NASA CR - 72740

R - 8385

FLEXIBLE ROTORDYNAMIC ANALYSIS

FINAL REPORT

by

F. A. Shen and E. Mogil

POWER SYSTEMS DIVISION - ROCKETDYNE
A Division of North American Rockwell Corporation

prepared for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

NASA Lewis Research Center
Contract NAS 3-13219
David P. Fleming, Project Manager

NOTICE

Requests for copies of this report should be referred to

National Aeronautics and Space Administration
Scientific and Technical Information Facility
P.O. Box 33
College Park, Md. 20740

FINAL REPORT

FLEXIBLE ROTOR DYNAMIC ANALYSIS

by

F. A. Shen and E. Mogil

POWER SYSTEMS DIVISION - ROCKETDYNE
A Division of North American Rockwell Corporation
6633 Canoga Avenue
Canoga Park, California

prepared for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

June 1, 1970

CONTRACT NAS 3-13219

Technical Management
NASA Lewis Research Center
Cleveland, Ohio
David P. Fleming

FORWARD

This report was prepared by Rocketdyne - Power Systems Division, A Division of North American Rockwell Corporation, under National Aeronautics and Space Administration Contract NAS 3-13219.

ABSTRACT

A digital computer program was developed to analyze a flexible rotor-bearing system. The program can be used to predict the dynamic performance under various operating conditions. An experimental test program was conducted utilizing a Mark-25 pump rotor, in a vacuum, to generate spin test data to be used for correlation with results of computer simulations of similar rotor unbalance conditions. The mathematical formulations, the computer program verification, the simulation model and test results are presented and discussed.

TABLE OF CONTENTS

Page No.

SUMMARY 1

INTRODUCTION 3

MATHEMATICAL FORMULATION 5

EQUATIONS 6
NOMENCLATURE 22
VERIFICATION OF THE COMPUTER PROGRAM 27
CAPABILITY AND LIMITATION OF THE CURRENT COMPUTER PROGRAM 33

COMPUTER PROGRAM USER'S INSTRUCTIONS 35

DESCRIPTION OF THE COMPUTER PROGRAM 35

INPUT PROCEDURE 41
Input 42
Special Notes on Input 52
Namelist Input Procedure and Sample Input Data Sheet 54
Non-Linear Bearing Stiffness Characteristics Representation 60
Output Format 63

EXPERIMENTAL PROGRAM 66

TEST SET-UP 67
Hardware 67
Instrumentation 67

TEST PROGRAM 69

Balancing Tests 69
Investigative Tests 69

PROCEDURE FOR TEST SET-UP 71

ROTORDYNAMICS SPIN TEST PROCEDURE 74

RESULTS 76

SIMULATION OF EXPERIMENTAL TESTS 113

SIMULATION MODEL 113

RESULTS OF THE COMPUTER SIMULATION 113

CONCLUSIONS AND RECOMMENDATIONS 126

APPENDIX A - TYPICAL COMPUTER PROGRAM OUTPUT 130

APPENDIX B - COMPUTER PROGRAM LISTING 144

TABLE OF CONTENTS
(Continued)

Page No.

Figure 1 - Relation Between the Secondary Frame of Reference and an Inertial Frame of Reference 18

Figure 2 - Schematic of the General Physical Model 19

Figure 3 - Rotating and Secondary Coordinate Systems Used In the Mathematical Model 21

Figure 4 - Data from a 5-Mass 95.5 rpm Flexible Rotor in a Steady-State Operation (Rotor Displacement) . . . 31

Figure 5 - Data from a 5-Mass 95.5 rpm Flexible Rotor in a Steady-State Operation (Whirl-to-Spin Frequency Ratio) 32

Figure 6 - General Flow Diagram Relating Various Subroutines and the MAIN Program 36

Figure 7 - Sample Input Data 50

Figure 8 - Bearing Stiffness Curve 61

Figure 9 - Mark-25 Rotor High Speed Test Set-Up 68

Figure 10 - Mark 25 Rotordynamic Data Inducer Forward End Displacement vs Angles 70

Figure 11 - Mark-25 Rotordynamic Data Inducer and Coupling Ends 72

Figure 12 - Bently and Balance Planes, Mark-25 Pump Rotor 79

Figure 13 - Test No. 1136 - Bently's 2, 8 and 16 Deflections vs Speed 81

Figure 14 - Test No. 1137 - Bently's 2, 8 and 16 Deflections vs Speed 82

Figure 15 - Test No. 1138 - Bently's 2, 8 and 16 Deflection vs Speed 83

Figure 16 - Test No. 1139 - Bently's 2, 8 and 16 Deflection vs Speed 84

Figure 17 - Test No. 1142 - Bently's 2, 8 and 16 Deflection vs Speed 85

Figure 18 - Test No. 1143 - Bently's 2, 8 and 16 Deflection vs Speed 86

TABLE OF CONTENTS
(Continued)

Page No.

Figure 19 - Test No. 1144 - Bently's 2, 8 and 16 Deflections vs Speed 87

Figure 20 - Test No. 1145 - Bently's 2, 8 and 16 Deflections vs Speed 88

Figure 21 - Test No. 1136 at 26,000 rpm, Deflection vs Station . 89

Figure 22 - Test No. 1136 at 28,000 rpm, Deflection vs Station . 90

Figure 23 - Test No. 1136 at 30,000 rpm, Deflection vs Station . 91

Figure 24 - Test No. 1137 at 26,000 rpm, Deflection vs Station . 92

Figure 25 - Test No. 1137 at 28,000 rpm, Deflection vs Station . 93

Figure 26 - Test No. 1137 at 30,000 rpm, Deflection vs Station . 94

Figure 27 - Test No. 1138 at 28,000 rpm, Deflection vs Station . 95

Figure 28 - Test No. 1138 at 30,000 rpm, Deflection vs Station . 96

Figure 29 - Test No. 1138 at 32,000 rpm, Deflection vs Station . 97

Figure 30 - Test No. 1139 at 28,000 rpm, Deflection vs Station . 98

Figure 31 - Test No. 1139 at 30,000 rpm, Deflection vs Station . 99

Figure 32 - Test No. 1139 at 32,000 rpm, Deflection vs Station . 100

Figure 33 - Test No. 1142 at 28,000 rpm, Deflection vs Station . 101

Figure 34 - Test No. 1142 at 30,000 rpm, Deflection vs Station . 103

Figure 35 - Test No. 1142 at 32,000 rpm, Deflection vs Station . 104

Figure 36 - Test No. 1143 at 26,000 rpm, Deflection vs Station . 104

Figure 37 - Test No. 1143 at 28,000 rpm, Deflection vs Station . 105

Figure 38 - Test No. 1143 at 30,000 rpm, Deflection vs Station . 106

Figure 39 - Test No. 1144 at 26,000 rpm, Deflection vs Station . 107

Figure 40 - Test No. 1144 at 28,000 rpm, Deflection vs Station . 108

Figure 41 - Test No. 1144 at 30,000 rpm, Deflection vs Station . 109

Figure 42 - Test No. 1145 at 26,000 rpm, Deflection vs Station . 110

Figure 43 - Test No. 1145 at 28,000 rpm, Deflection vs Station . 111

Figure 44 - Test No. 1145 at 30,000 rpm, Deflection vs Station . 112

TABLE OF CONTENTS
(Continued)

Page No.

TABLE 1 - 5-STATION ROTOR-BEARING CHECKOUT COMPARATIVE RESULTS 28

TABLE 2 - COMPARISON OF MARK 25, 17 MASS MODEL NEW PROGRAM VS OLD PROGRAM 29

TABLE 3 - UNBALANCE AND UPPER SPEEDS FOR EXPERIMENTAL TESTS ... 73

TABLE 4 - RELATIVE MATHEMATICAL AND EXPERIMENTAL MODEL DESCRIPTORS 77

TABLE 5 - MARK-25 PUMP ROTOR MASS AND INERTIA PROPERTIES 116

TABLE 6 - MARK-25 PUMP SIMULATION DATA 117

TABLE 7 - TEST 1136, SIMULATION AT 26,000 rpm 118

TABLE 8 - TEST 1137, SIMULATION AT 26,000 rpm 119

TABLE 9 - TEST 1138, SIMULATION AT 32,000 rpm 120

TABLE 10 - TEST 1139, SIMULATION AT 32,000 rpm 121

TABLE 11 - TEST 1142, SIMULATION AT 32,000 rpm 122

TABLE 12 - TEST 1143, SIMULATION AT 32,000 rpm 123

TABLE 13 - TEST 1144, SIMULATION AT 32,000 rpm 124

TABLE 14 - TEST 1145, SIMULATION AT 26,000 rpm 125

SUMMARY

A 10-month study contract, NAS 3-13219, was initiated to update the digital computer program that evolved from a previous study, NAS 3-7996. The work was conducted in three parts. They were 1) update the flexible rotor digital computer program, 2) perform experimental spin tests to generate data, and 3) simulate the experimental tests using the computer program and determine the degree of correlation.

A flexible-rotor dynamic analysis computer program based on Newton's laws of dynamics has been written and computation results verified. The rotor spin and whirl motion were treated as independent but dynamically interacting parameters. Accordingly various whirl-to-spin frequency ratios can be developed as the dynamics relationship dictates. The spin speed of the rotor is controlled by torque applied and combined stiffness, dissipation and inertial loading functions in spin and transverse motion. Both in-phase and out-of-phase stiffness and damping functions are included at all rotor stations. Computation results are of both printout and CRT graph types. After debugging, the computer program was verified in three separate verification steps and the computation results were found to be in good agreement with those from manual and other machine computations. User's instructions are also provided.

The experimental test program, utilizing a Mark-25 pump rotor, was performed to obtain data for correlation with the results of the computer program. The test program consisted of a series of high-speed balancing tests and the eight investigative tests. The investigative tests were simulated with the computer program, and the correlation between theoretical and test results was not very close. Some degree of correlation existed for unbalance conditions at the inducer end of the rotor. Apparently the rotor modeling was not sufficiently descriptive.

The computer program as developed under this study contract can be used as a tool in predicting the dynamic performance of a flexible rotor-bearing system under various operating conditions.

Recommendations are made for further studies to improve the degree of correlation and to increase the computer program capability.

INTRODUCTION

As a continuing effort in advancing the state-of-the-art in rotordynamic analysis, a 10 month program was initiated to develop a computer program to permit more accurate analysis of rotating assemblies, and to apply this program to predictions of the dynamic behavior of the Rocketdyne Mark-25 pump rotor.

The program was accomplished in three tasks: (1) revising and updating the computer program, (2) performing spin tests, and (3) simulation of the experimental data utilizing the computer program.

The mathematical formulation used for the effort reported in this contract, although based on the same general principle as the previous study (Contract NAS3-7996), has been completely rewritten to include the following parameters required by the contract

1. Rotor flexibility due to both shear and bending
2. Damping functions
3. Non-isotropic bearing mounts
4. Nonlinear bearing spring rates
5. Multiple bearing supports

and the following additional items:

6. Linear, speed-sensitive, support bearing stiffness;
7. Enlarged maximum number of rotor-stations to 25, from 12,
8. A displacement vector rotating coordinate system (whirl), as opposed to the spin-speed rotating coordinate used in the 1967 contract; and
9. Up-dating the CRT graphs to include multiple-bearing output data plots.

In addition, spin tests utilizing the Mark-25 pump rotor were performed in the Rocketdyne rotor dynamics facility, for correlation with the results of the flexible rotor dynamics computer program.

Simulation of the spin test and high speed balancing data was performed with the computer program, utilizing as program inputs the experimentally introduced unbalances.

A checked-out operating computer card deck for the flexible rotor dynamics computer program was delivered to the NASA Project Manager.

This final report presents the results of the research program conducted 4 September 1969 through 1 May 1970. During the course of development of the computer program, certain shortcomings in the program were noted. Conclusions and recommendations are given towards further improving the program computational-to-real time ratio and other rotordynamic parameters.

MATHEMATICAL FORMULATION (THEORY)

In writing the governing equations of a dynamic system, two general approaches may be taken, the energy approach and the momentum approach. The former is generally accomplished by means of Lagrange equations and the latter is carried out according to Newton's Laws of Dynamics. When the traction vs displacement relationship of a dynamic system can be clearly seen, it is faster and more direct to use the momentum approach as it is adopted in this analysis.

The basic assumptions made in the formulation of the general mathematical model are:

1. Representation of a continuous, elastic and distributed mass rotor by a set of discrete masses connected by weightless, elastic shaft sections. The polar and diametral mass moments of inertia and the mass of a rotor section are suitably lumped at the adjacent mass stations. The analysis of the transverse rotordynamic problem is based on a lumped parameter model. With an adequate number of discrete mass stations, the dynamic performance of a rotor can be accurately simulated.

2. Small lateral displacement of a rotor in comparison with the sectional length of a rotor*. This assumption is quite realistic for all rotor dynamics analysis. Under this assumption, the linear moduli of elasticity of the rotor may be used in establishing load and deflection relationship.

---

* Sectional length of a rotor refers to the length of a rotor section between adjacent discrete mass stations.

3. An axisymmetric rotor geometry and rotor mechanical properties are assumed. While a nonaxisymmetric rotor section will not introduce any solution difficulties, it would result in a very time-consuming computation process. The inclusion of the non-axisymmetric rotor design consideration in a computer analysis is generally not justified as most rotor geometries are or can be considered axisymmetric.

4. The foundation of the rotating machinery is anchored to the secondary frame of reference XYZ in Fig. 1. The secondary frame of reference is permitted to have only translatory motion at a constant velocity with respect to the inertial axes $X_oY_oZ_o$. The surface of the earth may be considered as a secondary inertial frame of reference with a constant velocity relative to the inertial frame of reference $X_oY_oZ_o$. Small angular motion of the secondary frame of reference from the earth rotation will have negligible effects on the dynamics of rotors except those of a gyroscope type of instrument.

Based on the above assumptions, a set of rotordynamics governing equations may be written. Figure 2 depicts a dynamic configuration of a rotor and its casing, and defines certain notations used in the equations. Figure 3 shows that the coordinates X_i and Y_i refers to the geometrical center of a rotor section.

EQUATIONS

The following 22 equations constitute the mathematical formulation for the physical model shown in Figure 2:

$$\ddot{\phi}\sum_{i}^{n}\left[I_{\rho_i}+m_i e_i^2\right]+\sum_{i}^{n}\left\{m_i e_i\left[\left(\ddot{Y}_i+g_Y\right)\cos\left(\phi+\alpha_i\right)-\left(\ddot{X}_i+g_X\right)\mathrm{SIN}\left(\phi+\alpha_i\right)\right]\right\}+\sum_{i}^{n}\left[C_{Z1i}\dot{\phi}^{C_{Zi}}+C_{Z2i}\dot{\phi}\right]-\left[M_{Z1}\dot{\phi}^{M_Z}+M_{Z2}\dot{\phi}+M_{Z3}\right]=0 \qquad (1)$$

Eq. (1) defines the torsional moment equilibrium relationship about Z-axis. The first group of terms,

$$\ddot{\phi}\sum_{i}^{n}\left[I_{\rho_i}+m_i e_i^2\right]$$

represents the inertia torque from the rotor polar mass moments of inertia about the rotor elastic centers. The second group,

$$\sum_{i}^{n}\left\{m_i e_i\left[\left(\ddot{Y}_i+g_Y\right)\cos\left(\phi+\alpha_i\right)-\left(\ddot{X}_i+g_X\right)\mathrm{SIN}\left(\phi+\alpha_i\right)\right]\right\}$$

denotes the torque derived from the rotor mass inertia forces. The third group

$$\sum_{i}^{n}\left[C_{Z1i}\dot{\phi}^{C_{Zi}}+C_{Z2i}\dot{\phi}\right]$$

is a summation of all rotor spin speed sensitive and constant dissipative torques, and

$$-\left[M_{Z1}\dot{\phi}^{M_Z}+M_{Z2}\dot{\phi}+M_{Z3}\right]=0$$

The last group represents the spin speed sensitive and constant rotor drive torques.

Eqs. (2) and (3) summarize all forces specified at rotor station i along X and Y axis, respectively.

$$-F_{Xi} = m_i \Bigg\{ \ddot{X}_i + g_X - e_i \Big[\ddot{\phi} \, \mathrm{SIN}(\phi+\alpha_i) + (\dot{\phi})^2 \mathrm{COS}(\phi+\alpha_i) \Big] \Bigg\}$$

$$+ \Big[K_i X_i + C_i \dot{X}_i + K_{pi} Y_i + C_{pi} \dot{Y}_i \Big] + \Big[K_{HDi} Y_i (\dot{\phi} - K_{Fi} \omega_i)$$

$$+ C_{HDi} \dot{Y}_i (\dot{\phi} - C_{Fi} \omega_i) \Big] + A_i \Bigg\{ \Bigg[\Bigg((\dot{\phi} - \dot{\phi}_0) \Big[N_{Bik}$$

$$+ B_{Bik} \Big(\sqrt{X_{Bi}^2 + Y_{Bi}^2} - \rho_{Bik} \Big) \Big) \Big] + K_{Bik} \Bigg] \Bigg[\Bigg[\left(X_{Bi}^2 + Y_{Bi}^2 \right)^{\frac{H_{BiK}-1}{2}}$$

$$+ D_{Bik} + \frac{E_{Bik}}{\sqrt{X_{Bi}^2 + Y_{Bi}^2}} \Bigg] X_{Bi} + C_{BXi} \dot{X}_i \Bigg\} \tag{2}$$

$$-F_{Yi} = m_i \Bigg\{ \ddot{Y}_i + g_Y + e_i \Big[\ddot{\phi} \, \mathrm{COS}(\phi+\alpha_i) - (\dot{\phi})^2 \mathrm{SIN}(\phi+\alpha_i) \Big] \Bigg\}$$

$$+ \Big[K_i Y_i + C_i \dot{Y}_i - K_{pi} X_i - C_{pi} \dot{X}_i - K_{HDi} X_i (\dot{\phi} - K_{Fi} \omega_i)$$

$$- C_{HDi} \dot{X}_i (\dot{\phi} - C_{Fi} \omega) \Big] + A_i \Bigg\{ \Bigg[\Bigg((\dot{\phi} - \dot{\phi}_0) \Big[N_{Bik}$$

$$+ B_{Bik} \Big(\sqrt{X_{Bi}^2 + Y_{Bi}^2} - \rho_{Bik} \Big) \Big) \Big] + K_{Bik} \Bigg] \Bigg[\Bigg[\left(X_{Bi}^2 + Y_{Bi}^2 \right)^{\frac{H_{Bik}-1}{2}}$$

$$+ D_{Bik} + \frac{E_{Bik}}{\sqrt{X_{Bi}^2 + Y_{Bi}^2}} \Bigg] Y_{Bi} + C_{BYi} \dot{Y}_i \Bigg\} \tag{3}$$

The first group in Eq. (2) and similarly with Eq. (3).

$$m_i \left\{ \ddot{X}_i + g_x - e_i \left[\ddot{\phi} \sin(\phi + \alpha_i) + (\dot{\phi})^2 \cos(\phi + \alpha_i) \right] \right\}$$

defines the inertia and gravity and/or G-loading forces along X-axis. The second group,

$$\left[K_i X_i + C_i \dot{X}_i + K_{pi} Y_i + C_{pi} \dot{Y}_i \right]$$

denotes the rotor surface forces at a rotor station.

$K_i X_i$, $C_i \dot{X}_i$, $K_{pi} Y_i$ and $C_{pi} \dot{Y}_i$

are in-phase stiffness force, in-phase damping force, out-of-phase stiffness force and out-of-phase damping force respectively. The in-phase force is in line but opposite to the displacement or velocity vector and the out-of-phase force leads 90° from the displacement or velocity vector. The third group

$$\left[K_{HDi} Y_i (\dot{\phi} - K_{Fi} \omega_i) + C_{HDi} \dot{Y}_i (\dot{\phi} - C_{Fi} \omega_i) \right]$$

represents the rotor surface forces at a rotor station. They are the combined rotor-spin and whirl speed sensitive out-of-phase stiffness and damping forces. These forces may be generated in a non-synchronous whirl motion such as that encountered in a fluid-film bearing or similar components. The last group,

$$A_i \left\{ \left[(\dot{\phi} - \dot{\phi}_0) \left[N_{Bik} + B_{Bik} \left(\sqrt{X_{Bi}^2 + Y_{Bi}^2} - \rho_{Bik} \right) \right] + K_{Bik} \right. \right.$$

$$\left. \left. \left[\left(X_{Bi}^2 + Y_{Bi}^2 \right)^{\frac{H_{Bik} - 1}{2}} + D_{Bik} + \frac{E_{Bik}}{\sqrt{X_{Bi}^2 + Y_{Bi}^2}} \right] X_{Bi} + C_{BXi} \dot{X}_i \right\}$$

denotes the nonlinear bearing forces at a bearing station. $\dot{\phi}_o$ is a rotor spin speed constant and $\dot{\phi}$ is the actual rotor spin speed. $(\dot{\phi} - \dot{\phi}_o)\, N_{BiK}$ is used to simulate the spin speed sensitive bearing stiffness at zero journal displacement. B_{BiK} is the spin speed and journal displacement sensitive stiffness coefficient. X_{Bi} and Y_{Bi} are the journal displacement coordinates and ρ_{BiK} is the lower bearing displacement limit for K th bearing stiffness section of bearing i. K_{BiK} is the journal displacement sensitive bearing stiffness.

$$K_{Bik}\left[\left(X_{Bi}^2+Y_{Bi}^2\right)^{\frac{H_{BiK}-1}{2}}+D_{Bik}+\frac{E_{Bik}}{\sqrt{X_{Bi}^2+Y_{Bi}^2}}\right]X_{Bi}$$

is the nonlinear bearing stiffness force function, and $C_{BXi}\dot{X}_i$ the X-component damping force at the bearing.

Eqs. (4) and (5) describe all the specified moments defined about Y and X axis respectively.

$$\begin{aligned}-\left(l_{i-1}+l_i\right)M_{Yi} = {} & I_{Di}\Big[\ddot{X}_{i+1}-\ddot{X}_{i-1}-\beta_i\left(l_{i-1}+l_i\right)\Big(\ddot{\phi}\,\mathrm{SIN}(\phi+\gamma_i) \\ & \qquad +(\dot{\phi})^2\cos(\phi+\gamma_i)\Big)\Big] \\ & +I_{Pi}\dot{\phi}\Big[\left(\dot{Y}_{i+1}-\dot{Y}_{i-1}\right)+\beta_i\left(l_{i-1}+l_i\right)\dot{\phi}\cos(\phi+\gamma_i)\Big] \\ & +K_{\phi i}\left(X_{i+1}-X_{i-1}\right)+C_{\phi i}\left(\dot{X}_{i+1}-\dot{X}_{i-1}\right) \\ & +K_{\phi Pi}\left(Y_{i+1}-Y_{i-1}\right)+C_{\phi Pi}\left(\dot{Y}_{i+1}-\dot{Y}_{i-1}\right) \\ & +K_{\phi HDi}\left(\dot{\phi}-K_{\phi Fi}\omega_i\right)\left(Y_{i+1}-Y_{i-1}\right)+C_{\phi HDi}\left(\dot{\phi}-C_{\phi Fi}\omega_i\right)\left(\dot{Y}_{i+1}-\dot{Y}_{i-1}\right)\end{aligned} \tag{4}$$

$$
\begin{aligned}
\left[l_{i-1}+l_i\right] M_{X_i} = {} & I_{Di}\left[\left(\ddot{Y}_{i+1}-\ddot{Y}_{i-1}\right)+\beta_i\left(l_{i-1}+l_i\right)\left[\ddot{\phi}\cos\left(\phi+\gamma_i\right)\right.\right. \\
& \left.\left. -\left(\dot{\phi}\right)^2 \sin\left(\phi+\gamma_i\right)\right]\right] \\
& -I_{pi}\dot{\phi}\left[\left(\dot{X}_{i+1}-\dot{X}_{i-1}\right)-\beta_i\left(l_{i-1}+l_i\right)\dot{\phi}\sin\left(\phi+\gamma_i\right)\right] \\
& +K_{\phi i}\left(Y_{i+1}-Y_{i-1}\right)+C_{\phi i}\left(\dot{Y}_{i+1}-\dot{Y}_{i-1}\right) \\
& -K_{\phi pi}\left(X_{i+1}-X_{i-1}\right)-C_{\phi pi}\left(\dot{X}_{i+1}-\dot{X}_{i-1}\right) \\
& -K_{\phi HDi}\left(\dot{\phi}-K_{\phi Fi}\,\omega_i\right)\left(X_{i+1}-X_{i-1}\right)-C_{\phi HDi}\left(\dot{\phi}-C_{\phi Fi}\,\omega_i\right)\left(\dot{X}_{i+1}-\dot{X}_{i-1}\right.
\end{aligned}
\tag{5}
$$

The first group in Eq. (4), and similarly with Eq. (5)

$$
I_{Di}\left[\ddot{X}_{i+1}-\ddot{X}_{i-1}-\beta_i\left(l_{i-1}+l_i\right)\left(\ddot{\phi}\,\mathrm{SIN}\left(\phi+\gamma_i\right)+\left(\dot{\phi}\right)^2\cos\left(\phi+\gamma_j\right)\right)\right]
$$

denotes the inertial moment resulting from the transverse mass moment of inertia at a rotor station. In calculating the bending moments at a rotor station the slopes and their time derivatives are approximated by that of the chord joining the rotor centers at the adjacent stations. Using the chord approximations of the slopes the number of unknowns involved in the solution of the rotordynamics equations was substantially reduced. Two additional unknown variables at each station, i.e., the force and moment slope coefficients would otherwise have to be included in the rotordynamics equations. β_i and γ_i are the initial misalignment and angular orientation of the axis of the mass moments of inertia with respect to rotor elastic center line and initial rotor spin angular position respectively.

The second group,

$$I_{\rho i}\,\dot{\phi}\left[\left(\dot{Y}_{i+1}-\dot{Y}_{i-1}\right)+\beta_i\left(l_{i-1}+l_i\right)\dot{\phi}\cos\left(\phi+\gamma_i\right)\right]$$

represents the inertia moment from the polar mass moment of inertia at a rotor station.

The third group

$$K_{\phi i}\left(X_{i+1}-X_{i-1}\right)+C_{\phi i}\left(\dot{X}_{i+1}-\dot{X}_{i-1}\right)$$

denotes the rotor surface forces and they are the in-phase stiffness and damping moment due to a rotor misalignment motions with respect to the rotor casing or bearing at a rotor station.

The fourth group,

$$K_{\phi Pi}\left(Y_{i+1}-Y_{i-1}\right)+C_{\phi Pi}\left(\dot{Y}_{i+1}-\dot{Y}_{i-1}\right)$$

similarly represents the rotor surface moments consisting of the out-of-phase stiffness and damping moment from rotor misalignment motions.

The last group,

$$K_{\phi HDi}\left(\dot{\phi}-K_{\phi Fi}\omega_i\right)\left(Y_{i+1}-Y_{i-1}\right)+C_{\phi HDI}\left(\dot{\phi}-C_{\phi Fi}\omega_i\right)\left(\dot{Y}_{i+1}-\dot{Y}_{i-1}\right)$$

denotes the rotor surface moments. They are the out-of-phase, rotor spin and whirl sensitive stiffness and damping moments, similar to the third group in Eq. (2) or (3).

Eqs. (6) and (7) define the force and moment and equilibrium relationships in XZ and YZ planes. This is where the force and moment influence coefficients c_{ij} and b_{ij} are used to solve the basic rotordynamics equations for every rotor station except that for the first and last non-linear bearings. The basic rotordynamics equations for the first and last bearing stations are Eqs. (8) through (11). To solve the system of rotordynamics equations, first convert into the rotating coordinates and then substitute F_{xi}, F_{yi}, M_{yi} and M_{xi} functions in Eqs. (2) through (5) into Eqs. (6) through (11). The values of c_{ij} and b_{ij} used are computed in subroutine "INFLC0."

Eqs. (1) and (6) through (11) are solved simultaneously for the acceleration terms by means of ISIMDD subroutine. Using the current values of the acceleration and velocity terms, integration over a time interval will be made to obtain the solution for a rotor motion corresponding to a new real time. The process repeats for each time interval until a desired total time period for rotor motion is covered.

The influence coefficients are calculated by finding the rotor deflections (in.) at various stations when a unit transverse force (1 lb) or unit transverse moment (1 in-lb) is applied at a station. Rotor deflections considering rotor shear and bending elasticity are separately computed and then combined. The influence coefficients are based on the rotor-configuration supported on two knife-edge bearings located at the first and last support bearings. The rotor dynamic relationship at these two bearings are defined by Eqs. (8) through (11). Sample results in force and moment influence coefficients are shown in Appendix A. The meaning of row and column numbers in relation to the definitions of the influence coefficients is explained.

$$\sum_{i}^{n} \left[F_{xi} C_{ij} + M_{yi} b_{ij} \right] + X_{b1} + \left(X_{b2} - X_{b1} \right) \frac{Z_j - S}{L} - X_j = 0 \qquad (6)$$

$$\sum_{i}^{n} \left[F_{yi} C_{ij} - M_{xi} b_{ij} \right] + Y_{b1} + \left(Y_{b2} - Y_{b1} \right) \frac{Z_j - S}{L} - Y_j = 0 \qquad (7)$$

Eqs. (8), (9), (10) and (11) denote the moment and force equilibrium relationships for the first and last nonlinear bearing station in X-Z and Y-Z plane, respectively. These equations are solved simultaneously with Eqs. (6) and (7).

$$\sum_{i}^{n}\left[\left(Q - Z_i\right)F_{X_i} - M_{Y_i}\right] = 0 \tag{8}$$

$$\sum_{i}^{n} F_{X_i} = 0 \tag{9}$$

$$\sum_{i}^{n}\left[\left(Q - Z_i\right)F_{Y_i} + M_{X_i}\right] = 0 \tag{10}$$

$$\sum_{i}^{n} F_{Y_i} = 0 \tag{11}$$

Eqs. (12) and (14) denote the force equilibrium relationships at the bearing-and-mount station along X and Y axis, respectively. The bearing and mount damping forces are not included. Since zero bearing mass assumption is used, the stiffness force in the bearing should only equal that in the mount and similar relationship exists for the damping forces.

$$A_i \left\{ \left([\dot{\phi} - \dot{\phi}_0] [N_{Bik} + B_{Bik}(\rho_{BXi} - \rho_{Bik})] + K_{Bik} \right) \left(\rho_{BXi}^{(H_{Bik}-1)} + D_{Bik} + \frac{E_{Bik}}{\rho_{BXi}} \right) X_{Bi} - K_{MXi} [X_i - X_{Bi}] \right\} = 0 \tag{12}$$

Eqs. (13) and (15) are expressions for ρ_{BXi} and ρ_{BYi} used in Eqs. (12) and (14) respectively.

$$\rho_{BXi} = \sqrt{X_{Bi}^2 + \left[\frac{Y_i}{\frac{K_{MXi}}{K_{MYi}} \left(\frac{X_i}{X_{Bi}} - 1 \right) + 1} \right]^2} \tag{13}$$

$$A_i \left\{ \left([\dot{\phi} - \dot{\phi}_0] [N_{Bik} + B_{Bik}(\rho_{BYi} - \rho_{Bik}) + K_{Bik} \right) \left(\rho_{BYi}^{(H_{Bik}-1)} + D_{Bik} + \frac{E_{Bik}}{\rho_{BYi}} \right) Y_{Bi} - K_{MYi} [Y_i - Y_{Bi}] \right\} = 0 \tag{14}$$

$$\rho_{BYi} = \sqrt{Y_{Bi}^2 + \left[\frac{X_i}{\frac{K_{MYi}}{K_{MXi}} \left(\frac{Y_i}{Y_{Bi}} - 1 \right) + 1} \right]^2} \tag{15}$$

Eqs. (12) through (15) establish the nonlinear bearing and mount stiffness forces along X and Y axes. The purpose of these equations is to solve for the nonlinear bearing journal displacements through an interative process, a faster method than the closed form solution.

Eqs. (16) through (21) define the various coordinates used in the basic rotordynamics equations. Under the zero bearing mass assumption, the bearing damping can be expressed in terms of the combined bearing-and-mount damping forces. The relationships are shown in Eqs (16) through (19).

$$\dot{X}_{Bi} = \frac{C_{BXi}\,\dot{X}_i}{C_{Bi}} \tag{16}$$

$$\dot{Y}_{Bi} = \frac{C_{BYi}\,\dot{Y}_i}{C_{Bi}} \tag{17}$$

$$\dot{X}_{Mi} = \frac{C_{BXi}\,\dot{X}_i}{C_{MXi}} \tag{18}$$

$$\dot{Y}_{Mi} = \frac{C_{BYi}\,\dot{Y}_i}{C_{MYi}} \tag{19}$$

Also based on zero-bearing-mass assumption the journal and mount displacement are collinear as shown in Eqs. (20) and (21).

$$X_{mi} = X_i - X_{Bi} \tag{20}$$

$$Y_{mi} = Y_i - Y_{Bi} \tag{21}$$

Eq. (22) defines the whirl speed of shaft center at station i.

$$\omega_i = \frac{\dot{Y}_i X_i - \dot{X}_i Y_i}{X_i^2 + Y_i^2} \tag{22}$$

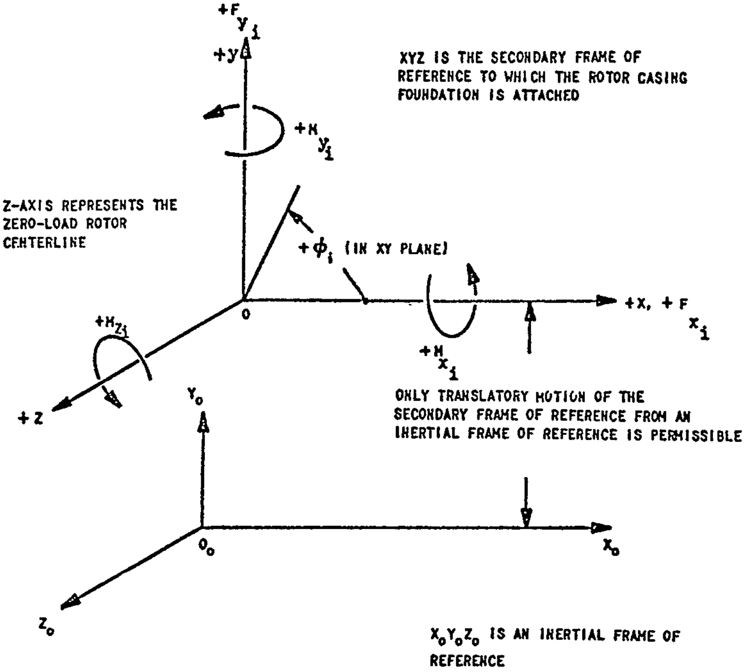
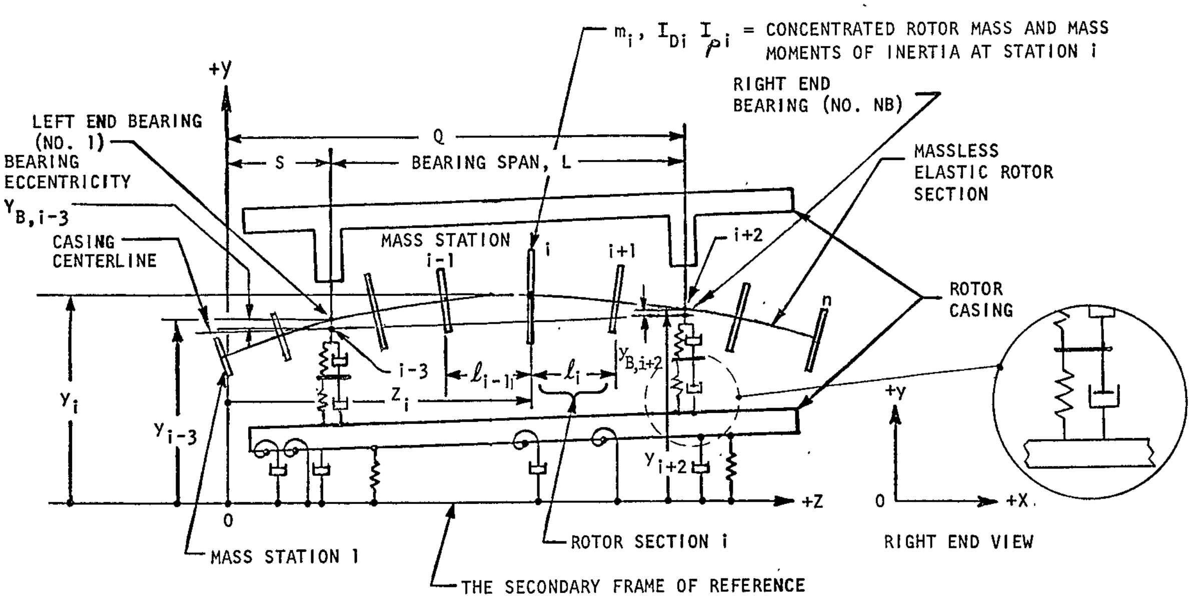


Figure 1 - Relation Between the Secondary Frame of Reference and an Inertial Frame of Reference



NOTE: 1. X_i AND $X_{B,i}$ ARE NOT SHOWN IN THE SKETCH

2. THE COORDINATES x_i AND y_i DO NOT NECESSARILY REPRESENT THE PHYSICAL DISTANCES AS SHOWN. THEY SHOULD BE INTERPRETED AS THE DISPLACEMENT FROM THEIR CORRESPONDING ZERO-LOAD POSITIONS.

3. IN THE PRESENT ANALYSIS, THE CASING IS ASSUMED TO BE RIGIDLY ATTACHED TO THE SECONDARY FRAME OF REFERENCE.

Figure 2. Schematic of the General Physical Model

The rotating coordinate system (Fig. 3) is used to reduce the integration time as it provides lower acceleration magnitude compared with that from conventional stationary coordinate system. The origin of the rotating coordinates coincides with the static rotor center at a station.
The static rotor center corresponds to that caused by g_x and g_y for the rotor with zero spin speed. The rotating coordinates used are synchronous with the rotor displacement vector at that rotor station. Let the static deflections at a rotor station be ΔX_i and ΔY_i; and ρ_i and θ_i be the displacement vector and angular displacement of the rotating coordinates respectively. Then

$$
\begin{aligned}
X_i &= \rho_i \mathrm{COS}\ \theta_i + \Delta X_i \\
Y_i &= \rho_i \mathrm{SIN}\ \theta_i + \Delta Y_o
\end{aligned}
$$

Differentiate once and twice with respect to time and using appropriate substitution, one obtains,

$$
\begin{aligned}
\dot{X}_i &= \dot{\rho}_i \mathrm{COS}\ \theta_i - \rho_i \dot{\theta}_i \mathrm{SIN}\ \theta_i \\
\dot{Y}_i &= \dot{\rho}_i \mathrm{SIN}\ \theta_i + \rho_i \dot{\theta}_i \mathrm{COS}\ \theta_i \\
\ddot{X}_i &= -\dot{\theta}_i \dot{Y}_i - \dot{\rho}_i \dot{\theta}_i\ \mathrm{SIN}\ \theta_i + \ddot{\rho}_i \mathrm{COS}\ \theta_i - \ddot{\theta}_i \rho_i \mathrm{SIN}\ \theta_i \\
\ddot{Y}_i &= \dot{\theta}_i \dot{X}_i + \dot{\rho}_i \dot{\theta}_i\ \mathrm{COS}\ \theta_i + \ddot{\rho}_i \mathrm{SIN}\ \theta_i + \ddot{\theta}_i \rho_i \mathrm{COS}\ \theta_i
\end{aligned}
$$

The variables X_i, Y_i, and their time derivatives in the rotordynamic equations are then replaced with the equivalent rotating coordinates prior to the solution of the rotordynamics equations.

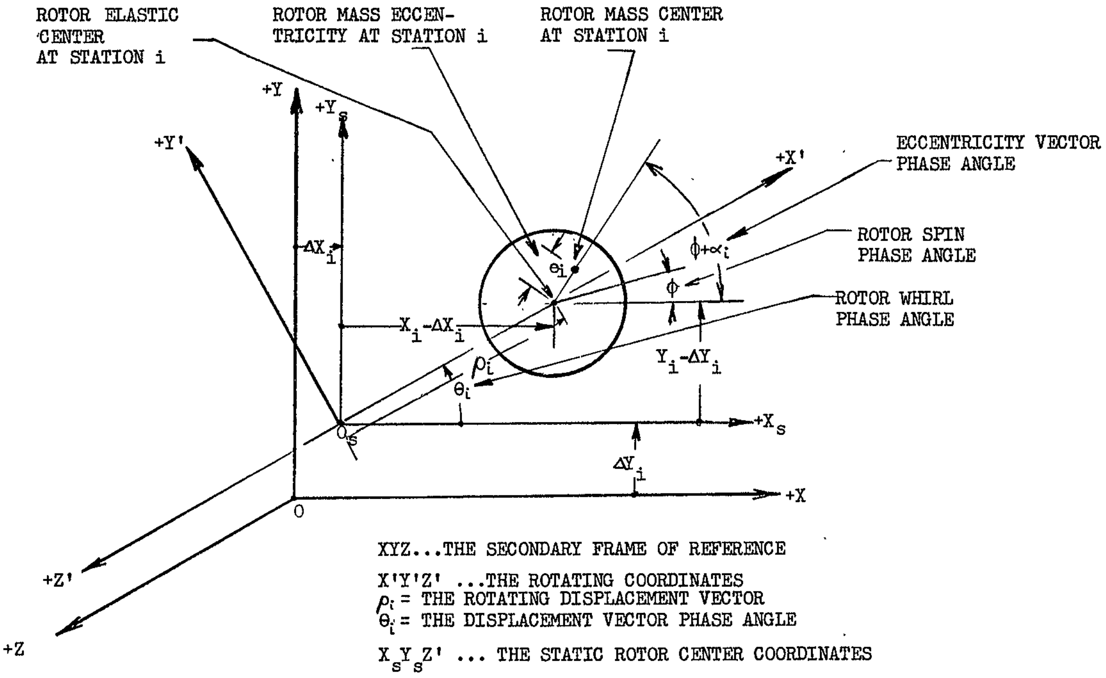


Figure 3 Rotating and Secondary Coordinate Systems Used in the Mathematical Model

NOMENCLATURE

A_i = bearing location notation, $A_i = 1$ implies a bearing at station i; $A_i = 0$, no bearing at station i dimensionless

b_{ij}, C_{ij} = moment and force influence coefficient, in/(in-lb) and in/lb, respectively

B_{Bik} = rotative speed sensitive bearing stiffness displacement coefficient for kth stiffness section of bearing i; for dimension, see Note 2

C_{Bi} = bearing damping coefficient for $\dot{X}_{Bi}$ and $\mathring{Y}_{Bi}$, (lb-sec)/in

C_{BXi}, C_{BYi} = equivalent bearing-and-mount damping coefficients for $\dot{X}_i$, $\dot{Y}_i$, respectively, (lb-sec)/in

C_{Fi} = whirl frequency factor for damping force coefficient, dimensionless

C_{HDi} = out-of-phase whirl-frequency sensitive damping force coefficient, (lb-sec^2)/(in-radian)

C_i, $C_{\phi i}$ = in-phase damping force and moment coefficient due to relative motion between rotor and casing (lb-sec)/in and (in-lb-sec)/rad, respectively

C_{MXi}, C_{MYi} = non-isotropic mount damping coefficient for $\dot{X}_{Mi}$, $\mathring{Y}_{Mi}$, respectively, (lb-sec)/in

C_{pi}, $C_{\phi pi}$ = out-of-phase damping force and moment coefficient due to relative motion between rotor and casing (lb-sec)/in and (in-lb-sec)/rad, respectively

C_{Zi} = torsional friction exponent, Eq. (1), dimensionless

C_{Z1i}, C_{Z2i} = torsional friction coefficients, Eq. (1), dimension of $C_{Z1i}\,\dot{\phi}^{CZi}$ or $C_{Z2i}\dot{\phi}$ in in-lb

$C_{\phi i}$ = damping moment coefficient, (lb-in-sec)/rad.

$C_{\phi Fi}$ = whirl-frequency factor for damping moment coefficient, dimensionless

$C_{\phi HDi}$ = out-of-phase, whirl frequency sensitive, damping moment coefficient, (lb-in-sec^2)/radian^2

D_{BiK}, E_{BiK} = nonlinear bearing stiffness coefficients for kth stiffness section of bearing i. For dimension, see Note 1.

e_i = mass eccentricity at rotor station i, in

F_{xi}, F_{yi} = forces in X and Y direction, lb

g_x, g_y = gravity or g-loading in X and Y direction, in/sec^2

H_{Bik} = nonlinear bearing stiffness exponent for kth stiffness section of bearing i. For dimension, see Note 1

I_{Di}, $I_{\rho i}$ = rotor diametral and polar mass moments of inertia at station i, lb-in-sec^2, respectively

K_{BiK} = rotative-speed sensitive bearing stiffness coefficient for Kth stiffness section of bearing i, lb/in

K_{Fi} = whirl frequency factor for stiffness force coefficient, dimensionless

K_i, $K_{\phi i}$ = in-phase stiffness force and moment coefficient due to relative motion between rotor and casing at station i, lb/in and (in-lb)/rad, respectively.

K_{HDi}, $K_{\phi HDi}$ = out-of-phase hydrodynamic force and moment coefficient of balance piston, (lb-sec)/(in-rad) and (lb-in-sec)/rad^2, respectively

K_{MXi}, K_{MYi} = non-isotropic stiffness coefficients for X_{Mi}, Y_{Mi}, respectively, lb/in

K_{pi}, $K_{\phi pi}$ = out-of-phase stiffness force and moment coefficient due to the relative motion between rotor and casing at station i, lb/in and in-lb/rad, respectively

$K_{\phi Fi}$ = whirl frequency factor for stiffness moment coefficient, dimensionless

l_i = rotor section length between the adjacent mass stations i and i + 1, in

L = length between bearings No. 1 and NB, in

m_i = rotor mass at station i, (lb-sec^2)/in

M_{xi}, M_{yi}, M_{zi} = moments about X, Y, and Z axes, respectively, in/lb

M_Z = exponent for speed sensitive rotor drive torque coefficient

M_{Z1}, M_{Z2}, M_{Z3} = coefficients for rotor drive torque, dimensions of $M_{Z1}\dot{\phi}^{MZ}$, $M_{Z2}\dot{\phi}$, or M_{Z3} are in-lb

n = total number of rotor discrete mass stations, dimensionless

N_{Bik} = rotative-speed sensitive bearing stiffness coefficient for Kth stiffness section of bearing i. For dimension, see Note 2.

NB = The last bearing number, or total number of non-linear stiffness bearings.

ℓ = length between No. NB bearing and mass station 1, in

S = length between No. 1 bearing and the mass station 1, in

t = real time, sec

$X_{Bi}, Y_{Bi}, \dot{X}_{Bi}, \mathring{Y}_{Bi}$ = bearing displacement and velocity coordinates, in and in/sec, respectively

$X_{Mi}, Y_{Mi}, \dot{X}_{Mi}, \mathring{Y}_{Mi}$ = mount displacement and velocity coordinates, in and in/sec, respectively

X_i, Y_i = displacement of the rotor geometrical center from its zero-load position in X and Y direction, respectively, at station i, in

XYZ = the secondary frame of reference to which the rotor casing is attached (Fig. 3)

$X_oY_oZ_o$ = an inertial frame of reference (Fig. 1)

Z_i = Z coordinate of ith mass from first mass, in

α_i = orientation of the ith mass eccentricity vector from that of the first rotor mass measured in the direction of rotation, rad

ω_i = angular whirl velocity of the rotor geometric center at station i, rad/sec

$\mathring{\phi}_o$ = a spin speed sensitive bearing stiffness parameter, rad/sec as defined in Eq. (2), (3), (12) and (14)

ϕ = spin speed angular displacement of the torsionally rigid rotor, measured from the positive X-axis, rad

NOTE 1: Group dimension of $K_{Bik}\left[(X_{Bi}^2 + Y_{Bi}^2)^{\frac{H_{Bik}-1}{2}} + D_{Bik} + \frac{E_{Bik}}{X_{Bi}^2 + Y_{Bi}^2}\right]$ is lb/in.

NOTE 2: Group dimension of $(\dot{\phi} - \phi_o)\ N_{Bik} + B_{Bik}\left(\sqrt{X_{Bi}^2 + Y_{Bi}^2} - \rho_{Bik}\right)$ is the same as that of K_{Bik}.

β_i = initial misalignment between the axis of the mass moments of inertia and the elastic axis of rotor section i, radians

γ_i = angular position of the X-Y plane projection of the axis of ith mass moments of inertia measured from that of the first, radians

ρ_{Bik} = lower bearing displacement limit for kth bearing stiffness section of bearing i, in

ρ_{BXi}, ρ_{BYi} = as defined by Eqs. (13) and (15), respectively, in.

Subscripts

i = pertinent to the ith station or ith rotor section

b1, b2 = pertinent to first and last nonlinear bearing stations respectively

Time Derivative Notation

$\dot{X}, \ddot{X} \ldots = \frac{dx}{dt}, \frac{d^2x}{dt^2} \ldots$ respectively, in/sec, in/sec^2

$\dot{\phi}, \ddot{\phi} = \frac{d\phi}{dt}, \frac{d^2\phi}{dt^2}$, respectively, rad/sec, rad/sec^2

VERIFICATION OF THE COMPUTER PROGRAM

After debugging the computer program three stages of verification of the program were made. The detailed description of each stage of verification follows:

Verification of the Basic Mathematical Formulation

As a preliminary checkout of the "Fortran" coding of the mathematical formulation, a 5-station flexible rotor-bearing configuration, operating under a steady-state condition was chosen. Hand computed, initial rotor motion conditions simulating a steady-state operation were used. The hand calculation was based on the equilibrium between the centrifugal loading and bearing restoring forces. The computer result in rotating coordinates indicated a very small residual acceleration on the order of 10^{-9} of the magnitude of the centrifugal acceleration.

The residual net acceleration, not being zero, is due to input data not having a sufficient number of significant figures and computation round-off errors. The comparative results are shown in Table 1.

Verification of the Computation Results

The computation results of a 17-station rotor model operating under a steady-state condition were compared with that from an existing rotor-dynamic response program based on a matrix iteration technique and they were found in good agreement within a maximum of 1.5% deviation for the case studied. Two different approaches in mathematical formulations are used in these two types of computer programs. The accuracy of the flexible rotor dynamic computer program is considered to be adequate. Table 2 lists the comparative results.

Evaluation of the Accuracy and Convergence of the Computation Process

Using a 5-station rotor test model with a spin speed of 95.5 rpm, a computation with many integration steps was performed. A reasonably large

TABLE 1

5-STATION ROTOR-BEARING CONFIGURATION CHECKOUT COMPARATIVE RESULTS

Station No.	1	2	3	4	5
DISPLACEMENT VECTORS					
Manually Calculated	.0017355"	.0009999"	.00077360"	.0009999"	.0017355"
Computer Program	.0017356"	.001"	.00077365"	.001"	.0017356"
COMPUTER*CALCULATED ACCELERATION IN ROTATING COORDINATES					
Radial	-3.716×10^{-9} in/sec	3.440×10^{-9}	$.1986\times10^{-9}$	3.440×10^{-9}	-3.716×10^{-9}
Tangential	8.116×10^{-16}	-1.970×10^{-14}	1.315×10^{-14}	-2.503×10^{-14}	8.116×10^{-16}

* Predicted Acceleration = zero.

TABLE 2

COMPARISON OF MARK 25, 17 MASS MODEL
NEW PROGRAM VS OLD PROGRAM

Station	Deflection at Station x 10^{-5}		% Difference
	New Transient Program	Old Steady-State Response Program	
1 (Inducer) End	.3554	.3548	.17
2	.3145	.3140	.16
3	.2028	.2023	.25
4	.1419	.1415	.28
5	.0845	.08414	.43
6	.04133	.04098	.85
7	.02310	.02276	1.50
8	.05076	.05099	.45
9	.1377	.1377	0
10	.1908	.1907	.05
11	.2364	.2363	.04
12	.3302	.3300	.06
13	.3517	.3514	.09
14	.3963	.3960	.08
15	.4159	.4156	.07
16	.5093	.5090	.06
17	.9782	.9784	.02

integration tolerance of 10% was used. The result in rotor displacement appears to stay close to the exact value. The instantaneous whirl to spin frequency ratio indicated a substantial oscillation at times, although the trend of convergence appeared definite. With a smaller integration tolerance as is usual (less than 10%), the magnitude of oscillations should be reduced accordingly. The integrated whirl displacement versus time function (not shown here) was reasonably smooth. Two graphs showing the variation of a deflection vector and whirl-to-spin frequency ratio are attached for reference (Figs. 4 and 5).

As evident from Fig. 4 and 5, the instantaneous whirl to spin frequency ratio appears to be coupled to the rotor displacement motion. The energy variation in the displacement motion is complemented by that in whirl motion through the mechanism similar to that of Coriolis acceleration. Because of the radius of whirl motion being rather small the angular velocity change is greatly magnified for a small change in rotary kinetic energy. Since the computer program for this rotor configuration was not run for a sufficient time period, it cannot be certain whether the relatively large oscillations between 0.12 and 0.14 second of real time were manifestation of numerical instability. It, however, does not appear likely to be the case.

The minimum DT (Delta Time) to avoid instability depends on the rotor mass-to-stiffness parameter ratio and no definite such relationships have yet been established.

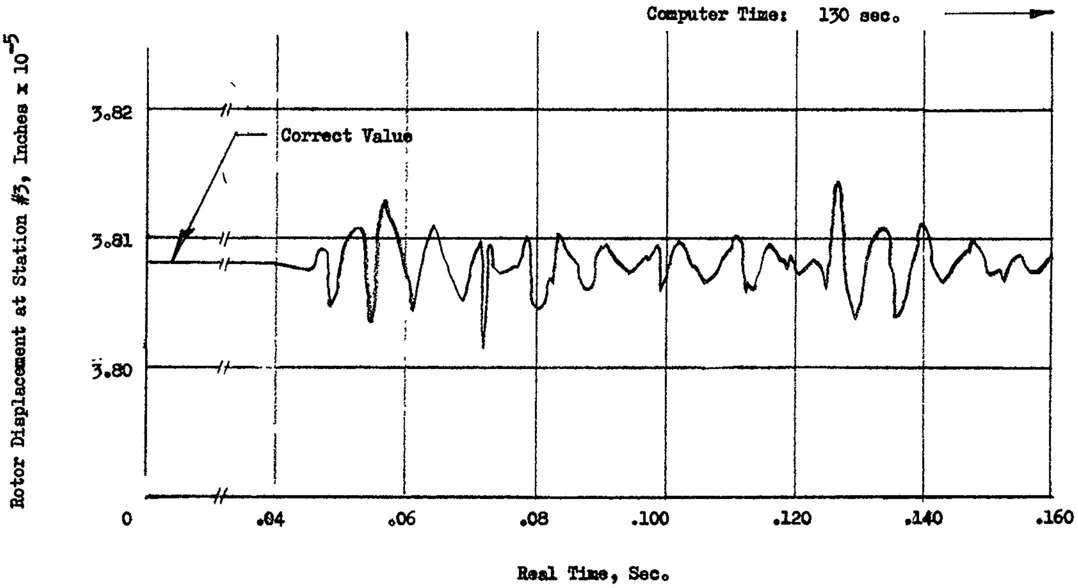


Figure 4 Data from a 5-Mass 95.5 rpm Flexible Rotor
in a Steady-State Operation

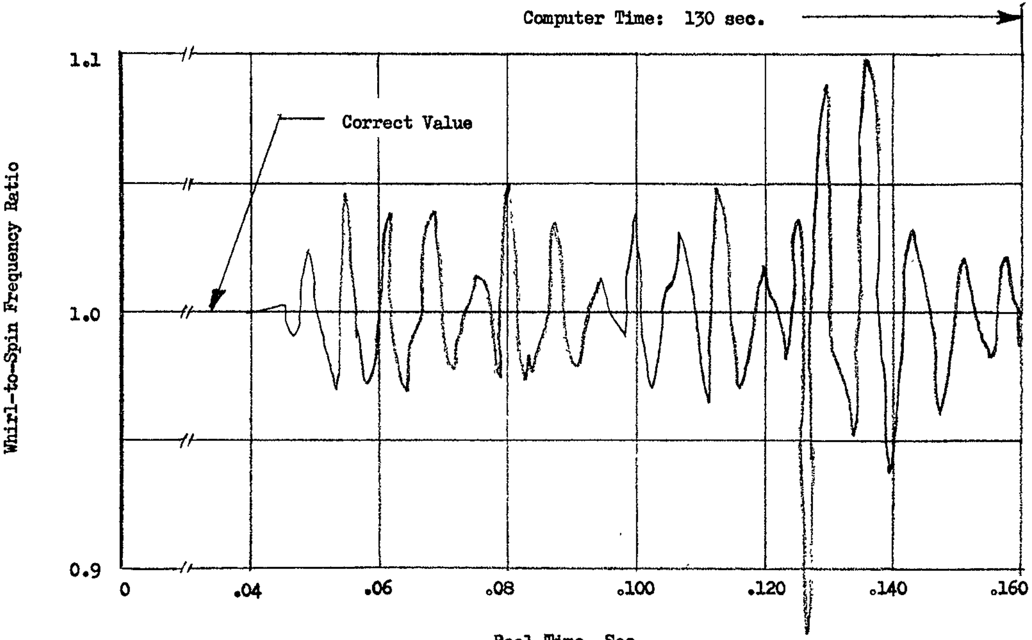


Figure 5 Data From a 5-Mass 95.5 rpm Flexible Rotor in a Steady-State Operation

CAPABILITY AND LIMITATIONS OF THE CURRENT COMPUTER PROGRAM

Capability of the Computer Program

The Computer Program was written to include effects of:

1. Rotor elasticity in shear and bending
2. A multiplicity of non-linear stiffness bearings each in turn supported on a non-isotropic stiffness and damping characteristic mount
3. General in-phase and out-of-phase stiffness and damping functions
4. Mass and mass moments of inertia unbalance vectors
5. Drive torque and dissipation functions

The application of the program may be illustrated as follows:

1. Transient spin speed analysis through critical speed range using speed sensitive drive torque and/or dissipation functions
2. Various whirl-to-spin frequency mode simulations in transient, steady-state or quasi (cyclic) steady-state operations
3. Study of the effects of rotor transverse gravity or constant acceleration
4. Investigation of the effects of general fluid dynamic or some electromagnetic excitation function with in-phase or out-of-phase, whirl and/or spin velocity sensitive, stiffness and/or damping functions.
5. Non-linear bearing stiffness effects
6. Study the effects of shear-and/or bending-wise flexible couplings on rotordynamic performance
7. Analysis of a general rotor motion in a non-axisymmetric and non-steady state operation.

