

ANALYSIS OF FLUID SLOSHING IN ARBITRARY TANKS HAVING  
ROTATIONAL SYMMETRY - A CORRECTION FOR LOW GRAVITY  
CONDITIONS  
VOLUME II  
COMPUTATIONAL APPLICATION

**N70-37813**

|                   |                               |            |
|-------------------|-------------------------------|------------|
| FACILITY FORM 602 | (ACCESSION NUMBER)            | (THRU)     |
|                   | 114                           | 1          |
|                   | (PAGES)                       | (CODE)     |
|                   | CR-102732                     | 12         |
|                   | (NASA CR OR TMX OR AD NUMBER) | (CATEGORY) |

ANALYSIS OF FLUID SLOSHING IN ARBITRARY TANKS HAVING  
ROTATIONAL SYMMETRY - A CORRECTION FOR LOW GRAVITY  
CONDITIONS  
VOLUME II  
COMPUTATIONAL APPLICATION

by  
T. S. Chandler and D. O. Lomen  
UNIDEV Inc.  
Huntsville, Alabama

March 1970  
Contract No. NAS8-21272  
UNIDEV Report UR-00011

Submitted to  
GEORGE C. MARSHALL SPACE FLIGHT CENTER  
National Aeronautics and Space Administration  
Huntsville, Alabama 35812

## FOREWARD

The study presented in this report was performed for NASA - George C. Marshall Space Flight Center. The effort was performed under Contract Number NAS8-21272 entitled: "Mechanical Analog of Liquid Propellants in Low "G" Fields". The project was initiated by Melpar, and completed by UNIDEV Inc., Huntsville, Alabama.

This volume constitutes the final report on the computational application of the project. The theoretical section is presented in Volume I.

## TABLE OF CONTENTS

| <u>Section</u> |                                     | <u>Page</u> |
|----------------|-------------------------------------|-------------|
| 1              | Program Description                 | 1           |
| 2              | Equations Programmed                | 2           |
| 3              | Utilization of the Computer Program | 4           |
|                | 3.1 Description of Input            | 4           |
|                | 3.2 Description of Output           | 9           |
| 4              | References                          | 13          |
| 5              | Sample Input                        | 14          |
| 6              | Sample Output                       | 16          |
| 7              | Program Listing                     | 20          |

## SECTION 1

### PROGRAM DESCRIPTION

The existing program (Tank Sloshing Version 1, 3712) was modified to accept the equations described in Section 2. To make the necessary changes, three new routines were added and the Matrix routine had to be modified.

Gauss's Quadrature Method was used as the integration scheme and was written as a new routine, INTG3. The integrands of equations (2.5) and (2.6) were placed in the remaining two new routines.

In the Modified Matrix routine,  $\tau_{nm}$ ,  $\beta_{nm}$ ,  $\bar{\beta}_{nm}$ ,  $Q_n$ ,  $\bar{Q}_n$ , correction to natural frequency, and  $v_{nm}$  are computed and printed out. Equation (2.4) was integrated analytically and was also placed in the Matrix routine.

## SECTION 2

### EQUATIONS PROGRAMMED

The equations which describe the sloshing of liquid propellant in a tank consisting of an arbitrary curve rotated about an axis of symmetry under low gravity conditions are derived in Reference 1 and are:

$$\tau_{nm} = \sum_{i=1}^{N^*} \sum_{j=1}^{N^*} c_i^n c_j^m \left[ \int_0^1 \frac{\partial \phi_i^*}{\partial R} \frac{\partial \phi_j^*}{\partial R} R dR + \int_0^1 \frac{\phi_i^* \phi_j^*}{R} dR \right] \quad (2.1)$$

$$v_{nm} = \left[ \sum_{i=1}^M c_i^n + \sum_{i=M+1}^{N^*} c_i^n J_1(j_i) \right] \left[ \sum_{i=1}^M c_i^m + \sum_{i=M+1}^{N^*} c_i^m J_1(j_i) \right] \quad (2.2)$$

where the set of functions  $\phi_n^*(R)$  were chosen as

$$\phi_n^*(R) = \begin{cases} R^{2n-1} & n=1,2,\dots, M < N^* \\ J_1(j_n R) & n=M+1,\dots, N^* \end{cases} \quad (2.3)$$

From equations (2.1) and (2.3)  $\tau_{nm}$  is calculated as

$$\tau_{nm} = \sum_{i=1}^{N^*} \sum_{j=1}^{N^*} c_i^n c_j^m \left[ \int_0^1 \left\{ (2i-1)(2j-1)+1 \right\} R^{2i+2j-3} dR \right], \quad i, j \leq M \quad (2.4)$$

$$\tau_{nm} = \sum_{i=1}^{N^*} \sum_{j=1}^{N^*} c_i^n c_j^m \left[ \int_0^1 \left\{ (2i-1)(j_j R) J_1^1(j_j R) + J_1(j_j R) \right\} R^{2i-2} dR \right],$$

$$i < M < j \quad (2.5)$$

$$\tau_{nm} = \sum_{i=1}^{N^*} \sum_{j=1}^{N^*} c_i^n c_j^m \left[ \int_0^1 \left\{ j_i j_j J_1^1(j_i R) J_1^1(j_j R) R + \frac{J_1(j_i R) J_1(j_j R)}{R} \right\} dR \right],$$

$$i, j > M \quad (2.6)$$

$$\beta_{nm} = \frac{-\pi a^2 L}{V \gamma_n} \tau_{nm} - \frac{a K_m \cos \gamma}{L} \left\{ 2 \sqrt{\frac{\gamma_m}{\gamma_n}} \delta_{nm} - \frac{2 \pi L^3}{V K_n K_m \gamma_n} \tau_{nm} + \frac{\pi L^3}{V \gamma_n} \left( \frac{1}{K_n K_m} - \frac{a^2}{L^2} \right) v_{nm} \right\} \quad (2.7)$$

where  $\gamma$  = contact angle

$$\bar{\beta}_{nn} = 0 \quad (2.8)$$

$$\bar{\beta}_{nm} = \beta_{nm} \text{ (with } K_n \text{ replaced by } K_m \text{)} \quad (2.9)$$

$$\frac{\omega - \omega_n}{\omega_n} = - \frac{\beta_{nn}}{2 B_o} \quad (2.10)$$

where  $B_o$  = Bond Number

$$\bar{Q}_n = \frac{\pi a L^2 \cos \gamma}{K_n V \gamma_n} \left( \tau_{nn} - \frac{1}{2} v_{nn} \right) + \frac{1}{2} \beta_{nn} \quad (2.11)$$

$$Q_n = \frac{\pi a^2}{V \gamma_n} \left( a K_n \cos \gamma v_{nn} - \frac{L \tau_{nn}}{2} \right) - \frac{a K_n \cos \gamma}{L} \quad (2.12)$$

For definition of terms see References 1 and 2.

## SECTION 3

### UTILIZATION OF THE COMPUTER PROGRAM

**3.1 DESCRIPTION OF INPUT.** There are three sets of required input data, as described in Sections 3.1.1, 3.1.2 and 3.1.4. An optional fourth set is described in Section 3.1.3. Input is subject to the restrictions outlined in Section 3.1.5.

Data is entered in allocated spaces on a coding form. Each digit of a number, a decimal point, or a minus sign occupies 1 space. Unless otherwise specified in the following sections, a number must be entered with a decimal point. A number that must not contain a decimal point must be right-adjusted. That is, all digits must occupy the last of those spaces allocated for the number.

All coordinates required as input must be in inch units, from a system in which the r-coordinate of the tank centerline is zero and the z-coordinate increases upward. The origin may be anywhere on the centerline.

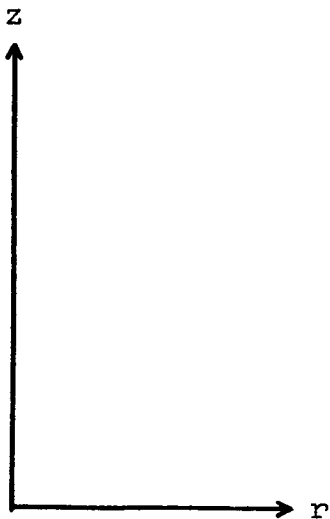


Figure 1. Coordinate System

**3.1.1 Problem Input.** The first three lines of the coding form contain general information about the problem. On the first line:

| <u>Spaces</u> | <u>Data</u>                                                         |
|---------------|---------------------------------------------------------------------|
| 1 through 72  | Enter any desired characters to be used as an identification title. |

On the second line:

| <u>Spaces</u> | <u>Data</u>             |
|---------------|-------------------------|
| 1 through 10  | Contact Angle (Radians) |
| 11 through 20 | Bond Number             |

On the third line:

| <u>Spaces</u> | <u>Data</u>                                                                                                   |
|---------------|---------------------------------------------------------------------------------------------------------------|
| 1 through 10  | Enter the number of segments of the curve. The number must be right-adjusted, without a decimal point.        |
| 11 through 20 | Enter the number of modes of oscillation desired. The number must be right-adjusted, without a decimal point. |
| 21 through 30 | Enter the liquid density (in kilograms per cubic meter).                                                      |
| 31 through 40 | Enter the r-coordinate (in inches) of the beginning of the segments of the curve.                             |
| 41 through 50 | Enter the z-coordinate (in inches) of the beginning of the segments of the curve.                             |
| 51 through 60 | Enter the number of baffles. The number must be right-adjusted, without a decimal point.                      |

3.1.2 Tank Geometry Input. Beginning with the fourth line of the coding form, a line of the data below is required for each segment of the curve. Segments must be ordered in a continuous, counterclockwise path around the curve. The number of these lines of input must equal the number entered in spaces 1 through 10 of the third line of Problem Input (Section 3.1.1).

| <u>Spaces</u> | <u>Data</u>                                                                                                                                                   |
|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 through 10  | Enter the r-coordinate (in inches) of the end of the particular segment, having proceeded along the segment in a counterclockwise direction around the curve. |
| 11 through 20 | Enter the z-coordinate (in inches) of the end of the particular segment, having proceeded along the segment in a counterclockwise direction around the curve. |

| <u>Spaces</u> | <u>Data</u>                                                                                                                                                                                                                                                                                                                                                             |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 21 through 30 | <p>Leave blank for a straight line segment; otherwise, enter as follows, depending on the type of segment:</p> <p>Elliptical. Enter the r-coordinate (in inches) of the center of the ellipse.</p> <p>Circular. Enter the r-coordinate (in inches) of the center of the circle.</p> <p>Parabolic. Enter the r-coordinate (in inches) of the vertex of the parabola.</p> |
| 31 through 40 | <p>Leave blank for a straight line segment; otherwise, enter as follows, depending on the type of segment:</p> <p>Elliptical. Enter the z-coordinate (in inches) of the center of the ellipse.</p> <p>Circular. Enter the z-coordinate (in inches) of the center of the circle.</p> <p>Parabolic. Enter the z-coordinate (in inches) of the vertex of the parabola.</p> |
| 41 through 50 | <p>Leave blank for a straight line segment; otherwise enter, depending on the type of the segment, as follows:</p> <p>Elliptical. Enter the semimajor axis (in inches) of the ellipse.</p> <p>Circular. Enter the radius (in inches) of the circle.</p> <p>Parabolic. Enter the directrix (in inches) of the parabola.</p>                                              |
| 51 through 60 | <p>Leave blank for a straight line or parabolic segment; otherwise enter, depending on the type of the segment, as follows:</p> <p>Elliptical. Enter the semiminor axis (in inches) of the ellipse.</p> <p>Circular. Enter the radius (in inches) of the circle.</p>                                                                                                    |

### Spaces

### Data

61 through 70      Leave blank for a straight line or circular segment; enter the amount of counterclockwise rotation (in degrees) from the norm (as shown in Figure 2) of the ellipse if the segment is elliptical, or of the parabola if the segment is parabolic. It may be left blank for an angle of zero degrees.

**3.1.3 Baffle Input.** This data is entered only if a number is entered in spaces 51 through 60 of the third line of Problem Input (Section 3.1.1). Beginning with the next line of the coding form, a line of the data below is required for each baffle. The number of these lines must equal the number entered in spaces 51 through 60 of the third line of Problem Input (Section 3.1.1).

### Spaces

### Data

1 through 10      Enter the z-coordinate (in inches) of the baffle.

11 through 20      Enter the r-coordinate (in inches) of the inner end of the baffle.

21 through 30      Enter the r-coordinate (in inches) of the outer end of the baffle.

### Normal Orientation of an Ellipse

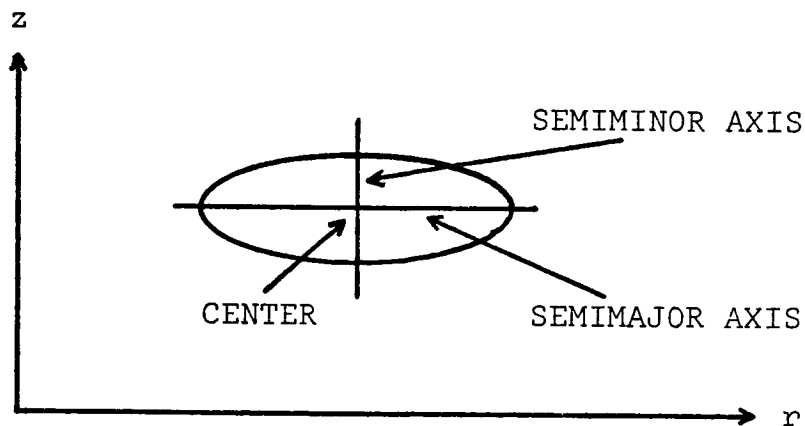
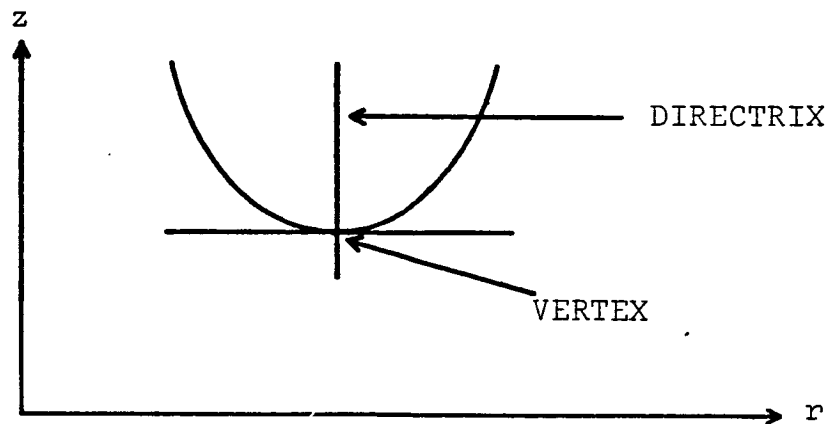


Figure 2. Normal Orientations  
for Tank Geometries

### Normal Orientation of a Parabola



3.1.4 Case Input. Each case to be run requires a line of input as listed below. Any number of cases may be specified.

| <u>Space(s)</u> | <u>Data</u>                                                                                                                                                                                  |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 through 10    | Enter the z-coordinate (in inches) of the undisturbed free surface.                                                                                                                          |
| 20              | Leave blank to indicate three degrees of freedom; otherwise, enter the digit 6 to indicate six degrees of freedom.                                                                           |
| 21 through 30   | Leave blank to indicate no force distribution coefficients; otherwise, enter the z-coordinate (in inches) of the point dividing the liquid into an UPPER and a LOWER region of partitions.   |
| 31 through 40   | Leave blank to indicate no force distribution coefficients; otherwise, enter the increment (in inches) along the z-axis between force distribution partitions in the UPPER region of liquid. |
| 41 through 50   | Leave blank to indicate no force distribution coefficients; otherwise, enter the increment (in inches) along the z-axis between force distribution partitions in the LOWER region of liquid. |

### 3.1.5 Restrictions on Input.

1. There can be no more than 50 segments.
2. There can be no more than 30 baffles.

3. There can be no more than 5 modes of oscillation.
4. Do not enter the tank centerline as a segment unless it is a segment of the curve.
5. The tank must not be completely full.
6. Each segment must be defined such that for a given r-coordinate, there is only a single value of the z-coordinate.

i.e. A toroid must be described by 4 segments of the same circle:

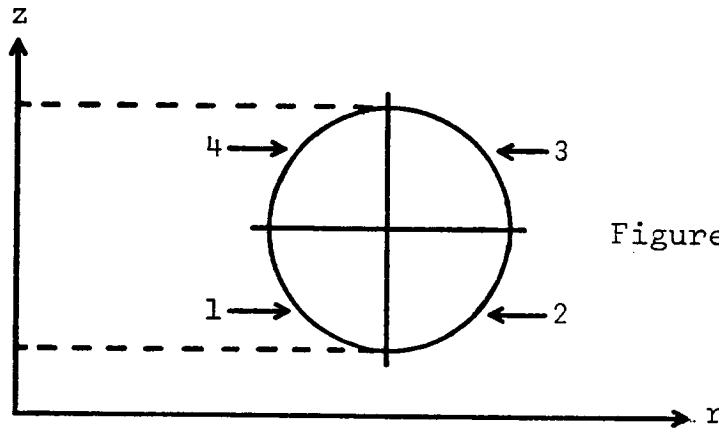


Figure 3. Toroid Geometry

3.2 DESCRIPTION OF OUTPUT. Program output consists of the following printed information:

A. Title Page

1. Name of the computer program
2. Identification title
3. Liquid density (in kilograms per cubic meter)
4. Contact Angle (radians)
5. Bond Number
6. Tank geometry - a description of all segments (in inches and degrees)
7. Lowest point in the tank (in inches)

B. Case Information (each case begins on a new page)

1. Identification title
2. Undisturbed free surface (in inches)
3. Mass of the liquid (in kg)
4. Undisturbed center of gravity (in inches)
5. Number of degrees of freedom
6. Eigenvalue statistics for each mode used - eigenvalue, and eigenvector and its normalizing factor
7. Force and moment equation nondimensional coefficients for each mode used:

$K_n$

$\gamma_n$

$b_n$

$h_n$

$\left. \begin{array}{l} a_n (=b_n) \\ d_n \end{array} \right\} \text{ for 6 degrees of freedom only}$

8. Maximum wave height at the outer radius of the undisturbed free surface/ $x_1$
  9. Kinetic energy/ $\dot{x}_1^2$  (in kg)
  10.  $I_{11}'$  and its 4 terms (in kg-m<sup>2</sup>)
  11.  $I_{22}'$  and its 4 terms (in kg-m<sup>2</sup>)
  12.  $I_{33}'$  (in kg-m<sup>2</sup>)
- for 6 degrees of freedom only
13. Force distribution coefficients for each partition used:  
z-coordinates of partition (in inches)  
 $\alpha_2$  coefficient (in kg), [ $A_n$  of (5.17)]

- $\ddot{u}$  coefficient (in kg-m), [ ${}_nB_u$  of (5.17)]  
 $\alpha_3\xi_1$  coefficient (in kg/m), [ ${}_nC_\xi$  of (5.17)]  
 $\alpha_3x_1$  coefficient (in kg/m), [ ${}_nC_\xi K_1 b_1$  of (5.22)]  
 $\ddot{\xi}_1$  coefficient (in kg), [ $Q_n$  of (5.24)]\*
14. Undisturbed center of pressure (in inches) only if force distribution coefficients have been calculated in the LOWER region of the liquid
15. Spring-mass analogy parameters for each mode used:
- $m_n$  (in kg)  
 $l_n$  (in inches)  
 $K_n^*/\alpha_3$  (in kg/m)
16. Spring-mass analogy parameters for ALL modes used:
- $m_o$  (in kg)  
 $l_o$  (in inches)  
 $I_o$  (in kg-m<sup>2</sup>)
- 17.
- $\tau$   
 $v$   
 $\beta$   
 $\overline{\beta}$   
 $Q$   
 $\overline{Q}$
18. Baffle effect on energy for each baffle used:
- Baffle z-coordinate (in inches)  
 Baffle r-coordinates (in inches)  
 Instantaneous dE/dt factor (in kg/m<sup>3/4</sup>)  
 (see (4.23) of Reference 2)

---

\* See Reference 2

Average  $dE/dt$  factor (in  $\text{kg/m}^{3/4}$ )  
(see (4.27) of Reference 2)

C. Diagnostics

If normal program calculations cannot be continued, an appropriate diagnostic comment is printed out and either execution is terminated or continued with assumptions, or a new case is begun.

## SECTION 4

### REFERENCES

1. Analysis of Fluid Sloshing in Arbitrary Tanks Having Rotational Symmetry - A Correction for Low Gravity Conditions, Vol. I, Theory, UNIDEV Report UR-00010, March 4, 1970.
2. D. O. Lomen, Analysis of Fluid Sloshing, General Dynamics Report GDC-DDE66-018, June 20, 1966

SECTION 5  
SAMPLE INPUT

Run Time - 2.76 min.

