NASA CR 65863

LIBRARY COPY

NOV 24 1967

MANNED SPACECRAFT CENTER
HOUSTON, TEXAS

SID 67-498

STUDY OF APOLLO WATER IMPACT
FINAL REPORT

VOLUME 11

USER'S MANUAL FOR UNSYMMETRIC SHELL OF REVOLUTION ANALYSIS

(Contract NAS9-4552, G.O. 5264)

May 1967

Prepared by

E. Carrion
S.C. Furuike
T. Nishimoto
(Authors)

GPO PRICE $ _____

CFSTI PRICE(S) $ _____

Hard copy (HC) _____

Microfiche (MF) _____

ff 653 July 65

Approved by

F.C. Hung
Program Manager
Structures and Materials

L.A. Harris
Assistant Manager
Science and Technology

NORTH AMERICAN AVIATION, INC.
SPACE DIVISION

N68-11939 (ACCESSION NUMBER) (THRU)
326 (PAGES) 1 (CODE)
CR-65863 (NASA CR OR TMX OR AD NUMBER) 32 (CATEGORY)
602(D)

PRECEDING PAGE BLANK NOT FILMED.

FOREWORD

This report was prepared by North American Aviation, Inc., Space Division, under NASA Contract NAS9-4552, for the National Aeronautics and Space Administration, Manned Space Flight Center, Houston, Texas with Dr. F.C. Hung, Program Manager and Mr. P.P. Radkowski, Assistant Program Manager. This work was administered under the direction of Structural Mechanics Division, MSC, Houston, Texas with Dr. F. Stebbins as the technical monitor.

This report is presented in eleven volumes for convenience in handling and distribution. All volumes are unclassified.

The objective of the study was to develop methods and Fortran IV computer programs to determine by the techniques described below, the hydro-elastic response of representation of the structure of the Apollo Command Module immediately following impact on the water. The development of theory, methods and computer programs is presented as Task I Hydrodynamic Pressures, Task II Structural Response and Task III Hydroelastic Response Analysis.

Under Task I - Computing program to extend flexible sphere using the Spencer and Shiffman approach has been developed. Analytical formulation by Dr. Li using nonlinear hydrodynamic theory on structural portion is formulated. In order to cover a wide range of impact conditions, future extensions are necessary in the following items:

a. Using linear hydrodynamic theory to include horizontal velocity and rotation.

b. Nonlinear hydrodynamic theory to develop computing program on spherical portion and to develop nonlinear theory on toroidal and conic sections.

Under Task II - Computing program and User's Manual were developed for nonsymmetrical loading on unsymmetrical elastic shells. To fully develop the theory and methods to cover realistic Apollo configuration the following extensions are recommended:

a. Modes of vibration and modal analysis.

b. Extension to nonsymmetric short time impulses.

PRECEDING PAGE BLANK NOT FILMED.

c. Linear buckling and elasto-plastic analysis

These technical extensions will not only be useful for Apollo and future Apollo growth configurations, but they will also be of value to other aeronautical and spacecraft programs.

The hydroelastic response of the flexible shell is obtained by the numerical solution of the combined hydrodynamic and shell equations. The results obtained herein are compared numerically with those derived by neglecting the interaction and applying rigid body pressures to the same elastic shell. The numerical results show that for an axially symmetric impact of the particular shell studied, the interaction between the shell and the fluid produces appreciable differences in the overall acceleration of the center of gravity of the shell, and in the distribution of the pressures and responses. However the maximum responses are within 15% of those produced when the interaction between the fluid and the shell is neglected. A brief summary of results is shown in the abstracts of individual volumes.

The volume number and authors are listed on the following page.

The contractor's designation for this report is SID 67-498.

INDEX FOR FINAL REPORT

"Apollo Water Impact"

Volume No.	Volume Title	Authors
1	Hydrodynamic Analysis of Apollo Water Impact	T. Li and T. Sugimura
2	Dynamic Response of Shells of Revolution During Vertical Impact Into Water - No Interaction	A.P. Cappelli, and J.P.D. Wilkinson
3	Dynamic Response of Shells of Revolution During Vertical Impact Into Water - Hydroelastic Interaction	J.P.D. Wilkinson, A.P. Cappelli, and R.N. Salzman
4	Comparison With Experiments	J.P.D. Wilkinson
5	User's Manual - No Interaction	J.P.D. Wilkinson
6	User's Manual - Interaction	J.P.D. Wilkinson and R.N. Salzman
7	Modification of Shell of Revolution Analysis	A.P. Cappelli and S.C. Furuike
8	Unsymmetric Shell of Revolution Analysis	A.P. Cappelli, T. Nishimoto, P.P. Radkowski and K.E. Pauley
9	Mode Shapes and Natural Frequencies Analysis	A.P. Cappelli
10	User's Manual for Modification of Shell of Revolution Analysis	A.P. Cappelli and S.C. Furuike
11	User's Manual for Unsymmetric Shell of Revolution Analysis	E. Carrión, S.C. Furuike and T. Nishimoto

PRECEDING PAGE BLANK NOT FILMED.

Volume 11

User's Manual
For the Unsymmetric Shell of Revolution Computer
Programs Static and Dynamic

ABSTRACT

The unsymmetric shell of revolution programs described in this volume were developed as a basic tool in the static and dynamic analysis of shells of revolution having arbitrary distribution of stiffness subjected to arbitrary loads and temperature.

The analytical basis for the two computer programs, static and dynamic, is presented. The analysis is restricted to linear-elastic thin shell theory. The basic equations include the effect of shear distortion. The analysis utilizes a Fourier series technique to separate the circumferential variation. The resulting set of equations are reduced to an algebraic set by the use of finite difference forms for variations of the other variables. These algebraic equations are then solved numerically by a direct matrix elimination procedure.

The computer programs were written in FORTRAN IV. In order to keep input data at a minimum extensive use of call function subroutines have been used. Call functions are subroutines which are coded and compiled by the user. This eliminates the necessity of large tabular inputs.

Solutions obtained from the Dynamic Unsymmetric Shell program are the displacement and rotation time histories. Also included in the output are the histories of internal forces and moments.

The solutions obtained from the Static Unsymmetric Shell program include displacements, rotations, internal force and moment resultants, and stresses. This output is presented in tabular form. The users of these programs should be forewarned that the programs are only a tool and insight must be used in relating results to an actual physical problem.

This volume is intended to supply the information necessary for the use of both the Dynamic Unsymmetric Shell program and the Static Unsymmetric Shell program. Detailed descriptions have been included in this volume to aid the user in performing extensions and modifications of these programs.

(ACCESSION NUMBER) (THRU)
320
(PAGES) (CODE)
CR-98163
(NASA CR OR TMX OR AD NUMBER) (CATEGORY)

PRECEDING PAGE BLANK NOT FILMED.

CONTENTS

Section		Page
	FOREWORD	iii
	ABSTRACT	vii
I.	THEORY	1
	1.1 Introduction	1
	1.2 Nomenclature	1
	1.3 Scope and Limitations	3
	1.4 Shell Coordinate System	4
	1.5 Equations of Motion	5
	1.6 Formulation into Solution Variables	8
	1.7 Circumferential Variable Separation	13
	1.8 Finite Difference Formulation in the Meridional Variable	26
	1.9 Finite Difference Formulation in the Time Variable	28
	1.10 Matrix Solution of the Difference Equations	30
	1.11 Static Analysis	32
	1.12 References	33
	1.13 Appendix	33
	1.13.1 Modification of Sanders' Equations	33
	1.13.2 Series Product Expansion	36
	1.13.3 Coefficients	36
II.	GENERAL DESCRIPTION OF THE COMPUTER PROGRAMS	45
	2.1 Introduction	45
	2.2 Program Capabilities and Limitations	45
	2.3 Sign Conventions and Dimensions	46
	2.4 Reference, Inner and Outer Surfaces	46
	2.5 Geometry	47
	2.5.1 Cone-Cylinder	47
	2.5.2 Sphere-Toroid	48
	2.5.3 Discrete Point	49
	2.6 Function Subroutine Parameters	50
	2.6.1 Stiffness Properties	50
	2.6.2 Thermal Loads	51

Section | Page

2.6.3 Elastic Foundations 51
2.6.4 External Damping 51
2.6.5 Mass Parameter 51
2.6.6 Pressure Loads 51
2.6.7 Parameters for Stresses 52
2.7 Stations 52
2.8 Boundary Conditions 53
2.9 Reference Quantities 53
2.10 Output Indicators 54

III. DETAILED USE OF THE PROGRAMS 55
3.1 Introduction 55
3.2 Deck Set-Up 56
3.3 Data Deck Set-Up 59
3.3.1 Title Cards 60
3.3.2 General Shell Data, GDA 60
3.3.3 Geometry Data, GMDA 62
3.4 Function Subprograms 64
3.5 Utility Subroutines 69
3.5.1 DECRD 69
3.5.2 MAD, MSU, MMY, INVMS 72
3.5.3 DINTRP, ENTERP 83
3.5.4 CODIMA 86
3.5.5 FILE 90
3.6 Sample Problem Number 1 95
3.6.1 Problem Description and Set-Up 95
3.6.2 Data Sheets 96
3.6.3 Function Subprograms Used 98
3.6.4 Output 102
3.7 Sample Problem Number 2 114
3.7.1 Problem Description and Set-Up 114
3.7.2 Data Sheets 116
3.7.3 Function Subprograms Used 119
3.8 Sample Problem Number 3 115
3.8.1 Problem Description and Set-Up 125
3.8.2 Data Sheets 126
3.8.3 Function Subprograms Used 128
3.8.4 Output 131
3.9 Sample Problem Number 4 167
3.9.1 Problem Description and Set-Up 167
3.9.2 Data Sheets 169
3.9.3 Function Subprograms Used 171

Section | Page

3.10 Error Indications, Pitfalls, Recommendations . . 176
3.11 Program Listings for Static Version 177
3.12 Program Listings for Dynamic Version; Load Map 230
3.13 Nomenclature 298

PRECEDING PAGE BLANK NOT FILMED.

ILLUSTRATIONS

Figure		Page
1.1	Geometry and Coordinates	4
1.2	Sign Convention	7
1.3	Stiffness Profile	15
2.1	Cone-Cylinder Geometry	48
2.2	Sphere-Toroid Geometry	48
2.3	Discrete Point Geometry	50

1.0 THEORY

1.1 INTRODUCTION

This section presents the analytical basis of the computer programs. The dynamic response problem is presented and the specialization for the static analysis is made.

The analysis is based on a modified form of the general first order linear shell theory of Sanders. These equations have been modified to include transverse shear distortion, see Appendix 1.13.1 The modified equilibrium equations are extended to include time dependence by D'Alembert's principle. Fourier analysis is used to separate variables in the circumferential direction and a system of finite difference approximations are used to reduce the partial differential equations to an algebraic set. This set is solved by using a direct matrix elimination procedure.

The material presented in this section is an extension and parallels of the work of Sanders[1], Budiansky and Radkowski[2], and Johnson and Greif[3]. The notation used is identical to that of Reference 2 except where noted.

1.2 NOMENCLATURE

Fourier Coefficients

u_n, v_n, w_n, $\phi_{\xi n}$, $\phi_{\theta n}$	= displacements and rotations
$p_{\xi n}$, $p_{\theta n}$, p_n	= loads
$t^T_{\xi n}$, $t^T_{\theta n}$, $m^T_{\xi n}$, $m^T_{\theta n}$	= temperature induced force terms
t^n_ξ, t^n_θ, $t^n_{\xi\theta}$, $t^n_{\xi\theta}$, q^n_ξ, q^n_θ, m^n_ξ, m^n_θ, $m^n_{\xi\theta}$	= force resultants, modified and effective
b_{mj}, d_{mj}, g_{mj}, g_{13j}	= stiffnesses
k_{mj}, c_{mj}, m_{mj}	= elastic foundation, damping, mass

$$\left.\begin{array}{l}
f_{5k+1,\,5j+1} \cdots\cdots f_{5k+5,\,5j+5} \\
g_{5k+1,\,5j+1} \cdots\cdots g_{5k+5,\,5j+5} \\
h_{5k+1,\,5j+1} \cdots\cdots h_{5k+5,\,5j+5} \\
k_{5k+1,\,5j+1} \cdots\cdots k_{5k+5,\,5j+5} \\
\alpha_{5k+1,\,5j+1} \cdots\cdots \alpha_{5k+5,\,5j+5} \\
\beta_{5k+1,\,5j+1} \cdots\cdots \beta_{5k+5,\,5j+5} \\
p_{5k+1} \cdots\cdots p_{5k+5}, \ell_1^k \cdots \ell_5^k \\
r_{5k+1,\,5j+1} \cdots\cdots r_{5k+5,\,5j+5} \\
s_{5k+1,\,5j+1} \cdots\cdots s_{5k+5,\,5j+5} \\
a_{5k+1} \cdots\cdots\cdots\cdots a_{5k+5}
\end{array}\right\} = \text{elements of } F,\ G,\ H,\ K,\ \alpha,\ \beta,\ p,\ \ell\ R,\ S, \text{ and } a$$

<u>Coordinates and Constants</u>

$\xi,\ \theta,\ \zeta$	= coordinates
r	= normal distance from shell axis
s	= meridional shell coordinate
$a,\ h_o,\ \sigma_o,\ E_o$	= reference constants
$\omega_\theta,\ \omega_\xi$	= nondimensional curvatures
$u,\ v,\ w,\ \phi_\xi,\ \phi_\theta$	= displacements and rotations
$\sigma_\xi,\ \sigma_\theta,\ \tau_{\xi\theta},\ \tau_{\xi\zeta},\ \tau_{\theta\zeta}$	= stresses
$\epsilon_\xi,\ \epsilon_\theta,\ \epsilon_{\xi\theta},\ k_\xi,\ k_\theta$ $k_{\xi\theta},\ \gamma_{\xi\zeta},\ \gamma_{\theta\zeta}$	= strains
$B_i,\ C_i,\ D_i,\ G_i\ \ K_i,\ M_i$	= isotropic stiffness functions
$E,\ \nu,\ \alpha$	= material properties
$q_\xi,\ q_\theta,\ q$	= loads
T	= temperature change

N_ξ, N_θ, $\overline{N}_{\xi\theta}$, M_ξ, M_θ, $\overline{M}_{\xi\theta}$, Q_ξ, Q_θ, $\hat{N}_{\xi\theta}$, N_ξ^T, N_θ^T, M_ξ^T, M_θ^T	= force resultants, modified, effective, and temperature induced

Matrices

F, G, H, K, α, β, Ω, Λ, R, S	= 5K x 5K order
p, ℓ, x	= 5K x 1 order

Indices

i, j, n	= Dummy
k	= k^{th} Fourier component

1.3 SCOPE AND LIMITATIONS

The shell theory on which these programs are based is restricted to linear, elastic, thin shell theory.

(a) The thickness of the shell at any point is small compared to the other dimensions.

(b) Deformations of the shell are small compared to the smallest radius of curvature.

(c) All material points of the shell deform elastically, obeying Hooke's law for transverse isotropic materials.

(d) The shell is "complete," i.e., its only boundaries are at meridian ends and inner and outer surfaces.

(e) The class of shells considered has a surface of revolution reference surface which is within or in close proximity of the shell walls such that $\int E\zeta d\zeta = 0$.

(f) The parameters of stiffness, e.g., in-plane stiffnesses are permitted to vary in both the meridional and circumferential directions. Implied is that parameters such as thickness, Young's modulus, etc., are permitted to vary in both the meridional and circumferential directions. The stiffness parameter's variation in the circumferential direction is restricted to those with a plane of symmetry.

(g) Arbitrary loads and temperature distributions are permissible. Excluded are problems with thermal distributions such that limitation (e) is not satisfied.

(h) The effects of transverse shear is included.

(i) Instability is not considered.

(j) Distributed mass, elastic foundation, and external damping is included. These distributions have the same symmetry conditions as the stiffness parameters (f).

1.4 SHELL COORDINATE SYSTEM

The geometry of a shell is defined entirely by specifying the form of the reference surface and the thickness of the shell at each point. For convenience, the shell coordinate system and geometrical relations used in Reference 2 will be adopted here. The reference surface is assumed to be a surface of revolution and is selected at a convenient location, within or in proximity to the shell walls. The first fundamental form of the reference surface is defined by

$$d\bar{s}^2 = ds^2 + r^2 d\theta^2 \tag{1.1}$$

where $d\bar{s}$ is a line element on the surface and s and θ denote orthogonal coordinates selected along lines of principal curvature.

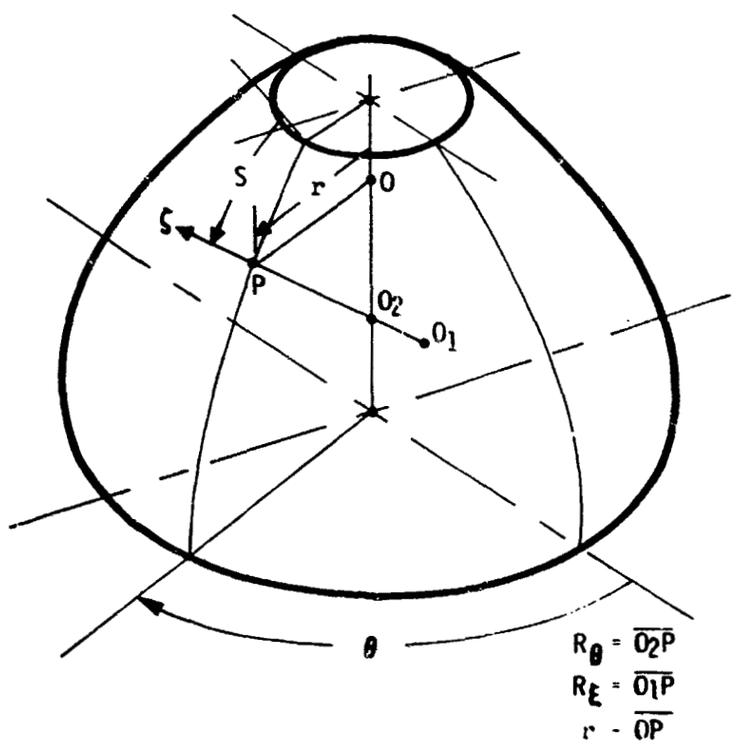


Figure 1.1. Geometry and Coordinates

The generator on the reference surface is defined by r(s) where r is the distance from the axis of revolution. The coordinate s is a measure of the meridional distance along an axisymmetric reference surface, and θ is a circumferential angle, as shown in Figure 1.1. The coordinate, ζ, is selected as a measure of the normal outward distance from the reference surface ($\zeta = 0$). The principal radii of curvature are

$$R_s = -\left[1 - \left(\frac{dr}{ds}\right)^2\right]^{1/2} \Big/ \frac{d^2r}{ds^2} \qquad R_\theta = r\left[1 - \left(\frac{dr}{ds}\right)^2\right]^{-1/2} \tag{1.2}$$

Introduce the nondimensional meridional coordinate $\xi = s/a$, where a is a reference length; then, with $\rho = r/a$, the nondimensional curvatures $\omega_\xi = a/R_s$ and $\omega_\theta = a/R_\theta$ can be found from the formulas

$$\omega_\xi = -(\gamma' + \gamma^2)/\omega_\theta \qquad \omega_\theta = \left[1 - (\rho')^2\right]^{1/2}/\rho \tag{1.3}$$

where

$$\gamma = \rho'/\rho$$

In these equations, and henceforth, $(\ )' = \frac{\partial(\ )}{\partial\xi}$

Finally from the Codazzi relation we obtain

$$\omega_\theta' = \gamma\,(\omega_\xi - \omega_\theta) \tag{1.4}$$

and the relation

$$\frac{\rho''}{\rho} = -\,\omega_\xi\omega_\theta \tag{1.5}$$

1.5 EQUATIONS OF MOTION

The general equilibrium equations for an arbitrary shell based on the first-order linear shell theory of Sanders are given in Reference 1. These equations are modified to include the effect of transverse shear distortion by the procedure suggested by Sanders.* These equilibrium equations are extended to equations of motion by use of D'Alembert's principle. These equations specialized for a shell whose reference surface is a surface of revolution are given as,

*See Appendix

$$a\left[\frac{\partial\rho}{\partial\xi} N_\xi + \rho\frac{\partial N_\xi}{\partial\xi} + \frac{\partial\overline{N}_{\xi\theta}}{\partial\theta} - \frac{\partial\rho}{\partial\xi} N_\theta\right] + a\rho\omega_\xi Q_\xi + \frac{1}{2}(\omega_\xi - \omega_\theta)\frac{\partial\overline{M}_{\xi\theta}}{\partial\theta} + a^2\rho q_\xi = a^2\rho\left[M\frac{\partial^2 U_\xi}{\partial t^2} + C_\xi\frac{\partial U_\xi}{\partial t} + K_\xi U_\xi\right] \tag{1.6.1}$$

$$a\left[\frac{\partial N_\theta}{\partial\theta} + 2\frac{\partial\rho}{\partial\xi}\overline{N}_{\xi\theta} + \rho\frac{\partial\overline{N}_{\xi\theta}}{\partial\xi}\right] + a\rho\omega_\theta Q_\theta + \frac{\rho}{2}\frac{\partial}{\partial\xi}\left[(\omega_\theta - \omega_\xi)\overline{M}_{\xi\theta}\right] + a^2\rho q_\theta = a^2\rho\left[M\frac{\partial^2 U_\theta}{\partial t^2} + C_\theta\frac{\partial U_\theta}{\partial t} + K_\theta U_\theta\right] \tag{1.6.2}$$

$$a\left[\frac{\partial\rho}{\partial\xi} Q_\xi + \rho\frac{\partial Q_\xi}{\partial\xi} + \frac{\partial Q_\theta}{\partial\theta} - \rho(\omega_\xi N_\xi + \omega_\theta N_\theta)\right] + a^2\rho q_\zeta = a^2\rho\left[M\frac{\partial^2 W}{\partial t^2} + C_\zeta\frac{\partial W}{\partial t} + K_\zeta W\right] \tag{1.6.3}$$

$$\frac{\partial\rho}{\partial\xi} M_\xi + \rho\frac{\partial M_\xi}{\partial\xi} + \frac{\partial\overline{M}_{\xi\theta}}{\partial\theta} - \frac{\partial\rho}{\partial\xi} M_\theta - a\rho Q_\xi = a\rho\left[M^*\frac{\partial^2\Phi_\xi}{\partial t^2} + C^*_\xi\frac{\partial\Phi_\xi}{\partial t} + K^*_\xi\Phi_\xi\right] \tag{1.6.4}$$

$$\frac{\partial M_\theta}{\partial\theta} + 2\frac{\partial\rho}{\partial\xi}\overline{M}_{\xi\theta} + \rho\frac{\partial M_{\xi\theta}}{\partial\xi} - a\rho Q_\theta = a\rho\left[M^*_\theta\frac{\partial^2\Phi_\theta}{\partial t} + C^*_\theta\frac{\partial\Phi_\theta}{\partial t} + K^*_\theta\Phi_\theta\right] \tag{1.6.5}$$

where

$M(\xi, \theta) = \rho(\xi, \theta)\, h(\xi, \theta)$ (mass/unit area of shell)

$M(\xi, \theta)^* = \dfrac{\rho(\xi, \theta)\, h^3(\xi, \theta)}{12}$ (mass moment/unit area of shell)

$C_{(\ )}(\xi, \theta)$ = external damping coefficient

$K_{(\ )}(\xi, \theta)$ = spring constants

Where the components of membrane force, transverse force and moment (about the reference surface) per unit length, and load per unit area (assumed to be applied at the reference surface) are shown in Figure 1-2.