Limitations

Several major assumptions in the mathematical formulation of the simplified computer program were made. The assumptions and resulting limitations are itemized as follows:

Torsional Rigid Rotor Assumed. For most rotor design properties this assumption appears to be realistic. Deviations from the real solution would occur, only when substantial torsional oscillations coupled with transverse rotor motion are encountered. A multiple-rotor system with small dissipation coefficient and having pulsating drive torque of a frequency at, or near, a torsional critical speed of the rotor would be the case that a torsionally flexible rotor model would provide more accurate results.

Rotor Casing Parameter not Considered. For rotors with large casing-to-rotor mass ratio or reasonably rigid attachment between the casing and the foundation, the effects of the assumption are minimal.

Rotor Hysteretic Damping Including the Effects of Bolted or Press-fit Rotor Joints not Considered. This assumption does not materially affect the computation results unless an elastically bent rotor of substantial hysteretic property operates under a non-synchronous whirl condition, or under a substantial gravity or transverse acceleration loading.

COMPUTER PROGRAM USER'S INSTRUCTIONS

DESCRIPTION OF THE COMPUTER PROGRAM

The Simplified, Flexible Rotor, Transient-Speed Rotordynamics Analysis Computer program consists of a MAIN program, 12 subroutines and a major library subroutine (ISIMDD). The program is written for IBM 360, Model 65 and Fortran H compiler language. The CRT plotting language is to be compatible with that available at NASA Lewis Research Center, Cleveland, Ohio*. A brief description of the functions of the MAIN program and subroutines and a general flow diagram, Figure 6, is provided, as a guide for the use of the program.

---

*CRT Plotting Reference, Robert G. Kannenberg, Lewis Research Center, Cleveland, Ohio: "Cinematic-Fortran Subprograms for Automatic Computer Microfilm Plotting, NASA TM X-1866" November 1969.

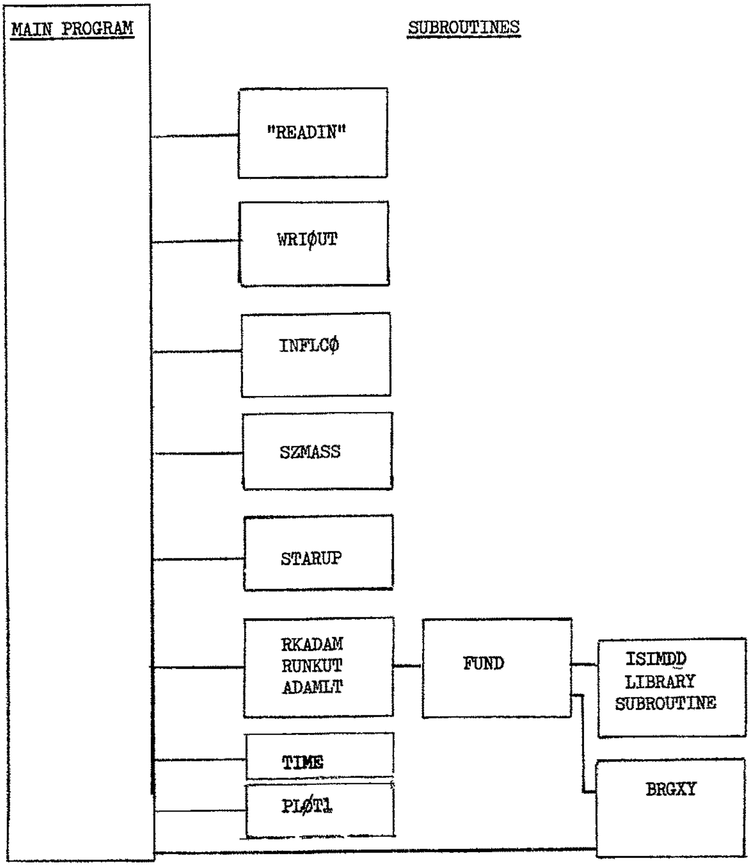


Figure 6 General Flow Diagram Relating Various Subroutines and the MAIN Program

PROGRAM OR SUBROUTINE NAME	FUNCTION
MAIN	This is the basic controlling program which originates the basic "call" statement. It also provides output write-out and CRT generating procedures. The rotor motion in stationary coordinates is converted into rotating coordinates before transmission to the integration subroutines. Similarly, the results of integration are first converted back to stationary coordinates before they are used in print-outs and CRT graphs.
READIN	This function is basically to *read input other than providing built-in average data values. "READIN" is called by "MAIN." 1. 2-card job descriptive title 2. NAMELIST/DATA 1/, the data which must be input 3. NAMELIST/DATA 2/, optional, when the input data in the data 2 group are not provided, the built-in average data values will be used. 4. Read punched cards from previous analysis for continued study. The read-in of the punch cards becomes effective when CØNTIN = 1 is specified in data as opposed to CØNTIN = 0 condition.

---

* Detail READIN information will be discussed in subsequent section.

PROGRAM OR SUBROUTINE NAME	FUNCTIONS
WRI0UT	This is to print out all the input data in groups according to their uses. Descriptive definitions and dimensional units are provided for each data set. "WRI0UT" is called by "MAIN."
STARUP	This subroutine is to generate accurate initial rotor motion conditions for starting computations in the basic program. Using the "STARUP" subroutine, computations in the basic program can proceed in an efficient manner. The present "STARUP" is in its simplified version; it includes only linear bearing and rotor excitation stiffness functions. Torque, damping and out-of-phase force parameters are not included. It is in a tentative status subject to updating to include all leftover parameters. "STARUP" is a steady-state subroutine and is called by "MAIN."
FUND	This subroutine provides Time derivatives required by the integration subroutines "RKADAM," "RUNKUT," and ADAMLT." To compute the derivatives, "FUND" obtains input data through "READIN," influence coefficient data from "INFLC0," and bearing data from "BRGXY."

PROGRAM OR SUBROUTINE NAME	FUNCTIONS
RKADAM, RUNKUT, & ADAMLT	These integration subroutines are based on an adjustable step, predictor-corrector, Adam-Moulton integration procedure. The starting points are established through forward difference Runge-Kutta approach although option in backward difference procedure is also provided. "TØLI" is an input data name which is used to regulate the computation tolerance in integration accuracy as may be desired. Higher accuracy may be obtained with smaller "TØLI" value such as .001 or .0001. Also, higher accuracy means longer computation time. Thus, an optimum balance between the degree of accuracy and computation time should be made to suit the circumstances. "RKADAM," "RUNKUT" and "ADAMLT" are called by "MAIN."
INFLCØ	This generates the required influence coefficients considering both bending and shear elasticity of the rotor. It is called by "MAIN" and the influence coefficients are also used by "STARUP" and "FUND."
ISIMDD	This is a simultaneous linear equation solution library subroutine which computes individual derivatives required by the integration subroutines. It is called by "FUND."

PROGRAM OR SUBROUTINE NAME	FUNCTIONS
SZMASS	This a subroutine which computes the rotor mass, transverse and polar moments of inertia, and rotor length coordinates. It is called by "MAIN."
BRGXY	This is a subroutine which computes journal displacements from their respective zero-load position The computations appropriately consider the nonlinear stiffness characteristics of the bearings, and non-isotropic mount force characteristics. "BRGXY" is called by "MAIN" AND "FUND."
TIME	This is to furnish real time for the iteration process from the tentative "real time" output of the integration subroutines.
PLOT1	It is to provide multiple bearing data for CRT plotting. It is used to generate journal deflections and bearing forces for a maximum number of 12 bearings. It is called by "MAIN."

INPUT PROCEDURE

In using the computer program a continuous rotor configuration is to be simulated by a discrete-mass rotor with appropriate massless elastic members in between adjacent rotor masses. In general, the minimum number of discrete masses used should be such that the rotordynamic mode shape can be possibly sustained. For instance, if a shaft operates in its third critical speed range the mode shape for a low stiffness bearing would be ; to sustain this mode a minimum of 5 masses is required. To obtain good accuracy in general, several times the minimum number of mass requirements are used. In rotor motion predominately influenced by mass eccentricity, damping, and stiffness function, the mode shape in a critical speed range may be substantially modified from that of a pure critical speed mode shape. Judgement must hence be exercised in selecting the number of discrete masses in representing a rotor configuration.

The rotor to be studied is first divided into consecutively numbered stations. The total number of stations may vary from 5 to 25, inclusively. Rotor sections between adjacent rotor stations are labeled with the same numbers as that of the left adjacent stations. The rotor property input data are appropriately subscripted according to the rotor station or section numbers. For non-linear stiffness bearing data two-dimensional subscripts are used. The first subscripts define their rotor station location and the second define the non-linear bearing stiffness sections.

The input data is printed prior to the writing of computational results

A detailed description of the input, output and usage of the program appears in the following sections. To supplement the input procedure description, a sample input code sheet is shown in Figure 7.

Input

A namelist input procedure is adopted as the major input format due to its flexibility in selecting input parameters and the liberal use of built-in average values when desired. For preliminary analysis, by making use of built-in data, the input data volume can be drastically reduced.

The complete input listing is as follows:

1. Read Title. Two consecutively located 80-character spaced cards must be used for job description title. The title description may use the first 72 spaces of each card and 73 through 80th space may be used for card identification number only.

2. NAMELIST/DATA 1/. The data group consists of key input data which must all be read in Namelist format. This sequence of read-in data is immaterial. Because the Data 1 parameters pertain to the rotor geometry and other essential descriptive information, average-value data cannot be built into the program. The physical input of these parameters is necessary. The names and their definitions contained in the NAMELIST/DATA 1/ are on page 44.

3. Name List/Data 2/. The majority of the input data is included in the "Data 2" namelist. All parameters contained in "Data 2" are optional input data; i.e., they need not be inputed if the built-in average values are adequate. However, a card with the name Data 2 is necessary even if the data field is empty.

The input names for Data 2, their definitions, and the built-in values are on page 45.

4. Reading Punched Cards. If continuation of a previous analysis is desired specify "CØNTIN=1" and attach the cards providing starting data from a previous computation.

CAUTION: If K is greater than 1 for one or more non-linear stiffness bearings, the input of double-subscripted BRØ(I,K) for appropriate K values must be input. Where I is the rotor station for the non-linear bearing, K varies from 1 through K for appropriate non-linear bearings. The input of BRØB (I,1) would be necessary when a different than built-in value of 0.005 inch is desired.

NAMELIST/DATA 1/

Name Used In The Computer Program	Equivalent Name in The Mathematical Model	Stored Value	Description
NS	N		Number of rotor stations (allowable range: $5 \leq NS \leq 25$).
TMAX			Total real time to be run, sec.
TSTØP			The computer time allowed for each set of data, minutes.
DD(NS-1)*			Outside diameters of rotor sections between adjacent rotor stations, in.
QL(NS-1)	l_i		Rotor section lengths between adjacent rotoı stations, in.
NB			Number of non-linear stiffness bearings. (Allowable range: $2 \leq NB \leq 12$)
IB(NB)			Rotor station numbers of non-linear stiffness bearings
K(NB)			Total number of stiffness sections for each of the non-linear stiffness bearings. (Allowable range: $1 \leq K \leq 6$)
ICØND			ICØND=1 Means starting a new rotor-bearing configuration. ICOND=0 Means read-in alternate initial conditions for the same rotor-bearing configuration.
CØNTIN			CØNTIN=0 Means starting a new rotor-dynamics analysis with initial conditions provided by the startup subroutine CØNTIN=1 Means continuation of a previous analysis by using the previous results on punched cards as the initial conditions.
FDØT			Initial rotor spin frequency, rpm.
WHIVEL			Initial rotor whirl frequency, rpm.

---

* (NS), (I) or (I, J) after a name specifies the name being an array of one or two dimension. The values of the indexes specify the current size of the array.

NAMELIST/DATA 2/

Name Used In The Computer Program	Equivalent Name in The Mathematical Model	Stored Value	Description
T		0	Initial real time, sec.
DT		.00001	Estimated initial integration step (real) time, sec. The computer program may modify the value of DT to suit tolerance requirements.
FD		0	Rotor spin angular displacement coordinate, degrees, (This value is added to data).
IASIGN		1	Rotor station at which whirl/spin frequency ratio will be plotted on CRT.
NØØRPM		1	The number of spin speeds in rpm at or near which 3-dimensional absolute rotor mode shape CRT graphs are required. The spin speed rpm values are listed under INPRPM array. (Allowable range: $0 \leq$ NØØRPM ≤ 50).
NPØINT		25	The number of points (one per integration step) for each CRT graph. (Allowable range: $1 \leq$ NPØINT ≤ 50). It applies to all CRT graphs except the 3-dimensional rotor mode shape plot.
CRT		0	CRT=0 Means CRT is not required, CRT=1 Means CRT is required.
ACCEL		0	ACCEL=0 Means a 3-dimensional steady-state rotor mode shape CRT would be provided if concurrently CRT=1. ACCEL=1 Means the steady-state rotor mode shape will not be provided.
INPRPM(NØØRPM)		0	The rotor spin speed rpm values at or near which CRT graphs for 3-dimensional absolute rotor mode shapes are required.
TØLI		.01	Integration tolerance, fraction.
TØLB		.001	Tolerance in computing bearing displacements, fraction.
GX	g_x	0	Gravity or G-loading in X direction, in/sec^2.

NAMELIST/DATA 2/

Name Used In The Computer Program	Equivalent Name in The Mathematical Mode	Stored Value	Description
GY	g_y	0	Gravity or G-loading in Y direction, in/sec^2.
TMZ	M_Z	0	Exponent for speed sensitive rotor drive torque
TMZ1	M_{Z1}	0	Coefficient for rotor drive torque
TMZ2	M_{Z2}	0	Coefficient for rotor drive torque, (in-lb-sec)/rad
TMZ3	M_{Z3}	0	Coefficient for rotor drive torque, in/lb.
D(NS-1)		0	Inside diameters of rotor sections between adjacent rotor stations, in.
DN(NS-1)		.283	Material densities of rotor sections between adjacent stations, lb/in^3.
EE(NS-1)		$3x10^7$	Young's moduli of rotor sections between adjacent stations, lb/in^2.
GG(NS-1)		$1.15x10^7$	Shear moduli of rotor sections between adjacent stations, lb/in^2.
EI(NS-1)		0	Direct input of the products of Young's moduli and area moments of inertia of rotor sections between adjacent stations, $lb\text{-}in^2$.
*GAK(NS-1)		0	Direct input of the products of shear moduli, cross-sectional areas and reciprocals of shear stress concentration factors between adjacent stations, lb.
AM(NS)		$1x10^{-16}$	Additional rotor masses at rotor stations, $(lb\text{-}sec^2)/in$.
AID(NS)		0	Additional rotor transverse mass moments of inertia at rotor stations, $(lb\text{-}in\text{-}sec^2)$.
AIRØ(NS)		$1x10^{-16}$	Additional rotor polar mass moments of inertia at rotor stations, $(lb\text{-}in\text{-}sec^2)$.

* GAK stands for $\frac{(GG)(\text{Rotor Cross-section Shear Area})}{K}$, where K denotes Maximum to average shear stress ratio expressible as: $\frac{4}{3}\frac{[(DD)^3 - D^3]}{[(DD)^2+D^2](DD-D)}$

NAMELIST/DATA 2/

Name Used In the Computer Program	Equivalent Name in The Mathematical Model	Stored Value	Description
ECC(NS)	e_i	$1x10^{-16}$	Rotor mass eccentricities at rotor stations, in.
ALFA(NS)	α_i	0	Phase angles for rotor mass eccentricity vectors at rotor stations measured from that of the first rotor station, degrees.
BETA(NS)	β_i	0	Initial misalignments between the axes of the mass moments of inertia and the elastic axes at rotor stations, degrees.
GAMMA(NS)	γ_i	0	Angular positions of the X-Y plane projections of the axes of mass moments of inertia at rotor stations measured from that at the first rotor station, degrees.
CZ(NS)	C_{Zi}	0	Torsional friction exponents at rotor stations, dimensionless.
CZ1(NS)	C_{Z1i}	0	Torsional friction coefficients at rotor stations, dimension of C_{Z1i}(FDØT)C_{Zi} in-lb.
CZ2(NS)	C_{Z2i}	0	Torsional friction coefficients at rotor stations, (in-lb-sec)/rad.
XKF(NS)	$K_{\phi i}$	0	Whirl-frequency factors for stiffness force coefficients at rotor stations, dimensionless.
XCF(NS)	$C_{\phi i}$	0	Whirl-frequency factors for damping force coefficients at rotor stations, dimensionless.
XKFF(NS)	$K_{\phi Fi}$	0	Whirl-frequency factors for stiffness moment coefficients at rotor stations.
XCFF(NS)	$C_{\phi Fi}$	0	Whirl-frequency factors for damping moment coefficients at rotor stations, dimensionless.
QK(NS)	K_i	0	In-phase stiffness force coefficients at rotor stations, lb/in.
QC(NS)	C_i	0	In-phase damping force coefficients at rotor stations, (lb-sec)/in.

NAMELIST/DATA 2/

Name Used In The Computer Program	Equivalent Name in The Mathematical Model	Stored Value	Description
QKP(NS)	K_{Pi}	0	Out-of-phase stiffness force coefficients at rotor stations, lb/in.
QCP(NS)	C_{Pi}	0	Out-of-phase damping force coefficients at rotor stations, (lb-sec)/in.
QKHD(NS)	K_{HDi}	0	Out-of-phase whirl and spin velocity sensitive stiffness force coefficients at rotor stations, (lb-sec)/(in-rad).
QCHD(NS)	C_{HDi}	0	Out-of-phase whirl and spin velocity sensitive damping force coefficients at rotor stations, (lb-sec^2)/(in-rad).
QKF(NS)	K_{Fi}	0	In-phase stiffness moment coefficients at rotor stations, in-lb.
QCF(NS)	C_{Fi}	0	In-phase damping moment coefficients at rotor stations, in-lb-sec.
QKPF(NS)	$K_{\phi Pi}$	0	Out-of-phase stiffness moment coefficients at rotor stations, in-lb.
QCPF(NS)	$C_{\phi Pi}$	0	Out-of-phase damping moment coefficients at rotor stations, in-lb-sec.
QKHDF(NS)	$K_{\phi HDi}$	0	Out-of-phase whirl and spin velocity sensitive stiffness moment coefficients at rotor stations, (lb-in-sec)/rad.
QCHDF(NS)	$C_{\phi HDi}$	0	Out-of-phase whirl and spin velocity sensitive damping moment coefficients at rotor stations, (lb-in-sec^2)/rad.
BKMX(NB) BKMY(NB)	K_{mxi} K_{myi}	10^{10}	Non-isotropic mount stiffness coefficients in X and Y-direction, respectively, for non-linear stiffness bearings, lb/in.
BCMX(NB)	C_{mxi}	1×10^{-16}	Non-isotropic mount damping coefficients in X-direction for non-linear stiffness bearings, (lb-sec)/in.
BCMY(NB)	C_{myi}	1×10^{-16}	Non-isotropic mount damping coefficients in Y-direction for non-linear stiffness bearings, (lb-sec)/in.
BCB(NB)	C_{Bi}	1×10^{-16}	Bearing damping coefficients for non-linear stiffness bearings, (lb-sec)/in.

Name Used In The Computer Program	Equivalent Name in The Mathematical Model	Stored Value	Description
*FDØFIX	$\dot{\phi}_o$	0	A bearing stiffness speed parameter, rpm
*BNB(NBK)	B_{NLK}	0	Rotor spin-speed sensitive bearing stiffness coefficients for K stiffness sections of non-linear stiffness bearing.
*BBB(NB, K)	B_{BiK}	0	Rotor spin-speed sensitive bearing stiffness coefficients for K stiffness sections of bearings for non-linear stiffness bearing.
*BRØB(NB, K)		.005 (for the 1st stiffness sections of all bearings)	Upper bearing-displacement limits for K stiffness sections of bearing, in.
*BKB(NB, K)	K_{BiK}	$1x10^6$	Non-linear bearing stiffness coefficients for K stiffness sections of bearing.
*BHB(NB, K)	H_{BiK}	1	Non-linear bearing stiffness exponents for K stiffness sections of bearing.
*BDB(NB, K)	D_{BiK}	0	Non-linear bearing stiffness coefficients for K stiffness sections of bearing.
*BEB(NB, K)	E_{BiK}	0	Non-linear bearing stiffness coefficients for K stiffness sections of bearing.

---

*See equations on p. 60

FORTRAN FIXED 10 DIGIT DECIMAL DATA

DECK NO. ______ PROGRAMMER ______ DATE ______ PAGE ____ OF ____ JOB NO. ______

	NUMBER	DESCRIPTION

```
 1  A  CHECKØUT  R
13  ØTØR  CØNFIGU
25  RATIØN  CØNSI
37  STS ØF  5 STA
49  TIØNS  AND  5
61  MASSES;  RØTØ
    IDENTIFICATION  73 |          | 80

 1  R  MASS  UNBAL
13  ANCES,  INCLUD
25  ED.  BEARINGS
37   ARE  AT  STAT
49  IØNS  1  &  5.A
61  PRIL/27/1970
    IDENTIFICATION  73 |          | 80

 1   &DATA1  NS=5
13  , TMAX=1,TSTØ
25  P=1.,DD=4*1.,Q
37  L=4*1.,NB=2,I
49  B=1,5,ICØND=
61  1,CØNTIN=0,F
    IDENTIFICATION  73 | DØT=100., | 80

 1   MXIVEL=100.,
13  K(1)=1.,K(5)=
25  1.,  &END
37
49
61
    IDENTIFICATION  73 |          | 80
```

```
 1   &DATA2  CRT=
13  1,ACCEL=1.,IN
25  PRPM=2,  &END
37
49
61
    IDENTIFICATION  73 |          | 80

 1  /*
13
25
37
49
61
    IDENTIFICATION  73 |          | 80

 1
13
25
37
49
61
    IDENTIFICATION  73 |          | 80

 1
13
25
37
49
61
    IDENTIFICATION  73 |          | 80
```

4. Read Punch-Card Input (Optional). The program has provision for continuing a previous rotordynamics analysis by using the results stored in punched cards from the previous analysis.

Double precision rotating coordinate rotor displacement and velocity values are stored in the punched cards. Each card contains 3 numerical values and a right adjusted card sequence identification number, beginning with 1. For this optional input, these cards must be placed consecutively and immediately after the regular input data of the first round, as shown below:

1. Title 2 cards
2. DATA 1/ which includes "CØNTIN=1" and "ICØND=1"
3. DATA 2/, optional
4. Punched Data Cards from previous computations when CØNTIN=1, and not to be used if CØNTIN=0

Data for alternate initial conditions on same rotor. Do not include ICØND data. (Includes data groups 1, 2, 3, and 4 as appropriate.)

Data for a new rotor configuration must include ICØND=1. (Includes data groups 1, 2, 3, and 4 as appropriate.)

/ * End Card

When read-in punched cards are desired "CØNTIN=1" must be included in /DATA 1/ located prior to the punched cards. When punched card read-in is effected, the regular, initial condition computation by "STARUP" will be by-passed.

Special Notes on Input

1. Alternate Provisions. In the input procedure there are two alternate input provisions. In certain rotor designs the flexibility of rotor may not be derived directly from the diameter and elastic modulus considerations, such as complex bolted rotor sections. In such circumstances, an optional combined EI or GAK may be more conveniently used in place of EE, DD, D or GG and shear stress concentration factor.

   EI is the product of Young's modulus of elasticity and sectional moment of inertia, GAK is the product of shear elastic modulus, rotor cross-sectional area and reciprocal of maximum to average shear stress ratios, where

   EE = Young's modulus of elasticity, psi
   GG = = Shear modulus of elasticity, psi
   DD = Outside rotor diameter, in
   D = Inside rotor diameter, in

   In using the option input, EI, EE set or GAK, GG set, only values of one of the two sets can be assigned. No values are to be assigned to the other set.

   The built-in average values of EE and GG for all rotor sections are:

   $$EE\ (I) = 3\text{x}10^7$$
   $$GG\ (I) = 1.5\text{x}10^7$$

   It should be noted that the option, EI or GAK, may be used for any rotor sections desired, and need not be used for all rotor sections.

2. Analysis with Various Alternate Operating Parameters. The program has the provision to analyze the effect of various sets of operating parameters on the performance of a given rotor-bearing

configuration. This can be accomplished by inputting; first, a complete data set including the rotor design and the initial set of values of operating parameters including "IC0ND=1" in Data 1/Name List; the second set of data for different values of certain parameters, but not including "IC0ND=1", may be input without repeating the same rotor design information. This procedure may be continued indefinitely until the analysis of a new rotor design is initiated. With the new rotor design data, "IC0ND=1" must be included in the first data input set.

The rotor-bearing configuration, referred to above, is defined as:

a. Rotor design, geometry and related physical properties which include the number of rotor stations (NS), transverse rotor gravity or accelerations (GX, GY), rotor diameters, length, density, elasticity, mass, and mass moments of inertia.

b. Bearing locations and number of bearings used.

The parameters other than that of rotor-bearing configurations may be repeated with different numerical values for a given rotor-bearing configuration. They are listed as:

a. Initial rotor motion conditions.

b. Bearing and mount damping and stiffness characteristics, number of non-linear stiffness sections.

c. Unbalance parameters for rotor masses and mass moments of inertia.

d. Rotor drive torque and torsional damping parameters.

e. All rotor in-phase and out-of-phase stiffness and damping parameters.

Namelist Input Procedure and Sample Input Data Sheet

According to the current North American Rockwell Corporation practice compatible with IBM 360 Fortran IV, compiler H language, the namelist input format is described in the following five pages.

A sample of a completed input data sheet is given in Appendix A.

NAMELIST INPUT-OUTPUT

Namelist offers a quick and relatively easy way of coding input-output. The programmer writes a NAMELIST statement, including one or more lists of variables and/or arrays he wishes to read in or print out, and assigns a name to each list. Thereafter, instead of rewriting an entire list, he references the list name in READ and WRITE statements.

On input, the number of items to be read and the place each one is to be stored is governed to a large extent by the input data cards. Each item is written in a format much like an arithmetic statement, with each variable explicitly named and set equal to the desired value. A few items or many can be read by the same READ statement, and the order of writing the data is unimportant.

On output, the members of the list will be printed in a standard format. The appearance of the printout is not suitable for formal reports, but is satisfactory as a substitute for a partial dump in error branches, for printout of input data, for intermediate results during checkout, or in any other case in which answers are needed and the format is not important

TRANSMITTAL STATEMENTS

General Form:

```
READ (a, b, END = c, ERR = d)

WRITE (a, b)
```

Where: a is the data set reference number.

b is the Namelist name.

See Section 204.2, Sequential Input-Output, for a complete description of these statements.

Note that Namelist cannot be used with a PUNCH statement. However, the statement

```
WRITE (14, b)
```

can be used to produce punched output.

NAMELIST STATEMENT

General Form:

```
NAMELIST /x/ a, b, ..., c /y/ d, e, ..., f
    /z/ g, h, ..., i
```

Where: x, y, and z are Namelist names.

a, b, c, ... are variable or array names.

The following rules apply to construction of a NAMELIST statement.

1. A Namelist name consists of from 1 through 6 alphanumeric characters, the first of which is alphabetic. The name is enclosed in slashes.

2. A Namelist name may be defined only once by its appearance in a NAMELIST statement, and it must be so defined before its use. After it is defined in the NAMELIST statement, the Namelist name may appear only in input or output statements thereafter in the program.

3. A NAMELIST name cannot be transmitted as an argument from a calling program to a subprogram. In a subprogram, dummy arguments cannot appear in a Namelist list.

4. The list of variable and array names belonging to a Namelist name ends with a new Namelist name (enclosed in slashes) or with the end of the NAMELIST statement.

5. A variable or array name may belong to one or more Namelist names. For example:

   ```
   DIMENSIØN A(10), I(5), L(6)
   NAMELIST  /NAM1/ A, D, I, K, L
       /NAM2/ A, C, K, M
   ```

   means that the arrays A, I, and L, and the variables D and K are included in the list for NAM1, and the array A and the variables C, K, and M are associated with the Namelist name, NAM2.

6. The specification statement(s) defining the type, size, and relative storage locations of arrays and variables must precede any NAMELIST statement in which those arrays and variables are mentioned.

INPUT DATA

The input data must be in a special form in order to be read with a Namelist read statement. Following are the published IBM specifications for the data:

1. The first character of each input record (card image) must be blank.
2. The second character in the first record (of a group of records) must be an &.
3. The Namelist name must start in column 3 of the first record of the group.
4. The Namelist name must be followed by a blank; it must not contain embedded blanks.
5. Input data items follow the Namelist name and are separated by commas. There must not be embedded blanks in the variable names, array names, or constants.
6. Input data items can be continued on succeeding records. A comma after the last item of data on a record is optional. Each succeeding record must begin with a complete variable name, array name, or constant (not with an equals sign).
7. The end of a group of data items is signaled by &END, anywhere in a record except in the first character position.
8. Constants in the data items may take any of the following forms:

   a. Integer

   b. Real

   c. Complex

   d. Logical

   e. Literal (input only)

   Constants are written exactly as in a source program, except that logical constants may be written in the form T or F as well as .TRUE. or .FALSE..

9. The form of the data items may be:

   Variable name = single constant

   The variable name must be one of the names in the Namelist list. It may be a single variable name or it may be subscripted.

   Array name = set of constants

   The array name must be one of the names in the Namelist list.

   The set of constants may be separated by commas, or may be in the form "k*constant" where k is an unsigned integer used to represent k constants.

   If the array name is not subscripted, the first of the set of constants will be stored in the first element of the array, with subsequent constants stored in consecutive elements.

   The number of constants must be less than or equal to the number of elements in the array.

10. If an item that appears in a Namelist list also appears in an EQUIVALENCE statement, only the name from the list may appear on an input record.

The following items are comments on the current IBM Namelist implementation (Release 15/16). As such, they are subject to change in future releases.

1. The first character of any input record is ignored.
2. Embedded and trailing blanks in a constant or exponent are treated as zeros. For example:

Input Data Item	Is Interpreted As
I = 4⊙,	I = 40,
J = 4o3,	J = 403,
E = 1.0E6⊙,	E = 1.0E60,
K = ,	K = 0,

3. Hexadecimal constants, of the form Znnnnnnnn, are accepted. (On output, they will be assigned the type of the name in the list.)
4. Literal constants, of the form 'aaaa', are accepted. (On output, they will be assigned the type of the name in the list.)

   When using Namelist to read a literal constant into an array, all items in the array following the literal constant will be set to EBCDIC blanks. Thus, if there are three words in the array ARR, the input data item

   ARR(1) = 'ABCD'

   will store the characters ABCD in the first word and will fill the next two words with blanks. The input data item

   ARR(1) = 'ABCDEFGHIJKL'

   will fill all three words in the array with the characters shown.

5. The following form is permitted:

   Subscripted array name = set of constants

   When the array name is subscripted, the first item of the set of constants will be placed in the element indicated, with subsequent constants in consecutive elements.

6. The form

   Array name = single constant

   is interpreted to mean that the single constant is to be put in the first element of the array.

7. A repetition factor can be inserted anywhere in the list. Assume that the array TERSE has a dimension of 10. The input data items to fill this array could be:

   ```
   TERSE = 4.0, 8*0.0, 6.0
   ```

   The first and last elements of the array will be set to 4.0 and 6.0, respectively; the intervening elements will be set to 0.0.

8. If the size of an array is exceeded, the error messages appear irrelevant. Depending on the context, the following might be printed:

   NAME NOT IN NAMELIST DICTIONARY

   or

   NAME LARGER THAN 8 CHARACTERS

9. If a real number is set equal to an integer, the decimal point is assumed to be at the end of the number.

Input Data Item	Is Interpreted As
REAL = 1234,	REAL = 1234.0,

10. If an integer is set equal to a real number, the decimal point is ignored.

Input Data Item	Is Interpreted As
INTGR = 2.345,	INTGR = 2345,

11. If an item is repeated in the group of Namelist input records, the last one read is used.

12. Since the last item on a card must be a data name or a constant, an identification field may not be punched in columns 73-80.

13. If EEH facilities (Region 226) are available, a CALL ERRSET statement will permit continuation of the program after Namelist errors are encountered. For example, the statement

    ```
    CALL ERRSET (221, 20, 20, 0, 1, 224)
    ```

    will permit up to 19 occurrences each of errors 221 through 224; the program will be terminated on the 20th error for any one of these error numbers. A message showing the card in error will be printed.

    Note that the standard fixup ignores all remaining items in the group of input records. Not only the erroneous item, but all the following items in the group are ignored.

INPUT PROCEDURE

When a READ statement references a Namelist name, the input of data is begun. The first data card (or record) is read and examined to verify that its name is consistent with the Namelist name in the READ statement.

If the specified Namelist name is not found, additional records are examined consecutively until there is a successful match, or until all the data for the program are exhausted. (This provides a method for skipping undesired blocks of data in some branches of the program.)

Reading continues until an &END is encountered. Any information following the &END is ignored.

OUTPUT

When the arrangement of the printed page is not significant, Namelist relieves the programmer of the effort of setting up formats.

When a WRITE statement references a Namelist name:

1. All variables and arrays, and their values, belonging to the Namelist name will be written, each according to its type. Arrays are written in columnwise order.

2. The output data will be written such that:

   a. The field for the data will be large enough to contain all significant digits.

   b. The output can be read by an input statement referencing the Namelist name.

The PUNCH statement cannot be used with Namelist. However, a WRITE (14, nname) statement, where nname is a Namelist name, can be employed to produce punched output. A suitable DD statement is also required; one is provided in AFSLINK.

EXAMPLE

Assume that the program contains the following statements:

```
COMPLEX   CØMPLX (5)
REAL*8    DBLE (5)
REAL*4    REAL (10)
INTEGER   INTGR (10)
LØGICAL   LØGCL (20)

NAMELIST /INPUT/ REAL, INTGR,
    CØMPLX, DBLE, LØGCL
        :
        :
READ (5, INPUT)
        :
        :
```

Also assume the four data cards in the right column. (For the example, the input is written in the same order as the Namelist list. This is for convenience comparing the input to the output; the data need not be written in that order.)

The first record is read from the device associated with data set reference number 5 (normally the input stream). The record is searched for an & in column 2, immediately followed by the Namelist name, INPUT, and the required blank after the name.

Since the search is successful, the data items are converted and placed in core. If desired, only one or two of the variable names could have appeared in the input records.

The ten real constants will be placed in REAL(1) through REAL(10). The constant 2 will be placed in INTGR(1), the constant 6 will be stored in INTGR(2) through INTGR(9), and the constant 10 will be stored in INTGR(10).

The arrays CØMPLX and DBLE will be similarly filled as indicated. The first two items of the array LØGCL will be set to .TRUE.; the remainder of that array will be set to .FALSE..

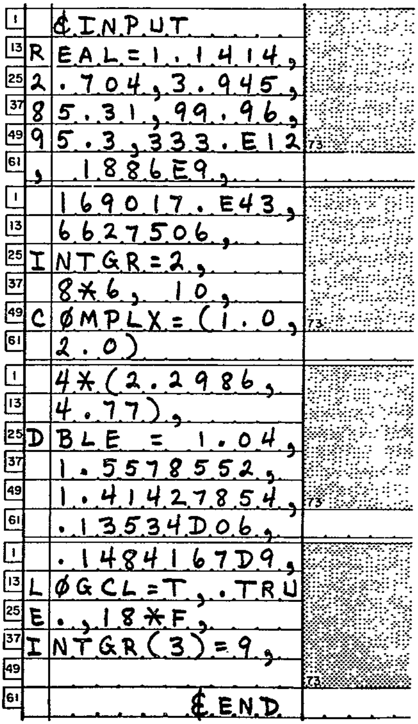
INTGR(3), previously set to 6, wil be reset to 9 by the last data item.

The &END signals termination of the input for this READ statement.

The example on the following page shows output produced by the statement:

```
WRITE (6, INPUT)
```

For the illustration, the output data are derived from the input shown below.



```
 &INPUT
REAL=1.1414,
2.704,3.945,
85.31,99.96,
95.3,333.E12
, 1886E9,
 169017.E43,
 6627506,
INTGR=2,
 8*6, 10,
CØMPLX=(1.0,
 2.0)
 4*(2.2986,
 4.77),
DBLE = 1.04,
 1.5578552,
 1.41427854,
 .13534D06,
 .1484167D9,
LØGCL=T,.TRU
E.,18*F,
INTGR(3)=9,

        &END
```

EXAMPLE OF NAMELIST OUTPUT

```
&INPUT
REAL=  1.1413994    ,  2.7039995    ,  3.9449997    ,  85.309998    ,  99.959991    ,
  95.299988    , 0.33299982E 15, 0.18859992E 13, 0.16901693E 49,  6627506.0    ,
INTGR=          2,          6,          9,          6,          6,          6,
        6,          6,          6,          10,COMPLX=
          (1.0000000,2.0000000),               (2.2985992,4.7699995),
          (2.2985992,4.7699995),               (2.2985992,4.7699995),
          (2.2985992,4.7699995),DBLE=  1.040000000000000    ,
 1.557855200000000    ,  1.414278540000000    ,  135340.0000000000    ,
 148416700.000000    ,LOGCL=T,T,F,F,F,F,F,F,F,F,F,F,F,F,F,F,F,F,F,F
&END
```

Standard Namelist printing was modified by a job control statement to reduce the width of the output to fit on this page. The JCL statement used was:

```
//G.FT06F001    DD  SYSØUT=A,DCB=(RECFM=VBA,LRECL=90,BLKSIZE=990)
```

If this technique is employed to reduce the width of the output, all records written on the data set defined by data set reference number 6 must fit within the reduced ~~record~~ length.

The maximum CTC block size for an ASP system is 1012.

Non-Linear Bearing Force Characteristic Representation

Provision is made in the program to simulate non-linear bearing stiffness characteristics; however, linear damping is used in the program. The non-linear bearing stiffness characteristics including simultaneously linear speed sensitive stiffness parameters may be represented with a mathematical formulation as follows: (To **eliminate translation of** parametric notations from that used in the mathematical formulation to that used in the computer program, the latter notations are used here).

$$\left[(\mathrm{FD\O T{-}FD\O FIX}) \left(\mathrm{BNB} + \mathrm{BBB}\,(\rho - \mathrm{BR\O B}) \right) + \mathrm{BKB} \right] (\rho^{\mathrm{BHB}} + (\mathrm{BDB})\rho + \mathrm{BEB}) = \mathrm{F}$$

where ρ is the resultant journal displacement from its zero-load position and F is the corresponding resultant bearing force.

BNB used in the computer program was denoted in the mathematical formulation portion of this report by N_{Bik}, and similarly

BBB by B_{Bik}

BRØB by ρ_{Bik}

BKB by K'_{Bik}

BHB by H_{Bik}

BDB by D_{Bik}

BEB by E_{Bik}

FDØT by $\dot{\phi}$

FDØFIX by $\dot{\phi}_o$

"FDØFIX" is a non-subscripted name in the current program. It could be extended to a subscripted array in the future extension of the computer program. To simplify the expression of the above equation, the subscripts i, k are omitted. The constants FDOFIX, BNB, BBB, etc., may be considered as that for a particular non-linear stiffness section of a bearing station. To briefly describe the method of determining the constants an arbitrary non-linear bearing stiffness characteristic is used as shown in Fig. 8.

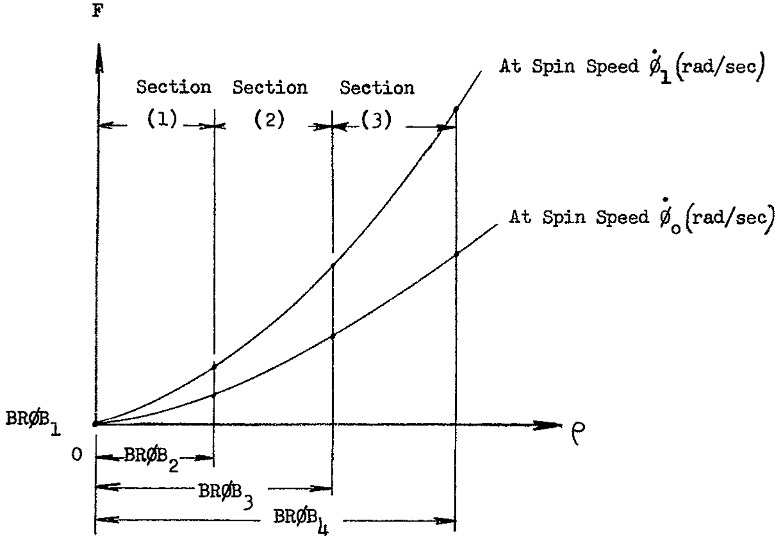


Fig. 8. Bearing Stiffness Curves

Assume the non-linear bearing stiffness section (2) is to be simulated. First use the non-linear stiffness characteristic at spin speed $\dot{\phi}_o$. Determine BHB according to the curvature, then solve for the three unknowns from the three equations,

$$BKB\left[\rho_i^{BHB} + (BDB)\,\rho_i + BEB\right] = F_i \qquad i = 1, 2, 3$$

corresponding to three selected points on the characteristic curve.

The reason for using BKB as a common factor is that BKB may be directly used as a linear stiffness factor for a linear bearing when BHB = 1, and BDB = BEB = 0. Having determined the values of BKB, BHB, BDB and BEB, the values of BNB and BBB can be readily computed from selected two ρ values on the upper curve at spin speed $\dot{\phi}_1$ in Fig. 8. The equations used to obtain the solutions are:

$$\left\{\left(\dot{\phi}_1 - \dot{\phi}_0\right)\left[\mathrm{BNB} + \mathrm{BBB}\left(\rho_1 - \mathrm{BR0B}_2\right)\right] + \mathrm{BKB}\right\}\left[\rho_1^{\mathrm{BHB}} + (\mathrm{BDB})\ \rho_1 + \mathrm{BEB}\right] = F_1$$

$$\left\{\left(\dot{\phi}_1 - \dot{\phi}_0\right)\left[\mathrm{BNB} + \mathrm{BBB}\left(\rho_2 - \mathrm{BR0B}_2\right)\right] + \mathrm{BKB}\right\}\left[\rho_2^{\mathrm{BHB}} + (\mathrm{BDB})\ \rho_2 + \mathrm{BEB}\right] = F_2$$

In a similar manner the non-linear bearing stiffness representation of all other sections may be made.

Output Format

To provide a permanent record, the input data are printed out prior to the computation results.

The results of computations are printed with CRT graphs as optional additional output.

1. Print-Out. The computation results are printed out in the following sequence:
    - a. Total rotor weight, lb
    - b. Total rotor mass, the sum of additional rotor mass (AM) and the rotor mass computed from the rotor geometry $(\text{lb} - \text{sec}^2)/\text{in}$
    - c. Total rotor polar mass moment of inertia, the sum of additional rotor polar mass moment of inertia (AIRØ) and the polar mass moment of inertia computed from the rotor geometry.
    - d. Force and moment influence coefficients
    - e. Initial rotor motion conditions
    - f. Elements in rotating coordinates consisting of:
        - 1) NS number of rotor displacement vector length, in
        - 2) NS number of rotor whirl displacements, rad
        - 3) NS number of rotor displacement vector velocities, in/sec
        - 4) NS number of rotor whirl velocities, rad/sec
        - 5) Rotor spin displacement rad
        - 6) Rotor spin velocity rad/sec
    - g. The appropriate derivatives consisting of:
        - 1) NS number of rotor displacement vector velocities in/sec
        - 2) NS number of rotor whirl velocities rad/sec
        - 3) NS number of rotor displacement vector radial accelerations in/sec^2

4)	NS number of rotor whirl accelerations	rad/sec^2
5)	Rotor spin velocity	rad/sec
6)	Rotor spin acceleration	rad/sec^2

h. X displacement array in stationary coordinates, (in.)

i. Y displacement array in stationary coordinates, (in.)

j. Rotor deflection vector array, in.

k. Phase angle array for rotor deflection vectors from x-axis in the direction of rotation, degrees

l. $\dot{X}$ velocity array in stationary coordinates, in.

m. Y velocity array in stationary coordinates, in.

n. Whirl frequency array, rpm

o. Whirl-to-spin frequency ratio array

p. Total number of revolutions

q. Spin speed, rpm

r. X bearing displacement coordinate, in.

s. Y bearing displacement coordinate, in.

t. X bearing velocity coordinate, in/sec.

u. Y bearing velocity coordinate, in/sec.

v. Journal displacement vector from bearing-center array, in.

w. Journal displacement vector phase angle in the direction of rotation from X-axis, degrees

x. Bearing reaction array, lb.

y. Bearing reaction vector phase angle in the direction of rotation from x-axis, degrees

z. Bearing force to journal deflection ratio, or equivalent linear bearing stiffness, lb/in/bearing.

A sample of printout of the computation results including that of input data is attached in Appendix A.

2. <u>CRT Plotting Output</u>. To facilitate the direct comprehension of the computation results and appreciation of the trend of rotordynamic

behavior, a series of 7 types of CRT are provided as an optional output controlled by an input word, "CRT." That is, CRT=1 for CRT output and CRT=0 for no CRT output.

The 7 types of CRT graphs are described as follows:

a. Rotor 3-dimensional mode shape at or near preselected input speeds, "INPRPM," for a rotor operating with spin-speed acceleration or deceleration. For a steady-state rotor spin speed, one CRT of this type is provided during the run. The spin speed, at which the mode shape is depicted, is labeled on top of the graph. Three-dimensional information of the graph is provided by labeling the phase angle at corresponding displacement vector stations.

b. Rotor spin speed versus time graph.

c. Rotor whirl-to-spin speed ratio versus spin speed graph for a pre selected rotor station as designated by the input, "IASIGN."

d. Bearing forces versus rotor spin speed graphs for all non-linear stiffness bearings.

e. Bearing displacements versus spin speed graphs for all non-linear stiffness bearings.

f. Maximum rotor displacement versus spin speed. The rotor station number for the maximum rotor deflection station for a spin speed is appropriately labeled for identification

g. Rotor displacement at station "IASIGN" versus spin speed.

The number of points included in a frame of the CRT graph may vary from 1 through 50 as specified by the value of the input word, "NPØINT."

EXPERIMENTAL PROGRAM

The purpose of Task II, Mark 25 Rotor Dynamics Tests, was to perform spin tests utilizing the Mark 25 pump rotor to obtain data for correlation with the results of the rotordynamics computer program.

The test program included an initial series of seven tests to minimize rotor unbalance, plus eight investigative tests on the Mark 25 pump rotor with an array of unbalance levels and locations. Instrumentation was provided to measure response amplitudes and phase angles.

MARK 25 TURBOPUMP

The Mark 25 pump is an axial-flow liquid hydrogen pump consisting of an inducer and four axial stages. The pump rotor is mounted by duplex paired bearings. The bearings used are 55-millimeter bore size. To maintain radial stiffness required for rotordynamic considerations (critical speed, rotor whirl, and minimum tip clearance without rubbing), the bearings are arranged in axially preloaded pairs, each pair free to move independently of the other.

The rotor is built-up rotor of eight disks clamped together with 18, (5/16) through bolts. The overhung inducer is mounted by 6, (3/8) bolts to the forward disk stub shaft. The rotor is driven through a ball spline coupling at the aft stub shaft. The overall rotor length is approximately 24 inches with a nominal diameter of 6.1 inches.

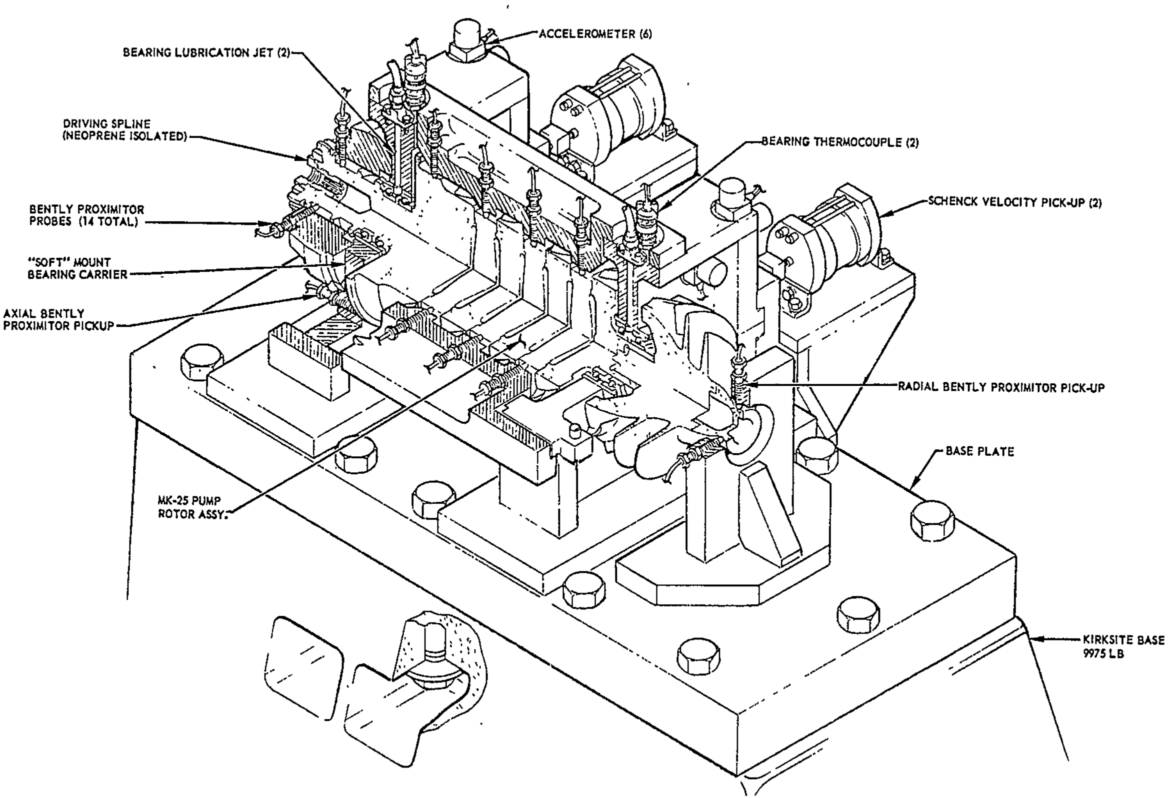


Figure 9. Mark-25 Rotor High Speed Test Set-Up

TEST SET-UP

Hardware

The test set-up consisted of a Mark 25 pump rotor mounted in a support pedestal which was bolted to a large seismic mass. The complete assembly was mounted inside a large vacuum chamber. The pump bearings were cooled by a Freon 21 system. Rotating speeds up to 60,000 rpm were provided by a prime mover (300 hp dynamometer) and a 10:1 gear ratio speed increaser. The rotor was driven by means of an aluminum quill shaft and a splined adapter with a neoprene isolator to minimize transmission of gear noises.

Instrumentation

The basic instrumentation consisted of the following:

1 channel,	rotor speed
14 channels,	rotor displacements
2 channels,	pedestal vibration
2 channels,	bearing temperatures

The primary instrumentation parameters were the rotor displacement measurements which were made with Bently proximity transducers. A typical installation of these transducers is shown in Fig. 9.

Two mounting bars for the displacement transducers were located along the rotor axis in two 90 degree planes and bolted to the bearing pedestals. The fourteen non-contacting displacement indicators were used to monitor the radial deflection of the rotor in six radial planes and the axial position of the rotor. A calibration spot face was used to determine actual displacement values during test runs.

TEST PROGRAM

The test program was accomplished in two parts. The first part was a set of spin tests to establish an initial balancing condition and the second part was to perform spin tests to investigate unbalance conditions. These unbalance conditions would produce response data for correlation with the mathematical computer model results.

A total of 16 tests were performed as part of this contract study. The first three tests were performed without the inducer. The next three were performed with the inducer. The remaining 10 tests were investigative. One of the investigative tests, 1140, was not used because of loss of data due to a tape recorder malfunction.

Balancing Tests

Three tests (1130 through 1132) were performed without the inducer to obtain an initial balance on the basic rotor. The inducer was then installed and test 1133 performed. During test 1133, the maximum displacement at the forward (inducer) end was .0174 inches at 26,000 rpm. A Post-test examination revealed an inadequate inducer pilot depth. The pilot depth was increased and tests 1134 and 1135 were performed. The maximum displacement for test 1134 was .0113 at 28,000 rpm, but was only .0051 at 26,000 rpm which indicated a significant balance improvement over test 1133. Figure 10 presents the inducer forward displacement for tests 1133 and 1134. Test 1135 was an attempt to achieve an improved balance but was terminated at 24,000 rpm due to excessive deflections.

Investigative Tests

The eight investigative tests, 1136 through 1139 inclusive, and 1142

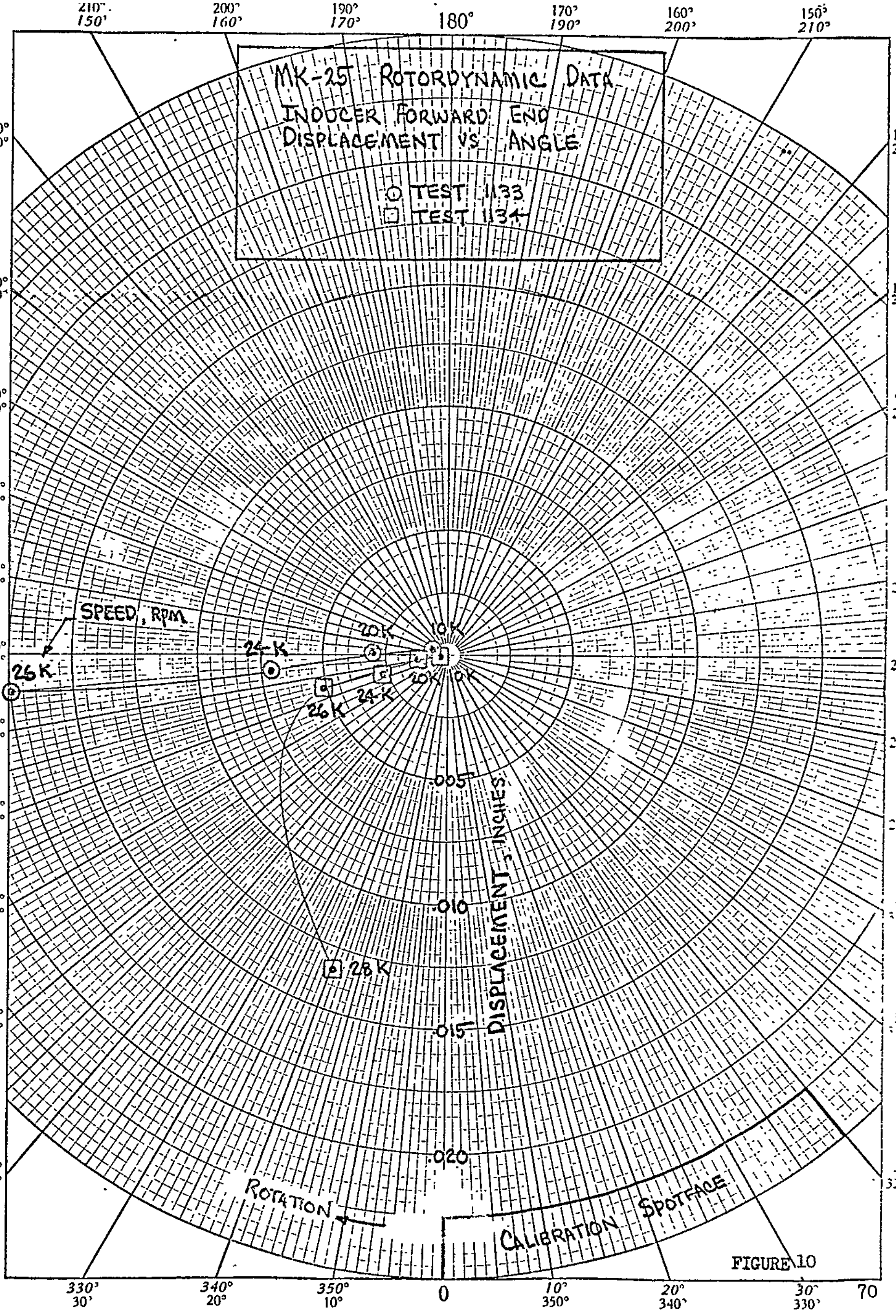

MK-25 ROTORDYNAMIC DATA
INDUCER FORWARD END
DISPLACEMENT VS ANGLE
○ TEST 1133
□ TEST 1134
SPEED, RPM
26 K
24 K
20 K
10 K
20 K
10 K
24 K
26 K
28 K
.005
.010
.015
.020
DISPLACEMENT, INCHES
ROTATION
CALIBRATION SPOTFACE
180°
0
210° 150°
200° 160°
190° 170°
170° 190°
160° 200°
150° 210°
220° 140°
230° 130°
240° 120°
250° 110°
260° 100°
270° 90°
280° 80°
290° 70°
300° 60°
310° 50°
320° 40°
330° 30°
340° 20°
350° 10°
10° 350°
20° 340°
30° 330°

FIGURE 10

through 1145 were performed with various unbalance conditions. Test 1141 was selected as the reference run since it was the best balance achieved. Response deflections of the rotor for the other tests are given as vector changes from the deflections observed in 1141. The forward inducer end and the coupling end displacements for test 1141 are shown in Figure 11. In order to include the effect of critical speed only the tests that operated up to 30,000 rpm or greater were used to generate investigative data. The critical speed was calculated to be approximately 28,000 rpm.

Table 3 lists the test run, the unbalance condition and the upper speed limit achieved. The unbalance weights were selected to allow operation to at least 30,000 rpm without creating excessive bearing loads or allowing a deterioration problem to develop.