ATLAS LDX TANK WITH RING BAFFLE

ALSO A BAFFLE TO SHOW BAFFLE OUTPUT

0.7854

10.0

7

3 114.3363

0.0

43.5

1

60.0

0.0

0.0

00.0

60.0

43.5

60.0

46.0

56.0

46.0

56.0

47.0

60.0

47.0

60.0

75.0

0.0

75.0

50.0

50.0

60.0

55.0

SECTION 6  
SAMPLE OUTPUT

# ATLAS LUX TANK WITH RING BAFFLE

BAFFLE TO SHOW BAFFLE OUTPUT

LIQUID DENSITY = 0.11434E 03 KG/M<sup>3</sup>

CONTACT ANGLE = 0.78540E 00 RADIANS

BOND NUMBER = 0.10000E 02

## TANK GEOMETRY LENGTH UNIT IS INCHES

| SEGMENT | TYPE          | FROM                       |   | TO                         |   | CHARACTERISTICS    |                                      |
|---------|---------------|----------------------------|---|----------------------------|---|--------------------|--------------------------------------|
|         |               | R                          | Z | R                          | Z | SEMI-MAJOR AXIS    | SEMI-MINOR AXIS<br>CENTER AT R       |
| 1       | ELLIPTICAL    | R = 0.0000<br>Z = 43.5000  |   | R = 60.0000<br>Z = 0.0000  |   | 60.0000<br>43.5000 | ROTATED BY<br>-0.0 DEGREES<br>0.0000 |
| 2       | STRAIGHT LINE | R = 60.0000<br>Z = 0.0000  |   | R = 60.0000<br>Z = 46.0000 |   |                    |                                      |
| 3       | STRAIGHT LINE | R = 60.0000<br>Z = 46.0000 |   | R = 56.0000<br>Z = 46.0000 |   |                    |                                      |
| 4       | STRAIGHT LINE | R = 56.0000<br>Z = 46.0000 |   | R = 56.0000<br>Z = 47.0000 |   |                    |                                      |
| 5       | STRAIGHT LINE | R = 56.0000<br>Z = 47.0000 |   | R = 60.0000<br>Z = 47.0000 |   |                    |                                      |
| 6       | STRAIGHT LINE | R = 60.0000<br>Z = 47.0000 |   | R = 60.0000<br>Z = 75.0000 |   |                    |                                      |
| 7       | STRAIGHT LINE | R = 60.0000<br>Z = 75.0000 |   | R = 0.0000<br>Z = 75.0000  |   |                    |                                      |

LOWEST POINT IN THE TANK IS AT 0.0000 INCHES

ATLAS LOX TANK WITH RING BAFFLE ALSO BAFFLE TO SHOW BAFFLE OUTPUT

UNDISTURBED FREE SURFACE IS AT 55.0000 INCHES

MASS OF THE LIQUID = 0.54822E 03 KG

UNDISTURBED CENTER OF GRAVITY IS AT 39.9454 INCHES

3 DEGREES OF FREEDOM

| EIGENVALUE = 0.10398E 01 | EIGENVECTOR NORMALIZING FACTOR = -0.40660E 01 | EIGENVECTOR  |
|--------------------------|-----------------------------------------------|--------------|
|                          |                                               | 0.58104E 00  |
|                          |                                               | -0.88389E 00 |
|                          |                                               | 0.10000E 01  |
|                          |                                               | -0.65231E 00 |
|                          |                                               | 0.18270E 00  |
|                          |                                               | 0.26805E 00  |
|                          |                                               | -0.27167E-01 |
|                          |                                               | 0.16794E-02  |
|                          |                                               | -0.94783E-03 |
|                          |                                               | 0.10035E-02  |

| EIGENVALUE = 0.49273E 01 | EIGENVECTOR NORMALIZING FACTOR = -0.12461E 02 | EIGENVECTOR  |
|--------------------------|-----------------------------------------------|--------------|
|                          |                                               | 0.96832E-01  |
|                          |                                               | -0.52558E 00 |
|                          |                                               | 0.10000E 01  |
|                          |                                               | -0.83875E 00 |
|                          |                                               | 0.26206E 00  |
|                          |                                               | -0.29099E-02 |
|                          |                                               | 0.12006E 00  |
|                          |                                               | -0.30529E-02 |
|                          |                                               | 0.19078E-02  |
|                          |                                               | 0.33567E-03  |

| EIGENVALUE = 0.83917E 01 | EIGENVECTOR NORMALIZING FACTOR = -0.74610E 01 | EIGENVECTOR  |
|--------------------------|-----------------------------------------------|--------------|
|                          |                                               | 0.56773E-01  |
|                          |                                               | -0.42327E-00 |
|                          |                                               | 0.10000E 01  |
|                          |                                               | -0.97724E 00 |
|                          |                                               | 0.34132E 00  |
|                          |                                               | 0.30398E-02  |
|                          |                                               | -0.14342E-01 |
|                          |                                               | 0.23121E 00  |
|                          |                                               | 0.20139E-02  |
|                          |                                               | 0.25987E-02  |

| COEFFICIENTS | MODE | KN          | GAMMA       | DN           | HN           |
|--------------|------|-------------|-------------|--------------|--------------|
| 1            |      | 0.26088E-00 | 0.33850E-01 | -0.81729E 01 | 0.32318E 01  |
| 2            |      | 0.12363E 01 | 0.75811E-03 | -0.76397E 01 | -0.39208E 01 |
| 3            |      | 0.21055E 01 | 0.12450E-02 | 0.17484E 01  | 0.18011E 01  |

MAX. EFA/AN(1) = 0.83967E 00

KINETIC ENERGY/AN(1) = 0.2 = 0.16169E 03 KG

STEP = 0.00000E 03 KG-M-2 THERE THE TERMS ARE 0.42813E 03 -0.15950E 03 0.64551E 02 0.56437E 03

# SIM. MASS PARAMETERS

| MODEL                    | MM<br>IN KG | LN<br>IN INCHES | KNSTAR/ALPHA(3)<br>IN KG/M |
|--------------------------|-------------|-----------------|----------------------------|
| 1                        | 0.32330E 03 | 0.30845E 02     | 0.22063E 03                |
| 2                        | 0.29989E 02 | 0.32618E 02     | 0.96950E 02                |
| 3                        | 0.93950E 01 | 0.40400E 02     | 0.24200E 02                |
| MO = 0.19046E 03 KG      |             |                 |                            |
| LO = 0.56542E 02 INCHES  |             |                 |                            |
| LO = 0.28093E 03 KG-M**2 |             |                 |                            |

TAU

|                |                |                |
|----------------|----------------|----------------|
| 0.19934091E 00 | 0.83717520E-02 | 0.20401593E-02 |
| 0.03717520E-02 | 0.37083925E-01 | 0.29440123E-02 |
| 0.20401593E-02 | 0.29440123E-02 | 0.15585736E 00 |

WU

|                 |                 |                 |
|-----------------|-----------------|-----------------|
| 0.15503641E 03  | -0.20740109E-01 | 0.26625504E-01  |
| -0.20740109E-01 | 0.27736289E-02  | -0.35606902E-02 |
| 0.26625504E-01  | -0.35606902E-02 | 0.45711130E-02  |

DELTA

|                 |                 |                 |
|-----------------|-----------------|-----------------|
| -0.89343067E-01 | -0.94790910E 00 | 0.24172068E 01  |
| -0.14001254E 02 | -0.20150681E 02 | -0.17436213E 02 |
| 0.73043508E 01  | -0.67987229E 01 | -0.59914721E 02 |

DELTA BAR

|                |                 |                 |
|----------------|-----------------|-----------------|
| 0.00000000E-38 | -0.12936326E 01 | 0.26461692E 01  |
| 0.14354215E 01 | 0.00000000E-38  | -0.17865881E 02 |
| 0.10295097E 01 | -0.65372191E 01 | 0.00000000E-38  |

Q

|                 |                 |                 |
|-----------------|-----------------|-----------------|
| -0.48852679E 00 | -0.10298663E 02 | -0.29666664E 02 |
| 0.13792442E 01  | -0.61433319E 01 | -0.23912374E-02 |

NATURAL FREQUENCY CORRECTION

|                |                |                |
|----------------|----------------|----------------|
| 0.44671833E-02 | 0.10075340E 01 | 0.29957360E 01 |
|----------------|----------------|----------------|

|        |                           |                            |                                         |                                        |
|--------|---------------------------|----------------------------|-----------------------------------------|----------------------------------------|
| BAFFLE | Z-COORDINATE<br>IN INCHES | K-COORDINATES<br>IN INCHES | INST. DE/DT FACTOR<br>IN KG/M**2 (3.74) | AVG. DE/DT FACTOR<br>IN KG/M**2 (3.74) |
| 1      | 50.0000                   | 50.0000                    | -0.55007E 02                            | -0.14476E 02                           |

SECTION 7  
PROGRAM LISTING

21

C

```

DO 50 I=1,NSEG
  READ (5,10) RPOINT(I),ZPOINT(I),RC(I),ZC(I),A(I),B(I),GAMMA(I)
10 FORMAT (7F10.0)
  IF (RPOINT(I-1).LT.0.0.OR.RPOINT(I).LT.0.0.OR.RC(I).LT.0.0.OR.
  2A(I).LT.0.0.OR.B(I).LT.0.0.OR.GAMMA(I).LT.0.0.OR.GAMMA(I).GT.360.0
  3) GO TO 65
  IF (ZPOINT(I).LT.ZBOT) ZBOT=ZPOINT(I)
  IF (B(I).EQ.0.0) GO TO 30
  TYPE(I)=1
  IF (A(I).NE.0(I)) GO TO 20
  WRITE (6,15) I,RPOINT(I-1),RPOINT(I),A(I),ZPOINT(I-1),ZPOINT(I),RC
  2(I),ZC(I)
15 FORMAT (1H0,2X,12,7X,3HCIRCULAR,6X,4HR = ,F10.4,7X,
  24HR = ,F10.4,5X,          9HRADIUS = ,F10.4          /26X,4HZ = ,
  3F10.4,7X,          4HZ = ,F10.4,5X,          14HCENTER AT R = ,
  4F10.4,7H AND Z = ,F10.4)
  GO TO 50
20 WRITE (6,25) I,RPOINT(I-1),RPOINT(I),A(I),ZPOINT(I-1),ZPOINT(I),
  2B(I),GAMMA(I),RC(I),ZC(I)
25 FORMAT (1H0,2X,12,6X,10HELLOPTICAL,6X,4HR = ,F10.4,7X,
  24HR = ,F10.4,5X,          18HSEMI-MAJOR AXIS = ,F10.4,9X,
  310HROTATED BY/26X,4HZ = ,F10.4,7X,          4HZ = ,F10.4,
  45X,18HSEMI-MINOR AXIS = ,F10.4,7X,          F6.1,8H DEGREES/66X,
  51HCENTER AT R = ,F10.4,9H AND Z = ,F10.4)
  GO TO 50
30 IF (A(I).NE.0.0) GO TO 40
  TYPE(I)=2
  WRITE (6,35) I,RPOINT(I-1),RPOINT(I),ZPOINT(I-1),ZPOINT(I)
35 FORMAT (1H0,2X,12,4X,13HSTRAIGHT LINE,4X,4HR = ,F10.4,7X,
  24HR = ,F10.4          /26X,4HZ = ,F10.4,7X,4HZ = ,F10.4)
  GO TO 50
40 TYPE(I)=3
  WRITE (6,45) I,RPOINT(I-1),RPOINT(I),A(I),ZPOINT(I-1),ZPOINT(I),GA
  2MMA(I),RC(I),ZC(I)
45 FORMAT (1H0,2X,12,6X,9HPARABOLIC,6X,4HR = ,F10.4,7X,
  24HR = ,F10.4,5X,          12HDIRECTRIX = ,F10.4,9X,
  310HROTATED BY/26X,4HZ = ,F10.4,7X,          4HZ = ,F10.4,
  436X,F6.1,8H DEGREES/66X,14HVERTEX AT R = ,F10.4,9H AND Z = ,F10.4)
50 CONTINUE
  WRITE (6,55) ZBOT
55 FORMAT (1H0,31HLOWEST POINT IN THE TANK IS AT ,F10.4,7H INCHES)
  IF (BEGZPT.EQ.ZPOINT(NSEG).AND.BEGRPT.EQ.RPOINT(NSEG).OR.
  2BEGZPT.NE.ZPOINT(NSEG).AND.BEGRPT.EQ.0.0.AND.RPOINT(NSEG).EQ.0.0)
  3GO TO 75
  WRITE (6,60)
60 FORMAT (1H0, 18HTANK IS NOT CLOSED)
  CALL EXIT
65 WRITE (6,70) I
70 FORMAT (1H0, 15HSEGMENT NUMBER ,12,1H IS INVALID)
  CALL EXIT
75 ZBOT=ZBOT*0.0254
  BEGRPT=BEGRPT*0.0254
  BEGZPT=BEGZPT*0.0254
  DO 60 I=1,NSEG
  RPOINT(I)=RPOINT(I)*0.0254

```

MAIN - EFN SOURCE STATEMENT - IFN(S) -

```

ZPOINT(I)=ZPOINT(I)*0.0254
RC(I)=RC(I)*0.0254
ZC(I)=ZC(I)*0.0254
A(I)=A(I)*0.0254
B(I)=B(I)*0.0254
GAM(I)=GAMMA(I)*PI/180.0

```

80 CONTINUE

C

C

C

C

READ BAFFLE INPUT CARD(S)

IF (NBAF.LT.1.OR.NBAF.GT.30) GO TO 100

DO 95 I=1,NBAF

READ (5,90) ZBAF(I),RBAFI(I),RBAFO(I)

90 FORMAT (3F10.0)

ZBAF(I)=ZBAF(I)\*0.0254

RBAFI(I)=RBAFI(I)\*0.0254

RBAFO(I)=RBAFO(I)\*0.0254

95 CONTINUE

C

C

C

C

READ A CASE INPUT CARD AND BEGIN A NEW CASE OR TERMINATE RUN

100 READ (5,105) LEVEL,IDEGR,ZDIV,DZ(1),DZ(2)

105 FORMAT (F10.0,I10,3F10.0)

LEVEL=LEVEL\*0.0254

ZDIV=ZDIV\*0.0254

DZ(1)=DZ(1)\*0.0254

DZ(2)=DZ(2)\*0.0254

IF (LEVEL.LE.ZBOT) GO TO 170

IF (IDEGR.NE.6) IDEGR=3

C

C

C

C

C

FIND THE OUTER RADIUS AND THE RATIO OF THE INNER RADIUS TO THE  
OUTER RADIUS OF THE UNDISTURBED FREE SURFACE

AS=0.0

TEMP(1)=0.0

IFLAG=0

DO 130 I=1,NSEG

IF (ZPOINT(I).EQ.ZPOINT(I-1)) GO TO 130

IF (IFLAG.EQ.0) GO TO 115

IFLAG=0

GO TO 130

115 ARG1=LEVEL-ZPOINT(I)

ARG2=LEVEL-ZPOINT(I-1)

IF (ARG1.NE.0.0) GO TO 120

RR=RPOINT(I)

IFLAG=1

GO TO 125

120 IF (ARG1/ABS(ARG1).EQ.ARG2/ABS(ARG2)) GO TO 130

CALL FINDR (I,LEVEL,RR)

125 IF (AS.LT.RR) AS=RR

IF (TEMP(1).GT.0.0) GO TO 135

TEMP(1)=RR

130 CONTINUE

IF (AS.LE.U.U), GO TO 170

EPSL=0.0

GO TO 140

135 IF (TEMP(1).GT.RK) TEMP(1)=RK

EPSL=TEMP(1)/AS

C

C

C

FIND THE UNDISTURBED CENTER OF GRAVITY AND THE LIQUID VOLUME

C

140 ZCG=0.0

CALL INTG1 (F1,LEVEL,ZBOT)

ZCG=ANS

CALL INTG1 (F2,LEVEL,ZBOT)

ZCG=ZCG/ANS

VOL=ANS\*PI

C

C

L=LEVEL-ZCG

D=ZCG-ZBOT

H=LEVEL-ZBOT

C

C

C

ESTABLISH THE CENTER OF ROTATION

C

G1=0.0

G2=0.0

G3=0

C

C

C

DETERMINE THE NUMBER OF EIGENFUNCTIONS TO BE USED FOR THE  
EIGENVALUE PROBLEM

C

C

TEMP(1)=68.029692/(2.0\*L/AS)

DO 145 I=1,5

J=6-I

IF (JN(J).LT.TEMP(1)) GO TO 150

145 CONTINUE

GO TO 170

150 MORDER=MPOLY+J

C

C

C

CHECK VALIDITY OF MODES

C

IF (MODES.LT.1.OR.MODES.GT.J) MODES=J

C

C

TEMP(1)=LEVEL/0.0254

TEMP(2)=RK\*VOL

TEMP(3)=ZCG/0.0254

WRITE (6,165) TITLE,TEMP,1DEGR

165 FORMAT (1H1,23X,12A6///

21X,31HUNDISTURBED FREE SURFACE IS AT ,F10.4,7H INCHES///

35X,21HMASS OF THE LIQUID = ,E12.5,3H KG//

45X,36HUNDISTURBED CENTER OF GRAVITY IS AT ,F10.4,7H INCHES//

55X,11,1X,10HDEGREES OF FREEDOM)

C

C

```
CON1=AS*COS(GAMMAA)
CON2=PI*AS**2*L/VOL
CON3=CON1/L
CON4=PI*L**3/VOL
CON5=2.0*CON4
CON6=(AS/L)**2
CON7=CON1*PI*L**2/VOL
CON8=PI*AS**2/VOL
CON9=L/2.0
CALL MATRIX
GO TO 100
```

C

C

```
170 TEMP(1)=LEVEL/0.0254
   WRITE (6,175) TITLE,TEMP(1)
175 FORMAT (1H1,23X,12A6///1X,28HUNDISTURBED FREE SURFACE AT .F10.4,
218H INCHES IS INVALID)
GO TO 100
```

C

C

END

SIBFTC MATRIX LIST, DECK

## SUBROUTINE MATRIX

```

DIMENSION BETA(10,10), BETAB(10,10), J(10), QB(10), OMEGAC(10)
DIMENSION XINT(10,10), XINT2(10,10), TAU(10,10), XNU(10,10),
& SUMM(10)
DIMENSION ZFORCE(3), ALPHIN(10), CCALC(5), CPRINT(5)
DIMENSION UJN(5)
DIMENSION AMN(10,10), BMN(10,10), A(10,10), B(10,10)
DIMENSION CNK(10,10), LAMBDA(10), KNSTAR(10)
DIMENSION KN(5), GAMMAN(5), UN(5), HN(5), LN(5), MN(5), KNSTAR(5), UN(5)
DIMENSION LTMAT(10,10), RTSIDE(10), QN(10), TERM(4)
COMMON NSEG, LEVEL, ZCG, L, D, H, AS, VOL, PI, EPSL, KHO, MODES, ZBOT, IDEGR,
ZG1, ZG2, ZG3, ZDIV, DZ(2)
COMMON /GAMBO/ GAMMAA, BU
COMMON /CON/ CON1, CON2, CON3, CON4, CON5, CON6, CON7, CON8, CON9
COMMON /BAF/ NBAF, ZBAF(30), RBAF1(30), RBAF0(30)
COMMON /ELEM/ MORDER, MPOLY, JN(5), NMORDER, NPOLY, ACDEF(10,10)
COMMON /GAUSS/ NORD, X(8), W(8)
COMMON /PARAM/ M, N, JNM, JNN, RR, ZZ, FUNC, ANS
COMMON /BES/ B11, B1D1, B1J, B1D2, X11, XJ1, I, J, J1, I1, JJ
REAL JN, JNM, JNN, JNN
REAL L, LAMBDA, KN, KNSTAR, LN, MN, MO, LO, IO, LEVEL
REAL I11P, I33P, LTMAT
EXTERNAL F1, F2, F4, F5, F6, F7, F8, F9, F12, F14, F15, F16, F17, F18
EXTERNAL F28, F29, F30
DATA UJN/0.58186522, -0.3461262, 0.27329994, -0.23330442, 0.20701265/

```

C  
C  
C  
C

## EIGENVALUE PROBLEM

```

DO 60 M=1, NMORDER
DO 60 N=1, NMORDER
IF (M.GT.NPOLY) GO TO 35
IF (N.GT.NPOLY) GO TO 15
CALL INTG2 (F6, LEVEL, ZBOT)
AMN(M,N) = -(4.*FLOAT(M*N) - 2.*FLOAT(M+N) + 2.)*ANS
BMN(M,N) = (1. - EPSL.*(2*(M+N)))/(2.*FLOAT(M+N))
GO TO 50
15 I=N-MPOLY
JNN=JN(I)
CALL INTG2 (F7, LEVEL, ZBOT)
AMN(M,N) = -EXP(-JNN*L/AS)/JNN*ANS
CALL BESSEL (JNN, B1P, B1D)
ARG=JNN*EPSL
CALL BESSEL (ARG, B1, B1D)
IF (M.GT.1) GO TO 25
BMN(1,N) = 1./JNN**3*(JNN*B1P-ARG*B1+ARG**2*B1D)
GO TO 50
25 BMN(M,N) = 1./JNN**2*(FLOAT(2*M-1)*B1P-EPSL.*(2*M-1)*(FLOAT(2*M-1)*
ZB1-ARG*B1D)-4.*FLOAT(H*(M-1))*BMN(M-1,N))
GO TO 50
35 I=N-MPOLY
JNM=JN(I)
I=N-MPOLY
JNN=JN(I)
CALL INTG2 (F8, LEVEL, ZBOT)

