( CR , / , L ) ;
READ (CR , / , W ) ;
 READ ( CR , / , IP , IT ) ;
```

```
READ(CR , / , OMEG ) ;
WRITE ( LP , HEAD2 , KX , KY , RX , RY , CX , CY , DX , DY , IP ,
 IT ,  L , W , OMEG ) ;
G← 32.2 × 12 ;
M← W/G ;
KXA←(2×KX)/M ; KYA←(2×KY)/M ; CXA←(2×CX)/M ; CYA←(2×CY)/M ;
RXA←(2×RX)/M ; RYA←(2×RY)/M ; DXA←(2×DX)/M ; DYA←(2×DY)/M ;
COEFFICIENT ( KXA , KYA , CXA , CYA , RXA , RYA , DXA , DYA ,L , COEF) ;
 WRITE ( LP , < X10 , "THE FOLLOWING GIVES THE CYLINDRICAL FREQ. :"> );
WRITE(LP[DBL]) ;
WRITE (LP , < "THE COEF OF THE POLYNOMIAL IN ASCENDING POWER ARE:">) ;
WRITE (LP , < 5(X2 , E11.4)> , FOR I←0 STEP 1 UNTIL N DO COEF[I] ) ;
WRITE(LP[DBL]) ;
WRITE (LP , HEAD3 ) ;
TNRTS← N ;
FOR I←0 STEP 1 UNTIL 100 DO TRRT[I] ← TIRT[I] ← 0 ;
MULLER (-1 , 0 , 1.0 ,30 , N , 1.0@-12 , 1.0@-12 , FALSE , FALSE , TRUE,
 FALSE , TRRT , TIRT , LP ) ;
WRITE ( LP ,OUT1 , FOR I← 1  STEP  1 UNTIL  TNRTS DO [TRRT[I] ,
 TIRT[I]]) ;
WRITE(LP[DBL]) ;
OMEGA←OMEG×6.28 ;
KTT←IT/M ; KPP←IP/M ; RT←(4×KTT)/(L×L) ; RP←(4×KPP)/(L×L) ;
IF KTT=0 AND KPP=0 THEN
BEGIN
N←2 ;
TNRTS←N  ;
COEF[0]← KXA×KYA-RXA×RYA ;
COEF[1]← KXA × CYA + KYA × CXA + RYA × ( RP × OMEGA - DXA) - RXA ×
 ( RP × OMEGA + DYA) ;
COEF[2]← RT × ( KXA+ KYA) + CXA× CYA+ ( RP × OMEGA + DYA ) ×
 ( RP × OMEGA - DXA ) ;
WRITE (LP , < X10 , "THE FOLLOWING GIVES THE CONICAL FREQ. :" > ) ;
WRITE (LP[DBL]) ;
WRITE (LP , < "THE COEF OF THE POLYNOMIAL IN ASCENDING POWER ARE:">) ;
WRITE(LP[DBL]) ;
WRITE (LP , < 5(X2 , E11.4)> , FOR I←0 STEP 1 UNTIL N DO COEF[I] ) ;
   WRITE(LP[DBL]) ;
FOR I←0 STEP 1 UNTIL 100 DO TRRT[I] ← TIRT[I] ← 0 ;
MULLER (-1 , 0 , 1.0 ,30 , N , 1.0@-12 , 1.0@-12 , FALSE , FALSE , TRUE,
 FALSE , TRRT , TIRT , LP ) ;
WRITE(LP,HEAD3) ;
   WRITE(LP[DBL]) ;
WRITE ( LP ,OUT1 , FOR I← 1  STEP  1 UNTIL  TNRTS DO [TRRT[I] ,
 TIRT[I]]) ;
   WRITE(LP[DBL]) ;
END
ELSE
BEGIN
COEF[0]← KXA×KYA-RXA×RYA ;
COEF[1]← KXA × CYA + KYA × CXA + RYA × ( RP × OMEGA - DXA) - RXA ×
 ( RP × OMEGA + DYA) ;
COEF[2]← RT × ( KXA+ KYA) + CXA× CYA+ ( RP × OMEGA + DYA ) ×
 ( RP × OMEGA - DXA ) ;
COEF[3]← (CXA+CYA) × RT ;
COEF [4]← RT × RT ;
```