PROCEDURE FOR TEST SET-UP

Prior to installation in the rotordynamics test facility, the pump rotor assembly was dynamically balanced at low speed (1000 rpm) on the Gisholt balancing machine. The balancing sequence was as follows:

1. All individual rotors and rotor spacers were dynamic balanced on an arbor.
2. The inducer was balanced on an arbor.
3. The rotors and rotor spacers were assembled into a composite rotor assembly by means of thru bolts, and this assembly mounted in balance bearings was dynamic balanced on the Gisholt machine.
4. The balance bearings were removed, and the rotor was assembled to a pump configuration with duplex bearings, special bearing cartridges, and all rotating parts except the inducer, spinner, and mounting studs and fasteners. Assembled in this manner the rotor was again dynamic balanced. The inducer was then installed on the rotor and this assembly balanced. The spinner was then added and balance corrections made. All components of the rotor were matchmarked.

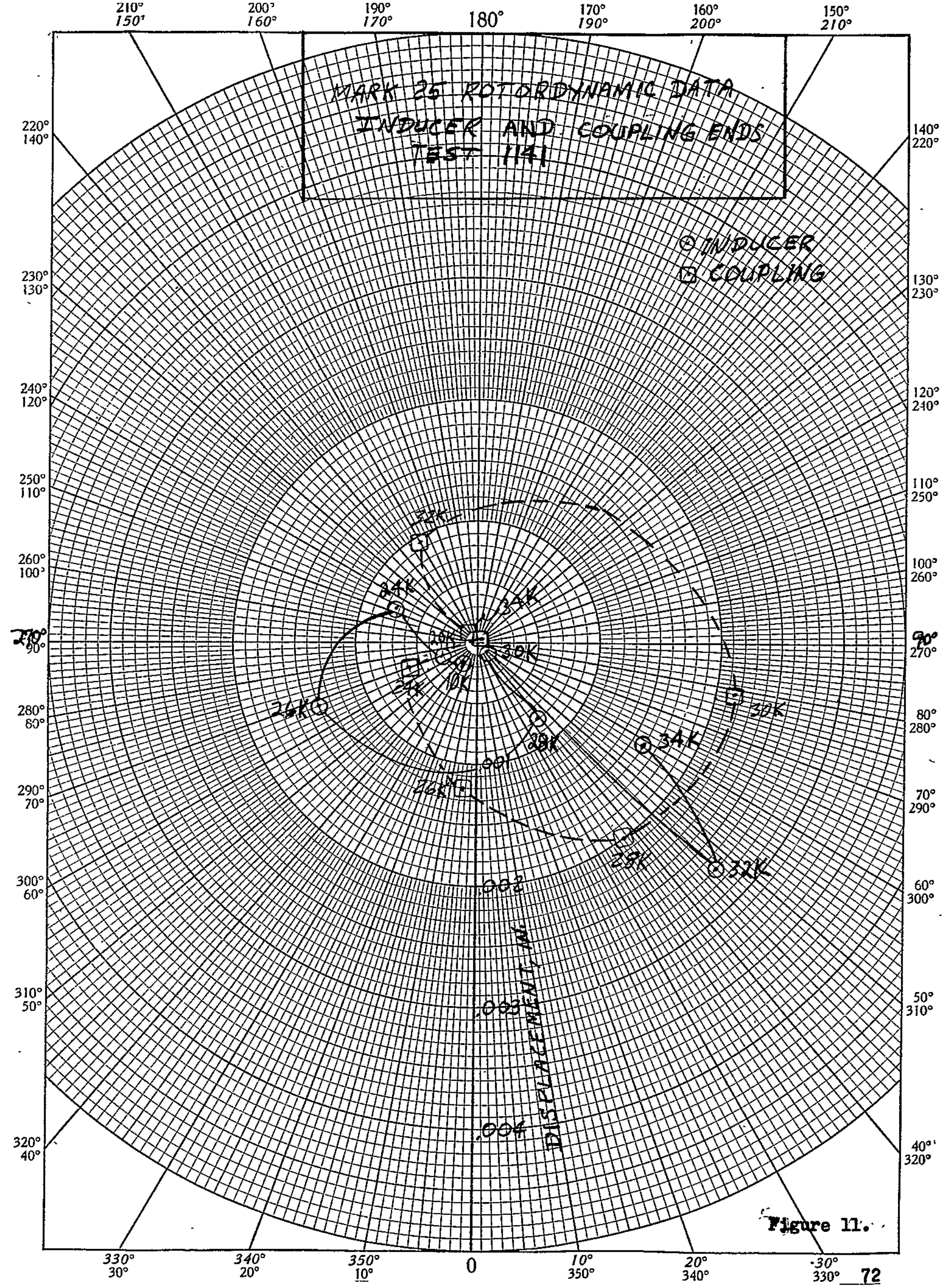

MARK 25 ROTORDYNAMIC DATA
INDUCER AND COUPLING ENDS
TEST 1141
⊙ INDUCER
☐ COUPLING
DISPLACEMENT, IN.
.001
.002
.003
.004
180°
0
210° 150°
200° 160°
190° 170°
170° 190°
160° 200°
150° 210°
220° 140°
140° 220°
230° 130°
130° 230°
240° 120°
120° 240°
250° 110°
110° 250°
260° 100°
100° 260°
270° 90°
90° 270°
280° 80°
80° 280°
290° 70°
70° 290°
300° 60°
60° 300°
310° 50°
50° 310°
320° 40°
40° 320°
330° 30°
340° 20°
350° 10°
10° 350°
20° 340°
30° 330°
32K
24K
34K
26K
30K
10K
26K
28K
34K
30K
26K
28K
32K

Figure 11.

TABLE 3

UNBALANCE AND UPPER SPEEDS
FOR EXPERIMENTAL TESTS

Test No.	Unbalance *		Location		Speed rpm
	Amount Gram-In	Angle Degrees	Plane No. Shown in Fig. 12	Inches From Inducer	
1136	2.12 3.40	138 139	2 3	1.8 5.2	30,000
1137	2.74 4.25	195 196	2 3	1.8 5.2	30,000
1138	0.26 0.365	-24 -24	2 3	1.8 5.2	34,000
1139	0.64 1.10	-32 -30	2 3	1.8 5.2	34,000
1142	1.47	121	10	12.9	32,000
1143	.118	165	9	24.0	30,000
1144	.51	125	7	19.5	30,000
1145	.52	115	4	7.8	30,000

* Reference to Test 1141 and machined spot face on the coupler.

5. The inducer, spinner, and mounting studs were removed from the rotor assembly and the rotor minus the above parts was installed in the cradle mount in the rotordynamics test facility.

6. After the rotor had been installed, all Bently proximitor probes were gapped, rotor runouts were taken and recorded, and Freon lube jets, bearing thermocouples, and accelerometers were installed.

7. After the rotor was initially balanced, the inducer and spinner were installed so that the complete rotor assembly could be balanced.

ROTORDYNAMICS SPIN TEST PROCEDURE

Prior to the first test each day, the speed increasing gear box was heated to operating temperature by recirculating lube oil through heaters by electrically driven pumps in order to ensure vertical alignment of the center lines on gear box pinion shaft and test rotor. Also, all electronic equipment, oscilloscopes, tape recorders, etc. were turned on to warm up and be checked out. After the above was established, the sequence of events was as follows:

1. The lube oil pumps and heaters were turned off, and the lube oil run tank was charged.

2. The test chamber was closed and the vacuum pumps were started.

3. The Freon pressurized lube system for pump rotor bearings was charged.

4. All on-line monitoring instruments were set up and calibrated and the appropriate test parameter calibrations were recorded on tape.

5. All other systems and controls in the control room and test cell were made ready for test.

6. When chamber pressure reached approximately 40 mm Hg. pressure, the Freon-21 bearing lube system was turned on while rotating the pump rotor slowly to chill the bearings, rotor and cradle to a stabilized operating temperature.

7. When the pump rotor had been chilled, the Freon flow was stopped, bearing purge was turned on, and all Bently dectector potentiometers were set for proper voltage output.

8. The test area was cleared and all stations in the control room were manned. The bearing purges were turned off; lube oil and Freon lube systems were turned on and flows were established.

9. The tape recorders and Brush recorders were turned on.

10. The test was started and rotor speed was manually programmed to follow a speed ramp run schedule with a series of speed plateaus. In addition to recording test parameters, one channel on each tape recorder was used to monitor voice, and rotor speed was announced in approximate intervals of five seconds as well as when speed was being changed. This channel also monitored voice communication between the control room and data acquisition room.

11. During the test, Bently output signals were monitored on four two-channel oscilloscopes.

12. At the completion of a test, vacuum was broken in the test chamber, and the tape played back.

13. Phase angle in degrees and rotor motion in mils for each speed plateau was determined from Polaroid photographs and oscillograph records.

14. The rotor quill shaft was disengaged so that the rotor could be turned, by hand, to check bearings.

15. The appropriate balance corrections or unbalance conditions were made on the rotor assembly.

16. The above procedure was repeated until all required tests were performed.

RESULTS

The test results of the experimental program are given in radial deflection of the rotor at six axial stations along the rotor. The experimental results are presented in graphical CRT plots. There are two types of data presentations for each test:

1. deflections vs speed for a given station, example: Figure 13
2. deflection vs station for a given speed, example: Figure 21

For the first type of plot, the stations selected were stations 2 (Bently 16), the inducer end; station 9 (Bently 8), the center of the rotor; and station 17 (Bently 2), the coupler end of the pump. The second type of plot was selected at or near speeds of maximum deflection at one or more stations. Three speeds are shown for each test. The total maximum deflection and the X and Y plane components are shown relative to the spot face at station 17, which was the zero degree (0^o) reference point on the rotor. Table 4 gives the relationship between the experimental and mathematical station descriptors.

In order to adjust the data to a form easily used for comparative purposes, the data was corrected to remove the effects of shaft runout and the unknown residual unbalance. The latter was accomplished by vectorily subtracting the data from the minimum response run (the reference run, 1141) from all other test runs. The unbalance condition listed in Table 3 is the "delta" unbalance between the test run and the minimum response run (1141). All angles referred to in the Table 3 and the graphical

TABLE 4

RELATIVE MATHEMATICAL AND EXPERIMENTAL MODEL DESCRIPTORS

Bently Meas. Location	Math. Model Station	Rotor Axial Position (Ref. Inducer end) Inches
16	2	0.8
10	8	8.75
8	9	11.5
6	11	14.5
4	13	17.25
2	17	22.9

plots are referenced to a spot face at plane 8 (Bently #2), Fig. 12

On the experimental data plots the following nomenclature is used:

DEFL 2 = deflection of the rotor at Bently #2 (Math Model Sta. 17) vs speed

DEFL 8 = deflection of the rotor at Bently #8 (Math Model Sta. 9) vs speed

DEFL 16 = deflection of the rotor at Bently #16 (Math Model Sta. 2) vs speed

X PLANE = deflection of the rotor vs station at indicated speed in the X-plane with reference to the spot face at Station 17.

Y PLANE = deflection of the rotor vs station at indicated speed in the Y-plane with reference to the spot face at Station 17

TOTAL = total deflection of the rotor (maximum) vs station

All deflections given are in mils (10^{-3} inches).

Figures 13 to 20, inclusive, are the deflection vs speed plots for test runs 1136 through 1139 and 1142 through 1145, inclusive. The speed range plotted is from 10,000 to 30,000 rpm except for Test 1142 which had an upper speed limit of 32,000 rpm, and Tests 1138 and 1139 which achieved upper speed limits of 34,000 rpm.

Figures 21 to 44, inclusive, are the deflection of the rotor vs rotor stations for three speed cases for each of the investigative tests. The abscissa is the distance from the inducer end in inches. In most cases, the applied unbalance tended to cause the local section of the pump rotor to deflect significantly; i.e., when the unbalance was at the

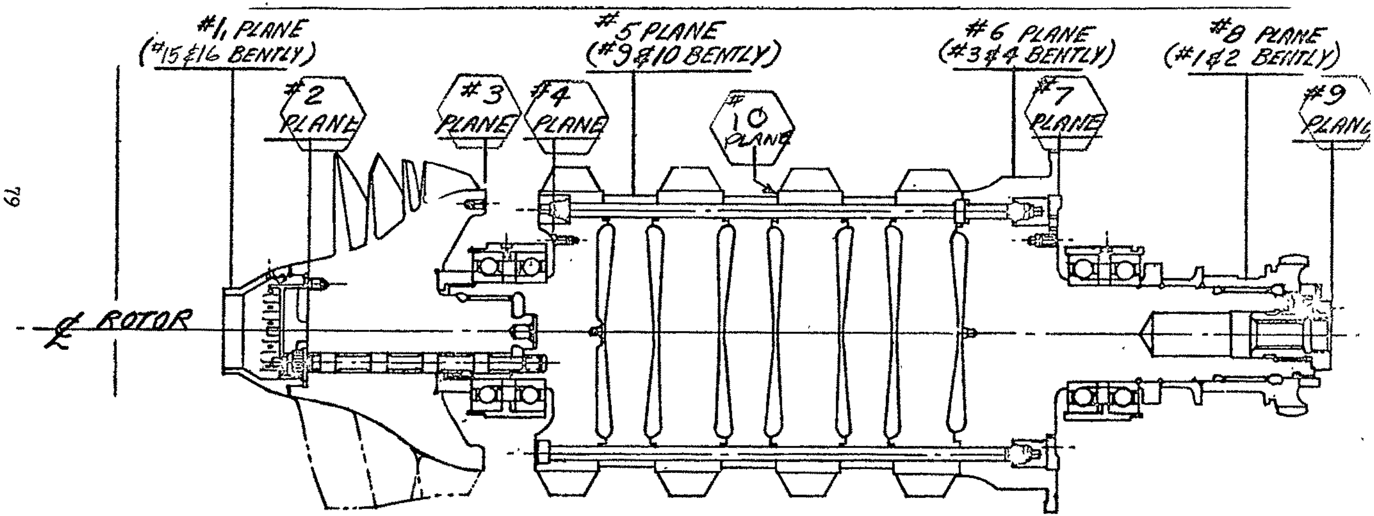


Figure 12. **Bently and Balance Planes, Mark-25 Pump Rotor**

inducer, the inducer end had the largest deflection; when the unbalance was in the center section (between the bearings) the center of the rotor deflected more. However, when the unbalance (a very small amount) was applied at the coupler end, the inducer displayed large deflections. This was primarily due to the inherent mode shape of the rotor in the speed range of interest.

TEST NO. 1136

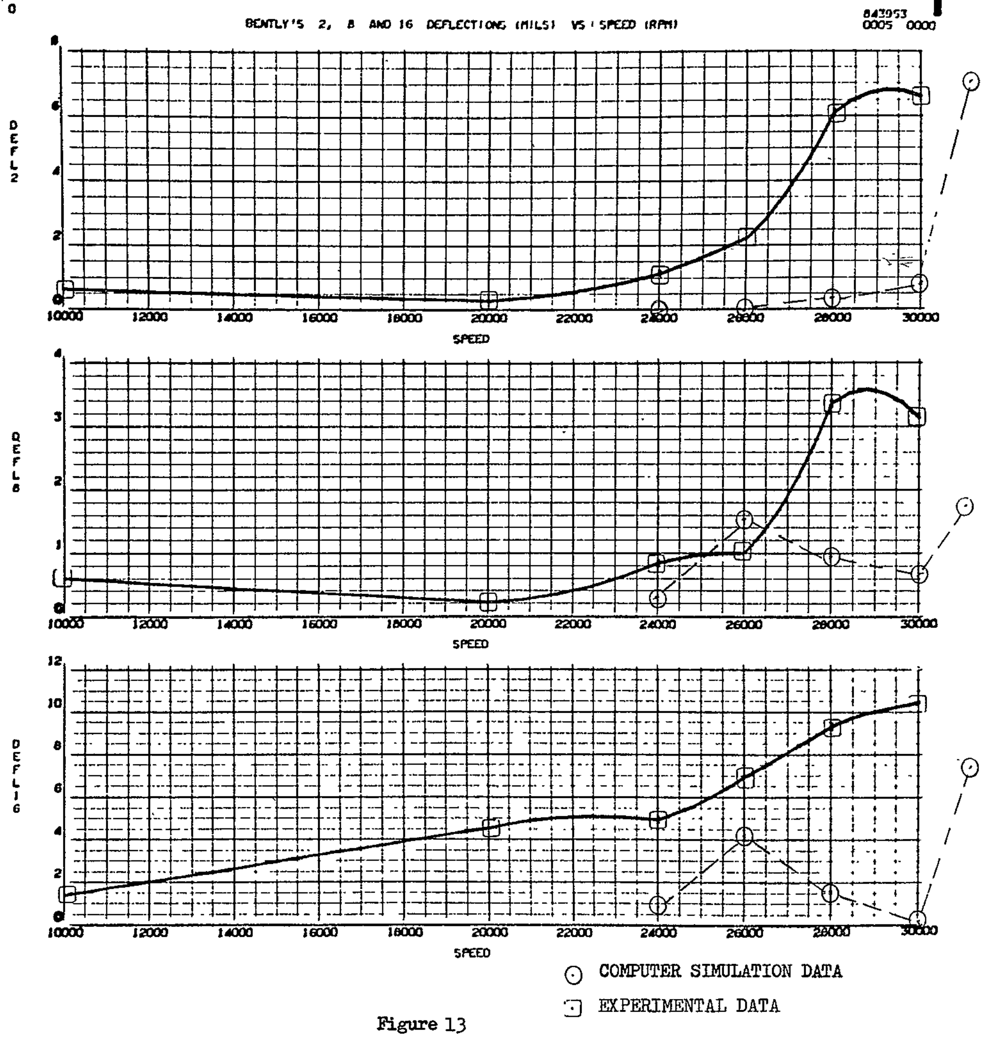


Figure 13

TEST NO. 1137

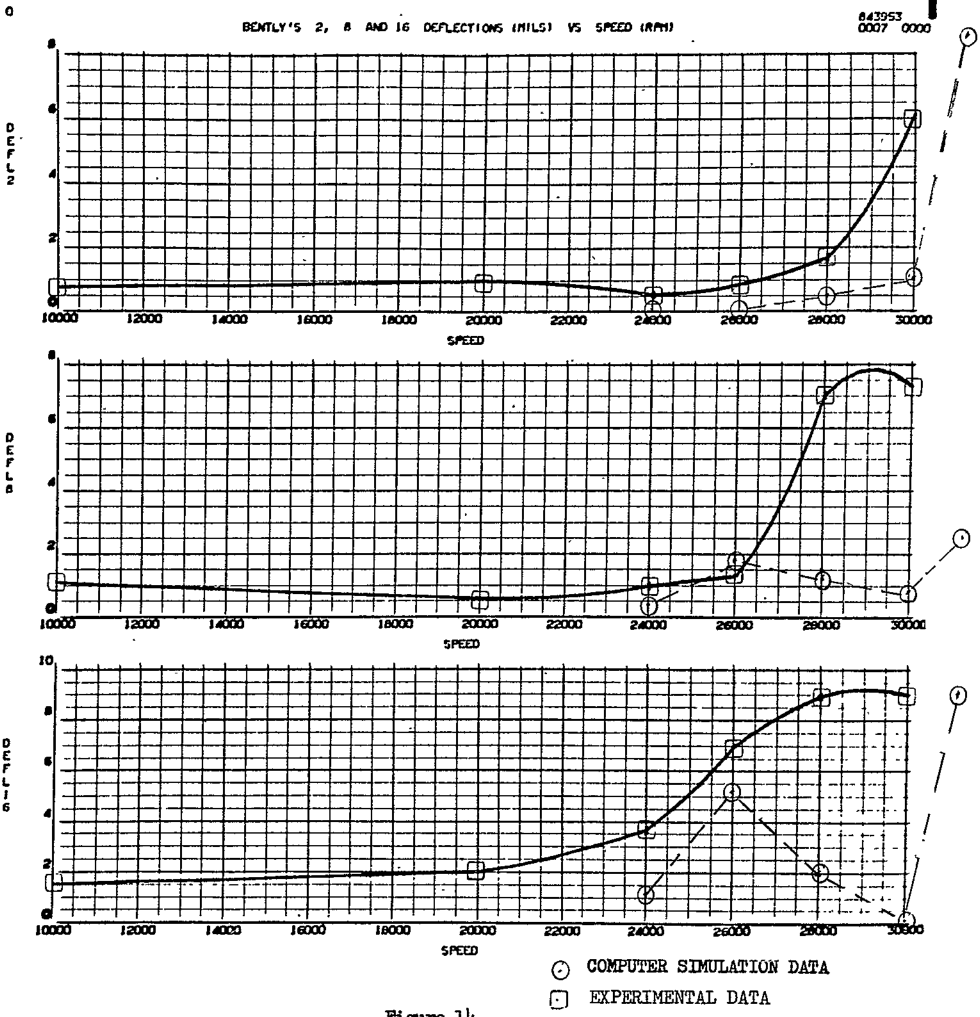


Figure 14

TEST 1138

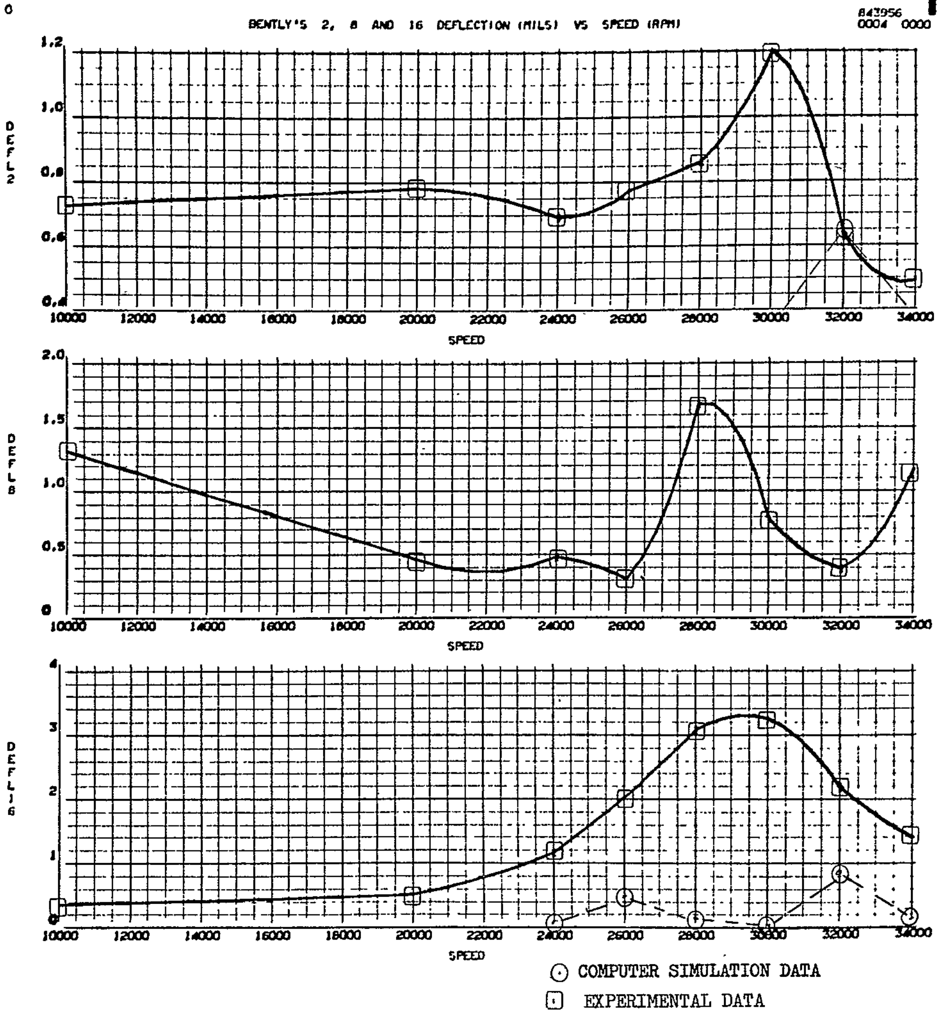


Figure 15

TEST 1139

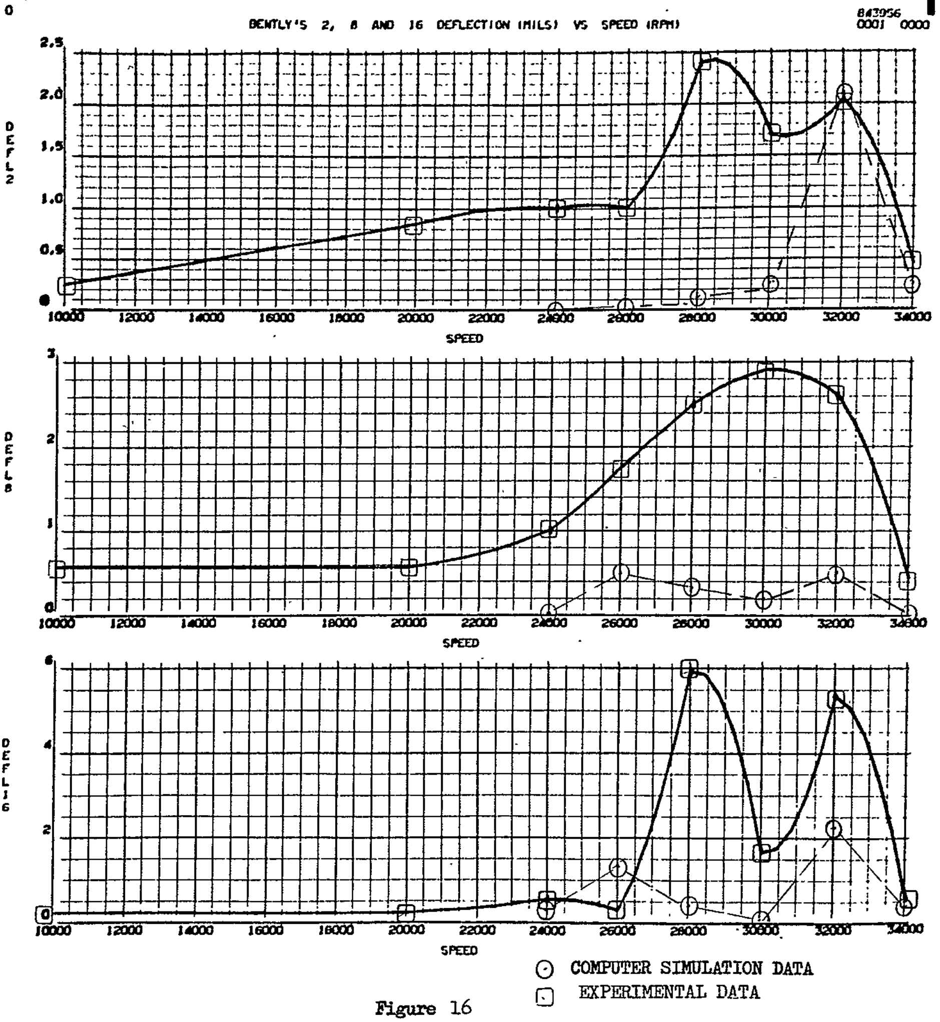


Figure 16

TEST 1142

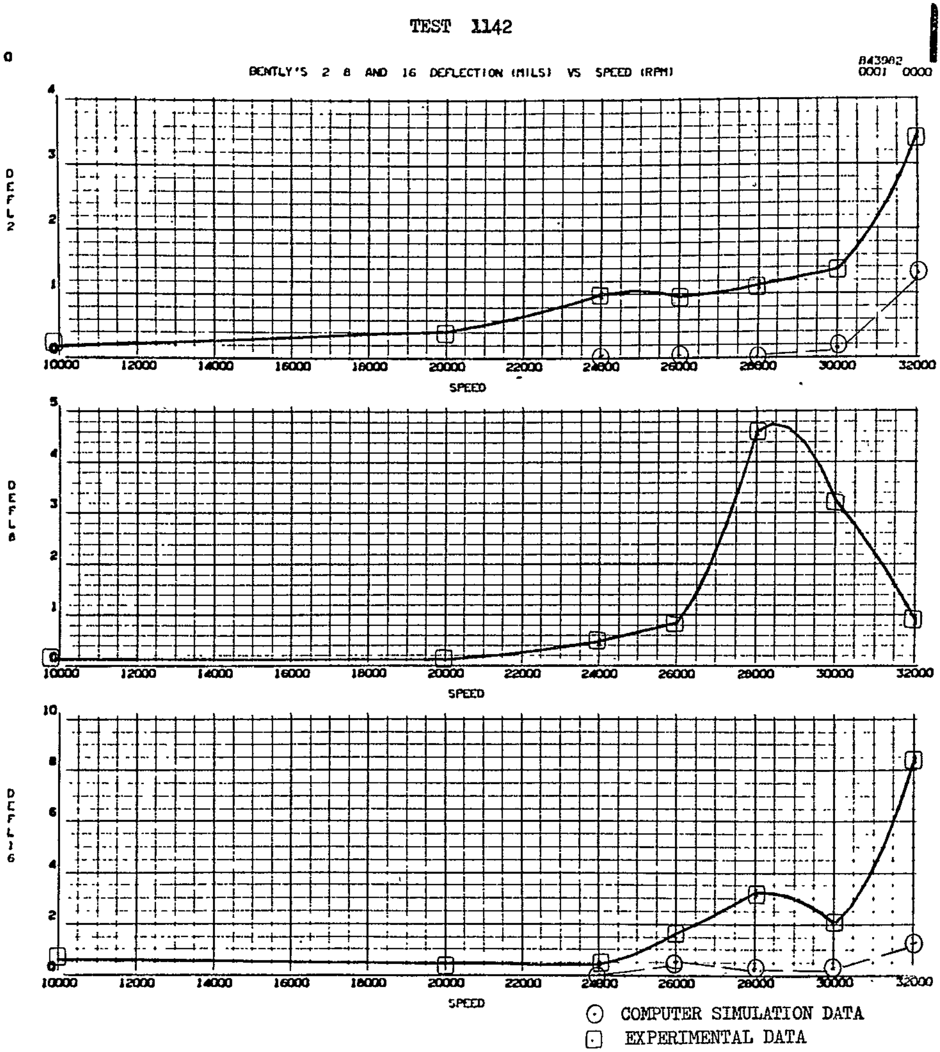


Figure 17

TEST NO. 1143

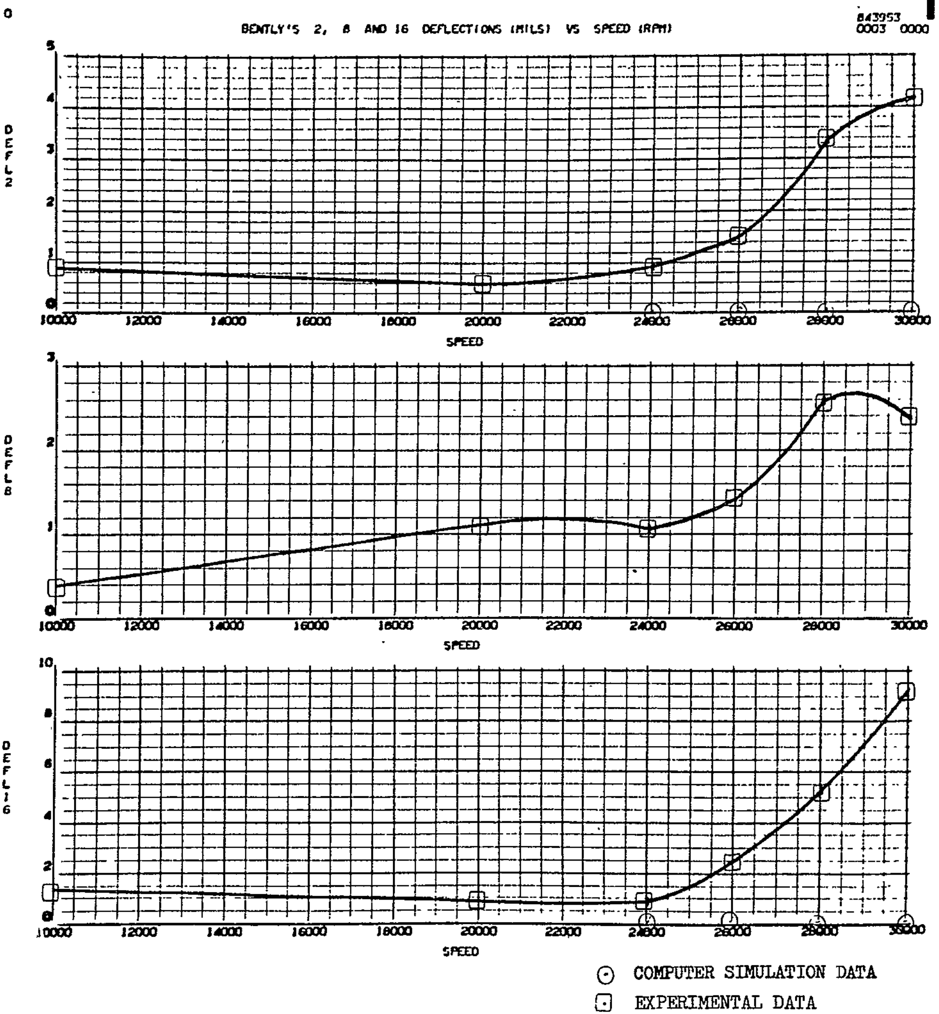
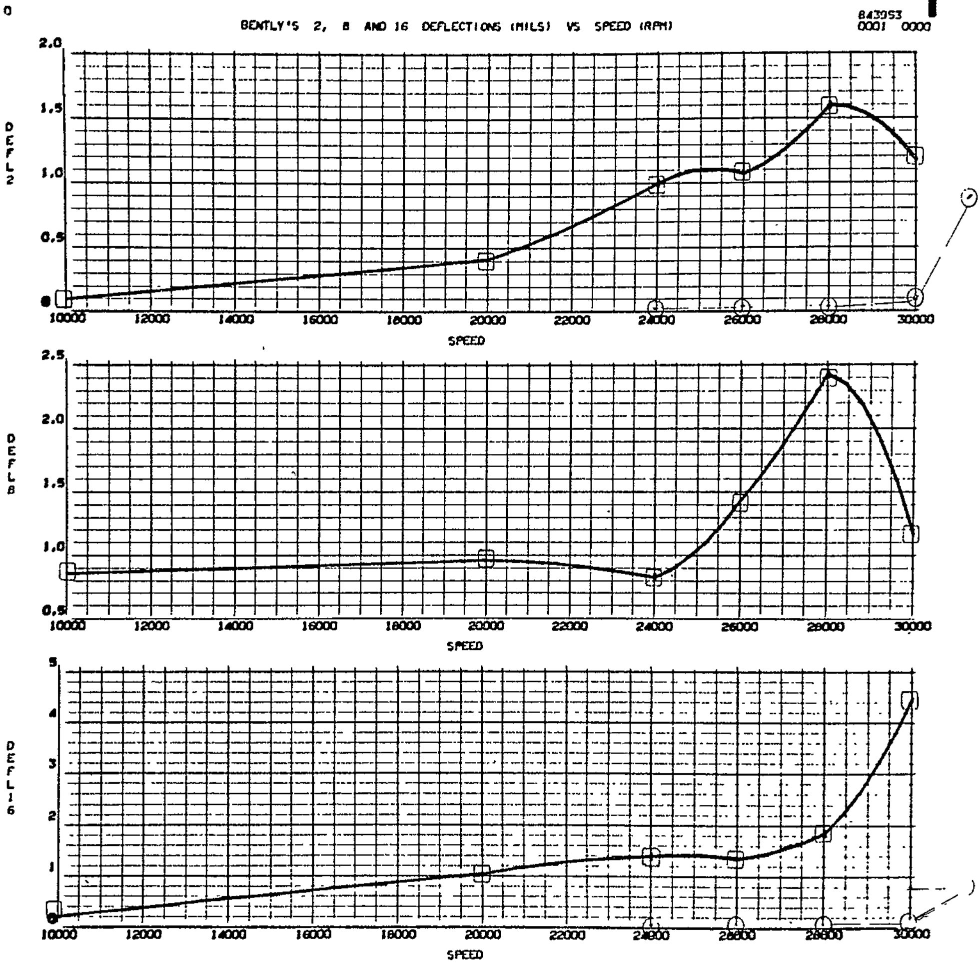


Figure 18

TEST NO. 1144

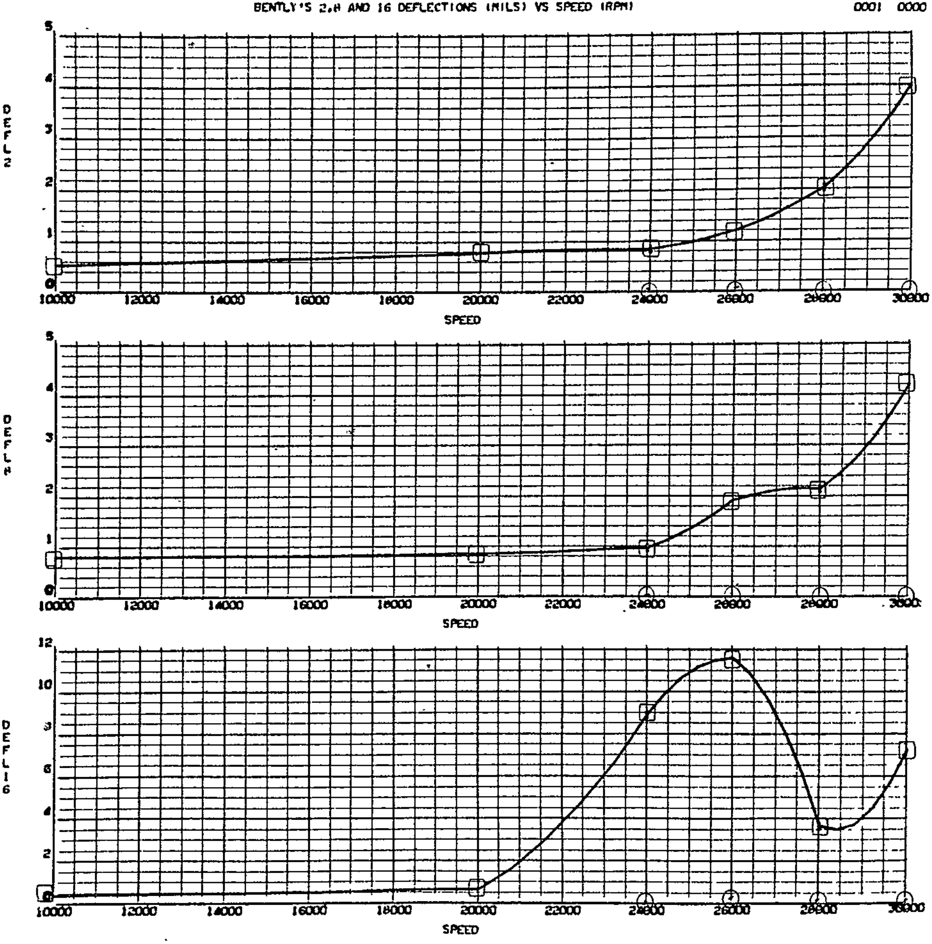


⊙ COMPUTER SIMULATION DATA

▢ EXPERIMENTAL DATA

Figure 19

TEST 1145



⊙ COMPUTER SIMULATION DATA

⊡ EXPERIMENTAL DATA

Figure 20

TEST 1136 26000 RPM

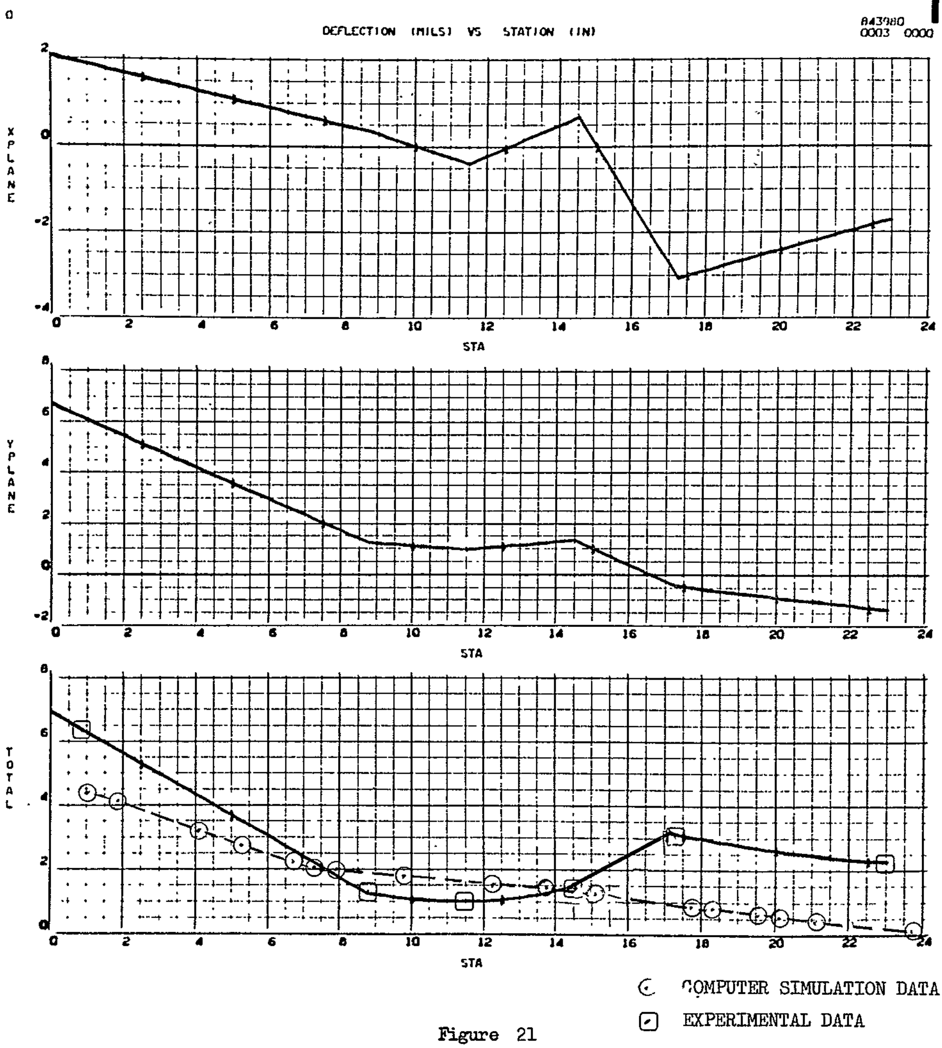


Figure 21

TEST 1136 28000 RPM

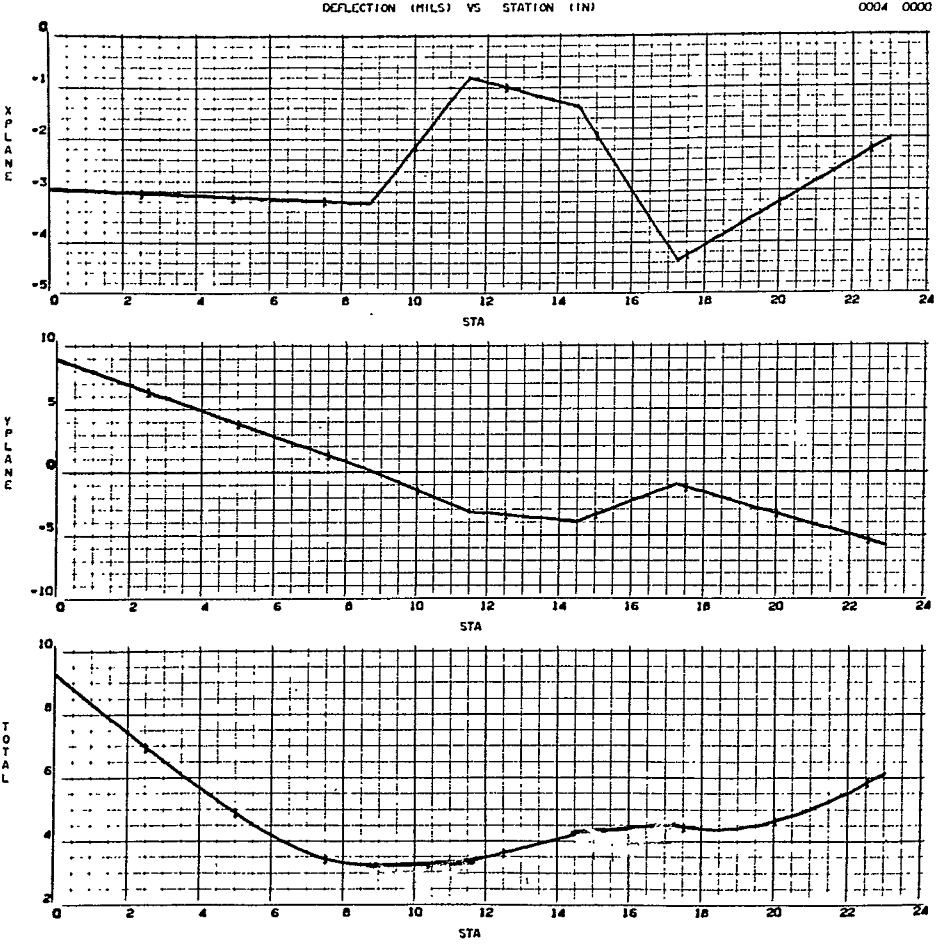


Figure 22

TEST 1136 30000 RPM

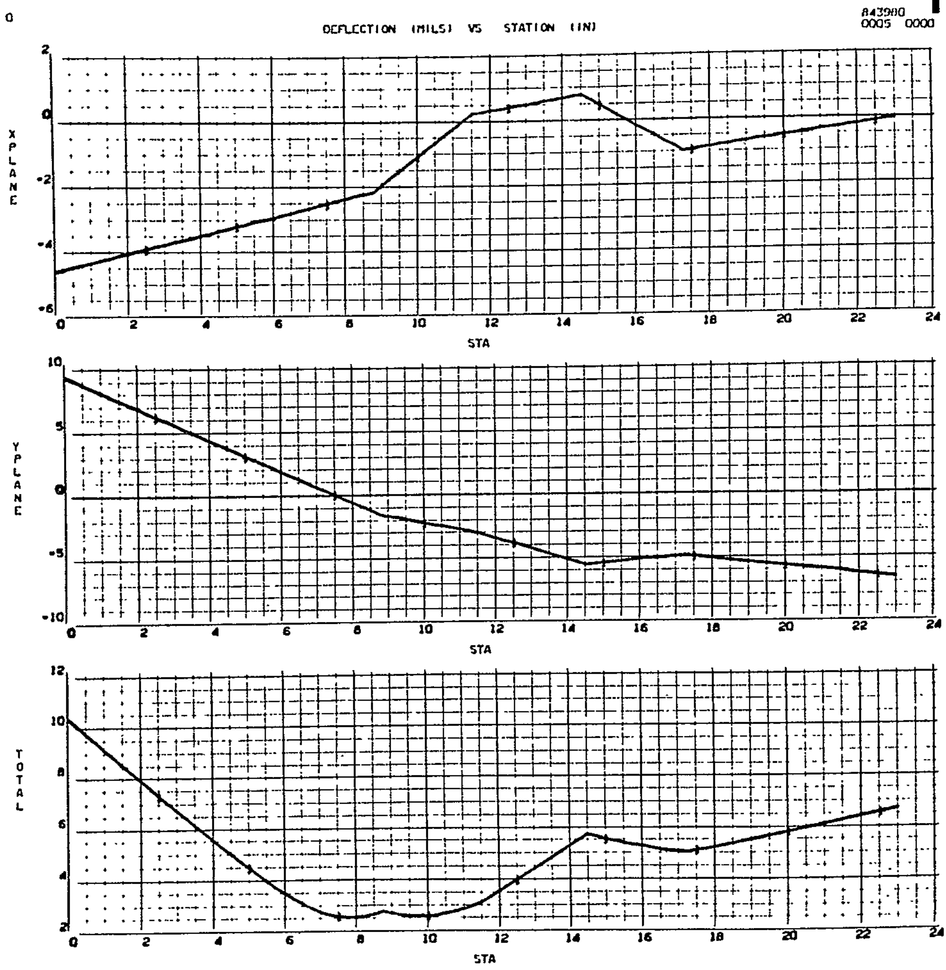
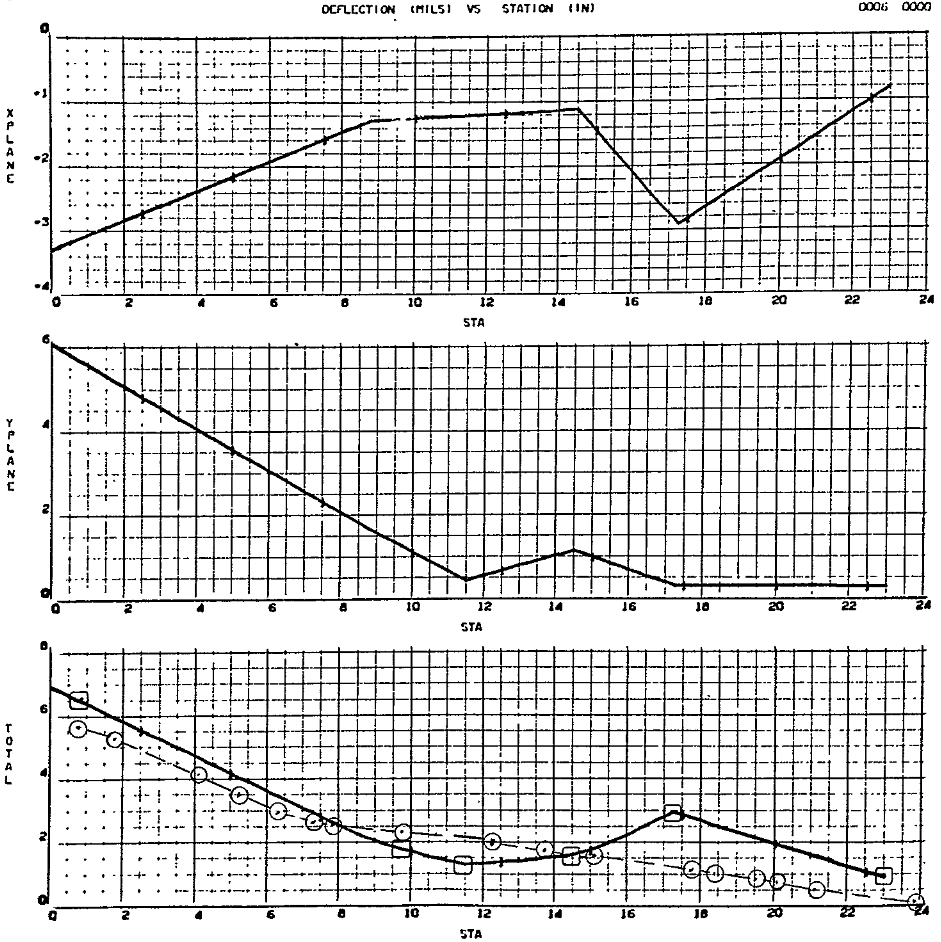