```

MATRIX - EFN SOURCE STATEMENT - IFN(S) -

```

SUMJ=JNM+JNN
AMN(M,N)=-EXP(-SUMJ*L/AS)/SUMJ*ANS
IF (M.EQ.N) GO TO 45
ARG1=JNN*EPSL
CALL BESSEL (ARG1,BIN,BIDN)
ARG2=JNM*EPSL
CALL BESSEL (ARG2,BIM,BIDM)
BMN(M,N)=(ARG2*BIN*BIDM-ARG1*BIM*BIDN)/(ARG2**2-ARG1**2)*EPSL**2
GO TO 50

```

```

45 CALL BESSEL (JNN,BIP,BID)
ARG=JNN*EPSL
CALL BESSEL (ARG,BI,BID)
BMN(M,N)=((JNN**2-1.)*BIP**2-(ARG*BI)**2-(ARG**2-1.)*BI**2)/
2(2**JNN**2)

```

```

50 AMN(N,M)=AMN(M,N)
BMN(N,M)=BMN(M,N)
A(1,N)=AMN(M,N)
A(N,M)=AMN(M,N)
B(1,N)=BMN(M,N)
B(N,M)=BMN(M,N)

```

```

60 CONTINUE

```

C

```

CALL JACOBI (JO,MORDER,IERR,B,A,CNK,LAMBDA,MODES)

```

```

IF (IERR.EQ.0) GO TO 100

```

```

WRITE (6,80)

```

```

80 FORMAT (1H0,35EIGENVALUE PROBLEM CANNOT BE SOLVED)

```

```

RETURN

```

C

C

C

C

NON-DIMENSIONAL COEFFICIENTS

```

100 DO 105 N=1,MODES

```

```

GAMMAN(N)=0.0

```

```

DO 105 K=1,MORDER

```

```

DO 105 J=1,MORDER

```

```

105 GAMMAN(N)=GAMMAN(N)+PI*L*AS**2/VOL*CNK(K,N)*CNK(J,N)*AMN(K,J)

```

```

DO 110 M=1,MORDER

```

```

CALL INTEG1 (F9,LEVEL,ZBOT)

```

```

110 HNSTAR(M)=ANS

```

```

DO 115 N=1,MODES

```

```

HN(N)=0.0

```

```

BN(N)=0.0

```

```

KN(N)=L/(LAMBDA(N)*AS)

```

```

DO 115 M=1,MORDER

```

```

HN(N)=HN(N)+2.*PI*AS**3/(VOL*GAMMAN(N)*KN(N))*CNK(M,N)*HNSTAR(M)

```

```

115 BN(N)=BN(N)+PI*AS**3/(VOL*GAMMAN(N))*CNK(M,N)*BMN(1,M)

```

```

IF (IDEGR.NE.6) GO TO 295

```

```

WRITE (6,130)

```

```

130 FORMAT (1H0,4X,21HCOEFFICIENTS MODE,10X,2HKN,14X,6HGAMMAN,12X,

```

```

27HAN = BN,13X,2HBN,16X,2HDN//)

```

```

DO 140 N=1,MODES

```

```

DN(N)=2.0*BN(N)-HN(N)

```

```

140 WRITE (6,145) N,KN(N),GAMMAN(N),BN(N),HN(N),DN(N)

```

```

145 FORMAT (24X,11,5E18.5)

```

```

GO TO 750

```

```

295 WRITE (6,300)

```

MATRIX - EFN SOURCE STATEMENT - IFN(S) -

300 FORMAT (1H0,4X,21HCoefficients      MODE,10X,2HKN,14X,6HGAMMA,14X,  
22HBN,16X,2HHN//)

DO 305 N=1,MODES

305 WRITE (6,310) N,KN(N),GAMMA(N),BN(N),HN(N)

310 FORMAT (24X,11,4E18.5)

C

C

C

MAXIMUM WAVE HEIGHT AT THE OUTER RADIUS OF THE UNDISTURBED

C

FREE SURFACE AND THE KINETIC ENERGY

C

750 TERM(1)=0.0

DO 770 J=1,MORDER

IF (J.GT.MPOLY) GO TO 760

TERM(1)=TERM(1)+CNK(J,1)

GO TO 770

760 I=J-MPOLY

TERM(1)=TERM(1)+CNK(J,1)\*JUN(I)

770 CONTINUE

TERM(1)=TERM(1)\*KN(1)\*BN(1)

TERM(2)=0.0

DO 780 J=1,MORDER

DO 780 K=1,MORDER

780 TERM(2)=TERM(2)+CNK(J,1)\*CNK(K,1)\*AMN(J,K)

TERM(2)=TERM(2)\*RHU\*PI\*AS\*L\*\*2\*BN(1)\*\*2/2.0

WRITE (6,785) TERM(1),TERM(2)

785 FORMAT (1H0,4X,17HMAX. ETA/XN(1) = ,E12.5//

25X,27HKINETIC ENERGY/XN(1)\*\*2 = ,E12.5,3H KG)

C

C

C

CALCULATE THE POLYNOMIAL COEFFICIENTS OF THE EIGENFUNCTIONS

C

ASSUMED FOR THE BVP NOS. 1 AND 2

C

DO 800 N=1,NPOLY

800 ACOEF(N,1)=1.0

DO 805 N=2,NPOLY

DO 805 K=2,N,2

805 ACOEF(N,K)=0.0

DO 810 N=3,NPOLY

DO 810 K=3,N,2

810 ACOEF(N,K)=-FLOAT((N-K+2)\*(N-K+1))/FLOAT(K\*\*2-1)\*ACOE(N,K-2)

C

C

C

BOUNDARY VALUE PROBLEM NO. 1

C

CALL INTG2 (F4,LEVEL,ZBOT)

TERM(1)=-PI\*RHO\*ANS

C

CALL INTG2 (F5,LEVEL,ZBOT)

TERM(2)=8./3.\*PI\*RHO\*ANS

C

TERM(3)=0.0

DO 223 M=1,MORDER

CALL INTG1 (F18,LEVEL,ZBOT)

RTSIDE(M)=2.0\*AS\*\*2/L\*\*2\*ANS

DO 223 N=M,MORDER

IF (M.GT.NPOLY) GO TO 222

MATRIX - EFN SOURCE STATEMENT - IFN(S) -

IF (N.GT.NPOLY) GO TO 220  
CALL INTG2 (F16,LEVEL,ZBOT)

LTMAT(N,N)=-ANS  
GO TO 223

220 CALL INTG2 (F17,LEVEL,ZBOT)

LTMAT(N,N)=-ANS  
GO TO 223

222 CALL INTG2 (F15,LEVEL,ZBOT)

LTMAT(N,N)=ANS

223 LTMAT(N,M)=LTMAT(M,N)

CALL SOLVE (NORDER,LTMAT,RTSIDE,1,1.0E-8,25,QN,IERR)

IF (IERR.GT.0) GO TO 245

WRITE (6,230)

230 FORMAT (1H0,4X,65HTHIRD TERM OF 111P CANNOT BE CALCULATED - IT IS  
2SET EQUAL TO ZERO)

GO TO 255

245 DO 250 M=1,NORDER

CALL INIG1 (F18,LEVEL,ZBOT)

250 TERM(3)=TERM(3)+2.0\*RHO\*PI\*L\*\*2\*AS\*\*3\*QN(M)\*ANS

C

255 TERM(4)=(G2\*\*2+G3\*\*2)\*RHO\*VOL

C

111P=TERM(1)+TERM(2)+TERM(3)+TERM(4)

WRITE (6,260) 111P,(TERM(1),I=1,4)

260 FORMAT (1H0,4X,7H111P = ,E12.5,28H KG-M\*\*2 WHERE THE TERMS ARE,  
24E16.5)

C

C

IF (IDEGR.NE.6) GO TO 2950

C

C

C

BOUNDARY VALUE PROBLEM NO. 2

C

C

C

122P IS THE SAME AS 111P

C

C

TERM(1) FOR THE BOUNDARY VALUE PROBLEM NO. 2 IS THE SAME

C

AS FOR THE BOUNDARY VALUE PROBLEM NO. 1

C

CALL INTG2 (F14,LEVEL,ZBOT)

TERM(2)=4.0\*RHO\*PI\*ANS

C

TERM(4)=(G1\*\*2+G3\*\*2)\*RHO\*VOL

C

IF (IERR.GT.0) GO TO 425

WRITE (6,410)

410 FORMAT (1H0,4X,65HTHIRD TERM OF 122P CANNOT BE CALCULATED - IT IS  
2SET EQUAL TO ZERO)

TERM(3)=0.0

GO TO 435

425 TERM(3)=111P-TERM(1)-TERM(2)-TERM(4)

C

435 WRITE (6,450) 111P,(TERM(1),I=1,4)

450 FORMAT (1H0,4X,7H122P = ,E12.5,28H KG-M\*\*2 WHERE THE TERMS ARE,  
24E16.5)

C

C  
C BOUNDARY VALUE PROBLEM NO. 3

C  
133P=(G1\*\*2+G2\*\*2)\*RHO\*VOL  
WRITE (6,500) 133P  
500 FORMAT (1H0,4X,7H133P = ,E12.5,8H KG-M\*\*2)

C  
C  
RETURN

C  
C  
C FORCE DISTRIBUTION COEFFICIENTS

C  
2950 IF (DZ(1).EQ.0.0.AND.DZ(2).EQ.0.0) GO TO 200  
ARG1=0.0  
ARG3=0.0  
DO 2955 J=1,5  
2955 CCALC(J)=0.0  
IF (DZ(1).NE.0.0) GO TO 2960  
I=2  
ZFORCE(1)=ZDIV-DZ(2)  
ZFORCE(2)=ZBUT  
GO TO 2965  
2960 I=1  
ZFORCE(1)=LEVEL-DZ(1)  
ZFORCE(2)=ZDIV  
2965 ZFORCE(3)=LEVEL  
ARG3=LEVEL/0.0254  
WRITE (6,2970) ARG3  
2970 FORMAT (1H0,4X,31HFORCE DISTRIBUTION COEFFICIENTS//  
26X,12H2-COORDINATE,  
35X,14HALPHA(2) COEF.,  
46X,13H1THETA.. COEF.,  
52X,20HALPHA(3)\*XI(1) COEF.,  
61X,20HALPHA(3)\*XN(1) COEF.,  
71X,13HXI(1).. COEF./  
88X,7HIN INCHES,  
911X,5HIN KG,  
213X,7HIN KG-M,  
321X,7HIN KG/M,  
423X,5HIN KG//  
5F17.4)

C  
3000 N=1  
ARG3=ZFORCE(1)/0.0254  
DO 3015 N=1,MORDER  
IF (N.GT.MPOLY) GO TO 3010  
CALL INTG2 (F3,LEVEL,ZFORCE(1))  
ALPHIN(N)=-(.4.\*FLOAT(N)-2.\*FLOAT(N+1)+2.)\*ANS  
GO TO 3015  
3010 J=N-MPOLY  
JNN=JN(J)  
CALL INTG2 (F7,LEVEL,ZFORCE(1))  
ALPHIN(N)=-EXP(-JNN\*L/AS)/JNN\*ANS  
3015 CONTINUE  
ARG1=0.0

```

DO 3020 N=1,MURDER
3020 ARG1=ARG1+PI*AS**2*CNK(N,1)*ALPHIN(N)
CALL INTG1 (F2,LEVEL,ZFORCE(1))
VM=PI*ANS

```

C

```

CPRINT(1)=CCALC(1)
CCALC(1)=-RHO*(VM-BN(1)*L*ARG1)
CPRINT(1)=CCALC(1)-CPRINT(1)

```

C

```

CPRINT(3)=CCALC(3)
CCALC(3)=RHO*ARG1
CPRINT(3)=CCALC(3)-CPRINT(3)

```

C

```

CPRINT(4)=CCALC(4)
CCALC(4)=CCALC(3)*KN(1)*BN(1)
CPRINT(4)=CCALC(4)-CPRINT(4)

```

C

```

CPRINT(5)=CCALC(5)
CCALC(5)=-L/KN(1)*CCALC(3)
CPRINT(5)=CCALC(5)-CPRINT(5)
IF (DZ(2).EQ.0.0) GO TO 3025
CALL INTG1 (F1,ZFORCE(3),ZFORCE(1))
ZCGM=ANS
CALL INTG1 (F2,ZFORCE(3),ZFORCE(1))
ZCGM=ZCGM/ANS+ZCG
ARG4=ARG4+CPRINT(5)*ZCGM
ARG5=ARG5+CPRINT(5)

```

C

```

3025 IF (IERR.GT.0) GO TO 3045
WRITE (6,3030) CPRINT(1),(CPRINT(J),J=3,5),ARG3
3030 FORMAT (17X,E19.5,19X,3E19.5/F17.4)
GO TO 3995

```

```

3045 ARG2=0.0
DO 3060 N=1,MURDER
IF (N.GT.NPOLY) GO TO 3050
CALL INTG2 (F16,LEVEL,ZFORCE(1))
GO TO 3060

```

```

3050 CALL INTG2 (F17,LEVEL,ZFORCE(1))
3060 ARG2=ARG2-PI*(AS*L)**2*QN(N)*ANS

```

C

```

CPRINT(2)=CCALC(2)
CALL INTG1 (F1,LEVEL,ZFORCE(1))
CCALC(2)=RHO*(PI*ANS-G3*VM-ARG2+(G3*BN(1)-L*(BN(1)-HJ(1)))*L*ARG1)
CPRINT(2)=CCALC(2)-CPRINT(2)

```

C

```

WRITE (6,3980) (CPRINT(J),J=1,5),ARG3
3980 FORMAT (17X,5E19.5/F17.4)

```

C

```

3995 ZFORCE(3)=ZFORCE(1)
ZFORCE(1)=ZFORCE(1)-DZ(1)
IF (ZFORCE(1).GE.ZFORCE(2)) GO TO 3000
IF (ZFORCE(2).EQ.ZFORCE(3)) GO TO 4000
ZFORCE(1)=ZFORCE(2)
GO TO 3000
4000 IF (DZ(2).EQ.0.0) GO TO 200
IF (1.EQ.2) GO TO 4005

```

```

I=2
ZFORCE(1)=ZDIV-DZ(2)
ZFORCE(2)=ZBOT
GO TO 3000

```

```

C
4005 ZCP=ARG4/ARG5/0.0254
      WRITE (6,4010) ZCP
4010 FORMAT (1H0,4X,37HUNDISTURBED CENTER OF PRESSURE IS AT ,F10.4,
      27H INCHES)

```

```

C
C
C      SPRING-MASS ANALOGY PARAMETERS
C

```

```

200 DO 205 N=1,MODES
      LN(N)=G3-L+L*HN(N)/BN(N)
      MN(N)=RHO*VOL*(GAMMAN(N)*BN(N)**2*KN(N)
205 KNSTAR(N)=MN(N)*KN(N)/L
      MD=RHO*VOL
      DO 210 N=1,MODES
210 MD=MD-MN(N)
      LD=G3*RHO*VOL/MD
      DO 215 N=1,MODES
215 LD=LD-RHO*VOL*(GAMMAN(N)*BN(N)*KN(N)+(G3*BN(N)-L*(BN(N)-HN(N))))
      2/MD
      ID=111P-MD*LD**2
      DO 290 N=1,MODES
290 ID=ID-MN(N)*LN(N)**2
      WRITE (6,315)
315 FORMAT (1H0,4X,31HSPRING-MASS PARAMETERS      MODE,9X,2HMD,15X,
      22HLD,9X,15HKNSTAR/ALPHA(3)/
      34X,5HIN KG,10X,9HIN INCHES,10X,7HIN KG/M//)
      DO 320 N=1,MODES
      LN(N)=LN(N)/0.0254
320 WRITE (6,325) N,MN(N),LN(N),KNSTAR(N)
325 FORMAT (34X,11,3E17.5)
      LD=LD/0.0254
      WRITE (6,330) MD,LD,ID
330 FORMAT (1H0,31X,5HMD = ,E12.5,3H KG/
      232X,5HLD = ,E12.5,7H INCHES/
      332X,5HID = ,E12.5,8H KG-M**2)

```

```

C
C      COMPUTE INTEGRAL FOR TAU
C

```

```

      DO 75 I=1,MORDER
      XI1=2*I-1
      DO 75 J=1,MORDER
      J1=2*J-1
      XJ1=J1
      IF(I.GT.MPOLY) GO TO 65
      IF(J.GT.MPOLY) GO TO 70
      ANS=((XI1*XJ1+1.0)/(XI1+XJ1))
      XINT(I,J)=ANS
      GO TO 75
65  II=1-MPOLY
      JNN=JN(II)
      JJ=J-MPOLY

```

```

JNM=JN(JJ)
CALL INTG3(F28)
XINT(1,J)=ANS
GO TO 75
70 JJ=J-HPOLY
JNM=JN(JJ)
CALL INTG3(F29)
XINT(1,J)=ANS
75 CONTINUE

C
C   COMPUTE TAU AND XNU
C
DO 95 M=1,MODES
DO 87 N=M,MODES
TAU(M,N)=0.0
DO 87 I=1,MORDER
DO 87 J=1,MORDER
IF(I.GT.J) GO TO 81
XINT2(1,J)=XINT(1,J)
GO TO 85
81 XINT2(1,J)=XINT(J,I)
85 TAU(M,N)=TAU(M,N)+CNK(I,N)*CNK(J,M)*XINT2(1,J)
87 TAU(M,M)=TAU(M,N)
SUMM(M)=0.0
DO 90 I=1,HPOLY
SUMM(M)=SUMM(M)+CNK(I,M)
90 CONTINUE
M1=MPOLY+1
DO 92 I=M1,MORDER
I1=I-HPOLY
JNM=JN(I1)
CALL BESSEL(JNM,B1,B1D)
SUMM(M)=SUMM(M)+CNK(I,M)*B1
92 CONTINUE
95 CONTINUE
DO 124 N=1,MODES
DO 126 M=1,MODES
XNU(N,M)=SUMM(N)*SUMM(M)
IF(N.EQ.0) GO TO 116
DELNM=0.0
GO TO 118
116 DELNM=1.0
BETAB(N,N)=0.0
GO TO 120
118 BETAB(N,N)=-CON2*TAU(N,N)/GAMMAN(N)-CON3*KN(N)*
$(-CON5*TAU(N,N)/(KN(N)*KN(N)*GAMMAN(N))+CON4*(1.0/(KN(N)*KN(N))
$-CON6)*XNU(N,N)/GAMMAN(N))
120 BETA(N,M)=-CON2*TAU(N,M)/GAMMAN(N)-CON3*KN(M)*(2.0*SQRT
$(GAMMAN(M)/GAMMAN(N))*DELM-CON5*TAU(N,M)/(KN(N)*KN(N)*GAMMAN(N))
$+CON4*(1.0/(KN(N)*KN(N))-CON6)*XNU(N,M)/GAMMAN(N))
126 CONTINUE
OMEGAC(N)=-0.5*BETA(N,N)/80
QB(N)=CON7*(TAU(N,N)-0.5*XNU(N,N))/(KN(N)*GAMMAN(N))
$+0.5*BETA(N,N)
Q(N)=CON8*(CON1*KN(N)*XNU(N,N)-CON9*TAU(N,N))/GAMMAN(N)
$-CON3*KN(N)

```

128 CONTINUE

```

WRITE(6,11117) ((TAU(I,J),J=1,MODES),I=1,MODES)
WRITE(6,11118) ((XNU(I,J),J=1,MODES),I=1,MODES)
WRITE(6,11119) ((BETA(I,J),J=1,MODES),I=1,MODES)
WRITE(6,11120) ((BETAB(I,J),J=1,MODES),I=1,MODES)
WRITE(6,11121) (Q(I),I=1,MODES)
WRITE(6,11122) (QB(I),I=1,MODES)
WRITE(6,11123) (OMEGAC(I),I=1,MODES)

```

11117 FORMAT(/75X,3HTAU/(25X,3E16.8))

11118 FORMAT(/75X,2HNU/(25X,3E16.8))

11119 FORMAT(/74X,4HBETA/(25X,3E16.8))

11120 FORMAT(/74X,8HBETA BAR/(25X,3E16.8))

11121 FORMAT(/51X,1HQ/(25X,3E16.8))

11122 FORMAT(/74X,5HQ BAR/(25X,3E16.8))

11123 FORMAT(/735X,28HNATURAL FREQUENCY CORRECTION/(25X,3E16.8))

C

C

C

C

BAFFLE EFFECT ON ENERGY

IF (NBAF.LT.1.OR.NBAF.GT.30) RETURN

WRITE (6,335)

335 FORMAT (1H0,6X,6HBAFFLE,6X,12HZ-COORDINATE,16X,13HR-COORDINATES,

214X,18HINST. DE/DT FACTOR,4X,17HAVG. DE/DT FACTOR/

321X,9HIN INCHES,19X,9HIN INCHES,28X,16HIN KG/M\*\*(.3/.4.))