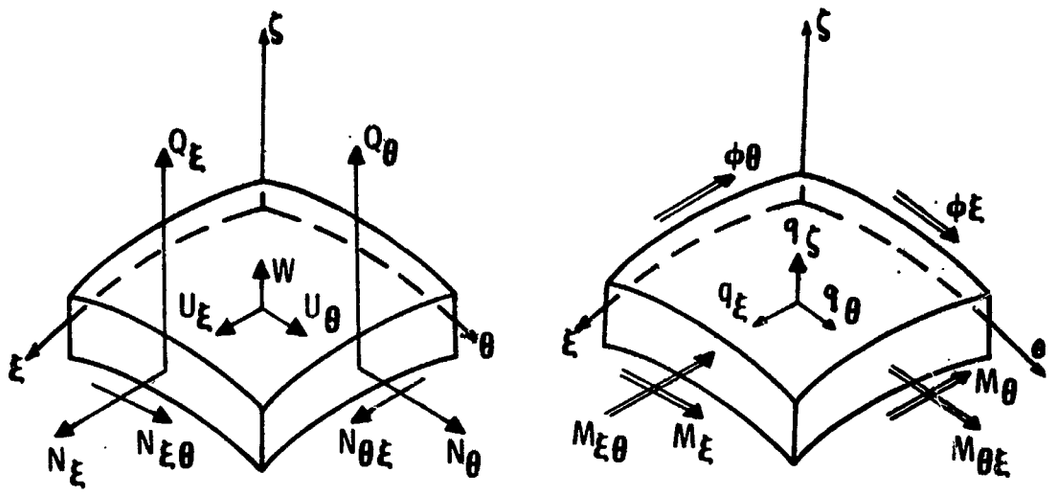


Figure 1-2. Sign Convention and Coordinates, Moments, Forces, Loads, Displacements, and Rotations

In the Sanders' first-order theory the inplane shearing forces $N_{\xi\theta}$ and $N_{\theta\xi}$ as well as twisting moments $M_{\xi\theta}$ and $M_{\theta\xi}$ are not handled separately, but instead are combined to form modified variables

$$\overline{N}_{\xi\theta} = \frac{1}{2}(N_{\xi\theta} + N_{\theta\xi}) \tag{1.7}$$

$$\overline{M}_{\xi\theta} = \frac{1}{2}(M_{\xi\theta} + M_{\theta\xi}) \tag{1.8}$$

and

$$\overline{k}_{\xi\theta} = \frac{1}{2}(k_{\xi\theta} + k_{\theta\xi}) \tag{1.9}$$

It is necessary when including the effects of transverse shear distortion to consider five equilibrium equations. Recall in Reference 2 that when shear deformation is neglected the shear forces are eliminated and resulting equilibrium equations are reduced to the consideration of three equations. The neglecting of transverse shear strains implies that normals to middle surface of the shell remain normal after deformation. The degree of error introduced by this assumption naturally depends on the magnitude of transverse shearing forces and shear rigidity of the shell. For discontinuous loads and shells having low shear rigidity, shear deformations may be comparable to bending and axial deformations and cannot be ignored.

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR;

1.6 FORMULATION INTO SOLUTION VARIABLES

The equations of motion are now expressed in terms of the solution variables, displacements and rotations.

The force and moment expressions in the equations of motion are determined by evaluating the following integrals through the thickness

$$
\begin{aligned}
N_\xi &= \int \sigma_\xi d\zeta & M_\xi &= \int \sigma_\xi \zeta d\zeta \\
N_\theta &= \int \sigma_\theta d\zeta & M_\theta &= \int \sigma_\theta \zeta d\zeta \\
\overline{N}_{\xi\theta} &= \int \tau_{\xi\theta} d\zeta & \overline{M}_{\xi\theta} & \int \tau_{\xi\theta} \zeta d\zeta \\
Q_\xi &= \int \tau_{\xi\zeta} d\zeta & Q_\theta &= \int \tau_{\theta\zeta} d\zeta
\end{aligned}
\tag{1.10}
$$

where in the above integrals we have neglected terms of order ζ/R, R is the minimum radius of curvature. The stresses used above are defined as:

σ_ξ, σ_θ	are normal stresses, acting on the faces
$\tau_{\xi\theta}$	is an in-plane shear stress acting parallel to the reference surface
$\tau_{\xi\zeta}$, $\tau_{\theta\zeta}$	are transverse shear stresses acting normal to the reference surface

By assuming that plane sections before remain plane after deformation, the strains at a distance ζ from the reference surface can be expressed in terms of the reference surface strains as follows:

$$
\begin{aligned}
\epsilon_\xi(\zeta) &= \epsilon_\xi + \zeta k_\xi \\
\epsilon_\theta(\zeta) &= \epsilon_\theta + \zeta k_\theta \\
\epsilon_{\xi\theta}(\zeta) &= \epsilon_{\xi\theta} + \zeta \bar{k}_{\xi\theta}
\end{aligned}
\tag{1.11}
$$

where ϵ_ξ, ϵ_θ and $\epsilon_{\xi\theta}$ are the strains of the reference surface and $\epsilon_{\xi\theta}(\zeta)$ is one-half the usual engineering strain.

The stress-strain-temperature relations for an isotropic material are

$$\sigma_\xi = \frac{E}{1 - \nu^2} \left| \epsilon_\xi + \nu\epsilon_\theta + \zeta (k_\xi + \nu k_\theta) - \alpha (1 + \nu) T \right|$$

$$\sigma_\theta = \frac{E}{1 - \nu^2} \left| \epsilon_\theta + \nu\epsilon_\xi + \zeta (k_\theta + \nu k_\xi) - \alpha (1 + \nu) T \right|$$

$$\tau_{\xi\theta} = G (\epsilon_{\xi\theta} + \zeta \bar{k}_{\xi\theta}) \tag{1.12}$$

$$\tau_{\xi\zeta} = G (\gamma_{\xi\zeta})$$

$$\tau_{\theta\zeta} = G (\gamma_{\theta\zeta})$$

where

$$G = \frac{E}{2(1 + \nu)}$$

Substituting these equations into Eqs. 1.10 and employing appropriate integrations through the thickness yield the following stress/resultants-strain relationships.

$$N_\xi = B_1 \epsilon_\xi + B_3 \epsilon_\theta + C_1 k_\xi + C_3 k_\theta - N_\xi^T$$

$$N_\theta = B_3 \epsilon_\xi + B_2 \epsilon_\theta + C_3 k_\xi + C_2 k_\theta - N_\theta^T$$

$$\bar{N}_{\xi\theta} = G_1 \epsilon_{\xi\theta} + G_{12} k_{\xi\theta}$$

$$Q_\xi = G_2 \gamma_{\xi\zeta} \tag{1.13}$$

$$M_\xi = C_1 \epsilon_\xi + C_3 \epsilon_\theta + D_1 k_\xi + D_3 k_\theta - M_\xi^T$$

$$M_\theta = C_3 \epsilon_\xi + C_2 \epsilon_\theta + D_3 k_\xi + D_2 k_\theta - M_\theta^T$$

$$\bar{M}_{\xi\theta} = G_{12} \epsilon_{\xi\theta} + G_{13} k_{\xi\theta}$$

$$Q_\theta = G_3 \gamma_{\theta\zeta}$$

where in the above equations the shell stiffnesses are given by

$$B_1 = B_2 = \int \frac{E}{1-\nu^2} d\zeta, \; B_3 = \int \frac{\nu E}{1-\nu^2} d\zeta, \; C_1 = C_2 = \int \frac{E\zeta}{1-\nu^2} d\zeta, \; C_3 = \int \frac{\nu E\zeta}{1-\nu^2} d\zeta$$

$$D_1 = D_2 = \int \frac{E}{1-\nu^2} \zeta^2 d\zeta, \; D_3 = \int \frac{\nu E\zeta^2}{1-\nu^2} d\zeta$$

$$G_1 = G_2 = G_3 = \int G d\zeta \tag{1.14}$$

$$G_{12} = \int G\zeta d\zeta$$

$$G_{13} = \int G\zeta^2 d\zeta$$

and thermal loads are

$$N_\xi^T = N_\theta^T = \int \frac{\alpha\,(1+\nu)}{1-\nu^2} ET d\zeta$$

$$M_\xi^T = M_\theta^T = \int \frac{\alpha\,(1+\nu)}{1-\nu^2} ET\zeta d\zeta \tag{1.15}$$

For the case of discontinuous material properties through the thickness the integration is taken piece-wise through the thickness. An additional assumption of constant Poisson's ratios through the thickness is made. Therefore the isotropic nature of the layers are retained in the stiffness parameters. The shell stiffness and thermal loads take the form,

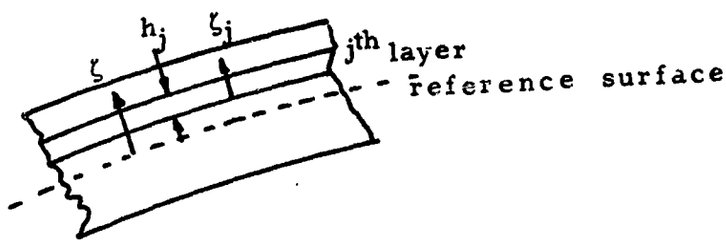
$$B_1 = B_2 = \frac{B_3}{\nu} = \frac{1}{1-\nu^2} \int E d\zeta, \; D_1 = D_2 = \frac{D_3}{\nu} = \frac{1}{1-\nu^2} \int E\zeta^2 d\zeta$$

$$C_1 = C_2 = \frac{C_3}{\nu} = \frac{1}{1-\nu^2} \int E\zeta d\zeta, \; G_1 = G_2 = G_3 = B_1 \frac{(1-\nu)}{2} \tag{1.16}$$

$$G_{12} = C_1 \frac{(1-\nu)}{2}, G_{13} = D_1 \frac{(1-\nu)}{2}$$

It follows that because of the reference surface choice, namely that the condition $\int E\zeta d\zeta = 0$ is satisfied, the integration takes the form

$$A = \int F_{(\zeta)} d\zeta = \sum_j \int_o^{h_j} F_j\,(\zeta_j)\, d\zeta_j \tag{1.17}$$



The reference surface strains and bending distortion terms may be defined in terms of displacements and rotations by the following expressions. The membrane strains of the reference surface are given by

$$\begin{aligned} \epsilon_\xi &= \frac{1}{a}\left[\frac{\partial U_\xi}{\partial \xi} + \omega_\xi W\right] \\ \epsilon_\theta &= \frac{1}{a}\left[\frac{1}{\rho}\frac{\partial U_\theta}{\partial \theta} + \gamma U_\xi + \omega_\theta W\right] \\ \epsilon_{\xi\theta} &= \frac{1}{2a}\left[\frac{1}{\rho}\frac{\partial U_\xi}{\partial \theta} + \frac{\partial U_\theta}{\partial \xi} - \gamma U_\theta\right] \end{aligned} \tag{1.18}$$

where U, V, W are displacements in the ξ, θ and ζ directions respectively. Transverse shear strains are given by

$$\begin{aligned} \gamma_{\xi\zeta} &= \Phi_\xi - \frac{1}{a}\left[\omega_\xi U_\xi - \frac{\partial W}{\partial \xi}\right] \\ \gamma_{\theta\zeta} &= \Phi_\theta - \frac{1}{a}\left[\omega_\theta U_\theta - \frac{1}{\rho}\frac{\partial W}{\partial \theta}\right] \end{aligned} \tag{1.19}$$

where Φ_ξ, Φ_θ are rotations.

The bending distortion terms are given by

$$k_\xi = \frac{1}{a}\frac{\partial \Phi_\xi}{\partial \xi}$$

$$k_\theta = \frac{1}{a}\left[\frac{1}{\rho}\frac{\partial \Phi_\theta}{\partial \theta} + \gamma \Phi_\xi\right] \tag{1.20}$$

$$\bar{k}_{\xi\theta} = \frac{1}{2a}\left[\frac{1}{\rho}\frac{\partial \Phi_\xi}{\partial \theta} + \frac{\partial \Phi_\theta}{\partial \xi} - \gamma \Phi_\theta + \frac{1}{2a}(\omega_\xi - \omega_\theta)\left(\frac{1}{\rho}\frac{\partial U_\xi}{\partial \theta} - \frac{\partial U_\theta}{\partial \xi} - \gamma U_\theta\right)\right]$$

Substituting equations (1.18, 1.19, 1.20) into equation 1.13 the force terms in the equations of motion can be expressed in terms of the displacements

$$N_\xi = \frac{1}{a}\left\{B_1 \frac{\partial U_\xi}{\partial \xi} + B_3 \gamma U_\xi + B_3\left(\frac{1}{\rho}\right)\frac{\partial U_\theta}{\partial \theta} + (B_1 \omega_\xi + B_2 \omega_\theta) W\right\} - N^T$$

$$N_\theta = \frac{1}{a}\left\{B_3 \frac{\partial U_\xi}{\partial \xi} + B_2 \gamma U_\xi + B_2\left(\frac{1}{\rho}\right)\frac{\partial U_\theta}{\partial \theta} + (B_3 \omega_\xi + B_2 \omega_\theta) W\right\} - N^T$$

$$N_{\xi\theta} = \frac{1}{2a}\left\{\frac{1}{\rho} G_1 \frac{\partial U_\xi}{\partial \theta} + G_1 \frac{\partial U_\theta}{\partial \xi} - \gamma G_1 U_\theta\right\}$$

$$Q_\xi = G_2\left\{-\frac{1}{a}\omega_\xi U_\xi + \frac{1}{a}\frac{\partial W}{\partial \xi} + \Phi_\xi\right\}$$

$$M_\xi = \frac{1}{a}\left\{D_1 \frac{\partial \Phi_\xi}{\partial \xi} + D_3 \gamma \Phi_\xi + D_3 \frac{1}{\rho}\frac{\partial \Phi_\theta}{\partial \theta}\right\} - M^T \tag{1.21}$$

$$M_\theta = \frac{1}{a}\left\{D_3 \frac{\partial \Phi_\xi}{\partial \xi} + D_2 \gamma \Phi_\xi + D_2 \frac{1}{\rho}\frac{\partial \Phi_\theta}{\partial \theta}\right\} - M^T$$

$$\overline{M}_{\xi\theta} = \frac{1}{2a}\left\{\frac{1}{2a} G_{13}(\omega_\xi - \omega_\theta)\left(\frac{1}{\rho}\frac{\partial U_\xi}{\partial \theta} - \frac{\partial U_\theta}{\partial \xi} - \gamma U_\theta\right) + G_{13}\left(\frac{1}{\rho}\frac{\partial \Phi_\xi}{\partial \theta} + \frac{\partial \Phi_\theta}{\partial \xi} - \gamma \Phi_\theta\right)\right\}$$

$$Q_\theta = G_3\left\{-\frac{1}{a}\omega_\theta U_\theta + \frac{1}{a\rho}\frac{\partial W}{\partial \theta} + \Phi_\theta\right\}$$

By employing the relations, equations 1.21 the equations of motion can be expressed in terms of the dependent variables, displacements and rotations.

1.7 CIRCUMFERENTIAL VARIABLE SEPARATION

The analysis utilizes a Fourier approach which will permit separation of variables and yield a more tractable set of shell equations. The procedure involves expanding of the pertinent variables in Fourier series with appropriate normalization to provide nondimensional Fourier coefficients of roughly comparable magnitudes for different variables. Letting σ_o, E_o, h_o be a reference stress level, Young's modulus, and thickness, respectively, solutions for the field equations are sought in the form

$$
\begin{aligned}
U_\xi &= \frac{a\sigma_o}{E_o} \sum_{n=o}^{\infty} u_n(\xi, t) \operatorname{Cos} n\theta \\
U_\theta &= \frac{a\sigma_o}{E_o} \sum_{n=1}^{\infty} v_n(\xi, t) \operatorname{Sin} n\theta \\
W &= \frac{a\sigma_o}{E_o} \sum_{n=0}^{\infty} w_n(\xi, t) \operatorname{Cos} n\theta \\
\Phi_\xi &= \frac{\sigma_o}{E_o} \sum_{n=0}^{\infty} \phi_{\xi n}(\xi, t) \operatorname{Cos} n\theta \\
\Phi_\theta &= \frac{\sigma_o}{E_o} \sum_{n=1}^{\infty} \phi_{\theta n}(\xi, t) \operatorname{Sin} n\theta
\end{aligned}
\tag{1.22}
$$

These Fourier expansions are consistent with loadings of form

$$
\begin{aligned}
q_\xi &= \frac{\sigma_o h_o}{a} \sum_{n=0}^{\infty} p_{\xi n}(\xi, t) \operatorname{Cos} n\theta \\
q_\theta &= \frac{\sigma_o h_o}{a} \sum_{n=1}^{\infty} p_{\theta n}(\xi, t) \operatorname{Sin} n\theta \\
q_\zeta &= \frac{\sigma_o h_o}{a} \sum_{n=0}^{\infty} p_{\zeta n}(\xi, t) \operatorname{Cos} n\theta
\end{aligned}
\tag{1.23}
$$

The above Fourier expansions are not the most general form. The expansions q_ζ, q_ξ are symmetrical expansions about $\theta = 0$. For full generality, they must be augmented by additional sine series expansions. The form q_θ in turn would be supplemented by cosine series. Consistently, the series expansions for displacements and rotations must be augmented for the general case.

Expansions for the temperature distributions may be described in a similar manner; however, since the thermal coefficients and Young's modulus can vary in the circumferential direction, it will be more convenient to expand the thermal load in Fourier series as follows

$$N^T = \sigma_o h_o \sum_{n=0}^{\infty} t_n^T \text{ Cos } n\theta$$

$$M^T = \frac{\sigma_o h_o^3}{a} \sum_{n=0}^{\infty} m_n^T \text{ Cos } n\theta \tag{1.24}$$

Where the Fourier components t_n^T and m_n^T are given by

$$t_n^T = \frac{2}{\pi} \int_o^{\pi} \frac{N^T}{\sigma_o h_o} \text{ Cos } n\theta \, d\theta$$

$$m_n^T = \frac{2}{\pi} \int_o^{\pi} \frac{aM^T}{\sigma_o h_o^3} \text{ Cos } n\theta \, d\theta$$

Since the stiffness mass, external damping and elastic foundation parameters are variable in the circumferential directions, these will also be expanded in a Fourier series. For example, the expansion for the extensional stiffness parameter is of the form

$$B = \sum_{j=0}^{\infty} b_j \text{ Cos } j\theta + \sum_{j=1}^{\infty} b_j \text{ Sin } j\theta \tag{1.25}$$

In many problems, there exists a plane of symmetry with respect to planform geometry. See Figure 1-3.