Figure 23

TEST 1137 26000 RPM



⊙ COMPUTER SIMULATION DATA

⊡ EXPERIMENTAL DATA

Figure 24

TEST 1137 28000 RPM

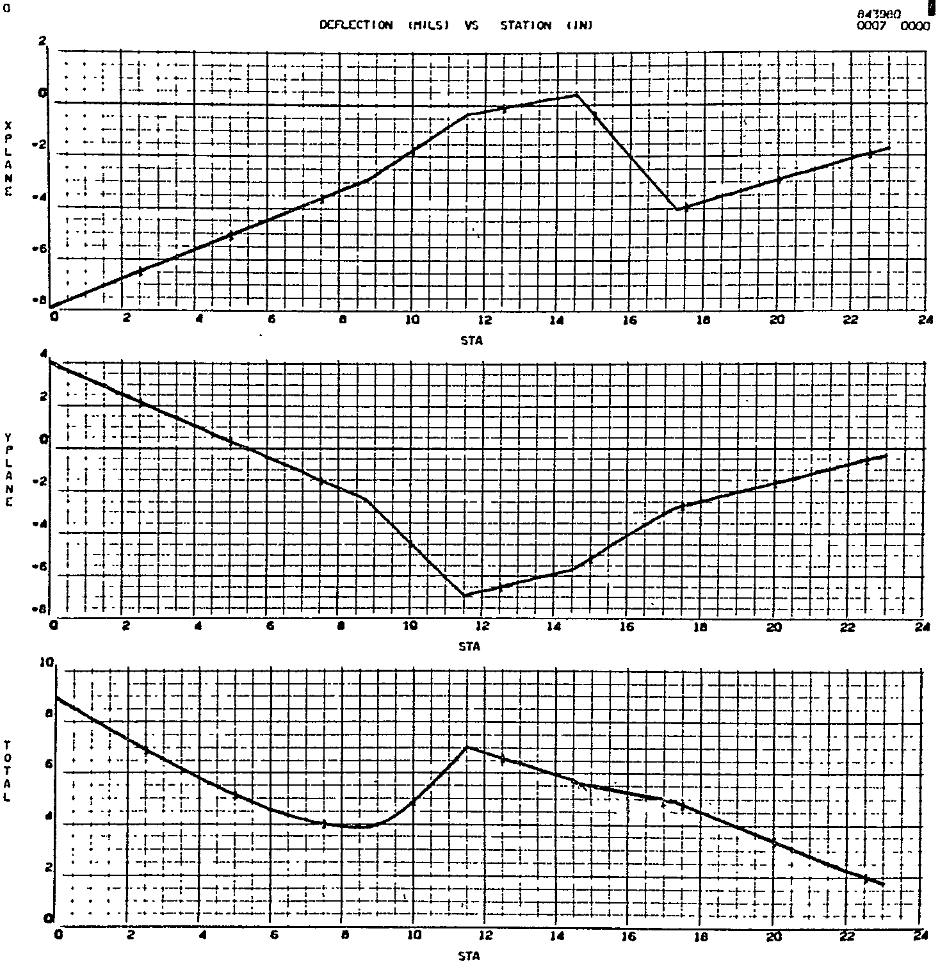


Figure 25

TEST 1137 30000 RPM

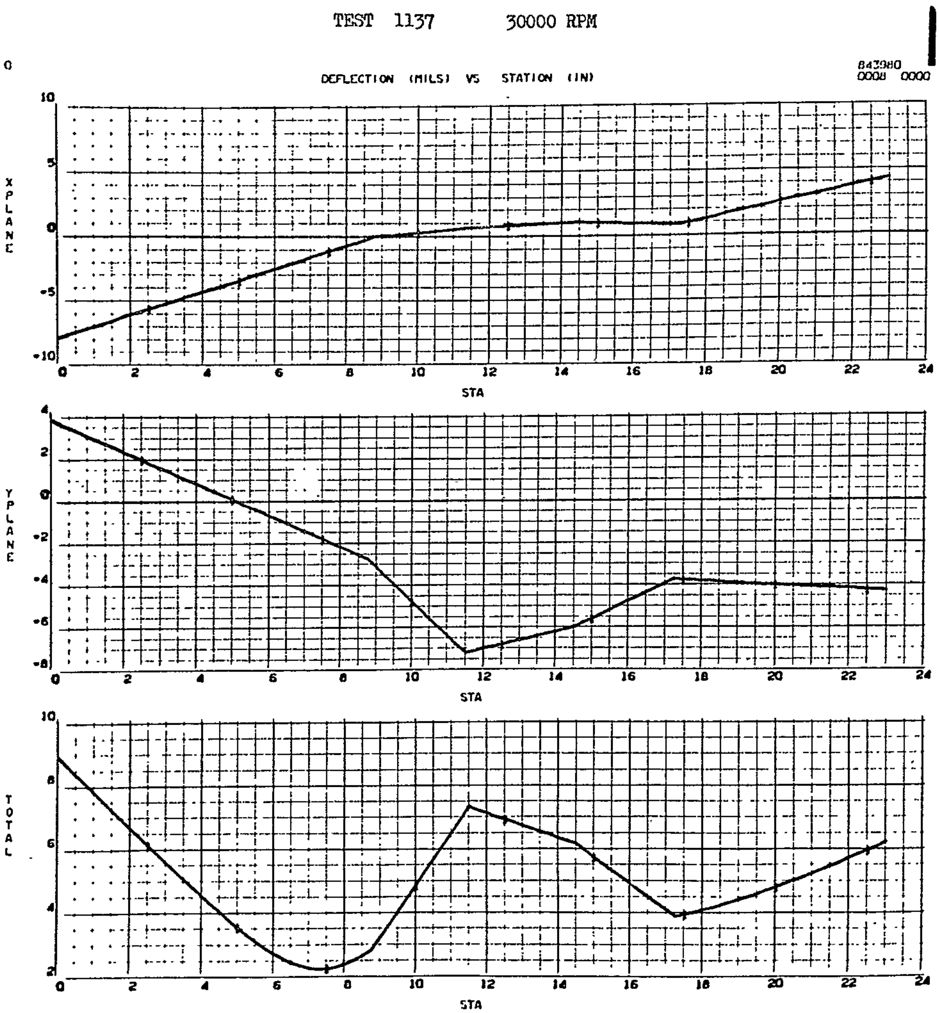


Figure 26

TEST 1138 28000 RPM

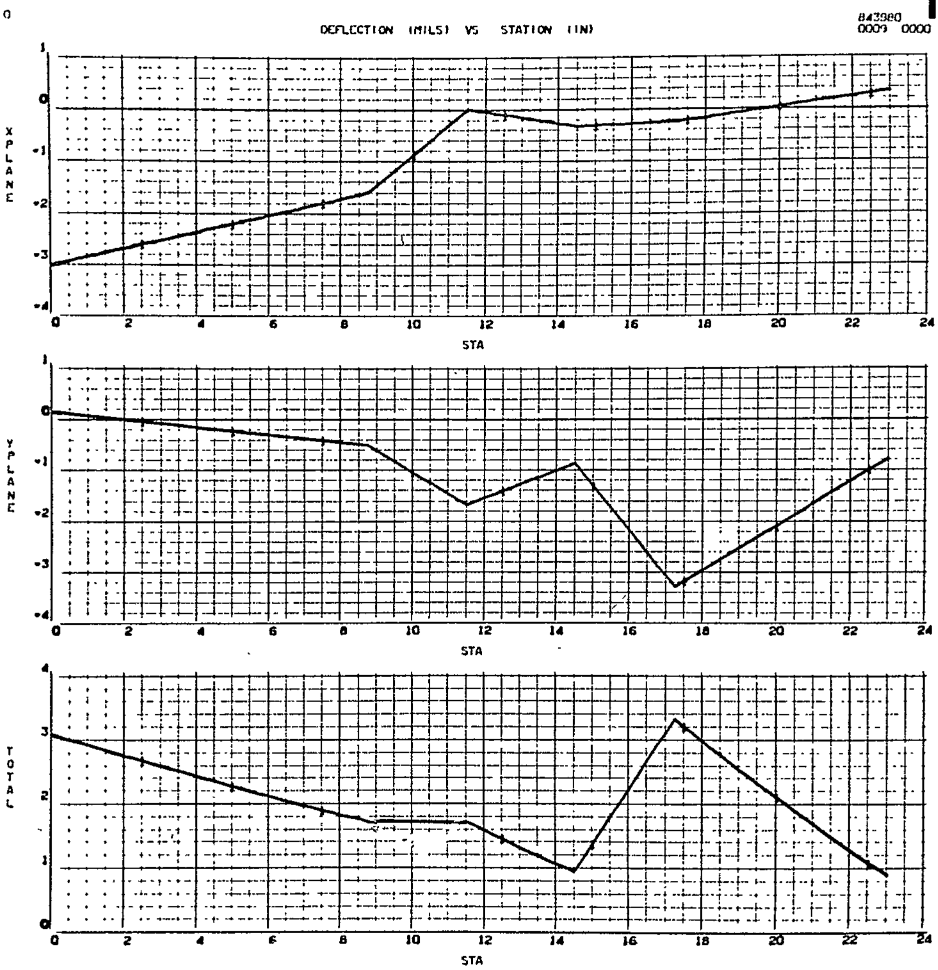


Figure 27

TEST 1138 30000 RPM

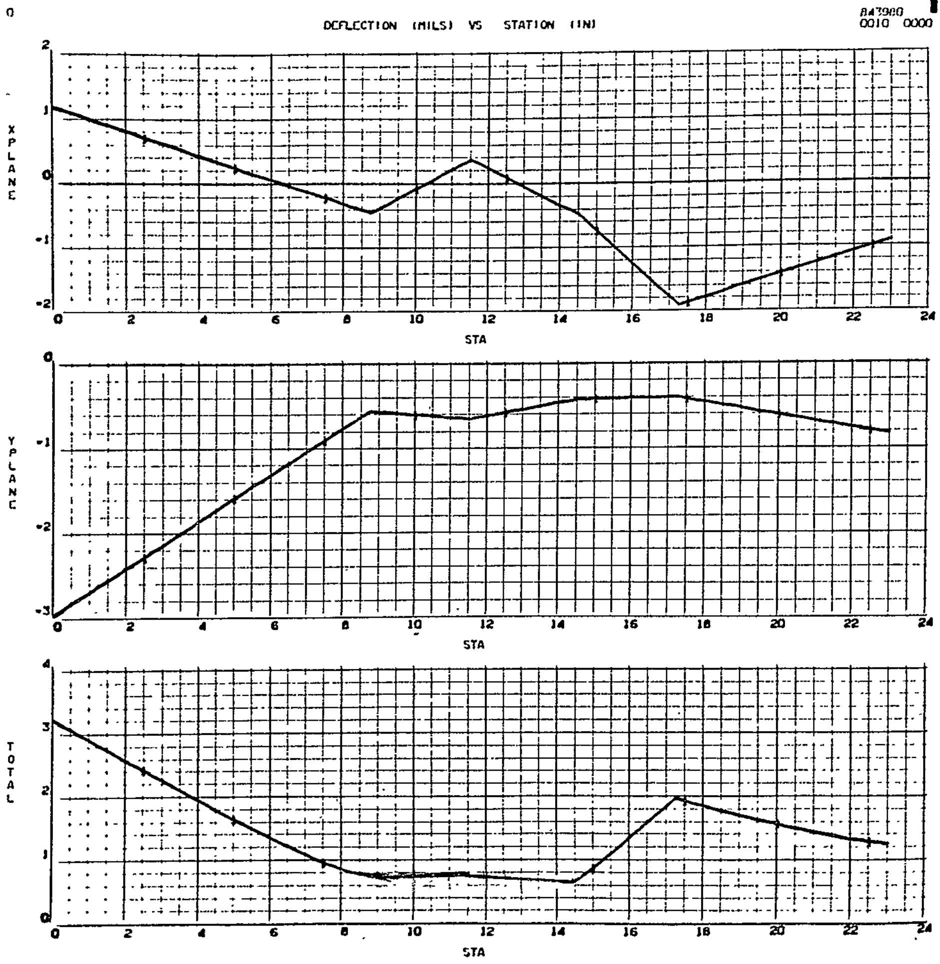
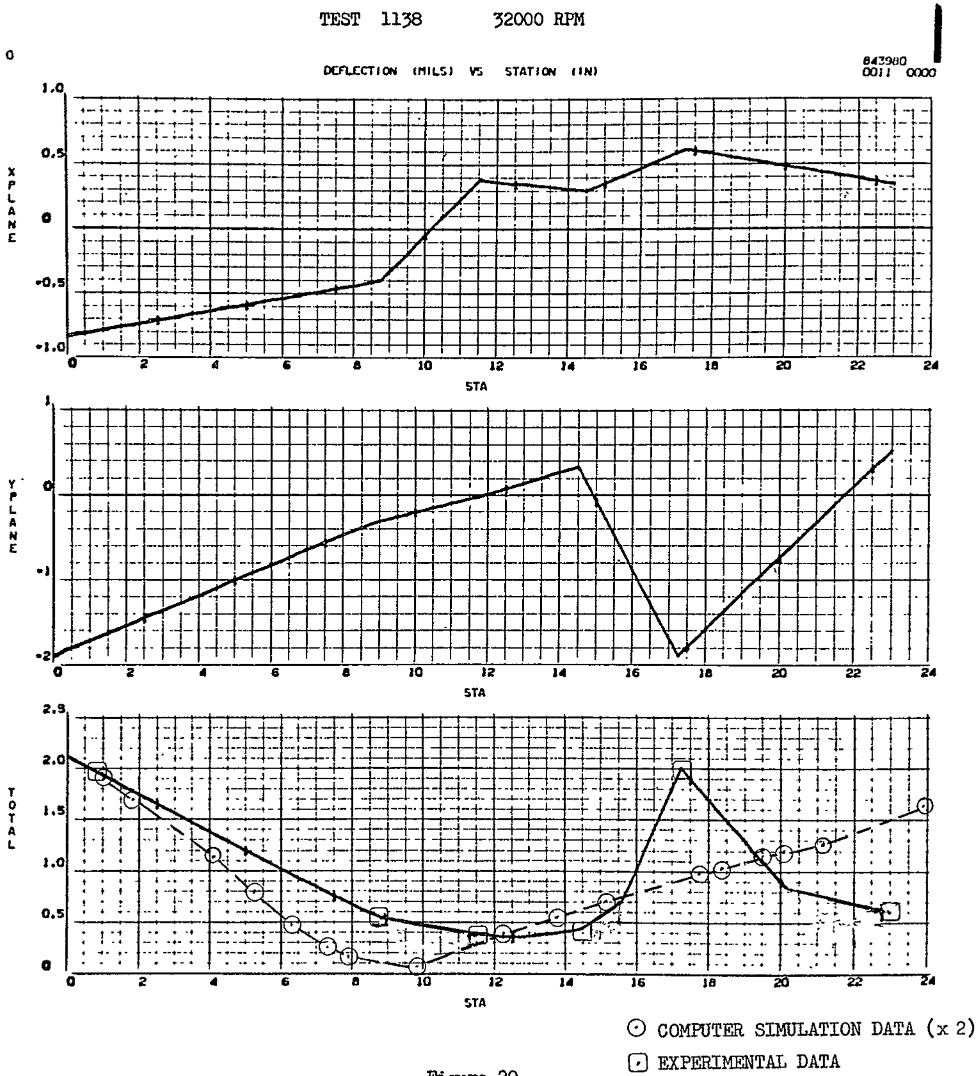


Figure 28



TEST 1138 32000 RPM

⊙ COMPUTER SIMULATION DATA (x 2)

⊡ EXPERIMENTAL DATA

Figure 29

TEST 1139 28000 RPM

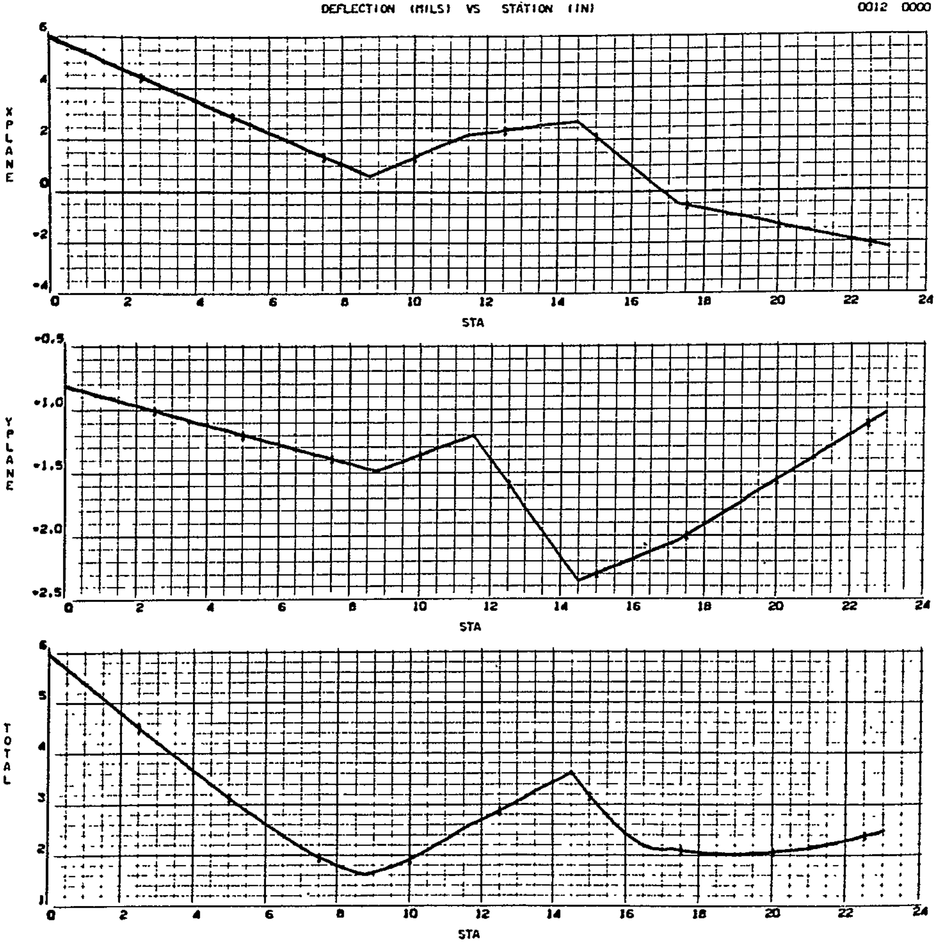


Figure 30

TEST 1139 30000 RPM

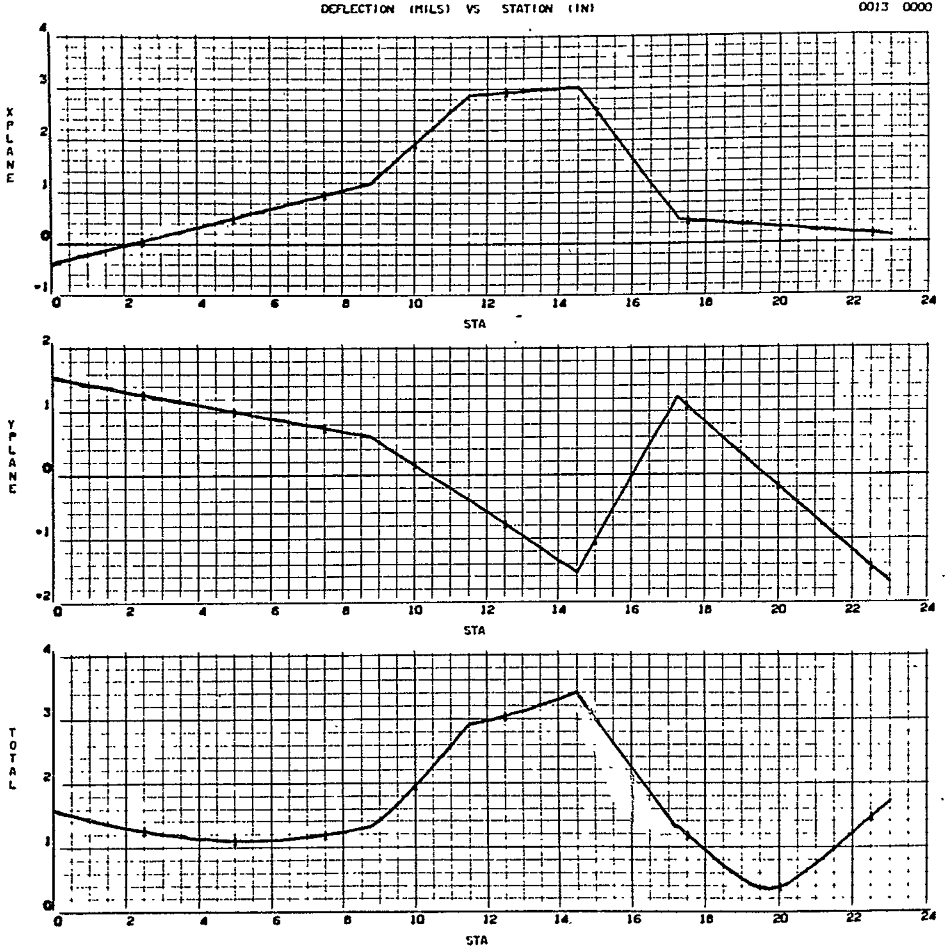


Figure 31

TEST 1139 32000 RPM

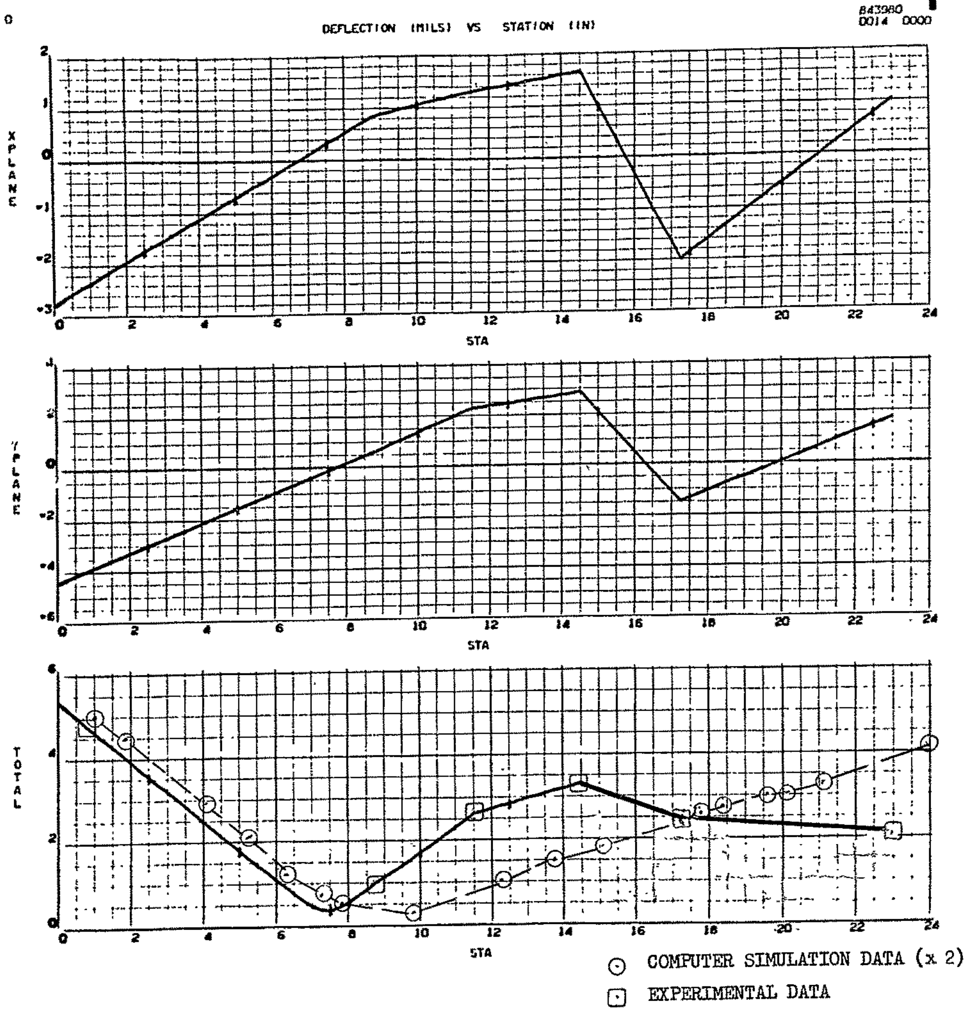


Figure 32

TEST 1142 28000 RPM

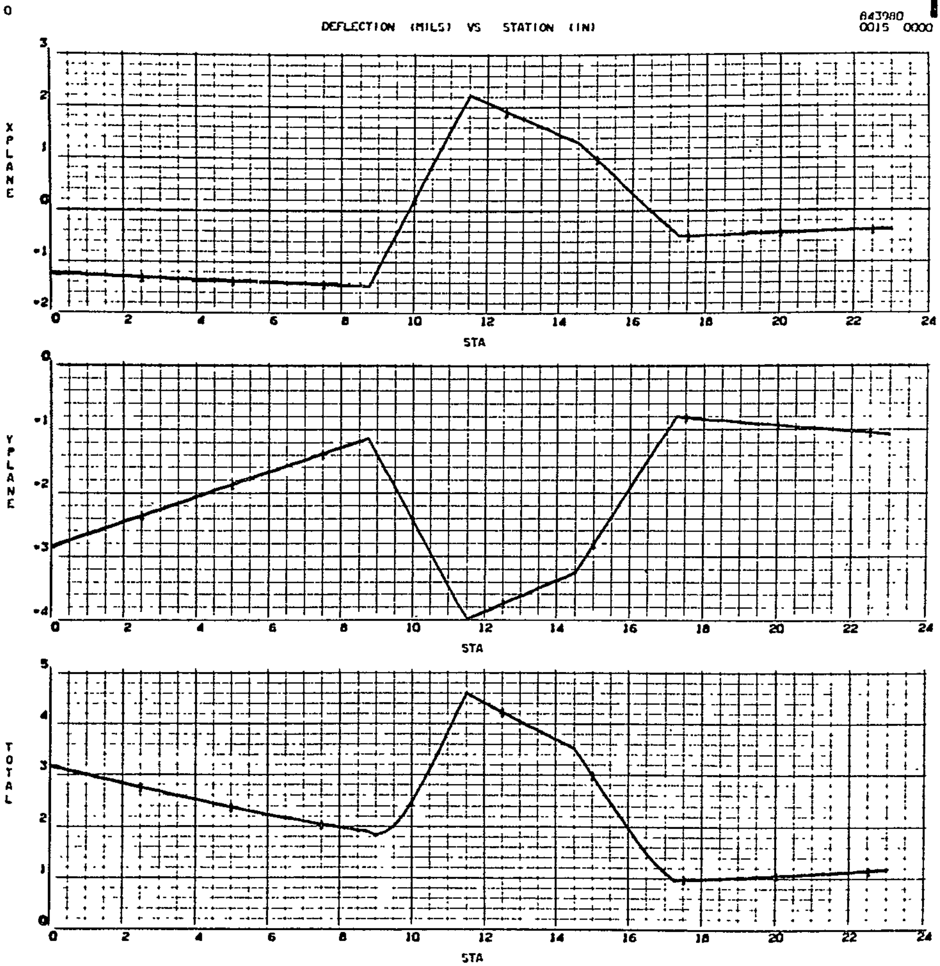


Figure 33

TEST 1142 30000 RPM

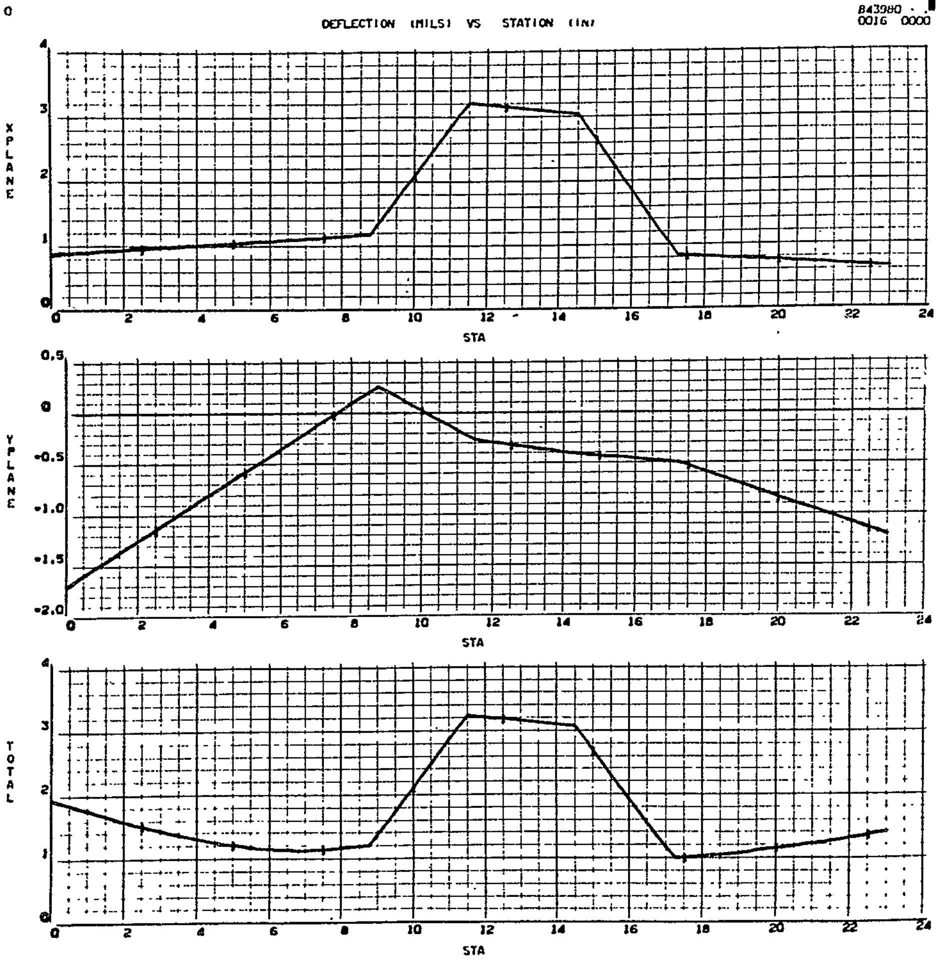
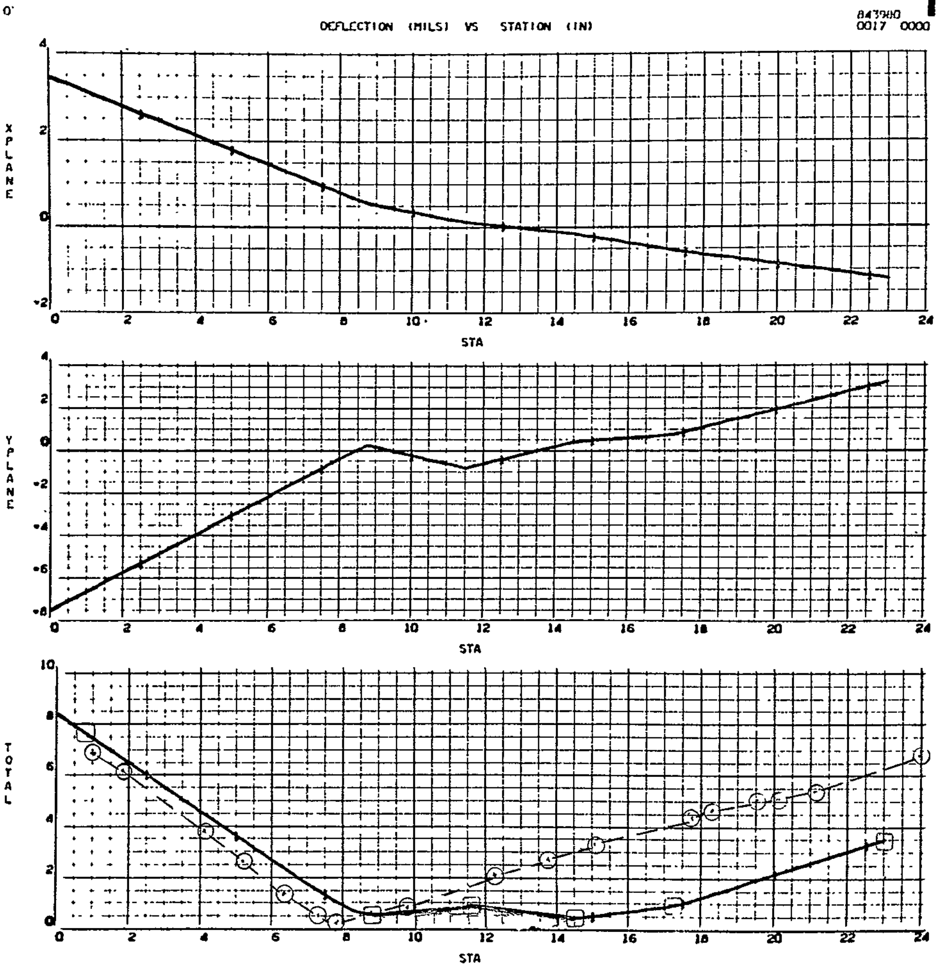


Figure 34

TEST 1142 32000 RPM



⊙ COMPUTER SIMULATION DATA (x 5)

⊡ EXPERIMENTAL DATA

Figure 35

TEST 1143 26000 RPM

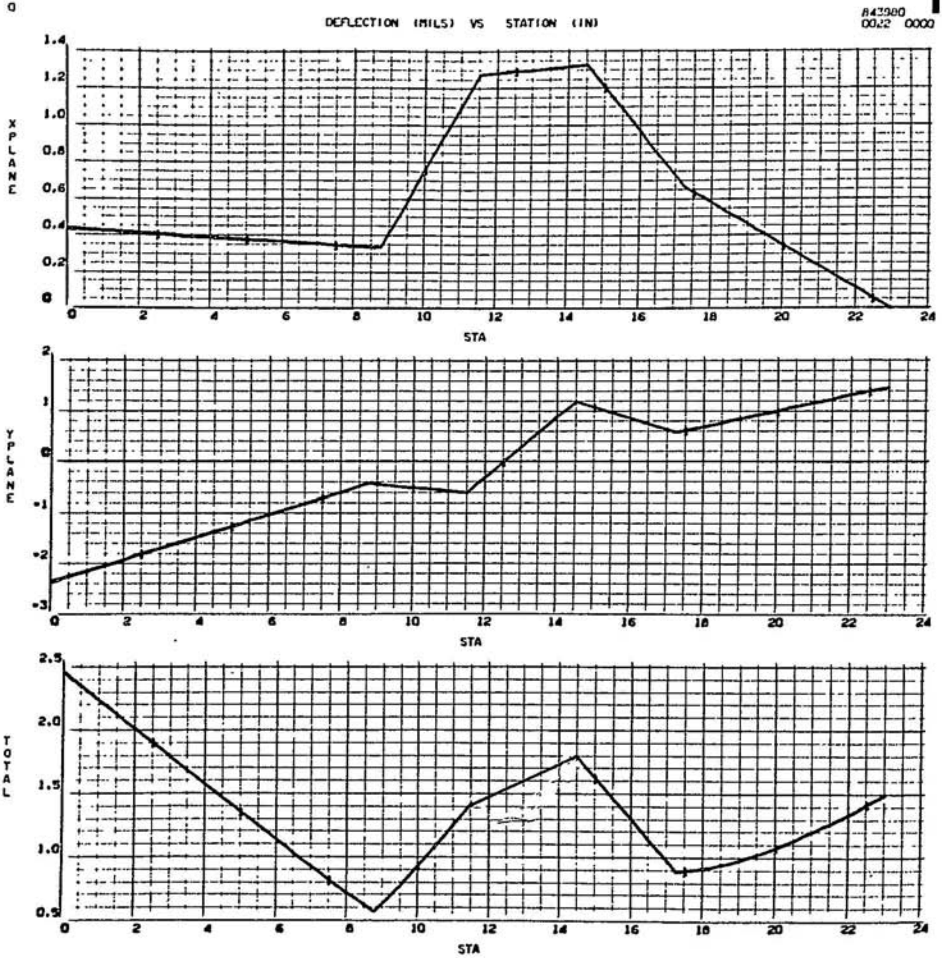


Figure 36

TEST 1143   28000 RPM

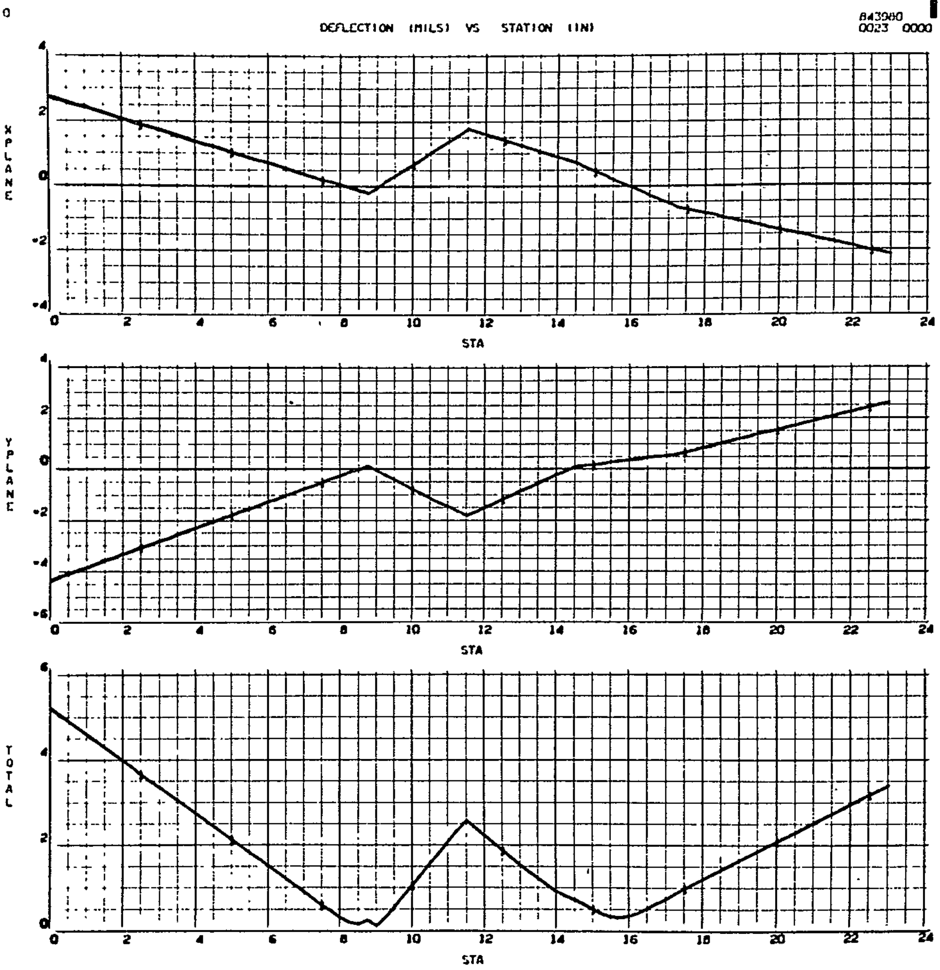


Figure 37

TEST 1143 30000 RPM

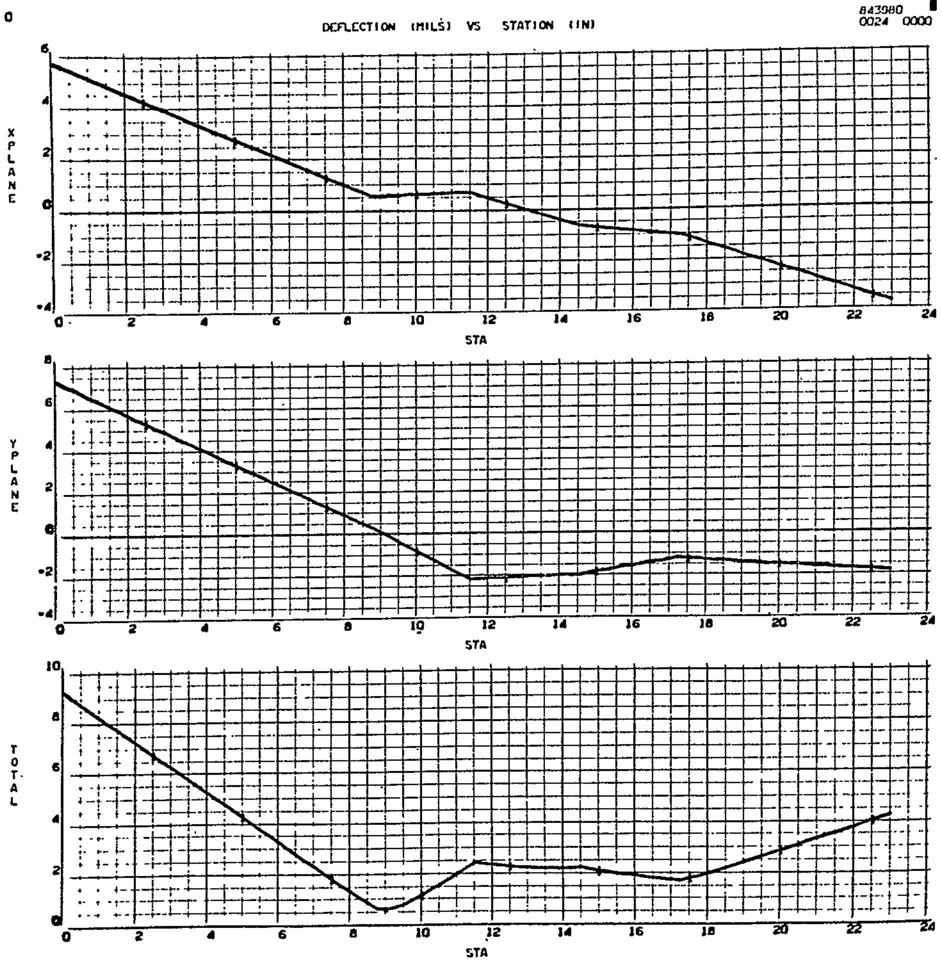


Figure 38

TEST 1144 26000 RPM

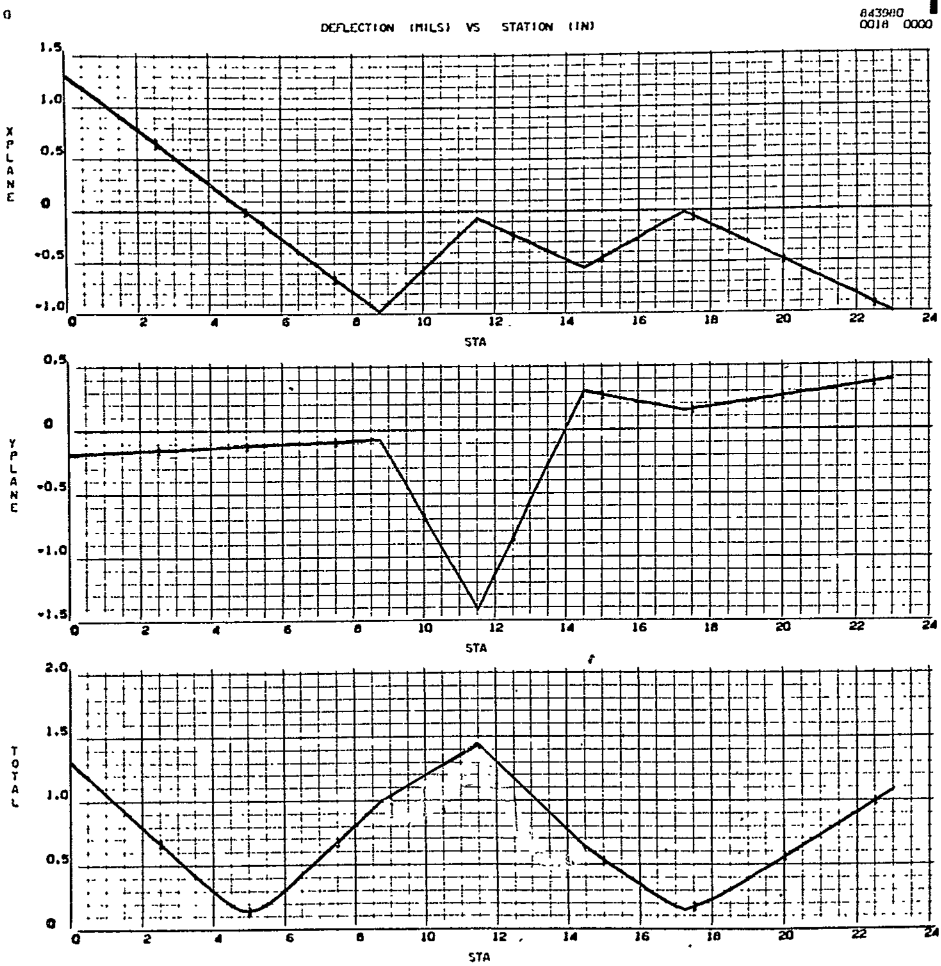


Figure 39

TEST 1144 28000 RPM

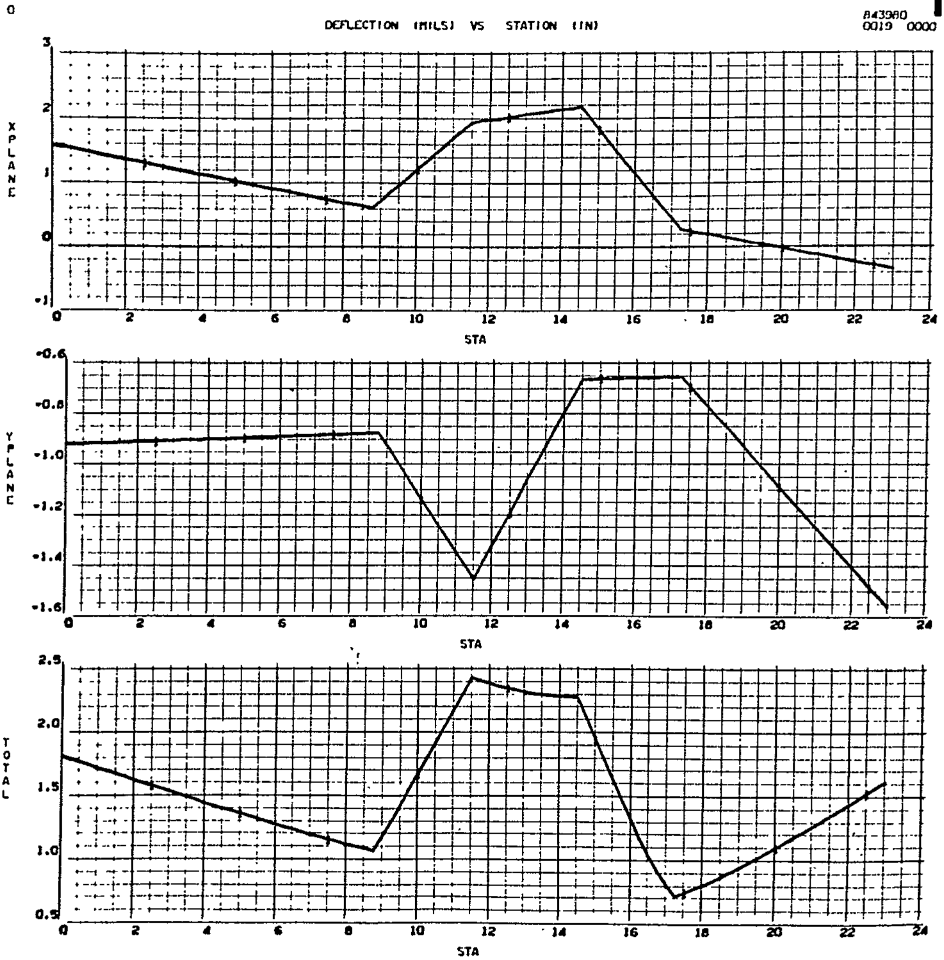


Figure 40

TEST 1144 30000 RPM

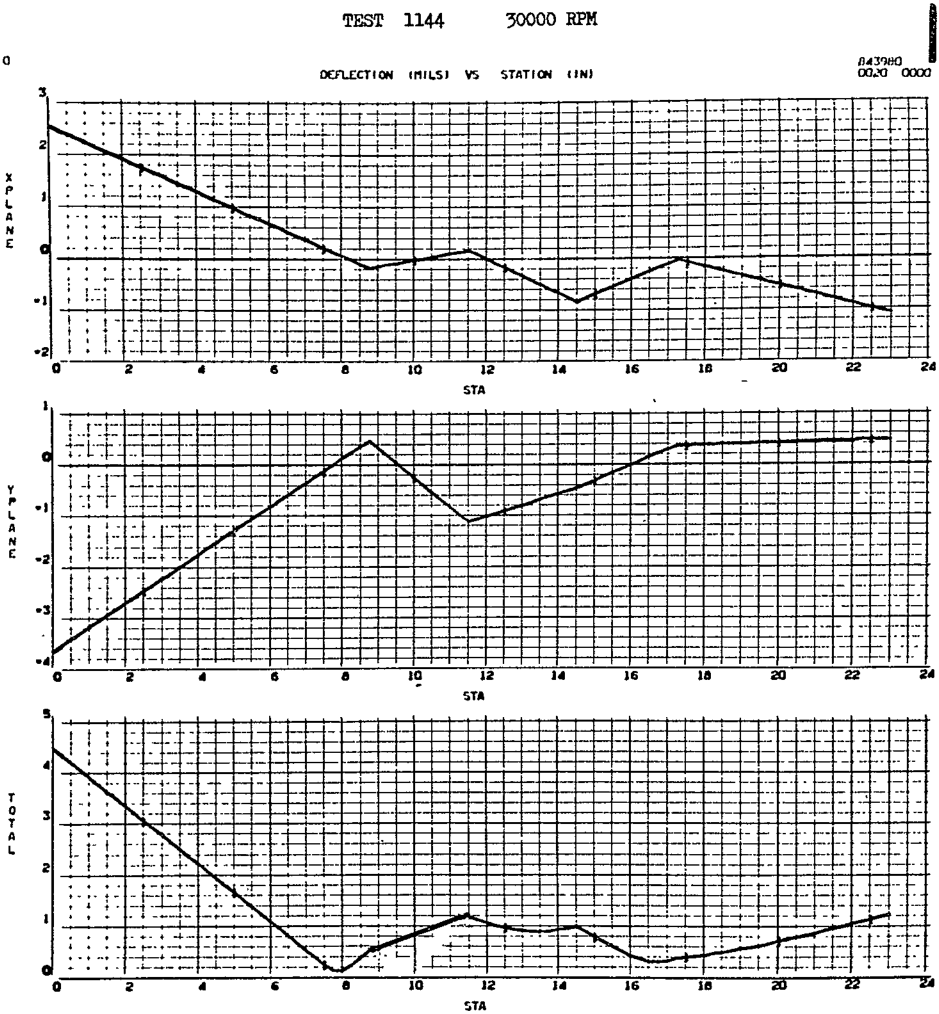
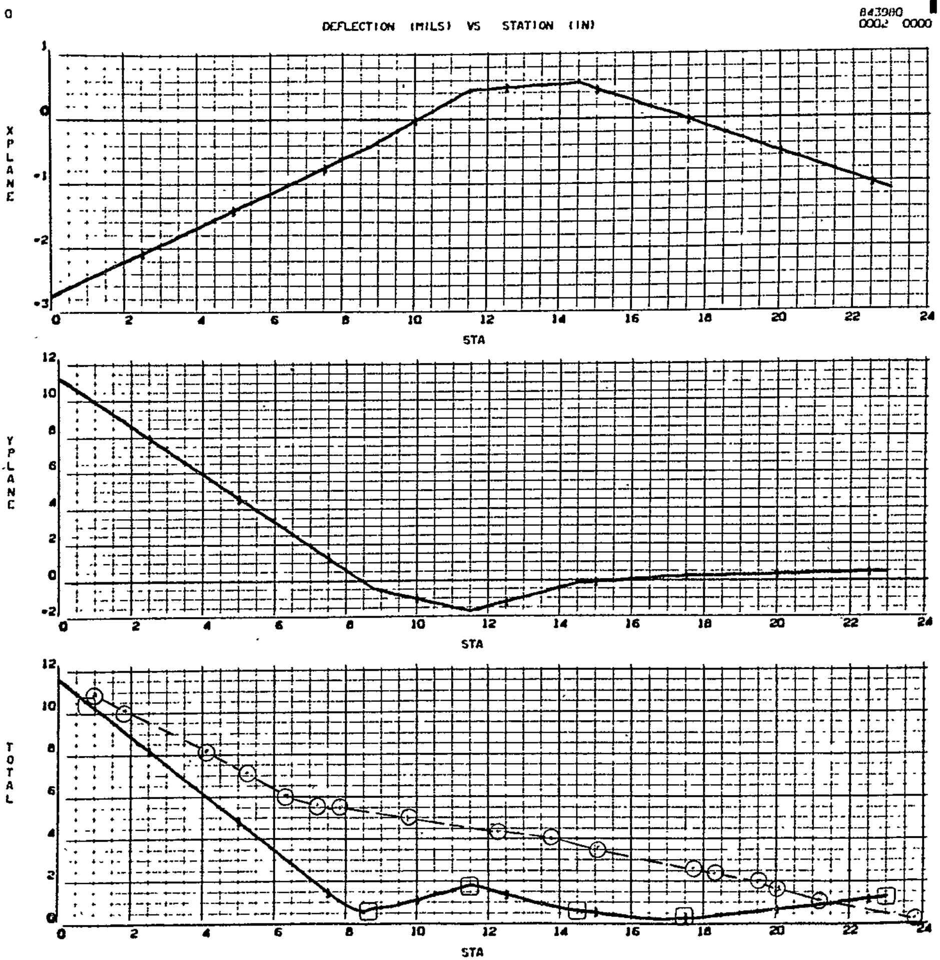


Figure 41

TEST 1145 26000 RPM



⊙ COMPUTER SIMULATION DATA (x 40)

⊡ EXPERIMENTAL DATA

Figure 42.

3

TEST 1145 28000 RPM

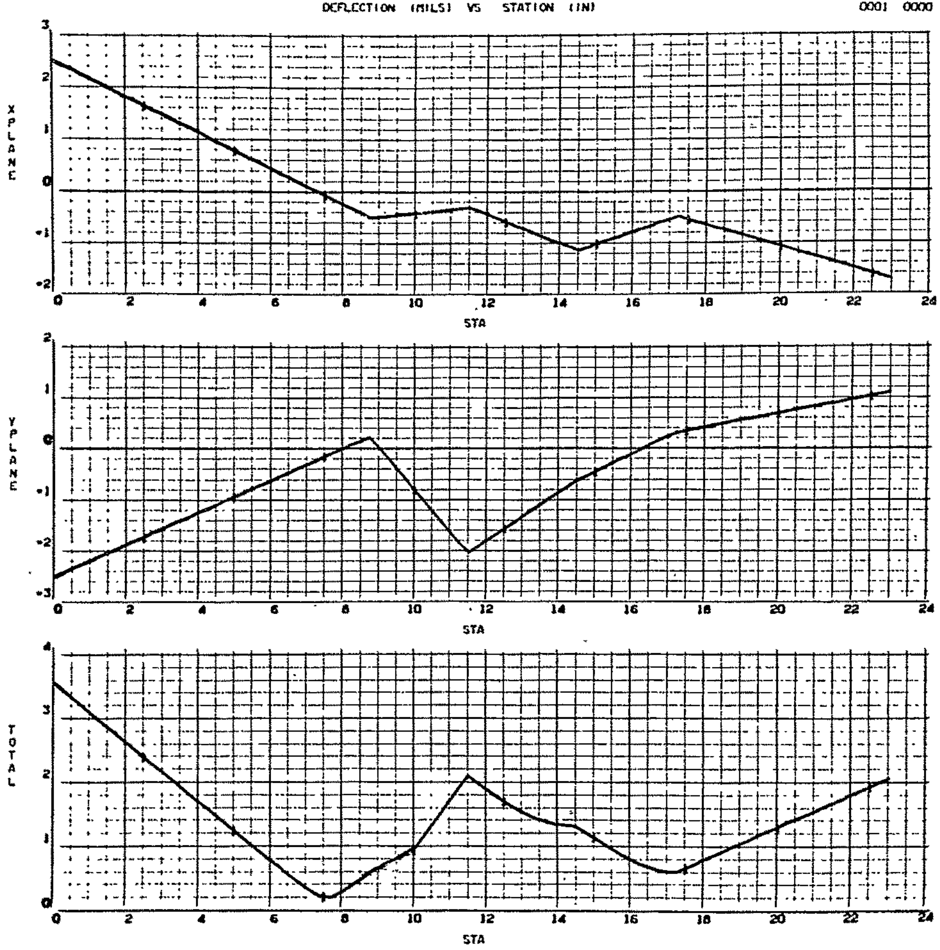


Figure 43

TEST 1145   30000 RPM

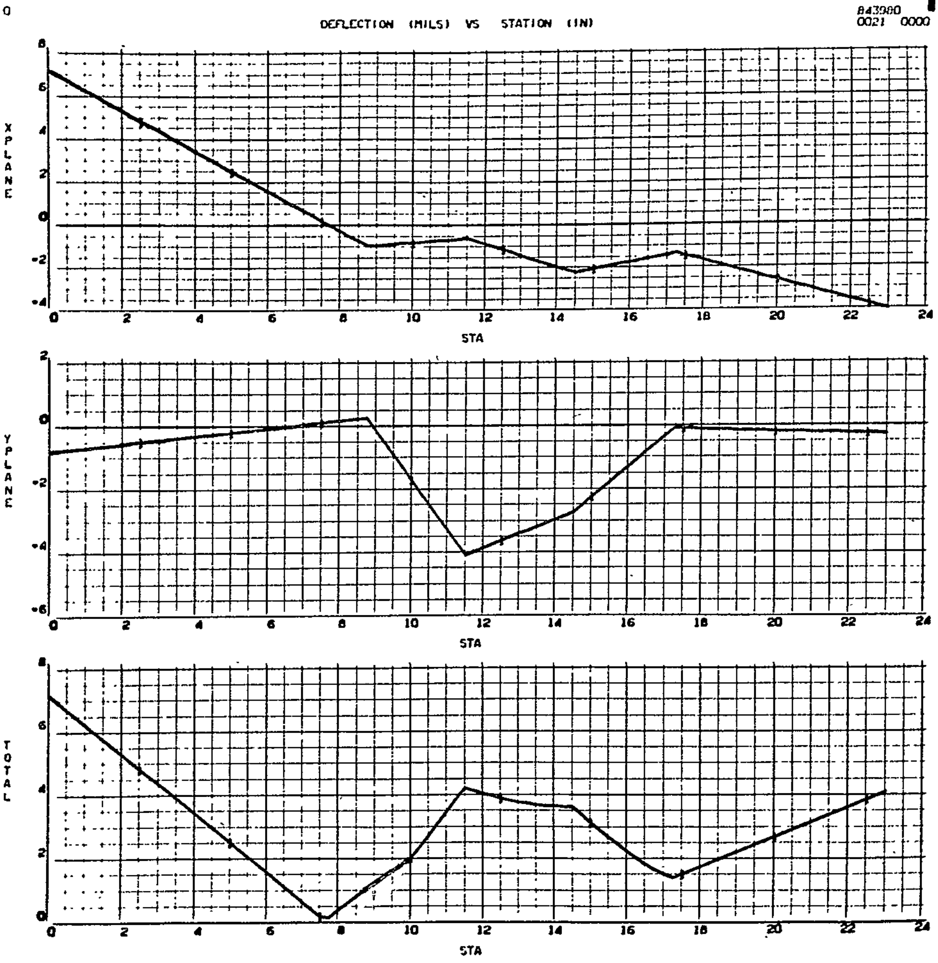


Figure 44

SIMULATION OF EXPERIMENTAL TESTS

SIMULATION MODEL

A discrete 17 mass model was constructed to describe the physical configuration of the Mark-25 pump rotor. Table 5 lists the mass and inertia properties for each of the 17 stations.

The four bearings, in two duplex pairs, were located at Stations 5, 6, 15 and 16. A speed sensitive linear stiffness characteristic was used for each of the four angular contract ball bearings.

Eight different cases of rotor mass unbalances corresponding to the experimental test runs were used in the computer program input data. These unbalance cases are listed in Table 6. Six steady state speed slices were simulated for each test run. The speeds were from 24,000 to 34,000 rpm in 2000 rpm increments.

RESULTS OF THE COMPUTER SIMULATION

The results of the computer simulation are listed in Tables 7 through 14. Although excellent correlation was achieved between the results of the new computer program versus a synchronous response program (see Table 2), large variances occurred in the simulation runs as compared to the experimental data.

For graphical comparison the simulation data is plotted on the experimental plots of deflection versus speed, and is identified by a small circle.* Also, comparison of the calculated vs. experimental mode shapes are shown in Figures 21, 24, 29, 32, 35 and 42. In some cases it was necessary to multiply the simulation data by a scale factor. However, the mode shapes compared favorably. The close agreement between the computer results and that from an existing synchronous rotordynamic response program suggests

*See Figures 13 through 20.

that the computer program is valid and able to generate accurate results. The discrepancies between the test and computer data should only result from inadequacies in test model simulation. Comparisons between the computer and test data (Figures 13 through 20) reveal,

1. Computer simulated deflections are generally smaller than the test data.
2. Within the speed range tested, reasonably negative curvatures (inverted U shape) of the deflection versus speed curves at the local maximum deflection regions are observed.
3. The local maximum deflection points from the computed data are in general at a higher speed than that from the test data.

Item 1 above suggests that the actual mass eccentricities of the test rotor may be greater than those used in the simulation. Since there are only a maximum of two unbalance locations for each unbalance configuration, it is possible that there are compensating local unbalances not detectable during the balance procedure. That is, the actual unbalance distribution of a rotor unbalance configuration is different from that used in the computer program. Although the postulate of actual eccentricities being larger than those used may tend to explain a good portion of the discrepancies between test and computer points, no such errors have been found in the test procedure.

Item 2 indicates that certain amount of damping exists. It appears that besides the small amount of inherent damping in the elastic bearing and mount components the high damping component Kirksite base for the test apparatus may contribute to the system as a significant damping parameter. These was no damping included in the computer analysis. The damping parameter has not been included in the STARUP subroutine.

Item 3 suggests that the support bearings stiffness and/or rotor rigidity are higher in the computer model than in the test mode, thus resulting in higher computed critical speeds. The test rotor is fabricated from individual disk components and is of reasonably complex structure. It is possible that the computer model may represent a more stiff rotor than the real one.

As far as the support bearing's stiffness simulation is concerned, the discrepancies between the actual value and that used in the computer program should not be very large. In the past experiences, the use of similar bearing stiffness appeared to give reasonable critical speed results comparable to those observed from related test data.