DO 400 I=1,NBAF

ARG3=ZBAF(I)/D.0254

ARG4=RBAFI(I)/D.0254

ARG5=RBAFO(I)/D.0254

IF (ZBAF(I).LT.LEVEL.AND.ZBAF(I).GE.230T.AND.RBAFI(I).NE.RBAFO(I).  
 2AND.RBAFI(I).GE.0.0.AND.RBAFO(I).GE.0.0) GO TO 345

WRITE (6,340) I,ARG3,ARG4,ARG5

340 FORMAT (9X,12,3F19.4,21X,9HNO EFFECT)

GO TO 400

345 ARG1=RBAFI(I)

ARG2=RBAFO(I)

IF (ARG1.LT.ARG2) GO TO 350

ARG1=RBAFO(I)

ARG2=RBAFI(I)

350 SUM=0.0

DO 390 JJ=1,2

DO 390 J=1,NU,0

GO TO (355,360),JJ

355 II=NORD-J+1

T=0.5\*A(II)

S=0.5-0.5\*A(II)

GO TO 370

360 T=0.5\*A(J)

S=0.5+0.5\*A(J)

370 RR=(ARG2-ARG1)\*S+ARG1

TERM(1)=0.0

K1=MPOLY+1

DO 380 K=K1,MURDER

KK=N-MPOLY

ARG=UN(KK)\*RR/AS

CALL BESSEL (ARG,B1,B10)

380 TERM(1)=TERM(1)+CNK(K,1)\*UN(KK)\*B1\*EXP(-UN(KK))\*((LEVEL-ZBAF(I))/AS)

```
SUM=SUM+ABS(TERM(1))*((5./2.)*RR*T*(ARG2-ARG1))
390 CONTINUE
TERM(1)=-SUM*RHO*18.0*1.23*(KNSTAR(1)/MN(1))*((1./4.)*
2SQRT((ARG2-ARG1)/PI)*(L*BN(1)/AS))*((5./2.))
TERM(2)=TERM(1)*(KNSTAR(1)/MN(1))*((5./4.))*4.0/3.0/PI
WRITE (6,395) 1,ARG3,ARG4,ARG5,TERM(1),TERM(2)
395 FORMAT (9X,I2,3F19.4,2E21.5)
400 CONTINUE
```

C

C

```
RETURN
END
```

SLUSH

0-74

STBFTC INTG1 LIST,DECK

SLUSH

02/4

INTG1 - EFF. SOURCE STATEMENT - IFN(S) -

SUBROUTINE INTG1 (F,LEV1,LEV2)

C  
C  
C  
C  
C  
C  
C

ROUTINE EVALUATES A CLOSED LINE INTEGRAL WITH RESPECT TO THE  
Z-COORDINATE FOR THAT PORTION OF THE TANK BETWEEN TWO LEVELS  
(LEV1 AND LEV2), WHERE F IS THE SUBROUTINE CONTAINING  
THE INTEGRAND

COMMON NSEG,LEVEL,ZCG,L,D,H,AS,VOL,PI,LPSL,RHO,MODES,ZBOT,IOEGR,  
ZG1,G2,G3,ZDIV,DZ(2)  
COMMON /GAUSS/HORD,X(8),M(8)  
COMMON /PARAM/H,N,JNM,JND,RK,ZZ,FUNC,ANS  
COMMON /SEG/TYPE(50),REGRPT,KPOINT(50),BEGZPT,ZPOINT(50),RC(50),  
ZC(50),A(50),B(50),GAMMA(50),GAM(50)  
REAL LEV1,LEV2

C  
C

ANS=0.0  
DO 1999 I=1,NSG

C  
C  
C  
C

LOCATE THE PORTION OF THE SEGMENT BETWEEN THE TWO LEVELS

IF (ZPOINT(I-1).EQ.ZPOINT(I)) GO TO 1999  
IF (ZPOINT(I-1).LE.LEV2.AND.ZPOINT(I).LE.LEV2) GO TO 1999  
IF (ZPOINT(I-1).GE.LEV1.AND.ZPOINT(I).GE.LEV1) GO TO 1999  
IF (ZPOINT(I-1).LE.LEV2.AND.ZPOINT(I).GT.LEV2) GO TO 10  
IF (ZPOINT(I-1).LT.LEV1.AND.ZPOINT(I).GE.LEV1) GO TO 20  
IF (ZPOINT(I-1).LT.LEV1.AND.ZPOINT(I).LE.LEV2) GO TO 25  
IF (ZPOINT(I-1).LT.LEV1.AND.ZPOINT(I).LT.LEV1) GO TO 30  
BEG=LEV1  
IF (ZPOINT(I).LE.LEV2) GO TO 5  
END=ZPOINT(I)  
GO TO 1035  
5 END=LEV2  
GO TO 1035  
10 BEG=LEV2  
IF (ZPOINT(I).GE.LEV1) GO TO 15  
END=ZPOINT(I)  
GO TO 1035  
15 END=LEV1  
GO TO 1035  
20 BEG=ZPOINT(I-1)  
END=LEV1  
GO TO 1035  
25 BEG=ZPOINT(I-1)  
END=LEV2  
GO TO 1035  
30 BEG=ZPOINT(I-1)  
END=ZPOINT(I)

C  
C  
C  
C

EVALUATE THE INTEGRAL ALONG THE SEGMENT BETWEEN THE TWO LEVELS

1035 SUM=0.0

SLUSH

INTG1

- EFN

SOURCE STATEMENT

-

IFN(S)

-

05/

```

DO 1040 JJ=1,5
DO 1040 J=1,NORH
GO TO (1036,1037),JJ
1036 II=NORD-J+1
T=0.5*N(II)
IF (END.GT.BEG) GO TO 2012
S=0.5+0.5*X(II)
GO TO 1038
2012 S=0.5-0.5*X(II)
GO TO 1038
1037 T=0.5*N(J)
IF (END.GT.BEG) GO TO 2022
S=0.5-0.5*X(J)
GO TO 1038
2022 S=0.5+0.5*X(J)
1038 ZZ=(END-BEG)*S+BEG
CALL FINDR (1,ZZ,RR)
CALL F
1040 SUM=SUM+FUNC*T
ANS=ANS+SUM*(END-BEG)

```

C

C

1999 CONTINUE

C

C

RETURN

END

SLOSH

02/4

SIBFTC INTG2 LIST DECK

SUBROUTINE INTG2 (F,LEV1,LEV2)

ROUTINE EVALUATES A CLOSED LINE INTEGRAL WITH RESPECT TO THE  
R-COORDINATE FOR THAT PORTION OF THE TANK BETWEEN TWO LEVELS  
(LEV1 AND LEV2), WHERE F IS THE SUBROUTINE CONTAINING  
THE INTEGRAND

COMMON NSEG,LEVEL,ZCG,L,D,H,AS,VOL,PI,EPSL,RHO,MODES,7BOT,IDEGR,  
ZG1,G2,G3,ZDIV,DZ(2)  
COMMON /GAUSS/NORD,X(8),W(8)  
COMMON /PARAM/M,D,ONH,ONR,RE,ZZ,FUNC,ANS  
COMMON /SEG/TYPE(50),BEGRPT,RPOINT(50),BEGZPT,ZPOINT(50),RC(50),  
Z2C(50),A(50),B(50),GAMMA(50),GAM(50)  
REAL LEV1,LEV2

SET UP LOGIC FLAGS

IBEGR=0  
IF (RPOINT(NSEG).EQ.BEGRPT.AND.ZPOINT(NSEG).NE.BEGZPT.AND.  
2LEV2.GE.BEGZPT.AND.LEV2.GE.ZPOINT(1)) IBEGR=1  
IOUTI=0  
IFLAG=0

TEMP=0.0  
BEGR=0.0  
ENGR=0.0  
ANS=0.0  
DO 130 I=1,NSFG

LOCATE THE PORTION OF THE SEGMENT BETWEEN THE TWO LEVELS

1 ILEV=0  
IF (ZPOINT(I-1).LE.LEV2.AND.ZPOINT(I).LE.LEV2) GO TO 110  
IF (ZPOINT(I-1).GE.LEV1.AND.ZPOINT(I).GE.LEV1) GO TO 110  
IF (ZPOINT(I-1).LE.LEV2.AND.ZPOINT(I).GT.LEV2) GO TO 20  
IF (ZPOINT(I-1).LT.LEV1.AND.ZPOINT(I).GE.LEV1) GO TO 30  
IF (ZPOINT(I-1).LT.LEV1.AND.ZPOINT(I).LE.LEV2) GO TO 40  
IF (ZPOINT(I-1).LT.LEV1.AND.ZPOINT(I).LT.LEV1) GO TO 50  
IF (ZPOINT(I).LE.LEV2) GO TO 10  
IF (RPOINT(I).EQ.RPOINT(I-1)) GO TO 7  
CALL FINDR (I,LEV1,ENGR)  
IF (IOUTI.EQ.0) GO TO 6  
5 IOUTI=0  
BEG=OUTI  
END=ENGR  
ILEV=1  
GO TO 80  
6 BEG=ENGR  
END=RPOINT(I)  
GO TO 80  
7 ENDR=RPOINT(I)

SLOSH

INTG2

- EFN SOURCE STATEMENT - IFN(S) -

057

```

IF (IOUT1.EQ.0) GO TO 10
GO TO 5
10 IF (RPOINT(I).EQ.RPOINT(I-1)) GO TO 15
CALL FINDR (I,LEV1,ENOR)
IF (IOUT1.NE.0) GO TO 5
CALL FINDR (I,LEV2,BEGR)
IBEGR=1
BEG=ENOR
END=BEGR
GO TO 30
15 ENOR=RPOINT(I)
IF (IOUT1.NE.0) GO TO 5
BEGR=RPOINT(I)
IBEGR=1
GO TO 110
20 IF (ZPOINT(I).LT.LEV1) GO TO 25
IOUT1=1
IF (RPOINT(I).EQ.RPOINT(I-1)) GO TO 23
CALL FINDR (I,LEV1,IOUT1)
CALL FINDR (I,LEV2,IOUT2)
IF (IBEGR.EQ.0) GO TO 22
21 IBEGR=0
BEG=BEGR
END=IOUT2
ILEV=2
GO TO 30
22 IF (BEGR.EQ.0.0) TEMP=OUT2
BEG=OUT2
END=OUT1
GO TO 30
23 OUT1=RPOINT(I)
24 OUT2=RPOINT(I)
IF (IBEGR.NE.0) GO TO 21
IF (BEGR.EQ.0.0) TEMP=OUT2
GO TO 110
25 IF (RPOINT(I).EQ.RPOINT(I-1)) GO TO 24
CALL FINDR (I,LEV2,IOUT2)
IF (IBEGR.NE.0) GO TO 21
IF (BEGR.EQ.0.0) TEMP=OUT2
BEG=OUT2
END=RPOINT(I)
GO TO 30
30 IOUT1=1
IF (RPOINT(I).EQ.RPOINT(I-1)) GO TO 35
CALL FINDR (I,LEV1,IOUT1)
BEG=RPOINT(I-1)
END=OUT1
GO TO 30
35 OUT1=RPOINT(I)
GO TO 110
40 IBEGR=1
IF (RPOINT(I).EQ.RPOINT(I-1)) GO TO 45
CALL FINDR (I,LEV2,BEGR)
BEG=RPOINT(I-1)
END=BEGR
GO TO 30

```

```

45 BEGR=RPOINT(I)
   GO TO 110
50 IF (RPOINT(I).EQ.RPOINT(I-1)) GO TO 110
   BEG=RPOINT(I-1)
   END=RPOINT(I)

```

```

C
C
C     EVALUATE THE INTEGRAL ALONG THE SEGMENT BETWEEN THE TWO LEVELS
C     OR ALONG ONE OF THE TWO LEVELS
C

```

```

80 SUM=0.0
   DO 90 JJ=1,2
   DO 90 J=1,NORD
   GO TO (1001,1002),JJ
1001 II=NORD-J+1
   T=0.5*W(II)
   IF (END.GT.BEG) GO TO 1010
   S=0.5+0.5*X(II)
   GO TO 85
1010 S=0.5-0.5*X(II)
   GO TO 85
1002 I=0.5*W(J)
   IF (END.GT.BEG) GO TO 1020
   S=0.5-0.5*X(J)
   GO TO 85
1020 S=0.5+0.5*X(J)
85  RR=(END-BEG)*S+BEG
   IF (ILEV.EQ.0) CALL FINDZ (I,RR,ZZ)
   IF (ILEV.EQ.1) ZZ=LEV1
   IF (ILEV.EQ.2) ZZ=LEV2
   CALL F
   SUM=SUM+FUNC*T
90  CONTINUE
   ANS=ANS+SUM*(END-BEG)

```

```

C
C
C     CHECK IF INTEGRATION HAS BEEN COMPLETED ALONG THE SEGMENT
C

```

```

IF (ILEV.NE.0.AND.IFLAG.EQ.0) GO TO 1

```

```

C
C
C     CHECK IF INTEGRATION HAS BEEN PERFORMED ON THE TWO LEVELS
C

```

```

110 IF (I.LT.NSE6) GO TO 130
   IFLAG=1
   IF (IBEGR.EQ.0) GO TO 115
   BEGR=0
   BEG=BEGR
   END=IEND
   ILEV=2
   GO TO 80
115 IF (IOUT1.EQ.0) GO TO 130
   IOUT1=0
   BEG=OUT1
   END=ENDR
   ILEV=1

```

SLOSH

INTG2

- EFN SOURCE STATEMENT - IFN(S) -

05/2

GO TO 80

130 CONTINUE

RETURN  
END

SLUSH

0270

SIBFTC INT3 LIST DECK

SLUSH

INT3

- EFN SOURCE STATEMENT - IFN(S) -

02/4

```
SUBROUTINE IN+G2(F)
COMMON/AN/XG(8),WG(8)
COMMON /GAUSS/NORD,X(8),W(8)
COMMON /PARAM/M,N,JNM,JNN,RR,ZZ,FUNC,ANS
ANS=0.0
DO 10 I=1,2
DO 10 J=1,NORD
GO 10 (2,4),1
2 U=-XG(J)
GO TO 5
4 U=XG(J)
5 RR=U+U.5
CALL F
10 ANS=ANS+FUNC*WG(J)
RETURN
END
```

SLUSH

02/2

51BFTC FINDR LIST DECK

```

SUBROUTINE FINDR (I,ZZ,OR)
COMMON /SEG/TYPE(50),BEGZPT,RPOINT(50),BEGZPT,ZPOINT(50),RC(50),
ZZC(50),A(50),B(50),GAMMA(50),GAM(50)
INTEGER TYPE,ARG
IF (ZZ-ZPOINT(1)) 2,1,2
1 RR=RPOINT(1)
RETURN
2 IF (ZZ-ZPOINT(1-1)) 4,3,4
3 RR=RPOINT(1-1)
RETURN
4 R1=0.0
R2=0.0
ARG=TYPE(1)
GO TO (5,35,40),ARG
5 C1=ZZ-ZC(1)
C2=SIN(GAM(1))
C3=COS(GAM(1))
C4=(C2*B(1))*2+(C3*A(1))*2
C5=(C3*B(1))*2+(C2*A(1))*2
C6=2.*C2*C3*(B(1))*2-A(1))*2
C7=(C1*C6))*2-4.*C5*(C4*C1))*2-(A(1)*B(1))*2
IF (C7) 100,10,10
10 R1=RC(1)+(-C1*C6+SQRT(C7))/(2.*C5)
R2=RC(1)+(-C1*C6-SQRT(C7))/(2.*C5)
GO TO 70
35 RR=RPOINT(1-1)+(RPOINT(1)-RPOINT(1-1))/(ZPOINT(1)-ZPOINT(1-1))*(ZZ
Z-ZPOINT(1-1))
RETURN
40 C1=ZZ-ZC(1)
IF (GAMMA(1).EQ.90.0) GO TO 50
IF (GAMMA(1).NE.270.0) GO TO 55
RR=RC(1)+C1**2/(2.0*A(1))
RETURN
50 RR=RC(1)-C1**2/(2.0*A(1))
RETURN
55 C2=SIN(GAM(1))
C3=COS(GAM(1))
C4=2.0*C2*C3
C5=C1*C4+2.*A(1)*C2
C6=C5**2-4.*C3**2*((C,*/2))*2-2.*A(1)*C1*C3)
IF (C6) 100,60,60
60 R1=RC(1)+(-C5+SQRT(C6))/(2.*C3**2)
R2=RC(1)+(-C5-SQRT(C6))/(2.*C3**2)
GO TO 70
100 WRITE (6,101) I
101 FORMAT (1H0,15HSEGMENT NUMBER ,12,11H IS INVALID)
CALL EXIT
70 IF (R1.LT.0.0) R1=0.0
IF (R2.LT.0.0) R2=0.0
IF ((R1-RPOINT(1))/ABS(R1-RPOINT(1))-(R1-RPOINT(1-1))/ABS(R1-RPOINT
2(1-1))) 94,90,94
94 RR=R1
RETURN
96 IF ((R2-RPOINT(1))/ABS(R2-RPOINT(1))-(R2-RPOINT(1-1))/ABS(R2-RPOINT
2(1-1))) 98,100,98

```

SLOSH  
FINDR

- EFN SOURCE STATEMENT - IFN(S) -

05/2

98 RR=R2  
RETURN  
END

SLOSH

02/2

518FTC FINDZ LIST DECK

```

SUBROUTINE FINDZ (I,RR,Z)
COMMON /SEG/TYPE(SU),BEGRPT,RPOINT(SU),BEGZPT,ZPOINT(SU),RC(SU),
2ZC(SU),A(SU),B(SU),GAMMA(SU),GAM(SU)
INTEGER TYPE,ARG
IF (RR-RPOINT(1)) 2,1,2
1 ZZ=ZPOINT(1)
RETURN
2 IF (RR-RPOINT(I-1)) 4,3,4
3 ZZ=ZPOINT(I-1)
RETURN
4 Z1=0.0
Z2=0.0
ARG=TYPE(I)
GO TO (5,20,25),ARG
5 C1=RR-RC(1)
C2=SIN(GAM(1))
C3=COS(GAM(1))
C4=(C2*B(1))*2+(C3*A(1))*2
C5=(C3*B(1))*2+(C2*A(1))*2
C6=2.*C2*C3*(B(1)*2-A(1)*2)
C7=(C1*C6)*2-4.*C4*(C1*2+C5-(A(1)*B(1))*2)
IF (C7) 100,10,10
10 Z1=ZC(1)+(-C1*C6+SQRT(C7))/(2.*C4)
Z2=ZC(1)+(-C1*C6-SQRT(C7))/(2.*C4)
GO TO 50
20 ZZ=ZPOINT(I-1)+(ZPOINT(I)-ZPOINT(I-1))/(RPOINT(I)-RPOINT(I-1))*(RR
2-RPOINT(I-1))
RETURN
25 C1=RR-RC(1)
IF (GAMMA(1).EQ.0.0,0.5,GAMMA(1).EQ.360.0) GO TO 35
IF (GAMMA(1).NE.180.0) GO TO 40
ZZ=ZC(1)-C1*2/(2.0*A(1))
RETURN
35 ZZ=ZC(1)+C1*2/(2.*A(1))
RETURN
40 C2=SIN(GAM(1))
C3=COS(GAM(1))
C4=2.0*C2*C3
C5=C1*C4-2.*A(1)*C3
C6=C5*2-4.*C2*2*((C.*C3)*2+2.*A(1)*C1*C2)
IF (C6) 100,45,45
45 Z1=ZC(1)+(-C5+SQRT(C6))/(2.*C2*2)
Z2=ZC(1)+(-C5-SQRT(C6))/(2.*C2*2)
GO TO 50
100 WRITE (6,101) I
101 FORMAT (1H,15HSEGMENT NUMBER ,I2,1H IS INVALID)
CALL EX11
50 IF ((Z1-ZPOINT(I))/ABS(Z1-ZPOINT(I))-(Z1-ZPOINT(I-1))/ABS(Z1-ZPOINT
2(I-1))) 55,60,55
55 ZZ=Z1
RETURN
60 IF ((Z2-ZPOINT(I))/ABS(Z2-ZPOINT(I))-(Z2-ZPOINT(I-1))/ABS(Z2-ZPOINT
2(I-1))) 65,100,65
65 ZZ=Z2
RETURN

```

SLOSH  
FINUZ

- EFN SOURCE STATEMENT - IFN(S) -

05/2

END

SLOSH

02/7

SIBFTC F1

LIST DECK

SLOSH  
F1

- EFN SOURCE STATEMENT - IFN(S) -

02/2

```
SUBROUTINE F1
COMMON NSEG,LEVEL,ZCG,L,D,H,AS,VOL,PI,EPSL,RHO,MODES,7BOT,IOEGR,
ZG1,G2,G3,ZDI7,DZ(2)
COMMON /PARAH/M,N,UNM,UNH,RR,ZZ,FUNC,ANS
FUNC=RR**2*(ZZ-ZCG)
RETURN
END
```