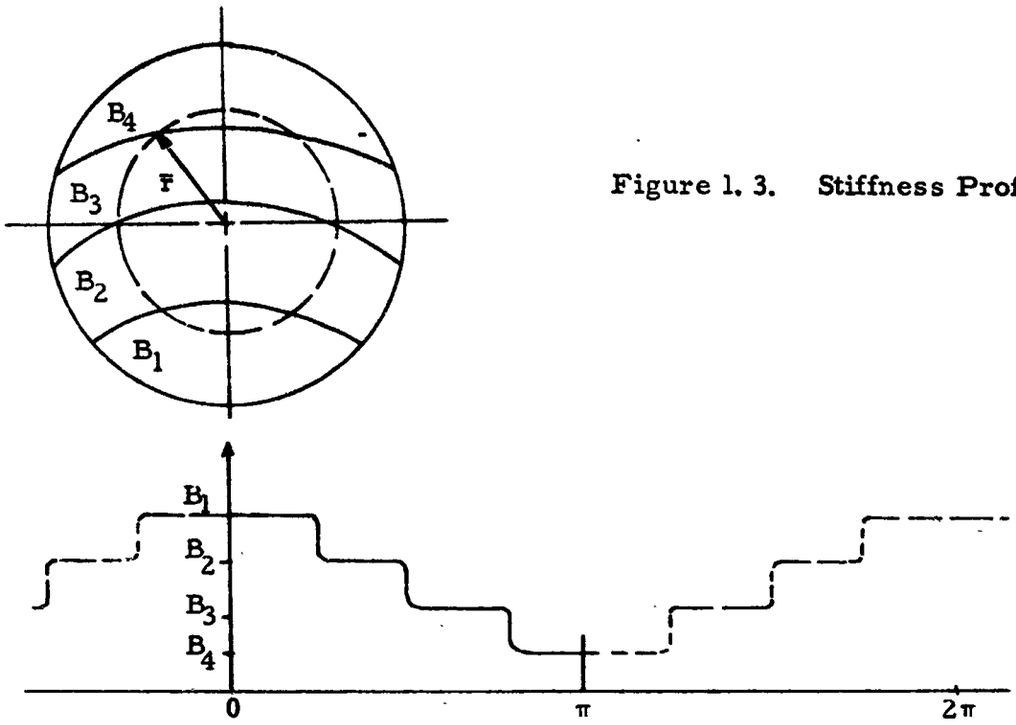


Figure 1.3. Stiffness Profile

A plane of symmetry is assumed here. The coefficients of B, viz., b_j are found by integrations of the form

$$b_j = \frac{2}{\pi} \int_0^{\pi} B(\xi, \theta) \cos j\theta d\theta \tag{1.26}$$

The Fourier expansions of the shell stiffness parameters (Eqs. 1.14), consistent with previous formulation are given by

$$B_m = E_o h_o \sum_{j=0}^{\infty} b_{mj}(\xi) \cos j\theta \qquad \text{Inplane Stiffness}$$

$$D_m = E_o h_o^3 \sum_{j=0}^{\infty} d_{mj}(\xi) \cos j\theta \qquad \text{Bending Stiffness} \tag{1.27}$$

$$C_m = \sum_{j=0}^{\infty} c_{mj}(\xi) \cos j\theta \qquad \text{External Damping}$$

$$K_m = \sum_{j=0}^{\infty} k_{mj}(\xi) \cos j\theta \qquad \text{Elastic Foundation}$$

$$M_m = \sum_{j=0}^{\infty} m_{mj}(\xi) \cos j\theta \qquad \text{Mass Property} \tag{1.27}$$

Substitution of the displacement and rotation series expansions (Eqs. 1.22) and the above stiffness expansions into Eqs. (1.21) and employing the proper trigonometric identities yields the following series expressions relating forces (moments) in terms of the Fourier coefficients of the displacement variables and stiffness parameters:

$$N_\xi = \sigma_o h_o \sum_{k=0}^{\infty} \left[\sum_{j=0}^{\infty} \left\{ B_1^{kj} \left(\frac{\partial u_j}{\partial \xi} + \omega_\xi w_j \right) + B_3^{kj} \left(\gamma u_j + \frac{k}{\rho} v_j + \omega_\theta w_j \right) \right\} - t_k^T \right] \cos k\theta$$

$$N_\theta = \sigma_o h_o \sum_{k=0}^{\infty} \left[\sum_{j=0}^{\infty} \left\{ B_3^{kj} \left(\frac{\partial u_j}{\partial \xi} + \omega_\xi w_j \right) + B_2^{kj} \left(\gamma u_j + \frac{k}{\rho} v_j + \omega_\theta w_j \right) \right\} - t_k^T \right] \cos k\theta$$

$$\overline{N}_{\xi\theta} = \sigma_o h_o \sum_{k=1}^{\infty} \left[\sum_{j=0}^{\infty} \frac{G_1^{kj}}{2} \left(\frac{\partial v_j}{\partial \xi} - \frac{k}{\rho} u_j - \gamma v_j \right) \right] \sin k\theta \tag{1.28}$$

$$Q_\xi = \sigma_o h_o \sum_{k=0}^{\infty} \left[\sum_{j=0}^{\infty} G_2^{kj} \left(- \omega_\xi u_j + \frac{\partial w_j}{\partial \xi} + \phi_{\xi j} \right) \right] \cos k\theta$$

$$M_\xi = \frac{\sigma_o h_o^3}{a} \sum_{k=0}^{\infty} \left[\sum_{j=0}^{\infty} \left\{ D_1^{kj} \frac{\partial \phi_{\xi j}}{\partial \xi} + D_3^{kj} \left(\gamma \phi_{\xi j} + \frac{k}{\rho} \phi_{\theta j} \right) \right\} - m^T \right] \cos k\theta$$

$$M_\theta = \frac{\sigma_o h_o^3}{a} \sum_{k=0}^{\infty} \left[\sum_{j=0}^{\infty} \left\{ D_3^{kj} \frac{\partial \phi_{\xi j}}{\partial \xi} + D_2^{kj} \left(\gamma \phi_{\xi j} + \frac{k}{\rho} \phi_{\theta j} \right) \right\} - m^T \right] \cos k\theta$$

$$\overline{M}_{\xi\theta} = \frac{\sigma_o h_o^3}{a} \sum_{k=1}^{\infty} \left[\sum_{j=0}^{\infty} \frac{G_{13}^{kj}}{2} \left(- \frac{k}{\rho} \phi_{\xi_j} + \frac{\partial \phi_{\theta j}}{\partial \xi} - \gamma \phi_{\theta_j} + \frac{1}{2} (\omega_\xi - \omega_\theta) \right) \right.$$

$$\left. \left(- \frac{k}{\rho} u_j - \frac{\partial v_j}{\partial \xi} - \gamma v_j \right) \right] \text{Sin } k\theta \qquad (1.28)$$

$$Q_\theta = \sigma_o h_o \sum_{n=1}^{\infty} \left[\sum_{j=0}^{\infty} \left\{ C_3^{kj} \left(- \omega_\theta v_j - \frac{k}{\rho} w_j + \phi_{\theta j} \right) \right\} \right] \text{Sin } k\theta$$

The translational inertial force terms are expressed as

$$M_\xi \ddot{U} = \sum_{k=0}^{\infty} \left[\sum_{j=0}^{\infty} M_1^{kj} \ddot{u}_j \right] \text{Cos } k\theta$$

$$M_\theta \ddot{V} = \sum_{k=1}^{\infty} \left[\sum_{j=0}^{\infty} M_2^{kj} \ddot{v}_j \right] \text{Sin } k\theta \qquad (1.29)$$

$$M_\zeta \ddot{W} = \sum_{k=0}^{\infty} \left[\sum_{j=0}^{\infty} M_3^{kj} \ddot{w}_j \right] \text{Cos } k\theta$$

The translational damping force is expressible as

$$C_{D_\xi} \dot{U} = \sum_{k=0}^{\infty} \left(\sum_{j=0}^{\infty} C_1^{kj} \dot{u}_j \right) \text{Cos } k\theta$$

$$C_{D_\theta} \dot{V} = \sum_{k=1}^{\infty} \left(\sum_{j=0}^{\infty} C_2^{kj} \dot{v}_j \right) \text{Sin } k\theta \qquad (1.30)$$

$$C_{D_\zeta} \dot{W} = \sum_{k=0}^{\infty} \left(\sum_{j=0}^{\infty} C_3^{kj} \dot{w}_j \right) \text{Cos } k\theta$$

The translational elastic foundation force is expressible as

$$K_\xi U = \sum_{k=0}^{\infty}\left(\sum_{j=0}^{\infty} k_1^{kj}\, u_j\right) \text{Cos}\, k\theta$$

$$K_\theta V = \sum_{k=1}^{\infty}\left(\sum_{j=0}^{\infty} k_2^{kj}\, v_j\right) \text{Sin}\, k\theta \qquad (1.31)$$

$$K_\zeta W = \sum_{k=0}^{\infty}\left(\sum_{j=0}^{\infty} k_3^{kj}\, w_j\right) \text{Cos}\, k\theta$$

The stiffness recursion relationships above are described in the form

$$A^{kj} = \frac{1}{2}\left\{ a_{(k+j)} + \left[1 - \delta^2 (j - n) + \delta\,(k)\right] a_{|k-j|} \right\}$$

$$\bar{A}^{kj} = \frac{1}{2}\left\{ - \bar{a}_{(k+j)} + \left[1 - \delta^2 (j - k) + \delta(k)\right] \bar{a}_{|k-j|} \right\} \qquad (1.32)$$

where the specific coefficients of interest (dropping k_j superscript) are given by

$$A = \left| B_1,\ B_2,\ B_3,\ D_1,\ D_2,\ D_3,\ g_2,\ M_1,\ M_3,\ C_1,\ C_3,\ K_1,\ K_3 \right|$$

$$a = \left| b_1,\ b_2,\ b_3,\ d_1,\ d_2,\ d_3,\ g_2,\ m_1,\ m_3,\ c_1,\ c_3,\ k_1,\ k_3 \right|$$

and

$$\bar{A} = \left| G_1,\ G_3,\ G_{13},\ M_2,\ C_2,\ K_2 \right|$$

$$\bar{a} = \left| g_1,\ g_3,\ g_{13},\ m_2,\ c_2,\ k_2 \right|$$

In the above expressions $\delta\,(m)$ is defined as

$$\delta\,(m) = \begin{cases} -1, & m < 0 \\ 0, & m = 0 \\ +1, & m > 0 \end{cases}$$

(*See Appendix 1.2 (multiplication of series expressions) for a more detailed description of A^{kj} and $\bar{A}^{kj}$.)

Substitution of the stress resultant expressions and dynamic force resultants (Eqs. 1.28 - 1.31) into the shell equilibrium equations (Eqs. 1.6) yields five finite series expressions in the circumferential coordinate relating the Fourier coefficients u_j, v_j, w_j, $\phi_{\xi j}$ and $\phi_{\theta j}$ of the displacement and rotation variables. For practical considerations we truncate the series solution of the dependent variables to K terms in the Fourier component. Employing the appropriate orthogonality relationships of Fourier series to these equilibrium expressions yields a system of 5K ordinary differential equations relating the 5K unknown Fourier coefficients. These equations are presented in a form amenable for computer programming and are given as follows:

$$
\begin{aligned}
\sum_{j=0}^{K-1} \Big[& f_{5k+\ell,\,5j+1}\, u_j'' + \\
& f_{5k+\ell,\,5j+2}\, v_j'' + f_{5k+\ell,\,5j+3}\, w_j'' + \\
& f_{5k+\ell,\,5j+4}\, \phi_{\xi j}'' + f_{5k+\ell,\,5j+5}\, \phi_{\theta j}'' + \\
& g_{5k+\ell,\,5j+1}\, u_j' + \cdots + g_{5k+\ell,\,5j+5}\, \phi_{\theta j} + \\
& h_{5k+\ell,\,5j+1}\, u_j + \cdots + h_{5k+\ell,\,5j+5}\, \phi_{\theta j} + \\
& k_{5k+\ell,\,5j+1}\, u_j + \cdots + k_{5k+\ell,\,5j+1}\, \phi_{\theta j} = \alpha_{5k+\ell,\,5j+1}\, \ddot{u}_j + \cdots + \\
& \alpha_{5k+\ell,\,5j+5}\, \ddot{\phi}_{\theta j} + \\
& \beta_{5k+\ell,\,5j+1}\, \dot{u}_j + \cdots + \\
& \beta_{5k+\ell,\,5j+5}\, \dot{\phi}_{\theta j} + p_{5k+\ell}
\end{aligned}
\tag{1.33}
$$

where

$\ell = 1, 2, 3, 4, 5$

$k, j = 0, \ldots, K - 1$

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

where the f, g, h, k, α and β coefficients are described in Appendix 1-B. (It should be noted that the form of the above equations is more complicated than was obtained in Reference 2 for analysis of shells of revolution. This complexity arises from the fact that the equilibrium equations cannot be decoupled for each Fourier component of displacement variables for the case of unsymmetric shell.)

The above equation can be conveniently written in matrix form as follows:

$$Fz'' + Gz' + (H + K)\, z = \alpha \ddot{z} + \beta \dot{z} + p \tag{1.34}$$

where F, G, H, K, α, β are square arrays and z, p are column arrays. These arrays are defined as,

$$F = \begin{bmatrix} f_{11}, & f_{12} & \cdot & \cdot & f_{1,5k} \\ f_{21} & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ f_{5k,1} & \cdot & \cdot & \cdot & f_{5k,5k} \end{bmatrix} \qquad G = \begin{bmatrix} g_{11} & g_{12} & \cdot & \cdot & \cdot \\ g_{21} & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & & \cdot & \cdot & \cdot \\ \cdot & & \cdot & \cdot & \cdot \end{bmatrix}$$

$$H = \begin{bmatrix} h_{11}, & h_{12} & \cdot & \cdot & \cdot \\ h_{21} & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix} \qquad K = \begin{bmatrix} k_{11}, & k_{12} & \cdot & \cdot & \cdot \\ k_{21} & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ k_{5k,1} & \cdot & \cdot & \cdot & \cdot \end{bmatrix}$$

$$\alpha = \begin{bmatrix} \alpha_{11}, & \alpha_{12} & \cdot & \cdot & \cdot \\ \alpha_{21} & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix} \qquad \beta \begin{bmatrix} \beta_{11}, & \beta_{12} & \cdot & \cdot & \cdot \\ \beta_{21} & \cdot & \cdot & \cdot & \cdot \\ \cdot & & \cdot & \cdot & \cdot \\ \cdot & & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix}$$

$$z = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \cdot \\ z_k \end{bmatrix} \qquad z_i = \begin{bmatrix} u_i \\ v_i \\ w_i \\ \phi_{\xi_i} \\ \phi_{\theta_i} \end{bmatrix} \qquad p = \begin{bmatrix} p_1 \\ p_2 \\ \cdot \\ \cdot \\ p_k \end{bmatrix} \qquad p_i = \begin{bmatrix} p_1^i \\ p_2^i \\ p_3^i \\ p_4^i \\ p_5^i \end{bmatrix}$$

The elements of the F, G and H, K, α, β, and p matrices are given in Appendix 1.13.1 and are presented there in a format which is designed to ease computer programming. The coefficients of P are the Fourier components of the applied external load and are known quantities for a specific loading case.

The more general case of shells having arbitrary distribution of stiffness, loads, damping, and elastic foundation can be considered in the same manner as the case of plane of symmetry of stiffness and loads. In this case the total Fourier series representation of all the variables, displacements, rotations, stiffness and loads, must be carried in the analysis. The analysis will follow the same format of the special case formulated previously.

Boundary Conditions

Consistent with Sanders' equilibrium equations, the boundary conditions for the specification of the forces or displacements, or constraint between them are described below. On the edge where ξ = constant (i.e., $\xi = 0$, and $\xi = \bar{s}$)

$$\begin{aligned} N_\xi &\text{ or } U_\xi \\ \hat{N}_{\xi\theta} &\text{ or } U_\theta \\ Q_\xi &\text{ or } W \\ M_\xi &\text{ or } \Phi_\xi \\ \bar{M}_{\xi\theta} &\text{ or } \Phi_\theta \end{aligned} \tag{1.35}$$

where

$$\hat{N}_{\xi\theta} = \bar{N}_{\xi\theta} + \frac{1}{2a}(\omega_\theta - \omega_\xi)\,\bar{M}_{\xi\theta}$$

These conditions can be expressed in matrix form by,

$$\bar{\Omega}\bar{y} + \bar{\Lambda}\bar{z} = \bar{\ell} \tag{1.36}$$

where $\bar{y}$, $\bar{\ell}$, $\bar{z}$ are column matrices and $\bar{\Omega}$, $\bar{\Lambda}$ are appropriate diagonal matrices

$$\bar{y} = \begin{bmatrix} N_\xi \\ \hat{N}_{\xi\theta} \\ Q_\xi \\ M_\xi \\ M_{\xi\theta} \end{bmatrix} \qquad \bar{z} = \begin{bmatrix} U_\xi \\ U_\theta \\ W \\ \Phi_\xi \\ \Phi_\theta \end{bmatrix} \qquad \bar{\ell} = \begin{bmatrix} \ell_1 \\ \ell_2 \\ \ell_3 \\ \ell_4 \\ \ell_5 \end{bmatrix}$$

$$\bar{\Omega} = \begin{bmatrix} \omega_1 & & & & \\ & \omega_2 & & 0 & \\ & & \omega_3 & & \\ & 0 & & \omega_4 & \\ & & & & \omega_5 \end{bmatrix} \qquad \bar{\Lambda} = \begin{bmatrix} \lambda_1 & & & & \\ & \lambda_2 & & 0 & \\ & & \lambda_3 & & \\ & 0 & & \lambda_4 & \\ & & & & \lambda_5 \end{bmatrix}$$

The logic which connects $\bar{\Omega}$, $\bar{\Lambda}$, $\bar{\ell}$ and the conditions desired are given in the following table:

Matrix Elements

Condition at Boundary	ω_i	λ_i	ℓ_i	
Displacement Prescribed	0	1	C_1	C_1 = value of displacement
Force Prescribed	1	0	C_2	C_2 = value of force
Constraint Condition	1	C_3	0	C_3 = constant relating force and displacement

For example, is Φ_ξ is given as a boundary condition then $\lambda_4 = 1$, $\omega_4 = 0$ and ℓ_4 is the prescribed value of Φ_ξ. Note C_i is nondimensionalized with the appropriate reference constants.

It will be convenient to expand forces and moments in Fourier series in manner consistent with the previous developments. Letting

$$
\begin{aligned}
N_\xi &= \sum t_\xi^n \text{ Cos } n\theta \\
\hat{N}_{\xi\theta} &= \sum \hat{t}_{\xi\theta}^n \text{ Sin } n\theta \\
Q_\xi &= \sum q_\xi^n \text{ Cos } n\theta \\
M_\xi &= \sum m_\xi^n \text{ Cos } n\theta \\
\overline{M}_{\xi\theta} &= \sum m_{\xi\theta}^n \text{ Sin } n\theta
\end{aligned}
\tag{1.37}
$$

and

$$
\begin{aligned}
\ell_1 &= \sum \ell_1^n \text{ Cos } n\theta \\
\ell_2 &= \sum \ell_2^n \text{ Sin } n\theta \\
\ell_3 &= \sum \ell_3^n \text{ Cos } n\theta \\
\ell_4 &= \sum \ell_4^n \text{ Cos } n\theta \\
\ell_5 &= \sum \ell_5^n \text{ Sin } n\theta
\end{aligned}
$$

The above series expressions together with Eqs. (1.21) are substituted into Eq. (1.36) and the circumferential variation separated, yielding the following matrix form for the relationship of the Fourier coefficients.

$$\Omega y + \Lambda z = \ell \tag{1.38}$$

where

$$
y = \begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ y_k \end{bmatrix} \qquad
y_i = \begin{bmatrix} t_\xi^i \\ \hat{t}_{\xi\theta}^i \\ q_\xi^i \\ m_\xi^i \\ m_{\xi\theta}^i \end{bmatrix} \qquad
z = \begin{bmatrix} z_1 \\ z_2 \\ \cdot \\ \cdot \\ z_k \end{bmatrix} \qquad
z_i = \begin{bmatrix} u_i \\ v_i \\ w_i \\ \phi_{\xi_i} \\ \phi_{\theta_i} \end{bmatrix}
$$

$$\Omega = \begin{bmatrix} \bar{\Omega} & & & \\ & \bar{\Omega} & & \\ & & \cdot & 0 \\ & C & \cdot & \\ & & & \bar{\Omega}_k \end{bmatrix} \qquad \Lambda = \begin{bmatrix} \bar{\Lambda} & & & \\ & \bar{\Lambda} & 0 & \\ & & \cdot & \\ & 0 & \cdot & \\ & & & \bar{\Lambda}_k \end{bmatrix}$$

$$\ell = \begin{bmatrix} \bar{\ell}_1 \\ \bar{\ell}_2 \\ \cdot \\ \cdot \\ \bar{\ell}_k \end{bmatrix} \qquad \ell_i = \begin{bmatrix} \ell_{1i} \\ \ell_{2i} \\ \ell_{3i} \\ \ell_{4i} \\ \ell_{5i} \end{bmatrix}$$

It will be desirable to express boundary conditions in terms of z. The substitution of Eqs. (1.37) in Eq. (1.27) with appropriate orthogonality operations yields a set of recursion expressions relating Fourier coefficients of forces and moments to the displacement and stiffness coefficients. These relationships are given by

$$t_{\xi}^{k} = \sum_{j=0}^{K-1} \Big[r_{5k+1,\,5j+1}\, u_j' + r_{5k+1,\,5j+4}\, \phi_{\xi j}' + s_{5k+1,\,5j+1}\, u_j + s_{5k+1,\,5j+2}\, v_j + s_{5k+1,\,5j+3}\, w_j + s_{5k+1,\,5j+4}\, \phi_{\xi j} + s_{5k+1,\,5j+5}\, \phi_{\theta j} \Big] + a_{5k+1} \tag{1.39}$$

$$\hat{t}_{\xi\theta}^{k} = \sum_{j=0}^{K-1} \Big[r_{5k+2,\,5j+2}\, v_j' + r_{5k+2,\,5j+5}\, \phi_{\theta j}' + s_{5k+2,\,5j+1}\, u_j + s_{5k+2,\,5j+2}\, v_j + s_{5k+2,\,5j+4}\, \phi_{\xi j} + s_{5k+2,\,5j+5}\, \phi_{\theta j} \Big]$$

$$q_\xi^k = \sum_{j=0}^{K-1} \left[r_{5k+3,\,5j+2}\, w_j' + s_{5k+3,\,5j+1}\, u_j + s_{5k+3,\,5j+4}\, \phi_{\xi j} \right]$$

$$m_\xi^k = \sum_{j=0}^{K-1} \Big[r_{5k+4,\,5j+1}\, u_j' + r_{5k+4,\,5j+4}\, \phi_{\xi j}' + s_{5k+4,\,5j+1}\, u_j$$

$$+ s_{5k+4,\,5j+2}\, v_j + s_{5k+4,\,5j+3}\, w_j + s_{5k+4,\,5j+4}\, \phi_{\xi j}$$

$$+ s_{5k+4,\,5j+5}\, \phi_{\theta j} \Big] + a_{5k+4}$$

$$m_{\xi\theta}^k = \sum_{j=0}^{K-1} \Big[r_{5k+5,\,5j+2}\, v_j' + r_{5k+5,\,5j+5}\, \phi_{\theta j}' + s_{5k+5,\,5j+1}\, u_j$$

$$+ s_{5k+5,\,5j+2}\, v_j + s_{5k+5,\,5j+4}\, \phi_{\xi j} + s_{5k+5,\,5j+5}\, \phi_{\theta j} \Big]$$

where

$k = 0, 1, 2, \ldots K - 1$

Coefficients are given in Appendix 1-3.