Tables 7 through 14 are copies of the actual printout from the computer simulation runs and list the following:

a. X displacement array
b. Y displacement array
c. Rotor deflection vector array
d. Rotor deflection vector phase angle array
e. X velocity array
f. Y velocity array
g. Whirl frequency array
h. Whirl to spin frequency ratio array

TABLE 5

MARK-25 PUMP ROTOR MASS AND INERTIA PROPERTIES

Station	Location Inches	Mass, $lb\text{-}sec^2/in$	I_D, $lb\text{-}in\text{-}sec^2$	I_P, $lb\text{-}in\text{-}sec^2$	$I_D - I_P$
1	1.0	.0023293	.0018504	.0022752	- .0004248
2	1.8	.008456	.027589	.047474	- .019885
3	4.02	.093548	.142081	.24449	- .10241
4	5.22	.017604	.052015	.089431	- .037416
5	6.32	.0056835	.0028311	.0046549	- .0018238
6	7.32	.0021558	.0010739	.0017656	- .0006917
7	7.81	.017136	.04451	.079042	- .049534
8	9.83	.054180	.1450	.2515	- .15662
9	12.24	.0331708	.0934	.1585	- .08504
10	13.73	.040357	.11800	.1914	- .09338
11	15.03	.0361565	.1019	.1834	- .10151
12	17.72	.030516	.0854	.1593	- .10791
13	18.32	.030516	.0854	.1593	- .10791
14	19.53	.021697	.0608	.1175	- .07673
15	20.02	.002661	.0037044	.0023834	+ .0013210
16	21.13	.010644	.014818	.009534	+ .005289
17	23.98	.011584	.012806	.013423	- .000617

TABLE 6

MARK-25 PUMP SIMULATION DATA

Run	Station	Mass lb-sec^2/in	Eccentricity, e	Phase, Degree
1136	2	.008456	.00143"	138.0°
	5	.005684	.00341"	139.0°
1137	2	.008456	.00185"	195.0°
	5	.005684	.00425"	196.0°
1138	2	.008456	.000175"	-24.4°
	5	.005684	.000366"	-23.5°
1139	2	.008456	.000429"	-32.3°
	5	.005684	.0011"	-29.7°
1141	Base	Run	0.0"	0.0°
1142	10	.04036	.0002065"	121.0°
1143	17	.01158	.000058"	165.0°
1144	13	.03052	.0000964"	125.0°
1145	7	.01714	.000174"	115.0°

TABLE 7

TEST 1136 SIMULATION AT 26,000 rpm

```
X DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
      -3.3700D-03  -3.1209D-03  -2.4481D-03  -2.0794D-03  -1.7235D-03  -1.5516D-03
      -1.5324D-03  -1.3899D-03  -1.1926D-03  -1.0622D-03  -9.4029D-04  -6.7142D-04
      -6.1005D-04  -4.8231D-04  -4.2161D-04  -2.9067D-04   2.9491D-05

Y DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
       2.8554D-03   2.6442D-03   2.0736D-03   1.7609D-03   1.4591D-03   1.3134D-03
       1.2971D-03   1.1763D-03   1.0092D-03   8.9876D-04   7.9555D-04   5.6788D-04
       5.1591D-04   4.0776D-04   3.5637D-04   2.4552D-04  -2.5563D-05

ROTOR DEFLECTION VECTOR ARRAY (IN.)
       4.4170D-03   4.0904D-03   3.2083D-03   2.7248D-03   2.2582D-03   2.0329D-03
       2.0077D-03   1.8209D-03   1.5623D-03   1.3914D-03   1.2317D-03   8.7937D-04
       7.9895D-04   6.3158D-04   5.5204D-04   3.8049D-04   3.9028D-05

PHASE ANGLE ARRAY FOR ROTOR DEFLECTION VECTORS, DEGREES
       1.3972D 02   1.3973D 02   1.3973D 02   1.3974D 02   1.3975D 02   1.3975D 02
       1.3975D 02   1.3976D 02   1.3976D 02   1.3976D 02   1.3977D 02   1.3978D 02
       1.3978D 02   1.3979D 02   1.3979D 02   1.3981D 02   3.1908D 02

X VELOCITY ARRAY IN STATIONARY COORDINATES,(IN/SEC)
      -7.7745D 00  -7.1994D 00  -5.6458D 00  -4.7944D 00  -3.9726D 00  -3.5758D 00
      -3.5317D 00  -3.2029D 00  -2.7478D 00  -2.4471D 00  -2.1661D 00  -1.5462D 00
      -1.4047D 00  -1.1102D 00  -9.7030D-01  -6.6846D-01   6.9622D-02

Y VELOCITY ARRAY IN STATIONARY COORDINATES,(IN/SEC)
      -9.1755D 00  -8.4973D 00  -6.6655D 00  -5.6616D 00  -4.6927D 00  -4.2249D 00
      -4.1722D 00  -3.7842D 00  -3.2470D 00  -2.8920D 00  -2.5601D 00  -1.8281D 00
      -1.6610D 00  -1.3132D 00  -1.1479D 00  -7.9143D-01   8.0276D-02

WHIRL FREQUENCY ARRAY,RPM
       2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04
       2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04
       2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04

WHIRL TO SPIN FREQUENCY RATIO ARRAY
       1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00
       1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00
       1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   9.9999D-01
```

TABLE 8

TEST 1137 SIMULATION AT 26,000 RPM

```
 X DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
        -5.3679D-03  -4.9705D-03  -3.8972D-03  -3.3091D-03  -2.7415D-03  -2.4676D-03
        -2.4369D-03  -2.2098D-03  -1.8957D-03  -1.6881D-03  -1.4942D-03  -1.0663D-03
        -9.6869D-04  -7.6546D-04  -6.6892D-04  -4.6066D-04   4.8728D-05

 Y DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
        -1.6599D-03  -1.5373D-03  -1.2059D-03  -1.0243D-03  -8.4901D-04  -7.6437D-04
        -7.5490D-04  -6.8470D-04  -5.8751D-04  -5.2327D-04  -4.6324D-04  -3.3079D-04
        -3.0056D-04  -2.3764D-04  -2.0773D-04  -1.4323D-04   1.4487D-05

 ROTOR DEFLECTION VECTOR ARRAY (IN.)
         5.6187D-03   5.2028D-03   4.0795D-03   3.4640D-03   2.8700D-03   2.5833D-03
         2.5511D-03   2.3135D-03   1.9846D-03   1.7674D-03   1.5643D-03   1.1165D-03
         1.0142D-03   8.0150D-04   7.0043D-04   4.8241D-04   5.0835D-05

 PHASE ANGLE ARRAY FOR ROTOR DEFLECTION VECTORS, DEGREES
         1.9718D 02   1.9719D 02   1.9719D 02   1.9720D 02   1.9721D 02   1.9721D 02
         1.9721D 02   1.9722D 02   1.9722D 02   1.9722D 02   1.9723D 02   1.9723D 02
         1.9724D 02   1.9725D 02   1.9725D 02   1.9727D 02   1.6557D 01

 X VELOCITY ARRAY IN STATIONARY COORDINATES,(IN/SEC)
         4.5196D 00   4.1856D 00   3.2834D 00   2.7889D 00   2.3120D 00   2.0817D 00
         2.0554D 00   1.8643D 00   1.5996D 00   1.4247D 00   1.2613D 00   9.0065D-01
         8.1833D-01   6.4702D-01   5.6558D-01   3.9004D-01  -3.9393D-02

 Y VELOCITY ARRAY IN STATIONARY COORDINATES,(IN /SEC)
        -1.4615D 01  -1.3533D 01  -1.0611D 01  -9.0097D 00  -7.4643D 00  -6.7184D 00
        -6.6349D 00  -6.0168D 00  -5.1614D 00  -4.5963D 00  -4.0682D 00  -2.9033D 00
        -2.6375D 00  -2.0841D 00  -1.8213D 00  -1.2542D 00   1.3268D-01

 WHIRL FREQUENCY ARRAY,RPM
         2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04
         2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04
         2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04

 WHIRL TO SPIN FREQUENCY RATIO ARRAY
         1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00
         1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00
         1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   9.9999D-01
```

TABLE 9

TEST 1138 SIMULATION AT 32,000 RPM

```
X DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
     -8.9574D-04  -7.9473D-04  -5.2262D-04  -3.7265D-04  -2.2741D-04  -1.2549D-04
     -3.7334D-05   4.1334D-05   1.8349D-04   2.6620D-04   3.3372D-04   4.6182D-04
      4.8891D-04   5.3972D-04   5.5193D-04   5.9336D-04   7.6412D-04

Y DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
      3.6902D-04   3.2740D-04   2.1530D-04   1.5352D-04   9.3674D-05   5.1662D-05
      3.5924D-05  -1.7135D-05  -7.5755D-05  -1.0986D-04  -1.3770D-04  -1.9052D-04
     -2.0169D-04  -2.2264D-04  -2.2767D-04  -2.4475D-04  -3.1516D-04

ROTOR DEFLECTION VECTOR ARRAY (IN.)
      9.6876D-04   8.5953D-04   5.6523D-04   4.0303D-04   2.4594D-04   1.3571D-04
      9.4434D-05   4.4745D-05   1.9851D-04   2.8798D-04   3.6102D-04   4.9958D-04
      5.2687D-04   5.8384D-04   5.9705D-04   6.4185D-04   8.2656D-04

PHASE ANGLE ARRAY FOR ROTOR DEFLECTION VECTORS, DEGREES
      1.5761D 02   1.5761D 02   1.5761D 02   1.5761D 02   1.5761D 02   1.5762D 02
      1.5764D 02   3.3748D 02   3.3757D 02   3.3757D 02   3.3758D 02   3.3758D 02
      3.3758D 02   3.3758D 02   3.3758D 02   3.3758D 02   3.3759D 02

X-VELOCITY ARRAY IN STATIONARY COORDINATES,(IN/SEC)
     -1.2366D 00  -1.0971D 00  -7.2142D-01  -5.1453D-01  -3.1323D-01  -1.7316D-01
     -1.2043D-01   5.7400D-02   2.5383D-01   3.6812D-01   4.6139D-01   6.3824D-01
      6.7558D-01   7.4629D-01   7.6059D-01   8.2080D-01   1.0569D 00

Y VELOCITY ARRAY IN STATIONARY COORDINATES,(IN/SEC)
     -3.0017D 00  -2.6632D 00  -1.7513D 00  -1.2487D 00  -7.6234D-01  -4.2052D-01
     -2.9264D-01   1.3852D-01   6.1490D-01   8.9205D-01   1.1183D 00   1.5477D 00
      1.6385D 00   1.8085D 00   1.8506D 00   1.9881D 00   2.5602D 00

WHIRL FREQUENCY ARRAY,RPM
      3.2000D 04   3.2000D 04   3.2000D 04   3.2000D 04   3.2001D 04   3.2000D 04
      3.2000D 04   3.2000D 04   3.2000D 04   3.2000D 04   3.2000D 04   3.2000D 04
      3.2000D 04   3.2000D 04   3.2001D 04   3.2000D 04   3.2000D 04

WHIRL TO SPIN FREQUENCY RATIO ARRAY
      1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   9.9999D-01
      9.9999D-01   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00
      1.0000D 00   9.9999D-01   1.0000D 00   9.9999D-01   9.9999D-01
```

TABLE 10

TEST 1139 SIMULATION AT 32,000 rpm

```
X DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
     -2.1879D-03  -1.9412D-03  -1.2765D-03  -9.1032D-04  -5.5573D-04  -3.0770D-04
     -2.1501D-04   9.7638D-05   4.4312D-04   6.4415D-04   8.0830D-04   1.1198D-03
      1.1856D-03   1.3093D-03   1.3390D-03   1.4398D-03   1.8550D-03

Y DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
      1.2669D-03   1.1241D-03   7.3920D-04   5.2711D-04   3.2173D-04   1.7785D-04
      1.2403D-04  -5.7470D-05  -2.5802D-04  -3.7470D-04  -4.6998D-04  -6.5074D-04
     -6.8896D-04  -7.6068D-04  -7.7794D-04  -8.3641D-04  -1.0774D-03

ROTOR DEFLECTION VECTOR ARRAY (IN.)
      2.5282D-03   2.2431D-03   1.4751D-03   1.0519D-03   6.4214D-04   3.5540D-04
      2.4822D-04   1.1330D-04   5.1277D-04   7.4520D-04   9.3500D-04   1.2951D-03
      1.3713D-03   1.5142D-03   1.5486D-03   1.6651D-03   2.1451D-03

PHASE ANGLE ARRAY FOR ROTOR DEFLECTION VECTORS, DEGREES
      1.4993D 02   1.4993D 02   1.4993D 02   1.4993D 02   1.4993D 02   1.4997D 02
      1.5002D 02   3.2952D 02   3.2979D 02   3.2981D 02   3.2982D 02   3.2984D 02
      3.2984D 02   3.2984D 02   3.2984D 02   3.2985D 02   3.2985D 02

X VELOCITY ARRAY IN STATIONARY COORDINATES,(IN/SEC)
     -4.2455D 00  -3.7667D 00  -2.4770D 00  -1.7666D 00  -1.0768D 00  -5.9614D-01
     -4.1573D-01   1.9254D-01   8.6457D-01   1.2556D 00   1.5748D 00   2.1803D 00
      2.3081D 00   2.5498D 00   2.6017D 00   2.8042D 00   3.6118D 00

Y VELOCITY ARRAY IN STATIONARY COORDINATES,(IN/SEC )
     -7.3317D 00  -6.5050D 00  -4.2778D 00  -3.0504D 00  -1.8631D 00  -1.0310D 00
     -7.2045D-01   3.2721D-01   1.4850D 00   2.1586D 00   2.7087D 00   3.7527D 00
      3.9736D 00   4.3869D 00   4.4902D 00   4.8240D 00   6.2150D 00

WHIRL FREQUENCY ARRAY,RPM
      3.2000D 04   3.2000D 04   3.2000D 04   3.2000D 04   3.2001D 04   3.2000D 04
      3.2000D 04   3.2000D 04   3.2000D 04   3.2000D 04   3.2000D 04   3.2000D 04
      3.2000D 04   3.2000D 04   3.2001D 04   3.2000D 04   3.2000D 04

WHIRL TO SPIN FREQUENCY RATIO ARRAY
      1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   9.9999D-01
      1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00
      1.0000D 00   9.9999D-01   1.0000D 00   9.9999D-01   9.9999D-01
```

TABLE 11

TEST 1142 SIMULATION AT 32,000 rpm

```
X DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
   -7.5056D-04  -6.6020D-04  -4.1601D-04  -2.8211D-04  -1.5348D-04  -6.0035D-05
   -2.3716D-05   9.5442D-05   2.2516D-04   2.9994D-04   3.6124D-04   4.7614D-04
    5.0023D-04   5.4497D-04   5.5465D-04   5.9021D-04   7.4478D-04

Y DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
    1.1668D-03   1.0263D-03   6.4671D-04   4.3856D-04   2.3859D-04   9.3327D-05
    3.6866D-05  -1.4837D-04  -3.5002D-04  -4.6628D-04  -5.6157D-04  -7.4018D-04
   -7.7764D-04  -8.4719D-04  -8.6223D-04  -9.1751D-04  -1.1578D-03

ROTOR DEFLECTION VECTOR ARRAY (IN.)
    1.3873D-03   1.2203D-03   7.6896D-04   5.2146D-04   2.8370D-04   1.1097D-04
    4.3837D-05   1.7642D-04   4.1619D-04   5.5442D-04   6.6772D-04   8.8010D-04
    9.2464D-04   1.0073D-03   1.0252D-03   1.0910D-03   1.3767D-03

PHASE ANGLE ARRAY FOR ROTOR DEFLECTION VECTORS, DEGREES
    1.2275D 02   1.2275D 02   1.2275D 02   1.2275D 02   1.2275D 02   1.2275D 02
    1.2275D 02   3.0275D 02   3.0275D 02   3.0275D 02   3.0275D 02   3.0275D 02
    3.0275D 02   3.0275D 02   3.0275D 02   3.0275D 02   3.0275D 02

X VELOCITY ARRAY IN STATIONARY COORDINATES,(IN./SEC.)
   -3.9099D 00  -3.4392D 00  -2.1671D 00  -1.4697D 00  -7.9891D-01  -3.1278D-01
   -1.2359D-01   4.9718D-01   1.1729D 00   1.5625D 00   1.8818D 00   2.4802D 00
    2.6056D 00   2.8391D 00   2.8871D 00   3.0752D 00   3.8806D 00

Y VELOCITY ARRAY IN STATIONARY COORDINATES,(IN./SEC.)
   -2.5152D 00  -2.2124D 00  -1.3941D 00  -9.4523D-01  -5.1536D-01  -2.0110D-01
   -7.9393D-02   3.1986D-01   7.5456D-01   1.0052D 00   1.2106D 00   1.5959D 00
    1.6767D 00   1.8259D 00   1.8623D 00   1.9768D 00   2.4945D 00

WHIRL FREQUENCY ARRAY,RPM
    3.2000D 04   3.2000D 04   3.2000D 04   3.2000D 04   3.2001D 04   3.1999D 04
    3.1999D 04   3.2000D 04   3.2000D 04   3.2000D 04   3.2000D 04   3.2000D 04
    3.2000D 04   3.2000D 04   3.2001D 04   3.2000D 04   3.2000D 04

WHIRL TO SPIN FREQUENCY RATIO ARRAY
    1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   9.9998D-01
    9.9998D-01   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00
    1.0000D 00   9.9999D-01   1.0000D 00   9.9999D-01   9.9999D-01
```

TABLE 12

TEST 1143 SIMULATION AT 32,000 rpm

```
X DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
     -3.0738D-04   -2.7112D-04   -1.7318D-04   -1.1944D-04   -6.7755D-05   -3.0700D-05
     -1.6576D-05    2.9509D-05    7.9637D-05    1.0850D-04    1.3181D-04    1.7536D-04
      1.8448D-04    2.0137D-04    2.0486D-04    2.1706D-04    2.6845D-04

Y DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
      7.2367D-05    6.3830D-05    4.0771D-05    2.8121D-05    1.5952D-05    7.2277D-06
      3.9025D-06   -6.9473D-06   -1.8749D-05   -2.5545D-05   -3.1031D-05   -4.1284D-05
     -4.3433D-05   -4.7408D-05   -4.8231D-05   -5.1102D-05   -6.3201D-05

ROTOR DEFLECTION VECTOR ARRAY (IN.)
      3.1578D-04    2.7853D-04✓   1.7791D-04    1.2271D-04    6.9608D-05    3.1539D-05
      1.7029D-05    3.0316D-05✓   8.1814D-05✓   1.1147D-04    1.3541D-04✓   1.8015D-04
      1.8953D-04✓   2.0687D-04    2.1046D-04    2.2299D-04    2.7578D-04✓

PHASE ANGLE ARRAY FOR ROTOR DEFLECTION VECTORS, DEGREES
      1.6675D 02    1.6675D 02✓   1.6675D 02    1.6675D 02    1.6675D 02    1.6675D 02
      1.6675D 02    3.4675D 02✓   3.4675D 02✓   3.4675D 02    3.4675D 02✓   3.4675D 02
      3.4675D 02✓   3.4675D 02    3.4675D 02    3.4675D 02    3.4675D 02✓

X VELOCITY ARRAY IN STATIONARY COORDINATES,(IN/SEC)
     -2.4249D-01   -2.1388D-01   -1.3660D-01   -9.4266D-02   -5.3219D-02   -2.4235D-02
     -1.3096D-02    2.3274D-02    6.2819D-02    8.5592D-02    1.0397D-01    1.3827D-01
      1.4545D-01    1.5895D-01    1.6080D-01    1.7147D-01    2.1208D-01

Y VELOCITY ARRAY IN STATIONARY COORDINATES,(IN/SEC)
     -1.0300D 00   -9.0853D-01   -5.8032D-01   -4.0025D-01   -2.2711D-01   -1.0287D-01
     -5.5542D-02    9.8887D-02    2.6687D-01    3.6360D-01    4.4169D-01    5.8764D-01
      6.1824D-01    6.7476D-01    6.8671D-01    7.2730D-01    8.9949D-01

WHIRL FREQUENCY ARRAY,RPM
      3.2000D 04    3.2000D 04    3.2000D 04    3.2000D 04    3.2001D 04    3.2000D 04
      3.2000D 04    3.2000D 04    3.2000D 04    3.2000D 04    3.2000D 04    3.2000D 04
      3.2000D 04    3.2000D 04    3.2001D 04    3.2000D 04    3.2000D 04

WHIRL TO SPIN FREQUENCY RATIO ARRAY
      1.0000D 00    1.0000D 00    1.0000D 00    1.0000D 00    1.0000D 00    9.9999D-01
      9.9999D-01    1.0000D 00    1.0000D 00    1.0000D 00    1.0000D 00    1.0000D 00
      1.0000D 00    9.9999D-01    1.0000D 00    9.9999D-01    9.9999D-01
```

TABLE 13

TEST 1144 SIMULATION AT 32.000 RPM

```
X DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
      -5.5082D-04   -4.6778D-04   -2.9749D-04   -2.0408D-04   -1.1427D-04   -4.9597D-05
      -2.4787D-05    5.6315D-05    1.4456D-04    1.9539D-04    2.3644D-04    3.1327D-04
       3.2943D-04    3.5970D-04    3.6639D-04    3.9057D-04    4.9461D-04

Y DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
       7.1080D-04    6.2639D-04    3.9836D-04    2.7328D-04    1.5301D-04    6.6414D-05
       3.5192D-05   -7.5409D-05   -1.9357D-04   -2.6164D-04   -3.1660D-04   -4.1949D-04
      -4.4113D-04   -4.8167D-04   -4.9062D-04   -5.2300D-04   -6.6231D-04

ROTOR DEFLECTION VECTOR ARRAY (IN.)
       8.8715D-04    7.8178D-04    4.9718D-04    3.4108D-04    1.9097D-04    8.2889D-05
       4.1426D-05    9.4116D-05    2.4159D-04    3.2654D-04    3.9514D-04    5.2356D-04
       5.5057D-04    6.0116D-04    6.1233D-04    6.5274D-04    8.2661D-04

PHASE ANGLE ARRAY FOR ROTOR DEFLECTION VECTORS, DEGREES
       1.2675D 02    1.2675D 02    1.2675D 02    1.2675D 02    1.2675D 02    1.2675D 02
       1.2675D 02    3.0675D 02    3.0675D 02    3.0675D 02    3.0675D 02    3.0675D 02
       3.0675D 02    3.0675D 02    3.0675D 02    3.0675D 02    3.0675D 02

X VELOCITY ARRAY IN STATIONARY COORDINATES,(IN/SEC)
      -2.3815D 00   -2.0990D 00   -1.3349D 00   -9.1583D-01   -5.1233D-01   -2.2258D-01
      -1.1120D-01    2.5269D-01    6.4864D-01    8.7673D-01    1.0609D 00    1.4056D 00
       1.4781D 00    1.6142D 00    1.6426D 00    1.7530D 00    2.2199D 00

Y VELOCITY ARRAY IN STATIONARY COORDINATES,(IN/SEC)
      -1.7788D 00   -1.5676D 00   -9.9695D-01   -6.8381D-01   -3.8351D-01   -1.6616D-01
      -8.3017D-02    1.8873D-01    4.8444D-01    6.5477D-01    7.9235D-01    1.0500D 00
       1.1042D 00    1.2052D 00    1.2299D 00    1.3082D 00    1.6567D 00

WHIRL FREQUENCY ARRAY,RPM
       3.2000D 04    3.2000D 04    3.2000D 04    3.2000D 04    3.2001D 04    3.2000D 04
       3.2000D 04    3.2000D 04    3.2000D 04    3.2000D 04    3.2000D 04    3.2000D 04
       3.2000D 04    3.2000D 04    3.2001D 04    3.2000D 04    3.2000D 04

WHIRL TO SPIN FREQUENCY RATIO ARRAY
       1.0000D 00    1.0000D 00    1.0000D 00    1.0000D 00    1.0000D 00    9.9999D-01
       9.9998D-01    1.0000D 00    1.0000D 00    1.0000D 00    1.0000D 00    1.0000D 00
       1.0000D 00    9.9999D-01    1.0000D 00    9.9999D-01    9.9999D-01
```

TABLE 14

TABLE 1145 // SIMULATION AT 26,000 RPM

```
X DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
     -1.2147D-04  -1.1321D-04  -9.1015D-05  -7.8748D-05  -6.6757D-05  -6.1523D-05
     -6.1292D-05  -5.6349D-05  -4.9176D-05  -4.4338D-05  -3.9749D-05  -2.9490D-05
     -2.7140D-05  -2.2215D-05  -1.9795D-05  -1.4632D-05  -2.3146D-06

Y DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
      2.4444D-04   2.2783D-04   1.8316D-04   1.5847D-04   1.3434D-04   1.2381D-04
      1.2335D-04   1.1340D-04   9.8963D-05   8.9226D-05   7.9992D-05   5.9347D-05
      5.4617D-05   4.4706D-05   3.9835D-05   2.9447D-05   4.6579D-06

ROTOR DEFLECTION VECTOR ARRAY (IN.)
      2.7296D-04   2.5440D-04   2.0453D-04   1.7696D-04   1.5002D-04   1.3825D-04
      1.3775D-04   1.2663D-04   1.1051D-04   9.9634D-05   8.9324D-05   6.6270D-05
      6.0988D-05   4.9921D-05   4.4482D-05   3.2882D-05   5.2013D-06

PHASE ANGLE ARRAY FOR ROTOR DEFLECTION VECTORS, DEGREES
      1.1642D 02   1.1642D 02   1.1642D 02   1.1642D 02   1.1642D 02   1.1642D 02
      1.1642D 02   1.1642D 02   1.1642D 02   1.1642D 02   1.1642D 02   1.1642D 02
      1.1642D 02   1.1642D 02   1.1642D 02   1.1642D 02   1.1642D 02

X VELOCITY ARRAY IN STATIONARY COORDINATES, (IN/SEC)
     -6.6555D-01  -6.2030D-01  -4.9869D-01  -4.3148D-01  -3.6577D-01  -3.3709D-01
     -3.3583D-01  -3.0875D-01  -2.6945D-01  -2.4294D-01  -2.1779D-01  -1.6158D-01
     -1.4870D-01  -1.2172D-01  -1.0845D-01  -8.0175D-02  -1.2684D-02

Y VELOCITY ARRAY IN STATIONARY COORDINATES,(IN./SEC)
     -3.3072D-01  -3.0824D-01  -2.4781D-01  -2.1441D-01  -1.8177D-01  -1.6754D-01
     -1.6688D-01  -1.5342D-01  -1.3389D-01  -1.2072D-01  -1.0823D-01  -8.0295D-02
     -7.3896D-02  -6.0484D-02  -5.3910D-02  -3.9839D-02  -6.2989D-03

WHIRL FREQUENCY ARRAY,RPM
      2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04
      2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04
      2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04   2.6000D 04

WHIRL TO SPIN FREQUENCY RATIO ARRAY
      1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00
      1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00
      1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   9.9999D-01
```

CONCLUSIONS AND RECOMMENDATIONS

CONCLUSIONS

The Rotordynamics Computer program, as developed under this contract, can serve as a useful tool in predicting the dynamic performance of a rotor-bearing system under various operating conditions. The drive and dissipation torque effects on an accelerating or decelerating rotor can be analyzed with this program.

In addition, the causes and effects of non-synchronous whirl rotor motion may also be studied by applying appropriate out-of-phase stiffness and positive or negative damping excitations. The computer program can be used as an analytical study tool for a general transient spin-speed and/or non-axisymmetric rotor motion.

The computer results of the simulation did not show good agreement, in general, with the test data. Some of the causes for the poor correlation may have been due to unknown characteristics not included in the simulation model. Some of these characteristics are:

1. unknown damping coefficient associated with bearing,
2. mount damping,
3. test mount pedestal stiffness and mass considerations,
4. bearing non-linear description and
5. internal hysteretic damping of the rotor.

In addition, the original intent to iterate the model was not funded; therefore, the time and funds to accomplish improved correlation between the computer results and the experimental data was not available. These items are discussed in the recommendations.

RECOMMENDATIONS

The period of the current contract provided adequate time to update the computer program and to obtain verification. An immediate follow-on effort is needed to study and correlate the integration process with the basic mathematical formulation such that a thorough understanding may be achieved. This would lead to possible improvement in computation-to-real time ratio. Other items needed to be included are the transverse dissipation, the out-of-phase stiffness and damping functions, and the non-linear stiffness bearing in the "STARUP" subroutine.

In addition, an iteration or parametric study is required to improve the Mark-25 model so that a better degree of correlation may be achieved between the simulation results and experimental data.

Recommendations of continued future efforts to improve capability of the computer program are divided into 3 groups, with group I representing the most important items, then group II, and then group III with the least important items.

Group I:

a. In depth study of the integrations between the basic mathematical formulation and integration process in an attempt to improve computation-to-real time ratio.

b. Up-date the "STARUP" subroutine to include other rotordynamics parameters such as:
    1. Rotor drive torque and dissipation functions.
    2. Rotor in-phase damping and in-phase and out-of-phase stiffness and damping parameters.

3. Non-linear bearing stiffness characteristics.

Group II:

a. Inclusion of the Bearing Mass Parameter. The bearing mass will be simulated by a concentrated mass located between a bearing and its mount. For a bearing with soft mount design and particularly in high speed operation, the bearing mass effects should be considered. In addition to the bearing mass parameter, the bearing transverse mass moment of inertia also affects the rotordynamic performance. For normal small operating bearing misalignment, the effects of the transverse mass moment of inertia parameter are relatively small and its inclusion in the rotordynamic study would be optional.

b. Formulation of Hydrodynamic Force Excitation Functions. Particular emphases are placed on the axial pump impeller to stator interactions. A linear hydrodynamic force characteristics should be sufficient at this stage of the program development. These force characteristics will include in-phase and out-of-phase excitations which may also be functions of spin and/or whirl frequencies. The hydrodynamic excitations of the close-clearance flow-control surfaces of the balance piston such as that used in the Mark-25 axial pumps were found to be small compared to other rotordynamic loads and hence they will not be included.

c. Feasibility Study of Advanced Rotating-Coordinate Systems. A feasibility study is suggested to investigate the practicality of using advanced rotating coordinates to facilitate the integration process by smoothing out the second order rotordynamic function fluctuations. The second order function fluctuations may result from cyclic force variations of the rotor mass unbalance in a non-synchronous whirl rotor motion or from non-isotropic bearing mount force characteristics.

Group III:

Inclusion of the below listed parameters as discussed:

a. Rotor Casing Mass and Transverse Mass Moment of Inertia, and the Casing Support Stiffness and Damping Parameter. In these suggested parameters, a rigid casing is assumed. The casing is considered to be supported on an inertial frame of reference (foundation) through elastic and damping members. The stiffness and damping parameters include both force and moment functions of displacements and misalignments respectively.

b. Rotor Material Hysteresis. This parameter includes also the effects of press-fit and/or bolted joints in an elastic rotor. This hysteresis parameter will include both in-phase and out-of-phase damping characteristics.

c. Transverse Sinusoidal Ambient Vibrations. A fixed-amplitude and constant-frequency sinusoidal vibration is considered to be imposed on an otherwise fixed foundation. It appears that for a rotor-bearing system operating under a high-frequency ambient vibration condition, the computer time may become excessive unless only the steady-state rotordynamic performance is studied. An approximate treatment in including the ambient vibration parameter is to apply an equivalent cyclic G-loading directly to the rotor. This treatment would substantially reduce the complexity of the formulation required for the inclusion of the ambient vibration parameter.

APPENDIX A

```
                  A 5-STATION 5-MASS TEST  ROTOR MODEL, BEARINGS LOCATED AT STATIONS 1 AND
                  5                                     LOS  ANGELES,CALIF.APRIL/4/1970


NS       =   5              NUMBER OF ROTOR STATIONS (ALLOWABLE RANGE:5=<NS=<25)
IASIGN   =   1              A ROTOR STATION NUMBER AT WHICH THE WHIRL/SPIN FREQUENCY RATIO WILL BE
                                 PLOTTED ON CRT
NOORPM   =   1              THE NUMBER OF SPIN SPEEDS IN RPM AT OR NEAR WHICH 3-DIMENSIONAL ABSOLUTE ROTOR
                                 MODE SHAPE CRT GRAPHS ARE REQUIRED.   THE SPIN SPEED RPM VALUES ARE LISTED
                                 UNDER INPRPM ARRAY.
                                 (ALLOWABLE RANGE: 0=<NOORPM=<50)
NPOINT = 25                 THE NUMBER OF POINTS (ONE PER EACH INTEGRATION STEP) FOR EACH CRT GRAPH.
                                 (ALLOWABLE RANGE: 1=<NPOINT=<50)
CRT      =   0              CRT=0 MEANS CRT IS NOT REQUIRED
                            CRT=1 MEANS CRT IS REQUIRED
ACCEL    =   0              ACCEL=0 MEANS A 3-DIMENSIONAL ROTOR MODE SHAPE CRT CORRESPONDING TO THAT AT BEG-
                                 INNING  OF THE RUN WOULD BE PROVIDED IF CONCURRENTLY CRT=1.
                                 ACCEL=1 MEANS ONLY THE TRANSIENT-SPEED ROTOR MODE SHAPES AT OR NEAR INPRPM
                                VALUES WILL BE PROVIDED IF CONCURRENTLY CRT = 1.
INPRPM ARRAY                THE ROTOR SPIN SPEED RPM VALUES AT OR NEAR WHICH CRT GRAPHS FOR
                            3-DIMENSIONAL ABSOLUTE ROTOR MODE SHAPES ARE REQUIRED
      0.0


T        =    0.0           INITIAL REAL TIME, SEC.
DT       =    1.0000D-05    ESTIMATED INITIAL INTEGRATION STEP (REAL)TIME, SEC.
TMAX     =    2.0000D-02    TOTAL REAL TIME TO BE RUN, SEC.
TOLI     =    1.0000D-02    INTEGRATION TOLERANCE, FRACTION
TOLB     =    1.0000D-03    TOLERANCE IN COMPUTING BEARING DISPLACEMENTS, FRACTION
TSTOP    =    5.0000D-01    THE COMPUTER TIME ALLOWED FOR EACH SET OF DATA, MINUTES.


GX       =    0.0           GRAVITY OR G-LOADING IN X DIRECTION, IN/SEC**2
GY       =    0.0           GRAVITY OR G-LOADING IN Y DIRECTION, IN/SEC**2
TMZ      =    0.0           EXPONENT FOR SPEED SENSITIVE ROTOR DRIVE TORQUE
TMZ1     =    0.0           COEFFICIENT FOR ROTOR DRIVE TORQUE
TMZ2     =    0.0           COEFFICIENT FOR ROTOR DRIVE TORQUE, (IN-LB-SEC)/RAD.
TMZ3     =    0.0           COEFFICIENT FOR ROTOR DRIVE TORQUE,  IN-LB
```

```
DD ARRAY          OUTSIDE DIAMETERS OF ROTOR SECTIONS BETWEEN ADJACENT ROTOR STATIONS, IN.
            1.0000D 01   1.0000D 01   1.0000D 01   1.0000D 01

D ARRAY           INSIDE DIAMETERS OF ROTOR SECTIONS BETWEEN ADJACENT ROTOR STATIONS, IN.
            0.0          0.0          0.0          0.0

QL ARRAY          ROTOR SECTION LENGTHS BETWEEN ADJACENT ROTOR STATIONS, IN.
            1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00

DN ARRAY          MATERIAL DENSITIES OF ROTOR SECTIONS BETWEEN ADJACENT STATIONS, LB/IN**3
            3.0000D-01   3.0000D-01   3.0000D-01   3.0000D-01

EE ARRAY          YOUNGS MODULI OF ROTOR SECTIONS BETWEEN ADJACENT STATIONS, LB/IN**2
            0.0          0.0          0.0          0.0

GG ARRAY          SHEAR MODULI OF ROTOR SECTIONS BETWEEN ADJACENT STATIONS, LB/IN**2
            1.1500D 07   1.1500D 07   1.1500D 07   1.1500D 07

EI  ARRAY         DIRECT INPUT OF THE PRODUCTS OF YOUNGS MODULI AND AREA MOMENTS OF INERTIA OF ROTOR
                  SECTIONS BETWEEN ADJACENT STATIONS, LB-IN**2
            1.6667D 04   1.6667D 04   1.6667D 04   1.6667D 04

GAK ARRAY         DIRECT INPUT OF THE PRODUCTS OF SHEAR MODULI, CROSS-SECTIONAL AREAS
                  AND RECIPROCALS OF SHEAR STRESS CONCENTRATION FACTORS BETWEEN ADJACENT STATIONS, LB.
            0.0          0.0          0.0          0.0

AM ARRAY          ADDITIONAL ROTOR MASSES AT ROTOR STATIONS, (LB-SEC**2)/IN.
     0.0          0.0          0.0          0.0          0.0

AID ARRAY         ADDITIONAL ROTOR TRANSVERSE MASS MOMENTS OF INERTIA AT ROTOR STATIONS, (LB-IN-SEC**2)
     0.0          0.0          0.0          0.0          0.0

AIRO ARRAY        ADDITIONAL ROTOR POLAR MASS MOMENTS OF INERTIA AT ROTOR STATIONS, (LB-IN-SEC**2)
     1.0000D-16   1.0000D-16   1.0000D-16   1.0000D-16   1.0000D-16

ECC ARRAY         ROTOR MASS ECCENTRICITIES AT ROTOR STATIONS, IN.
     1.0000D-03   1.0000D-03   1.0000D-03   1.0000D-03   1.0000D-03

ALFA ARRAY        PHASE ANGLES FOR ROTOR MASS ECCENTRICITY VECTORS AT ROTOR STATIONS MEASURED FROM THE
                  INITIAL ROTOR SPIN ANGULAR POSITION, DEGREES
     4.5000D 01   4.5000D 01   4.5000D 01   4.5000D 01   4.5000D 01
```

```
BETA ARRAY          INITIAL MISALIGNMENTS BETWEEN THE AXES OF THE MASS MOMENTS OF INERTIA
                    AND THE ELASTIC AXES AT ROTOR STATIONS, DEGREES
    0.0             0.0           0.0           0.0           0.0

GAMMA ARRAY         ANGULAR POSITIONS OF THE X-Y PLANE PROJECTIONS OF THE AXES OF MASS MOMENTS OF
                    INERTIA AT ROTOR STATIONS MEASURED FROM THAT AT THE FIRST ROTOR STATION, DEGREES
    0.0             0.0           0.0           0.0           0.0

CZ ARRAY            TORSONAL FRICTION EXPONENTS AT ROTOR STATIONS, DIMENSIONLESS
    0.0             0.0           0.0           0.0           0.0

CZ1 ARRAY           TORSIONAL FRICTION COEFFICIENTS AT ROTOR STATIONS, DIMENSION OF CZ1(I)*FDOT**CZ(I)
                    IS IN-LB.
    0.0             0.0           0.0           0.0           0.0

CZ2 ARRAY           TORSIONAL FRICTION COEFFICIENTS AT ROTOR STATIONS, (IN-LB-SEC)/RAD.
    0.0             0.0           0.0           0.0           0.0

XKF ARRAY           WHIRL-FREQUENCY FACTORS FOR STIFFNESS FORCE COEFFICIENTS AT ROTOR STATIONS,
                    DIMENSIONLESS
    0.0             0.0           0.0           0.0           0.0

XCF ARRAY           WHIRL-FREQUENCY FACTORS FOR DAMPING FORCE COEFFICIENTS AT ROTOR STATIONS,
                    DIMENSIONLESS
    0.0             0.0           0.0           0.0           0.0

XKFF ARRAY          WHIRL-FREQUENCY FACTORS FOR STIFFNESS MOMENT COEFFICIENTS AT ROTOR STATIONS,
                    DIMENSIONLESS
    0.0             0.0           0.0           0.0           0.0

XCFF ARRAY          WHIRL-FREQUENCY FACTORS FOR DAMPING MOMENT COEFFICIENTS AT ROTOR STATIONS,
                    DIMENSIONLESS
    0.0             0.0           0.0           0.0           0.0

QK ARRAY            IN-PHASE STIFFNESS FORCE COEFFICIENTS AT ROTOR STATIONS, LB/IN.
    0.0             0.0           0.0           0.0           0.0

QC ARRAY            IN-PHASE DAMPING FORCE COEFFICIENTS AT ROTOR STATIONS, (LB-SEC)/IN.
    0.0             0.0           0.0           0.0           0.0
```

```
QKP ARRAY          OUT-OF-PHASE STIFFNESS FORCE COEFFICIENTS AT ROTOR STATIONS, LB/IN.
     0.0           0.0          0.0          0.0          0.0

QCP ARRAY          OUT-OF-PHASE DAMPING FORCE COEFFICIENTS AT ROTOR STATIONS, (LB-SEC)/IN.
     0.0           0.0          0.0          0.0          0.0

QKHD ARRAY         OUT-OF-PHASE WHIRL AND SPIN VELOCITY SENSITIVE STIFFNESS FORCE COEFFICIENTS AT ROTOR
                   STATIONS,(LB-SEC)/(IN-RAD)
     0.0           0.0          0.0          0.0          0.0

QCHD ARRAY         OUT-OF-PHASE WHIRL AND SPIN VELOCITY SENSITIVE DAMPING FORCE COEFFICIENTS AT ROTOR
                   STATIONS,(LB-SEC**2)/(IN-RAD)
     0.0           0.0          0.0          0.0          0.0

QKF ARRAY          IN-PHASE STIFFNESS MOMENT COEFFICIENTS AT ROTOR STATIONS, IN-LB.
     0.0           0.0          0.0          0.0          0.0

QCF ARRAY          IN-PHASE DAMPING MOMENT COEFFICIENTS AT ROTOR STATIONS, IN-LB-SEC.
     0.0           0.0          0.0          0.0          0.0

QKPF ARRAY         OUT-OF-PHASE STIFFNESS MOMENT COEFFICIENTS AT ROTOR STATIONS, IN-LB.
     0.0           0.0          0.0          0.0          0.0

QCPF ARRAY         OUT-OF-PHASE DAMPING MOMENT COEFFICIENTS AT ROTOR STATIONS, IN-LB-SEC.
     0.0           0.0          0.0          0.0          0.0

QKHDF ARRAY        OUT-OF-PHASE WHIRL AND SPIN VELOCITY SENSITIVE STIFFNESS MOMENT COEFFICIENTS AT
                   ROTOR STATIONS, (LB-IN-SEC)/RAD.
     0.0           0.0          0.0          0.0          0.0

QCHDF ARRAY        OUT-OF-PHASE WHIRL AND SPIN VELOCITY SENSITIVE DAMPING MOMENT COEFFICIENTS AT
                   ROTOR STATIONS, (LB-IN-SEC**2)/RAD.
     0.0           0.0          0.0          0.0          0.0
```

```
NB        =      2    NUMBER OF NON-LINEAR STIFFNESS BEARINGS. (ALLOWABLE RANGE: 2=<NB=<12)

IB ARRAY              ROTOR STATION NUMBERS FOR NON-LINEAR STIFFNESS BEARINGS
    1           5

K ARRAY               TOTAL NUMBER OF STIFFNESS SECTIONS FOR EACH OF THE NON-LINEAR STIFFNESS BEARINGS.
                      (ALLOWABLE RANGE: 1=<K=<4)
    1           1

BKMX ARRAY            NON-ISOTROPIC MOUNT STIFFNESS COEFFICIENTS IN X-DIRECTION FOR
                      NON-LINEAR STIFFNESS BEARINGS, LB/IN.
      8.0000D 04      8.0000D 04

BKMY ARRAY            NON-ISOTROPIC MOUNT STIFFNESS COEFFICIENTS IN Y-DIRECTION FOR
                      NON-LINEAR STIFFNESS BEARINGS, LB/IN.
      8.0000D 04      8.0000D 04

BCMX ARRAY            NON-ISOTROPIC MOUNT DAMPING COEFFICIENTS IN X-DIRECTION FOR
                      NON-LINEAR STIFFNESS BEARINGS,(LB-SEC)/IN.
      1.0000D-16      1.0000D-16

BCMY ARRAY            NON-ISOTROPIC MOUNT DAMPING COEFFICIENTS IN Y-DIRECTION FOR
                      NON-LINEAR STIFFNESS BEARINGS, (LB-SEC)/IN.
      1.0000D-16      1.0000D-16

BCB ARRAY             BEARING DAMPING COEFFICIENTS FOR NON-LINEAR STIFFNESS BEARINGS, (LB-SEC)/IN.
      1.0000D-16      1.0000D-16
```

```
                                   NONLINEAR BEARING SPECIFICATIONS

BNB ARRAY                ROTOR SPIN-SPEED SENSITIVE BEARING STIFFNESS COEFFICIENTS FOR K STIFFNESS
                         SECTIONS OF NON-LINEAR STIFFNESS BEARING  NUMBER  1
     0.0

BNB ARRAY                ROTOR SPIN-SPEED SENSITIVE BEARING STIFFNESS COEFFICIENTS FOR K STIFFNESS
                         SECTIONS OF NON-LINEAR STIFFNESS BEARING  NUMBER  2
     0.0

BBB ARRAY                ROTOR SPIN-SPEED SENSITIVE BEARING STIFFNESS COEFFICIENTS FOR K STIFFNESS
                         SECTIONS OF BEARINGS FOR NON-LINEAR STIFFNESS BEARING NUMBER  1
     0.0

BBB ARRAY                ROTOR SPIN-SPEED SENSITIVE BEARING STIFFNESS COEFFICIENTS FOR K STIFFNESS
                         SECTIONS OF BEARINGS FOR NON-LINEAR STIFFNESS BEARING NUMBER  2
     0.0

BROB ARRAY               UPPER BEARING-DISPLACEMENT LIMITS FOR K STIFFNESS SECTIONS OF BEARING
                         NUMBER  1, IN.
     5.0000D-03

BROB ARRAY               UPPER BEARING-DISPLACEMENT LIMITS FOR K STIFFNESS SECTIONS OF BEARING
                         NUMBER  2, IN.
     5.0000D-03

BKB ARRAY                NON-LINEAR BEARING STIFFNESS COEFFICIENTS FOR K STIFFNESS
                         SECTIONS OF BEARING
                         NUMBER  1
     8.0000D 04

BKB ARRAY                NON-LINEAR BEARING STIFFNESS COEFFICIENTS FOR K STIFFNESS
                         SECTIONS OF BEARING
                         NUMBER  2
     8.0000D 04

BHB ARRAY                NON-LINEAR BEARING STIFFNESS EXPONENTS FOR K STIFFNESS SECTIONS OF BEARING
                         NUMBER  1
     1.0000D 00
```

```
BHB ARRAY           NON-LINEAR BEARING STIFFNESS EXPONENTS FOR K STIFFNESS SECTIONS OF BEARING
                    NUMBER  2
    1.0000D 00

BDB ARRAY           NON-LINEAR BEARING STIFFNESS COEFFICIENTS FOR K STIFFNESS SECTIONS OF BEARING
                    NUMBER  1
    0.0

BDB ARRAY           NON-LINEAR BEARING STIFFNESS COEFFICIENTS FOR K STIFFNESS SECTIONS OF BEARING
                    NUMBER  2
    0.0

BEB ARRAY           NON-LINEAR BEARING STIFFNESS COEFFICIENTS FOR K STIFFNESS SECTIONS OF BEARING
                    NUMBER  1
    0.0

BEB ARRAY           NON-LINEAR BEARING STIFFNESS COEFFICIENTS FOR K STIFFNESS SECTIONS OF BEARING
                    NUMBER  2
    0.0
```

```
                                  INITIAL ROTOR MOTION SPECIFICATIONS


ICOND  =  1          ICOND=1 MEANS STARTING A NEW ROTOR-BEARING CONFIGURATION
                     ICOND=0 MEANS READ-IN ALTERNATE INITIAL CONDITIONS FOR THE SAME
                             ROTOR-BEARING CONFIGURATION

CONTIN =  0          CONTIN=0 MEANS STARTING A NEW ROTORDYNAMICS ANALYSIS WITH INITIAL CONDTIONS
                             PROVIDED BY THE STARUP SUBROUTINE
                     CONTIN=1 MEANS CONTINUITION OF A PREVIOUS ANALYSIS BY USING THE PREVIOUS
                             RESULTS ON PUNCHED CARDS AS THE INITIAL CONDTIONS

FD       =           0.0             ROTOR SPIN ANGULAR DISPLACEMENT COORDINATE, DEGREES.
FDOT     =           1.00000D 02     ROTOR SPIN FREQUENCY, RPM.
WHIVEL   =           1.00000D 02     ROTOR WHIRL FREQUENCY,RPM.
FDOFIX   =           0.0             A BEARING STIFFNESS SPEED SENSITIVE PARAMETER,  RPM


    .....THE END OF INPUT DATA.....



      TOTAL ROTOR WEIGHT,LB =    9.4248D 01
      TOTAL ROTOR MASS,(LB*SEC**2)/IN =   2.4411D-01
      TOTAL ROTOR POLAR MASS MOMENT OF INERTIA, LB*IN*SEC**2  =   3.0514D 00
```

```
     FORCE INFLUENCE COEFFICIENTS

  ROW   1
0.0            0.0           0.0           0.0           0.0

  ROW   2
0.0            4.500110D-05  5.500073D-05  3.500036D-05  0.0

  ROW   3
0.0            5.500073D-05  8.000146D-05  5.500073D-05  0.0

  ROW   4
0.0            3.500036D-05  5.500073D-05  4.500110D-05  0.0

  ROW   5
0.0            0.0           0.0           0.0           0.0
```

THE TABULATED FORCE INFLUENCE COEFFICIENTS ABOVE ARE THOSE WITH RESPECT TO THE STRAIGHT LINE CONNECTING THE FIRST AND LAST JOURNAL CENTERS OF THE NON-LINEAR STIFFNESS BEARINGS. THE ROW NUMBER REPRESENTS THE ROTOR STATION NUMBER WHERE A UNIT TRANSVERSE LOAD OF 1 LB IS APPLIED, AND THE COLUMN NUMBER NOT SHOWN ABOVE, DENOTES THE ROTOR STATION NUMBER WHERE THE RESULTING DEFLECTIONS IN INCHES ARE DESCRIBED. SINCE THE FORCE INFLUENCE COEFFICIENTS ARE SYMMETRIC ABOUT THE DIAGONALS, ROW AND COLUMN NUMBERS MAY BE INTERCHANGED.

```
       MOMENT INFLUENCE COEFFICIENTS

  ROW   1
0.0           5.249999D-05   5.999999D-05   3.749999D-05   0.0

  ROW   2
0.0           2.999999D-05   4.499999D-05   2.999999D-05   0.0

  ROW   3
0.0          -7.499999D-06   3.388132D-21   7.499999D-06   0.0

  ROW   4
0.0          -2.999999D-05  -4.499999D-05  -2.999999D-05   0.0

  ROW   5
0.0          -3.749999D-05  -5.999999D-05  -5.249999D-05   0.0
```

THE SIGNIFICANCE OF THE TABULATED MOMENT INFLUENCE COEFFICIENTS CAN BE SIMILARLY INTERPRETED AS THAT OF THE FORCE INFLUENCE COEFFICIENTS, EXCEPT THAT A UNIT MOMENT (1 IN-LB) APPLICATION IS USED INSTEAD OF A FORCE APPLICATION. THE COLUMN NUMBER REPRESENTS THE ROTOR STATION NUMBER WHERE THE RESULTING DEFLECTIONS IN INCHES ARE DESCRIBED. SINCE THE MOMENT INFLUENCE COEFFICIENTS ARE NO LONGER SYMMETRIC ABOUT A DIAGONAL, INTERCHANGE OF THE COLUMN AND ROW IS NOT PERMISSIBLE.

```
ELEMENTS IN ROTATING COORDINATES:
      3.3499D-07   1.2365D-06   1.6037D-06   1.2365D-06   3.3499D-07   7.8540D-01
      7.8540D-01   7.8540D-01   7.8540D-01   7.8540D-01   1.1302D-22   0.0
      1.2592D-22   0.0          1.5071D-22   1.0472D 01   1.0472D 01   1.0472D 01
      1.0472D 01   1.0472D 01   0.0          1.0472D 01
```

THE ROTATING ELEMENTS SHOWN ABOVE ARE THOSE UNDER THE INITIAL TIME CONDITIONS, i.e., AT A STARTING TIME.

THE FIRST NS ELEMENTS IN ROTATING COORDINATES REPRESENT THE ROTOR RADIAL DISPLACEMENTS AT STATIONS 1 THROUGH NS WITH RESPECT TO RESPECTIVE STATIC ROTOR DEFLECTION CENTERS IN INCHES. THE STATIC ROTOR DEFLECTION CENTERS ARE THE ROTOR POSITIONS UNDER ZERO SPIN SPEED CONDITIONS BUT INCLUDING THE EFFECTS OF g_x AND g_y.

THE SECOND NS ELEMENTS DENOTE THE ABSOLUTE ANGULAR DISPLACEMENTS OF THE RADIAL DISPLACEMENT VECTORS IN RADIANS.

THE THIRD NS ELEMENTS REPRESENT THE RADIAL VELOCITIES OF THE RADIAL DISPLACEMENT VECTORS IN INCHES PER SECOND.

THE FOURTH NS ELEMENTS DENOTE THE ANGULAR VELOCITIES OF THE DISPLACEMENT VECTORS IN RADIANS PER SECOND.

THE SECOND LAST ELEMENT REPRESENTS THE ROTOR SPIN ANGULAR DISPLACEMENT IN RADIANS, AND THE LAST ELEMENT DENOTES THE ROTOR SPIN VELOCITY IN RADIANS PER SECOND.

```
          THE APPROPRIATE DERIVATIVES BD,S

 1.130344D-22   0.0            1.259229D-22   0.0            1.507126D-22   1.047198D 01   1.047198D 01
 1.047199D 01   1.047198D 01   1.047198D 01  -2.458659D-13   4.614597D-14   1.538291D-13   4.615359D-14
-2.458619D-13   1.465407D-11   6.001429D-11  -5.327994D-11   5.968109D-11   3.114569D-11   1.047198D 01
-1.406093D-21
```

THE BD'S LISTED ABOVE ARE THE CORRESPONDING TIME (SECOND) DERIVATIVES OF THE PREVIOUSLY LISTED ELEMENTS IN ROTATING COORDINATES. THESE BD'S ARE EVALUATED AT THE LAST SPECIFIED REAL TIME.

```
                                      AT THE TIME:  T= 1.925625D-03  SEC.

 X DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
      2.3205D-07   8.5652D-07   1.1109D-06   8.5652D-07   2.3205D-07

 Y DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(IN.)
      2.4160D-07   8.9178D-07   1.1567D-06   8.9178D-07   2.4160D-07

 ROTOR DEFLECTION VECTOR ARRAY (IN.)
      3.3499D-07   1.2365D-06   1.6037D-06   1.2365D-06   3.3499D-07

 PHASE ANGLE ARRAY FOR ROTOR DEFLECTION VECTORS, DEGREES
      4.6155D 01   4.6155D 01   4.6155D 01   4.6155D 01   4.6155D 01

 X VELOCITY ARRAY IN STATIONARY COORDINATES,(IN.)
     -2.5300D-06  -9.3387D-06  -1.2112D-05  -9.3387D-06  -2.5300D-06

 Y VELOCITY ARRAY IN STATIONARY COORDINATES,(IN.)
      2.4300D-06   8.9695D-06   1.1634D-05   8.9695D-06   2.4300D-06

 WHIRL FREQUENCY ARRAY,RPM
      1.0000D 02   1.0000D 02   1.0000D 02   1.0000D 02   1.0000D 02

 WHIRL TO SPIN FREQUENCY RATIO ARRAY
      1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00   1.0000D 00


 TOTAL NO. OF REVOLUTIONS =   3.2094D-03
 SPIN SPEED, RPM          =   1.0000D 02
```

```
XB ARRAY            BEARING DISPLACEMENT COORDINATE, IN.
      1.1602D-07    1.1602D-07

YB ARRAY            BEARING DISPLACEMENT COORDINATE, IN.
      1.2080D-07    1.2080D-07

XBDOT ARRAY         BEARING VELOCITY COORDINATE, IN/SEC.
     -1.2650D-06   -1.2650D-06

YBDOT ARRAY         BEARING VELOCITY COORDINATE, IN/SEC.
      1.2150D-06    1.2150D-06


BRGPO ARRAY         JOURNAL DISPLACEMENT FROM BEARING-CENTER ARRAY, IN.
      1.6749D-07    1.6749D-07

PHABRO ARRAY         JOURNAL DISPLACEMENT VECTOR PHASE ANGLE COUNTER-CLOCKWISE FROM X-AXIS, DEGREES
      4.6155D 01    4.6155D 01

BRGFOR ARRAY        BEARING REACTION ARRAY, LB.
      1.3400D-02    1.3400D-02

PHAFOR ARRAY        BEARING REACTION VECTOR PHASE ANGLE COUNTER-CLOCKWISE FROM X-AXIS, DEGREES
      4.6155D 01    4.6155D 01

STIFF ARRAY         BEARING FORCE TO JOURNAL DEFLECTION RATIO, OR EQUIVALENT LINEAR BEARING
                     STIFFNESS, LB/IN/BEARING
      8.0000D 04    8.0000D 04



          THE APPROPRIATE DERIVATIVES  BD,S

-3.401671D-16   1.813953D-16   3.897550D-16   1.814151D-16  -3.401572D-16   1.047198D 01   1.047198D 01
 1.047198D 01   1.047198D 01   1.047198D 01  -7.113311D-14   1.673471D-13   2.748897D-13   1.673572D-13
-7.112303D-14  -3.799449D-09  -1.679891D-09  -1.629238D-09  -1.678542D-09  -3.766101D-09   1.047198D 01
-1.211355D-20
```

APPENDIX B

COMPUTER PROGRAM LISTINGS

A complete set of the computer program Fortran Listing is attached. This includes also three library subroutines "ISIMDD," "TIMEV" and COUNTV" which may not be available in all the computation facilities. Subroutine CONTV is a part of TIMEV subroutine.