```
        WRITE (LP , < X10 , "THE FOLLOWING GIVES THE CONICAL FREQ. :" > ) ;
        WRITE (LP[DBL]) ;
        WRITE (LP , < "THE COEF OF THE POLYNOMIAL IN ASCENDING POWER ARE:">) ;
           WRITE(LP[DBL]) ;
        WRITE (LP , < 5(X2 , E11.4)> , FOR I←0 STEP 1 UNTIL N DO COEF[I] ) ;
           WRITE(LP[DBL]) ;
        FOR I←0 STEP 1 UNTIL 100 DO TRRT[I] ← TIRT[I] ← 0 ;
        MULLER (-1 , 0 , 1.0 ,30 , N , 1.0@-12 , 1.0@-12 , FALSE , FALSE , TRUE,
         FALSE , TRRT , TIRT , LP ) ;
        WRITE(LP,HEAD3) ;
           WRITE(LP[DBL]) ;
        WRITE ( LP ,OUT1 , FOR I← 1  STEP  1 UNTIL  TNRTS DO [TRRT[I] ,
         TIRT[I]]) ;
           WRITE(LP[DBL]) ;
        END;
        FOR A←1  STEP 1 UNTIL  3  DO
        WRITE ( LP ,FMTYME[A] , (TIME(A) - TYME[A]) / 60) ;
        WRITE ( LP ,FMTYME[0] , TIME(0)) ;
        WRITE (LP[PAGE]) ;
        GO TO AGAIN ;
        FINIS: END .
      SQRT IS SEGMENT NUMBER 0021,PRT ADDRESS IS 0133
      OUTPUT(W) IS SEGMENT NUMBER 0022,PRT ADDRESS IS 0136
      BLOCK CONTROL IS SEGMENT NUMBER 0023,PRT ADDRESS IS 0005
      INPUT(W) IS SEGMENT NUMBFR 0024,PRT ADDRESS IS 0147
      GO TO SOLVER  IS SEGMENT NUMBER 0025,PRT ADDRESS IS 0152
      ALGOL WRITE   IS SEGMENT NUMBER 0026,PRT ADDRESS IS 0014
      ALGOL READ    IS SEGMENT NUMBER 0027,PRT ADDRESS IS 0015
      ALGOL SELECT  IS SEGMENT NUMBER 0028,PRT ADDRESS IS 0016
COMPILATION TIME =    69 SECONDS.
NUMBER OF ERRORS DETECTED = 000. LAST ERROR ON CARD #
NUMBER OF SEQUENCE ERRORS COUNTED =    0.
NUMBER OF SLOW WARNINGS =    0.
PRT SIZE= 132;     TOTAL SEGMENT SIZE= 1288 WORDS.
DISK STORAGE REQ.=    65 SEGS.;     NO. SEGS.=  29.
ESTIMATED CORE STORAGE REQUIREMENT =  2436 WORDS.
```

TABLE E-I. - STABILITY ANALYSIS OF FOUR DEGREE FREEDOM SYSTEM

KX= 3.6000@+04LB/IN	KY= 3.6000@+04LB/IN	RX= 6.2500@+03LB/IN	RY=-6.2500@+03LB/IN
CX= 3.2000@+00LB.SEC/IN	CY= 3.2000@+00LB.SEC/IN	DX= 0.0000@+00LB.SEC/IN	DY= 0.0000@+00LB.SEC/IN
IP= 6.0000@-02LB-SEC-IN2	IT= 1.2600@+00LB-SEC-IN2	L= 7.0000@+00IN.	W= 1.8000@+01LBS.

SPEED= 6.2200@+02RPS

THE FOLLOWING GIVES THE CYLINDRICAL FREQ. :

THE COEF OF THE POLYNOMIAL IN ASCENDING POWER ARE:

2.4609@+12 4.2469@+08 3.1101@+06 2.7477@+02 1.0000@+00

REAL	IMAGINARY
-1.76372621191@+02	1.24598397060@+03
-1.76372621191@+02	-1.24598397060@+03
3.89859545356@+01	1.24598397060@+03
3.89859545356@+01	-1.24598397060@+03

THE FOLLOWING GIVES THE CONICAL FREQ. :

THE COEF OF THE POLYNOMIAL IN ASCENDING POWER ARE:

2.4609@+12 2.0428@+08 7.0129@+06 6.0670@+02 4.8753@+00

REAL	IMAGINARY
4.42580032346@+01	7.51603975630@+02
4.42580032346@+01	-7.51603975630@+02
-1.06480225459@+02	9.37611594680@+02
-1.06480225459@+02	-9.37611594680@+02

TOTAL TIME	8.32	SECONDS
PROCESSOR TIME	3.62	SECONDS
I/O TIME	9.12	SECONDS
DATE	067361	

```
KX= 4.0000@+04LB/IN          KY= 3.0000@+04LB/IN          RX= 6.2500@+03LB/IN          RY=-6.2500@+03LB/IN
CX= 6.4000@+00LB.SEC/IN      CY= 6.4000@+00LB.SEC/IN      DX= 0.0000@+00LB.SEC/IN      DY= 0.0000@+00LB.SEC/IN
IP= 6.0000@-02LB-SEC-IN2     IT= 1.2600@+00LB-SEC-IN2     L= 7.0000@+00IN.             W= 1.8000@+01LBS.
                                         SPEED= 6.2200@+02RPS

          THE FOLLOWING GIVES THE CYLINDRICAL FREQ. :

THE COEF OF THE POLYNOMIAL IN ASCENDING POWER ARE:
   2.2839@+12   8.2579@+08   3.0808@+06   5.4955@+02   1.0000@+00

                        REAL                IMAGINARY
               -2.03375900748@+02     1.21989595846@+03
               -2.03375900748@+02    -1.21989595846@+03
               -7.13974325710@+01     1.21989595848@+03
               -7.13974325710@+01    -1.21989595848@+03

          THE FOLLOWING GIVES THE CONICAL FREQ. :

THE COEF OF THE POLYNOMIAL IN ASCENDING POWER ARE:

   2.2839@+12   6.0537@+08   6.8800@+06   1.2134@+03   4.8753@+00

                        REAL                IMAGINARY

                9.46436045130@+00     7.25562350470@+02
                9.46436045130@+00    -7.25562350470@+02
               -1.33908804900@+02     9.33704419740@+02
               -1.33908804900@+02    -9.33704419740@+02

TOTAL TIME          12.45   SECONDS
PROCESSOR TIME       6.77   SECONDS
I/O TIME            12.37   SECONDS
DATE               067361
```

NASA-Langley, 1969 — 15 E-4920