Equation 1.39 can be written in matrix notation as

$$y = Rz' + Sz + a \tag{1.40}$$

Hence, the boundary conditions (1.38) become

$$\Omega R z' + (\Lambda + \Omega s)\, z = \ell - \Omega a \tag{1.41}$$

The form of Eq. (1.41) is modified if the shell has a pole (i.e., $r = o$) because the coefficients of the differential equations become singular for this case. Following a similar limiting process as described by Greenbaum (Reference 5) the conditions supplied at the pole are:

For Fourier index = 0

$$u_o = v_o = \phi_{\xi o} = \phi_{\theta o} = q_{\xi o} = 0 \tag{1.42}$$

For Fourier index = 1

$$u_1 + v_1 = w_1 = \phi_{\xi_1} + \phi_{\theta_1} = t_{\xi_1} - \hat{t}_{\xi\theta_1} = m_{\xi_1} - m_{\xi\theta_1} = 0 \tag{1.42}$$

For Fourier index ≥ 2

$$u_i = v_i = w_i = \phi_{\xi_i} = \phi_\theta = 0$$

1.8 FINITE DIFFERENCE FORMULATION IN THE MERIDIONAL VARIABLE

In a manner similar to that described in Reference 2 the partial differential equation in the matrix form (Equation 1.34) is reduced by a system of finite difference approximations. The variation in the meridional coordinate of the Fourier coefficients are described point-wise in Eq. (1.34). The following are finite difference forms for the partial differentials in the meridional coordinate at interior points.

$$\begin{aligned} \frac{\partial^2 f}{\partial \xi^2} &= \frac{1}{\Delta^2}\left(f_{i+1} - 2f_i + f_{i-1}\right) \\ \frac{\partial f}{\partial \xi} &= \frac{1}{2\Delta}\left(f_{i+1} - f_{i-1}\right) \end{aligned} \tag{1.43}$$

where Δ is the increment along ξ and subscripts denote the discrete value of the function taken.

The forms at boundary points (initial)

$$\left(\frac{\partial f}{\partial \xi}\right)_1 = \frac{1}{2\Delta}\left(3f_1 - 4f_2 + f_3\right) \tag{1.44}$$

terminal

$$\left(\frac{\partial f}{\partial \xi}\right)_N = \frac{1}{2\Delta}\left(- f_{N-2} + 4f_{N-1} - 3f_N\right) \tag{1.45}$$

The result of the application of the various finite difference forms can be stated compactly as the following set of equations:

$$A_o z_2 + B_o z_1 + C_o z_o = g_o$$

$$A_i z_{i+1} + B_i z_i + C_i z_{i-1} = g_i + 2\Delta\left(\alpha_i \ddot{z}_i + \beta_i \dot{z}_i\right) \tag{1.46}$$

$$A_N z_N + B_N z_{N-1} + C_N z_{N-2} = g_N$$

Where

$$A_o = -\Omega_o R_o$$

$$B_o = 2\frac{\Omega o}{\Delta} R_o$$

$$C_o = \Lambda_o + \Omega_o S_o - 3\frac{\Omega o}{2\Delta} R_o$$

$$g_o = \ell_o - \Omega_o a_o$$

the subscript (o) refers to the conditions at s = 0.

For $i \neq o, N$

$$A_i = \frac{2F_i}{\Delta} + G_i$$

$$B_i = -\frac{4F_i}{\Delta} + 2\Delta\,(H_i + K_i)$$

$$C_i = \frac{2F_i}{\Delta} - G_i \tag{1.47}$$

$$g_i = 2\Delta p_i$$

Finally for i = N or conditions at $s = \bar{s}$

$$A_N = \Lambda_N + \Omega_N S_N + 3\frac{\Omega_N}{2\Delta} R_N$$

$$B_N = -2\frac{\Omega_N}{\Delta} R_N \tag{1.48}$$

$$C_N = \frac{\Omega_N}{2\Delta} R_N$$

$$g_N = \ell_N - \Omega_N a_N$$

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR;

1.9 FINITE DIFFERENCE FORMULATION IN THE TIME VARIABLE

By the use of difference equations the above differential equation in matrix form may be transformed into a set of algebraic equations involving the variable z_i at successive values of time.

The most commonly used are the central difference forms; however, from a numerical stability aspect, the difference forms of Houbolt 3, 4 are used. These forms are

$$\ddot{z} = \frac{2z_j - 5z_{j-1} + 4z_{j-2} - z_{j-3}}{\delta^2} \tag{1.49}$$

$$\dot{z} = \frac{11z_j - 18z_{j-1} + 9z_{j-2} - 2z_{j-3}}{6\delta} \tag{1.50}$$

Where the subscript j refers to the time interval j = 0, 1, 2, ... and δ is the time increment.

Introducing these expressions in Eq. 1.46 results in the following set of algebraic equations for the shell response problem.

$$\begin{aligned} A_0 z_{2,j} + B_0 z_{1,j} + C_0 z_{0,j} &= g_0 \\ A_i^* z_{i+1,j} + B_{i,j}^* z_{i,j} + C_i^* z_{i-1,j} &= g_{i,j}^* \\ A_N z_{N,j} + B_N z_{N-1,j} + C_N z_{N-2,j} &= g_N \end{aligned} \tag{1.51}$$

where

$$A_i^* = \delta A_i$$

$$B_{i,j}^* = \delta B_i + 4\frac{\Delta}{\delta}\alpha_i + \frac{11}{3}\Delta\beta_i$$

$$C_i^* = \delta C_i$$

$$g_{i,j}^* = \delta g_i + L_i z_{i,j-1} + M_i z_{i,j-2} + N_i z_{i,j-3}$$

and

$$L_i = 10\frac{\Delta}{\delta}\alpha_i + 6\Delta\beta_i$$

$$M_i = -8\frac{\Delta}{\delta}\alpha_i - 3\Delta\beta_i$$

$$N_i = 2\frac{\Delta}{\delta}\alpha_i + \frac{2}{3}\Delta\beta_i$$

In the real problem no values of z_i exists for less than zero. The assumption that z_i does exist before t = o is a means of allowing the recurrence from Eq. (1.51) to apply at the origin as well as later values of time. Furthermore, no violation is made as long as the initial conditions of t = o are satisfied.

To obtain values for fictitious terms j = -1, -2 a procedure similar to that described by Houbolt is used. The procedure will require a modification of $B^*_{i,j}$, $g^*_{i,j}$ for j = 1, 2.

The difference equations for the first and second derivatives at the third increment of four successive increments are given by

$$\ddot{z}_{i,j} = \frac{1}{\delta^2}(z_{i,j+1} - 2z_{i,j} + z_{i,j-1})$$
$$\dot{z}_{i,j} = \frac{1}{6\delta}(2z_{i,j+1} + 3z_{i,j} - 6z_{i,j-1} + z_{i,j-2}) \tag{1.52}$$

Applying the equations at t = o, i.e., j = o

$$\ddot{z}_{i,0} = \frac{1}{\delta^2} = (z_{i,1} - 2z_{i,0} + z_{i,-1})$$
$$\dot{z}_{i,0} = \frac{1}{6\delta}(2z_{i,1} + 3z_{i,0} - 6z_{i,-1} + z_{i,-2}) \tag{1.53}$$

The initial conditions are that the displacements and velocities are prescribed at t = o. By application of Newton's second law, a secondary initial condition can be established, i.e., acceleration immediately following application of the initial forces. These conditions are

$$z_{i,o} = d_{i,o}$$
$$\dot{z}_{i,o} = v_{i,o} \tag{1.54}$$
$$\ddot{z}_{i,o} = a_{i,o}$$

Where $d_{i,o}$, $v_{i,o}$, $a_{i,o}$ are column n matrices formed of the respective coefficients of the Fourier expansions on θ of the initial displacements, velocities, and accelerations at the meridional location i.

Substitution of these values into Eq. (1.53) yields the following relations

$$
\begin{aligned}
z_{i,o} &= d_{i,o} \\
z_{i,-1} &= \delta^2 a_{i,o} + 2d_{i,o} - z_{i,1} \\
z_{i,-2} &= 6\delta^2 a_{i,o} + 6\delta v_{i,o} + a d_{i,o} - 8z_{i,1}
\end{aligned}
\tag{1.55}
$$

Substitution of these relations in Eq. (1.50) for j = 1 yields the following change in the definitions of Eq. (1.52)

$$
\begin{aligned}
B^*_{i,1} &= \delta B_i + 12\frac{\Delta}{\delta}\alpha_i + 6\Delta\beta_i \\
g^*_{i,1} &= \delta g_i + \left(12\frac{\Delta}{\delta}\alpha_i + 6\Delta\beta_i\right) d_{i,o} + \left(4\Delta\delta\alpha_i - \frac{7}{3}\delta^2\Delta\beta_i\right) a_{i,o} + \\
&\quad (12\Delta\alpha_i + 4\Delta\delta\beta_i)\, v_{i,o}
\end{aligned}
\tag{1.56}
$$

Substitution of definitions Eq. (1.50) for j = 2 yields the following change in definitions of Eq. (1.52)

$$
\begin{aligned}
g^*_{i,2} &= \delta g_i + \left(8\frac{\Delta}{\delta}\alpha_i + \frac{16}{3}\beta_i\right) z_{i,1} + \left(-4\frac{\Delta}{\delta}\alpha_i - \frac{5}{3}\Delta\beta_i\right) d_{i,o} + \\
&\quad \left(2\frac{\Delta}{\delta}\alpha_i + \frac{2}{3}\Delta\beta_i\right)\delta^2 a_{i,o}
\end{aligned}
\tag{1.57}
$$

The set of Eq. (1.52) and the additional definitions at the first two time intervals Eqs. (1.56, 1.57) is now the algebraic statement of the dynamic response problem.

1.10 MATRIX SOLUTION OF THE DIFFERENCE EQUATIONS

The set of matrix equations defined in Eqs. 1.51, 1.56, 1.57 will be solved by the same procedure described in Reference 2. This procedure is essentially a Gaussian elimination performed on the partitioned arrays. A slight modification of the elimination procedure described in Reference 6 is used here. Considering the first and second equations of Eq. (1.52) at the jth time interval

$$
\begin{aligned}
A_o z_{2,j} + B_o z_{1,j} + C_o z_{o,j} &= g_o \\
A_1 z_{2,j} + B^*_1 z_{1,j} + C_1 z_{o,j} &= g^*_{1,j}
\end{aligned}
\tag{1.58}
$$

PAGE 31 AND 32 ARE MISSING FROM ORIGINAL DOCUMENT.

~~PRECEDING PAGE BLANK NOT FILMED.~~

The matrix solution of these difference equations are solved by the procedure described in Section 1.10. By defining the starred quantities, ( )*, without the star the numerical procedure for the static analysis is completely defined.

1.12 REFERENCES

1. Sanders, J. Lyell, Jr., "An Improved First-Approximation Theory for Thin Shells," NASA TR R-24, 1959.

2. Budiansky and Radkowski, "Numerical Analysis of Unsymmetrical Bending of Shells of Revolution," AIAA Journal, Vol. 1, No. 8, August 1963.

3. Johnson, D. E., and Greif, R., "Dynamic Response of a Cylindrical Shell; Two Numerical Methods," AIAA J. 4, 486-494 (1966).

4. Houbolt, J. C., "A Recurrence Matrix Solution for the Dynamic Response of Aircraft in Gusts," NACA TN 2060, January 1950.

5. Greenbaum, G. A., "Comments on 'Numerical Analysis of Unsymmetrical Bending of Shells of Revolution,'" AIAA Journal, Vol. 2, No. 3 (March 1964), pp 590, 591.

6. Potters, M. L. "A Matrix Method for the Solution of a Second Order Difference Equation in Two Variables," Mathematch Centrum, Amsterdam, Holland, Report MR 19 (1955).

1.13 APPENDIX

1.13.1 Modification of Sander's Equations

Virtual change in the strain energy within C

$$\delta\bar{U} = \iint (N_{11}\delta\epsilon_{11} + N_{12}\delta\epsilon_{12} + N_{21}\delta\epsilon_{21} + N_{22}\delta\epsilon_{22} + M_{11}\delta k_{11} + M_{12}\delta k_{12} + M_{21}\delta k_{21} + M_{22}\delta k_{22} + Q_1\delta\gamma_1 + Q_2\delta\gamma_2)\, ds \tag{1}$$

$$\epsilon_{12} = \epsilon_{21} \tag{2}$$

$$k_{12} - k_{21} = \left(\frac{1}{R_2} - \frac{1}{R_1}\right)\epsilon_{12} + \frac{1}{\alpha_1\alpha_2}\,\partial\frac{\alpha_2\gamma_2}{\partial\xi_1} - \frac{1}{\alpha_1\alpha_2}\,\partial\frac{\alpha_1\gamma_1}{\partial\xi_2} \tag{3}$$

~~PRECEDING PAGE BLANK NOT FILMED.~~

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

Substituting (2) → (1)

$$= \iint \Big| N_{11}\delta\epsilon_{11} + (N_{12} + N_{21})\,\delta\epsilon_{12} + N_{22}\delta\epsilon_{22} + M_{11}\delta k_{11} +$$

$$\frac{1}{2} M_{12}\delta k_{12} + \frac{1}{2} M_{12}\delta k_{12} + \frac{1}{2} M_{21}\delta k_{21} + \frac{1}{2} M_{21}\delta k_{21} +$$

$$\frac{1}{2} M_{21}\delta k_{12} - \frac{1}{2} M_{21}\delta k_{12} + \frac{1}{2} M_{12}\delta k_{21} - \frac{1}{2} M_{12}\delta k_{21} + \tag{4}$$

$$M_{22}\delta k_{22} + Q_1\delta\gamma_1 + Q_2\delta\gamma_2 \Big| \alpha_1\alpha_2 d_{\xi 1} d_{\xi 2}$$

Rearrange

$$= \iint \Big| N_{11}\delta\epsilon_{11} + (N_{12} + N_{21})\,\delta\epsilon_{12} + N_{22}\delta\epsilon_{22} + M_{11}\delta k_{11} +$$

$$\frac{1}{2}(M_{12} - M_{21})\,\delta\,(k_{12} - k_{21}) + \frac{1}{2}(M_{12} + M_{21})\,\delta k_{21} + \tag{5}$$

$$\frac{1}{2}(M_{12} + M_{21})\,\delta k_{12} + M_{22}\delta k_{22} + Q_1\delta\gamma_1 + Q_2\delta\gamma_2 \Big| \alpha_1\alpha_2 d_{\xi 1} d_{\xi 2}$$

Substitute 3 → 5.

$$\iint \Big| N_{11}\delta\epsilon_{11} + (N_{12} + N_{21})\,\delta\epsilon_{12} + N_{22}\delta\epsilon_{22} + M_{11}\delta k_{11} +$$

$$\frac{1}{2}(M_{12} - M_{21})\,\delta \left(\left(\frac{1}{R_2} - \frac{1}{R_1} \right) \epsilon_{12} + \frac{1}{\alpha_1\alpha_2} \frac{\partial \alpha_2\gamma_2}{\partial \xi_1} - \right.$$

$$\left. \frac{1}{\alpha_1\alpha_2} \frac{\partial \alpha_1\gamma_1}{\partial \xi_2} \right) + \frac{1}{2}(M_{21} + M_{12})\,\delta\,(k_{12} + k_{21}) + M_{22}\delta k_{22} +$$

$$Q_1\delta\gamma_1 + Q_2\delta\gamma_2 \Big| \alpha_1\alpha_2 d_{\xi 1} d_{\xi 2}$$

Group.

$$\iint \Bigg\{ N_{11}\delta\epsilon_{11} + 2\left(\frac{1}{2}(N_{12}+N_{21}) + \frac{1}{4}\left(\frac{1}{R_2}-\frac{1}{R_1}\right)(M_{12}-M_{21})\right)\delta\epsilon_{12} +$$

$$N_{22}\delta\epsilon_{22} + M_{11}\delta k_{11} + \frac{1}{2\alpha_1\alpha_2}(M_{12}-M_{21})\,\delta\left(\frac{\partial\alpha_2\gamma_2}{\partial\xi_1}-\frac{\partial\alpha_1\gamma_1}{\partial\xi_2}\right) +$$

$$\frac{1}{2}(M_{21}+M_{12})\,\delta\,(k_{12}+k_{21}) + M_{22}\delta k_{22} + Q_1\delta\gamma_1 + Q_2\delta\gamma_2 \Bigg\}\,\alpha_1\alpha_2 d_{\xi 1} d_{\xi 2}$$

Expanding $\delta\,\frac{\partial}{\delta\xi}(\alpha\gamma)$

$$\delta\left(\frac{\partial\alpha_2\gamma_2}{\partial\xi_1}-\frac{\partial\alpha_1\gamma_1}{\partial\xi_2}\right) = \delta\left(\gamma_2\frac{\partial\alpha_2}{\partial\xi_1}+\alpha_2\frac{\partial\gamma_2}{\partial\xi_1}-\gamma_1\frac{\partial\alpha_1}{\partial\xi_2}-\alpha_1\frac{\partial\gamma_1}{\partial\xi_2}\right)$$

Substitute and Group.

$$\iint \Bigg\{ N_{11}\delta\epsilon_{11} + 2\left[\frac{1}{2}(N_{12}+N_{21}) + \frac{1}{4}\left(\frac{1}{R_2}-\frac{1}{R_1}\right)(M_{12}-M_{21})\right]\delta\epsilon_{12} +$$

$$N_{22}\delta\epsilon_{22} + M_{11}\delta k_{11} + \frac{1}{2}(M_{21}+M_{12})\,\delta\,(k_{12}+k_{21}) + M_{22}\delta k_{22} +$$

$$\left[Q_1 - \frac{1}{2\alpha_1\alpha_2}\frac{\partial\alpha_1}{\partial\xi_2}(M_{12}-M_{21})\right]\delta\gamma_1 + \left[Q_2 + \frac{1}{2\alpha_1\alpha_2}\frac{\partial\alpha_2}{\partial\xi_1}(M_{12}-M_{21})\right]$$

$$\delta\gamma_2 + \frac{1}{2}(M_{12}-M_{21})\,\delta\left(\frac{1}{\alpha_1}\frac{\partial\gamma_2}{\partial\xi_1}-\frac{1}{\alpha_2}\frac{\partial\gamma_1}{\partial\xi_2}\right)\Bigg\}\,\alpha_1\alpha_2 d_{\xi 1} d_{\xi 2}$$