The main program and subroutine names and corresponding memory length required are:

NAME	*MEMORY REQUIRED IN HEXADECIMAL BYTES
MAIN	2EC2
READIN	D34
WRI0UT	38AC
STARUP	1581A
FUND	2EEA
BRGXY	9EE
INFLC0	B12
SZMASS	5F8
PL0T1	29A
RKADAM	3CC3
RUNKUT	6DE
ADAMLT	D2C
TIME	20C
ISIMDD	7D6
TIMEV	254

*The memory capacity required for "MAIN""WRIOUT" and "PL0T1" may slightly deviate from the exact values resulting from changing to NASA CRT Subroutines and minor modifications after memory capacity counts.

```
C      A SIMPLIFIED FLEXIBLE ROTOR DYNAMICS COMPUTER PROGRAM WRITTEN FOR 00768000
C     NASA LEWIS RESEARCH CENTER, CLEVELAND, OHIO, BY FRED. A. SHEN,     00769000
C      POWER SYSTEMS DIVISIONS, NORTH AMERICAN ROCKWELL CORP.,LOS ANGELES00770000
C        CALIFORNIA, MAY 1970.                                           00771000
      IMPLICIT REAL*8 (A-H,O-Z)                                          00772000
      INTEGER CRT,ACCEL,CONTIN                                           00773000
      REAL TITLE,INPRPM,Z,ROO,TT,RPM,WHRATI,FORC,BRGR,ROMAX,ROSTA        00774000
     1,RSTAR4,RSTAR5,RSTAR6                                              00775000
        DIMENSION DD(25),D(25),QL(25),DN(25),           AM(25),AID(2500776000
     1),AIRO(25),ECC(25),ALFA(25),BETA(25),GAMMA(25)                     00777000
     2,TITLE(36),YSAVE(102),SP(3)                                        00778000
       DIMENSION QK(25),QC(25),QKP(25),QCP(25),QKHD(25),QCHD(25),QKF(2500781000
     1),QCF(25),QKPF(25),QCPF(25),QKHDF(25),QCHDF(25)                    00782000
     2,IB(12),K(25)                                                      00783000
     3,BKMX(25),BKMY(25),BCMX(25),BCMY(25),BCB(25),          RO(25)      00784000
     4,BNB(25,6),BBB(25,6),                BROB(25,7),                   00785000
     5C(25,25),B(25,25)                                                  00786000
       DIMENSION XX(25),YY(25),XDOT(25),YDOT(25)                         00789000
     1,XB(25),YB(25),XBDOT(25),YBDOT(25)                                 00790000
     1,QM(25),DELTX(25),DELTY(25),WHRVLO(25),WHIRPM(25),Y(102),KK(25),   00793000
     2BRGRO(25),PHABRO(25),BRGFOR(25),PHAFOR(25),WHRATI(50),STIFF(25)    00794000
     4,ROSQ(25),WHRATO(25),ROMAX(50),ISTATN(50),RPM(50),ROSTA(50),   INPR00797000
     5PM(50),ROO(25),PHAROO(25),Z(25),BRGR(50,12),FORC(50,12),TT(50),    00798000
     6SZ(25),ZQ(25),SZOL(25),ZQOL(25),QZ(25),ZSOL(25),QZOL(25)           00799000
      COMMON/MAREAD/TITLE,INPRPM,          T,DT,TMAX, TMZ,TMZ1,TMZ2,TMZ300803000
     1,       TOLI,  NOORPM,       IASIGN,CRT,ACCEL                      00804000
      COMMON/MAFUF1/DD,D,QL,  NS,NB,IB                                   00806000
     COMMON/MAFUF2/   SZ,QZ,ZQ,SZOL,ZSOL,QZOL,ZQOL,QMLOV,QLL             00807000
      COMMON/MAFUF3/  IB1,IBNB                                           00808000
     COMMON/MAFUF4/ Z                                                    00809000
     1       /MAFUO/ITIM,INT,KK                                          00810000
      COMMON/MAFU14/ FDOFIX                                              00810500
      COMMON/MAFU1/        TSTOP,K                                       00811000
      COMMON/MAFU2/ DN,AM,AID,AIRO,ECC,ALFA,BETA,GAMMA,GX,GY             00812000
     1       /MAFU3/WHIVEL                                               00813000
      COMMON/MAFU5/  QK,QC,QKP,QCP,QKHD,QCHD,QKF,QCF,QKPF,QCPF,QKHDF,QC00815000
     1HDF,     BKMX,BKMY,BCMX,BCMY,BCB                                   00816000
      COMMON/MAFU6/ BNB,BBB,BROB                                         00817000
     COMMON /MAFU7/ROSQ,    WHRVLO,DELTX,DELTY                           00818000
      COMMON/MAFU9/  C,B                                                 00820000
     1       /MAFU10/RO,PHAROO                                           00821000
       COMMON/MAFU11/FD,FDOT                                             00822000
```

```
     1       /MAFU13/Y                                                          00823000
     X       /SFP1/RPM,NPOINT,ICOND,CONTIN                                      00824000
     1       /SFP2/NSM1,NSM2,NSM3,NSP1,NS2,NS3,NS4,NS4P1,N,NS2P1                00824100
     1       /GARBG7/XX,YY,XB,YB,XDOT,YDOT,XBDOT,YBDOT                          00824200
     2       /GARBG9/YSAVE,WHIRPM,BRGRO,PHABRO,BRGFOR,PHAFOR,WHRATO,            00824300
     3PHARO(25),ROMAX,ISTATN,ROO,BRGR,FORC,TT,WHRATI                            00824400
      COMMON/MAFU12/  OM                                                        00825000
      DATA PI/3.141592653589793 24/                                             00825500
  404 FORMAT (7X,1P6E13.4)                                                      00826000
        A=180/PI                                                                00826100
        H=30/PI                                                                 00826200
        V=1/(2*PI)                                                              00826300
        GO TO 2                                                                 00826400
    1  KA=0                                                                     00826410
       DO 5 I=1,N,3                                                             00826420
       SP(1)=Y(I)                                                               00826430
       SP(2)=Y(I+1)                                                             00826440
       SP(3)=Y(I+2)                                                             00826450
       KA=KA+1                                                                  00826460
    5  PUNCH 106,SP,KA                                                          00826470
  106  FORMAT(1P3D22.15,6X,I8)                                                  00826480
    2 CALL COUNTV                                                               00826700
        TMIN=0                                                                  00826710
      ICC=0                                                                     00826800
      CALL READIN                                                               00828000
        CALL WRIOUT                                                             00829000
       TR=T                                                                     00829500
      IC=0                                                                      00830000
      ITIM=1                                                                    00831000
      IND=0                                                                     00832000
      INT=0                                                                     00833000
      MM=1                                                                      00834000
        MAXSHF=0                                                                00836000
      WHIVEL=WHIVEL/H                                                           00837000
       IF(ICOND.EQ.0) GO TO 3                                                   00838000
          DO 33 I=1,NB                                                          00845000
           IBI=IB(I)                                                            00846000
   33      KK(IBI)=1                                                            00847000
        IB1=IB(1)                                                               00857000
        IBNB=IB(NB)                                                             00858000
        CALL SZMASS                                                             00859000
      CALL INFLCO(C,B)                                                          00860000
```

```
        DO 22 J=1,NS                                        00861000
    DELT=0.                                                 00862000
        DO 23 I=1,NS                                        00863000
 23 DELT=DELT+C(I,J)*QM(I)                                  00865000
    DELTX(J)=DELT*GX                                        00866000
 22 DELTY(J)=DELT*GY                                        00867000
  3  ICOND=0                                                00867050
    IF(CONTIN.EQ.1)GOTO80                                   00867100
    GO TO 8                                                 00867110
  7 ITIM=1                                                  00867120
    DT=0.1*DT                                               00867130
  8 CALL STARUP                                             00867200
        DO 95  I=1,NS                                       00868000
        INS=I+NS                                            00869000
     I2NS=I+NS2                                             00870000
     I3NS=I+NS3                                             00871000
     PHAROO(I)=PHAROO(I)/A                                  00871100
     XNUTR=RO(I)*DCOS(PHAROO(I))                            00871200
     YNUTR=RO(I)*DSIN(PHAROO(I))                            00871300
     XDOT(I)=-WHIVEL*YNUTR                                  00871400
     YDOT(I)=WHIVEL*XNUTR                                   00871500
     Y(I)=DSQRT(XNUTR**2+YNUTR**2)                          00874000
     Y(INS)=DATAN(YNUTR/XNUTR)                              00875000
     Y(I2NS)=(XNUTR*XDOT(I)+YNUTR*YDOT(I))/Y(I)             00876000
 95  Y(I3NS)=(XNUTR*YDOT(I)-YNUTR*XDOT(I))/Y(I)**2          00877000
     Y(NS4P1)=FD/A                                          00879000
     Y(N)=FDOT/H                                            00880000
 80  WRITE(6,1555)                                          00880100
1555 FORMAT(1H04X'ELEMENTS IN ROTATING COORDINATES:')       00880200
     WRITE(6,404)(Y(I),I=1,N)                               00880300
100  IF(CRT.EQ.0) GO TO 107                                 00881000
       IF(T.GE.TMAX.OR.TMIN.GE.TSTOP) GO TO 1040            00881500
         PPM=0                                              00882000
         P1=FDOT                                            00883000
107  TSAVE=T                                                00884000
         DO 41 I=1,N                                        00885000
 41      YSAVE(I)=Y(I)                                      00886000
 24  IF(T.GE.TMAX) GO TO 900                                00887000
     ICC=ICC+1                                              00887100
C    IF(ICC.LT.10) GO TO 97                                 00887200
     CALL TIMEV(TSEC)                                       00887300
     TMIN=TSEC/60                                           00887400
```

```
      IF(TMIN.GE.TSTOP) GO TO 1
      ICC=0
   97 CALL RKADAM(N,TR,Y,DT,IND,ITIM,TOLI,IERR)
      DO 4 I=1,N
      IF(Y(I).GE.1.D14) GO TO 7
    4 CONTINUE
        CALL TIME(TR,DT,T,ITIM)
      IF(IERR.EQ.0)GOTO108
      WRITE(6,311)T
  311 FORMAT('-UNSUCCESSFUL SOLUTION AT T ='1PD12.5)
      STOP
  108   IF(CRT.EQ.0) GO TO 110
          PM1=INPRPM(MM)-P1*H
      PM2=INPRPM(MM)-H*Y(N)
          P12=PM1*PM2
          IF(P12.GT.0) GO TO 110
          MM=MM+1
      IF(DABS(PM2).GT.DABS(PM1))GOTO1000
  110     DO 32  I=1,NS
          INS=I+NS
      I2NS=I+NS2
      I3NS=I+NS3
      COSAB=DCOS(Y(INS))
      SINAB=DSIN(Y(INS))
      XNUTR=Y(I)*COSAB
      YNUTR=Y(I)*SINAB
      XX(I)=XNUTR-DELTX(I)
      YY(I)=YNUTR-DELTY(I)
          PHARO(I)=Y(INS)
      XDOT(I)=Y(I2NS)*COSAB-Y(I)*Y(I3NS)*SINAB
      YDOT(I)=Y(I2NS)*SINAB+Y(I)*Y(I3NS)*COSAB
   32 WHRVLO(I)=Y(I3NS)
      FDOT=Y(N)
      CALL BRGXY
       A1=0
       B1=0
          DO 317 I=1,NB
          IBI=IB(I)
      BRGRO(IBI)=DSQRT(XB(IBI)**2+YB(IBI)**2)
       KP1=K(IBI)+1
       IF(BRGRO(IBI).GT.BROB(IBI,KP1)) GO TO 27
          KKB=KK(IBI)
```

```
   25 IF(BRGRO(IBI).LT.BROB(IBI,KKB))GOTO29                              00932000
   26  KKB1=KKB+1                                                        00933000
      IF(BRGRO(IBI).LE.BROB(IBI,KKB1))GO TO 28                           00934000
      KKB=KKB+1                                                          00935000
       A1=A1+1                                                           00936000
      GO TO 26                                                           00936700
   27 WRITE(6,200)                                                       00938000
  200 FORMAT(' BEARING DEFLECTION EXCEEDS BEARING CLEARANCE')            00939000
      GO TO  1                                                           00940000
   29     KKB=KKB-1                                                      00941000
          B1=B1+1.                                                       00942000
      GO TO 25                                                           00943000
   28     KK(IBI)=KKB                                                    00944000
  317     CONTINUE                                                       00945000
       A1B1=A1+B1                                                        00947000
      IF(A1B1.EQ.0) GO TO 318                                            00948000
      ITIM=1                                                             01034000
      T=TSAVE                                                            01035000
        TR=T                                                             01035100
      DO 102 I=1,N                                                       01036000
  102 Y(I)=YSAVE(I)                                                      01037000
        MAXSHF=MAXSHF+1                                                  01038000
      IF(MAXSHF.GE.10) GO TO 320                                         01039000
      GOTO24                                                             01040000
  320 WRITE(6,330)                                                       01041000
  330 FORMAT(1HO,4X,'THE MAXIMUM NUMBER OF SHIFTING OF STIFFNESS SECTION01042000
     1S IS TENTATIVELY LIMITED TO 10'/4X,'TO AVOID INFINITE CYCLIC COMPU01043000
     2TATIONS DUE TO INADEQUATE INPUT DATA.')                            01044000
        GO TO 1                                                          01045000
  318 IC=IC+1                                                            01046000
      ITIM=0                                                             01047000
      DO 43 I=1,NB                                                       01048000
          IBI=IB(I)                                                      01049000
      BRGRO(IBI)=DSQRT(XB(IBI)**2+YB(IBI)**2)                            01050000
      PHABRO(IBI)=DATN2D(YB(IBI),XB(IBI))                                01051000
      IF(PHABRO(IBI).LT.0) PHABRO(IBI)=360.+PHABRO(IBI)                  01052000
      BRGFOX=BKMX(IBI)*(XX(IBI)-XB(IBI))                                 01053000
      BRGFOY=BKMY(IBI)*(YY(IBI)-YB(IBI))                                 01054000
      BRGFOR(IBI)=DSQRT(BRGFOX**2+BRGFOY**2)                             01055000
      STIFF(IBI)=BRGFOR(IBI)/BRGRO(IBI)                                  01055100
      BRGFOX=BRGFOX+XDOT(IBI)*BCB(IBI)                                   01055200
      BRGFOY=BRGFOY+YDOT(IBI)*BCB(IBI)                                   01055300
```

```
      BRGFOR(IBI)=DSQRT(BRGFOX**2+BRGFOY**2)                            01055400
      PHAFOR(IBI)=DATN2D(BRGFOY,BRGFOX)                                 01056000
   43 IF(PHAFOR(IBI).LT.0) PHAFOR(IBI)=360.+PHAFOR(IBI)                 01057000
      FDOT=Y(N)                                                         01059000
      REVOLN=V*Y(NS4P1)                                                 01060000
          DO 45 I=1,NS                                                  01061000
          WHIRPM(I)=          H*WHRVLO(I)                               01062000
   45 WHRATO(I)=WHRVLO(I)/FDOT                                          01063000
          WHRATI(IC)=WHRATO(IASIGN)                                     01064000
          RPMM=          H*FDOT                                         01065000
          RPM(IC)=RPMM                                                  01066000
          TT(IC)=T                                                      01067000
          WRITE(6,1001)T                                                01068000
 1001 FORMAT (1H1//34X,           16HAT THE TIME:  T=1PD13.6,2X,'SEC.')01069000
      WRITE (6,701)                                                     01070000
  701 FORMAT(1H0,4X,52HX DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(I01071000
     1N.))                                                              01072000
      WRITE  (6,404) (XX(I),I=1,NS)                                     01073000
      WRITE  (6,702)                                                    01074000
  702 FORMAT(1H0,4X,52HY DISPLACEMENT ARRAY IN STATIONARY COORDINATES,(I01075000
     1N.))                                                              01076000
      WRITE  (6,404) (YY(I),I=1,NS)                                     01077000
      WRITE(6,707)                                                      01078000
  707 FORMAT(1H0,4X,35HROTOR DEFLECTION VECTOR ARRAY (IN.))             01079000
        DO 34 I=1,NS                                                    01080000
      RO(I)=Y(I)                                                        01081000
      PHARO(I)=DATN2D(YY(I),XX(I))                                      01082000
   34 IF(PHARO(I).LT.0) PHARO(I)=360.+PHARO(I)                          01083000
      WRITE(6,404) (RO(I),I=1,NS)                                       01084000
      WRITE(6,708)                                                      01085000
  708 FORMAT(1H0,4X,  'PHASE ANGLE ARRAY FOR ROTOR DEFLECTION VECTORS, D01086000
     1EGREES')                                                          01087000
      WRITE(6,404) (PHARO(I),I=1,NS)                                    01088000
      WRITE  (6,703)                                                    01089000
  703 FORMAT(1H0,4X,  'X VELOCITY ARRAY IN STATIONARY COORDINATES,(IN/SE01090000
     1C)')                                                              01090100
      WRITE  (6,404) (XDOT(I),I=1,NS)                                   01091000
      WRITE  (6,704)                                                    01092000
  704 FORMAT(1H0,4X,  'Y VELOCITY ARRAY IN STATIONARY COORDINATES,(IN/SE01093000
     1C)')                                                              01093100
      WRITE (6,404) (YDOT(I),I=1,NS)                                    01094000
      WRITE  (6,705)                                                    01095000
```

```
  705 FORMAT (1H0,4X,25HWHIRL FREQUENCY ARRAY,RPM)
      WRITE  (6,404) (WHIRPM(I),I=1,NS)
      WRITE  (6,706)
  706 FORMAT (1H0,4X,35HWHIRL TO SPIN FREQUENCY RATIO ARRAY)
      WRITE  (6,404) (WHRATO(I),I=1,NS)
      WRITE(6,1002)REVOLN,RPMM
 1002 FORMAT(///5X,26HTOTAL NO. OF REVOLUTIONS =1PD13.4/ 5X,26HSPIN SPEE
     1D, RPM             =1PD13.4)
      WRITE (6,457)
  457 FORMAT(/1H1///5X,8HXB ARRAY,10X, 36HBEARING DISPLACEMENT COORDINAT
     1E, IN.)
      WRITE (6,404) (XB(IB(I)),I=1,NB)
      WRITE (6,458)
  458 FORMAT (1H0,4X,8HYB ARRAY,10X,36HBEARING DISPLACEMENT COORDINATE,
     1IN.)
      WRITE (6,404) (YB(IB(I)),I=1,NB)
      WRITE (6,459)
  459 FORMAT (1H0,4X,11HXBDOT ARRAY,7X,   'BEARING VELOCITY COORDINATE,
     1IN/SEC.')
      WRITE (6,404) (XBDOT(IB(I)),I=1,NB)
      WRITE(6,460)
  460 FORMAT (1H0,4X,11HYBDOT ARRAY,7X,   'BEARING VELOCITY COORDINATE,
     1IN/SEC.')
      WRITE (6,404) (YBDOT(IB(I)),I=1,NB)
      WRITE (6,1003)
 1003 FORMAT (//1H0,4X, 11HBRGRO ARRAY,7X,   'JOURNAL DISPLACEMENT FROM
     1BEARING-CENTER ARRAY, IN.')
      WRITE (6,404) (BRGRO(IB(I)),I=1,NB)
      WRITE (6,1010)
 1010 FORMAT (1H0,4X,12HPHABRO ARRAY,7X,   'JOURNAL DISPLACEMENT VECTOR
     1PHASE ANGLE COUNTER-CLOCKWISE FROM X-AXIS, DEGREES')
      WRITE (6,404) (PHABRO(IB(I)),I=1,NB)
      WRITE (6,1008)
 1008 FORMAT(1H0,4X,12HBRGFOR ARRAY,6X,'BEARING REACTION ARRAY, LB.')
      WRITE (6,404) (BRGFOR(IB(I)),I=1,NB)
      WRITE (6,1009)
 1009 FORMAT (1H0,4X, 12HPHAFOR ARRAY, 6X,   'BEARING REACTION VECTOR PH
     1ASE ANGLE COUNTER-CLOCKWISE FROM X-AXIS, DEGREES')
      WRITE (6,404) (PHAFOR(IB(I)),I=1,NB)
      WRITE(6,1020)
 1020 FORMAT(1H0,4X,11HSTIFF ARRAY,7X'BEARING FORCE TO JOURNAL DEFLECTIO
     1N RATIO, OR EQUIVALENT LINEAR BEARING'/24X,'STIFFNESS, LB/IN/BEARI
```

```
     2NG')                                                              01134310
      WRITE(6,404) (STIFF(IB(I)),I=1,NB)                                01134400
      IF(CRT.NE.0) GO TO 105                                            01135000
      IC=1                                                              01135100
      GO TO 107                                                         01135200
  105 DO 500 I=1,NB                                                     01136000
      J=IB(I)                                                           01137000
      FORC(IC,I)=BRGFOR(J)                                              01138000
  500 BRGR(IC,I)=BRGRO(J)                                               01139000
      IF(ACCEL.EQ.0)GOTO1000                                            01140000
      IF(P12.GT.0.0) GO TO 1027                                         01141000
      GO TO 1000                                                        01142000
  900 IF(CRT.EQ.0.OR.ACCEL.NE.0)GOTO1                                   01143000
 1000 ACCEL =1                                                          01145000
      DO 1104  I=1,NS                                                   01146000
 1104 ROO(I) =RO(I)                                                     01147000
C     PLOT 1                                                            01147005
      REAL CHAR11(21),CHAR21(7),CHAR31(8),CHARSS(4),SYMBOL/'*'/         01147010
      DATA CHAR11/'ROTOR 3-DIMENSIONAL MODE SHAPE WITH PHASE ANGLES (DEG01147020
     1REES) LABELED AS SHOWN, AT RPM='/,CHAR21/'ROTOR AXIAL LENGTH,  INC01147030
     2HES'/,CHAR31/'ROTOR DEFLECTION VECTOR, INCHES'/                   01147040
      CALL LRLEGN(TITLE,72,0,1.15,9.99,0.)                              01147050
      CALL LRLEGN(TITLE(19),72,0,1.15,9.83,0.)                          01147060
      CALL LRLEGN(CHAR11,84,0,1.463,9.67,0.)                            01147070
      CALL LRCNVT(RPMM,3,CHARS ,4,13,5)                                 01147080
      CALL LRLEGN(CHARSS,13,0,8.6,9.67,0.)                              01147090
      CALL LRLEGN(CHAR21,27,0,4.31,0.,0.)                               01147100
      CALL LRCURV(Z,ROO,NS,2,SYMBOL,0.)                                 01147110
      DO 1005 I=1,NS                                                    01155000
      RSTAR4=PHARO(I)                                                   01155010
      CALL LRCNVT(RSTAR4,3,CHARSS,3,4,0)                                01155020
      CALL LRLABL(CHARSS,4,0,Z(I),ROO(I),0.)                            01155030
 1005 CONTINUE                                                          01160000
      CALL LRLEGN(CHAR31,31,1,0.,4.6,1.)                                01160010
      IF(DABS(PM2).LE.DABS(PM1)) GO TO 1027                             01161000
      IF(PPM.EQ.1) GO TO 100                                            01162000
      PPM=1                                                             01163000
      GO TO 110                                                         01164000
 1027 PPM=0                                                             01165000
      J=1                                                               01167000
      DO 1209 I=1,NS                                                    01168000
 1209 IF(RO(J).LT.RO(I))J=I                                             01169000
```

```
      ROMAX(IC)=RO(J)                                                   01170000
      ISTATN(IC)=J                                                      01171000
      ROSTA(IC)=RO(IASIGN)                                              01171500
      IF(IC.LT.NPOINT) GO TO 100                                        01171600
 1040 NPOINT =IC                                                        01172000
C     PLOT 2                                                            01173000
      CALL LRLEGN(TITLE,72,0,1.15,9.99,0.)                              01173500
      CALL LRLEGN(TITLE(19),72,0,1.15,9.83,0.)                          01174000
      REAL CHAR12(7)/'ROTOR SPIN SPEED VERSUS TIME'/,CHAR22(4)/'TIME, SE01174500
     1CONDS   '/,CHAR32(6)/'ROTOR SPIN SPEED, RPM    '/                 01175000
      CALL LRCURV(TT,RPM,NPOINT,2,SYMBOL,0.)                            01175500
      CALL LRCURV(TT,RPM,NPOINT,3,SYMBOL,0.)                            01176000
      CALL LRLEGN(CHAR12,28,0,3.50,9.67,0.)                             01176500
      CALL LRLEGN(CHAR22,13,0,4.86,0.,0.)                               01177000
      CALL LRLEGN(CHAR32,21,1,0.,4.95,1.)                               01177100
      IF(ACCEL.EQ.0)GOTO2000                                            01177200
C     PLOT 3                                                            01177400
      CALL LRLEGN(TITLE,72,0,1.15,9.99,0.)                              01177500
      CALL LRLEGN(TITLE(19),72,0,1.15,9.83,0.)                          01178000
      REAL CHAR13(19)/'ROTOR WHIRL-TO-SPIN FREQUENCY RATIO VERSUS ROTOR 01178500
     1SPIN SPEED AT ROTOR STATION'/                                     01179000
      CALL LRCURV(RPM,WHRATI,NPOINT,2,SYMBOL,0.)                        01179500
      CALL LRCURV(RPM,WHRATI,NPOINT,3,SYMBOL,0.)                        01180000
      CALL LRCNVT(IASIGN,1,CHARSS,1,3,0)                                01181000
      CALL LRLEGN(CHARSS,3,0,7.85,9.67,0.)                              01181500
      CALL LRLEGN(CHAR13,76,0,1.795,9.67,0.)                            01182000
      CALL LRLEGN(CHAR13,35,1,0.,3.47,0.)                               01182500
      CALL LRLEGN(CHAR32,21,0,4.55,0.,1.)                               01183000
C     PLOT 4                                                            01184000
C     PLOT RPM VS FORC FUNCTIONS                                        01185000
      CALL PLOT1(FORC)                                                  01186000
      CALL LRLEGN(TITLE,72,0,1.15,9.99,0.)                              01186100
      CALL LRLEGN(TITLE(19),72,0,1.15,9.83,0.)                          01186200
      REAL CHAR14(24)/'BEARING REACTIONS VERSUS ROTOR SPIN SPEED WITH BE01186300
     1ARING LOCATION STATION NUMBERS LABELED AS SHOWN'/,CHAR24(7)/'BEARI01186400
     2NG REACTIONS, POUNDS    '/                                        01186500
      CALL LRLEGN(CHAR14,96,0,1.385,9.67,0.)                            01186600
      CALL LRLEGN(CHAR32,21,0,4.55,0.,0.)                               01186700
      CALL LRLEGN(CHAR24,25,1,0.,4.37,1.)                               01186800
C     PLOT 5                                                            01192000
C     PLOT RPM VS BRGR FUNCTIONS                                        01193000
      CALL PLOT1(BRGR)                                                  01194000
```

```
      CALL LRLEGN(TITLE,72,0,1.15,9.99,0.)                              01194100
      CALL LRLEGN(TITLE(19),72,0,1.15,9.83,0.)                          01194200
      REAL CHAR15(25)/'JOURNAL DISPLACEMENT VERSUS ROTOR SPIN SPEED WITH01194300
     1 BEARING LOCATION STATION NUMBERS LABELED AS SHOWN '/,CHAR25(8)/  01194400
     2'JOURNAL DISPLACEMENTS, INCHES    '/                              01194500
      CALL LRLEGN(CHAR15,99,0,1.21,9.67,0.)                             01194600
      CALL LRLEGN(CHAR32,21,0,4.55,0.,0.)                               01194700
      CALL LRLEGN(CHAR25,29,1,0.,4.52,1.)                               01194800
C     PLOT 6                                                            01200000
      CALL LRANGE(0.,0.,0.,0.)                                          01200005
      REAL CHAR16(13)/'MAXIMUM ROTOR DEFLECTIONS VERSUS ROTOR SPIN SPEED01201000
     1   '/,CHAR26(22)/'(THE STATION NUMBERS WHERE THE MAXIMUM DEFLECTIO01202000
     2NS OCCUR ARE SHOWN) '/,CHAR36(9)/'MAXIMUM ROTOR DEFLECTIONS, INCHE01203000
     3S   '/                                                            01204000
      CALL LRCURV(RPM,ROMAX,NPOINT,2,SYMBOL,0.)                         01205000
      DO 1006 I=1,NPOINT                                                01206000
      CALL LRCNVT(ISTATN(I),1,CHARSS,1,3,0)                             01207000
 1006 CALL LRLABL(CHARSS,3,0,RPM(I),ROMAX(I),0.)                        01208000
      CALL LRLEGN(TITLE,72,0,1.15,9.99,0.)                              01209000
      CALL LRLEGN(TITLE(19),72,0,1.15,9.873,0.)                         01210000
      CALL LRLEGN(CHAR16,49,0,3.45,9.756,0.)                            01211000
      CALL LRLEGN(CHAR26,67,0,2.75,9.639,0.)                            01212000
      CALL LRLEGN(CHAR32,21,0,4.55,0.,0.)                               01213000
      CALL LRLEGN(CHAR36,33,1,0.,4.52,1.)                               01214000
C     PLOT 7                                                            01214010
      REAL CHAR17(17)/'(THE STATION NUMBER WHERE THE ROTOR DEFLECTIONS O01214020
     1CCUR IS SHOWN)'/                                                  01214030
      CALL LRCURV(RPM,ROSTA,NPOINT,2,SYMBOL,0.)                         01214040
      DO 1007 I=1,NPOINT                                                01214050
      CALL LRCNVT(IASIGN,1,CHARSS,1,3,0)                                01214060
 1007 CALL LRLABL(CHARSS,3,0,RPM(I),ROSTA(I),0.)                        01214070
      CALL LRLEGN(TITLE,72,0,1.15,9.99,0.)                              01214080
      CALL LRLEGN(TITLE(19),72,0,1.15,9.873,0.)                         01214090
      CALL LRLEGN(CHAR16(3),41,0,3.45,9.756,0.)                         01214100
      CALL LRLEGN(CHAR17,63,0,2.75,9.639,0.)                            01214110
      CALL LRLEGN(CHAR32,21,0,4.55,0.,0.)                               01214120
      CALL LRLEGN(CHAR36(3),25,1,0.,4.25,1.)                            01214130
 2000 IC=0                                                              01215000
        IF(T.GE.TMAX.OR.TMIN.GE.TSTOP) GO TO 1                          01215100
      GOTO100                                                           01216000
        END                                                             01217000
```

Card Count 419

```
      SUBROUTINE READIN                                                 01218000
      IMPLICIT REAL*8 (A-H,O-Z)                                         01219000
      INTEGER CRT,ACCEL,CONTIN                                          01220000
      REAL TITLE,INPRPM, RPM                                            01221000
        DIMENSION DD(25),D(25),QL(25),DN(25),EE(25),GG(25),AM(25),AID(2501222000
     1),AIRO(25),ECC(25),ALFA(25),BETA(25),GAMMA(25), EI(25),GAK(25)    01223000
     2,TITLE(36),INPRPM(50),RPM(50)                                     01224000
        DIMENSION CZ(25),CZ1(25),CZ2(25)                                01225000
        DIMENSION XKF(25),XCF(25),XKFF(25),XCFF(25)                     01226000
        DIMENSION QK(25),QC(25),QKP(25),QCP(25),QKHD(25),QCHD(25),QKF(2501227000
     1),QCF(25),QKPF(25),QCPF(25),QKHDF(25),QCHDF(25)                   01228000
     2,IB(12),K(25)                                                     01229000
     3,BKMX(25),BKMY(25),BCMX(25),BCMY(25),                   BCB(25),  01230000
     4 BNB(25,6),BBB(25,6),BROB(25,7),BKB(25,6),BHB(25,6),BDB(25,6),    01231000
     5BEB(25,6),             Y(102)                                     01232000
      COMMON/MAREAD/ TITLE,INPRPM,         T,DT,TMAX, TMZ,TMZ1,TMZ2,TMZ301234000
     1,      TOLI,  NOORPM,        IASIGN,CRT,ACCEL                     01235000
     1      /SFP1/RPM,NPOINT,ICOND,CONTIN                               01236000
       COMMON/MAFC/  EE,GG, EI,GAK                                      01237000
       COMMON/MAFUF1/DD,D,QL,  NS,NB,IB                                 01238000
       COMMON/MAFU14/ FDOFIX                                            01238500
       COMMON/MAFU1/        TSTOP,K                                     01239000
       COMMON/MAFU2/ DN,AM,AID,AIRO,ECC,ALFA,BETA,GAMMA,GX,GY           01240000
     1      /MAFU3/WHIVEL                                               01241000
       COMMON/MAFU4/ CZ,CZ1,CZ2,  XKF,XCF,XKFF,XCFF                     01242000
       COMMON/MAFU5/  QK,QC,QKP,QCP,QKHD,QCHD,QKF,QCF,QKPF,QCPF,QKHDF,QC01243000
     1HDF,    BKMX,BKMY,BCMX,BCMY,BCB                                   01244000
       COMMON/MAFU6/ BNB,BBB,BROB,BKB,BHB,BDB,BEB                       01245000
      COMMON/MAFU8/  TOLB                                               01246000
     1      /MAFU13/Y                                                   01247000
        COMMON/MAFU11/FD,FDOT                                           01248000
     1      /SFP2/NSM1,NSM2,NSM3,NSP1,NS2,NS3,NS4,NS4P1,NS4P2,NS2P1     01248100
      NAMELIST/DATA1/NS,   TMAX,TSTOP,DD,QL,NB,IB,K,ICOND,CONTIN, FDOT,W01249000
     1HIVEL                                                             01249100
      NAMELIST/DATA2/ T,FD,DT,FDOFIX,                                   01250000
     1               IASIGN,NOORPM,NPOINT,CRT,ACCEL,INPRPM,TOLI,TOLB,GX,01251000
     1GY,TMZ,TMZ1,TMZ2,TMZ3,     D,DN,EE,GG, EI,GAK,AM,AID,AIRO,ECC,ALFA01252000
     2,BETA,GAMMA,CZ,CZ1,CZ2,XKF,XCF,XKFF,XCFF,QK,QC,QKP,QCP,QKHD,QCHD,Q01253000
     3KF,QCF,QKPF,QCPF,QKHDF,QCHDF,  BKMX,BKMY,BCMX,BCMY,BCB,BNB,BBB,BRO01254000
     4B,BKB,BHB,BDB,BEB                                                 01255000
      READ (5,5) TITLE                                                  01257000
    5 FORMAT(18A4)                                                      01258000
```

```
      READ(5,DATA1)                                                     01259000
        IF(NS.LT.5) GO TO 12                                            01261000
        GO TO 11                                                        01262000
   12 WRITE (6,377)                                                     01263000
  377 FORMAT(///1H0,10X, 47HMINIMUM ALLOWABLE NUMBER OF ROTOR STATIONS I01264000
     1S 5)                                                              01265000
      CALL EXIT                                                         01266000
   11   FD=0                                                            01266500
        IASIGN=1                                                        01267000
        NOORPM=1                                                        01268000
        NPOINT=25                                                       01269000
        CRT=0                                                           01270000
        ACCEL=0                                                         01271000
        INPRPM(1)=0                                                     01272000
        T=0                                                             01273000
        DT=.00001                                                       01273500
        TOLI=.01                                                        01274000
        TOLB=.001                                                       01275000
        GX=0                                                            01276000
        GY=0                                                            01277000
        TMZ=0                                                           01278000
        TMZ1=0                                                          01279000
        TMZ2=0                                                          01280000
        TMZ3=0                                                          01281000
      NSM1=NS-1                                                         01281200
      NSM2=NS-2                                                         01281400
      NSM3=NS-3                                                         01281600
      NSP1=NS+1                                                         01281800
      NS2=2*NS                                                          01282000
      NS3=3*NS                                                          01282200
      NS4=4*NS                                                          01282400
      NS4P1=NS4+1                                                       01282600
      NS4P2=NS4+2                                                       01282800
      NS2P1=NS2+1                                                       01283000
      DO 14 I=1,NSM1                                                    01283200
        D(I)=0                                                          01284000
        DN(I)=.283                                                      01285000
        EE(I)=3.D7                                                      01286000
        GG(I)=1.15D7                                                    01287000
         EI(I)=0                                                        01288000
   14   GAK(I)=0                                                        01289000
        DO 16 I=1,NS                                                    01290000
```

```
      AM(I)=1.D-16                                                      01291000
      AID(I)=0                                                          01292000
      AIRO(I)=1.D-16                                                    01293000
      ECC(I)=1.D-16                                                     01294000
      ALFA(I)=0                                                         01295000
      BETA(I)=0                                                         01296000
      GAMMA(I)=0                                                        01297000
      CZ(I)=0                                                           01298000
      CZ1(I)=0                                                          01299000
      CZ2(I)=0                                                          01300000
      XKF(I)=0                                                          01301000
      XCF(I)=0                                                          01302000
      XKFF(I)=0                                                         01303000
      XCFF(I)=0                                                         01304000
      QK(I)=0                                                           01305000
      QC(I)=0                                                           01306000
      QKP(I)=0                                                          01307000
      QCP(I)=0                                                          01308000
      QKHD(I)=0                                                         01309000
      QCHD(I)=0.                                                        01310000
      QKF(I)=0                                                          01311000
      QCF(I)=0                                                          01312000
      QKPF(I)=0                                                         01313000
      QCPF(I)=0                                                         01314000
      QKHDF(I)=0                                                        01315000
 16   QCHDF(I)=0                                                        01316000
                                                                        01317000
      DO 20 N1=1,NB                                                     01318000
      I=IB(N1)                                                          01319000
        BKMX(I)=1.D10                                                   01321000
        BKMY(I)=1.D10                                                   01322000
        BCMX(I)=1.D-16                                                  01323000
        BCMY(I)=1.D-16                                                  01324000
 20     BCB(I)= 1.D-16                                                  01325000
        FDOFIX=0                                                        01325500
      DO 21 N1=1,NB                                                     01326000
      I=IB(N1)                                                          01327000
      KI=K(I)                                                           01328000
        DO 21 J=1,KI                                                    01329000
        BNB(I,J)=0                                                      01330000
        3BB(I,J)=0                                                      01331000
        3KB(I,J)=1.D6                                                   01332000
```

```
        BHB(I,J)=1                                                01333000
        BDB(I,J)=0                                                01334000
 21     BEB(I,J)=0                                                01335000
      DO 22 N1=1,NB                                               01336000
      I=IB(N1)                                                    01337000
 22     BROB(I,1)=.005                                            01340000
 15   FORMAT(6E12.8)                                              01341000
      READ(5,DATA2)                                               01342000
      DT=2.*DT                                                    01343000
        DO 24 N1=1,NB                                             01343100
        I=IB(N1)                                                  01343200
        KI=K(I)                                                   01343300
        DO 24 J=1,KI                                              01343400
        JP1=J+1                                                   01343500
 24     BROB(I,JP1)=BROB(I,J)                                     01343600
        DO 26 N1=1,NB                                             01343700
        I=IB(N1)                                                  01343800
 26    BROB(I,1)=0                                                01343900
        DD(NS)=1.                                                 01344000
        D(NS)= 0.                                                 01345000
        QL(NS)=1.                                                 01346000
        DN(NS)=0.                                                 01347000
        EE(NS)=1.                                                 01348000
        GG(NS)=1.                                                 01349000
         EI(NS)=0.                                                01350000
        GAK(NS)=0.                                                01351000
      IF(CONTIN.EQ.0) GO TO 17                                    01352000
      READ(5,410) (Y(I),I=1,NS4P2)                                01355000
410   FORMAT(3D22.15)                                             01356000
 17   DO 10 I=1,NS                                                01356100
      IF(EI(I).NE.0)   EE(I)=0                                    01356200
      IF(GAK(I).NE.0)  GG(I)=0                                    01356300
 10   CONTINUE                                                    01356400
      RETURN                                                      01367000
      END                                                         01368000
```

Card Count 161

```
      SUBROUTINE  WRIOUT                                                01570000
      IMPLICIT REAL*8 (A-H,O-Z)                                         01571000
      INTEGER CRT,ACCEL,CONTIN                                          01572000
      REAL TITLE, INPRPM,RPM                                            01573000
         DIMENSION DD(25),D(25),QL(25),DN(25),EE(25),GG(25),AM(25),AID(2501575000
     1),AIRO(25),ECC(25),ALFA(25),BETA(25),GAMMA(25), EI(25),GAK(25)    01576000
     2,TITLE(36)                                                        01577000
         DIMENSION CZ(25),CZ1(25),CZ2(25)                               01578000
         DIMENSION XKF(25),XCF(25),XKFF(25),XCFF(25)                    01579000
         DIMENSION QK(25),QC(25),QKP(25),QCP(25),QKHD(25),QCHD(25),QKF(2501580000
     1),QCF(25),QKPF(25),QCPF(25),QKHDF(25),QCHDF(25)                   01581000
     2,IB(12),K(25)                                                     01582000
     3,BKMX(25),BKMY(25),BCMX(25),BCMY(25),BCB(25)                      01583000
     4,BNB(25,6),BBB(25,6),BROB(25,7),BKB(25,6),BHB(25,6),BDB(25,6),    01584000
     5BEB(25,6),            RPM(50),INPRPM(50)                          01585000
      COMMON/MAREAD/ TITLE,INPRPM,        T,DT,TMAX, TMZ,TMZ1,TMZ2,TMZ301587000
     1,      TOLI,  NOORPM,       IASIGN,CRT,ACCEL                      01588000
       COMMON/MAFC/  EE,GG, EI,GAK                                      01589000
       COMMON/MAFUF1/DD,D,QL,  NS,NB,IB                                 01590000
       COMMON/MAFU14/ FDOFIX                                            01590500
       COMMON/MAFU1/        TSTOP,K                                     01591000
       COMMON/MAFU2/ DN,AM,AID,AIRO,ECC,ALFA,BETA,GAMMA,GX,GY           01592000
     1       /MAFU3/WHIVEL                                              01593000
       COMMON/MAFU4/ CZ,CZ1,CZ2,  XKF,XCF,XKFF,XCFF                     01594000
       COMMON/MAFU5/  QK,QC,QKP,QCP,QKHD,QCHD,QKF,QCF,QKPF,QCPF,QKHDF,QC01595000
     1HDF,    BKMX,BKMY,BCMX,BCMY,BCB                                   01596000
       COMMON/MAFU6/ BNB,BBB,BROB,BKB,BHB,BDB,BEB                       01597000
      COMMON/MAFU8/  TOLB                                               01598000
        COMMON/MAFU11/FD,FDOT                                           01601000
     X       /MAFU13/Y(102)                                             01601100
     X       /SFP1/RPM,NPOINT,ICOND,CONTIN                              01602000
     1       /SFP2/N1,NSM2,NSM3,NSP1,NS2,NS3,NS4,IDUM1(3)               01602100
     2       /GARBG3/ICO,M,KI1,KI,I                                     01602200
      WRITE(6,471)                                                      01602300
      WRITE(6,400)  TITLE                                               01603000
  399   FORMAT(10I10)                                                   01604000
  400 FORMAT (1H0/  (20X,18A4))                                         01605000
  404 FORMAT (7X,1P6E13.4)                                              01606000
  490 FORMAT (14X,1P6E13.4)                                             01607000
      WRITE (6,401) NS, IASIGN, NOORPM, NPOINT, CRT, ACCEL              01608000
  401 FORMAT (1H0/, 4X,8HNS     =I3,14X,   'NUMBER OF ROTOR STATIONS (AL01609000
     1LOWABLE RANGE:5=<NS=<25)'/4X,                                     01610000
```

```
     18HIASIGN =I3,14X,  70HA ROTOR STATION NUMBER AT WHICH THE WHIRL/SP01611000
     2IN FREQUENCY RATIO WILL BE/34X,'PLOTTED ON CRT'/4X,8HNOORPM =I3,1401612000
     3X,'THE NUMBER OF SPIN SPEEDS IN RPM AT OR NEAR WHICH 3-DIMENSIONAL01613000
     1 ABSOLUTE ROTOR'/34X'MODE SHAPE CRT GRAPHS ARE REQUIRED.   THE SPI01614000
     2N SPEED RPM VALUES ARE LISTED'/34X'UNDER INPRPM ARRAY.'/34X,'(ALLO01615000
     2WABLE RANGE: 0=<NOORPM=<50)'/                             4X,8HNPOIN01616000
     3T =I3,                                                             01617000
     614X,'THE NUMBER OF POINTS (ONE PER EACH INTEGRATION STEP) FOR EACH01618000
     7 CRT GRAPH.'/34X,'(ALLOWABLE RANGE: 1=<NPOINT=<50)'/               01619000
     8             4X,8HCRT     =I3,14X,31HCRT=0 MEANS CRT IS NOT REQUIRE01620000
     8D/29X,  27HCRT=1 MEANS CRT IS REQUIRED/4X,8HACCEL   =I3,14X,'ACCEL=01621000
     90 MEANS A 3-DIMENSIONAL ROTOR MODE SHAPE CRT CORRESPONDING TO THAT01622000
     9 AT'/34X, 'BEGINNING OF THE RUN WILL BE PROVIDED IF CONCURRENTLY C01623000
     9RT=1.'/29X,'ACCEL=1 MEANS ONLY THE TRANSIENT-SPEED ROTOR MODE SHAP01624000
     9ES AT OR NEAR INPRPM'/34X,'VALUES WILL BE PROVIDED IF CONCURRENTLY01625000
     9 CRT=1.')                                                          01625100
      WRITE (6,453)                                                      01626000
  453 FORMAT(1H0,  3X,12HINPRPM ARRAY, 13X,'THE ROTOR SPIN SPEED RPM VAL01627000
     1UES AT OR NEAR WHICH CRT GRAPHS FOR'/29X,53H3-DIMENSIONAL ABSOLUTE01628000
     2 ROTOR MODE SHAPES ARE REQUIRED)                                   01629000
      WRITE (6,404) (INPRPM(I),I=1,NOORPM)                               01630000
      DT=DT/2.                                                           01630500
      WRITE (6,402) T, DT, TMAX, TOLI, TOLB,TSTOP                        01631000
  402 FORMAT (  1H0/,4X,8HT       =1PD13.4,4X,  23HINITIAL REAL TIME, SEC01632000
     1./4X,8HDT      =1PD13.4,4X'ESTIMATED INITIAL '                     01633000
     2                            'INTEGRATION STEP (REAL)TIME, SEC.'/4X8H01634000
     2TMAX    =1PD13.4,4X,31HTOTAL REAL TIME TO BE RUN, SEC./4X,8HTOLI   01635000
     3=1PD13.4,4X, 31HINTEGRATION TOLERANCE, FRACTION/4X,8HTOLB    =1PD1301636000
     4.4,4X,54HTOLERANCE IN COMPUTING BEARING DISPLACEMENTS, FRACTION/4X01637000
     5,8HTSTOP  =1PD13.4,4X,'THE COMPUTER TIME ALLOWED FOR EACH SET OF D01637100
     5ATA, MINUTES.')                                                    01637200
      WRITE (6,403) GX, GY, TMZ, TMZ1, TMZ2, TMZ3                        01638000
  403 FORMAT (  1H0/,4X,8HGX      =1PD13.4,4X,  46HGRAVITY OR G-LOADING I01639000
     1N X DIRECTION, IN/SEC**2/4X,8HGY      =1PD13.4, 4X,  46HGRAVITY OR 01640000
     2G-LOADING IN Y DIRECTION, IN/SEC**2/4X,8HTMZ     =1PD13.4,4X,   47H01641000
     3EXPONENT FOR SPEED SENSITIVE ROTOR DRIVE TORQUE/4X,8HTMZ1    =1PD1301642000
     4.4,4X,  34HCOEFFICIENT FOR ROTOR DRIVE TORQUE/4X,8HTMZ2    =1PD13.401643000
     5,4X,    52HCOEFFICIENT FOR ROTOR DRIVE TORQUE, (IN-LB-SEC)/RAD./4X01644000
     6,8HTMZ3    =1PD13.4,4X,         43HCOEFFICIENT FOR ROTOR DRIVE TORQU01645000
     7E,  IN-LB )                                                        01646000
      WRITE (6,405)                                                      01647000
  405 FORMAT (1H1/  5X,'DD ARRAY',10X,'OUTSIDE DIAMETERS OF ROTOR SECTIO01648000
```

```
     1NS BETWEEN ADJACENT ROTOR STATIONS, IN.')                         01649000
      WRITE (6,490) (DD(I),I=1,N1)                                      01651000
      WRITE (6,406)                                                     01652000
  406 FORMAT (1H0,4X,'D ARRAY',11X,   'INSIDE DIAMETERS OF ROTOR SECTION01653000
     1S BETWEEN ADJACENT ROTOR STATIONS, IN.')                          01654000
      WRITE (6,490) (D(I),I=1,N1)                                       01655000
      WRITE (6,407)                                                     01656000
  407 FORMAT (1H0,4X,'QL ARRAY', 10X, 'ROTOR SECTION LENGTHS BETWEEN ADJ01657000
     1ACENT ROTOR STATIONS, IN.')                                       01658000
      WRITE (6,490) (QL(I),I=1,N1)                                      01659000
      WRITE (6,408)                                                     01660000
  408 FORMAT (1H0,4X,'DN ARRAY',10X,'MATERIAL DENSITIES OF ROTOR SECTION01661000
     1S BETWEEN ADJACENT STATIONS, LB/IN**3')                           01662000
      WRITE (6,490) (DN(I),I=1,N1)                                      01663000
      WRITE (6,409)                                                     01664000
  409 FORMAT (1H0,4X,'EE ARRAY',10X,   'YOUNGS MODULI OF ROTOR SECTIONS 01665000
     1BETWEEN ADJACENT STATIONS, LB/IN**2')                             01666000
      WRITE (6,490) (EE(I),I=1,N1)                                      01667000
      WRITE (6,410)                                                     01668000
  410 FORMAT (1H0,4X,'GG ARRAY',10X,   'SHEAR MODULI OF ROTOR SECTIONS B01669000
     1ETWEEN ADJACENT STATIONS, LB/IN**2')                              01670000
      WRITE (6,490) (GG(I),I=1,N1)                                      01671000
      WRITE (6,418)                                                     01672000
  418 FORMAT (1H0,4X,'EI  ARRAY',9X, 'DIRECT INPUT OF THE PRODUCTS OF YO01673000
     1UNGS MODULI AND AREA MOMENTS OF INERTIA OF ROTOR'/23X,'SECTIONS BE01674000
     2TWEEN ADJACENT STATIONS, LB-IN**2')                               01675000
      WRITE (6,490)  (EI(I),I=1,N1)                                     01676000
      WRITE (6,419)                                                     01677000
  419 FORMAT (1H0,4X,'GAK ARRAY',9X, 'DIRECT INPUT OF THE PRODUCTS OF SH01678000
     1EAR MODULI, CROSS-SECTIONAL AREAS'/23X'AND RECIPROCALS OF SHEAR ST01679000
     2RESS CONCENTRATION FACTORS BETWEEN ADJACENT STATIONS, LB.')       01680000
      WRITE (6,490) (GAK(I),I=1,N1)                                     01681000
      WRITE (6,411)                                                     01682000
  411 FORMAT(1H0,  4X'AM ARRAY', 10X, 'ADDITIONAL ROTOR MASSES AT ROTOR 01683000
     1STATIONS, (LB-SEC**2)/IN.')                                       01684000
      WRITE (6,404) (AM(I),I=1,NS)                                      01685000
      WRITE (6,412)                                                     01686000
  412 FORMAT (1H0,4X,'AID ARRAY', 9X, 'ADDITIONAL ROTOR TRANSVERSE MASS 01687000
     1MOMENTS OF INERTIA AT ROTOR STATIONS, (LB-IN-SEC**2)')            01688000
      WRITE (6,404) (AID(I),I=1,NS)                                     01689000
      WRITE (6,413)                                                     01690000
  413 FORMAT (1H0,4X,'AIRO ARRAY',8X, 'ADDITIONAL ROTOR POLAR MASS MOMEN01691000
```

```
     1TS OF INERTIA AT ROTOR STATIONS, (LB-IN-SEC**2)')                 01692000
      WRITE (6,404) (AIRO(I),I=1,NS)                                    01693000
      WRITE (6,414)                                                     01694000
  414 FORMAT (1H0,4X,'ECC ARRAY', 9X, 'ROTOR MASS ECCENTRICITIES AT ROTO01695000
     1R STATIONS, IN.')                                                 01696000
      WRITE (6,404) (ECC(I),I=1,NS)                                     01697000
      WRITE (6,415)                                                     01698000
  415 FORMAT (1H0,4X,'ALFA ARRAY',8X, 'PHASE ANGLES FOR ROTOR MASS ECCEN01699000
     1TRICITY VECTORS AT ROTOR STATIONS MEASURED FROM THE'/23X,'INITIAL 01700000
     2ROTOR SPIN ANGULAR POSITION, DEGREES')                            01701000
      WRITE (6,404) (ALFA(I),I=1,NS)                                    01702000
      WRITE (6,416)                                                     01703000
  416 FORMAT (1H0//,5X,10HBETA ARRAY, 8X,       'INITIAL MISALIGNMENTS B01704000
     1ETWEEN THE AXES OF THE MASS MOMENTS OF INERTIA'/23X,  'AND THE ELA01705000
     2STIC AXES AT ROTOR STATIONS, DEGREES')                            01706000
      WRITE (6,404) (BETA(I),I=1,NS)                                    01707000
      WRITE (6,417)                                                     01708000
  417 FORMAT (1H0,4X,'GAMMA ARRAY',7X,'ANGULAR POSITIONS OF THE X-Y PLAN01709000
     1E PROJECTIONS '                                                   01710000
     1             'OF THE AXES OF MASS MOMENTS OF'/23X,'INERTIA AT ROTOR 01711000
     2STATIONS MEASURED FROM THAT AT THE FIRST ROTOR STATION, DEGREES') 01712000
      WRITE (6,404) (GAMMA(I),I=1,NS)                                   01713000
      WRITE (6,422)                                                     01714000
  422 FORMAT (1H0,4X,'CZ ARRAY', 10X,'TORSONAL FRICTION EXPONENTS AT ROT01715000
     1OR STATIONS, DIMENSIONLESS')                                      01716000
      WRITE (6,404) (CZ(I),I=1,NS)                                      01717000
      WRITE (6,423)                                                     01718000
  423 FORMAT (1H0,4X,'CZ1 ARRAY',9X, 'TORSIONAL FRICTION COEFFICIENTS AT01719000
     1 ROTOR STATIONS, DIMENSION OF CZ1(I)*FDOT**CZ(I)'/23X,'IS IN-LB.')01720000
      WRITE (6,404) (CZ1(I),I=1,NS)                                     01721000
      WRITE (6,424)                                                     01722000
  424 FORMAT (1H0,4X,'CZ2 ARRAY',9X, 'TORSIONAL FRICTION COEFFICIENTS AT01723000
     1 ROTOR STATIONS, (IN-LB-SEC)/RAD.')                               01724000
      WRITE (6,404) (CZ2(I),I=1,NS)                                     01725000
      WRITE (6,425)                                                     01726000
  425 FORMAT(1H0,  4X,9HXKF ARRAY,9X, 'WHIRL-FREQUENCY FACTORS FOR STIFF01727000
     1NESS FORCE COEFFICIENTS AT ROTOR STATIONS,'/23X, 'DIMENSIONLESS') 01728000
      WRITE (6,404) (XKF(I),I=1,NS)                                     01729000
      WRITE (6,426)                                                     01730000
  426 FORMAT (1H0,4X,9HXCF ARRAY, 9X, 'WHIRL-FREQUENCY FACTORS FOR DAMPI01731000
     1NG FORCE COEFFICIENTS AT ROTOR STATIONS,'/ 23X,  'DIMENSIONLESS') 01732000
      WRITE (6,404) (XCF(I),I=1,NS)                                     01733000
```

```
      WRITE (6,427)
  427 FORMAT (1H0,4X,10HXKFF ARRAY,8X,'WHIRL-FREQUENCY FACTORS FOR STIFF
     1NESS MOMENT COEFFICIENTS AT ROTOR STATIONS,'/23X,'DIMENSIONLESS')
      WRITE (6,404) (XKFF(I),I=1,NS)
      WRITE (6,428)
  428 FORMAT (1H0,4X,10HXCFF ARRAY,8X,'WHIRL-FREQUENCY FACTORS FOR DAMPI
     1NG MOMENT COEFFICIENTS AT ROTOR STATIONS,'  /23X,'DIMENSIONLESS')
      WRITE (6,404) (XCFF(I),I=1,NS)
      WRITE (6,429)
  429 FORMAT (1H0,  4X,'QK ARRAY',10X,'IN-PHASE STIFFNESS FORCE COEFFICI
     1ENTS AT ROTOR STATIONS, LB/IN.')
      WRITE (6,404) (QK(I),I=1,NS)
      WRITE (6,430)
  430 FORMAT (1H0,4X,8HQC ARRAY,10X,  'IN-PHASE DAMPING FORCE COEFFICIEN
     1TS AT ROTOR STATIONS, (LB-SEC)/IN.')
      WRITE (6,404) (QC(I),I=1,NS)
      WRITE (6,431)
  431 FORMAT (1H0,4X,9HQKP ARRAY, 9X, 'OUT-OF-PHASE STIFFNESS FORCE COEF
     1FICIENTS AT ROTOR STATIONS, LB/IN.')
      WRITE (6,404) (QKP(I),I=1,NS)
      WRITE (6,432)
  432 FORMAT (1H0,4X,9HQCP ARRAY, 9X, 'OUT-OF-PHASE DAMPING FORCE COEFFI
     1CIENTS AT ROTOR STATIONS, (LB-SEC)/IN.')
      WRITE (6,404) (QCP(I),I=1,NS)
      WRITE (6,433)
  433 FORMAT (1H0,4X,10HQKHD ARRAY,8X,'OUT-OF-PHASE WHIRL AND SPIN VELOC
     1ITY SENSITIVE STIFFNESS FORCE COEFFICIENTS AT ROTOR'/23X,'STATIONS
     1,(LB-SEC)/(IN-RAD)')
      WRITE (6,404) (QKHD(I),I=1,NS)
      WRITE (6,434)
  434 FORMAT (1H0,4X,10HQCHD ARRAY,8X,'OUT-OF-PHASE WHIRL AND SPIN VELOC
     1ITY SENSITIVE DAMPING FORCE COEFFICIENTS AT ROTOR'/23X,'STATIONS,(
     2LB-SEC**2)/(IN-RAD)')
      WRITE (6,404) (QCHD(I),I=1,NS)
      WRITE (6,435)
  435 FORMAT (1H0,4X,9HQKF ARRAY,9X,  'IN-PHASE STIFFNESS MOMENT COEFFIC
     1IENTS AT ROTOR STATIONS, IN-LB.')
      WRITE (6,404) (QKF(I),I=1,NS)
      WRITE (6,436)
  436 FORMAT (1H0,4X,9HQCF ARRAY,9X,  'IN-PHASE DAMPING MOMENT COEFFICIE
     1NTS AT ROTOR STATIONS, IN-LB-SEC.')
      WRITE (6,404) (QCF(I),I=1,NS)
```

```
      WRITE (6,437)
  437 FORMAT (1H0,4X,10HQKPF ARRAY,8X,'OUT-OF-PHASE STIFFNESS MOMENT COE
     1FFICIENTS AT ROTOR STATIONS, IN-LB.')
      WRITE (6,404) (QKPF(I),I=1,NS)
      WRITE (6,438)
  438 FORMAT (1H0,4X,10HQCPF ARRAY,8X,'OUT-OF-PHASE DAMPING MOMENT COEFF
     1ICIENTS AT ROTOR STATIONS, IN-LB-SEC.')
      WRITE (6,404) (QCPF(I),I=1,NS)
      WRITE (6,439)
  439 FORMAT (1H0,4X,'QKHDF ARRAY',7X,'OUT-OF-PHASE WHIRL AND SPIN VELOC
     1ITY SENSITIVE STIFFNESS MOMENT COEFFICIENTS AT'/23X,  'ROTOR STATI
     2ONS, (LB-IN-SEC)/RAD.')
      WRITE (6,404) (QKHDF(I),I=1,NS)
      WRITE (6,440)
  440 FORMAT (1H0,4X,'QCHDF ARRAY',7X,'OUT-OF-PHASE WHIRL AND SPIN VELOC
     1ITY SENSITIVE DAMPING MOMENT COEFFICIENTS AT'/23X, 'ROTOR STATIONS
     2, (LB-IN-SEC**2)/RAD.')
      WRITE (6,404) (QCHDF(I),I=1,NS)
      WRITE (6,461) NB
  461 FORMAT (1H1/, 5X,12HNB      =     I3, 3X,'NUMBER OF NON-LINEAR STIFF
     1NESS BEARINGS. (ALLOWABLE RANGE: 2=<NB=<12)')
      WRITE (6,462)
  462 FORMAT(  1H0,4X8HIB ARRAY,10X,'ROTOR STATION NUMBERS FOR NON-LINEA
     1R STIFFNESS BEARINGS')
      WRITE (6,399) (IB(I),I=1,NB)
      WRITE (6,463)
  463 FORMAT(  1H0,4X,'K ARRAY',11X,  'TOTAL NUMBER OF STIFFNESS SECTION
     1S FOR EACH OF THE NON-LINEAR STIFFNESS BEARINGS.'/23X,'(ALLOWABLE
     2RANGE: 1=<K=<6)')
      WRITE (6,399) (K(IB(I)),I=1,NB)
      WRITE (6,441)
  441 FORMAT(  1H0,4X,10HBKMX ARRAY,8X,63HNON-ISOTROPIC MOUNT STIFFNESS
     1COEFFICIENTS IN X-DIRECTION FOR  /23X, 'NON-LINEAR STIFFNESS BEARI
     2NGS, LB/IN.')
      WRITE (6,404) (BKMX(IB(I)),I=1,NB)
      WRITE (6,442)
  442 FORMAT(  1H0,4X,'BKMY ARRAY'8X,  63HNON-ISOTROPIC MOUNT STIFFNESS
     1COEFFICIENTS IN Y-DIRECTION FOR  /23X, 'NON-LINEAR STIFFNESS BEARI
     2NGS, LB/IN.')
      WRITE (6,404) (BKMY(IB(I)),I=1,NB)
      WRITE (6,443)
  443 FORMAT(  1H0,4X'BCMX ARRAY',8X, 'NON-ISOTROPIC MOUNT DAMPING COEFF
```

```
     1ICIENTS IN X-DIRECTION FOR'/23X,'NON-LINEAR STIFFNESS BEARINGS,(LB01818000
     1-SEC)/IN.')                                                       01819000
      WRITE (6,404) (BCMX(IB(I)),I=1,NB)                                01820000
      WRITE (6,444)                                                     01821000
  444 FORMAT(  1H0,4X,10HBCMY ARRAY,8X,68HNON-ISOTROPIC MOUNT DAMPING CO01822000
     1EFFICIENTS IN Y-DIRECTION FOR        /23X,'NON-LINEAR STIFFNESS B01823000
     2EARINGS, (LB-SEC)/IN.')                                           01824000
      WRITE (6,404) (BCMY(IB(I)),I=1,NB)                                01825000
      WRITE (6,445)                                                     01826000
  445 FORMAT(  1H0,4X,9HBCB ARRAY,9X, 'BEARING DAMPING COEFFICIENTS FOR 01827000
     1NON-LINEAR STIFFNESS BEARINGS, (LB-SEC)/IN.')                     01828000
      WRITE (6,404) (BCB(IB(I)),I=1,NB)                                 01829000
      WRITE(6,420)                                                      01830000
  420 FORMAT(1H1//37X,'NONLINEAR BEARING SPECIFICATIONS')               01830500
  471 FORMAT(1H1)                                                       01831000
      DO 19 I=1,NB                                                      01832000
      KI=K(IB(I))                                                       01833000
      WRITE (6,464) I                                                   01834000
  464 FORMAT(1H0,4X, 9HBNB ARRAY,9X,  'ROTOR SPIN-SPEED SENSITIVE BEARIN01835000
     1G STIFFNESS COEFFICIENTS FOR K STIFFNESS'/23X,       'SECTIONS OF 01836000
     2NON-LINEAR STIFFNESS BEARING  NUMBER'I3)                          01837000
   19 WRITE (6,404) (BNB(IB(I),M),M=1,KI)                               01838000
      DO 20 I=1,NB                                                      01839000
      KI=K(IB(I))                                                       01840000
      WRITE (6,465) I                                                   01841000
  465 FORMAT (1H0,4X,9HBBB ARRAY,9X,  'ROTOR SPIN-SPEED SENSITIVE BEARIN01842000
     1G STIFFNESS COEFFICIENTS FOR K STIFFNESS'/23X,       'SECTIONS OF 01843000
     2BEARINGS FOR NON-LINEAR STIFFNESS BEARING NUMBER' I3)             01844000
   20 WRITE (6,404) (BBB(IB(I),M),M=1,KI)                               01845000
      DO 13 I=1,NB                                                      01846000
      KI1=K(IB(I))+1                                                    01847000
      BROB(IB(I),1)=0.0                                                 01848000
      WRITE (6,466) I                                                   01849000
  466 FORMAT (1H0,  4X,'BROB ARRAY',8X,'UPPER BEARING-DISPLACEMENT LIMIT01850000
     1S FOR K STIFFNESS SECTIONS OF BEARING '/23X,'NUMBER'I3,', IN.')   01851000
   13 WRITE (6,404) (BROB(IB(I),M),M=2,KI1)                             01852000
      DO 14 I=1,NB                                                      01853000
      KI=K(IB(I))                                                       01854000
      WRITE (6,467) I                                                   01855000
  467 FORMAT (1H0,4X,9HBKB ARRAY,9X,                  'NON-LINEAR BEARIN01856000
     1G STIFFNESS COEFFICIENTS FOR K STIFFNESS'       /23X, 'SECTIONS OF01857000
     2 BEARING '/23X,'NUMBER'I3)                                        01858000
```

```
   14 WRITE (6,404) (BKB(IB(I),M),M=1,KI)                               01859000
      DO 21 I=1,NB                                                      01860000
      KI=K(IB(I))                                                       01861000
      WRITE (6,468) I                                                   01862000
  468 FORMAT (1H0,4X,9HBHB ARRAY,9X,  'NON-LINEAR BEARING STIFFNESS EXPO01863000
     1NENTS FOR K STIFFNESS SECTIONS OF BEARING'/23X, 'NUMBER'I3)       01864000
   21 WRITE (6,404) (BHB(IB(I),M),M=1,KI)                               01865000
      DO 16 I=1,NB                                                      01866000
      KI=K(IB(I))                                                       01867000
      WRITE (6,469) I                                                   01868000
  469 FORMAT (1H0,4X,9HBDB ARRAY,9X,  'NON-LINEAR BEARING STIFFNESS COEF01869000
     1FICIENTS FOR K STIFFNESS SECTIONS OF BEARING'/23X, 'NUMBER'I3)    01870000
   16 WRITE (6,404) (BDB(IB(I),M),M=1,KI)                               01871000
      DO 17 I=1,NB                                                      01872000
      KI=K(IB(I))                                                       01873000
      WRITE (6,470) I                                                   01874000
  470 FORMAT (1H0,4X,9HBEB ARRAY,9X,  'NON-LINEAR BEARING STIFFNESS COEF01875000
     1FICIENTS FOR K STIFFNESS SECTIONS OF BEARING'/23X, 'NUMBER'I3)    01876000
   17 WRITE (6,404) (BEB(IB(I),M),M=1,KI)                               01877000
      WRITE (6,448)                                                     01878000
  448 FORMAT (1H1 //37X, 35HINITIAL ROTOR MOTION SPECIFICATIONS)         01879000
        ICO=ICOND                                                       01880000
      WRITE(6,450) ICO,CONTIN                                           01881000
  450 FORMAT(1H0/,5X'ICOND  ='I3,7X,'ICOND=1 MEANS STARTING A NEW ROTOR-01881100
     2BEARING CONFIGURATION'/23X'ICOND=0 MEANS READ-IN ALTERNATE INITIAL01881200
     3 CONDITIONS FOR THE SAME'/32X,'ROTOR-BEARING CONFIGURATION'//5X,'C01881300
     4ONTIN ='I3,7X'CONTIN=0 MEANS STARTING A NEW ROTORDYNAMICS ANALYSIS01881400
     5 WITH INITIAL CONDTIONS'/32X,'PROVIDED BY THE STARUP SUBROUTINE'/201881500
     63X,'CONTIN=1 MEANS CONTINUITION OF A PREVIOUS ANALYSIS BY USING TH01881600
     7E PREVIOUS'/32X,'RESULTS ON PUNCHED CARDS AS THE INITIAL CONDTIONS01881610
     8')                                                                01881620
        IF(CONTIN.EQ.1) GO TO 480                                       01881700
      WRITE (6,446)FD,FDOT,WHIVEL,FDOFIX                                01881800
  446 FORMAT(  1H0,4X,                                                  01881900
     1               15HFD      =      1PD13.4,5X'ROTOR SPIN ANGULAR DIS01882000
     1PLACEMENT COORDINATE, DEGREES.'/5X,     15HFDOT    =      1PD13.4,01882100
     25X,'ROTOR SPIN FREQUENCY, RPM.'/5X,15HWHIVEL  =      1PD13.4, 5X,'01882200
     3ROTOR WHIRL FREQUENCY,RPM.'/5X,15HFDOFIX  =      1PD13.4, 5X,     01882300
     4 'A BEARING STIFFNESS SPEED SENSITIVE PARAMETER,  RPM')           01882350
        R=3.14159265358979324/30                                        01882310
        FDOFIX=FDOFIX*R                                                 01882320
        GO TO 500                                                       01882400
```

```
  480 WRITE(6,491)                                                      01882500
  491 FORMAT(1H0,30X,'RESTART ROTOR DEFLECTION AND VELOCITY ARRAY')     01882600
      WRITE(6,492)                                                      01882700
  492 FORMAT(1H0,4X,'ROTOR DEFLECTION VECTOR LENGTH, IN.')              01882800
      WRITE(6,404) (Y(I),I=1,NS)                                        01882900
      WRITE(6,493)                                                      01883000
  493 FORMAT(1H0,4X,'ROTOR DEFLECTION VECTOR ANGULAR POSITION, RAD.')   01883100
        NS2P1=NSP1+NS                                                   01883300
        NS3P1=NS2P1+NS                                                  01883400
        NS4P1=NS3P1+NS                                                  01883500
        NS4P2=NS4P1+1                                                   01883600
      WRITE(6,404) (Y(I),I=NSP1,NS2)                                    01884000
      WRITE(6,494)                                                      01884100
  494 FORMAT(1H0,4X,'ROTOR DEFLECTION VECTOR VELOCITIES')               01884200
      WRITE(6,404) (Y(I),I=NS2P1,NS3)                                   01884300
      WRITE(6,495)                                                      01884400
  495 FORMAT(1H0,4X,'ROTOR DEFLECTION VECTOR ANGULAR VELOCITIES')       01884500
      WRITE(6,404) (Y(I),I=NS3P1,NS4)                                   01884600
      WRITE(6,496) Y(NS4P1),Y(NS4P2)                                    01884700
  496 FORMAT(1H0,4X,15HF       =      1PD13.4,5X,'ROTOR SPIN ANGULAR DIS01884800
     1PLACEMENT COORDINATE, RAD.'/5X, 15HFDOT    =      1PD13.4,5X,'ROTO01884900
     2R SPIN VELOCITY, RAD./SEC.')                                      01885000
  500 WRITE(6,501)                                                      01885010
  501 FORMAT(//1H0,7X,'.....THE END OF INPUT DATA.....')                01885020
      RETURN                                                            01885100
        END                                                             01885200
```