SLOSH

02/

SIBFTC F2

LIST DECK

SLOSH  
F2

- EFN SOURCE STATEMENT - IFN(S) -

02/2

```
SUBROUTINE F2  
COMMON /PARAM/M,N,JNM,JAN,RR,ZZ,FUNC,ANS  
FUNC=RR**2  
RETURN  
END
```

SLOSH

02/2

SIBFTC F4

LIST.DECK

SLOSH

F4

- EFM SOURCE STATEMENT - IFN(S) -

02/2

```
SUBROUTINE F4
COMMON NSEG,LEVEL,ZCG,L,D,H,AS,VOL,P1,EPSL,RHO,MODES,7BOT,1DEGR,
ZG1,G2,G3,ZDIV,DZ(2)
COMMON /PARAM/M,N,JNH,JNN,RR,ZZ,FUNC,ANS
FUNC=RR**3*(ZZ-ZCG)+2./3.*RR*(ZZ-ZCG)**3
RETURN
END
```

SLOSH

05/4

SIBFTC F5

LIST DECK

SLOSH  
F5

- EFN SOURCE STATEMENT - IFN(S) -

02/2

```
SUBROUTINE F5  
COMMON NSEG,LEVEL,ZCG,L,D,H,AS,VOL,PI,EPSL,RHO,MODES,7BOT,IDEGR,  
2G1,G2,G3,ZDIV,DZ(2)  
COMMON /PARAM/M,N,JNM,JNN,RR,ZZ,FUNC,ANS  
FUNC=RR*(ZZ-ZCG)**3  
RETURN  
END
```

SLOSH

02/2

318FTC F6

LIST DECK

SLUSH  
F6

- EFN SOURCE STATEMENT - IFN(S) -

02/2

```
SUBROUTINE F6  
COMMON NSEG,LEVEL,ZCG,L,D,H,AS,VOL,PI,EPSL,RHO,MODES,780T,IOEGR,  
ZG1,G2,G3,ZDIV,DZ(2)  
COMMON /PARAM/N,N,JNM,JNN,RR,ZZ,FUNC,ANS  
FUNC=(RR/AS)**(2*(N+N)-3)*(ZZ-ZCG)/AS**2  
RETURN  
END
```

SLUSH

02/2

SIBFTC F7

LIST DECK

SLUSH

F7

- EFN SOURCE STATEMENT - IFN(S) -

02/4

```
SUBROUTINE F7
COMMON NSEG,L,VEL,ZCG,L,D,H,AS,VOL,P1,EPSL,RHO,MODES,7BOT,1DEGR,
ZG1,G2,G3,ZDIV,DZ(2)
COMMON /PARAM/H,H,JNN,JNN,RR,ZZ,FUNC,ANS
REAL JNN
ARG=JNN*RR/AS
CALL BESSEL (ARG,B1,B1D)
FUNC=(FLOAT(2*M-1)*(RR/AS))*((2*M-1)*JNN*B1D+(RR/AS))*((2*M-2)*B1)
2EXP(JNN*(ZZ-ZCG)/AS)/AS
RETURN
END
```

SLOSH

02/4

SIBFTC F3

LIST.DECK

SLOSH  
F8

- EF., SOURCE STATEMENT - IFN(S) -

02/2

```
SUBROUTINE F8  
COMMON NSEG,LEVEL,ZCG,L,D,H,AS,VOL,PI,EPSL,RHO,MODES,7BOT,IOEGR,  
2G1,G2,G3,ZDIV,DZ(2)  
COMMON /PARAM/M,N,JNM,JNN,RR,ZZ,FUNC,ANS  
REAL JNM,JNN  
RNUN=RR/AS  
ARG=JNM*RNUN  
CALL BESSEL (ARG,BIM,BINM)  
ARG=JNN*RNUN  
CALL BESSEL (ARG,BIN,BIDN)  
FUNC=((ARG*JNM*BIDM*BIDN+BIM*BIN/RNUN+ARG*JNM*BIM*BIN)/AS)*  
2EXP((JNM+JNN)*(ZZ-ZCG)/AS)  
RETURN  
END
```

SLOSH

02/2

SIBFTC F9 LIST,DECK

SLOSH

F9

- EFN SOURCE STATEMENT - IFN(S) -

02/2

SUBROUTINE F9

COMMON NSEG,LEVEL,ZCG,L,D,H,AS,VOL,PI,EPSL,RHO,MODES,7BOT,IDEGR,  
2G1,G2,G3,ZDIV,DZ(2)

COMMON /ELEM/NORDER,MPOLY,JN(5),NORDER,NPOLY,ACOE(10,10)

COMMON /PARAM/M,N,JNM,JNN,RR,ZZ,FUNC,ANS

REAL JN,JNM,L

I=M-MPOLY

IF (I) 5,5,10

5 FUNC=((ZZ-ZCG)\*RR\*\* (2\*M))/AS\*\* (2\*M+2)

RETURN

10 JNM=JN(I)

ARG=JNM\*RR/AS

CALL BESSEL (ARG,B1,B,D)

FUNC=(ZZ-ZCG)\*RR\*B1\*EXP(JNM\*(ZZ-ZCG-L)/AS)/AS\*\*3

RETURN

END

SLOSH

02/2

518FTC F12

LIST DECK

SL05H

F12

- EFN SOURCE STATEMENT - IFN(S) -

02/2

```
SUBROUTINE F12
COMMON NSEG,LEVEL,ZCG,L,D,H,AS,VOL,PI,EPSL,RHO,MODES,7BOT,IDEGR,
ZG1,G2,G3,ZDIV,DZ(2)
COMMON /ELEM/NORDER,MPOL,Y,JN(S),NORDER,NPOLY,ACOE(10,10)
COMMON /PARAM/M,N,JNM,JNH,RR,ZZ,FUNC,ANS
IF (M.GT.NPOLY) GO TO 10
FUNC=0.0
IF (ZZ.EQ.ZCG) RETURN
DO 5 I=1,M
5 FUNC=FUNC+ACOE(M,I)*(RR/AS)**(1+2)*((ZZ-ZCG)/AS)**(M-1)/AS
RETURN
10 ARG=FLOAT(M-NPOLY)*PI*RR/H
CALL PHODBS (ARG,PBS,PBSP)
FUNC=PBS*CO$ (FLOAT(M-NPOLY)*PI/H*(ZZ-ZCG+D))*RR**2/AS**3
RETURN
END
```

SLUSH

02/2

518FTC F14

LIST DECK

SLUSH

F14

- EFN SOURCE STATEMENT - IFN(S) -

02/2

```
SUBROUTINE F14
COMMON NSEG,LEVEL,ZCG,L,D,H,AS,VOL,PI,EPSL,RHO,MODES,7BOT,IDEGR,
2G1,G2,G3,ZDIV,DZ(2)
COMMON /PARAM/M,N,UNN,UNN,RR,ZZ,FUNC,ANS
FUNC=RR**3*(ZZ-ZCG)
RETURN
END
```

SLOSH

02/2

SIBFTC FIS

LIST DECK

```

SUBROUTINE F15
COMMON /NSEG,LEVEL,ZCG,L,D,H,AS,VOL,PI,EPSL,RHO,MODES,BOT,IDEGR,
2G1,G2,G3,ZDIV,DZ(2)
COMMON /ELEM/NORDER,NPOLY,JN(S),NORDER,NPOLY,ACOE(10,10)
COMMON /PARAM/N,N,JN1,JN2,RR,ZZ,FUNC,ANS
C=FLOAT(N-NPOLY)*PI/H
ARG=C*RR
CALL PMODBS (ARG,PBSN,PASPN)
IF (M.EQ.N) GO TO 5
ZARG=(ZZ-ZCG)/AS
DD=FLOAT(M-NPOLY)*PI/H
ARG=DD*RR
CALL PMODBS (ARG,PBSN,PASPM)
A=(PBSM*PBSN*RR*C*DD+PBSM*PBSN/RR)*AS
B=C*DD*RR*AS*PBSM*PBSN
P=SIN(C*D)
Q=SIN(DD*D)
R=COS(C*D)
S=COS(DD*D)
E=A*P+Q+B*R+S
F=A*R+S+B*Q*P
G=B*R*Q-A*P*S
HH=B*P*S-A*R*Q
P=(C+DD)*AS
Q=(C-DD)*AS
R=P*ZARG
S=Q*ZARG
FUNC=((E-F)/P*SIN(R)-(E+F)/Q*SIN(S)+(G+HH)/P*COS(R)+(G-HH)/Q*
2COS(S))/AS/2.0
RETURN
5 A=(RR*PBSN**2*C**2+PBSN**2/RR)*AS
B=RR*PBSN**2*C**2*AS
FUNC=(-A*(ZZ-ZCG)/AS+(A-B)/(C*AS)*(C*(ZZ-ZCG+B)/2.0-B-0.25*SIN(2.0*B.C
2*(ZZ-ZCG+D)))/AS
RETURN
END

```

SLOSH

02/2

SIBFTC F16

LIST,DECK

```
SUBROUTINE F1A
COMMON NSEG,LEVEL,ZCG,L,D,H,AS,VOL,PI,EPSL,RHO,MODES,7BOT,IDEGR,
ZG1,G2,G3,ZDIV,DZ(2)
COMMON /ELEM/MORDER,NPOLY,JN(5),NORDER,NPOLY,ACDEF(10,10)
COMMON /PARAM/M,N,JNM,JNN,RR,ZZ,FUNC,ANS
ZARG=(ZZ-ZCG)/AS
FUNC=0.0
DO 5 K=1,N
DO 5 I=1,M
5 FUNC=FUNC+FLOAT(K*I+1)/FLOAT(M+N-I-K+1)*ACDEF(N,K)*ACDEF(M,I)*
2(RR/AS)**(K+I-1)*ZARG**((M+N-I-K+1)/AS)
DO 10 K=1,N
DO 10 I=1,N
IF (N.EQ.K.OR.M.EQ.I) GO TO 10
FUNC=FUNC+FLOAT((N-K)*(M-I))/FLOAT(N+M-K-I-1)*ACDEF(N,K)*
2ACDEF(M,I)*(RR/AS)**(K+I+1)*ZARG**((N+M-K-I-1)/AS)
10 CONTINUE
RETURN
END
```

SLOSH

02/4

SIBFTC F17

LIST DECK

```

SUBROUTINE F17
COMMON NSEG,LEVEL,ZCG,L,D,H,AS,VOL,PI,EPSL,RHO,MODES,7BOT,1DEGR,
2G1,G2,G3,ZDIV,DZ(2)
COMMON /ELEM/MURDER,MPOI,Y,JN(5),NORDER,NPOLY,ACDEF(16,10)
COMMON /PARAM/M,N,JNM,JNR,RR,ZZ,FUNC,ANS
FUNC=0.0
ZARG=(ZZ-ZCG)/AS
IF (ZARG.EQ.0.0) RETURN
STOR=FLOAT(N-NPOLY)*PI*AS/H
ARG=FLOAT(N-NPOLY)*PI*RR/H
CALL PMODBS (ARG,PBS,PBSP)
ARG=FLOAT(N-NPOLY)*PI/H*(ZZ-ZCG+0)
DO 50 I=1,M
TEMP=0.0
NM=N-1
IF (NM.EQ.0) GO TO 35
GO TO (30,25,20,15,10,5),NM
5 TEMP=TEMP-FLOAT((NM)*(NM-1)*(NM-2)*(NM-3)*(NM-4)*(NM-5))/STOR**7*
2ZARG**((NM-6)*SIN(ARG)
10 TEMP=TEMP+FLOAT((NM)*(NM-1)*(NM-2)*(NM-3)*(NM-4))/STOR**6*ZARG**
2(NM-5)*COS(ARG)
15 TEMP=TEMP+FLOAT((NM)*(NM-1)*(NM-2)*(NM-3))/STOR**5*ZARG**((NM-4)*
2SIN(ARG)
20 TEMP=TEMP-FLOAT((NM)*(NM-1)*(NM-2))/STOR**4*ZARG**((NM-3)*COS(ARG)
25 TEMP=TEMP-FLOAT((NM)*(NM-1))/STOR**3*ZARG**((NM-2)*SIN(ARG)
30 TEMP=TEMP+FLOAT(NM)/STOR**2*ZARG**((NM-1)*COS(ARG)
35 TEMP=TEMP+ZARG**NM/STOR*SIN(ARG)
50 FUNC=FUNC+(STOR*PBSP*FLOAT(1)*(RR/AS)**I+PBS*(RR/AS)**(I-1))*
2ACDEF(M,I)*TEMP/AS
DO 100 I=1,M
IF (M.EQ.1) RETURN
TEMP=0.0
NM=N-1-I-1
IF (NM.EQ.0) GO TO 80
GO TO (75,70,65,60,55),NM
55 TEMP=TEMP+FLOAT((NM)*(NM-1)*(NM-2)*(NM-3)*(NM-4))/STOR**6*ZARG**
2(NM-5)*SIN(ARG)
60 TEMP=TEMP-FLOAT((NM)*(NM-1)*(NM-2)*(NM-3))/STOR**5*ZARG**((NM-4)*
2COS(ARG)
65 TEMP=TEMP-FLOAT((NM)*(NM-1)*(NM-2))/STOR**4*ZARG**((NM-3)*SIN(ARG)
70 TEMP=TEMP+FLOAT((NM)*(NM-1))/STOR**3*ZARG**((NM-2)*COS(ARG)
75 TEMP=TEMP+FLOAT(NM)/STOR**2*ZARG**((NM-1)*SIN(ARG)
80 TEMP=TEMP-ZARG**NM/STOR*COS(ARG)
100 FUNC=FUNC-STOR*PBS*ACDEF(M,I)*FLOAT(M-1)*(RR/AS)**(I+1)*TEMP/AS
RETURN
END

```

SLOSH

05/2

SIBFTC F16 LIST,DECK

BEGIN COMPILING 224018

SLUSH

F18

- EFN SOURCE STATEMENT - IFN(5) -

02/4

```

SUBROUTINE F18
COMMON NSEN,LEVEL,ZCG,L,D,H,AS,VOL,PI,EPSL,RHO,MODES,7BOT,IDEGR,
2G1,G2,G3,ZDIV,DZ(2)
COMMON /ELEM/MORDER,NPOLY,UN(5),MORDER,NPOLY,ACOE(10,10)
COMMON /PARAM/M,R,JNM,JNN,RR,ZZ,FUNC,ANS
IF (H.GT.NPOLY) GO TO 10
FUNC=0.0
DO 5 I=1,M
5 FUNC=FUNC+ACOE(M,I)*(RR/AS)**(I+1)*((ZZ-ZCG)/AS)**(M-I+1)/AS
RETURN
10 ARG=FLOAT(M-NPOLY)*PI*RR/H
CALL PHODBS (ARG,PBS,pbSP)
FUNC=RR/AS*(ZZ-ZCG)/AS*PBS*COS(FLOAT(M-NPOLY)*PI/H*(ZZ-ZCG+D))/AS
RETURN
END

```

SIABTC F280

LIST, DECK

SUBROUTINE F28

COMMON NSEG,LEVEL,ZCG,L,D,H,AS,VOL,PI,EPSL,RHO,MODES,ZBOT,IDEGR,

ZG1,G2,G3,ZDIV,DZ(2)

COMMON /PARAM/M,N,JNM,JNN,RR,ZZ,FUNC,ANS

COMMON/BES/B1I,B1D1,B1J,B1DJ,X1I,XJ1,I,J,J1,I1,JJ

COMMON /ELEM/NORDER,MPOLY,JN(S),NORDER,NPOLY,ACUEF(10,10)

REAL JN,JNM,JNN

ARG=JNM\*RR

CALL BESSEL(ARG,B1I,B1D1)

ARG=JNM\*RR

CALL BESSEL(ARG,B1J,B1DJ)

FUNC=(JN(I1)\*JN(JJ)\*B1D1\*B1DJ\*RR +B1I\*B1J /RR)

RETURN

END

SIBFTC F290

LIST, DECK

SUBROUTINE F29

COMMON NSEG,LEVEL,ZCG,L,D,H,AS,VOL,PI,EPSL,RHO,MODES,ZBOT,IDEGR,  
ZG1,G2,G3,ZDIV,UZ(2)

COMMON /ELEM/MORDER,NPOLY,JN(S),MORDER,NPOLY,ACOE(10,10)

COMMON /PARAM/M,N,JNM,JNI,RR,ZZ,FUNC,ANS

COMMON/BES/B1I,B1DI,B1J,B1DJ,X1I,XJ1,I,J,J1,I1,JJ

REAL JN,JNM

ARG=JNM\*RR

CALL BESSEL(ARG,B1J,B1DJ)

IF(RR)1,2,1

1 FUNC=(X1I\*ARG\*B1DJ+B1J)\*(RR )\*\*(2\*1-2)

GO TO 3

2 FUNC=0.0

3 CONTINUE

RETURN

END

SLOSH

02/2

SIBFIC JACOBI LIST DECK

```

CJACOBI      BUFFERING ROUTINE BETWEEN FORTRAN AND EGZF (SHARE)
SUBROUTINE JACOBI (MAX,NACT,NOGO,A,B,VECTOR,VALUE,MODES)

C
C      CALL JACOBI (MAX,NACT,NOGO,A,B,VECTOR,VALUE,MODES)
C      WHERE
C      MAX      DIMENSIONED ORDER OF A AND B MATRICES (LIMIT IS 40)
C      NACT     WORKING ORDER OF A AND B MATRICES
C      NOGO     =0 METHOD SUCCESSFUL
C      NOGO     =1 MATRIX B IS NOT POSITIVE DEFINITE
C      NOGO     =-1 ILL CONDITIONED MATRIX F (SEE SHARE WRITEUP-NYEUV)
C      A       MATRIX A OF  $AX=(\lambda B)A$ 
C      B       MATRIX B OF  $AX=(\lambda B)A$ 
C      VECTOR  EIGENVECTOR STORAGE MATRIX - VECTORS BY COLUMNS
C      VALUE   EIGENVALUE STORAGE VECTOR
C      MODES   NUMBER OF EIGENVALUES DESIRED

C      THIS ROUTINE SOLVES THE EIGENVALUE PROBLEM
C       $AX=(\lambda B)A$ 
C      WHERE A AND B MUST BE SYMMETRIC AND B MUST BE POSITIVE DEFINITE

C      THE ORIGINAL A AND B MATRICES ARE DESTROYED BY THE
C      EXECUTION OF SUBROUTINE EGZF

C      A RESTRICTION IN EGZF LIMITS DIMENSION OF A AND B
C      MATRICES TO 40x40 MAXIMUM. TO CHANGE THIS RESTRICTION
C      CHANGE LOCATION COMMON OF EGZF (NOW SET AT BSS 54) (N+14)

C      SUBROUTINES INCLUDED IN THIS PACKAGE ARE
C      JACOBI
C      RMEGZF
C      DIMENSION A(MAX,1),B(MAX,1),VECTOR(MAX,1),VALUE(1)
C      N=NACT
C      I1=0
C      READY A MATRIX FOR EGZF
C      DO 10 I2=1,N
C      DO 10 I3=1,N
C      I1=I1+1
C      10 A(I1,I)=A(I3,I2)
C      READY B MATRIX FOR EGZF
C      I4=0
C      DO 20 I5=1,N
C      DO 20 I6=1,I5
C      I4=I4+1
C      20 B(I4,I)=B(I6,I5)
C      RMEGZF-CALLING SEQUENCE TO EGZF
C      CALL RMEGZF (A,B,VECTOR,N,NOGO)
C      IF(NOGO) 100,25,100
C      EGZF STORAGE TO FORTRAN STORAGE
C      25 K1=0
C      DO 27 I7=1,N
C      DO 26 I8=1,N
C      K1=K1+1
C      26 B(I8,I7)=VECTOR(K1,I)
C      K2=(I7-1)*N+I7
C      27 A(I7,I)=A(K2,I)

```

```

C          ARRANGE EIGENVALUES IN DECREASING ORDER
C          REARRANGE EIGENVECTORS TO CORRESPOND TO EIGENVALUES
C          NORMALIZE EIGENVECTORS W.R.T. THEIR LARGEST ELEMENT
DO 90 J1=1,MODES
  VALUE(J1)=0.
DO 40 J2=J1,N
  IF(VALUE(J1)-A(J2,1)) 30,40,40
30  VALUE(J1)=A(J2,1)
   L1=J2
40  CONTINUE
   A(L1,1)=A(J1,1)
DO 50 J3=1,N
  VECTOR(J3,J1)=B(J3,L1)
50  B(J3,L1)=B(J3,J1)
   FNORM=0.
DO 70 J4=1,N
  IF(ABS(FNORM)-ABS(VECTOR(J4,J1))) 60,70,70
60  FNORM=VECTOR(J4,J1)
70  CONTINUE
DO 80 J5=1,N
  VECTOR(J5,J1)=VECTOR(J5,J1)/FNORM
  VALP=1./VALUE(J1)
  WRITE (6,85) VALP,FNORM,(VECTOR(J5,J1),J5=1,N)
85  FORMAT (1H0,4X,13HEIGENVALUE = ,E12.5,10X,33HEIGENVECTOR NORMALIZI
2NG FACTOR = ,E12.5,10X,11HEIGENVECTOR// (E106.5))
90  CONTINUE
100 RETURN
END