Assumption

$$M_{12} - M_{21} \rightarrow o,\ \frac{\delta\partial\gamma_i}{\partial\xi_j} << \delta\,(\epsilon,\ k)$$

$$\iint \Big| N_{11}\delta\epsilon_{11} + 2\overline{N}_{12}\delta\epsilon_{12} + N_{22}\delta\epsilon_{22} + M_{11}\delta k_{11} + M_{22}\delta k_{22} +$$

$$2\overline{M}_{12}\delta\overline{k}_{12} + Q_1\delta\gamma_1\ \ Q_2\delta\gamma_2 \Big|\,\alpha_1\alpha_2 d_{\xi 1} d_{\xi 2}$$

$$\overline{N}_{12} = \frac{1}{2}(N_{12} + N_{21})$$

$$\overline{M}_{12} = \frac{1}{2}(M_{12} + M_{21})$$

$$\overline{k}_{12} = \frac{1}{2}(k_{12} + k_{21})$$

1.132 Multiplication of Series Expansions

The relationships A^{nj} and $\overline{A}^{nj}$ used in the text are found by multiplying term by term the series expansions of stiffness and strain and noting a recurring sequence of the resulting expressions

$$\text{e.g.,} \left(\sum_{j=0}^{K-1} b_j \operatorname{Cos} j\theta\right)\left(\sum_{n=0}^{K-1} \epsilon_{\xi n} \operatorname{Cos} n\theta\right) = \sum_{n=0}^{K-1}\left(\sum_{j=0}^{K-1} A^{nj} \epsilon_{\xi j}\right) \operatorname{Cos} n\theta$$

$$= \sum_{n=0}^{K-1}\left\{\sum_{j=0}^{K-1} \frac{1}{2}\left[b^{(n+j)} + \left[1 - \delta^2 (j-k) + \delta(k)\right] b^{|n-j|}\right] \epsilon_{\xi j}\right\} \operatorname{Cos} n\theta$$

$$\left(\sum_{j=0}^{K-1} b_j \operatorname{Cos} j\theta\right)\left(\sum_{n=1}^{K-1} \overline{\epsilon}_{\xi n} \operatorname{Sin} n\theta\right) = \sum_{n=1}^{K-1}\left(\sum_{j=0}^{K-1} \overline{A}^{nj} \overline{\epsilon}_{\xi j}\right) \operatorname{Sin} n\theta$$

$$= \sum_{n=1}^{K-1}\left\{\sum_{j=0}^{K-1} \frac{1}{2}\left[-b^{(n+j)} + \left[1 - \delta^2 (j-k) + \delta(k)\right] b^{|n-j|}\right] \overline{\epsilon}_{\xi j}\right\} \operatorname{Sin} n\theta$$

where

$$\delta(m) = \begin{cases} -1 & m < 0 \\ 0 & m = 0 \\ +1 & m > 0 \end{cases}$$

1.133 Coefficients

Stiffness parameters related in an isotropic manner

$$B_1 = B_2 = \frac{B_3}{\nu}$$

$$D_1 = D_2 = \frac{D_3}{\nu}$$

$$G_1 = G_2 = G_3 = \frac{B_1 (1 - \nu)}{2}$$

$$G_{13} = \frac{D_1 (1 - \nu)}{2}$$

$$f_{5k+1,\,5j+1} = B_1^{kj}$$

$$f_{5k+1,\,5j+4} = 0$$

$$f_{5k+2,\,5j+2} = \frac{G_1^{kj}}{2} + \frac{\lambda^2}{8} (\omega_\theta - \omega_\xi)^2 G_{13}^{kj}$$

$$f_{5k+2,\,5j+5} = \frac{\lambda^2}{4} (\omega_\theta - \omega_\xi) G_{13}^{kj}$$

$$f_{5k+3,\,5j+3} = G_2^{kj}$$

$$f_{5k+4,\,5j+1} = 0$$

$$f_{5k+4,\,5j+4} = \lambda^2 D_1^{kj}$$

$$f_{5k+5,\,5j+2} = \frac{\lambda^2}{4} (\omega_\theta - \omega_\xi) G_{13}^{kj}$$

$$f_{5k+5,\,5j+5} = \frac{\lambda^2}{2} G_{13}^{kj}$$

$$g_{5k+1,\,5j+1} = B_1^{kj\,\prime} + \gamma B_1^{kj}$$

$$g_{5k+1,\,5j+2} = \frac{k}{\rho} B_3^{kj} + \frac{k}{2\rho} G_1^{kj} - \frac{\lambda^2 k}{8\rho} (\omega_\theta - \omega_\xi)^2 G_{13}^{kj}$$

$$g_{5k+1,\,5j+3} = \omega_\xi B_1^{kj} + \omega_\theta B_3^{kj} + \omega_\xi G_2^{kj}$$

$$g_{5k+1,\,5j+4} = 0$$

$$g_{5k+1,\,5j+5} = +\frac{\lambda^2 k}{4\rho}(\omega_\xi - \omega_\theta)\, G_{13}^{kj}$$

$$g_{5k+2,\,5j+1} = -\frac{k}{2\rho} G_1^{kj} + \frac{\lambda^2 k}{8\rho}(\omega_\theta - \omega_\xi)^2\, G_{13}^{kj} - \frac{k}{\rho} B_3^{kj}$$

$$g_{5k+2,\,5j+2} = \frac{G_1^{kj\prime}}{2} + \frac{\gamma}{2} G_1^{kj} + \frac{\lambda^2}{8}(\omega_\theta - \omega_\xi)^2\, G_{13}^{kj\prime} -$$

$$\frac{\lambda^2}{8}\left(\gamma(\omega_\theta - \omega_\xi)^2 + 2\omega_\xi'(\omega_\theta - \omega_\xi)\right) G_{13}^{kj}$$

$$g_{5k+2,\,5j+4} = -\frac{\lambda^2}{4}\frac{k}{\rho}(\omega_\theta - \omega_\xi)\, G_{13}^{kj}$$

$$g_{5k+2,\,5j+5} = +\frac{\lambda^2}{4}(\omega_\theta - \omega_\xi)\, G_{13}^{kj\prime} + \frac{\lambda^2}{4}\left(2\gamma(\omega_\xi - \omega_\theta) - \omega_\xi'\right) G_{13}^{kj}$$

$$g_{5k+3,\,5j+1} = -\omega_\xi G_2^{kj} - \omega_\xi B_1^{kj} - \omega_\theta B_3^{kj}$$

$$g_{5k+3,\,5j+3} = G_2^{kj\prime} + \gamma G_2^{kj}$$

$$g_{5k+3,\,5j+4} = G_2^{kj}$$

$$g_{5k+4,\,5j+1} = 0$$

$$g_{5k+4,\,5j+2} = +\frac{\lambda^2}{4}\frac{k}{\rho}(\omega_\theta - \omega_\xi)\, G_{13}^{kj}$$

$$g_{5k+4,\,5j+3} = -G_2^{kj}$$

$$g_{5k+4,\,5j+4} = \lambda^2 D_1^{kj\prime} + \gamma\lambda^2 D_1^{kj}$$

$$g_{5k+4,\,5j+5} = \lambda^2 \frac{k}{\rho} D_3^{kj} + \frac{\lambda^2}{2} \frac{k}{\rho} G_{13}^{kj}$$

$$g_{5k+5,\,5j+1} = \frac{\lambda^2}{4} \frac{k}{\rho} (\omega_\theta - \omega_\xi)\, G_{13}^{kj}$$

$$g_{5k+5,\,5j+2} = +\frac{\lambda^2}{4} (\omega_\theta - \omega_\xi)\, G_{13}^{kj\prime} + \frac{\lambda^2}{4} \left(2\gamma\, (\omega_\theta - \omega_\xi) - \omega_\xi'\right) G_{13}^{kj}$$

$$g_{5k+5,\,5j+4} = -\frac{\lambda^2}{2} \frac{k}{\rho} G_{13}^{kj} - \lambda^2 \frac{k}{\rho} D_3^{kj}$$

$$g_{5k+5,\,5j+5} = \frac{\lambda^2}{2} G_{13}^{kj\prime} + \frac{\lambda^2}{2} \gamma G_{13}^{kj}$$

$$h_{5k+1,\,5j+1} = \gamma B_3^{kj\prime} - \omega_\xi \omega_\theta B_3^{kj} - \gamma^2 B_2^{kj} - \frac{k^2}{2\rho^2} G_1^{kj} - \frac{\lambda^2}{8} \frac{k^2}{\rho^2} (\omega_\theta - \omega_\xi)^2 G_{13}^{kj} - \omega_\xi^2 G_2^{kj}$$

$$h_{5k+1,\,5j+2} = \frac{k}{\rho} B_3^{kj\prime} - \frac{k}{\rho} B_2^{kj} - \frac{\gamma}{2} \frac{k}{\rho} G_1^{kj} - (\omega_\xi - \omega_\theta)^2 \frac{\gamma\lambda^2}{8} \frac{k}{\rho} G_{13}^{kj}$$

$$h_{5k+1,\,5j+3} = \omega_\xi B_1^{kj\prime} + \omega_\theta B_3^{kj\prime} + (\omega_\xi' + \gamma\omega_\xi)\, B_1^{kj} - \gamma\omega_\theta B_2^{kj}$$

$$h_{5k+1,\,5j+4} = +\,\omega_\xi G_2^{kj} + \frac{\lambda^2}{4} \frac{k^2}{\rho^2} (\omega_\theta - \omega_\xi)\, G_{13}^{kj}$$

$$h_{5k+1,\,5j+5} = +\frac{\lambda^2 \gamma}{4} \frac{k}{\rho} (\omega_\theta - \omega_\xi)\, G_{13}^{kj}$$

$$h_{5k+2,\,5j+1} = -\frac{k}{2\rho} G_1^{kj\prime} - \frac{\gamma k}{2\rho} G_1^{kj} - \frac{k}{\rho} \gamma B_2^{kj} + \frac{\lambda^2}{8} \frac{k}{\rho} (\omega_\theta - \omega_\xi)^2 G_{13}^{kj\prime} - \frac{\lambda^2}{8} \frac{k}{\rho} (\omega_\theta - \omega_\xi) \left(3\gamma\, (\omega_\theta - \omega_\xi) - 2\omega_\xi'\right) G_{13}^{kj}$$

$$h_{5k+2,\,5j+2} = \frac{1}{2}(\omega_\xi\omega_\theta - \gamma^2)\,G_1^{kj} - \frac{\gamma}{2}\,G_1^{kj\prime} - \frac{k^2}{\rho^2}\,B_2^{kj} - \omega_\theta^2 G_{13}^{kj} +$$

$$\frac{\lambda^2\gamma}{8}(\omega_\theta - \omega_\xi)^2\,G_{13}^{kj\prime} - \frac{\lambda^2}{8}\Big((\omega_\theta - \omega_\xi)^2(\omega_\xi\omega_\theta + 3\gamma^2) -$$

$$(\omega_\xi - \omega_\theta)\,2\gamma\omega_\xi'\Big)\,G_{13}^{kj}$$

$$h_{5k+2,\,5j+3} = -\frac{k}{\rho}\,\omega_\xi B_3^{kj} - \frac{k}{\rho}\,\omega_\theta B_2^{kj} - \omega_\theta\frac{k}{\rho}G_3^{kj}$$

$$h_{5k+2,\,5j+4} = +\frac{\lambda^2}{4}\frac{k}{\rho}\gamma\left(2(\omega_\theta - \omega_\xi) + \omega_\xi'\right)G_{13}^{kj} - \frac{\lambda^2}{4}\frac{k}{\rho}(\omega_\theta - \omega_\xi)\,G_{13}^{kj\prime}$$

$$h_{5k+2,\,5j+5} = +\,\omega_\theta G_3^{kj} - \frac{\lambda^2\gamma}{4}(\omega_\theta - \omega_\xi)\,G_{13}^{kj\prime} + \frac{\lambda^2}{4}(\omega_\theta - \omega_\xi)(\omega_\xi\omega_\theta + 2\gamma^2)\,G_{13}^{kj} +$$

$$\frac{\lambda^2\gamma}{4}\,\omega_\xi' G_{13}^{kj}$$

$$h_{5k+3,\,5j+1} = -(\omega_\xi' + \gamma\omega_\xi)\,G_2^{kj} - \omega_\xi\gamma B_3^{kj} - \omega_\theta\gamma B_2^{kj} - \omega_\xi G_2^{kj\prime}$$

$$h_{5k+3,\,5j+2} = -\frac{k}{\rho}\,\omega_\xi B_3^{kj} - \frac{k}{\rho}\,\omega_\theta B_2^{kj} - \omega_\theta\frac{k}{\rho}G_3^{kj}$$

$$h_{5k+3,\,5j+3} = -\frac{k^2}{\rho^2}\,G_3^{kj} - \omega_\xi^2 B_1^{kj} - 2\omega_\theta\omega_\xi B_3^{kj} - \omega_\theta^2 B_2^{kj}$$

$$h_{5k+3,\,5j+4} = G_2^{kj\prime} + \gamma G_2^{kj}$$

$$h_{5k+3,\,5j+5} = \frac{k}{\rho}\,G_3^{kj}$$

$$h_{5k+4,\,5j+1} = +\frac{k^2}{4\rho^2}\lambda^2(\omega_\theta - \omega_\xi)\,G_{13}^{kj} + \omega_\xi G_2^{kj}$$

$$h_{5k+4,\,5j+2} = +\frac{\lambda^2}{4}\frac{k}{\rho}\gamma(\omega_\theta - \omega_\xi)\,G_{13}^{kj}$$

$$h_{5k+4,\,5j+3} = 0$$

$$h_{5k+4,\,5j+4} = -\lambda^2\omega_\xi\omega_\theta D_3^{kj} + \lambda^2\gamma D_3^{kj\prime} - \frac{\lambda^2 k^2}{2\rho^2} G_{13}^{kj} - \gamma^2\lambda^2 D_2^{kj} - G_2^{kj}$$

$$h_{5k+4,\,5j+5} = \lambda^2 \frac{k}{\rho} D_3^{kj\prime} - \frac{\lambda^2\gamma}{2}\frac{k}{\rho} G_{13}^{kj} - \lambda^2\gamma \frac{k}{\rho} D_2^{kj}$$

$$h_{5k+5,\,5j+1} = +\frac{\lambda^2}{4}\frac{k}{\rho}(\omega_\theta - \omega_\xi)\, G_{13}^{kj\prime} - \frac{\lambda^2}{4}\frac{k}{\rho}\,\omega_\xi' G_{13}^{kj}$$

$$h_{5k+5,\,5j+2} = +\frac{\lambda^2\gamma}{4}(\omega_\theta - \omega_\xi)\, G_{13}^{kj\prime} - \frac{\lambda^2}{4}\left(\gamma\omega_\xi' + \omega_\xi\omega_\theta(\omega_\theta - \omega_\xi)\right) G_{13}^{kj} + \omega_\theta G_3^{kj}$$

$$h_{5k+5,\,5j+3} = +\frac{k}{\rho} G_3^{kj}$$

$$h_{5k+5,\,5j+4} = -\frac{\lambda^2}{2}\frac{k}{\rho} G_{13}^{kj\prime} - \lambda^2\frac{k}{\rho}\gamma D_2^{kj} - \frac{\lambda^2}{2}\frac{k}{\rho}\gamma G_{13}^{kj}$$

$$h_{5k+5,\,5j+5} = -\frac{\lambda^2\gamma}{2} G_{13}^{kj\prime} - \lambda^2\frac{k^2}{\rho^2} D_2^{kj} - G_3^{kj} + \frac{\lambda^2}{2}(\omega_\xi\omega_\theta - \gamma^2)\, G_{13}^{kj}$$

$$p_{5k+1} = -p_\xi^k + t_{\xi T}^{k\prime} + \gamma\,(t_{\xi T}^k - t_{\theta T}^k)$$

$$p_{5k+2} = -p_\theta^k - \frac{k}{\rho} t_{\theta T}^k$$

$$p_{5k+3} = -p^k - \omega_\xi t_{\xi T}^k - \omega_\theta t_{\theta T}^k$$

$$p_{5k+4} = \lambda^2 m_{\xi T}^{k\prime} + \lambda^2\gamma m_{\xi T}^k - \lambda^2\gamma m_{\theta T}^k$$

$$p_{5k+5} = -\frac{k}{\rho}\lambda^2 m_{\theta T}^k$$

$$r_{5k+1,\,5j+1} = B_1^{kj}$$

$$r_{5k+1,\,5j+4} = 0$$

$$r_{5k+2,\,5j+2} = \frac{1}{2} G_1^{kj} + \frac{\lambda^2}{8} (\omega_\theta - \omega_\xi)^2 G_{13}^{kj}$$

$$r_{5k+2,\,5j+5} = + \frac{\lambda^2}{4} (\omega_\theta - \omega_\xi) G_{13}^{kj}$$

$$r_{5k+3,\,5j+3} = G_2^{kj}$$

$$r_{5k+4,\,5j+1} = 0$$

$$r_{5k+4,\,5j+4} = D_1^{kj}$$

$$r_{5k+5,\,5j+2} = + \left(\frac{\omega_\theta - \omega_\xi}{4} \right) G_{13}^{kj}$$

$$r_{5k+5,\,5j+5} = \frac{1}{2} G_{13}^{kj}$$

$$s_{5k+1,\,5j+1} = \gamma B_3^{kj}$$

$$s_{5k+1,\,5j+2} = \frac{k}{\rho} B_3^{kj}$$

$$s_{5k+1,\,5j+3} = \omega_\xi B_1^{kj} + \omega_\theta B_3^{kj}$$

$$s_{5k+1,\,5j+4} = 0$$

$$s_{5k+1,\,5j+5} = 0$$

$$s_{5k+2,\,5j+1} = - \frac{k}{2\rho} G_1^{kj} + \frac{\lambda^2}{8} \frac{k}{\rho} (\omega_\theta - \omega_\xi)^2 G_{13}^{kj}$$

$$s_{5k+2,\,5j+2} = -\frac{\gamma}{2} G_1^{kj} + \frac{\lambda^2}{8}\gamma\,(\omega_\theta - \omega_\xi)^2\, G_{13}^{kj}$$

$$s_{5k+2,\,5j+4} = -\frac{\lambda^2}{4}\frac{k}{\rho}(\omega_\theta - \omega_\xi)\, G_{13}^{kj}$$

$$s_{5k+2,\,5j+5} = -\frac{\gamma\lambda^2}{4}(\omega_\theta - \omega_\xi)\, G_{13}^{kj}$$

$$s_{5k+3,\,5j+4} = -\omega_\xi G_2^{kj}$$

$$s_{5k+3,\,5j+4} = G_2^{kj}$$

$$s_{5k+4,\,5j+1} = 0$$

$$s_{5k+4,\,5j+2} = 0$$

$$s_{5k+4,\,5j+3} = 0$$

$$s_{5k+4,\,5j+4} = \gamma D_3^{kj}$$

$$s_{5k+4,\,5j+5} = \frac{k}{\rho} D_3^{kj}$$

$$s_{5k+5,\,5j+1} = +\frac{k}{4\rho}(\omega_\theta - \omega_\xi)\, G_{13}^{kj}$$

$$s_{5k+5,\,5j+2} = +\frac{\gamma}{4}(\omega_\theta - \omega_\xi)\, G_{13}^{kj}$$

$$s_{5k+5,\,5j+4} = -\frac{k}{2\rho} G_{13}^{kj}$$

$$s_{5k+5,\,5j+5} = -\frac{\gamma}{2} G_{13}^{kj}$$

$$a_{5k+1} = - t^{k}_{\xi T}$$

$$a_{5k+4} = - m^{k}_{\xi T}$$

All other r, s, a are equal to zero.

2.0 GENERAL DESCRIPTION OF COMPUTER PROGRAMS

2.1 INTRODUCTION

Both the Static and Dynamic analysis described in Section 1.0 are programmed for solution on the IBM 7094 digital computer using the IBM FORTRAN IV code. This section will give a general description of the two computer programs. The scope and limitations of the computer programs are outlined. The necessary quantities in selecting a mathematical model for the best utilization of the programs are described. The detailed description, the machine program, input instructions, flow diagram, sample data sheets, etc., are given in Section 3.0.

2.2 PROGRAM CAPABILITIES AND LIMITATIONS

Before describing some of the general program characteristics, it will perhaps be worthwhile to list some of the capabilities which are not generally present in other shell analysis programs. Also included in this list are limitations in the program that have resulted due to theoretical restrictions, computer storage capacity, economic considerations, etc.

a. Shells having meridional and circumferential variation in the stiffness properties can be analyzed. The circumferential variation must have a property of symmetry about some plane, θ = constant (constant = 0).

b. Static shells can be subjected to loads and temperatures which may vary meridionally and circumferentially with the limitation that these distributions must exhibit the same property of symmetry as the stiffness. The time dependent loadings for the dynamic response of shells must have the same conditions on the special distributions and also can vary arbitrarily in time.

c. For static shells, elastic foundations can be considered. The distribution of these parameters are subject to the same condition of symmetry described for stiffness. For the dynamic response, both elastic foundations and external dampings are considered. These are also subject to the same symmetry conditions imposed on the stiffness parameters.

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

d. Shells analyzed must have a surface of revolution reference surface. The middle surface of the shell has been taken as the reference surface for these computer programs, i.e., the section properties must be symmetric about the reference surface ($\int E\zeta d\zeta = 0$).

e. User compiled call functions are used in place of tables to input data of a 2 and 3 variable nature.

f. As many as 100 spacial integration intervals can be considered.

g. The Fourier expansions can be taken to 10 terms.

2.3 SIGN CONVENTIONS AND DIMENSIONS

The sign conventions used in the programs are illustrated in Figures (1.2, 1.3) in Section 1.0. To augment briefly the stresses σ_ξ, σ_θ and membrane forces N_ξ, N_θ are positive when they tend to produce tension and negative when they are in compression. The moments M_ξ, M_θ are positive in sign when they tend to produce tensile stresses in the inner surfaces and compressive stresses in the outer surface (see Section 2.4). The extensional displacement u, transverse deflection w, and meridional rotation Φ_ξ, are positive when the ξ and ζ coordinates are increased respectively.

In using the program, all data specified must be dimensionally consistent.

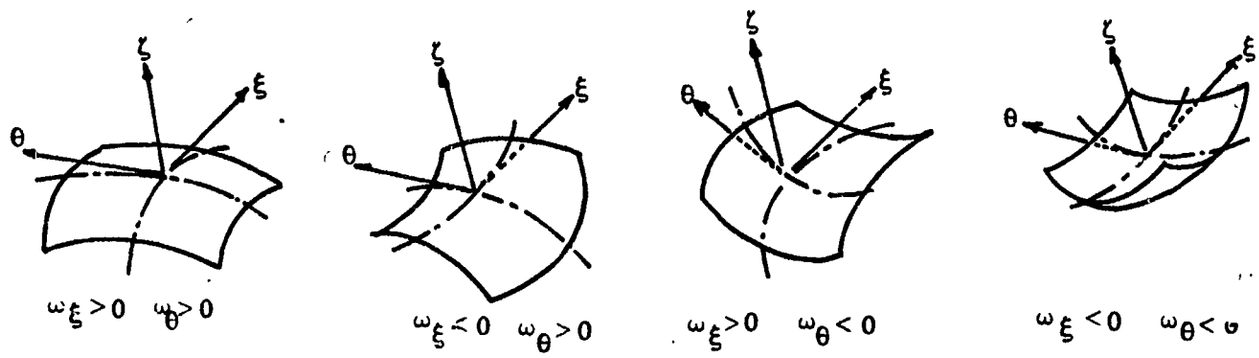
2.4 REFERENCE, INNER, AND OUTER SURFACES

The reference surface $\zeta = 0$ is chosen such that the requirements of limitation (c) above be satisfied. The cross sectional properties are then evaluated based upon this reference surface. As discussed in 1.15, a substantial simplification is obtained when specifying key geometric functions (e.g., ρ, γ), if the reference surface is chosen according to convenience anywhere within the shell wall. However, the shell stiffness parameter should be evaluated systematically along the lines discussed in Section 1.6 Eq. 1.17.

It will be convenient to refer to inner and outer surfaces of the shell. One can keep the inner and outer surface definitions clear by remembering that in direction of increasing value of ξ the outer surface is on the left and inner surface is on the right when the geometry is drawn with axial distance increasing from top to bottom and radial distance from left to right as shown in Figure 1.1.

2.5 GEOMETRY (GIN)

Geometric parameters must be defined at each station location. The sign convention for the curvature parameters, ω_ξ, ω_θ are defined in the figures below.



In order to assist the analyst in defining the set of geometry parameters with a minimum number of input parameters, several options for specific classes of geometries are made available. The options are described below with their identifying code number (GIN).