Card Count 362

```
      SUBROUTINE STARUP                                                 00001000
      IMPLICIT REAL*8 (A-H,O-Z)                                         00002000
      REAL Z,TITLE,INPRPM                                               00004000
      DIMENSION             QL(25),DD2(25),D2(25),DD4(25),D4(25),QL2(25)00005000
     1,QLDNDD(25),DN(25),Q6LDND(25),DDPLD(25),DDL(25),Q1LDND(25),QM(25),00006000
     2AM(25),QID(25),AID(25),QIRO(25),AIRO(25)                          00007000
      DIMENSION Z(25),         QME(25),                 ZS(25),SZ(25),ZQ00008000
     1(25),RO(25)                                                       00009000
      DIMENSION C(25,25),B(25,25)                                       00010000
      DIMENSION Y(102)                                                  00011000
      DIMENSION XX(25),YY(25),IB(12)                                    00013000
      DIMENSION AA(51,25),BB(51,25),              IA(51)                00015000
      DIMENSION                     COSFAL(25),SINFAL(25),QMECOS(25),QM00016000
     1ESIN(25)                                                          00017000
      DIMENSION CMESIN(25,25),CMECOS(25,25)                             00018000
      DIMENSION QIRFDO(25)                                              00019000
      DIMENSION  DD(25),D(25),ECC(25),ALFA(25), BETA(25),GAMMA(25)      00022000
      DIMENSION            QZ(25),QZOL(25),ZQOL(25),SZOL(25),ZSOL(25)   00023000
      DIMENSION QK(25),QC(25),QKP(25),QCP(25),QKHD(25),QCHD(25),QKF(2500024000
     1),QCF(25),QKPF(25),QCPF(25),QKHDF(25),QCHDF(25)                   00025000
     3,BKMX(25),BKMY(25),BCMX(25),BCMY(25),BCB(25),BKB(25,6),QKSAVE(25),00026000
     4QCSAVE(25),BNB(25,6),BBB(25,6),BROB(25,7),BHB(25,6),BDB(25,6),    00027000
     4BEB(25,6)                                                         00027500
      DIMENSION    CZ(25),CZ1(25),CZ2(25),BRO(25,25),QRO(25)            00028000
      DIMENSION  CQME(25,25),CONST(50),AS(50,50),PHAROO(25),BS(25,50)   00029000
      DIMENSION  TITLE(36), INPRPM(50)                                  00030000
      COMMON/MAFUF1/DD,D,QL,  NS,NB,IB                                  00031000
      COMMON/MAFUF2/   SZ,QZ,ZQ,SZOL,ZSOL,QZOL,ZQOL,QMLOV,QLL           00032000
      COMMON/MAFUF3/  IB1,IBNB                                          00033000
      COMMON/MAFUF4/ Z                                                  00034000
      COMMON/MAFU14/ FDOFIX                                             00035500
      COMMON/MAFU2/ DN,AM,AID,AIRO,ECC,ALFA,BETA,GAMMA                  00036000
      COMMON/MAFU4/ CZ,CZ1,CZ2                                          00037000
      COMMON/MAFU5/  QK,QC,QKP,QCP,QKHD,QCHD,QKF,QCF,QKPF,QCPF,QKHDF,QC00038000
     1HDF,BKMX,BKMY,BCMX,BCMY,BCB                                       00039000
      COMMON/MAFU6/BNB,BBB,BROB,BKB,BHB,BDB,BEB                         00040000
      COMMON/MAFU8/  TOLB                                               00042000
      COMMON/MAFU9/  C,B                                                00043000
      COMMON/MAFU10/ RO,PHAROO                                          00044000
      COMMON/MAFU11/ FD,FDOT                                            00045000
      COMMON/MAIFU/ Q,S                                                 00046000
      COMMON/MAREAD/ TITLE,INPRPM,       TTT,DT,TMAX, TMZ,TMZ1,TMZ2,TMZ300047000
```

```
     1,     TOLI                                                         00048000
      COMMON/MAFU13/Y                                                    00049000
        FDSAVE=FD                                                        00051000
      FOSAVE=FDOT                                                        00052000
        PI= 3.14159265358979324                                          00059000
        G=386.088                                                        00060000
        U=4./3.                                                          00061000
        V=PI/180.                                                        00062000
        W=PI/(128.*G)                                                    00063000
        E=PI/(8.*G)                                                      00064000
        F=FD*V                                                           00065000
        FDOT=FDOT*V*6.                                                   00066000
        DO 90 I=1,NB                                                     00053000
        IBI=IB(I)                                                        00054000
        QKSAVE(IBI)=QK(IBI)                                              00055000
        QCSAVE(IBI)=QC(IBI)                                              00056000
      QK(IBI)=QK(IBI)+1/(2/(BKMX(IBI)+BKMY(IBI))+1/(          (BKB(IBI,00057000
     1)+BNB(IBI,1)*(FDOT-FDOFIX))))                                      00057500
      IF(BCB(IBI).EQ.0.)BCB(IBI)=1.D-16                                  00058000
      IF(BCMX(IBI).EQ.0.)BCMX(IBI)=1.D-16                                00058100
      IF(BCMY(IBI).EQ.0.)BCMY(IBI)=1.D-16                                00058200
      QC(IBI)=1./(1./BCB(IBI)+2./(BCMX(IBI)+BCMY(IBI)))+QC(IBI)          00058300
   90 QC(IBI)=0.                                                         00058400
      DO 50       I=1,NS                                                 00067000
        ZS(I)=-SZ(I)                                                     00068000
      DD2(I)= DD(I)**2                                                   00069000
      D2 (I)= D(I)**2                                                    00070000
      DD4(I)=DD2(I)**2                                                   00071000
      D4(I) =D2(I)**2                                                    00072000
      QL2(I) = QL(I)**2                                                  00073000
      QLDNDD(I)=QL(I)* DN(I)*(DD2(I)-D2(I))                              00074000
      Q6LDND(I) =               W*QLDNDD(I)                              00075000
      DDPLD(I) = DD2(I)+D2(I)                                            00076000
      DDL(I) = DDPLD(I)+        U*QL2(I)                                 00077000
   50 Q1LDND(I) =             E*QLDNDD(I)                                00078000
      QM(1)=Q1LDND(1) + AM(1)                                            00079000
      QM(NS)=Q1LDND(NS-1) +AM(NS)                                        00080000
      QID(1) = Q6LDND(1)*DDL(1) + AID(1)                                 00081000
      QID(NS)= Q6LDND(NS-1)*DDL(NS-1)    +AID(NS)                        00082000
      QIRO(1)=2.*Q6LDND(1)*DDPLD(1)+AIRO(1)                              00083000
      QIRO(NS)=2.*Q6LDND(NS-1)*DDPLD(NS-1) +AIRO(NS)                     00084000
        NSM1=NS-1                                                        00085000
```

```
      DO 55 I=2,NSM1
      QM(I) =Q1LDND(I-1)+Q1LDND(I) +    AM(I)
      QID(I)=Q6LDND(I-1)*DDL(I-1)+Q6LDND(I)*DDL(I)+AID(I)
   55 QIRO(I) = 2.*(Q6LDND(I-1)*DDPLD(I-1)+Q6LDND(I)*DDPLD(I)) +AIRO(I)
      DO 103 I=1,NS
  103 QME(I)=QM(I)*ECC(I)
      WRITE(6, 80 )
   80 FORMAT (1H1///14X,28HFORCE INFLUENCE COEFFICIENTS)
      DO 81   I=1,NS
   81 WRITE(6,592) I,(C(I,J),J=1,NS)
  592 FORMAT(1H04X3HROWI3/(1P7D15.6))
      WRITE (6,82 )
   82 FORMAT (1H1///14X,29HMOMENT INFLUENCE COEFFICIENTS)
      DO  83  I=1,NS
   83 WRITE(6,592) I,(B(I,J),J=1,NS)
          DO 45 J=1,NS
          AA(J,1)=-C(1,J)*QM(1)+QID(1)*B(1,J)/QL(1)+QID(2)*B(2,J)/(QL(2)
     1    +QL(1))
         AA(J,2)=-C(2,J)*QM(2)-QID(1)*B(1,J)/QL(1)+QID(3)*B(3,J)/(QL(3)+
     1QL(2))
          NSM2=NS-2
          NSM3=NS-3
         AA(J,NSM1)=-C(NSM1,J)*QM(NSM1)-QID(NSM2)*B(NSM2,J)/(QL(NSM2)+QL
     1(NSM3))+QID(NS)*B(NS,J)/QL(NSM1)
   45    AA(J,NS)=-C(NS,J)*QM(NS)-QID(NSM1)*B(NSM1,J)/(QL(NSM1)+QL(NSM2)
     1)-QID(NS)*B(NS,J)/QL(NSM1)
          DO 47 J=1,NS
          DO 47 I=3,NSM2
   47    AA(J,I)=-C(I,J)*QM(I)-QID(I-1)*B(I-1,J)/(QL(I-1)+QL(I-2))+QID(I
     1+1)*B(I+1,J)/(QL(I+1)+QL(I))
         AA(IB1,1) = -(Q-Z(1))*QM(1)-QID(1)/QL(1)-QID(2)/(QL(2)+QL(1))
         AA(IB1,2) = -(Q-Z(2))*QM(2)+QID(1)/QL(1)-QID(3)/(QL(3)+QL(2))
         AA(IB1,NSM1)=-(Q-Z(NSM1))*QM(NSM1)+QID(NSM2)/(QL(NSM2)+QL(NSM3)
     1)-QID(NS)/QL(NSM1)
         AA(IB1,NS) = -(Q-Z(NS))*QM(NS)+QID(NSM1)/(QL(NSM1)+QL(NSM2))+QI
     1D(NS)/QL(NSM1)
          DO 49 I=3,NSM2
   49     AA(IB1,I)=-(Q-Z(I))*QM(I)+QID(I-1)/(QL(I-1)+QL(I-2))-QID(I+1)/
     1(QL(I+1)+QL(I))
        AA(IBNB,1) = -(Z(1)-S)*QM(1)+QID(1)/QL(1)+QID(2)/(QL(2)+QL(1))
        AA(IBNB,2) = -(Z(2)-S)*QM(2)-QID(1)/QL(1)+QID(3)/(QL(3)+QL(2))
        AA(IBNB,NSM1)=-(Z(NSM1)-S)*QM(NSM1)-QID(NSM2)/(QL(NSM2)+QL(NSM3)
```

```
     1)+QID(NS)/QL(NSM1)                                                00139000
      AA(IBNB,NS) = -(Z(NS)-S)*QM(NS)-QID(NSM1)/(QL(NSM1)+QL(NSM2))-QI00140000
     1D(NS)/QL(NSM1)                                                    00141000
      DO 48 I=3,NSM2                                                    00142000
48     AA(IBNB,I)=-(Z(I)-S)*QM(I)-QID(I-1)/(QL(I-1)+QL(I-2))+QID(I+1)/00143000
     1(QL(I+1)+QL(I))                                                   00144000
        NSP1=NS+1                                                       00145000
        NS2 =2*NS                                                       00147000
        DO  53 K=NSP1,NS2                                               00148000
        J=K-NS                                                          00149000
       BB(K,1)=-C(1,J)*QM(1)+QID(1)*B(1,J)/QL(1)+QID(2)*B(2,J)/(QL(2)+00150000
     1QL(1))                                                            00151000
       BB(K,2)=-C(2,J)*QM(2)-QID(1)*B(1,J)/QL(1)+QID(3)*B(3,J)/(QL(3)+00152000
     1QL(2))                                                            00153000
       BB(K,NSM1)=-C(NSM1,J)*QM(NSM1)-QID(NSM2)*B(NSM2,J)/(QL(NSM2)+QL00154000
     1(NSM3))+QID(NS)*B(NS,J)/QL(NSM1)                                  00155000
53     BB(K,NS)=-C(NS,J)*QM(NS)-QID(NSM1)*B(NSM1,J)/(QL(NSM1)+QL(NSM2)00156000
     1)-QID(NS)*B(NS,J)/QL(NSM1)                                        00157000
        DO 65  K=NSP1,NS2                                               00158000
        J=K-NS                                                          00159000
        DO 65  I=3,NSM2                                                 00160000
65     BB(K,I)=-C(I,J)*QM(I)-QID(I-1)*B(I-1,J)/(QL(I-1)+QL(I-2))+QID(I00161000
     1+1)*B(I+1,J)/(QL(I+1)+QL(I))                                      00162000
        NSB1=NS+IB1                                                     00163000
       BB(NSB1,1)=-(Q-Z(1))*QM(1)-QID(1)/QL(1)-QID(2)/(QL(2)+QL(1))     00164000
       BB(NSB1,2)=-(Q-Z(2))*QM(2)+QID(1)/QL(1)-QID(3)/(QL(3)+QL(2))     00165000
        BB(NSB1,NSM1)=-(Q-Z(NSM1))*QM(NSM1)+QID(NSM2)/(QL(NSM2)+QL(NSM00166000
     13))-QID(NS)/QL(NSM1)                                              00167000
       BB(NSB1,NS)=-(Q-Z(NS))*QM(NS)+QID(NSM1)/(QL(NSM1)+QL(NSM2))+QID00168000
     1(NS)/QL(NSM1)                                                     00169000
        DO 66 I=3,NSM2                                                  00170000
66     BB(NSB1,I)=-(Q-Z(I))*QM(I)+QID(I-1)/(QL(I-1)+QL(I-2))-QID(I+1)/00171000
     1(QL(I+1)+QL(I))                                                   00172000
        NSNB=NS+IBNB                                                    00173000
       BB(NSNB,1)=-(Z(1)-S)*QM(1)+QID(1)/QL(1)+QID(2)/(QL(2)+QL(1))     00174000
       BB(NSNB,2)=-(Z(2)-S)*QM(2)-QID(1)/QL(1)+QID(3)/(QL(3)+QL(2))     00175000
        BB(NSNB,NSM1)=-(Z(NSM1)-S)*QM(NSM1)-QID(NSM2)/(QL(NSM2)+QL(NSM00176000
     13))+QID(NS)/QL(NSM1)                                              00177000
       BB(NSNB,NS)=-(Z(NS)-S)*QM(NS)-QID(NSM1)/(QL(NSM1)+QL(NSM2))-QID00178000
     1(NS)/QL(NSM1)                                                     00179000
        DO 56 I=3,NSM2                                                  00180000
56     BB(NSNB,I)=-(Z(I)-S)*QM(I)-QID(I-1)/(QL(I-1)+QL(I-2))+QID(I+1)/00181000
```

```
     1(QL(I+1)+QL(I))
            DO 67 J=1,NS
            DO 67 I=1,NS
            QME(I)=QM(I)*ECC(I)
            CQME(I,J)=C(I,J)*QME(I)
         FALFA = F+ ALFA(I)*V
       SINFAL(I)=DSIN(FALFA)
            QMESIN(I)=QME(I)*SINFAL(I)
       COSFAL(I)=DCOS(FALFA)
            QMECOS(I)=QME(I)*COSFAL(I)
            CMESIN(I,J)=CQME(I,J)*SINFAL(I)
  67        CMECOS(I,J)=CQME(I,J)*COSFAL(I)
            DO 71 I=1,NS
  71        QIRFDO(I)=QIRO(I)*FDOT
            NS2 =2*NS
            FDOTSQ=FDOT**2
       FF=0.0
       DO 690 J=1,NS2
       DO 690 I=1,NS2
 690      AS(J,I)=0
       DO 700 J=1,NS
          BRO(J,1)= B(1,J)*QIRO(1)/QL(1) +B(2,J)*QIRO(2)/(QL(1)+QL(2))
          BRO(J,2)=-B(1,J)*QIRO(1)/QL(1) +B(3,J)*QIRO(3)/(QL(2)+QL(3))
          BRO(J,NSM1)=-B(NSM2,J)*QIRO(NSM2)/(QL(NSM3)+QL(NSM2))+ B(NS,J)*
     1QIRO(NS)/QL(NSM1)
 700      BRO(J,NS)= -B(NSM1,J)*QIRO(NSM1)/(QL(NSM2)+QL(NSM1)) -B(NS,J)*QI
     1RO(NS)/QL(NSM1)
       DO 710 J=1,NS
       DO 710 I=3,NSM2
 710      BRO(J,I)= -B(I-1,J)*QIRO(I-1)/ (QL(I-2)+QL(I-1))+ B(I+1,J)*QIRO(
     1I+1)/(QL(I)+QL(I+1))
          QRO(1)= QIRO(1)/QL(1)+ QIRO(2)/ (QL(1)+QL(2))
          QRO(2)= -QIRO(1)/QL(1) +QIRO(3)/ (QL(2)+QL(3))
          QRO(NSM1)= -QIRO(NSM2)/ (QL(NSM3)+QL(NSM2)) +QIRO(NS)/QL(NSM1)
          QRO(NS) = -QIRO(NSM1)/ (QL(NSM2)+QL(NSM1)) -QIRO(NS)/ QL(NSM1)
       DO 720 I=3,NSM2
 720      QRO(I)= -QIRO(I-1) /(QL(I-2)+QL(I-1)) +QIRO(I+1)/ (QL(I)+QL(I+1)
     1)
       DO 730 J=1,NS
          CONST(J)=0
       DO 740 I=1,NS
 740      CONST(J)=CONST(J)-CMECOS(I,J)
```

```
  730    CONST(J)= FDOTSQ*CONST(J)                                        00227000
         CONST(IB1)=0                                                     00228000
         CONST(IBNB)=0                                                    00229000
       DO 750 I=1,NS                                                      00230000
         CONST(IB1)=CONST(IB1)  -QZ(I)* QMECOS(I)                         00231000
  750    CONST(IBNB)=CONST(IBNB) -ZS(I) *QMECOS(I)                        00232000
         CONST(IB1)=FDOTSQ*CONST(IB1)                                     00233000
         CONST(IBNB)= FDOTSQ*CONST(IBNB)                                  00234000
       DO 760 J=NSP1,NS2                                                  00235000
         K=J-NS                                                           00236000
         CONST(J)=0                                                       00237000
       DO 770 I=1,NS                                                      00238000
  770    CONST(J)= CONST(J)-CMESIN(I,K)                                   00239000
  760    CONST(J)= FDOTSQ*CONST(J)                                        00240000
         CONST(NSB1)=0                                                    00241000
         CONST(NSNB)=0                                                    00242000
       DO 775 I=1,NS                                                      00243000
         CONST(NSB1)= CONST(NSB1) -QZ(I)*QMESIN(I)                        00244000
  775    CONST(NSNB)= CONST(NSNB) -ZS(I)*QMESIN(I)                        00245000
         CONST(NSB1)= FDOTSQ* CONST(NSB1)                                 00246000
         CONST(NSNB)= FDOTSQ* CONST(NSNB)                                 00247000
       DO 790 J=1,NS                                                      00248000
       DO 790 I=1,NS                                                      00249000
         AS(J,I) =-C(I,J)*QK(I) +FDOTSQ*BRO(J,I)                          20052000
  790    BS(J,I)=AS(J,I)                                                  20052500
       DO 800 J=1,NS                                                      00251000
       DO 800 I=NSP1,NS2                                                  00252000
         M=I-NS                                                           00253000
         AS(J,I)= C(M,J)*FDOT*QC(M)                                       20055000
  800    BS(J,I)=-AS(J,I)                                                 20055500
       DO 810 J=1,NS                                                      00255000
         AS(J,IB1)= AS(J,IB1)+1-ZS(J)/QLL                                 00256000
         BS(J,IB1)=AS(J,IB1)                                              20057500
         AS(J,IBNB)=AS(J,IBNB)+ZS(J)/QLL                                  00257000
         BS(J,IBNB)=AS(J,IBNB)                                            20058500
         AS(J,J)    = AS(J,J)-1                                           20059000
  810    BS(J,J)=AS(J,J)                                                  20059500
       DO 820 I=1,NS                                                      00259000
         AS(IB1,I)=-QZ(I)*QK(I)- FDOTSQ*QRO(I)                            20065000
  820    BS(IB1,I)=AS(IB1,I)                                              20065500
       DO 830 I=NSP1,NS2                                                  00261000
         M=I-NS                                                           00262000
```

```
       AS(IB1,I)= QZ(M)*FDOT*QC(M)                                      20068000
  830  BS(IB1,I)=-AS(IB1,I)                                             20068500
      DO 840 I=1,NS                                                     00264000
       AS(IBNB,I)= -ZS(I)*QK(I) +FDOTSQ* QRO(I)                         20075000
  840  BS(IBNB,I)=AS(IBNB,I)                                            20075500
      DO 850 I=NSP1,NS2                                                 00266000
       M=I-NS                                                           00267000
       AS(IBNB,I)= ZS(M)*QC(M)*FDOT                                     20077000
  850  BS(IBNB,I)=-AS(IBNB,I)                                           20077500
      DO 860 J=1,NS                                                     00271000
      DO 860 I=1,NS                                                     00272000
  860  AS(J,I)= AS(J,I) -FDOTSQ*AA(J,I)                                 00273000
      DO 950 J=1,NS                                                     00297000
      DO 950 I=1,NS                                                     00298000
       K=J+NS                                                           00299000
  950  BS(J,I)= BS(J,I) -FDOTSQ*BB(K,I)                                 00300000
       DO 630 J=NSP1,NS2                                                00301000
       M=J-NS                                                           00302000
      DO 630 I=NSP1,NS2                                                 00303000
       K=I-NS                                                           00304000
        AS(J,I)=BS(M,K)                                                 00305000
  630  AS(J,K)=BS(M,I)                                                  00306000
      KL=ISIMDD(50,NS2,1,AS,CONST,FF,IA)                                00321000
       DO 620 I=1,NS                                                    00322000
       IPNS=I+NS                                                        00323000
        XX(I)=AS(I,1)                                                   00324000
  620   YY(I)=AS(IPNS,1)                                                00325000
      WRITE(6,471)                                                      00325500
       WRITE(6,600)                                                     00326000
  600 FORMAT(1H0,4X,'COMPUTED STARTING XX ARRAY')                       00327000
       WRITE(6,404) (XX(I),I=1,NS)                                      00328000
       WRITE(6,610)                                                     00329000
  610 FORMAT(1H0,4X,'COMPUTED STARTING YY ARRAY')                       00330000
       WRITE(6,404) (YY(I),I=1,NS)                                      00331000
      DO 870 I=1,NS                                                     00332000
      RO(I)=DSQRT(XX(I)**2+YY(I)**2)                                    00333000
      PHAROO(I)=DATN2D(YY(I),XX(I))                                     00334000
  870  IF(PHAROO(I).LT.0) PHAROO(I)=360+PHAROO(I)                       00335000
  404 FORMAT (7X,1P6E13.4)                                              00336000
  471 FORMAT(1H1)                                                       00337000
      WRITE (6,447)                                                     00338000
  447 FORMAT(1H0,4X,                                                8HRO 00339000
```

```
     1ARRAY,10X,'INITIAL DISPLACEMENT VECTORS (WHICH CAN NOT BE ZERO) FR00340000
     1OM THEIR RESPECTIVE ZERO LOAD'/23X,'POSITIONS, IN.')               00341000
      WRITE (6,404) (RO(I),I=1,NS)                                      00342000
      WRITE (6,449)                                                     00343000
  449 FORMAT(1H0,4X'PHAROO ARRAY',6X'INITIAL PHASE ANGLES FOR THE DISPLA00344000
     1CEMENT VECTORS, DEGREES.'/23X,'THE PHASE ANGLES CAN NOT BE ZERO OR00345000
     2 MULTIPLES OF 90 DEGREES.')                                       00346000
      WRITE (6,404) (PHAROO(I),I=1,NS)                                  00347000
      WRITE(6,471)                                                      00350000
      IF (KL.NE.3)  GO TO 560                                           00351000
      WRITE (6,540) KL                                                  00352000
  540 FORMAT(1H1/10X,4HKL =I1)                                          00353000
      CALL EXIT                                                         00354000
  560   FD=FDSAVE                                                       00355000
      FDOT=FOSAVE                                                       00356000
        DO 980 I=1,NB                                                   00357000
        IBI=IB(I)                                                       00358000
        QK(IBI)=QKSAVE(IBI)                                             00359000
  980   QC(IBI)=QCSAVE(IBI)                                             00360000
      RETURN                                                            00361000
      END                                                               00362000
```

Card Count 315

```
      SUBROUTINE FUND(N,T,Y,BD)                                         00102000
      IMPLICIT REAL*8 (A-H,O-Z)                                         00103000
      REAL Z,TITLE,INPRPM                                               00105000
      DIMENSION  QL(25),DN(25),                                         00106000
     1QM(25),Z(25),QME(25),SZ(25),ZQ(25),QMZ(25),ROSQ(25),       Y(102),00107000
     2AM(25),QID(25),AID(25),QIRO(25),AIRO(25)                          00108000
     3,DELTX(25),DELTY(25),COSAB(25),SINAB(25),XDOT(25),YDOT(25),XX(25),00109000
     4YY(25),IB(12),XBDOT(25),YBDOT(25),BCB(25),BCMX(25),BCMY(25),KK(25)00110000
     5,C(25,25),B(25,25),BKMX(25),BKMY(25)                              00111000
        DIMENSION BNB(25,6),BBB(25,6),BROB(25,7),BKB(25,6),BHB(25,6),BDB00122000
     1  (25,6),BEB(25,6)                                                00123000
        DIMENSION AA(51,25),BB(51,25),AROT(51,51),IA(51)                00124000
     1,CC(51),DRO(51),DSTA(51)                                          00125000
        DIMENSION                    COSFAL(25),SINFAL(25),QMECOS(25),QM00126000
     1ESIN(25),QIDBSN(25),QIDBCS(25),QMEFSQ(25),QKHDF(25),QCHDF(25),   Q00127000
     2IRFDO(25),QKHDFF(25),QCHDFF(25),BRGROP(25),XB(25),YB(25),AMX(25), 00131000
     3AMY(25),CZ(25),CZ1(25),CZ2(25),COSFGA(25),SINFGA(25)              00131100
        DIMENSION AFX(25),AFY(25),AMX1(25),AMY1(25),AMX2(25),AMY2(25)   00132000
        DIMENSION BD(102),WHRVLO(25),  QKHDD(25),QCHDD(25)              00133000
        DIMENSION  DD(25),D(25),ECC(25),ALFA(25), BETA(25),GAMMA(25),   00134000
     1  A(25)                                                           00135000
        DIMENSION            QZ(25),QZOL(25),ZQOL(25),SZOL(25),ZSOL(25) 00136000
        DIMENSION  XKF(25),XCF(25),XKFF(25),XCFF(25),  QK(25),QC(25),QKP00139000
     1(25),QCP(25),QKHD(25),QCHD(25),QKF(25),QCF(25),QKPF(25),QCPF(25)  00140000
      DIMENSION  TITLE(36), INPRPM(50)                                  00143000
       COMMON/MAFUF1/DD,D,QL,  NS,NB,IB                                 00144000
      COMMON/MAFUF2/   SZ,QZ,ZQ,SZOL,ZSOL,QZOL,ZQOL,QMLOV,QLL           00145000
       COMMON/MAFUF3/  IB1,IBNB                                         00146000
      COMMON/MAFUF4/ Z                                                  00147000
       COMMON/MAFU14/ FDOFIX                                            00147600
       COMMON/MAFU0/ ITIM,INT,  KK                                      00148000
       COMMON/MAFU2/ DN,AM,AID,AIRO,ECC,ALFA,BETA,GAMMA,GX,GY           00149000
       COMMON/MAFU4/ CZ,CZ1,CZ2,  XKF,XCF,XKFF,XCFF                     00150000
       COMMON/MAFU5/  QK,QC,QKP,QCP,QKHD,QCHD,QKF,QCF,QKPF,QCPF,QKHDF,QC00151000
     1HDF,    BKMX,BKMY,BCMX,BCMY,BCB                                   00152000
       COMMON/MAFU6/ BNB,BBB,BROB,BKB,BHB,BDB,BEB                       00153000
      COMMON /MAFU7/ROSQ,   WHRVLO,DELTX,DELTY                          00154000
      COMMON/MAFU8/  TOLB                                               00155000
        COMMON/MAFU9/  C,B                                              00156000
     X      /MAFU12/QM,QID,QIRO                                         00157000
     1       /MAFU11/FD,FDOT                                            00157100
      COMMON/MAIFU/ Q,S                                                 00158000
```

```
      COMMON/MAREAD/ TITLE,INPRPM,       TTT,DT,TMAX, TMZ,TMZ1,TMZ2,TMZ300159000
     1,      TOLI                                                       00160000
     2      /SFP2/NSM1,NSM2,NSM3,NSP1,NS2,NS3,NS4,NS4P1,NS4P2,NS2P1     00160010
     3      /GARBG5/QME                                                 00160020
     4      /GARBG6/QMZ,AA,BB,AROT,CC,DRO,DSTA,                         00160030
     5COSAB,SINAB,QIRFDO,QKHDFF,QCHDFF,BRGROP,AFX,AFY,AMX1,AMX2,AMY1,   00160040
     6AMY2,QCHDD,QKHDD,A,AMX,AMY,COSFGA,SINFGA,                         00160050
     7TRIG,COSFAL,SINFAL,QMECOS,QMESIN,QIDBET,QIDBSN,QIDBCS,QMEFSQ,     00160060
     8IA                                                                00160070
     9      /GARBG7/XX,YY,XB,YB,XDOT,YDOT,XBDOT,YBDOT                   00160080
      IF (INT .EQ.2) GO TO 406                                          00161000
        PI= 3.14159265358979324                                         00162000
        V=PI/180.                                                       00165000
      DO 103 I=1,NS                                                     00200000
  103 QME(I)=QM(I)*ECC(I)                                               00201000
          INT=2                                                         00203000
  406 DO 2 I=1,N                                                        00203000
      IF(Y(I).GE.1.D14) GO TO 3                                         00204000
    2 CONTINUE                                                          00205000
      GO TO 4                                                           00206000
    3 RETURN                                                            00207000
    4 FDOT=Y(NS4P2)                                                     00210000
          DO 79 I=1,NS                                                  00211000
          INS=I+NS                                                      00212000
      I2NS=I+NS2                                                        00213000
      I3NS=I+NS3                                                        00214000
      COSAB(I)=DCOS(Y(INS))                                             00215000
      SINAB(I)=DSIN(Y(INS))                                             00216000
      XNUTR=Y(I)*COSAB(I)                                               00217000
      YNUTR=Y(I)*SINAB(I)                                               00218000
      XX(I)=XNUTR-DELTX(I)                                              00219000
      YY(I)=YNUTR-DELTY(I)                                              00220000
          XDOT(I)=Y(I2NS)*COSAB(I)-Y(I)*Y(I3NS)*SINAB(I)                00221000
          YDOT(I)=Y(I2NS)*SINAB(I)+Y(I)*Y(I3NS)*COSAB(I)                00222000
   79     WHRVLO(I)=Y(I3NS)                                             00223000
   80 CALL BRGXY                                                        00223100
          DO 45 J=1,NS                                                  00312000
          AA(J,1)=-C(1,J)*QM(1)+QID(1)*B(1,J)/QL(1)+QID(2)*B(2,J)/(QL(2)00313000
     1    +QL(1))                                                       00314000
         AA(J,2)=-C(2,J)*QM(2)-QID(1)*B(1,J)/QL(1)+QID(3)*B(3,J)/(QL(3)+00315000
     1QL(2))                                                            00316000
         AA(J,NSM1)=-C(NSM1,J)*QM(NSM1)-QID(NSM2)*B(NSM2,J)/(QL(NSM2)+QL00319000
```

```
     1(NSM3))+QID(NS)*B(NS,J)/QL(NSM1)                                  00320000
   45    AA(J,NS)=-C(NS,J)*QM(NS)-QID(NSM1)*B(NSM1,J)/(QL(NSM1)+QL(NSM2)00321000
     1)-QID(NS)*B(NS,J)/QL(NSM1)                                        00322000
          DO 47 J=1,NS                                                  00323000
          DO 47 I=3,NSM2                                                00324000
   47    AA(J,I)=-C(I,J)*QM(I)-QID(I-1)*B(I-1,J)/(QL(I-1)+QL(I-2))+QID(I00325000
     1+1)*B(I+1,J)/(QL(I+1)+QL(I))                                      00326000
         AA(IB1,1) = -(Q-Z(1))*QM(1)-QID(1)/QL(1)-QID(2)/(QL(2)+QL(1))  00327000
         AA(IB1,2) = -(Q-Z(2))*QM(2)+QID(1)/QL(1)-QID(3)/(QL(3)+QL(2))  00328000
         AA(IB1,NSM1)=-(Q-Z(NSM1))*QM(NSM1)+QID(NSM2)/(QL(NSM2)+QL(NSM3)00329000
     1)-QID(NS)/QL(NSM1)                                                00330000
         AA(IB1,NS) = -(Q-Z(NS))*QM(NS)+QID(NSM1)/(QL(NSM1)+QL(NSM2))+QI00331000
     1D(NS)/QL(NSM1)                                                    00332000
          DO 49 I=3,NSM2                                                00333000
   49     AA(IB1,I)=-(Q-Z(I))*QM(I)+QID(I-1)/(QL(I-1)+QL(I-2))-QID(I+1)/00334000
     1(QL(I+1)+QL(I))                                                   00335000
        AA(IBNB,1) = -(Z(1)-S)*QM(1)+QID(1)/QL(1)+QID(2)/(QL(2)+QL(1))  00336000
        AA(IBNB,2) = -(Z(2)-S)*QM(2)-QID(1)/QL(1)+QID(3)/(QL(3)+QL(2))  00337000
        AA(IBNB,NSM1)=-(Z(NSM1)-S)*QM(NSM1)-QID(NSM2)/(QL(NSM2)+QL(NSM3)00338000
     1)+QID(NS)/QL(NSM1)                                                00339000
        AA(IBNB,NS) = -(Z(NS)-S)*QM(NS)-QID(NSM1)/(QL(NSM1)+QL(NSM2))-QI00340000
     1D(NS)/QL(NSM1)                                                    00341000
        DO 48 I=3,NSM2                                                  00342000
   48    AA(IBNB,I)=-(Z(I)-S)*QM(I)-QID(I-1)/(QL(I-1)+QL(I-2))+QID(I+1)/00343000
     1(QL(I+1)+QL(I))                                                   00344000
          DO  53 K=NSP1,NS2                                             00348000
          J=K-NS                                                        00349000
         BB(K,1)=-C(1,J)*QM(1)+QID(1)*B(1,J)/QL(1)+QID(2)*B(2,J)/(QL(2)+00350000
     1QL(1))                                                            00351000
         BB(K,2)=-C(2,J)*QM(2)-QID(1)*B(1,J)/QL(1)+QID(3)*B(3,J)/(QL(3)+00352000
     1QL(2))                                                            00353000
         BB(K,NSM1)=-C(NSM1,J)*QM(NSM1)-QID(NSM2)*B(NSM2,J)/(QL(NSM2)+QL00354000
     1(NSM3))+QID(NS)*B(NS,J)/QL(NSM1)                                  00355000
   53    BB(K,NS)=-C(NS,J)*QM(NS)-QID(NSM1)*B(NSM1,J)/(QL(NSM1)+QL(NSM2)00356000
     1)-QID(NS)*B(NS,J)/QL(NSM1)                                        00357000
          DO 65  K=NSP1,NS2                                             00358000
          J=K-NS                                                        00359000
          DO 65  I=3,NSM2                                               00360000
   65    BB(K,I)=-C(I,J)*QM(I)-QID(I-1)*B(I-1,J)/(QL(I-1)+QL(I-2))+QID(I00361000
     1+1)*B(I+1,J)/(QL(I+1)+QL(I))                                      00362000
          NSB1=NS+IB1                                                   00363000
         BB(NSB1,1)=-(Q-Z(1))*QM(1)-QID(1)/QL(1)-QID(2)/(QL(2)+QL(1))   00364000
```

```
      BB(NSB1,2)=-(Q-Z(2))*QM(2)+QID(1)/QL(1)-QID(3)/(QL(3)+QL(2))       00365000
       BB(NSB1,NSM1)=-(Q-Z(NSM1))*QM(NSM1)+QID(NSM2)/(QL(NSM2)+QL(NSM00366000
 13))-QID(NS)/QL(NSM1)                                                  00367000
      BB(NSB1,NS)=-(Q-Z(NS))*QM(NS)+QID(NSM1)/(QL(NSM1)+QL(NSM2))+QID00368000
 1(NS)/QL(NSM1)                                                         00369000
       DO 66 I=3,NSM2                                                   00370000
66    BB(NSB1,I)=-(Q-Z(I))*QM(I)+QID(I-1)/(QL(I-1)+QL(I-2))-QID(I+1)/00371000
 1(QL(I+1)+QL(I))                                                       00372000
       NSNB=NS+IBNB                                                     00373000
      BB(NSNB,1)=-(Z(1)-S)*QM(1)+QID(1)/QL(1)+QID(2)/(QL(2)+QL(1))      00374000
      BB(NSNB,2)=-(Z(2)-S)*QM(2)-QID(1)/QL(1)+QID(3)/(QL(3)+QL(2))      00375000
       BB(NSNB,NSM1)=-(Z(NSM1)-S)*QM(NSM1)-QID(NSM2)/(QL(NSM2)+QL(NSM00376000
 13))+QID(NS)/QL(NSM1)                                                  00377000
      BB(NSNB,NS)=-(Z(NS)-S)*QM(NS)-QID(NSM1)/(QL(NSM1)+QL(NSM2))-QID00378000
 1(NS)/QL(NSM1)                                                         00379000
       DO 56 I=3,NSM2                                                   00380000
56    BB(NSNB,I)=-(Z(I)-S)*QM(I)-QID(I-1)/(QL(I-1)+QL(I-2))+QID(I+1)/00381000
 1(QL(I+1)+QL(I))                                                       00382000
       DO 76 I=1,NS                                                     00383000
     FGAMMA   = Y(NS4P1)+GAMMA(I)*V                                     00384000
     COSFGA(I)=DCOS(FGAMMA)                                             00385000
   SINFGA(I)=DSIN(FGAMMA)                                               00386000
   FALFA=Y(NS4P1)+V*ALFA(I)                                             00386020
   SINFAL(I)=DSIN(FALFA)                                                00386030
   COSFAL(I)=DCOS(FALFA)                                                00386040
   QMESIN(I)=QME(I)*SINFAL(I)                                           00386050
   QMECOS(I)=QME(I)*COSFAL(I)                                           00386060
   BETA(I)=V*BETA(I)                                                    00386070
   QIDBET=QID(I)*BETA(I)                                                00386080
   QIDBSN(I)=QIDBET*SINFGA(I)                                           00386090
76 QIDBCS(I)=QIDBET*COSFGA(I)                                           00386100
       DO 67 J=1,NS                                                     00387000
       CC(J)=0.0                                                        00388000
       DO 67 I=1,NS                                                     00389000
67 CC(J)=CC(J)+B(I,J)*QIDBSN(I)+C(I,J)*QME(I)*SINFAL(I)                 00406000
      CC(IB1)=0.0                                                       00407000
       CC(IBNB)=0.0                                                     00408000
       DO 78 I=1,NS                                                     00409000
   QMZ(I)=Q-Z(I)                                                        00410000
       CIB1=QMZ(I)*QMESIN(I)-QIDBSN(I)                                  00411000
       CIBNB=(Z(I)-S)*QMESIN(I)+QIDBSN(I)                               00412000
       CC(IB1)=CC(IB1)+CIB1                                             00413000
```

```
 78      CC(IBNB)=CC(IBNB)+CIBNB                                          00414000
         DO  71 J=NSP1,NS2                                                00416000
      K=J-NS                                                              00417000
        CC(J)=0.0                                                         00418000
         DO 71 I=1,NS                                                     00419000
 71 CC(J)=CC(J)-COSFAL(I)*C(I,K)*QME(I)-B(I,K)*QIDBCS(I)                  00421000
        CC(NSB1)=0.0                                                      00422000
        CC(NSNB)=0.0                                                      00423000
         DO 73 I=1,NS                                                     00424000
         CIB1=-QMZ(I)*QMECOS(I)+QIDBCS(I)                                 00425000
         CIBNB=-(Z(I)-S)*QMECOS(I)-QIDBCS(I)                              00426000
        CC(NSB1)=CC(NSB1)+CIB1                                            00427000
 73     CC(NSNB)=CC(NSNB)+CIBNB                                           00428000
         CC(NS2P1)=0.0                                                    00430000
         DO 75 I=1,NS                                                     00431000
        AA(NS2P1,I)=-QMESIN(I)                                            00432000
        BB(NS2P1,I)=QMECOS(I)                                             00433000
         C2NP1=QIRO(I)+QME(I)*ECC(I)                                      00434000
 75      CC(NS2P1)=CC(NS2P1)+C2NP1                                        00435000
         FDOTSQ=FDOT**2                                                   00436000
         DO 201 I=1,NS                                                    00437000
     I3NS=I+NS3                                                           00438000
         QMEFSQ(I)=QME(I)*FDOTSQ                                          00439000
         QKHDD(I)=QKHD(I)*(FDOT-XKF(I)*Y(I3NS))                           00440000
         QCHDD(I)=QCHD(I)*(FDOT-XCF(I)*Y(I3NS))                           00441000
         QIRFDO(I)=QIRO(I)*FDOT                                           00444000
         QKHDFF(I)=QKHDF(I)*(FDOT-XKFF(I)*Y(I3NS))                        00445000
201      QCHDFF(I)=QCHDF(I)*(FDOT-XCFF(I)*Y(I3NS))                        00446000
      FOSAVE=FDOT                                                         00446500
      FDOT=FDOT-FDOFIX                                                    00446550
         DO 202 I=1,NB                                                    00447000
         IBI=IB(I)                                                        00448000
         KKB=KK(IBI)                                                      00449000
     BRO=DSQRT(XB(IBI)**2+YB(IBI)**2)                                     00451000
202  BRGROP(IBI)=(FDOT*(BNB(IBI,KKB)+BBB(IBI,KKB)*(BRO-BROB(IBI,KKB)))+00452000
    1BKB(IBI,KKB))*(BRO**(BHB(IBI,KKB)-1.)+BDB(IBI,KKB)+BEB(IBI,KKB)/   00454000
    2BRO)                                                                 00455000
        FDOT=FOSAVE                                                       00455500
         DO 63 I=1,NS                                                     00456000
 63      A(I)=0.0                                                         00457000
         DO 54 I=1,NB                                                     00458000
         IBI=IB(I)                                                        00459000
```

```
 54      A(IBI)=BRGROP(IBI)*XB(IBI)+BCB(IBI)*XBDOT(IBI)
         DO 203 I=1,NS
203      AFX(I)=-(QM(I)*GX-QMEFSQ(I)*COSFAL(I)+QK(I)*XX(I)+QC(I)*XDOT(I
    1)+QKP(I)*YY(I)+QCP(I)*YDOT(I)+QKHDD(I)*YY(I)+QCHDD(I)*YDOT(I))-A(I
    2)
         DO 204 I=1,NB
         IBI=IB(I)
204      A(IBI)=BRGROP(IBI)*YB(IBI)+BCB(IBI)*YBDOT(IBI)
         DO 205 I=1,NS
     TRIG=(QID(I)-QIRO(I))*FDOTSQ*BETA(I)
         AFY(I)=-(QM(I)*GY-QMEFSQ(I)*SINFAL(I)+QK(I)*YY(I)+QC(I)*YDOT(I
    1)-QKP(I)*XX(I)-QCP(I)*XDOT(I)-QKHDD(I)*XX(I)-QCHDD(I)*XDOT(I))-A(I
    2)
     AMY1(I)=TRIG*COSFGA(I)
205  AMX1(I)=-TRIG*SINFGA(I)
         I=1
         J=2
         K=1
 57      AMY2(I)=-1./QL(K)*(QIRFDO(I)*(YDOT(J)-YDOT(K))+QKF(I)*(XX(J)-X
    1X(K))+
    1     QCF(I)*(XDOT(J)-XDOT(K))+QKPF(I)*(YY(J)-YY(K))+QCPF(I)*(YDOT(
    2J)-YDOT(K))+QKHDFF(I)*(YY(J)-YY(K))+QCHDFF(I)*(YDOT(J)-YDOT(K)))
         AMX2(I)=1./QL(K)*(-QIRFDO(I)*(XDOT(J)-XDOT(K))+QKF(I)*(YY(J)-Y
    1Y(K))+QCF(I)*(YDOT(J)-YDOT(K))-QKPF(I)*(XX(J)-XX(K))-QCPF(I)*(XDOT
    2(J)-XDOT(K))-QKHDFF(I)*(XX(J)-XX(K))-QCHDFF(I)*(XDOT(J)-XDOT(K)))
        IF (I.GT.1) GO TO 58
         I=NS
         J=NS
     K=NSM1
         GO TO  57
 58 DO 59 I=2,NSM1
         AMY2(I)= -1./(QL(I-1)+QL(I))*(QIRFDO(I)*(YDOT(I+1)-YDOT(I-1))+
    1QKF(I)*(XX(I+1)-XX(I-1))+QCF(I)*(XDOT(I+1)-XDOT(I-1))+QKPF(I)*(YY(
    2I+1)-YY(I-1))+QCPF(I)*(YDOT(I+1)-YDOT(I-1))+QKHDFF(I)*(YY(I+1)-YY(
    3I-1))+QCHDFF(I)*(YDOT(I+1)-YDOT(I-1)))
 59      AMX2(I)=1./(QL(I-1)+QL(I))*(-QIRFDO(I)*(XDOT(I+1)-XDOT(I-1))+Q
    1KF(I)*(YY(I+1)-YY(I-1))+QCF(I)*(XDOT(I+1)-XDOT(I-1))-QKPF(I)*(XX(I
    2+1)-XX(I-1))-QCPF(I)*(XDOT(I+1)-XDOT(I-1))-QKHDFF(I)*(XX(I+1)-XX(I
    3-1))-QCHDFF(I)*(XDOT(I+1)-XDOT(I-1)))
          DO 60 I=1,NS
          AMY(I)=AMY1(I)+AMY2(I)
 60       AMX(I)=AMX1(I)+AMX2(I)
```

```
          DO 69 J=1,NS
      DST=0.
          DO 44 I=1,NS
          DSTAT = -C(I,J)*AFX(I)-B(I,J)*AMY(I)
   44 DST=DST+DSTAT
   69 DSTA(J)=DST-XX(IB1)-(XX(IBNB)-XX(IB1))*(Z(J)-S)/QLL+XX(J)
          DSTA(IB1)=0.0
          DSTA(IBNB)=0.0
          DO 43 I=1,NS
          DST1  =  -((Q-Z(I))*AFX(I)-AMY(I))
          DSTA(IB1)=DSTA(IB1)+DST1
          DSTNB=-(Z(I)-S)*AFX(I)-AMY(I)
   43     DSTA(IBNB)=DSTA(IBNB)+DSTNB
          DO 42 J=NSP1,NS2
        K=J-NS
      DST=0.
          DO 41 I=1,NS
          DSTAT=-C(I,K)*AFY(I)+B(I,K)*AMX(I)
   41 DST=DST+DSTAT
   42 DSTA(J)=DST-YY(IB1)-(YY(IBNB)-YY(IB1))*(Z(K)-S)/QLL+YY(K)
          DSTA(NSB1)=0.0
          DSTA(NSNB)=0.0
          DO 70 I=1,NS
         DSTN1=-(Q-Z(I))*AFY(I)-AMX(I)
          DSTA(NSB1)=DSTA(NSB1)+DSTN1
          DSTNB=-(Z(I)-S)*AFY(I)+AMX(I)
   70     DSTA(NSNB)=DSTA(NSNB)+DSTNB
          DSTA(NS2P1)=           TMZ1*FDOT**TMZ+TMZ2*FDOT+TMZ3
          DO 72 I=1,NS
      DSR=-(QME(I)*(GY*COSFAL(I)-GX*SINFAL(I))+(CZ1(I)*FDOT**CZ(I)+
     1CZ2(I)*FDOT))
   72     DSTA(NS2P1)=DSTA(NS2P1)+DSR
          DO 172 J=1,NS
          JNS=J+NS
          DO 172 I=1,NS
          INS=I+NS
          AROT(J,I)=AA(J,I)*COSAB(I)
         AROT(J,INS)=-AA(J,I)*SINAB(I)*Y(I)
         AROT(JNS,I)= BB(JNS,I)*SINAB(I)
         AROT(JNS,INS)=BB(JNS,I)*COSAB(I)*Y(I)
         AROT(NS2P1,I)=AA(NS2P1,I)*COSAB(I)+BB(NS2P1,I)*SINAB(I)
  172 AROT(NS2P1,INS)=-AA(NS2P1,I)*SINAB(I)*Y(I)+BB(NS2P1,I)*COSAB(I)*Y(
```

```
     1I)
          DO 173 J=1,NS
      TRIG=0.
          DO 174 I=1,NS
          INS =I+NS
      I2NS=I+NS2
      I3NS=I+NS3
          DROT=AA(J,I)*(YDOT(I)*Y(I3NS)+Y(I2NS)*Y(I3NS)*SINAB(I))
  174 TRIG=TRIG+DROT
  173 DRO(J)=TRIG+DSTA(J)
          DO 175 J=NSP1,NS2
      TRIG=0.
          DO 176 I=1,NS
          INS=I+NS
      I2NS=I+NS2
      I3NS=I+NS3
          DROT=-BB(J,I)*(XDOT(I)*Y(I3NS)+Y(I2NS)*Y(I3NS)*COSAB(I))
  176 TRIG=TRIG+DROT
  175 DRO(J)=TRIG+DSTA(J)
          DRO(NS2P1)=DSTA(NS2P1)
          DO 177 I=1,NS
          INS=I+NS
      I2NS=I+NS2
      I3NS=I+NS3
          DROT=AA(NS2P1,I)*(YDOT(I)*Y(I3NS)+Y(I2NS)*Y(I3NS)*SINAB(I))-BB
     1(NS2P1,I)*(XDOT(I)*Y(I3NS)+Y(I2NS)*Y(I3NS)*COSAB(I))
  177     DRO(NS2P1)=DRO(NS2P1)+DROT
          DO 39 J=1,NS2P1
   39     AROT(J,NS2P1)=CC(J)
      FF=0.0
      KL=ISIMDD(51,NS2P1,1,AROT,DRO,FF,IA)
          DO 600 I=1,NS
          INS=I+NS
      I2NS=I+NS2
      I3NS=I+NS3
          BD(I)=Y(I2NS)
          BD(INS)=Y(I3NS)
          BD(I2NS)=AROT(I,1)
  600     BD(I3NS)=AROT(INS,1)
      BD(NS4P1)=Y(NS4P2)
      BD(NS4P2)=AROT(NS2P1,1)
      WRITE (6,620) (BD(I),I=1,N)
```

```
620 FORMAT (1H1///14X,33HTHE APPROPRIATE DERIVATIVES  BD,S//(1P7D15.6)
   1)
    IF (KL.NE.3)  GO TO 560
    WRITE (6,540) KL
540 FORMAT(1H1/10X,4HKL =I1)
    CALL EXIT
560 RETURN
    END
```