```

SLGSH

05/2

516FTC BLK

LIST,DEK

SLUSH

BLK

- EF. SOURCE STATEMENT - IFN(S) -

02/4

BLOCK DATA

COMMON /ELEM/MORDER,MPOL,Y,UN(S),MORDER,NPOLY,ACOEFF(10,10)

COMMON /GAUSS/NORD,X(8),W(8)

REAL UN

DATA MPOLY/5/

DATA MORDER,NPOLY/9,6/

DATA UN/1.8411837,5.3314427,8.5303163,11.706004,14.862588/

DATA NORD,X,W/8,0.095012509,0.26160355,0.45801677,0.61787624,  
20.7554044,0.8656312,0.94457502,0.98940093,0.18945061,0.18260371,  
30.16915651,0.14959598,0.12462897,0.095158511,0.062253423,  
40.027152459/

END

SLUSH

02/4

SIBFTC BESSEL LIST.DECK

```

SUBROUTINE BESSEL(ARG,B,,DBDX)
  DIMENSION A(7),B(7),C(7),D(7),E(7),F(7)
  DOUBLE PRECISION DPARG,DPSUM,DPTH,DPTHF,DP1,DPSUM1
  DATA (A(I),I=1,7)/.5,-.56249965,.21093573,-.03954729,.0044331
  *9,-.00031761,.00001109 /
  DATA (B(I),I=1,7)/.79788456,.00000156,.01659667,.00017105,-.00
  *249511,.00113655,-.00020033 /
  DATA (C(I),I=1,7)/-2.35619449,.12499612,.00005650,-.00637879,.
  *00074346,.00079624,-.00029166 /
  DATA (D(I),I=1,7)/1.0,-2.2499997,.12656208,-.3163846,.0444479
  *,-.0039444,.0002100 /
  DATA (E(I),I=1,7)/.79788456,-.00000077,-.00552740,-.00009512,
  *00137237,-.00072005,.00014476 /
  DATA (F(I),I=1,7)/-.78539816,-.04166397,-.00003954,.00262573,
  *-.00054125,-.00029333,.00013558 /
  DPTH=0.0
  DPSUM=0.0
  IF (ARG)9990,1000,100
100 IF (ARG-3.) 200,200,500
200 DPARG=ARG/J.0
  DO 300 I=1,7
    DPTH=DPTH+D(I)*DPARG*(2*I-2)
300 DPSUM=DPSUM+A(I)*DPARG**((2*I-2)
    B1= ARG*DPSUM
    DBDX=DPTH-B1/ARG
    GO TO 10000
500 DPARG=3.0/ARG
    DPSUM1=0.0
    DPTHF=0.0
    DO 600 I=1,7
      DP1=DPARG**((I-1)
      DPTHF=DPTHF+F(I)*DP1
      DPSUM1=DPSUM1+E(I)*DP1
      DPTH=DPTH+C(I)*DP1
600 DPSUM=DPSUM+B(I)*DP1
    DUM=SQRT(1.0/ARG)
    DPTH=ARG+DPTH
    DPTHF=DPTHF+ARG
    B1=DUM *DPSUM*ACOS(DPTH)
    DBDX=DUM*DPSUM1*UCOS(DPTHF)-B1/ARG
    GO TO 10000
1000 B1=0.0
    DBDX=.5
    GO TO 10000
9990 WRITE (6,9991) ARG
9991 FORMAT (1H0,4X,26HBESSEL FUNCTION ARGUMENT OF ,E12.4,
21H IS INVALID)
  CALL EXIT
10000 RETURN
  END

```

## SIBFTC PHODBS LIST DECK

```

SUBROUTINE PMODBS( X,BMI,BMID )
C COMPUTES MODIFIED BESSEL FCN 1 AND ITS DERIVATIVE BY POLY. APPROX.
DIMENSION A(6),B(6),C(9),D(9)
DOUBLE PRECISION T,S,Z,BMI
DATA A/.00032411,.00301532,.02658733,.15084934,.51498869,.8789059
*4 /,B /.0045813,.0360768,.2659732,1.2067492,3.0899424,3.5156227 /
DATA C/-.00420059,.01787654,-.02895312,.02282967,-.01031555,
*.00163801,-.00362018,-.03988024, .37894228 /, D/ .00392377,-.
*01647633,.02635537,-.02057706,.00916281,-.00157565,.00225319,
*.01328592, .39894228 /
IF ( X .EQ. 0. ) GO TO 600
100 T = X/3.75
120 S = 1+T
150 IF ( X .GT. 3.75 ) GO TO 200
160 BMI =X*(((A(1) )*S+A(2))*S+A(3))*S+A(4))*S+A(5))*S+A(6))*S+.5
*)
BMD= (((B(1) )*S+B(2))*S+B(3))*S+B(4))*S+B(5))*S+B(6))*S+1.
GO TO 400
200 Z=X
BMI=DEXP (Z)/DSQRT (Z) * ( (((C(1)/T +C(2))/T+C(3))/T+C(4))
*/T+C(5))/T+C(6))/T+C(7))/T+C(8))/T+C(9) )
BMD=DEXP (Z)/DSQRT (Z) * ( (((D(1)/T +D(2))/T+D(3))/T+D(4))
*/T+D(5))/T+D(6))/T+D(7))/T+D(8))/T+D(9) )
400 BMID = BMD-BMI/X
500 RETURN
600 BMI = 0.
BMID = .5
GO TO 500
END

```

SLUSH

0-72

SIBFTC SOLVE LIST DECK

SOLVE - EFN SOURCE STATEMENT - IFN(S) -

```

CSOLVE  LINEAR EQUATION SOLVER WITH ITERATIVE IMPROVEMENT  VERSION IV  SLV
SUBROUTINE SOLVE(NN,A,B,IN,EPS,ITMAX,X,IT)  SLV
C SOLVES AX=B WHERE A IS NAN MATRIX AND B IS NAN VECTOR  SLV
C IN=  SLV
C 1 FOR FIRST ENTRY  SLV
C 2 FOR SUBSEQUENT ENTRIES WITH NEW B  SLV
C 3 TO RESTORE A AND B  SLV
C EPS AND ITMAX ARE PARAMETERS IN THE ITERATION  SLV
C IT=  SLV
C -1 IF A IS SINGULAR  SLV
C 0 IF NOT CONVERGENT  SLV
C NUMBER OF ITERATIONS IF CONVERGENT  SLV
C CALLS MAP SUBROUTINES ILOG2,DOT,SDOT AND DAD  SLV
C  SLV
C TO MODIFY DIMENSIONS, CHANGE THE NEXT 3 (NOT 2 BUT 3) CARDS.  SLV
C DIMENSION A(10,10),B(10),X(10),AA(10,10),DX(10),R(10),  SLV
C 2 Z(10),RM(10),IKP(10)
MA=10
C MA MUST = DECLARED DIMENSION OF SYSTEM  SLV
EQUIVALENCE(R,DX)  SLV
GO TO (1000,2000,3000),IN  SLV
1000 N=NN  SLV
NM1=N-1  SLV
NPI=N+1  SLV
C  SLV
C EQUILIBRATION  SLV
C  SLV
DO 510 I=1,N  SLV
KTOP=ILOG2(A(I,1))  SLV
DO 503 J=2,N  SLV
503 KTOP=MAXD(KTOP,ILOG2(A(I,J)))  SLV
RM(I)=2.0**(-KTOP)  SLV
DO 509 J=1,N  SLV
509 A(I,J)=A(I,J)*RM(I)  SLV
510 CONTINUE  SLV
C  SLV
C SAVE EQUILIBRATED DATA  SLV
C  SLV
DO 546 I=1,N  SLV
DO 548 J=1,N  SLV
548 AA(I,J)=A(I,J)  SLV
C  SLV
C GAUSSIAN ELIMINATION WITH PARTIAL PIVOTING  SLV
C  SLV
DO 99 M=1,NM1  SLV
TOP=ABS(A(M,M))  SLV
IMAX=M  SLV
DO 12 I=M,N  SLV
IF(TOP=ABS(A(I,M)))10,12,12  SLV
10 TOP=ABS(A(I,M))  SLV
IMAX=I  SLV
12 CONTINUE  SLV
IF(TOP)14,13,14  SLV
13 IT=-1  SLV
C *SINGULAR*  SLV

```

SLOSH

02/2

SOLVE

- EFL. SOURCE STATEMENT - IFN(5) -

```

      RETURN
14      IRP(N)=IMAX
23      IF (IMAX-N) 20,29,24
24      DO 25 J=1,N
          TEMP=A(N,J)
          A(N,J)=A(IMAX,J)
          A(IMAX,J)=TEMP
25      MP1=N+1
29      DO 33 I=MP1,N
          EM=A(I,M)/A(M,M)
          A(I,M)=EM
          IF (EM) 31,33,31
31      DO 32 J=MP1,N
32      A(I,J)=A(I,J)-A(N,J)*EM
33      CONTINUE
99      CONTINUE
      IRP(N)=N
      IF (A(N,N)) 120,113,120
113     IT=-1
      RETURN
120     CONTINUE
C      STORAGE FOR A NOW CONTAINS TRIANGULAR L AND U SO THAT (L+I)*U=A
C
C      DUPLICATE INTERCHANGES IN DATA
C
      DO 229 I=1,N
          IP=IRP(I)
          IF (I-IP) 221,229,221
221      DO 222 J=1,N
          TEMP=AA(I,J)
          AA(I,J)=AA(IP,J)
          AA(IP,J)=TEMP
222
229     CONTINUE
C
C      PROCESS RIGHT HAND SIDE
C
2000    CONTINUE
      DO 601 I=1,N
601      B(I)=B(I)-KM(I)
      DO 609 I=1,N+1
          IP=IRP(I)
          TEMP=B(I)
          B(I)=B(IP)
          B(IP)=TEMP
609     CONTINUE
C
C      SOLVE FOR FIRST APPROXIMATION TO X
C
199     DO 200 I=1,N
200     Z(I)=-SDOT(I-1,A(I,1),MA,Z(I),1,-B(I))
          DO 201 K=1,N
              I=MP1-K
201     X(I)=-SDOT(N-I,A(I,I+1),MA,X(I+1),1,-Z(I))/A(I,I)
C
C      ITERATIVE IMPROVEMENT
C

```

SLOSH

02/2

SOLVE - EFN SOURCE STATEMENT - IFN(S) -

```

      IF(ITMAX)370,370,300
300  TOP=0.0
      DO 303 I=1,N
303  TOP=AMAX1(TOP,ABS(X(I)))
      EPSX=EPS*TOP
      DO 369 IT=1,ITMAX
C      FIND RESIDUALS
      DO 319 I=1,N
319  R(I)=-DOT(N,AA(I,1),MA,X(I),1,-B(I))
C      FIND INCREMENT
      DO 329 I=1,N
329  Z(I)=-SDOT(I-1,A(I,1),MA,Z(I),1,-R(I))
      DO 339 K=1,N
      I=NP1-K
339  DX(I)=-SDOT(N-I,A(I,I+1),MA,DX(I+1),1,-Z(I))/A(I,I)
C      INCREMENT AND TEST CONVERGENCE
      TOP=0.0
      DO 342 I=1,N
      TEMP=A(I)
      X(I)=DAD(X(I),DX(I))
      DELX=ABS(X(I)-TEMP)
      TOP=AMAX1(TOP,DELX)
342  CONTINUE
      IF(TOP-EPSX)381,381,369
369  CONTINUE
370  IT=0
381  RETURN
C
C      RESTORE A AND B
C
3000 CONTINUE
      DO 709 K=1,N
      I=NP1-K
      IP=IKP(1)
      IF(I-IP)701,709,701
701  TEMP=B(I)
      B(I)=B(IP)
      B(IP)=TEMP
      DO 702 J=1,N
      TEMP=AA(I,J)
      AA(I,J)=AA(IP,J)
702  AA(IP,J)=TEMP
709  CONTINUE
      DO 729 I=1,N
      b(I)=B(I)/RM(I)
      DO 729 J=1,N
      A(I,J)=AA(I,J)/RM(I)
729  CONTINUE
      RETURN
      END

```

SIBMAP DØT 84,DECK,LIST

\* DØT AND FRIENDS ROUTINES FOR USE WITH SOLVE

ENTRY DØT (N,A(1),MA,B(1),MB,C) DOUBLE INNER PRODUCT

ENTRY SDØT (N,A(1),MA,B(1),MB,C) INNER PRODUCT

ENTRY ILØG2 (A) FLOATING POINT EXPONENT

ENTRY DAD (A,B) ADD WITH ROUND

\* SNAD MACRØ M STORE NEGATIVE ØF ADDRESS IN DECREMENT

SUB =Ø1ØØØØØ COMPLEMENT IF POSITIVE

ALS 18

STD M

ENDM SNAD

\* DØT SAVE 1,2,4

STZ S

STZ S+1

CLA\* 8,4 C

LDQ C+1

STØ C

CLA\* 3,4 N

TZE NØNE SKIP LOOP IF N = 0

STØ N

CLA 4,4 BASE ADDRESS ØF A

PAC ,1 X1=-(BASE ØF A)

CLA\* 5,4 MA

SNAD MA

CLA 6,4 BASE ADDRESS ØF B

PAC ,2 X2=-(BASE ØF B)

CLA\* 7,4 MB

SNAD MB

LXA N,4 X4=N

LØØP LDQ 0,1 A(1)

FMP 0,2 B(1)

DFAD S

DST S

MA TXI \*\*1,1,\*\* (X1)=(X1)+MA

MB TXI \*\*1,2,\*\* (X2)=(X2)+MB

TIX LØØP,4,1 END ØF MAIN LØØP

NØNE DFAD C

FRN

RETURN DØT

\* SDØT SAVE 1,2,4

STZ S

CLA\* 8,4

STØ C

CLA\* 3,4

TZE SNØNE

STØ N

CLA 4,4

PAC ,1

CLA\* 5,4

SNAD SMA

CLA 6,4

PAC ,2

CLA\* 7,4

SNAD SMB

LXA N,4

SLØØP LDQ 0,1

FMP 0,2

FAD S

DØT40010

DØT40020

DØT40030

DØT40040

DØT40050

DØT40060

DØT40070

DØT40080

DØT40090

DØT40100

DØT40110

DØT40120

DØT40130

DØT40140

DØT40150

DØT40160

DØT40170

DØT40180

DØT40190

DØT40200

DØT40210

DØT40220

DØT40230

DØT40240

DØT40250

DØT40260

DØT40270

DØT40280

DØT40290

DØT40300

DØT40310

DØT40320

DØT40330

DØT40340

DØT40350

DØT40360

DØT40370

DØT40380

DØT40390

DØT40400

DØT40410

DØT40420

DØT40430

DØT40440

DØT40450

DØT40460

DØT40470

DØT40480

DØT40490

DØT40500

DØT40510

DØT40520

DØT40530

DØT40540

DØT40550

DØT40560

DØT40570

DØT40580

DØT40590

DØT40600

|         |        |                                                 |  |            |
|---------|--------|-------------------------------------------------|--|------------|
|         | STØ    | S                                               |  | DØT40610   |
| SMA     | TXI    | **1,1,**                                        |  | DØT40620   |
| SMB     | TXI    | **1,2,**                                        |  | DØT40630   |
|         | TIX    | SLØØP,4,1                                       |  | DØT40640   |
| ØNE     | FAD    | C                                               |  | DØT40650   |
|         | RETURN | SDØT                                            |  | DØT40660   |
| *       |        |                                                 |  | DØT40670   |
| ILØG2   | CAL*   | 3,4                                             |  | DØT40680   |
|         | ANA    | =Ø37700000000                                   |  | DØT40690   |
|         | SUB    | =Ø200000000000                                  |  | DØT40700   |
|         | ARS    | 27                                              |  | DØT40710   |
|         | TRA    | 1,4                                             |  | DØT40720   |
| *       |        |                                                 |  | DØT40730   |
| DAD     | CLA*   | 3,4                                             |  | DØT40740   |
|         | FAD*   | 4,4                                             |  | DØT40750   |
|         | FRN    |                                                 |  | DØT40760   |
|         | TRA    | 1,4                                             |  | DØT40770   |
| *       |        |                                                 |  | DØT40780   |
|         | EVEN   |                                                 |  | DØT40790   |
| C       | PZE    |                                                 |  | DØT40800   |
|         | PZE    |                                                 |  | DØT40810   |
| S       | PZE    |                                                 |  | DØT40820   |
|         | PZE    |                                                 |  | DØT40830   |
| N       | PZE    |                                                 |  | DØT40840   |
|         | END    |                                                 |  | DØT40850   |
| \$IRMAP | RWEG2F | 800,LIST,REF,DECK,M9C,RELMØD,NØ(),NØMFTC        |  | RWEG2F0000 |
| *RWEG2F |        | RØUTINE BETWEEN JACØBI AND EG2F                 |  | RWEG2F0010 |
| *       |        | TØ SET UP SYMBOLIC CALLING SEQUENCE             |  | RWEG2F0020 |
| EG2F    | ENTRY  | RWEG2F                                          |  | RWEG2F0030 |
|         | SAVE   | (1,2,4) SAVE INDEX REGISTERS.                   |  | RWEG2F0040 |
|         | CLA    | 3,4                                             |  | RWEG2F0050 |
|         | STA    | SUB+1                                           |  | RWEG2F0060 |
|         | CLA    | 4,4 B.                                          |  | RWEG2F0070 |
|         | PAX    | ,2                                              |  | RWEG2F0080 |
|         | SXD    | SUB+1,2 A.                                      |  | RWEG2F0090 |
|         | CLA    | 5,4 VECTØR.                                     |  | RWEG2F0100 |
|         | STA    | SUB+2                                           |  | RWEG2F0110 |
|         | CLA*   | 6,4 N.                                          |  | RWEG2F0120 |
|         | PAX    | ,2                                              |  | RWEG2F0130 |
|         | SXD    | SUB+2,2                                         |  | RWEG2F0140 |
|         | CLA    | 7,4 NØGØ.                                       |  | RWEG2F0150 |
|         | STA    | TEMP                                            |  | RWEG2F0160 |
|         | ØCT    | 47600000016 EMTM ØN 7094 CØMPUTER, NØP ØN 7090  |  | RWEG2F0170 |
| SUB     | TSX    | EG2F,4 EG2F CALLING SEQUENCE.                   |  | RWEG2F0180 |
|         | PZE    | **,**, A,,B.                                    |  | RWEG2F0190 |
|         | PZE    | **,**, V,,N.                                    |  | RWEG2F0200 |
|         | TRA    | ERRØR ERRØR EXIT.                               |  | RWEG2F0210 |
|         | STZ*   | TEMP NØRMAL EXIT.                               |  | RWEG2F0220 |
| RET1    | ØCT    | Ø76000000016 LMTM ØN 7094 CØMPUTER, NØP ØN 7090 |  | RWEG2F0230 |
|         | RETURN | RWEG2F                                          |  | RWEG2F0240 |
| TEMP    | PZE    |                                                 |  | RWEG2F0250 |
| ERRØR   | TZE    | ACC                                             |  | RWEG2F0260 |
|         | CLS    | =1 NØGØ                                         |  | RWEG2F0270 |
|         | STØ*   | TEMP =-1.                                       |  | RWEG2F0280 |
|         | TRA    | RET1                                            |  | RWEG2F0290 |
| CC      | CLA    | =1 NØGØ                                         |  | RWEG2F0300 |
|         | STØ*   | TEMP =1.                                        |  | RWEG2F0310 |
|         | TRA    | RET1                                            |  | RWEG2F0320 |
|         | EJECT  |                                                 |  | RWEG2F0330 |
|         | TTL    | SHARE SUBRØUTINE EG2F                           |  | RWEG2F0340 |
| EG2F    | SXD    | EG2F+Ø209,4                                     |  | RWEG2F0350 |
|         |        |                                                 |  | EG2F0360   |