2.5.1 Cone-Cylinder Option (GIN) = 1.0)

This geometry option may be specified for a complete range of regional configurations generated by a straight line, e.g., circular plates, divergent cones, cylinders and convergent cones. A minimum of 3 input parameters are required. The input parameters required are defined as follows:

1. RA1 - Radial distance from axis of revolution to the first station (i = 1) of the region.

2. AXL - meridional length of shell.

3. ANX - angle the generator makes with the axis of revolution.

Figure 2.3 illustrates the geometric parameters used in describing the cone cylinder option.

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

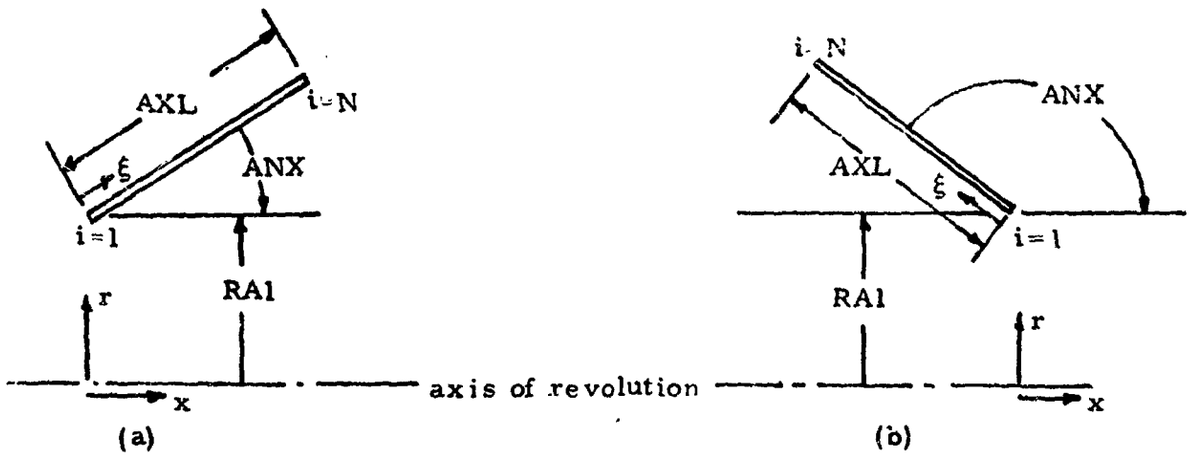


Figure 2.1 Cone Cylinder Geometry

Both RA1 and AXL are positive quantities. The parameter ANX is given in degrees and is positive clockwise measured from the generator to the positive X axis as shown in Figure 2.1.

2.5.2 Sphere-Toroid (GIN = 2.0)

This option may be specified for a complete range of regional configuration generated by a circular curve. Four input parameters are necessary for defining a sphere-toroid as shown in Figure 2.4.

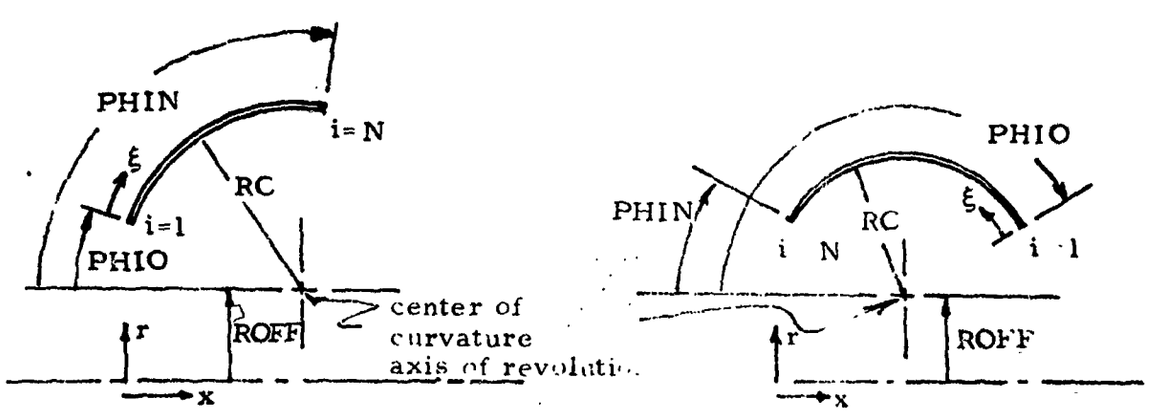


Figure 2.2 Sphere Toroid Geometry

The input parameters are:

1. RC - Radius of curvature of the generator

2. ROFF - Offset distance measured from axis of revolution to the center of meridional curvature.

3. PHIO - Angular position in degrees of the beginning of a region measured clockwise positive about the center of curvature from an axis parallel to the axis of revolution.

4. PHIN - Angular position of the end of the region.

2.5.3 Discrete Point Option (GIN = ±4.0)

This option was developed for use on regions where the generator cannot be described by one of the other options or where a curved generator is given by a set of discrete points. As a consequence of various possible ways the geometry may be supplied to the analyst, several variations of input data format can be accommodated.

On a positive indicator (GIN = +4.0), the program will set up the necessary geometric parameter from the input data which describes the generator by discrete radial and axial distances. The input quantities to the program are EM (number of points given), RIPT (radial distance from axis of revolution at input points), XIPT (axial coordinates of the input points). The set of RIPT and XIPT must include the first and last points of the region. XIPT must be given in ascending magnitudes. On a negative indicator (GIN = -4.0), the coordinates of the discrete points are given in radial and surface or arc length, the surface length coordinate is input directly in the XIPT locations.

An interpolation routine is used to obtain appropriate geometric parameters at station points from the original input values. The parameters such as curvatures are computed using finite difference forms of the station set. A least squares method is used to minimize the scatter of these computations. To hold the errors in curvatures to less than 10 percent, the number of points described by RIPT and XIPT should be at least as great as the number of stations. For some situations such as locations of major changes in the generator curve, it will be necessary to input a denser population of RIPT and XIPT. (See Figure 2.3.) Because of the difficulty involved in the least squares and interpolation routines, extreme care must be exercised in the use of this option in order to obtain an adequate description of shell geometry. A significant improvement in results is obtained if the additional recommendations described are adhered to.

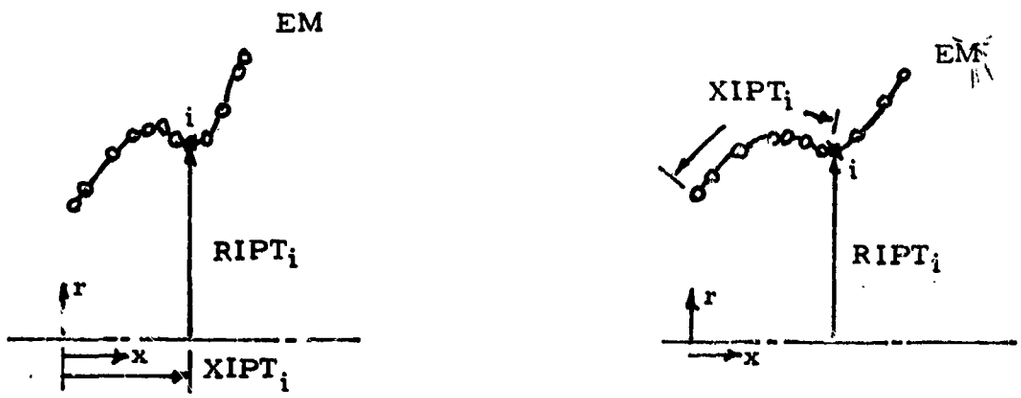


Figure 2.3

When the meridional and circumferential radii of curvatures are available, they can be input at discrete points and curve-fit to give a better description of the curvatures. If possible, it is strongly recommended that this capability be used since the errors in curvatures are reduced considerably to better control curvatures and less input points of the generator are required. This data is input in the location RCURV and RCURZ for radius of curvatures R_ξ and R_θ, respectively (Section 1.3). RCURV and RCURZ values must correspond with the points described by RIPT and XIPT. This is an optional input to both GIN = +4.0 and GIN = -4.0. When no values are input at RCURV and RCURZ locations, the curvatures will be computed from the discrete point set of RIPT and XIPT.

2.6 CALL FUNCTION PARAMETERS

Input data of a multivariable nature are defined to the computer programs by the use of call function subroutines. These parameters are described functional in terms of the meridional and circumferential variables (PEL, ZTA). The time variable (TU) is available for describing the loading history. Call function subroutines are to be coded and compiled by the user. Specific instructions for the coding of these subroutines are given in Section 3. Data that will be input in this manner are described in the following sections.

2.6.1 Stiffness Properties

The stiffness parameters are described in Section 1.6, Equations 1.16. It is sufficient to describe the inplane stiffness B_1, and the bending stiffness D_1 and Poisson's ratio in order to define all the necessary parameters. The

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

function (BFCN, Static, DBBDD, Dynamics) will be written to define these parameters over the reference surface of the shell.

$$B_1 = B_1(\xi, \theta)$$

$$D_1 = D_1(\xi, \theta)$$

2.6.2 Thermal Loads

The thermal load terms are described in Section 1.6, Equation 1.15 The call function (TFCN, Static, DTMP, Dynamics) will be written to define these parameters over the reference surface of the shell.

$$N_\xi^T = N_\xi^T(\xi, \theta)$$

$$M_\xi^T = M_\xi^T(\xi, \theta)$$

2.6.3 Elastic Foundations

The elastic foundation parameters are defined in Section 1.6, Equation 1.6. For the static program these parameters will be defined in function subroutine KFCN. For the dynamic program these parameters will be defined in DKMP.

2.6.4 External Damping

The external damping parameters are defined in Section 1.6, Equation 1.6. In the Dynamics program this function subroutine, DKMP will define both the elastic foundation terms and the external damping parameters.

2.6.5 Mass Properties

The mass properties for the Dynamic program defined in Section 1.6, Equation1.6 are defined to the computer program in the subroutine DMASS.

2.6.6 Pressure Loads

The pressure loading for the static case is defined to the program in the subroutine PFCN. For the dynamic case the time dependent loading is defined to the computer program in the subroutine PRSS. Positive pressure yields positive displacement.

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

2.6.7 Parameters for Stresses

For the static program, the parameters necessary for the computation of stresses are defined to the computer program by the following subroutines, STR1FN, STR2FN, TTF1, TTF2. These subroutines define the following parameters:

STR1FN, for the first stress print layer, EI1, Young's Modulus, DN1, distance from the reference surface to the stress print locations and POIS1, Poisson's ratio.

STR2FN, these are parameters necessary for stresses at a second layer. The definitions of EI2, DN2, POIS2 correspond to the first stress print parameters.

TTF1, this subroutine defines the temperature TMP1 and thermal expansion coefficient ALF1 at the first stress print location.

TTF2, these parameters are for the second stress print and the TMP2, ALF2 have definitions corresponding to TTF1.

2.7 STATIONS IN REGIONS: (EN)

The machine program achieves a shell solution by integration of finite difference equations along the meridian or arc length distance of the shell. The number of integration points (called stations) located in the region under consideration is assigned the EN code value. The stations are equally spaced with the initial point located on the reference surface at the beginning of the region designated station 1 ($i = 1$ or $s = 0$) and the last or EN-th station at the end of the region called station N ($i = N$ or $s = \bar{s}$). The numbering of stations proceeds in direction of positive meridional coordinate assigned to the respective region. The maximum number of stations permissible in a region is 100 (minimum 5). The regional input data are specified at stations on the reference surface of each region.

The length of the finite difference "lump" of shell is computed internal to the program from the length or wrap distance and the number of stations (EN) in the region. This finite difference increment of integration is defined as DEL in the program and printout.

The machine running time increases with the number of integration steps considered per region. The type of shell problem considered should dictate the size of the grid mesh or number of stations considered. This comes with experience and how the results are to be used. As a general rule, it is recommended that more integration intervals be used where rapid change in variables occurs along the length of the shell

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR;

3.0 DETAILED USE OF PROGRAMS

3.1 INTRODUCTION

The Unsymmetric Shell Computer Program is written in the FORTRAN IV language and makes use of the overlay feature and the ALTIO option of that language. ALTIO sacrifices speed of execution in favor of providing an additional 1900 core locations. (It does not make use of buffered input-output.) If the programs are to be compiled for use without the ALTIO option it will be necessary to add an additional BACKSPACE command at each location where the backspace moved the tape over an END OF FILE. These have been noted in the listings.

The program has been checked out in NAASYS, the NAA adaption of the IBM 7090/7094 IBSYS/IBJOB system and used the NAASYS library routines shown in the load map, Section 3.12m.

The NAASYS input tape is 'UNIT05'; the output tape is 'UNIT06'. In addition to these files, the program u e. units 3, 7, 8, 9, 10, 11, 12 and 13 as scratch tapes or for overlay storage during execution. NAASYS itself, is stored on 'UNIT01'.

The program is made up of an executive program and eleven links, four of which are called by the executive program, three by the DATLNK subroutine and four by the PANDX subroutine. The name of the main program in each link and a description of its use, follows.

Link No.	Name	Purpose
0	Executive	Reads the title cards and general data; sets up tape numbers and rewinds all tapes; controls the flow of execution of other links.
1	DATLNK	Acts as a sub-executive program to control GEOM, DATLYR and DATLD so that these three subroutines may share common auxiliary routines CODIMA, ENTERP and DINTRP.
2	GEOM	Reads geometry parameters, GMDA. Sets up top and bottom boundary matrices. Calculates DEL, R, XSI, WTHX, WFEX, GAMA, RHOX, WFEPX and stores the latter 5 arrays on tape. Prints general data, boundary matrices and pertinent geometry data on indicator.

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

Link No.	Name	Purpose
3	DATLYR	Sets up stiffness and temperature coefficients by function subprograms. These are stored on tape.
4	DATLD	This is the Fourier Load coefficients generator. Pressures, PFE, PTH and PN are supplied by function subprograms. These coefficients are preserved on tape.
5	PANDX	A sub-executive program which, together with the subroutines it controls--RSLT, FGHPE, DYLMN, GMTX--generates the recursion terms and the 'P' and 'X' matrices of equations 1.60 Section 1.10.
6	RSLT	Sets up the boundary matrices needed in computing 'P' and 'X' at the top and bottom of the shell. (See Section 1.10.)
7	FGHPE	Sets up the F, G, and H matrices in equilibrium equations 1.34 of Section 1.7.
8	DYLMN*	Sets up the L, M, and N matrices for the dynamic response equations of Section 1.9. These are written on tape 13.
9	GMTX*	Forms the PE matrix of equation 1.34 of Section 1.7. Computes 'G**', 'B**', 'P' and 'X' matrices.
10	ZMTRX	Computes 'Z', the solution matrix.
11	SUMS	Sums the Fourier components and computes internal loads, i.e., bending moments, transverse shear forces, and membrane forces. (In the static version, stresses may also be computed in this link with the parameters defined by function subprograms.)

*Dynamics version only

3.2 DECK SET-UP

In Figure 3A of this section we have shown the set-up of the column-binary program deck with the necessary control cards for each link. A list of the deck names corresponding to the subprogram names is given in the table at the end of the section.

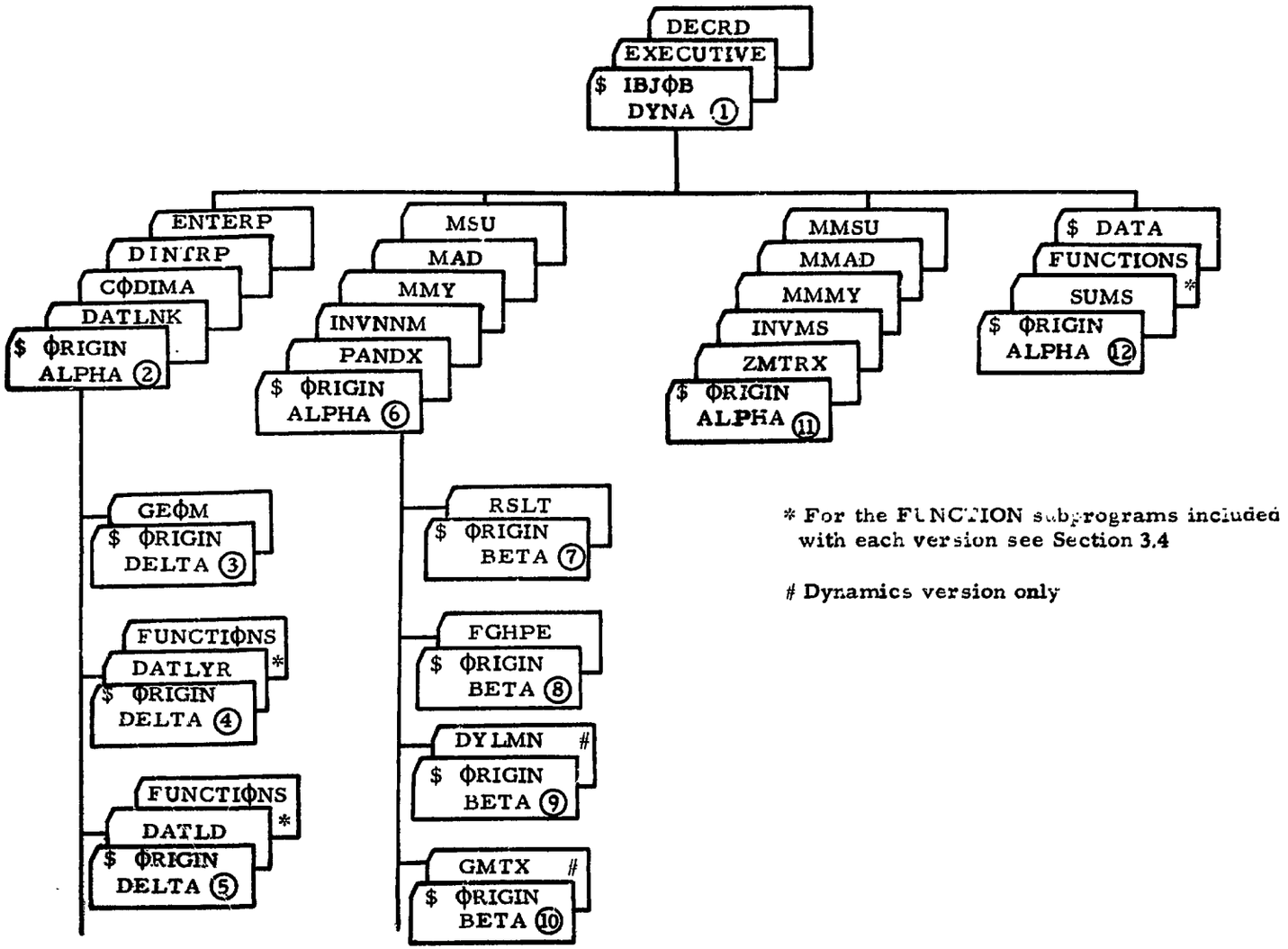

DECRD
EXECUTIVE
$ IBJØB
DYNA 1
ENTERP
DINTRP
CØDIMA
DATLNK
$ ØRIGIN
ALPHA 2
GEØM
$ ØRIGIN
DELTA 3
FUNCTIØNS
DATLYR
$ ØRIGIN
DELTA 4
FUNCTIØNS
DATLD
$ ØRIGIN
DELTA 5
MSU
MAD
MMY
INVNNM
PANDX
$ ØRIGIN
ALPHA 6
RSLT
$ ØRIGIN
BETA 7
FGHPE
$ ØRIGIN
BETA 8
DYLMN #
$ ØRIGIN
BETA 9
GMTX #
$ ØRIGIN
BETA 10
MMSU
MMAD
MMMY
INVMS
ZMTRX
$ ØRIGIN
ALPHA 11
$ DATA
FUNCTIONS
SUMS
$ ØRIGIN
ALPHA 12
* For the FUNCTION subprograms included with each version see Section 3.4
# Dynamics version only

Figure 3A.

TABLE OF SUBROUTINE NAMES AND DECK NAMES

Subroutine Name	Static Deck	Dynamics Deck
DATLNK	DLNK	DLKDY
GEOM	GMTRY	GMYD
DATLYR	STIFF	STFDY
DATLD	LOAD	DLDY
PANDX	PEANDX	PXDYN
RSLT	BNDRY	BNDD
FGHPE	FGMTX	FGHDY
DYLMN		LMNDY
GMTX		GDYN
ZMTRX	SOLTN	SLND
SUMS	SUSM	SUMD

The $IBJOB, $ORIGIN and $DATA cards are single control cards. The circled number found on the first two mentioned control cards, indicate the order in which they, plus the associated decks of that link, should be stacked. For example, those second level subroutines preceded by a $ORIGIN DELTA card - GEOM, DATLYR and DATLD will be found in the deck before the first level subroutine, PANDX, because they are executed in this order.

It is imperative that the utility subroutines be kept with each deck as shown. Since only one link of the same level may occupy core at a given time, the utility subprograms CODIMA, ENTERP and DINTRP are stored with the DATLNK link so that they may be shared by GEOM, DATLYR and DATLD.

Additional control cards preceding the $IBJOB card are likely to vary somewhat with the installation. An IBM systems handbook should be consulted. The cards used at Space and Information Systems Division are shown in Figure 3B, below.

```
9 020020 0  $JOB    IBJOB     055705055945  32  192120BB04702FURU IKE

9 020021 0  $IBJOB  PLASTIC
9 020323 0  $IBSYS
```

Figure 3B

3.3 DATA DECK SET-UP

Data decks should be stacked as follows:

1. Three cards (72 columns each) of title data.

2. GDA, general shell data, read by the EXECUTIVE program.

3. GMDA, geometry data, read by the GEØM subroutine.

All other parameters such as stiffnesses, springs, damping, mass, temperature and pressure loadings, and section properties used in stress computations are defined by function subprograms. These subprograms will be written by the user and compiled for each particular run. A discussion of how to write the various function subprograms is found in Section 3.4.