Card Count 344

```
      SUBROUTINE BRGXY                                                  00223020
      IMPLICIT REAL*8 (A-H,O-Z)                                         00223030
       COMMON/MAFU14/ FDOFIX                                            00223035
      COMMON/MAFUF1/DUM1(75),NS,NB,IB(12)                               00223040
     X        /MAFU0/IDUM1(2),KK(25)                                    00223045
     1        /MAFU5/DUM2(300),BKMX(25),BKMY(25),BCMX(25),BCMY(25),BCB(25)00223050
     2        /MAFU6/BNB(25,6),BBB(25,6),BROB(25,7),BKB(25,6),BHB(25,6), 00223060
     3BDB(25,6),BEB(25,6)                                               00223070
     4        /MAFU8/TOLB                                               00223080
     5        /MAFU11/FD,FDOT                                           00223090
     6        /GARBG7/XX(25),YY(25),XB(25),YB(25),XDOT(25),YDOT(25),    00223100
     7XBDOT(25),YBDOT(25)                                               00223110
     8        /GARBG8/XB0,XB1,XB2,YB0,YB1,YB2,BROX0,BROX1,BROX2,BROY0,  00223120
     9BROY1,BROY2,FBRGX0,FBRGX1,FBRGX2,FBRGY0,FBRGY1,FBRGY2,I,IBI,KKB   00223130
        FUSAVE=FDOT                                                     00223200
        FDOT=FDOT-FDOFIX                                                00223300
   10 DO 162 I=1,NB                                                     00224000
         IBI=IB(I)                                                      00225000
        IF (BCMX(IBI).NE.0)  GO TO 91                                   00226000
        BCMX(IBI)=1.D-16                                                00227000
   91    XBDOT(IBI)=XDOT(IBI)/(1+BCB(IBI)/BCMX(IBI))                    00228000
        IF (BCMY(IBI).NE.0)  GO TO 101                                  00229000
        BCMY(IBI)=1.D-16                                                00230000
  101    YBDOT(IBI)=YDOT(IBI)/(1+BCB(IBI)/BCMY(IBI))                    00231000
      IF(BCB(IBI).EQ.0.)BCB(IBI)=1.D-16                                 00234000
         KKB=KK(IBI)                                                    00240000
      XB2=0.3D0*XX(IBI)                                                 00241000
      XB1=0.1D0*XX(IBI)                                                 00242000
      BROX2=DSQRT(XB2**2+(YY(IBI)/(BKMX(IBI)/BKMY(IBI)*(XX(IBI)/XB2-1.) 00243000
     1+1.))**2)                                                         00244000
      BROX1=DSQRT(XB1**2+(YY(IBI)/(BKMX(IBI)/BKMY(IBI)*(XX(IBI)/XB1-1.) 00245000
     1+1.))**2)                                                         00246000
      FBRGX2=(FDOT*(BNB(IBI,KKB)+BBB(IBI,KKB)*(BROX2-BROB(IBI,KKB))) +  00247000
     1BKB(IBI,KKB))*(BROX2**(BHB(IBI,KKB)-1.)+BDB(IBI,KKB)+BEB(IBI,KKB)/00248000
     2BROX2)*XB2-BKMX(IBI)*(XX(IBI)-XB2)                                00249000
      FBRGX1=(FDOT*(BNB(IBI,KKB)+BBB(IBI,KKB)*(BROX1-BROB(IBI,KKB))) +  00250000
     1BKB(IBI,KKB))*(BROX1**(BHB(IBI,KKB)-1.)+BDB(IBI,KKB)+BEB(IBI,KKB)/00251000
     2BROX1)*XB1-BKMX(IBI)*(XX(IBI)-XB1)                                00252000
   52 XB0=XB1-FBRGX1/(FBRGX2-FBRGX1)*(XB2-XB1)                          00253000
      BROX0=DSQRT(XB0**2+(YY(IBI)/(BKMX(IBI)/BKMY(IBI)*(XX(IBI)/XB0-1.) 00254000
     1+1.))**2)                                                         00255000
      FBRGX0=(FDOT*(BNB(IBI,KKB)+BBB(IBI,KKB)*(BROX0-BROB(IBI,KKB))) +  00256000
```

```
     1BKB(IBI,KKB))*(BROX0**(BHB(IBI,KKB)-1.)+BDB(IBI,KKB)+BEB(IBI,KKB)/00257000
     2BROX0)*XB0-BKMX(IBI)*(XX(IBI)-XB0)                                00258000
      PERCEX=XB0/XX(IBI)                                                00258500
      IF(DABS(1.-PERCEX).LE.1.D-8)GOTO152                               00259000
      IF(DABS(PERCEX).LT.1.D-8)GOTO152                                  00259500
      PERCEX=DABS(FBRGX0/(BKMX(IBI)*(XX(IBI)-XB0)))                     00260000
        IF(PERCEX.LT.TOLB) GO TO 152                                    00265000
      IF(DABS(XB2).GE.DABS(XB1))XB2=XB1                                 00265500
      XB1=XB0                                                           00267000
      FBRGX2=FBRGX1                                                     00268000
      FBRGX1=FBRGX0                                                     00269000
        GO TO 52                                                        00270000
  152 XB(IBI)=XB0                                                       00271000
      YB2=0.3D0*YY(IBI)                                                 00275000
      YB1=0.1D0*YY(IBI)                                                 00276000
      BROY2=DSQRT(YB2**2+(XX(IBI)/(BKMY(IBI)/BKMX(IBI)*(YY(IBI)/YB2-1.) 00277000
     1+1.))**2)                                                         00278000
      BROY1=DSQRT(YB1**2+(XX(IBI)/(BKMY(IBI)/BKMX(IBI)*(YY(IBI)/YB1-1.) 00279000
     1+1.))**2)                                                         00280000
      FBRGY2=(FDOT*(BNB(IBI,KKB)+BBB(IBI,KKB)*(BROY2-BROB(IBI,KKB)))+   00281000
     1BKB(IBI,KKB))*(BROY2**(BHB(IBI,KKB)-1.)+BDB(IBI,KKB)+BEB(IBI,KKB)/00282000
     2BROY2)*YB2-BKMY(IBI)*(YY(IBI)-YB2)                                00283000
      FBRGY1=(FDOT*(BNB(IBI,KKB)+BBB(IBI,KKB)*(BROY1-BROB(IBI,KKB))) +  00284000
     1BKB(IBI,KKB))*(BROY1**(BHB(IBI,KKB)-1.)+BDB(IBI,KKB)+BEB(IBI,KKB)/00285000
     2BROY1)*YB1-BKMY(IBI)*(YY(IBI)-YB1)                                00286000
   64 YB0=YB1-FBRGY1/(FBRGY2-FBRGY1)*(YB2-YB1)                          00287000
      BROY0=DSQRT(YB0**2+(XX(IBI)/(BKMY(IBI)/BKMX(IBI)*(YY(IBI)/YB0-1.) 00288000
     1+1.))**2)                                                         00289000
      FBRGY0=(FDOT*(BNB(IBI,KKB)+BBB(IBI,KKB)*(BROY0-BROB(IBI,KKB))) +  00290000
     1BKB(IBI,KKB))*(BROY0**(BHB(IBI,KKB)-1.)+BDB(IBI,KKB)+BEB(IBI,KKB)/00291000
     2BROY0)*YB0-BKMY(IBI)*(YY(IBI)-YB0)                                00292000
      PERCEY=YB0/YY(IBI)                                                00292500
      IF(DABS(1.-PERCEY).LE.1.D-8)GOTO162                               00293000
      IF(DABS(PERCEY).LT.1.D-8)GOTO162                                  00293500
      PERCEY=DABS(FBRGY0/(BKMY(IBI)*(YY(IBI)-YB0)))                     00294000
        IF(PERCEY.LT.TOLB) GO TO 162                                    00299000
      IF(DABS(YB2).GE.DABS(YB1))YB2=YB1                                 00300000
      YB1=YB0                                                           00301000
      FBRGY2=FBRGY1                                                     00302000
      FBRGY1=FBRGY0                                                     00303000
        GO TO 64                                                        00304000
  162 YB(IBI)=YB0                                                       00305000
         FDOT=FOSAVE                                                    00305500
      RETURN                                                            00306000
      END                                                               00307000
```

Card Count 87

```
      SUBROUTINE   INFLCO (C,B)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 MOF,MOM
      REAL Z
      DIMENSION  DD(25), D(25), QL(25),EE(25), GG(25), EIO(25), GAK(25),
     1SZ(25),ZQ(25),SZOL(25),ZQOL(25),QZ(25),ZSOL(25),QZOL(25),IB(12),
     2SHERGA(25),ROF(25,25),ROM(25,25),AFALEI(25),QLEI(25),SQL2EI(25)
     2,C(25,25),B(25,25), Z(25),       FOF(25,25), MOF(25,25), FOM(25,25
     3),MOM(25,25)
       COMMON/MAFUF1/DD,D,QL,  NS,NB,IB
      COMMON/MAFUF2/   SZ,QZ,ZQ,SZOL,ZSOL,QZOL,ZQOL,QMLOV,QLL
       COMMON/MAFUF3/  IB1,IB2
      COMMON/MAFUF4/ Z
       COMMON/MAFC/  EE,GG,EIO,GAK
     1      /SFP2/NSM1,IDUM1(9)
     2      /GARBG2/FOF,MOF,FOM,MOM,QLEI,SQL2EI,ROF,ROM,AFALEI,SHERGA,A,
     3E,F,G,GAQL,THETAM,THETAF,DSQ,DDSQ,D1,D2,I,J,K
      DATA PI/3.1415926535897932/
        A=PI/4.
      E=A/16.
      F=4./3.D0
      G=F/2.
      DO 200  J=1,NSM1
      DSQ=D(J)**2
      DDSQ=DD(J)**2
      D1=DDSQ-DSQ
      D2=DDSQ+DSQ
      QLEI(J)=QL(J)/(E*D1*D2*EE(J)+EIO(J))
       SQL2EI(J)=0.5* QL(J)* QLEI(J)
      GAQL=GAK(J)/QL(J)+A*GG(J)*D1/(QL(J)*(F*(DDSQ*DD(J)-DSQ*D(J))/( D2*
     1(DD(J)-D(J)))))
        SHERGA(J)=1./GAQL
  200 AFALEI(J)=SHERGA(J)+G*QL(J)*SQL2EI(J)
        DO 400 I=1,NS
        DO 400 J=1,NS
      FOF(I,J)=0.
      MOF(I,J)=0.
      FOM(I,J)=0.
      MOM(I,J)=0.
        C(I,J)=0
  400   B(I,J)=0
      DO 140      I=1,IB1
```

```
      IF(IB1.EQ.1) GO TO 122                                            00647000
  112 K=I+1                                                             00654000
       DO  120   J=K,IB1                                                00655000
        FOF(I,J) =-1.0                                                  00656000
        MOF(I,J) =Z(J)-Z(I)                                             00657000
  120   MOM(I,J) = -1.0                                                 00659000
  122 K=IB1+1                                                           00660000
       DO 130   J=K,IB2                                                 00661000
        FOF(I,J)= SZOL(I)                                               00662000
      MOF(I,J)=QZ(J)*SZOL(I)                                            00663000
        FOM(I,J)=QMLOV                                                  00664000
  130   MOM(I,J)=ZQOL(J)                                                00665000
  140 CONTINUE                                                          00673000
      KK=IB1+1                                                          00674000
      DO 440    I=KK,   IB2                                             00675000
  412 K=IB1+1                                                           00682000
      DO 420   J=K,I                                                    00683000
       FOF(I,J) = QZOL(I)                                               00684000
      MOF(I,J)=SZ(J)*QZOL(I)                                            00685000
       FOM(I,J) =QMLOV                                                  00686000
  420  MOM(I,J)  =ZSOL(J)                                               00687000
      K=I+1                                                             00688000
      IF(K.GT.IB2)GOTO440                                               00689000
       DO 430   J=K,   IB2                                              00690000
        FOF(I,J) =SZOL(I)                                               00691000
      MOF(I,J)=-ZQ(J)*SZOL(I)                                           00692000
        FOM(I,J)=QMLOV                                                  00693000
  430   MOM(I,J)=ZQOL(J)                                                00694000
  440 CONTINUE                                                          00702000
      IF (IB2.EQ.NS)GO TO 542                                           00703000
      KK=IB2+1                                                          00704000
      DO  540    I=KK,   NS                                             00705000
  512 K=IB1+1                                                           00712000
        DO 520   J=K,    IB2                                            00713000
         FOF(I,J) =QZOL(I)                                              00714000
      MOF(I,J)=SZ(J)*QZOL(I)                                            00715000
         FOM(I,J)=QMLOV                                                 00716000
  520    MOM(I,J) =ZSOL(J)                                              00717000
      K=IB2+1                                                           00718000
       DO 530    J=K,    I                                              00719000
        FOF(I,J) =1.                                                    00720000
        MOF(I,J) =Z(I)-Z(J)                                             00721000
```

```
530   MOM(I,J) =1.0                                                     00723000
540 CONTINUE                                                            00731000
542 DO 600  I=1,NS                                                      00732000
    ROF(I,1)=0.                                                         00732200
    ROM(I,1)=0.                                                         00732400
    THETAF=0.                                                           00732600
    THETAM=0.                                                           00732800
    DO 600    J=2,NS                                                    00733000
    ROF(I,J)=ROF(I,J-1)+QL(J-1)*THETAF+AFALEI(J-1)*FOF(I,J)+MOF(I,J)*   00734000
   1SQL2EI(J-1)                                                         00735000
    ROM(I,J)=ROM(I,J-1)+QL(J-1)*THETAM+AFALEI(J-1)*FOM(I,J)+MOM(I,J)*   00736000
   1SQL2EI(J-1)                                                         00737000
    THETAF=THETAF+SQL2EI(J-1)*FOF(I,J)+QLEI(J-1)*MOF(I,J)               00738000
600 THETAM=THETAM+SQL2EI(J-1)*FOM(I,J)+QLEI(J-1)*MOM(I,J)               00739000
    DO 700    I=1,NS                                                    00742000
    DO 700    J=1,NS                                                    00743000
    C(I,J)=ROF(I,J)-ROF(I,IB1)+SZOL(J)*(ROF(I,IB2)-ROF(I,IB1))          00748000
700 B(I,J)=ROM(I,J)-ROM(I,IB1)+SZOL(J)*(ROM(I,IB2)-ROM(I,IB1))          00749000
    RETURN                                                              00750000
    END                                                                 00751000
```

Card Count 104

```
      SUBROUTINE  SZMASS                                                01496000
        IMPLICIT REAL*8 (A-H,O-Z)                                       01497000
        REAL Z                                                          01498000
      DIMENSION  Z(25),QL(25),SZ(25),QZ(25),ZQ(25),SZOL(25),ZSOL(25),  Q01500000
     1ZOL(25),ZQOL(25), DD(25),D(25),DN(25),  Q6LDND(25),DDPLD(25),DDL(201501000
     25),Q1LDND(25), QM(25),AM(25),QID(25),AID(25),QIRO(25),AIRO(25)    01502000
      DIMENSION  ECC(25),ALFA(25),BETA(25),GAMMA(25)                    01503000
     1,IB(12)                                                           01503100
      COMMON/MAFU2/   DN,AM,AID,AIRO,ECC,ALFA,BETA,GAMMA,GX,GY          01504000
      COMMON/MAFU12/  QM,QID,QIRO                                       01505000
     1     ./SFP2/NSM1,IDUM1(9)                                         01505100
     2       /GARBG1/QMASS,QLDNDD,QL2,Q1LDND,DDL,DDPLD,Q6LDND,D2,       01505200
     3DD2,WEIT,POLARA,I                                                 01505300
        COMMON/MAFUF1/DD,D,QL,NS,NB,IB                                  01506000
        COMMON/MAFUF2/SZ,QZ,ZQ,SZOL,ZSOL,QZOL,ZQOL,QMLOV,QLL            01507000
        COMMON/MAFUF3/ IB1,IBNB                                         01508000
        COMMON/MAFUF4/Z/MAIFU/Q,S                                       01509000
      DATA PI/3.1415926535897932/                                       01510000
        U=4./3.D0                                                       01512000
      W=PI/(128.*386.088)                                               01514000
      E=16.*W                                                           01515000
      Z(1)=0                                                            01518000
      DO 105  I=2,NS                                                    01519000
  105 Z(I)=Z(I-1)+QL(I-1)                                               01520000
      Q=Z(IBNB)                                                         01522000
      S=Z(IB1)                                                          01521000
      DO 103 I=1,NS                                                     01523000
      SZ(I)=Z(IB1)-Z(I)                                                 01525000
      ZQ(I)=Z(I)-Z(IBNB)                                                01526000
  103 QZ(I)=-ZQ(I)                                                      01527000
      QLL=Z(IBNB)-Z(IB1)                                                01528000
      QMLOV=-1./QLL                                                     01530000
        DO 104  I=1,NS                                                  01531000
      SZOL(I)=SZ(I)/QLL                                                 01532000
        ZSOL(I)=-SZOL(I)                                                01533000
      QZOL(I)=QZ(I)/QLL                                                 01534000
  104   ZQOL(I)=-QZOL(I)                                                01535000
      DO 50       I=1,NS                                                01536000
      DD2    = DD(I)**2                                                 01537000
      D2     = D(I)**2                                                  01538000
      QL2      = QL(I)**2                                               01541000
      QLDNDD    =QL(I)* DN(I)*(DD2    -D2    )                          01542000
```

```
      Q6LDND(I) =            W*QLDNDD                                    01543000
      DDPLD(I) = DD2   +D2                                               01544000
      DDL(I) = DDPLD(I)+       U*QL2                                     01545000
   50 Q1LDND(I) =          E*QLDNDD                                      01546000
      QM(1)=Q1LDND(1) + AM(1)                                            01547000
      QM(NS)=Q1LDND(NS-1) +AM(NS)                                        01548000
      QID(1) = Q6LDND(1)*DDL(1) + AID(1)                                 01549000
      QID(NS)= Q6LDND(NS-1)*DDL(NS-1)   +AID(NS)                         01550000
      QIRO(1)=2.*Q6LDND(1)*DDPLD(1)+AIRO(1)                              01551000
      QIRO(NS)=2.*Q6LDND(NS-1)*DDPLD(NS-1) +AIRO(NS)                     01552000
      DO 55 I=2,NSM1                                                     01554000
      QM(I) =Q1LDND(I-1)+Q1LDND(I) +    AM(I)                            01555000
      QID(I)=Q6LDND(I-1)*DDL(I-1)+Q6LDND(I)*DDL(I)+AID(I)                01556000
   55 QIRO(I) = 2.*(Q6LDND(I-1)*DDPLD(I-1)+Q6LDND(I)*DDPLD(I)) +AIRO(I)  01557000
   62 QMASS=0.0                                                          01558000
      POLARA=0.0                                                         01559000
      DO 40 I=1,NS                                                       01560000
      QMASS=QMASS+QM(I)                                                  01561000
   40 POLARA=POLARA+QIRO(I)                                              01562000
      WEIT =386.088*QMASS                                                01563000
      WRITE (6,77) WEIT,QMASS,POLARA                                     01564000
   77 FORMAT(1H0///,9X'TOTAL ROTOR WEIGHT,LB ='1PD13.4/ 9X,33HTOTAL ROTO01565000
     1R MASS,(LB*SEC**2)/IN =1PD13.4/ 9X,57HTOTAL ROTOR POLAR MASS MOMEN01566000
     2T OF INERTIA, LB*IN*SEC**2  =1PD13.4/)                             01567000
        RETURN                                                           01568000
        END                                                              01569000
```

Card Count 68

```
      SUBROUTINE PLOT1(F)                                               00752000
      REAL F(50,12), Y(2)                                               00753000
      COMMON/SFP1/RPM(50),NPOINT                                        00754000
     1       /GARBG4/Y,I,J                                              00755100
     2       /MAFUF1/DUM1(151),NB,IB(12)                                00755200
C     GET LIMITS OF F & RPM                                             00757000
      Y(1)=1.E70                                                        00760000
      Y(2)=-1.E70                                                       00761000
      DO 10 I=1,NB                                                      00762000
      DO 10 J=1,NPOINT                                                  00763000
      Y(1)=AMIN1(Y(1),F(J,I))                                           00764000
   10 Y(2)=AMAX1(Y(2),F(J,I))                                           00765000
      CALL LRANGE(RPM(1),RPM(NPOINT),Y(1),Y(2))                         00767000
C     PLOT EACH OF THE NB FUNCTIONS                                     00768000
      DO 20 I=1,NB                                                      00769000
      CALL LRCURV(RPM,F(1,I),NPOINT,2,Y,0.)                             00769100
      CALL LRCNVT(IB(I),1,Y,1,3,0)                                      00769200
      DO 20 J=1,NPOINT                                                  00769300
   20 CALL LRLABL(Y,3,0,RPM(J),F(J,I),0.)                               00769400
      RETURN                                                            00770100
      END                                                               00771000
```

Card Count 21

```
C     SOLUTION TO A SYSTEM OF 1ST ORDER ORDINARY DIFFERENTIAL EQUATIONS RKAD0010
C     OF THE INITIAL VALUE TYPE.  THE FOLLOWING METHODS ARE AVAILABLE---RKAD0020
C  1. ADAMS-MOULTON PREDICTOR-CORRECTOR FIXED INCREMENT                 RKAD0030
C  2. ADAMS-MOULTON PREDICTOR-CORRECTOR VARIABLE INCREMENT              RKAD0040
C  3. RUNGE-KUTTA (ALSO USED TO GENERATE STARTING VALUES FOR A-M METHDS)RKAD0050
      SUBROUTINE RKADAM(N,T,Y,H,IND,ITIM,TOL,NERR)                      RKAD0060
      IMPLICIT REAL*8 (A-H,O-Z)                                         RKAD0070
      REAL*8 T,Y,H,TOL                                                  RKAD0080
      DIMENSION  Y(1),F(102,7),YB(102,5),A(102,4),YSAVE(102)            RKAD0090
C     N  ORDER OF SYSTEM (IF REDIMENSIONING REQUIRED, CHANGE NN IN DATA RKAD0100
C        STATEMENT AND ALSO THE 1ST SUBSCRIPT OF F AND YB IN DIMENSION  RKAD0110
C        STATEMENT).                                FROM CALLING PROGRAMRKAD0115
C     T  INDEPENDENT VARIABLE -- UPON ENTRY TO RKADAM FROM CALLING      RKAD0120
C        PROGRAM X IS AT BEGINNING OF STEP.  UPON RETURN TO CALLING     RKAD0130
C        PROGRAM X IS AT END OF STEP.                                   RKAD0140
C     Y  SOLUTION VECTOR OF DEPENDENT VARIABLES AS A FUNCTION OF X      RKAD0150
C     H  INCREMENT(ALGEBRAIC) -- UPON ENTRY TO RKADAM FROM CALLING      RKAD0160
C        PROGRAM H IS THE TRIAL INCREMENT FOR THIS STEP.  UPON RETURN TORKAD0170
C        CALLING PROGRAM H IS THE TRIAL INCREMENT FOR THE NEXT STEP.    RKAD0180
C     IND  FLAG TO SELECT METHOD                    FROM CALLING PROGRAMRKAD0190
C        =0 ADAMS-MOULTON PREDICTOR-CORRECTOR VARIABLE INCREMENT H      RKAD0200
C        =1 RUNGE-KUTTA FIXED INCREMENT H                               RKAD0210
C        =2 ADAMS-MOULTON FIXED INCREMENT H                             RKAD0220
C     ITIM  RESTART FLAG (APPLIES ONLY FOR IND=0,2) FROM CALLING PROGRAMRKAD0230
C        =+1 RESTART WITH FORWARDS(IN THE DIRECTION OF SIGN(H))         RKAD0240
C        INTEGRATION BY RUNGE-KUTTA TO GET STARTING VALUES              RKAD0250
C        =-1 SAME AS +1 EXCEPT USES BACKWARDS (-SIGN(H)) INTEGRATION    RKAD0260
C        =0  CONTINUE INTEGRATING                                       RKAD0270
C     TOL  APPLIES ONLY TO IND=0.  ALLOWABLE RELATIVE ERROR BETWEEN THE RKAD0280
C          PREDICTED AND CORRECTED SOLUTIONS.       FROM CALLING PROGRAMRKAD0290
C     NERR  ERROR FLAG RETURNED TO CALLING PROGRAM                      RKAD0300
C        =0 SOLUTION IS VALID                                           RKAD0310
C        =1 SOLUTION INVALID -- N IS INVALID OR ELSE H HAS GONE TO 0    RKAD0320
      DATA NN/102/                                                      RKAD0330
      NERR=0                                                            RKAD0340
        DO 200 I=1,N                                                    RKAD0341
  200   YSAVE(I)=Y(I)                                                   RKAD0342
      IF(IND.EQ.1)GOTO10                                                RKAD0350
      IF(ITIM.EQ.0)GOTO170                                              RKAD0360
   10 NS=0                                                              RKAD0370
      CALL OVERFL(K)                                                    RKAD0375
C     TEST VALIDITY OF N                                                RKAD0380
```

```
      IF(N.GE.1.AND.N.LE.NN)GOTO40                                      RKAD0390
   20 NERR=1                                                            RKAD0400
   30   DO 220 I=1,N                                                    RKAD0401
  220   Y(I)=0.5*(Y(I)+YSAVE(I))                                        RKAD0402
        RETURN                                                          RKAD0411
   40 IF(IND.NE.1)GOTO50                                                RKAD0420
C     RUNGE-KUTTA                                                       RKAD0430
      CALL RUNKUT(N,H,T,Y,Y,A)                                          RKAD0440
      T=T+H                                                             RKAD0450
      GOTO30                                                            RKAD0460
C     ADAMS-MOULTON                                                     RKAD0470
   50 IF(ITIM.LT.0)GOTO180                                              RKAD0480
C     RESTART FORWARDS INTEGRATION (IN DIRECTION OF X+H                 RKAD0490
      ISTFLG=1                                                          RKAD0500
      IF(IND.EQ.2)ISTFLG=0                                              RKAD0505
      XB=T                                                              RKAD0510
      DO60I=1,N                                                         RKAD0520
   60 YB(I,1)=Y(I)                                                      RKAD0530
      L=0                                                               RKAD0535
   70 DO80I=1,3                                                         RKAD0540
      T=XB+H*(I-1)                                                      RKAD0560
      J=5-I                                                             RKAD0567
      CALL RUNKUT(N,H,T,Y,YB(1,I+1),A)                                  RKAD0570
      DO 75 K=1,N                                                       RKAD0575
      IF(DABS(YB(K,I+1)).LE.1.D12)GOTO73                                RKAD0577
      H=H/10.                                                           RKAD0579
      GOTO 83                                                           RKAD0581
   73 IF(L.NE.0.AND.I.EQ.1)GOTO75                                       RKAD0583
      F(K,J)=A(K,1)                                                     RKAD0585
   75 Y(K)=YB(K,I+1)                                                    RKAD0590
   80 CONTINUE                                                          RKAD0595
      T=T+H                                                             RKAD0605
      CALL FUND(N,T,Y,F)                                                RKAD0607
      DO 2 I=1,N                                                        RKAD0608
      IF(Y(I).GE.1.D14) GO TO 3                                         RKAD0609
    2 CONTINUE                                                          RKAD0610
      GO TO 4                                                           RKAD0611
    3 RETURN                                                            RKAD0612
    4 CALL ADAMLT(N,T,Y,H,IND,ITIM,TOL,NERR,ISTFLG,F,A)                 RKAD0615
      IF(ISTFLG.EQ.0)GOTO110                                            RKAD0620
      H=H/2                                                             RKAD0640
   83 T=XB                                                              RKAD0645
```

```
      CALL OVERFL(K)                                                    RKAD0650
      IF(K.EQ.3)GOTO100                                                 RKAD0660
      DO90I=1,N                                                         RKAD0670
   90 Y(I)=YB(I,1)                                                      RKAD0680
      L=1                                                               RKAD0685
      GOTO70                                                            RKAD0690
  100 H=0.                                                              RKAD0700
      GOTO20                                                            RKAD0710
C     FOR ITIM = +1, FEED STARTING VALUES BACK TO CALLING PROGRAM ONE ATRKAD0720
C     A TIME (RUNGE-KUTTA SOLNS 1ST 4 PTS AND THEN A-M SOLN FOR 5TH PT) RKAD0730
  110 DO120I=1,N                                                        RKAD0740
  120 YB(I,5)=Y(I)                                                      RKAD0750
      NS=2                                                              RKAD0760
  130 T=XB+H*(NS-1)                                                     RKAD0770
      DO140I=1,N                                                        RKAD0780
  140 Y(I)=YB(I,NS)                                                     RKAD0790
      NS=NS+1                                                           RKAD0800
      IF(NS.GT.5)NS=0                                                   RKAD0810
      GOTO30                                                            RKAD0820
C     CONTINUE INTEGRATION PROCEDURE (A-M)                              RKAD0830
  170 IF(NS)190,190,130                                                 RKAD0840
  180 ISTFLG=0                                                          RKAD0850
  190 CALL ADAMLT(N,T,Y,H,IND,ITIM,TOL,NERR,ISTFLG,F,A)                 RKAD0860
      GOTO30                                                            RKAD0870
      END                                                               RKAD0880
```

Card Count 109

```
C     4TH ORDER RUNGE-KUTTA INTEGRATION FOR A SYSTEM OF 1ST ORDER,      RUNK0010
C     ORDINARY DIFFERENTIAL EQNS.                                       RUNK0015
      SUBROUTINE RUNKUT(N,H,X,Y,YY,A)                                   RUNK0020
C     SEE DIMENSION STATEMENT FOR LIMITATION ON ORDER OF SYSTEM         RUNK0022
C     CHANGE DIMENSION AS IS REQUIRED, I.E.,A(MAXORDER,4), V(MAXORDER)  RUNK0024
      IMPLICIT REAL*8 (A-H,O-Z)                                         RUNK0025
      REAL*8 X,H,Y(1),YY(1)                                             RUNK0030
      DIMENSION  A(102,4),V(102)                                        RUNK0040
C     N = ORDER OF SYSTEM                                               RUNK0050
C     H = INTEGRATION STEP                                              RUNK0060
C     X = INDEPENDENT VARIABLE AT BEGINNING OF STEP                     RUNK0070
C     Y = VECTOR OF DEPENDENT VARIABLES AT BEGINNING OF STEP            RUNK0080
C     YY = SOLUTION VECTOR OF DEPENDENT VARIABLES AT END OF STEP        RUNK0090
      X1=H/2.                                                           RUNK0100
      X2=X+X1                                                           RUNK0110
      X3=X+H                                                            RUNK0120
      CALL FUND(N,X,Y,A(1,1))                                           RUNK0130
      DO 2 I=1,N                                                        RUNK0131
      IF(Y(I).GE.1.D14) GO TO 3                                         RUNK0132
    2 CONTINUE                                                          RUNK0133
      GO TO 4                                                           RUNK0134
    3 RETURN                                                            RUNK0135
    4 DO 10 I=1,N                                                       RUNK0140
   10 V(I)=Y(I)+X1*A(I,1)                                               RUNK0150
      CALL FUND(N,X2,V,A(1,2))                                          RUNK0160
      DO 12 I=1,N                                                       RUNK0161
      IF(Y(I).GE.1.D14) GO TO 13                                        RUNK0162
   12 CONTINUE                                                          RUNK0163
      GO TO 14                                                          RUNK0164
   13 RETURN                                                            RUNK0165
   14 DO 20 I=1,N                                                       RUNK0170
   20 V(I)=Y(I)+X1*A(I,2)                                               RUNK0180
      CALL FUND(N,X2,V,A(1,3))                                          RUNK0190
      DO 22 I=1,N                                                       RUNK0191
      IF(Y(I).GE.1.D14) GO TO 23                                        RUNK0192
   22 CONTINUE                                                          RUNK0193
      GO TO 24                                                          RUNK0194
   23 RETURN                                                            RUNK0195
   24 DO 30 I=1,N                                                       RUNK0200
   30 V(I)=Y(I)+H*A(I,3)                                                RUNK0300
      CALL FUND(N,X3,V,A(1,4))                                          RUNK0310
      DO 32 I=1,N                                                       RUNK0311
```

```
      IF(Y(I).GE.1.D14) GO TO 33                                        RUNK0312
   32 CONTINUE                                                          RUNK0313
      GO TO 34                                                          RUNK0314
   33 RETURN                                                            RUNK0315
   34 DO 40 I=1,N                                                       RUNK0320
   40 YY(I)=Y(I)+H*(A(I,1)+2.*(A(I,2)+A(I,3))+A(I,4))/6.0               RUNK0330
      RETURN                                                            RUNK0340
      END                                                               RUNK0350
```

Card Count 50

```
C     ADAMS-MOULTON INTEGRATER FOR A SYSTEM OF 1ST ORDER ORDINARY       ADAM0010
C     DIFFERENTIAL EQUATIONS OF THE INITIAL VALUE TYPE.                 ADAM0020
C     CHOICE OF FIXED OR VARIABLE INCREMENTING.                         ADAM0030
C     IF VARIABLE INCREMENTING THE FOLLOWING RULES WILL APPLY ---       ADAM0040
C        DEFINE YP=PREDICTOR SOLUTION VECTOR                            ADAM0050
C        DEFINE YC=CORRECTOR SOLUTION VECTOR                            ADAM0060
C        DEFINE RE=DIFF.RATIO OF YP & YC IF ABS(YC).GT.1 ELSE JUST DIFF.ADAM0070
C  1. IF TOL*0.02.LE.RE.LE.TOL IS TRUE THEN H IS UNCHANGED AND YC IS SOLADAM0080
C  2. IF RE(I).GT.TOL FOR ANY I OF RE THEN H IS HALVED AND YP AND YC AREADAM0090
C     RECOMPUTED.  HALVING THE INCREMENT IS NOT RESTRICTED.             ADAM0100
C     THIS STEP.  H IS REPLACED BY 2*H AND RETURNED (WITH YC) AS NEW    ADAM0120
C     INCREMENT FOR THE NEXT STEP.  H MAY BE DOUBLED ONLY AFTER 4       ADAM0130
C     SUCCESSIVE STEPS USING THE SAME INCREMENT (AN ATTEMPT TO MAINTAIN ADAM0140
C     STABILITY IN THE SOLUTION).                                       ADAM0150
C  4. IF H IS HALVED AN ENTIRE SET OF DERIVATIVES ARE COMPUTED USING    ADAM0160
C     BACKWARDS (IN THE DIRECTION OF -SIGN(H)) INTEGRATION BY RUNGE-KUTAADAM0170
C  5. IF H IS DOUBLED PREVIOUS SAVED DERIVATIVES ARE USED.              ADAM0180
C                                                                       ADAM0190
      SUBROUTINE ADAMLT(N,X,Y,H,IND,ITIM,TOL,NERR,ISTFLG,F,A)           ADAM0200
      IMPLICIT REAL*8 (A-H,O-Z)                                         ADAM0210
      REAL*8 X,Y,H,TOL,F                                                ADAM0220
      DIMENSION  Y(1),F(102,7),YP(102),YC(102),A(102,4)                 ADAM0230
C     SEE COMMENTS IN SUBROUTINE RKADAM FOR DESCRIPTION OF ARGUMENTS.   ADAM0240
C     IF REDIMENSIONING IS DONE IN RKADAM THEN NN MUST BE CHANGED       ADAM0250
C     ACCORDINGLY IN F(NN,7), YP(NN), AND YC(NN) IN ADAMLT DIMENSION STMADAM0260
      IF(ITIM.NE.0)GOTO60                                               ADAM0270
      IF(IND.EQ.2)GOTO10                                                ADAM0280
      IF(IC.EQ.-1)GOTO30                                                ADAM0290
   10 DO20I=1,N                                                         ADAM0300
      F(I,6)=F(I,5)                                                     ADAM0310
      F(I,5)=F(I,4)                                                     ADAM0320
      F(I,4)=F(I,3)                                                     ADAM0330
      F(I,3)=F(I,2)                                                     ADAM0340
   20 F(I,2)=F(I,1)                                                     ADAM0350
      GOTO50                                                            ADAM0360
C     DOUBLE INCREMENT BEING ATTEMPTED THIS STEP (THIS ENTRY)           ADAM0370
   30 IC=0                                                              ADAM0380
      DO40I=1,N                                                         ADAM0390
      F(I,3)=F(I,4)                                                     ADAM0400
   40 F(I,4)=F(I,6)                                                     ADAM0410
   50 CALL FUND(N,X,Y,F(1,1))                                           ADAM0420
      GOTO90                                                            ADAM0430
```

```
C     RESTART                                                           ADAM0440
   60 IC=0                                                              ADAM0460
      IF(ITIM.GT.0)GOTO90                                               ADAM0470
      DO 70 I=1,N                                                       ADAM0475
   70 YP(I)=Y(I)                                                        ADAM0477
C     GET NEW SET OF DERIVATIVES BY BACKWARDS INTEGRATIONS (RUNGE-KUTTA)ADAM0480
      XB=X                                                              ADAM0490
      DO80I=1,3                                                         ADAM0500
      CALL RUNKUT(N,-H,XB,YP,YP,A)                                      ADAM0510
      XB=X-H*I                                                          ADAM0520
      DO 78 K=1,N                                                       ADAM0530
   78 F(K,I)=A(K,1)                                                     ADAM0532
   80 CONTINUE                                                          ADAM0535
      CALL FUND(N,XB,YP,F(1,4))                                         ADAM0537
   90 HH=H/24.D0                                                        ADAM0540
C     PREDICTOR SOLUTION                                                ADAM0550
      DO100I=1,N                                                        ADAM0560
  100 YP(I)=Y(I)+HH*(55.D0*F(I,1)-59.D0*F(I,2)+37.D0*F(I,3)-9.D0*F(I,4))ADAM0570
  130 CALL FUND(N,X+H,YP,F(1,7))                                        ADAM0650
C     CORRECTOR SOLUTION                                                ADAM0660
      DO140I=1,N                                                        ADAM0670
  140 YC(I)=Y(I)+HH*(19.D0*F(I,1)-5.D0*F(I,2)+F(I,3)+9.D0*F(I,7))       ADAM0680
C     TEST FOR FIXED INCREMENT OPTION                                   ADAM0682
      IF(IND.EQ.0)GOTO145                                               ADAM0684
      X=X+H                                                             ADAM0686
      GOTO170                                                           ADAM0688
C     TEST RELATIVE ERROR                                               ADAM0690
  145 S=1                                                               ADAM0700
      DO150I=1,N                                                        ADAM0710
      T=DABS(YC(I)-YP(I))                                               ADAM0720
      U=DMAX1(DABS(YC(I)),1.D0)                                         ADAM0730
      V=U*TOL                                                           ADAM0740
      W=V*0.02D0                                                        ADAM0750
      IF(T.GT.V)GOTO200                                                 ADAM0760
      IF(W.LE.T)S=0.D0                                                  ADAM0770
  150 CONTINUE                                                          ADAM0780
      ISTFLG=0                                                          ADAM0790
      X=X+H                                                             ADAM0800
C     TEST IF 4 STEPS HAVE ELAPSED USING SAME INCREMENT FOR DOUBLNG TESTADAM0810
      IF(IC.LT.3)GOTO160                                                ADAM0820
      IF(S.EQ.0)GOTO170                                                 ADAM0830
C     SET H TO 2*H FOR NEXT STEP                                        ADAM0840
```

```
      IC=-1                                                             ADAM0850
      H=2.D0*H                                                          ADAM0860
      GOTO170                                                           ADAM0870
  160 IC=1+IC                                                           ADAM0880
  170 DO180I=1,N                                                        ADAM0890
  180 Y(I)=YC(I)                                                        ADAM0900
      GOTO120                                                           ADAM0910
  200 IF(ISTFLG.NE.0)GOTO120                                            ADAM0920
C     HALVE THE INCREMENT AND RECOMPUTE                                 ADAM0930
      H=H/2.D0                                                          ADAM0940
      CALL OVERFL(K)                                                    ADAM0950
      IF(K.NE.3)GOTO60                                                  ADAM0960
      H=0                                                               ADAM0970
      NERR=1                                                            ADAM0980
  120 RETURN                                                            ADAM0990
      END                                                               ADAM1000
```

Card Count 100

```
       SUBROUTINE TIME(T,H,TA,ITIM)                                      00000100
      IMPLICIT REAL*8 (A-H,O-Z)                                          00000200
      IF(ITIM.EQ.0) GO TO 10                                             00000300
      TA=0.5*H                                                           00000400
      ICOUNT=1                                                           00000500
      RETURN                                                             00000600
   10 ICOUNT=ICOUNT+1                                                    00000700
      IF(ICOUNT.EQ.2) GO TO 2                                            00000800
      IF(ICOUNT.EQ.3) GO TO 3                                            00000900
      IF(ICOUNT.EQ.4) GO TO 4                                            00001000
      IF(ICOUNT.GE.5) GO TO 5                                            00001100
    2 TA=1.25*H                                                          00001200
      RETURN                                                             00001300
    3 TA=2.125*H                                                         00001400
      RETURN                                                             00001500
    4 TA=3.0625*H                                                        00001600
      TSAVE=T                                                            00001700
      RETURN                                                             00001800
    5 TA=TA+0.5*(T-TSAVE)                                                00001900
      TSAVE=T                                                            00002000
      ICOUNT=ICOUNT-1                                                    00002100
      RETURN                                                             00002200
      END                                                                00002300
```

Card Count 23

```
C     SUBPROGRAM TO SOLVE SIMULTANEOUS LINEAR EQUATIONS
C       ARGUMENTS-
C
C         DATE- 1/13/67   MODIFIED  FOR COMPILATION IN RELEASE 14
C
C                   DSM       DIMENSIONED SIZE OF COEFFICIENT MATRIX
C                   NE        ACTUAL NUMBER OF EQUATIONS FOR THIS CALL
C                   NC        NUMBER OF COLUMNS IN CONSTANT MATRIX
C                   A         COEFFICIENT MATRIX
C                   B         CONSTANT MATRIX
C                   DET       INPUT - SCALE FACTOR, OUTPUT - FACTOR TIMES
C                             DETERMINANT VALUE OF COEFFICIENT MATRIX
C                   C         TEMPORARY STORAGE FOR SUBROUTINE
C                   ISIMEQ    RETURNS 1 IF OK, 2 IF OVFLO, 3 IF SINGULAR
C                   IF NC IS NEGATIVE, THE INVERSE OF THE COEFFICIENT
C                   MATRIX IS REQUIRED, MATRIX B IS SET UP AS IDENTITY.
      FUNCTION   ISIMDD(DSM, NE, NC, A, B, DET, C )
      LOGICAL DVO
      INTEGER DSM, C, T, SUB1, SUB2,    R, D
      DOUBLE PRECISION   B, PIVOT, DET, S
      DIMENSION   B(1),C(1)
C     INITIALIZE
      N           =  NE
      D = DSM
      M = IABS(NC)
      ISIMDD      =  1
      DVO         =  .FALSE.
      DO 1 I = 1,N
    1 C(I) = I
      IF(NC) 5, 15, 15
C     INVERSE REQUIRED
    5 SUB2 = 0
      DO 10 J = 1,N
        SUB1 = SUB2
        DO 6 I = 1,N
        SUB1 = SUB1 + 1
    6   B(SUB1) = 0.0
       SUB1  = SUB2 + J
      B(SUB1)     =  1.0D0
   10 SUB2 = SUB2 + D
      GO TO 15
      ENTRY IDETDD(DSM, NE, A, DET)
```

```
      DOUBLE PRECISION   A                                              AWCU0450
      DIMENSION  A(1)                                                   AWCU0460
      N          =  NE                                                  AWCU0470
      D          =  DSM                                                 AWCU0480
      IDETDD     =  1                                                   AWCU0490
      DVO        =  .TRUE.                                              AWCU0500
C     START MAIN LOOP                                                   AWCU0510
   15 DO 1000 L = 1,N                                                   AWCU0520
         LP1 = L+ 1                                                     AWCU0530
        DO 40 I = L,N                                                   AWCU0540
      PIVOT      =  0.0D0                                               AWCU0550
          SUB1 = (L-1) * D + I                                          AWCU0560
          SUB2 = SUB1                                                   AWCU0570
            DO 20 J = L,N                                               AWCU0580
      IF(DABS(PIVOT) .GE. DABS(A(SUB1)))  GO TO 20                      AWCU0590
             PIVOT = A(SUB1)                                            AWCU0600
             JB = J                                                     AWCU0610
   20       SUB1 = SUB1 + D                                             AWCU0620
C     COMPUTE DETERMINANT                                               AWCU0630
      CALL OVERFL(T)                                                    AWCU0640
        DET = DET * PIVOT                                               AWCU0650
      IF(.NOT. DVO)  GO TO 24                                           AWCU0660
        CALL OVERFL(T)                                                  AWCU0670
      IF(T .EQ. 1)  IDETDD = 2                                          AWCU0680
C     TEST FOR SINGULAR MATRIX                                          AWCU0690
   24 IF(PIVOT .EQ. 0.0D0)GO TO 2000                                    AWCU0700
            DO 25 J = L,N                                               AWCU0710
            A(SUB2) = A(SUB2) / PIVOT                                   AWCU0720
   25       SUB2 = SUB2 + D                                             AWCU0730
         IF (DVO) GO TO 35                                              AWCU0740
           SUB1 = I                                                     AWCU0750
            DO 30 J = 1,M                                               AWCU0760
            B(SUB1) = B(SUB1) / PIVOT                                   AWCU0770
   30       SUB1 = SUB1 + D                                             AWCU0780
   35    IF (I .EQ. L) JP = JB                                          AWCU0790
   40   CONTINUE                                                        AWCU0800
C     INTERCHANGE COLUMNS                                               AWCU0810
  100   IF (JP .EQ. L) GO TO 260                                        AWCU0820
      IF (DVO) GO TO 110                                                AWCU0830
        T = C(L)                                                        AWCU0840
        C(L) = C(JP)                                                    AWCU0850
        C(JP) = T                                                       AWCU0860
```

```
  110     R = D * L - D                                                 AWCU0870
          T = D * JP - D                                                AWCU0880
          DO 120 I = 1,N                                                AWCU0890
          SUB1 = R + I                                                  AWCU0900
          SUB2 =  T + I                                                 AWCU0910
          S = A(SUB1)                                                   AWCU0920
          A(SUB1) = A(SUB2)                                             AWCU0930
  120     A(SUB2) = S                                                   AWCU0940
        DET = -DET                                                      AWCU0950
C     REDUCE PIVOT COLUMN                                               AWCU0960
  260   R = D* L - D                                                    AWCU0970
        DO 400 I = 1,N                                                  AWCU0980
         IP = R + I                                                     AWCU0990
         PIVOT = A(IP)                                                  AWCU1000
        IF (I .EQ. L .OR. PIVOT .EQ. 0.0) GO TO 400                     AWCU1010
         SUB1 = L                                                       AWCU1020
         SUB2 = I                                                       AWCU1030
          DO 360 J = 1,N                                                AWCU1040
           IF (J .LT. LP1) GO TO 300                                    AWCU1050
            S = PIVOT * A(SUB1)                                         AWCU1060
            A(SUB2) = A(SUB2) - S                                       AWCU1070
      IF(DABS(A(SUB2)) .LT. DABS(3.0E-8*S))  A(SUB2) =  0.0             AWCU1080
  300     IF (DVO .OR. J .GT. M) GO TO 350                              AWCU1090
            B(SUB2) = B(SUB2) - PIVOT * B(SUB1)                         AWCU1100
  350      SUB1 = SUB1 + D                                              AWCU1110
  360     SUB2 = SUB2 + D                                               AWCU1120
  400   CONTINUE                                                        AWCU1130
 1000 CONTINUE                                                          AWCU1140
      IF (DVO) GO TO 1500                                               AWCU1150
C     REARRANGE VARIABLES                                               AWCU1160
 1100 DO  1201  L=1,N                                                   AWCU1170
       SUB1 = C(L)                                                      AWCU1180
       SUB2 = L                                                         AWCU1190
        DO 1200 J = 1,M                                                 AWCU1200
         A(SUB1) = B(SUB2)                                              AWCU1210
         SUB1 = SUB1 + D                                                AWCU1220
 1200    SUB2 = SUB2 + D                                                AWCU1230
 1201 CONTINUE                                                          AWCU1240
 1500 RETURN                                                            AWCU1250
C     SINGULAR COEFFICIENT MATRIX                                       AWCU1260
 2000 IF(DVO)    GO TO 3000                                             AWCU1270
      ISIMDD      =  3                                                  AWCU1280
      GO TO 1500                                                        AWCU1290
 3000 IDETDD      =  3                                                  AWCU1300
      GO TO 1500                                                        AWCU1310
      END                                                               AWCU1320
```

```
AWCG     TITLE '       TIMEV -   READ ELAPSED TIME IN SECONDS           AWCG0030
COUNTV   START   0                                                      AWCG0040
*********************************************************************** AWCG0050
*                                                                       AWCG0060
*  ROUTINE NAME - TIMEV - READ ELAPSED TIME                             AWCG0070
*          DATE - 02/01/68   UPDATED 04/10/68  FOR RELEASE 14           AWCG0080
*         NORTH AMERICAN AVIATION,INC.--CORPORATE LIBRARY ROUTINE     * AWCG0090
*                                                                       AWCG0100
*********************************************************************** AWCG0110
         SPACE 4                                                        AWCG0120
*********************************************************************** AWCG0130
*                                                                       AWCG0140
*  FORTRAN LIBRARY ROUTINE                                              AWCG0150
*                                                                       AWCG0160
*  CALL COUNTV                                                          AWCG0170
*         THE TIME OF DAY IS SAVED WITHIN THE COUNTV ROUTINE            AWCG0180
*         WHICH IN EFFECT SETS A PSEUDO-TIMER TO ZERO.                  AWCG0190
*                                                                       AWCG0200
*  CALL TIMEV (T)                                                       AWCG0210
*         THE TIME INTERVAL IN SECONDS SINCE THE LAST CALL COUNTV       AWCG0220
*         IS STORED IN LOCATION T AS A SINGLE PRECISION FLOATING        AWCG0230
*         POINT NUMBER.                                                 AWCG0240
*                                                                       AWCG0250
*  RESTRICTIONS  -  NOT REUSABLE BY ANOTHER TASK SINCE COUNTV           AWCG0260
*         STORES TIME OF DAY IN A LOCATION WITHIN THE ROUTINE.          AWCG0270
*                                                                       AWCG0280
*         REQUIRES THE SUPERVISOR MACRO 'TIME'                          AWCG0290
*                                                                       AWCG0300
*********************************************************************** AWCG0310
         EJECT                                                          AWCG0320
*  MAKE IT LOOK LIKE THE TIMER IS SET TO ZERO                           AWCG0330
         ENTRY TIMEV                                                    AWCG0340
         ENTRY TIMERV                                                   AWCG0350
         ENTRY COUNTV                                                   AWCG0360
         SAVE   (14,12),,COUNTV                                         AWCG0370
         BALR  10,0                                                     AWCG0380
         USING *,10                                                     AWCG0390
         ST    13,SAVE+4                                                AWCG0400
         LA    2,SAVE                                                   AWCG0410
         ST    2,8(13)                                                  AWCG0420
         MVI   FSTSW,1             SET FLAG TO SAY COUNTV WAS CALLED    AWCG0430
* XOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOXOX AWCG0440
```

```
         TIME  TU                  TIME IN REG 0                        AWCG0450
         ST    0,TZERO                                                  AWCG0460
         L     13,SAVE+4                                                AWCG0470
         LM    14,12,12(13)                                             AWCG0480
         MVI   12(13),X'FF'                                             AWCG0490
         LA    15,0                                                     AWCG0500
         BR    14                  RETURN                               AWCG0510
         EJECT                                                          AWCG0520
*  READ ELAPSED TIME IN SECONDS                                         AWCG0530
TIMERV   EQU   *                                                        AWCG0540
TIMEV    SAVE  (14,12),,TIMEV                                           AWCG0550
         BALR  10,0                                                     AWCG0560
         USING *,10                                                     AWCG0570
         ST    13,SAVE+4                                                AWCG0580
         LA    2,SAVE                                                   AWCG0590
         ST    2,8(13)                                                  AWCG0600
         L     4,0(1)              LOC OF USERS TIME CELL               AWCG0610
         TIME  TU                  TIME IN REG 0                        AWCG0620
         TM    FSTSW,1             DID COUNTV GET CALLED                AWCG0630
         BNZ   T1                                                  (N1) AWCG0640
         ST    0,TZERO             COUNTV WASN'T CALLED            (N1) AWCG0650
         MVI   FSTSW,1 SET FLAG TO PRETEND COUNTV NOW BEING CALLED (N1) AWCG0660
         SER   2,2                 ZERO FL PT REG                  (N1) AWCG0670
         B     T3                                                  (N1) AWCG0680
*        *     *                                                   (N1) AWCG0690
T1       SL    0,TZERO             COUNTV WAS CALLED               (N1) AWCG0700
         BC    3,T2                IS DIFFERENCE NEGATIVE          (N1) AWCG0710
         AL    0,DAY               YES ADD 24 HRS                  (N1) AWCG0720
T2       ST    0,BITS2             ELAPSED TIME IN REG 0           (N1) AWCG0730
         LD    2,BITS1             ELAPSED TIME IN FREG 2          (N1) AWCG0740
         DD    2,FACTOR            CONVERT TIMER UNITS TO SECONDS. (N1) AWCG0750
T3       STE   2,0(4)              STORE IN USERS TIME CELL             AWCG0760
         L     13,SAVE+4                                                AWCG0770
         LM    14,12,12(13)                                             AWCG0780
         MVI   12(13),X'FF'                                             AWCG0790
         LA    15,0                                                     AWCG0800
         BR    14                  RETURN                               AWCG0810
         EJECT                                                          AWCG0820
FSTSW    DC    X'0'                SET FIRST TIME THRU                  AWCG0830
FACTOR   DC    D'38400'            300*2**7= 38400                 (N1) AWCG0840
BITS1    DC    X'4E000000'         1-ST HALF OF DOUBLE-WORD 'BITS1'(N1) AWCG0850
BITS2    DC    X'00000000'         2-ND HALF OF DOUBLE-WORD 'BITS1'(N1) AWCG0860
```

```
DAY      DC X'C5C10000' =3317760000= 38400*86400= 300*2**7*60*60*24(N1) AWCG0870
TZERO    DC    F'0'                FIRST TIME READING                   AWCG0880
SAVE     DC    18F'0'              SAVE AREA                            AWCG0890
*THE TIME TU MACRO RETURNS THE TIME OF DAY AS 32-BIT UNSIGNED INTEGER.  AWCG0900
*BIT 31= 1/(300*2**7) SEC.= 26 MICRO-SEC.                               AWCG0910
*BIT 24= 1/300 SEC.                                                     AWCG0920
*BIT 0= 2**24/300 SEC.= 15.5 HOURS                                      AWCG0930
*MAX. TIME = 2**25/300 SEC. = 31 HOURS WHICH IS ENOUGH TO CONTAIN THE   AWCG0940
*   TIME OF DAY BASED ON A 24-HOUR CLOCK.                               AWCG0950
*MAX. RESOLUTION = 1/60 SEC. STANDARD OR 26 MICRO-SEC. FOR AN OPTIONAL  AWCG0960
*   HIGH-RESOLUTION TIMER WHICH IS NOT STANDARD AT MOST INSTALLATIONS.  AWCG0970
         END                                                            AWCG0980
/*                                                                      AWCG0990
```

Card Count 97