|     |             |                   |          |
|-----|-------------|-------------------|----------|
| SXD | EG2F+0210,1 |                   | EG2F0004 |
| SXD | EG2F+0211,2 |                   | EG2F0005 |
| CLA | 1,4         |                   | EG2F0006 |
| STA | EG2F+0215   | L2C A MATRIX      | EG2F0007 |
| STA | EG2F+0042   |                   | EG2F0008 |
| STA | EG2F+0066   |                   | EG2F0009 |
| STA | EG2F+0068   |                   | EG2F0010 |
| ARS | 18          |                   | EG2F0011 |
| STA | EG2F+0029   | L2C B MATRIX      | EG2F0012 |
| STA | EG2F+0214   |                   | EG2F0013 |
| CLA | 2,4         |                   | EG2F0014 |
| STA | EG2F+0216   | L2C V MATRIX      | EG2F0015 |
| STA | EG2F+0030   |                   | EG2F0016 |
| STD | EG2F+0207   |                   | EG2F0017 |
| ALS | 18          |                   | EG2F0018 |
| STD | EG2F+0042   |                   | EG2F0019 |
| LDQ | EG2F+0207   |                   | EG2F0020 |
| MPY | EG2F+0207   |                   | EG2F0021 |
| ALS | 17          |                   | EG2F0022 |
| C0M |             |                   | EG2F0023 |
| ADD | EG2F+0208   |                   | EG2F0024 |
| STD | EG2F+0213   | -(N SQUARED)      | EG2F0025 |
| CLA | EG2F+ 207   |                   | EG2F0026 |
| C0M |             |                   | EG2F0027 |
| ADD | EG2F+0208   |                   | EG2F0028 |
| STD | EG2F+0029   |                   | EG2F0029 |
| STD | EG2F+0212   |                   | EG2F0030 |
| TSX | EG2F+0217,4 |                   | EG2F0031 |
| PZE | *,0,*       |                   | EG2F0032 |
| PZE | *           |                   | EG2F0033 |
| TRA | EG2F+0053   |                   | EG2F0034 |
| TSX | EG2F+0072,4 | F0RM M(-1)TAM(-1) | EG2F0035 |
| LXD | EG2F+0209,4 |                   | EG2F0036 |
| CLA | 1,4         |                   | EG2F0037 |
| TPL | EG2F+0038   |                   | EG2F0038 |
| CLS | EG2F+0207   |                   | EG2F0039 |
| TRA | EG2F+0039   |                   | EG2F0040 |
| CLA | EG2F+0207   |                   | EG2F0041 |
| ARS | 18          |                   | EG2F0042 |
| ST0 | EG2F+0043   |                   | EG2F0043 |
| TSX | EG2F+0370,4 |                   | EG2F0044 |
| PZE | *,0,*       |                   | EG2F0045 |
| PZE | *           |                   | EG2F0046 |
| TRA | EG2F+0055   |                   | EG2F0047 |
| TRA | EG2F+0060   |                   | EG2F0048 |
| TSX | EG2F+0181,4 |                   | EG2F0049 |
| TSX | EG2F+0132,4 |                   | EG2F0050 |
| TSX | EG2F+0191,4 |                   | EG2F0051 |
| LXD | EG2F+0209,4 |                   | EG2F0052 |
| LXD | EG2F+0210,1 |                   | EG2F0053 |
| LXD | EG2F+0211,2 |                   | EG2F0054 |
| TRA | 4,4         |                   | EG2F0055 |
| PXD |             |                   | EG2F0056 |
| TRA | EG2F+0056   |                   | EG2F0057 |
| CLA | EG2F+0344   |                   | EG2F0058 |
| LXD | EG2F+0210,1 |                   | EG2F0059 |
| LXD | EG2F+0211,2 |                   | EG2F0060 |
| LXD | EG2F+0209,4 |                   | EG2F0061 |
| TRA | 3,4         |                   | EG2F0062 |
| ST0 | COMMON+000  |                   | EG2F0063 |
| CLA | EG2F+0207   |                   | EG2F0064 |

|                    |    |          |
|--------------------|----|----------|
| PDX 0,1            | N  | EG2F0065 |
| CØM                |    | EG2F0066 |
| STD EG2F+0069      |    | EG2F0067 |
| LXD EG2F+0342,2    |    | EG2F0068 |
| CLA *,2            |    | EG2F0069 |
| FDP CØMMØN+000     |    | EG2F0070 |
| STQ *,2            |    | EG2F0071 |
| TXI EG2F+0070,2,*  |    | EG2F0072 |
| TIX EG2F+0066,1,1  |    | EG2F0073 |
| TRA EG2F+0049      |    | EG2F0074 |
| SXD CØMMØN+004,4   |    | EG2F0075 |
| CLA EG2F+0213      |    | EG2F0076 |
| STD EG2F+0101      |    | EG2F0077 |
| STD EG2F+0124      |    | EG2F0078 |
| STD EG2F+0127      |    | EG2F0079 |
| CLA EG2F+0212      |    | EG2F0080 |
| STD EG2F+0126      |    | EG2F0081 |
| CLA EG2F+0214      |    | EG2F0082 |
| STA EG2F+0116      |    | EG2F0083 |
| CLA EG2F+0215      |    | EG2F0084 |
| STA EG2F+0117      |    | EG2F0085 |
| CLA EG2F+0216      |    | EG2F0086 |
| STA EG2F+0099      |    | EG2F0087 |
| STA EG2F+0118      |    | EG2F0088 |
| STA EG2F+0119      |    | EG2F0089 |
| TSX EG2F+0097,4    |    | EG2F0090 |
| CLA EG2F+0216      |    | EG2F0091 |
| STA EG2F+0117      |    | EG2F0092 |
| CLA EG2F+0215      |    | EG2F0093 |
| STA EG2F+0099      |    | EG2F0094 |
| STA EG2F+0118      |    | EG2F0095 |
| STA EG2F+0119      |    | EG2F0096 |
| TSX EG2F+0097,4    |    | EG2F0097 |
| LXD CØMMØN+004,4   |    | EG2F0098 |
| TRA 1,4            |    | EG2F0099 |
| SXD CØMMØN+003,4   |    | EG2F0100 |
| LXA EG2F+0342,4    |    | EG2F0101 |
| STZ *,4            |    | EG2F0102 |
| TXI EG2F+0101,4,-1 |    | EG2F0103 |
| TXH EG2F+0099,4,*  |    | EG2F0104 |
| CLA EG2F+0341      |    | EG2F0105 |
| STØ CØMMØN+002     |    | EG2F0106 |
| LXA EG2F+0342,4    |    | EG2F0107 |
| SXD EG2F+0122,4    |    | EG2F0108 |
| CLA EG2F+0122      |    | EG2F0109 |
| STD CØMMØN+001     |    | EG2F0110 |
| ADD CØMMØN+002     |    | EG2F0111 |
| STD EG2F+0122      |    | EG2F0112 |
| CLA CØMMØN+002     |    | EG2F0113 |
| ADD EG2F+0341      |    | EG2F0114 |
| STD CØMMØN+002     |    | EG2F0115 |
| STZ CØMMØN+000     |    | EG2F0116 |
| LXD CØMMØN+000,2   | 00 | EG2F0117 |
| LXD CØMMØN+001,1   |    | EG2F0118 |
| LDC *,1            |    | EG2F0119 |
| FMP *,2            |    | EG2F0120 |
| FAD *,4            |    | EG2F0121 |
| STØ *,4            |    | EG2F0122 |
| TXI EG2F+0121,1,-1 |    | EG2F0123 |
| TXI EG2F+0122,2,-1 |    | EG2F0124 |
| TXH EG2F+0116,1,*  |    | EG2F0125 |

TXI EG2F+0124,4,-1

TXL EG2F+0130,4,\*

LXD C0MM0N+000,2 00

TXI EG2F+0127,2,\*

TXL EG2F+0106,2,\*

SXD C0MM0N+000,2 00

TRA EG2F+0114

LXD C0MM0N+003,4

TRA 1,4

SXD C0MM0N+004,4

CLA EG2F+0214

STA EG2F+0154

CLA EG2F+0216

STA EG2F+0155

STA EG2F+0165

CLA EG2F+0213

STD EG2F+0163

STD EG2F+0168

CLA EG2F+0212

STD EG2F+0162

STD EG2F+0169

CLA EG2F+0341

STD C0MM0N+002

STZ C0MM0N+000

STZ C0MM0N+001

LXA EG2F+0342,4

LXD C0MM0N+001,1

LXD C0MM0N+000,2 00

CLA C0MM0N+002

STD EG2F+0158

STZ C0MM0N+003

LDQ \*,1

FMP \*,2

FAD C0MM0N+003

ST0 C0MM0N+003

TXI EG2F+0159,1,\*

CLA EG2F+0158

ADD EG2F+0341

STD EG2F+0158

TXI EG2F+0163,2,\*

TXH EG2F+0154,2,\*

CLA C0MM0N+003

ST0 \*,4

TXI EG2F+0167,4,-1

SXD C0MM0N+000,4 00

TXL EG2F+0179,4,\*

TXH EG2F+0149,4,\*

CLA EG2F+0169

ADD EG2F+0212

STD EG2F+0169

CLA C0MM0N+002

ADD EG2F+0341

STD C0MM0N+002

ADD C0MM0N+001

STD C0MM0N+001

TRA EG2F+0149

LXD C0MM0N+004,4

TRA 1,4

CLA EG2F+0212

STD EG2F+0199

CLA EG2F+0213

EG2F0126

EG2F0127

EG2F0128

EG2F0129

EG2F0130

EG2F0131

EG2F0132

EG2F0133

EG2F0134

EG2F0135

EG2F0136

EG2F0137

EG2F0138

EG2F0139

EG2F0140

EG2F0141

EG2F0142

EG2F0143

EG2F0144

EG2F0145

EG2F0146

EG2F0147

EG2F0148

EG2F0149

EG2F0150

EG2F0151

EG2F0152

EG2F0153

EG2F0154

EG2F0155

EG2F0156

EG2F0157

EG2F0158

EG2F0159

EG2F0160

EG2F0161

EG2F0162

EG2F0163

EG2F0164

EG2F0165

EG2F0166

EG2F0167

EG2F0168

EG2F0169

EG2F0170

EG2F0171

EG2F0172

EG2F0173

EG2F0174

EG2F0175

EG2F0176

EG2F0177

EG2F0178

EG2F0179

EG2F0180

EG2F0181

EG2F0182

EG2F0183

EG2F0184

EG2F0185

EG2F0186

|     |                |    |          |
|-----|----------------|----|----------|
| STD | EG2F+0200      |    | EG2FC187 |
| STD | EG2F+0205      |    | EG2FC188 |
| CLA | EG2F+0216      |    | EG2FC189 |
| STA | EG2F+0194      |    | EG2FC190 |
| STA | EG2F+0195      |    | EG2FC191 |
| STA | EG2F+0196      |    | EG2FC192 |
| STA | EG2F+0197      |    | EG2FC193 |
| STZ | COMMON+000     |    | EG2FC194 |
| LXD | COMMON+000,1   | 00 | EG2FC195 |
| LXD | COMMON+000,2   | 00 | EG2FC196 |
| CLA | *,1            |    | EG2FC197 |
| LDQ | *,2            |    | EG2FC198 |
| STQ | *,1            |    | EG2FC199 |
| STQ | *,2            |    | EG2FC200 |
| TXI | EG2F+0199,1,-1 |    | EG2FC201 |
| TXI | EG2F+0200,2,*  |    | EG2FC202 |
| TXH | EG2F+0194,2,*  |    | EG2FC203 |
| CLA | COMMON+000     |    | EG2FC204 |
| ADD | EG2F+0212      |    | EG2FC205 |
| ADD | EG2F+0341      |    | EG2FC206 |
| STD | COMMON+000     |    | EG2FC207 |
| TXH | EG2F+0192,1,*  |    | EG2FC208 |
| TRA | 1,4            |    | EG2FC209 |
| PZE |                |    | EG2FC210 |
| PZE | 0,0,1          |    | EG2FC211 |
| PZE |                |    | EG2FC212 |
| PZE |                |    | EG2FC213 |
| PZE |                |    | EG2FC214 |
| PZE |                |    | EG2FC215 |
| PZE |                |    | EG2FC216 |
| PZE |                |    | EG2FC217 |
| PZE |                |    | EG2FC218 |
| PZE |                |    | EG2FC219 |
| SXD | COMMON+010,1   |    | EG2FC220 |
| SXD | COMMON+011,2   |    | EG2FC221 |
| CLA | 2,4            |    | EG2FC222 |
| STA | EG2F+0295      |    | EG2FC223 |
| STA | EG2F+0296      |    | EG2FC224 |
| SUB | EG2F+0344      |    | EG2FC225 |
| STA | EG2F+0266      |    | EG2FC226 |
| STA | EG2F+0283      |    | EG2FC227 |
| CLA | 1,4            |    | EG2FC228 |
| STD | EG2F+0311      |    | EG2FC229 |
| SXD | COMMON+004,4   |    | EG2FC230 |
| LXD | EG2F+0341,2    |    | EG2FC231 |
| STA | EG2F+0282      |    | EG2FC232 |
| STA | EG2F+0236      |    | EG2FC233 |
| STA | EG2F+0209      |    | EG2FC234 |
| STA | EG2F+0326      |    | EG2FC235 |
| STA | EG2F+0328      |    | EG2FC236 |
| STA | EG2F+0258      |    | EG2FC237 |
| STA | EG2F+0259      |    | EG2FC238 |
| CLA | *              |    | EG2FC239 |
| STZ | COMMON+013,2   |    | EG2FC240 |
| CLA | EG2F+0343      |    | EG2FC241 |
| STZ | *              |    | EG2FC242 |
| CLA | EG2F+0342      |    | EG2FC243 |
| STZ | COMMON+005     |    | EG2FC244 |
| CLA | COMMON+005     |    | EG2FC245 |
| ADD | EG2F+0344      |    | EG2FC246 |
| STZ | COMMON+005     |    | EG2FC247 |

|                    |                      |          |
|--------------------|----------------------|----------|
| ADD EG2F+0259      |                      | EG2FC248 |
| STA EG2F+0259      |                      | EG2FC249 |
| STA EG2F+0273      |                      | EG2FC250 |
| STA EG2F+0285      |                      | EG2FC251 |
| STA EG2F+0284      |                      | EG2FC252 |
| STA EG2F+0308      |                      | EG2FC253 |
| STA EG2F+0304      |                      | EG2FC254 |
| SXD EG2F+0267,2    | COMP LAMBDA          | EG2FC255 |
| LXD EG2F+0342,4    |                      | EG2FC256 |
| LXD EG2F+0341,1    |                      | EG2FC257 |
| SXD EG2F+0264,1    |                      | EG2FC258 |
| LXD EG2F+0342,1    |                      | EG2FC259 |
| STZ COMMON+006     |                      | EG2FC260 |
| LDQ *,4            |                      | EG2FC261 |
| FMP *,1            |                      | EG2FC262 |
| FAD COMMON+006     |                      | EG2FC263 |
| ST0 COMMON+006     |                      | EG2FC264 |
| TXI EG2F+0265,4,-1 |                      | EG2FC265 |
| TXI EG2F+0264,1,-1 |                      | EG2FC266 |
| TXH EG2F+0258,1,*  |                      | EG2FC267 |
| FDP COMMON+013,1   |                      | EG2FC268 |
| STQ *,1            |                      | EG2FC269 |
| TXH EG2F+0269,1,*  |                      | EG2FC270 |
| TRA EG2F+0271      |                      | EG2FC271 |
| LXD EG2F+0264,1    |                      | EG2FC272 |
| TXI EG2F+0255,1,-1 |                      | EG2FC273 |
| SXD EG2F+0275,2    |                      | EG2FC274 |
| LXD EG2F+0342,4    |                      | EG2FC275 |
| STZ *,4            |                      | EG2FC276 |
| TXI EG2F+0275,4,-1 |                      | EG2FC277 |
| TXH EG2F+0273,4,*  |                      | EG2FC278 |
| TXI EG2F+0277,2,-1 |                      | EG2FC279 |
| SXD EG2F+0290,2    |                      | EG2FC280 |
| LXD EG2F+0342,4    |                      | EG2FC281 |
| LXD EG2F+0341,2    |                      | EG2FC282 |
| LXD EG2F+0342,1    |                      | EG2FC283 |
| SXD EG2F+0288,2    |                      | EG2FC284 |
| LDQ *,4            | A LAMBDA TIMES A ROW | EG2FC285 |
| FMP *,2            |                      | EG2FC286 |
| FAD *,1            | VECTOR WHICH WILL BE | EG2FC287 |
| ST0 *,1            | THE NEW ROW          | EG2FC288 |
| TXI EG2F+0287,4,-1 |                      | EG2FC289 |
| TXI EG2F+0288,1,-1 |                      | EG2FC290 |
| TXH EG2F+0282,1,*  |                      | EG2FC291 |
| TXI EG2F+0290,2,-1 |                      | EG2FC292 |
| TXH EG2F+0280,2,*  | END COMP. NEW ROW    | EG2FC293 |
| STZ COMMON+008     |                      | EG2FC294 |
| LXD EG2F+0288,2    |                      | EG2FC295 |
| SXD EG2F+0303,2    |                      | EG2FC296 |
| LXD EG2F+0342,4    |                      | EG2FC297 |
| LDQ *,4            |                      | EG2FC298 |
| FMP *,4            |                      | EG2FC299 |
| ST0 COMMON+007     |                      | EG2FC300 |
| LDQ COMMON+007     |                      | EG2FC301 |
| FMP COMMON+014,4   |                      | EG2FC302 |
| FAD COMMON+008     |                      | EG2FC303 |
| ST0 COMMON+008     |                      | EG2FC304 |
| TXI EG2F+0303,4,-1 |                      | EG2FC305 |
| TXH EG2F+0295,4,*  |                      | EG2FC306 |
| CLA *,1            |                      | EG2FC307 |
| FSB COMMON+008     |                      | EG2FC308 |

|                    |                            |          |
|--------------------|----------------------------|----------|
| STO COMMON+014,2   |                            | EG2F0309 |
| CLA EG2F+0343      |                            | EG2F0310 |
| STO *,1            |                            | EG2F0311 |
| TXI EG2F+0310,2,-1 |                            | EG2F0312 |
| LXD EG2F+0267,1    |                            | EG2F0312 |
| TXH EG2F+0242,2,*  |                            | EG2F0312 |
| SXD EG2F+0321,2    |                            | EG2F0314 |
| TXI EG2F+0314,2,-2 | -(N+2)                     | EG2F0315 |
| SXD EG2F+0334,2    |                            | EG2F0316 |
| LXD EG2F+0342,1    |                            | EG2F0317 |
| CLA COMMON+014,1   |                            | EG2F0318 |
| TSX EG2F+0345,4    |                            | EG2F0319 |
| TRA EG2F+0339      |                            | EG2F0320 |
| STO COMMON+014,1   |                            | EG2F0321 |
| TXI EG2F+0321,1,-1 |                            | EG2F0322 |
| TXH EG2F+0316,1,*  |                            | EG2F0323 |
| LXD EG2F+0341,6    |                            | EG2F0324 |
| LXD EG2F+0342,1    |                            | EG2F0325 |
| TXI EG2F+0325,2,-1 |                            | EG2F0326 |
| SXD EG2F+0330,4    |                            | EG2F0327 |
| CLA *,1            |                            | EG2F0328 |
| FDP COMMON+012,2   |                            | EG2F0329 |
| STO *,1            |                            | EG2F0330 |
| TXI EG2F+0330,1,-1 |                            | EG2F0331 |
| TXH EG2F+0326,1,*  |                            | EG2F0332 |
| SXD EG2F+0332,2    |                            | EG2F0333 |
| TXI EG2F+0333,4,*  |                            | EG2F0334 |
| TXI EG2F+0334,2,-1 |                            | EG2F0335 |
| TXH EG2F+0325,2,*  |                            | EG2F0336 |
| LXD COMMON+010,1   |                            | EG2F0337 |
| LXD COMMON+011,2   |                            | EG2F0338 |
| LXD COMMON+004,4   |                            | EG2F0339 |
| TRA 4,4            |                            | EG2F0340 |
| LXD COMMON+004,4   |                            | EG2F0341 |
| TRA 3,4            |                            | EG2F0342 |
| PZE 1,0,-1         |                            | EG2F0343 |
| PZE                |                            | EG2F0344 |
| DEC -1.            |                            | EG2F0345 |
| PZE 1              |                            | EG2F0346 |
| STO COMMON+000     | SAVE ARGUMENT SIGN         | EG2F0347 |
| SSP                | N                          | EG2F0348 |
| TZE EG2F+0365      | GO TO EXIT IF ZERO         | EG2F0349 |
| STO COMMON+001     | SAVE N                     | EG2F0350 |
| ANA EG2F+0368      | COMPUTE TRIAL VALUE, X     | EG2F0351 |
| LRS 1              | X                          | EG2F0352 |
| ADD COMMON+001     | X                          | EG2F0353 |
| LRS 1              | X                          | EG2F0354 |
| ADD EG2F+0369      | X                          | EG2F0355 |
| SXD COMMON+000,4   | GO RETURN ADDRESS          | EG2F0356 |
| LXA EG2F+0346,4    | SET INDEX FOR 3 ITERATIONS | EG2F0357 |
| STO COMMON+002     | SAVE X                     | EG2F0358 |
| CLA COMMON+001     | COMPUTE SQUARE ROOT        | EG2F0359 |
| FDP COMMON+002     | X N/X                      | EG2F0360 |
| STQ COMMON+003     | X                          | EG2F0361 |
| CLA COMMON+003     | X N/X                      | EG2F0362 |
| FAD COMMON+002     | X + X                      | EG2F0363 |
| SUB EG2F+0368      | X DIVIDE BY 2              | EG2F0364 |
| TIX EG2F+0356,4,1  | REPEAT LOOP                | EG2F0365 |
| LXD COMMON+000,4   | GOORE EXIT ADDRESS         | EG2F0366 |
| LDQ COMMON+000     | TEST SIGN OF ARGUMENT      | EG2F0367 |
| TQP 2,4            | IF + , SKIP ONE            | EG2F0368 |
|                    |                            | EG2F0369 |