With the exception of the three title cards, each group of data listed above should have a minus sign in column 1 of the last card, since this data is read by the DECRD subroutine. See Section 3.5.1. In the instructions below, the DECRD index of each input quantity is given.

3.3.1 Title Cards

Three title cards form the first three cards of any data deck for each case. These cards are useful in identifying the run at a later date. They may include a brief problem description, the date of the run, a reference, etc.

These cards may not be omitted, but they may be blank, if desired If the cards are forgotten, the error indication from DECRD may occur for a multiple case run in which title cards are present for the second case, or the job may terminate with an EXECUTIØN ENDED designation (as explained in Section 3.5.1). A more serious situation occurs when the data from subsequent subroutines are read out of turn and the program is placed in an endless loop.

3.3.2 GDA, General Shell Data

All input data must be dimensionally consistent. It should be noted that all nondimensionalization is done internal to the program, thus all inputs must be supplied with appropriate dimensions (e.g., transverse load PN is input with dimensions P/L^2).

DECRD Index	Name	Description and Comments
1	A0	Reference length (a)
2	H0	Reference thickness (h_o)
3	E0	Reference Young's modulus (E_o)
4	SIG0	Reference stress (σ_o)
5	PØI	Poisson's ratio (ν)
6	ENF	The number of Fourier components (10 maximum)

DECRD Index	Name	Description and Comments
7	BCIT	Boundary condition indicator at first station. = 1. closed apex = 2. pinned = 3. clamped = 4. free = 5. roller w = 6. roller u = 10. special boundary matrices read in with geometry data. Must use 10 whenever non-zero values are prescribed in boundary matrices ℓ_o or ℓ_n.
8	BCIB	Boundary condition indicator at last station. (As BCIT)
9	PFLAG	Print indicator for input data. = 0. prints general data and boundary matrices = 1. prints above information and input data for the particular geometry configuration selected by GIN and the computed values for r, x, ω_θ, ω_ϕ, ρ and γ. = -1. prints all of above information and stiffness coefficients, elastic coefficients, thermal load and moment and pressure loads.
10	CEXT	Number of time cycles
11	DELT	Time increment in seconds

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS

DECRD Index	Name	Description and Comments
12	THT	Circumferential angle θ (degrees). Five values of THETA may be chosen. The deflections, rotations, internal loads and stresses will be printed at these values.
18	FPRNT	Fourier component print values. Three prints are permitted. Two intermediate prints of the Fourier summing are possible for checking convergence. The last FPRNT value given should be the same as the value given for ENF in the general data, i.e., GDA (6).

Note: If THT and FPRNT are not entered, the program will set THT (1) = 0.0 and FPRNT (1) = ENF.

The last card of GDA data should have a minus (-) in column 1.

3.3.3 GMDA, Geometry Data

The GMDA data array is zeroed before the data is read. This means that any data with a value = 0. need not be entered.

DECRD Index	Name	Description and Comments
1	GIN	Geometry indicator = 1. cone-cylinder = 2. sphere-toroid = ±4. discrete points
2	EN	Number of meridional stations (100 maximum)
3	ENLAY	Number of layers

When GIN = 1.0; see Section 2.5.1

4	RA1	Radial distance from axis of revolution to station 1. (L)*

*L - unit of length

DECRD Index	Name	Description and Comments
5	AXL	Meridional length of shell (L)
6	ANY	Angle the generator makes with the axis of revolution (degrees)

When GIN = 2.0; see Section 2.5.2

4	RC	Radius of curvature of the generator (L)
5	RØFF	Offset distance measured from axis of revolution to center of meridional curvature (L)
6	PHI0	Initial opening angle from the vertical axis (degrees)
7	PHIN	Final opening angle from the vertical axis (degrees)

When GIN = 4.0; see Section 2.5.3

8	EM	Number of RIPT's given (12 minimum, 100 maximum)
9	RIPT	Discrete radial distances
109	XIPT	Discrete axial or vertical distances (or arc lengths)
209	RCURV	Meridional radii of curvature
309	RCURZ	Circumferential radii of curvature

Boundary matrices, when not set by indicator: only the elements which might possibly be non-zero are included in the arrays. The explanation below assumes the user is familiar with Section 2.8, Boundary Conditions.

409	BNDTX (1) = $\Omega(1,1)$	Elements of force boundary matrix (Ω_0) at station 1
410	$\Omega(1,2)$	
411	$\Omega(2,2)$	
412	$\Omega(3,3)$	

DECRD Index	Name	Description and Comments
413	$\Omega(4, 4)$	
414	$\Omega(5, 4)$	
415	$\Omega(5, 5)$	
416	BNDTX (8) = $\Lambda(1, 1)$	Elements of displacement boundary matrix (Λ_o) at station 1
417	$\Lambda(2, 2)$	
418	$\Lambda(3, 3)$	
419	$\Lambda(4, 4)$	
420	$\Lambda(5, 5)$	
421	BNDTX (13) = ℓ_1	Column boundary matrix (ℓ) at station 1. The five elements are in consecutive locations.
426	BNDTB (1) = $\Omega(1, 1)$	Elements of force boundary matrix (Ω_N) at station N, stored in the same manner as Ω_o above.
433	BNDTB (8) = $\Lambda(1, 1)$	Elements of displacement boundary matrix (Λ_N) at station N; diagonal elements only.
438	BNDTB (13) = ℓ_1	Matrix (ℓ) for the last station

The last card of GMDA data should have a minus (-) in column 1.

3.4 FUNCTIONAL SUBPROGRAMS

The FUNCTION subprogram is an independently written and compiled program that is executed wherever its name appears in an arithmetic statement in another routine. The object (binary) or source (FORTRAN) deck must be included in the job deck even though the FUNCTION value equals zero; otherwise, the program that references the FUNCTION will not be executed.

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR;

Since a FUNCTION is a separately compiled subprogram, the variables and statement numbers within it do not relate to any other component of the program. Communication between the FUNCTION and its calling program takes place at execution time. The value of the FUNCTION is always returned to the calling program; values may also be returned via the FUNCTION arguments and/or variables assigned to a common region.

General Form:

```
FUNCTION      name (a, b, ... n)
    .
    .
    .
name = ...
    .
    .
    .
RETURN
    .
    .
    .
END
```

Name is a subprogram name (1-6 alphanumeric characters, the first of which is alphabetic). FUNCTION names used in this program are listed at the end of this section.

a, b, --n are nonsubscripted variable or array names used to transfer values between the FUNCTION subprogram and the program that references it. There must be at least one argument.

Basic Description:

The FUNCTION statement must be the first statement in the subprogram definition. (Comments cards excluded.) The name must not be the same as a library or built-in function unless the name is included in an EXTERNAL statement which precedes the first use of the FUNCTION by the calling program.

The FUNCTION subprogram may contain any FORTRAN statement except another FUNCTION statement, a SUBROUTINE statement, or a BLOCK DATA subprogram.

A variable in a COMMON block may be referenced if the rules for use of COMMON are followed.

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

The name of the FUNCTION must be assigned a value at least once within the subprogram. This value may be assigned by the appearance of the function name on the left side of an arithmetic statement (e.g. SOMEF = X/Y), or by its appearance in the list of a READ statement within the subprogram.

Arguments:

The relationship between variable names used as arguments in the calling program and the variables used as arguments in the FUNCTION subprogram is illustrated in the following example.

Calling Program

```
        .
        .
        .
   A = SOMEF (B, C)
        .
        .
        .
```

Subprogram

```
      FUNCTION SOMEF (X, Y)
50 SOMEF = X/Y
55 RETURN
      END
```

The value of the variable B of the calling program will be used as the value of the subprogram variable X; and C for Y. Thus, if B = 10.0 and C = 5.0, then A = 2.0.

The arguments used in the unsymmetrical shell of revolution program are PEL, ZTA and TU, where PEL is the meridional distance to a station, ZTA is the circumferential distance and TU is the elapsed time. If additional values are necessary to define a FUNCTION it will be necessary to include a READ statement in the subprogram and to supply the required data cards to satisfy the READ at execution time.

RETURN and END Statements:

A FUNCTION subprogram must contain an END statement and at least one RETURN statement. The END statement specifies the physical end of the subprogram for the compiler. The RETURN statement signifies a logical conclusion of the computation and returns any computed value and control to the calling program.

Multiple Entries:

The normal entry into a FUNCTION subprogram is made by a function reference in an arithmetic expression. Entry is made at the first executable statement following the FUNCTION statement. It is also possible to enter a FUNCTION by means of a reference to an ENTRY statement in the FUNCTION.

General Form

ENTRY name (a, b, ... n)

name is the name of an entry point. It must follow all the rules given for the FUNCTION name.

a, b, ... n are the dummy arguments corresponding to actual arguments supplied by a function reference.

The ENTRY statement is not executable. When inserted in a series of statements, the ENTRY statement has no effect on the logical flow of the subprogram. Entry to the subprogram is made at the first executable statement following the ENTRY statement, so it is easy to start using a subprogram at any desired point.

Within a FUNCTION subprogram, only the FUNCTION name may be used as the variable to carry a result back to the calling program. The ENTRY name may not be used for this purpose. The following example illustrates this rule.

Calling Program

```
        .
        .
        .
   A = ONEF (B, C)
        .
        .
        .
   F = TWOF (D, E)
        .
        .
        .
```

Subprogram

```
      FUNCTION ONEF (X, Y)
5 ONEF = X * Y
10 RETURN
```

```
      ENTRY TWOF (X, Y)
  ---> ONEF = X + Y
20 RETURN
      END
```

Note the use of the primary function name to return the function value to the calling program, even when the reference was to an ENTRY name.

Additional examples showing the use of FUNCTION subprograms with multiple ENTRY statements may be found in Sections 3.6, 3.7, 3.8 and 3.9.

Following is a table which gives the FUNCTION names, ENTRY names, calling program names and the assigned deck names ($ IBFTC manes) for the static and dynamic versions of the Unsymmetrical Shell of Revolution Program. For their definitions see Program Nomenclature, Section 3.13.

	STATIC	DYNAMIC
Calling Program DATLYR		
FUNCTION BBB (PEL, ZTA) ENTRY DDD (ZTA)	BFCN	DBBDD
FUNCTION ENTT (PEL, ZTA) ENTRY EMTT (PEL, ZTA)	TFCN	DTMP
FUNCTION *DKK1 (PEL, ZTA) ENTRY *DKK2 (PEL, ZTA) ENTRY DKK3 (PEL, ZTA) ENTRY DMP1 (PEL, ZTA) ENTRY DMP2 (PEL, ZTA) ENTRY DMP3 (PEL, ZTA)	KFCN	DKDMP
FUNCTION DMM1 (PEL, ZTA) ENTRY DMM4 (PEL, ZTA)		DMASS
Calling Program DATLD		
FUNCTION PPPN (PEL, ZTA) ENTRY PPPH (PEL, ZTA) ENTRY PPPF (PEL, ZTA)	PFCN	

*Deleted in the static deck because of core storage problems, therefore DKK3 becomes the FUNCTION name.

		STATIC	DYNAMIC
FUNCTION	PPPN (PEL, ZTA, TU)		
ENTRY	PPPH (PEL, ZTA, TU)		DPRSS
ENTRY	PPPF (PEL, ZTA, TU)		

Calling Program SUMS

FUNCTION	EI1 (PEL, ZTA)	
ENTRY	DN1 (PEL, ZTA)	STR1FN
ENTRY	POIS1 (PEL, ZTA)	
FUNCTION	EI2 (PEL, ZTA)	
ENTRY	DN2 (PEL, ZTA)	STR2FN
ENTRY	POIS2 (PEL, ZTA)	
FUNCTION	TMP1 (PEL, ZTA)	TTF1
ENTRY	ALF1 (PEL, ZTA)	
FUNCTION	TMP2 (PEL, ZTA)	TTF2
ENTRY	ALF2 (PEL, ZTA)	

3.5 UTILITY SUBROUTINES

3.5.1 DECRD Subroutine

All data, with the exception of the three title cards, is read by means of the DECRD subroutine, included with the symbolic decks.

This routine provides the facility for reading a variable number of pieces of floating point data into specified elements of an array; these elements may be either in sequential or in nonconsecutive locations. Only the information specified is actually read into storage.

```
                    16
        4.6
        0.

        1.      +17
      3.3        -2          24
```

The fixed point number (index) in the first field on each card defines the position of the first piece of data on the card. If the index is 1, the first piece of data will be stored in the first location reserved for the array; if it is 16, the first word will be placed in the 16th position, etc. The remaining fields on each card contain information for the successive locations of the array. If one or more fields are left blank, no information is read into the locations corresponding to these fields; the information already in these locations is unaltered.

The sample data sheets shown in Section 3.6.2 have 6 fields of 12 card columns each, and an identification field of 8 columns for sorting purposes.

a. The index must be written to the extreme right of the first field; it may not be zero or blank. (No decimal point)

b. The programmer should keep in mind the way in which FORTRAN stores arrays having double or triple subscripts, e.g., A(1, 1), A(2, 1), A(3, 1), A(1, 2), A(2, 2), etc.

c. The floating point (REAL) data should be entered with a decimal point (anywhere in the field) and an exponent, when necessary, written to the extreme right of the field and preceded by a '+' or '-'.

d. Reading data is concluded by placing a negative sign in column 1 of the last card to be read.

e. Zero should always be entered as '0.'. A'-0.' or '.0' will be recognized as a blank.

ERROR indication: If the index is zero or blank, the comment "****BAD INDEX ON DECRD CARD=" will be printed, followed by a print-out of the columns 1-80 of the defective card. The job will be terminated.

If the data for the array in the CALL statement has been completely read, and no negative sign has been encountered in column 1 of the last card sent, data intended for subsequent CALL's will be read into the incorrect array. When there are no data cards to satisfy the appetite of a CALL DECRD statement, the job will terminate with an EXECUTION ENDED designation.

If this occurs before all expected results have been printed, check the last card of each data block for the negative sign in column 1.

In order to use the DECRD routine in conjunction with the ALTIO option, it is necessary to physically load a copy of the routine with the program deck. ALTIO will not cause the proper adjustments to DECRD from the library tape at load time, so drop-in decks must be used.

```
$IBFTC DECRD                                                            DECRD000
      SUBROUTINE DECRD(D)                                               DECRD005
      DIMENSION  FLT(5), ID(2), D(1)                                    DECRD010
   10 READ (5,100) LOC, FLT, ID                                         DECRD015
  100 FORMAT (I12, 5F12.0, 1A6, 1A2)                                    DECRD020
      IF (LOC .EQ. 0)   GO TO 500                                       DECRD025
   15 K = IABS(LOC) - 1                                                 DECRD030
      DO 20 I = 1,5                                                     DECRD035
      IF (SIGN(1.0,FLT(I)).LT.0.0 .AND. FLT(I) .EQ. 0.0)  GO TO 20      DECRD040
      J = K + I                                                         DECRD045
      D(J) = FLT(I)                                                     DECRD050
   20 CONTINUE                                                          DECRD055
      IF (LOC .LT. 0)  GO TO 1000                                       DECRD060
      GO TO 10                                                          DECRD065
  500 WRITE (6,200) ID                                                  DECRD070
  200  FORMAT(10H0BAD DATA 1A6,1A2)                                     DECRD075
      CALL EXIT                                                         DECRD080
 1000 RETURN                                                            DECRD085
      END                                                               DECRD090
```

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

3.5.2 MAD, MSU, MMY, INVMS

These four subroutines perform matrix addition, subtraction, multiplication and inversion, respectively. They are extremely simple in their approach and must be recompiled to change dimensions for use in other decks. There are no error indications given in the 1st three routines other than the usual NAASYS trapping information for underflows, overflows and divide checks. When data has been entered correctly, these subroutines will present no problems.

The inversion routine has an error indicator, IX, which is set at 0 or -1 for a singular matrix. This is returned to the calling program through the argument list and may be tested after return. The Unsymmetrical Shell of Revolution Program makes such a test, prints the comment "SINGULAR MATRIX I = XX" and terminates the job. When this error occurs, check the data—especially any special boundary matrices e ered, and the FUNCTION subprograms.

The Dynamics version of the program uses a MAP (machine language coded) copy of the INVMS subroutine. The FORTRAN language routine causes the GDYN link to exceed core storage. Listings for both versions are included.

```
$IBFTC MADD
C     MATRIX ADD SUBROUTINE                                    DECK NO. 8K-903
C
C  ARGUMENTS
C     L  NO. OF ROWS
C     M  NO. OF COLS
C     A(I,J)      MRA
C     B(I,J)      MAD
C     C(I,J)      MSR
      SUBROUTINE MAD(L,M,A,B,C)
      DIMENSION A(50,50), B(50,50), C(50,50)
      DO 30 I=1,L
      DO 30 J=1,M
   30 C(I,J)=A(I,J)+B(I,J)
      RETURN
      END
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

```
$IBFTC MSUB
C       MATRIX SUBTRACT SUBROUTINE                                  DECK NO. 8K-904 00000400
C                                                                                   00000410
C   ARGUMENTS                                                                       00000420
C       L  NO. OF ROWS                                                              00000430
C       M. NO. OF COLS                                                              00000440
C       A(I,J)    MRA                                                               00000450
C       B(I,J)    MSU                                                               00000460
C       C(I,J)    MSR                                                               00000470
        SUBROUTINE MSU(L,M,A,B,C)                                                   00000480
        DIMENSION A(50,50), B(50,50), C(50,50)                                      00000490
        DO 30 I=1,L                                                                 00000500
        DO 30 J=1,M                                                                 00000510
   30   C(I,J)=A(I,J)-B(I,J)                                                        00000520
        RETURN                                                                      00000530
        END
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR;

```
$IBFTC MMPY
C       MATRIX MULTIPLY SUBROUTINE                          DECK NO. BK-901
C
C   ARGUMENTS
C       L   NO. OF ROWS X MATRIX
C       M   NO. OF COLS X MATRIX
C       N   NO. OF COLS Y MATRIX
C       X(I,K)    MRA
C       Y(K,J)    MMY
C       Z(I,J)    MSP
        SUBROUTINE MMY(L,M,N,X,Y,Z)
        DIMENSION X(50,50), Y(50,50), Z(50,50)
        DO 30 I=1,L
        DO 30 J=1,N
        Z(I,J)=0.0
        DO 30 K=1,M
   30   Z(I,J)=Z(I,J)+X(I,K)*Y(K,J)
        RETURN
        END
```

```
$IBFTC INVNSP                                                           INVS0000
C   ***  7094 FORTRAN IV SINGLE PRECISION GENERAL DIMENSION FIND INVERSEINVS0001
C OF MATRIX USING SINGLE PRECISION OPERATIONS                           INVS0002
C                                                                       INVS0003
C ARGUMENTS                                                             INVS0004
C  A  ORIGINAL MATRIX DIMENSIONED N X N                                 INVS0005
C  N  ROW DIMENSION OF A AND AI AS GIVEN BY DIMENSION STATEMENT         INVS0006
C  M1 NUMBER OF ROWS USED BY A AND AI M1 MUST BE LESS THAN OR EQUAL TO NINVS0007
C  IX ERROR INDICATOR                                                   INVS0008
C  D  WORKING ARRAY MUST BE SINGLE PRECISION AND DIMENSIONED M1 X M1    INVS0009
C                                                                       INVS0010
      SUBROUTINE INVMS  (A,N,M1,IX,D,AI)                                INVS0011
C                                                                       INVS0012
      DIMENSION  A(N,1), D(M1,1), IKOL(50), AI(N)                       INVS0020
      REAL     MULT                                                     INVS0030
      EQUIVALENCE  (I1,I4,I7  ),  (I2,I5,J,K),  (I3,L),                 INVS0040
     1    (PROD,MULT,AMX)                                               INVS0050
C                                                                       INVS0059
      M  =  M1                                                          INVS0060
      DO    3  I=1,M                                                    INVS0070
      DO    3  J=1,M                                                    INVS0080
    3 D(I,J)  =  A(I,J)                                                 INVS0090
C                                                                       INVS0099
      DO 1000   I=1,M                                                   INVS0100
      I1  =  I                                                          INVS0110
      DO 5  J=1,M                                                       INVS0120
    5    AI(J)     =   ABS(D(I,J))                                      INVS0130
         IF (I1 - 1)     6,11,6                                         INVS0140
    6    DO 10  I2=2,I1                                                 INVS0150
             I3  =  IKOL(I2-1)                                          INVS0160
   10     AI(I3)     =  0.                                              INVS0170
   11 CONTINUE                                                          INVS0180
      AMX  = AI(1)                                                      INVS0190
      I4  =  1                                                          INVS0200
         DO 15  I5=2,M                                                  INVS0210
            IF(AI(I5)   .LE. AMX)     GO TO 15                          INVS0220
         AMX  = AI(I5)                                                  INVS0230
         I4  =  I5                                                      INVS0240
   15    CONTINUE                                                       INVS0250
      IKOL(I)  =  I4                                                    INVS0260
      PROD  =  D(I,I4)                                                  INVS0270
      AT      =     1.                                                  INVS0280
      IF  (A(I,I4) .EQ. 0.)       GO TO 17                              INVS0290
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR;