TRA 1,4 IF - , DO NOT SKIP ONE  
 DEC 134217728,8657043456 0\*2 EXP -127, 1/2 \* EXP -64  
 REM NYEVV 328 CARDS EIGENVALUES AND EIGENVECTORS  
 SXD COMMON+000,1 00 SAVE IRS  
 SXD COMMON+001,2  
 SXD COMMON+002,4  
 PXD SAVE SLTS  
 LXA EG2F+0694,1  
 ALS 1  
 SLT 5,1  
 ADD EG2F+0375  
 TIX EG2F+0375,1,1  
 STA EG2F+0373  
 CLA 1,4 M  
 PAX 0,1  
 STA EG2F+0488  
 STA EG2F+0534  
 STA EG2F+0536  
 STA EG2F+0538  
 STA EG2F+0558  
 ADM 2,4 M+N  
 STA EG2F+0482  
 STA EG2F+0489  
 STA EG2F+0539  
 STA EG2F+0543  
 CAL 2,4 N  
 STA EG2F+0381  
 ADD EG2F+0375 N+1  
 STA EG2F+0464  
 ALS 18  
 STD EG2F+0604  
 SUB EG2F+0695 N-1  
 STD EG2F+0465  
 COM -N  
 STD EG2F+0560  
 STD EG2F+0561  
 SUB EG2F+0696 -(N+1)  
 STD EG2F+0567  
 LDQ 2,4 N\*\*2  
 MPY 2,4  
 STQ COMMON+003  
 CLA 1,4 M+N\*\*2  
 ADD COMMON+003  
 STA EG2F+0441  
 STA EG2F+0442  
 STA EG2F+0611  
 STA EG2F+0623  
 STA EG2F+0625  
 STA EG2F+0638  
 STA EG2F+0641  
 COM -V  
 ADD EG2F+0696  
 STQ COMMON+004  
 PXD 0,1  
 ADD COMMON+004 -(V-M)  
 STD COMMON+004  
 ADM EG2F+0567 -(V-M)-N  
 STD COMMON+005  
 CLA 2,4 FL0AT N  
 SSP  
 ADD EG2F+0697

EG2F0370  
 EG2F0371  
 EG2F0372  
 EG2F0373  
 EG2F0374  
 EG2F0375  
 EG2F0376  
 EG2F0377  
 EG2F0378  
 EG2F0379  
 EG2F0380  
 EG2F0381  
 EG2F0382  
 EG2F0383  
 EG2F0384  
 EG2F0385  
 EG2F0386  
 EG2F0387  
 EG2F0388  
 EG2F0389  
 EG2F0390  
 EG2F0391  
 EG2F0392  
 EG2F0393  
 EG2F0394  
 EG2F0395  
 EG2F0396  
 EG2F0397  
 EG2F0398  
 EG2F0399  
 EG2F0400  
 EG2F0401  
 EG2F0402  
 EG2F0403  
 EG2F0404  
 EG2F0405  
 EG2F0406  
 EG2F0407  
 EG2F0408  
 EG2F0409  
 EG2F0410  
 EG2F0411  
 EG2F0412  
 EG2F0413  
 EG2F0414  
 EG2F0415  
 EG2F0416  
 EG2F0417  
 EG2F0418  
 EG2F0419  
 EG2F0420  
 EG2F0421  
 EG2F0422  
 EG2F0423  
 EG2F0424  
 EG2F0425  
 EG2F0426  
 EG2F0427  
 EG2F0428  
 EG2F0429  
 EG2F0430

|                   |                      |          |
|-------------------|----------------------|----------|
| FAD EG2F+0697     |                      | EG2F0431 |
| ST0 C0MM0N+006    |                      | EG2F0432 |
| CLA 2,4           | SET N0 VECT0RS       | EG2F0433 |
| TPL EG2F+0594     |                      | EG2F0434 |
| CLA EG2F+0689     |                      | EG2F0435 |
| ST0 EG2F+0528     |                      | EG2F0436 |
| CLA EG2F+0704     |                      | EG2F0437 |
| ST0 C0MM0N+007    |                      | EG2F0438 |
| CLA EG2F+0690     |                      | EG2F0439 |
| ST0 EG2F+0593     |                      | EG2F0440 |
| T0V EG2F+0439     |                      | EG2F0441 |
| STZ C0MM0N+008    | CALCULATE N0RM       | EG2F0442 |
| LXA C0MM0N+003,7  |                      | EG2F0443 |
| LDQ 0,1           |                      | EG2F0444 |
| FMP 0,1           |                      | EG2F0445 |
| T0V EG2F+0611     |                      | EG2F0446 |
| FAD C0MM0N+008    |                      | EG2F0447 |
| T0V EG2F+0651     |                      | EG2F0448 |
| ST0 C0MM0N+008    |                      | EG2F0449 |
| TIX EG2F+0441,1,1 |                      | EG2F0450 |
| SLF               |                      | EG2F0451 |
| TSX EG2F+0655,4   |                      | EG2F0452 |
| FDH C0MM0N+006    |                      | EG2F0453 |
| ST0 C0MM0N+006    |                      | EG2F0454 |
| CLA C0MM0N+006    |                      | EG2F0455 |
| SUB EG2F+0698     |                      | EG2F0456 |
| ST0 C0MM0N+008    | FINAL N0RM           | EG2F0457 |
| CLA C0MM0N+006    |                      | EG2F0458 |
| SUB EG2F+0699     |                      | EG2F0459 |
| ST0 C0MM0N+006    | PR0VISIONAL N0RM     | EG2F0460 |
| LXA EG2F+0569,1   | SET -(J-2)N-(K-1)=0  | EG2F0461 |
| CLA EG2F+0534     | UNDERSET J=1         | EG2F0462 |
| STA EG2F+0487     |                      | EG2F0463 |
| STA EG2F+0522     |                      | EG2F0464 |
| CLA EG2F+0696     | SET J-1=1            | EG2F0465 |
| ST0 C0MM0N+003    |                      | EG2F0466 |
| PDX 0,2           | SET J-K=1            | EG2F0467 |
| TXH EG2F+0576,2   | J-K GREATER THAN N-1 | EG2F0468 |
| CLA EG2F+0522     | J T0 J+1             | EG2F0469 |
| ADD EG2F+0381     |                      | EG2F0470 |
| STA EG2F+0522     |                      | EG2F0471 |
| STA EG2F+0525     |                      | EG2F0472 |
| STA EG2F+0557     |                      | EG2F0473 |
| CLA EG2F+0487     |                      | EG2F0474 |
| ADM EG2F+0464     |                      | EG2F0475 |
| STA EG2F+0487     |                      | EG2F0476 |
| STA EG2F+0540     |                      | EG2F0477 |
| STA EG2F+0542     |                      | EG2F0478 |
| LXA EG2F+0569,4   | SET (1-K)(N+1)=0     | EG2F0479 |
| CLA EG2F+0534     | SET K=1              | EG2F0480 |
| STA EG2F+0521     |                      | EG2F0481 |
| STA EG2F+0524     |                      | EG2F0482 |
| STA EG2F+0555     |                      | EG2F0483 |
| CLA C0MM0N+006    | N0RM                 | EG2F0484 |
| SRM 0,1           | -A(J,K)              | EG2F0485 |
| TPL EG2F+0564     | GREATER THAN 0       | EG2F0486 |
| SLN 4             | SMALLER              | EG2F0487 |
| SXD EG2F+0689,2   | SAVE IRS             | EG2F0488 |
| SXD EG2F+0654,4   |                      | EG2F0489 |
| CLA               | A(J,J)               | EG2F0490 |
| FSB 0,4           | -A(K,K)              | EG2F0491 |

|                    |                      |          |
|--------------------|----------------------|----------|
| FDH 0,1            | /A(J,K)              | EG2FC492 |
| STQ C0MM0N+011     |                      | EG2FC493 |
| CLA C0MM0N+011     |                      | EG2FC494 |
| TMI EG2F+0494      |                      | EG2FC495 |
| SLN 3              | SL3 0N IF G POSITIVE | EG2FC496 |
| SSP                |                      | EG2FC497 |
| SUB EG2F+0699      | G                    | EG2FC498 |
| ST0 C0MM0N+011     |                      | EG2FC499 |
| LDQ C0MM0N+011     | G**2                 | EG2FC500 |
| FMP C0MM0N+011     |                      | EG2FC501 |
| FAD EG2F+0704      | 1+G**2               | EG2FC502 |
| ST0 C0MM0N+012     |                      | EG2FC503 |
| TSX EG2F+0655,4    | (1+G**2)**1/2        | EG2FC504 |
| LRS 35             |                      | EG2FC505 |
| FMP C0MM0N+011     | G(1+G**2)**1/2       | EG2FC506 |
| FAD C0MM0N+012     | +1+G**2              | EG2FC507 |
| ST0 C0MM0N+011     |                      | EG2FC508 |
| CLA EG2F+0705      | (SINT)**2            | EG2FC509 |
| FDH C0MM0N+011     |                      | EG2FC510 |
| STQ C0MM0N+011     |                      | EG2FC511 |
| CLA C0MM0N+011     |                      | EG2FC512 |
| TSX EG2F+0655,4    | SIN T                | EG2FC513 |
| SLT 3              |                      | EG2FC514 |
| SSM                |                      | EG2FC515 |
| ST0 C0MM0N+012     |                      | EG2FC516 |
| CLA EG2F+0704      |                      | EG2FC517 |
| FSB C0MM0N+011     |                      | EG2FC518 |
| TSX EG2F+0655,4    | C0ST                 | EG2FC519 |
| ST0 C0MM0N+011     |                      | EG2FC520 |
| LXA EG2F+0569,2    | SET MAIN MATRIX      | EG2FC521 |
| CLA EG2F+0560      |                      | EG2FC522 |
| STD EG2F+0527      | SET I-1=0            | EG2FC523 |
| CLA 0,2            | A(K,I)               | EG2FC524 |
| LDQ 0,2            | A(J,I)               | EG2FC525 |
| TSX EG2F+0673,4    | B(K,I),B(J,I)        | EG2FC526 |
| ST0 0,2            |                      | EG2FC527 |
| STQ 0,2            |                      | EG2FC528 |
| TXI EG2F+0527,2,-1 | I T0 I+1             | EG2FC529 |
| TXH EG2F+0521,2    | I=N                  | EG2FC530 |
| HTR 0              |                      | EG2FC531 |
| TRA EG2F+0607      |                      | EG2FC532 |
| CAL EG2F+0689      | J-K                  | EG2FC533 |
| C0M                |                      | EG2FC534 |
| ADM EG2F+0654      | +(1-K)(N+1)          | EG2FC535 |
| PDX 0,4            |                      | EG2FC536 |
| LDQ 0,4            | A(K,J) IN MQ         | EG2FC537 |
| LXD EG2F+0654,2    | (1-K)(N+1)           | EG2FC538 |
| CLA 0,2            | A(K,K) IN AC         | EG2FC539 |
| TSX EG2F+0673,4    |                      | EG2FC540 |
| ST0 0,2            | B(K,K)               | EG2FC541 |
| CLA 0,1            | A(J,K) IN AC         | EG2FC542 |
| LDQ                | A(J,J) IN MQ         | EG2FC543 |
| TSX EG2F+0673,4    |                      | EG2FC544 |
| STQ                | B(J,J)               | EG2FC545 |
| ST0 0,1            | B(J,K)               | EG2FC546 |
| CLA C0MM0N+003     | J                    | EG2FC547 |
| C0M                |                      | EG2FC548 |
| ADD EG2F+0696      |                      | EG2FC549 |
| PDX 0,2            |                      | EG2FC550 |
| CAL EG2F+0689      | K-J                  | EG2FC551 |
| C0M                |                      | EG2FC552 |

|                    |                     |          |
|--------------------|---------------------|----------|
| ADD EG2F+0696      | M+K-J               | EG2FC552 |
| ARS 18             |                     | EG2FC554 |
| ADD EG2F+0534      |                     | EG2FC555 |
| STA EG2F+0556      |                     | EG2FC556 |
| LXA EG2F+0569,4    | SET I-1=0           | EG2FC557 |
| CLA 0,4            | A(K,I)              | EG2FC558 |
| ST0 0,2            | A(I,K)              | EG2FC559 |
| CLA 0,4            | A(J,I)              | EG2FC560 |
| ST0 0,2            | A(I,J)              | EG2FC561 |
| TXI EG2F+0560,4,-1 | I T0 I+1            | EG2FC562 |
| TXL EG2F+0562,4    | I=N                 | EG2FC563 |
| TXI EG2F+0555,2    |                     | EG2FC564 |
| LXD EG2F+0689,2    | RESTORE IRS         | EG2FC565 |
| LXD EG2F+0654,4    |                     | EG2FC566 |
| TNX EG2F+0569,2,1  | J-K=1               | EG2FC567 |
| CLA EG2F+0521      | K T0 K+1            | EG2FC568 |
| ADD EG2F+0381      |                     | EG2FC569 |
| TXI EG2F+0568,4    |                     | EG2FC570 |
| TXI EG2F+0478,1,-1 |                     | EG2FC571 |
| PXD 0,1            | M T0 M+1            | EG2FC572 |
| ADD C0MM0N+003     | AND K=1             | EG2FC573 |
| ADD EG2F+0567      |                     | EG2FC574 |
| PDX 0,1            |                     | EG2FC575 |
| CLA C0MM0N+003     |                     | EG2FC576 |
| ADD EG2F+0696      |                     | EG2FC577 |
| TRA EG2F+0463      |                     | EG2FC578 |
| SLT 4              | IS ITERATION IDLE   | EG2FC579 |
| TRA EG2F+0579      |                     | EG2FC580 |
| TRA EG2F+0458      |                     | EG2FC581 |
| CLA C0MM0N+006     | IS NORM FINAL       | EG2FC582 |
| L0Q C0MM0N+008     |                     | EG2FC583 |
| TLQ EG2F+0456      |                     | EG2FC584 |
| LXA EG2F+0694,1    | RESTORE SLTS        | EG2FC585 |
| CLA EG2F+0373      |                     | EG2FC586 |
| LRS 3              |                     | EG2FC587 |
| LBT                |                     | EG2FC588 |
| SLN 5,1            |                     | EG2FC589 |
| LLS 1              |                     | EG2FC590 |
| TIX EG2F+0585,1,1  |                     | EG2FC591 |
| LXD C0MM0N+000,1   | 00 RESTORE IRS      | EG2FC592 |
| LXD C0MM0N+001,2   |                     | EG2FC593 |
| LXD C0MM0N+002,4   |                     | EG2FC594 |
| CLA C0MM0N+007     |                     | EG2FC595 |
| HTR 0              |                     | EG2FC596 |
| CLA 1,4            | SETTING FOR VECTORS | EG2FC597 |
| ARS 18             |                     | EG2FC598 |
| ADD C0MM0N+003     |                     | EG2FC599 |
| STA EG2F+0600      |                     | EG2FC600 |
| STA EG2F+0603      |                     | EG2FC601 |
| LXA C0MM2N+003,3   | SET IDENTITY MATRIX | EG2FC602 |
| STZ 0,1            |                     | EG2FC603 |
| TIX EG2F+0600,1,1  |                     | EG2FC604 |
| CLA EG2F+0704      |                     | EG2FC605 |
| ST0 0,2            |                     | EG2FC606 |
| TIX EG2F+0603,2    |                     | EG2FC607 |
| CLA EG2F+0691      |                     | EG2FC608 |
| TRA EG2F+0463      |                     | EG2FC609 |
| SLN 2              | SET MT T0 VECTORS   | EG2FC610 |
| LXD C0MM2N+004,2   |                     | EG2FC611 |
| CLA C0MM0N+005     |                     | EG2FC612 |
| TRA EG2F+0520      |                     | EG2FC613 |

|                   |                            |          |
|-------------------|----------------------------|----------|
| CAL 0,1           | UNDERFLOW                  | EG2F0614 |
| ANA EG2F+0701     |                            | EG2F0615 |
| SUB EG2F+0702     |                            | EG2F0616 |
| TPL EG2F+0646     |                            | EG2F0617 |
| SLN 3             | SLN3 FOR UNDERFLOW         | EG2F0618 |
| SLT 2             | TEST FOR PREVIOUS OVERFLOW | EG2F0619 |
| TRA EG2F+0632     |                            | EG2F0620 |
| PXD 3             |                            | EG2F0621 |
| STD EG2F+0627     |                            | EG2F0622 |
| CLA EG2F+0692     | SET ERROR RETURN           | EG2F0623 |
| STQ EG2F+0593     |                            | EG2F0624 |
| SLF               |                            | EG2F0625 |
| CLA 0,4           | UNSCALE MATRIX             | EG2F0626 |
| FDH C0MM0N+007    |                            | EG2F0627 |
| STQ 0,4           |                            | EG2F0628 |
| TNX EG2F+0582,4,1 |                            | EG2F0629 |
| TXH EG2F+0623,4   |                            | EG2F0630 |
| CLA C0MM0N+007    |                            | EG2F0631 |
| FDH C0MM0N+013    |                            | EG2F0632 |
| STQ C0MM0N+007    |                            | EG2F0633 |
| TRA EG2F+0618     |                            | EG2F0634 |
| CHS               | SCALE MATRIX               | EG2F0635 |
| ADD EG2F+0704     |                            | EG2F0636 |
| STQ C0MM0N+013    |                            | EG2F0637 |
| LDQ C0MM2N+013    |                            | EG2F0638 |
| FMP C0MM0N+007    |                            | EG2F0639 |
| STQ C0MM0N+007    |                            | EG2F0640 |
| LDQ 0,2           |                            | EG2F0641 |
| FMP C0MM2N+013    |                            | EG2F0642 |
| TRV EG2F+0653     |                            | EG2F0643 |
| STQ 0,2           |                            | EG2F0644 |
| TIX EG2F+0638,2,1 |                            | EG2F0645 |
| CLA EG2F+0693     |                            | EG2F0646 |
| STQ EG2F+0593     |                            | EG2F0647 |
| TRA EG2F+0439     |                            | EG2F0648 |
| SUB EG2F+0703     | FMP OVERFLOW               | EG2F0649 |
| SLN 2             |                            | EG2F0650 |
| SLT 3             |                            | EG2F0651 |
| TRA EG2F+0632     | SCALE                      | EG2F0652 |
| TRA EG2F+0618     | UNSCALE                    | EG2F0653 |
| CLA EG2F+0699     | FAD OVERFLOW               | EG2F0654 |
| TRA EG2F+0647     |                            | EG2F0655 |
| SXD EG2F+0627,2   | SCALING OVERFLOW           | EG2F0656 |
| TXL EG2F+0623     |                            | EG2F0657 |
| SXD EG2F+0694,4   | SQUARE ROOT                | EG2F0658 |
| LXA EG2F+0618,4   |                            | EG2F0659 |
| STQ C0MM0N+013    |                            | EG2F0660 |
| ANA EG2F+0699     |                            | EG2F0661 |
| ARS 1             |                            | EG2F0662 |
| ADD C0MM2N+013    |                            | EG2F0663 |
| ARS 1             |                            | EG2F0664 |
| ADD EG2F+0700     |                            | EG2F0665 |
| STQ C0MM0N+009    |                            | EG2F0666 |
| CLA C0MM2N+013    |                            | EG2F0667 |
| FDH C0MM0N+009    |                            | EG2F0668 |
| CLA C0MM0N+009    |                            | EG2F0669 |
| STQ C0MM0N+009    |                            | EG2F0670 |
| FAD C0MM0N+009    |                            | EG2F0671 |
| SUB EG2F+0699     |                            | EG2F0672 |
| TIX EG2F+0663,4,1 |                            | EG2F0673 |
| LXD EG2F+0694,4   |                            | EG2F0674 |

TRA 1,4

STQ COMMON+013

STQ COMMON+009

FMP COMMON+011

STQ COMMON+010

LDQ COMMON+013

FMP COMMON+012

FAD COMMON+010

STQ COMMON+010

LDQ COMMON+009

FMP COMMON+012

STQ COMMON+009

LDQ COMMON+013

FMP COMMON+011

FSB COMMON+009

LDQ COMMON+010

TRA 1,4

TXL EG2F+0530

TRA 5,4

SLT 2

TRA 3,4

TRA 4,4

ØCT 4,2000000,1000000,233000000000,330000000000,1000000000

ØCT 100400000000,377000000000,101000000000,176000000000

DEC 1.,.5

COMMON BSS 54

END

MATRIX TRANSFORMATION

EG2FC675

EG2FC676

EG2FC677

EG2FC678

EG2FC679

EG2FC680

EG2FC681

EG2FC682

EG2FC683

EG2FC684

EG2FC685

EG2FC686

EG2FC687

EG2FC688

EG2FC689

EG2FC690

EG2FC691

EG2FC692

EG2FC693

EG2FC694

EG2FC695

EG2FC696

EG2FC697

EG2FC698

EG2FC699

RWEG2F100

RWEG2F100-