```
      AT    =  ABS(PROD / A(I,I4))                                  INVS0300
   17 IF  (I .EQ. 1)                AMN   =  1.                    INVS0310
      AMN  =   AMIN1 (AT,AMN)                                       INVS0320
         D(I,I4)  =  1.                                             INVS0330
          IF (PROD)   20,2000,20                                    INVS0340
   20        DO 30 K=1,M                                            INVS0350
   30        D(I,K)  =  D(I,K) / PROD                               INVS0360
             DO 40 L=1,M                                            INVS0370
             IF (L .EQ. I)   GO TO 40                               INVS0380
             MULT  =  D(L,I4)                                       INVS0390
             D(L,I4)  =  0.                                         INVS0400
                  DO 35 K=1,M                                       INVS0410
   35             D(L,K)  =  D(L,K) - MULT * D(I,K)                INVS0420
   40        CONTINUE                                               INVS0430
 1000   CONTINUE                                                    INVS0440
C                                                                   INVS0449
      DO 1015 I=1,M                                                 INVS0450
        I7 = IKOL(I)                                                INVS0460
          DO 1010 J=1,M                                             INVS0470
        I8 = IKOL(J)                                                INVS0480
 1010   A (I7,J)   =    D(I,I8)                                     INVS0490
 1015    CONTINUE                                                   INVS0500
C                                                                   INVS0509
      IX  =  1                                                      INVS0510
      AMN  =  ALOG10 (1./AMN)                                       INVS0520
      IF (AMN .GT. 6.)    IX  =  -1                                 INVS0530
 1030 D(1,1)  =  AMN                                                INVS0540
      RETURN                                                        INVS0550
 2000 IX  =  0                                                      INVS0560
      GO TO 1030                                                    INVS0570
      END                                                           INVS0580
```

```
$IBMAP INVMS                                                    30710000
       LBL     INVMS,                                           30710010
INVS   CLA*    5,4                                              30710020
       SXA     EX1,1                                            30710030
       SXA     EX2,2                                            30710040
       SXA     EX3,3                                            30710050
       STA     N1                                               30710060
       STA     N3                                               30710070
       STA     N4                                               30710080
       STA     N6                                               30710090
       STA     N7                                               30710100
       STA     N10                                              30710110
       STA     N13                                              30710120
       STA     N14                                              30710130
       STA     FIN                                              30710140
       PAC     ,1                                               30710150
       SCD     N12,1                                            30710160
       SXD     MN,1                                             30710170
       ADD     ROWTA                                            30710180
       STA     R1                                               30710190
       STA     R2                                               30710200
       STA     R3                                               30710210
       STA     R4                                               30710220
       CLA*    4,4                                              30710230
       STO     N                                                30710240
       PAX     ,1                                               30710250
       SXD     N2,1                                             30710260
       SXD     N5,1                                             30710270
       SXD     N8,1                                             30710280
       SXD     N9,1                                             30710290
       SXD     N11,1                                            30710300
       LDQ*    5,4                                              30710310
       MPY     N                                                30710320
       XCA                                                      30710330
       SUB     N                                                30710340
       SUB     =1                                               30710350
       ADD     3,4                                              30710360
       STA     MX1       (MX+N**2-N-1)                          30710370
       STA     MX2                                              30710380
       STA     MX3                                              30710390
       STA     MX4                                              30710400
       STA     MX5                                              30710410
       STA     MX6                                              30710420
```

114

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

```
        CLA     3,4
        STA     H+1
        SUB     =1
        STA     ST1
        AXC     1,2         (-I)
        SXD     X1,2
SEARCH  SXA     S2,2
        SXD     ROWNI+1,2
        TXI     *+1,2,-1
        SXD     ROWNI+2,2
        TXI     *+1,2,1
N1      AXT     **,3
MX1     CLA     **,2
R1      SLW     **,3
N2      TXI     *+1,2,**          (N)
        TIX     MX1,3,1
        AXC     1,3
X1      TXH     *+2,3,** (-M-1)
        TRA     N3
        CLA     COLNOS-1,3
        PAX     ,2
R2      STZ     **,2
        TIX     X1,3,1
N3      AXT     **,3              (N)
        SXA     J,3
R3      CLA     **,3
X3      TIX     R4,3,1
        TRA     X5
R4      CAS     **,3
        TRA     X3
        TRA     X3
        SXA     J,3
        TRA     R3
X5      LXD     X1,3
        TXI     *+1,3,-1
        SXD     X1,3
        CLA     J
        STO     COLNOS-2,3
S2      AXT     **,2
        LDQ     J
        MPY     N
        XCA
        SUB     N
```

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR;

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

```
        SUB     =1                                      30710860
        ADD     3,4                                     30710870
        STA     J1                                      30710880
        STA     J2                                      30710890
        STA     J3                                      30710900
        STA     J4                                      30710910
J1      CLA     **,2                                    30710920
        TZE     SING                                    30710930
        STO     PROD                                    30710940
        CLA     =1.                                     30710950
J2      STO     **,2                                    30710960
N4      AXT     **,3                                    30710970
MX2     NULL                                            30710980
ROWI    CLA     **,2                                    30710990
        FDP     PROD                                    30711000
MX3     NULL                                            30711010
        STQ     **,2                                    30711020
N5      TXI     *+1,2,**                                30711030
        TIX     ROWI,3,1                                30711040
N6      AXC     **,1                                    30711050
ROWNI   SXA     S1,1                                    30711060
        TXH     *+2,1,**                                30711070
        TXH     S1+1,1,**                               30711080
J3      CLA     **,1                                    30711090
        STO     MULT                                    30711100
J4      STZ     **,1                                    30711110
N7      AXT     **,3                                    30711120
        XEC     S2                                      30711130
MX4     LDQ     **,2                                    30711140
        FMP     MULT                                    30711150
        STO     TEMP                                    30711160
MX5     CLA     **,1                                    30711170
        FSB     TEMP                                    30711180
MX6     STO     **,1                                    30711190
N8      TXI     *+1,2,**                                30711200
N9      TXI     *+1,1,**                                30711210
        TIX     MX4,3,1                                 30711220
S1      AXT     **,1                                    30711230
        TXH     *+2,1,-2                                30711240
        TXI     ROWNI,1,1                               30711250
        XEC     S2                                      30711260
MN      TXL     FIN,2,**        (-N)                    30711270
        TIX     SEARCH,2,1                              30711280
```

```
FIN     AXC     **,1                                    307I1290
ST2     SXA     SAV1,1                                  307I1300
        CLA     COLNOS-1,1                              307I1310
        PAC     ,2                                      307I1320
N10     AXT     **,3                                    307I1330
ST1     CLA     **,1            MX-1                    307I1340
DUM1    STO     TOL-1,2         DUMY-1                  307I1350
N11     TIX     *+1,1,**        N                       307I1360
N12     TIX     *+1,2,**        N                       307I1370
        TIX     ST1,3,1                                 307I1380
SAV1    AXT     **,1            RESTORE  (-I)           307I1390
        TXH     *+2,1,-2                                307I1400
        TXI     ST2,1,1                                 307I1410
N13     AXC     **,1            N                       307I1420
H2      SXA     H1,1                                    307I1430
        LDQ     COLNOS-1,1                              307I1440
        MPY*    5,4                                     307I1450
        XCA                                             307I1460
        SUB*    5,4                                     307I1470
        PAC     ,2              (-J)                    307I1480
        PCA     ,1                                      307I1490
        XCA                                             307I1500
        MPY     N                                       307I1510
        XCA                                             307I1520
        SUB     N                                       307I1530
        PAC     ,1                                      307I1540
N14     AXT     **,3            N                       307I1550
H       CLA     TOL,2           DUMY                    307I1560
        STO     **,1            MX                      307I1570
        TXI     *+1,1,-1                                307I1580
        TXI     *+1,2,-1                                307I1590
        TIX     H,3,1                                   307I1600
H1      AXT     **,1            RESTORE  (-I)           307I1610
        TXH     *+2,1,-2                                307I1620
        TXI     H2,1,1                                  307I1630
        CLA     =1                                      307I1640
IXSTO   STO*    6,4                                     307I1650
EX1     AXT     **,1                                    307I1660
EX2     AXT     **,2                                    307I1670
EX3     AXT     **,3                                    307I1680
        TRA     1,4                                     307I1690
SING    STZ*    6,4                                     307I1700
        TRA     EX1                                     307I1710
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR;

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

```
        REM     STORAGE     1                    307I1720
N       PZE                 1                    307I1730
J       PZE                                      307I1740
TOL     EQU     BLK1                             307I1750
COLNOS  EQU     BLK2                             307I1760
DUMY    EQU     TOL         N**2                 307I1770
TEMP    EQU     DUMY+2                           307I1780
MULT    EQU     DUMY+3                           307I1790
ROWTAB  EQU     DUMY+4                           307I1800
PROD    EQU     MULT        1                    307I1810
ROWTA   CLA     ROWTAB                           307I1820
        END                                      307I1830
```

3.5.3 DINTRP, ENTERP

These subroutines perform linear double and single interpolation. DINTRP makes use of ENTERP in interpolation for values along a particular curve. They are included for use by FUNCTION subprograms called by DATLYN or DATLD.

In DINTRP when the first argument is not bounded by the given table (curves) the statement

"ARGUMENT EXCEEDS EXTENT OF TABLE IN DINTRP." is printed, followed by

ARGUMENT = (1 PE 12.4)
TABLE VALUES (printed 6/line)

and the job is terminated.

When the argument in the single interpolation subroutine, ENTERP, exceeds the limits of the table, the routine selects the value at either end of the table and continues after printing.

"LIMITS OF TABLE EXCEEDED BY ARGUMENT = (1 PE 12.4)
(1 PE 12.4) = VALUE USED FROM TABLE"

Values entered in the tables should always be given in increasing algebraic order, both in terms of the numbers used to designate each curve of the family, and the values assigned to the points along the curve.

```
$IBFTC ENTP
C     LINEAR  INTERPOLATION  SUBROUTINE **ENTERP**          6J-997
C
C     SELECTS THE VALUF AT FITHER END OF TABLE WHFN ARGUMENT EXCEFDS
C     LIMIT, THEN CONTINUES
C
C SUBROUTINE ARGUMENTS
C     X          VALUE TO LOOK UP IN TABLE
C     TAB(1)     NO. OF PAIRS OF ARGUMENTS AND VALUES IN TABLE
C     TAB(2),ETC ARGUMENTS AND FUNCTIONS INTERLACED
C
      FUNCTION ENTERP (X,TAB)
C
      DIMENSION TAB(101)
      IF (TAB) 9,9,8
    9 ENTERP = - TAB
      RETURN
    8 N = TAB
      DO 5 I=1,N
    1 IF (TAB(2*I)-X) 5,4,3
    3 IF (I-1) 6,6,7
    7 ENTERP = TAB(2*I-1) + (X-TAB(2*I-2)) * (TAB(2*I+1) - TAB(2*I-1))
     V   / (TAB(2*I) - TAB(2*I-2))
      RETURN
    4 ENTERP = TAB(2*I+1)
      RETURN
    5 CONTINUE
      M = 2*N+1
      K = M
  105 WRITE  ( 6,10)   X, TAB(K)
   10 FORMAT (// 10X, 39HLIMITS OF TABLE EXCEEDED BY ARGUMENT = 1PE12.4
     1 / 10X, E12.4, 24H = VALUE USED FROM TABLE  )
      ENTERP = TAB(K)
      RETURN
    6 M = 2*N+1
      K = 3
      GO TO 105
      END
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

```
$IBFTC DNTP
C     DOUBLE INTERPOLATION FUNCTION SUBROUTINE                    J.L. FICK
C   ARGUMENTS
C     X      USED TO DETERMINE WHICH TABLE TO USE (WHICH CURVE OF FAMILY)
C     Y      ARGUMENT USED AS X COORDINATE OF POINTS ON CURVE
C     TAB(1)      NO. OF TABLES (NO. OF CURVES IN FAMILY)
C     TAB(2)      NO. TO REPRESENT FIRST CURVE. MUST BE ITS LOWEST VALUE
C     TAB(3)      NO. OF ARGUMENTS IN FIRST TABLE (POINTS ALONG CURVE)
C     TAB(4),ETC.     X AND Y VALUES ALONG CURVE (INTERLACED)
C     TAB(2*TAB(3)+4)      LIKE TAB(2) FOR 2ND CURVE.  THIS IS FOLLOWED
C     BY ITS NO. OF PTS., VALUES, AND REPEATED THE NO. OF TIMES NECES-
C     SARY BASED ON THE NO. OF TABLES LISTED IN TAB(1)
C
      FUNCTION DINTRP(X,Y,TAB)
      DIMENSION TAB(1)
      N = TAB + .5
      J = 0
      K = 0
    3 DO 50 I=1,N
C     M SELECTS ONE OF FAMILY OF CURVES EQUAL TO OR ABOVE X VALUE
      M = I+J
C     MU REPRESENTS NEXT LOWEST CURVE IN FAMILY
      MU = I+K-1
      IF(TAB(M+1)-X) 20,10,4
    4 IF(I-1) 5,5,7
   10 DINTRP=ENTERP(Y,TAB(M+2))
      RETURN
    7 Z2 = ENTERP(Y,TAB(M+2))
      Z1 = ENTERP(Y,TAB(MU+2))
      DINTRP = Z1 + (X - TAB(MU+1)) * (Z2 - Z1) / (TAB(M+1) - TAB(MU+1))
      RETURN
   20 K = J
   50 J =  INT (TAB(M+2) + .5) * 2 + 1 + J
    5 MP1 = M + 1
    6 WRITE (6, 30)  X, (TAB(L), L=1,MP1)
   30 FORMAT(//5X, 43HARGUMENT EXCEEDS EXTENT OF TABLE IN DINTRP. /10X,
     1 10HARGUMENT =,1PE12.4 /10X, 12HTABLE VALUES, 5X,1P6E12.4 /
     2   (27X, 6E12.4) )
      CALL EXIT
      STOP
      END
$*
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

3.5.4 CODIMA

CODIMA is a curve fitting subroutine which has the following properties:

1. To the straight portions of any curve defined by three points on a straight line, a straight line will be fitted.

2. To the smooth portion of any curve, a smooth curve will be fitted.

3. The method maintains continuous first derivative except at the ends of a straight segment.

4. The method will fit curves with "corners" or "sharp turns" without the large deviation usually found in other methods.

An interpolation method is developed such that some of the considerations taken when an engineer fits a curve with a french curve are formulated. This is the CODIM (controlled deviation interpolation method) concept.

The method will interpolate in a more engineering manner in the sense that:

1. The first derivative is continuous except at the ends of straight segments defined by three points on a straight line.

2. No large deviation will be found when slope changes are large.

3. Ability to change value and slope rapidly.

4. Ability to fit straight lines on straight line portions of the curve and fit smooth arcs through the smooth portions of the curve.

The method fits a polynomial through an interval with information given by "previous points" (points to the left) and another polynomial through the interval with information given by "subsequent points" (points to the right). These two polynomials are then compared for compatibility. If they differ, a weighted average of the polynomials is taken in a way such that the polynomial that deviates less from the straight line connecting the points defining the interval is given more weight. For simplicity, parabolae are used over higher degree polynomials, in the CODIMA version.

```
$IBFTC CODS                                                             CODIM000
C     PARABOLIC CURVE FITTING SUBROUTINE  (THREE POINTS)                CODIM001
C                                                                       CODIM009
      SUBROUTINE CODIMA (N1, X, Y, XI, YI, N2, SHAPE)                   CODIM010
C                                                                       CODIM011
C  ARGUMENTS                                                            CODIM012
C     N1          NO. OF POINTS TO ITERPOLATE                           CODIM013
C     X           LOCATION OF POINTS TO BE INTERPOLATED                 CODIM014
C     Y           ANSWERS                                               CODIM015
C     XI          INDEPENDENT ARGUMENT                                  CODIM016
C     YI          DEPENDENT ARGUMENT                                    CODIM017
C     N2          NO. OF ARGUMENTS                                      CODIM018
C     SHAPE       0 = FITS END WITH STRAIGHT LINE   1 = CURVE,LAST 3 PTS.CODIM019
C                                                                       CODIM020
      DIMENSION X(1),Y(1),XI(1),YI(1),D(2),A(2),B(2),C(2)               CODIM025
C                                                                       CODIM029
  100 IN = 0                                                            CODIM030
      XK = SHAPE                                                        CODIM040
C                                                                       CODIM049
      DO 800 N = 1,N1                                                   CODIM050
C                                                                       CODIM059
      IF (N2-2) 110,115,120                                             CODIM060
  110 Y(N) = YI(N2)                                                     CODIM070
      GO TO 800                                                         CODIM080
C                                                                       CODIM089
  115 Y(N) = (YI(2)-YI(1))/(XI(2)-XI(1))*  (X(N)-XI(1))+YI(1)           CODIM090
      GO TO 800                                                         CODIM100
C                                                                       CODIM109
  120 J = 1                                                             CODIM110
  125 IF(XI(J)-X(N)) 130,140,150                                        CODIM120
  140 Y(N) = YI(J)                                                      CODIM130
      GO TO 800                                                         CODIM140
C                                                                       CODIM149
  130 J = J+1                                                           CODIM150
      IF(J-N2) 125,125,145                                              CODIM160
  145 Y(N) = (YI(N2)-YI(N2-1))/(XI(N2)-XI(N2-1))*(X(N)-XI(N2-1))        CODIM170
     1        + YI(N2 - 1)                                              CODIM180
      GO TO 800                                                         CODIM190
C                                                                       CODIM199
  150 IF(J-2) 115,155,160                                               CODIM200
  155 K = 3                                                             CODIM210
      JJ = 1                                                            CODIM220
      GO TO 185                                                         CODIM230
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

```
  160 IF(J-N2) 170,165,145                                          CODIM240
  165 K = N2-1                                                      CODIM250
      JJ = 2                                                        CODIM260
      GO TO 185                                                     CODIM270
  170 IF(J-IN) 180,300,180                                          CODIM280
  180 JJ = 3                                                        CODIM290
      K = J                                                         CODIM300
C                                                                   CODIM309
  185 DO 200 M = 1,2                                                CODIM310
      X1 = XI(K-1)-XI(K)                                            CODIM320
      X2 = XI(K)-XI(K-2)                                            CODIM330
      X3 = XI(K-2)-XI(K-1)                                          CODIM340
      Y1 = YI(K-1)-YI(K)                                            CODIM350
      Y2 = YI(K)-YI(K-2)                                            CODIM360
      Y3 = YI(K-2)-YI(K-1)                                          CODIM370
      XX1 = XI(K-2)**2                                              CODIM380
      XX2 = XI(K-1)**2                                              CODIM390
      XX3 = XI(K)**2                                                CODIM400
      D(M) = XX1*X1 +XX2*X2+ XX3*X3                                 CODIM410
      A(M) = (YI(K-2)*X1 +YI(K-1)*X2+ YI(K)*X3)/D(M)                CODIM420
      B(M) = (XX1*Y1 + XX2*Y2+XX3*Y3)/D(M)                          CODIM430
      C(M) = YI(K-2)- A(M)*XX1 -B(M)*XI(K-2)                        CODIM440
  200 K = K+1                                                       CODIM450
  300 P1 = X(N)*(A(1)*X(N)+B(1)) +C(1)                              CODIM460
      P2 = X(N)*(A(2)*X(N)+B(2)) +C(2)                              CODIM470
      AL = (X(N)-XI(J-1))/(XI(J)-XI(J-1))                           CODIM480
      S  = YI(J)*AL + YI(J-1)*(1.0-AL)                              CODIM490
      GO TO (320,330,350),JJ                                        CODIM500
C                                                                   CODIM509
  320 P2 = P1                                                       CODIM510
      AL = (X(N)-XI(1))/(XI(2)-XI(1))                               CODIM520
      S = AL*YI(2) + (1.0-AL)*YI(1)                                 CODIM530
      IF (SHAPE) 321,322, 322                                       CODIM540
  321 XM1 = ABS (YI(2) - YI(1)) / (XI(2) - XI(1))                   CODIM550
      XM2 = ABS (YI(3) - YI(2)) / (XI(3) - XI(2))                   CODIM560
      XK = 1. - ABS (XM1 -XM2) / (XM1 + XM2)                        CODIM570
  322 P1 = S + XK *(P2-S)                                           CODIM580
      GO TO 350                                                     CODIM590
C                                                                   CODIM599
  330 P1 = P2                                                       CODIM600
      AL = (X(N)-XI(N2-1))/ (XI(N2)-XI(N2-1))                       CODIM610
      S = AL* YI(N2) +(1.0-AL)*YI(N2-1)                             CODIM620
      IF (SHAPE) 331,332, 332                                       CODIM630
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

```
  331 XM1 = ABS (YI(N2 - 1) - YI(N2)) /(XI(N2 -1) - XI(N2))          CODIM640
      XM2 = ABS (YI(N2 -2) - YI(N2 -1)) / (XI(N2 -2) - XI(N2 -1))    CODIM650
      XK = 1. - ABS (XM1 -XM2) / (XM1 + XM2)                         CODIM660
  332 P2 = S + XK*(P1-S)                                             CODIM670
C                                                                    CODIM679
  350 E1 = ABS (P1-S)                                                CODIM680
      E2 = ABS (P2-S)                                                CODIM690
      IN = J                                                         CODIM700
      IF(E1+E2) 700,700-750                                          CODIM710
  700 Y(N) = S                                                       CODIM720
      GO TO 800                                                      CODIM730
  750 YNUM = E1 * AL * P2 + (1. - AL) * E2 * P1                      CODIM740
      YDEN = E1 * AL + (1. - AL) * E2                                CODIM750
      Y(N) = YNUM / YDEN                                             CODIM760
  800 CONTINUE                                                       CODIM770
C                                                                    CODIM779
  900 RETURN                                                         CODIM780
      END                                                            CODIM790
```

3.5.5 FILE

FILE is a function subprogram by which any number of files (or end of file-marks) on a tape may be skipped, as specified by the programmer. Alternatively, the tape may be backspaced any number of files. The tape is positioned at the beginning of the desired file ready to read or write the first record. (Extent: 45 locations.)

Availability: On the FORTRAN library tape.

Use: The specified tape can be positioned as indicated by means of the statement:

A = FILE(I, J)

A	Used to make the statement format consistent with FORTRAN rules.
I	FORTRAN tape number of one of the available tapes; must be a positive fixed point constant or variable.
J	The number of files to be skipped or backspaced, including the file in which the tape is positioned. It must be a fixed point FORTRAN expression. J>O, skip J<O, backspace J should never be zero. If a zero argument is used, one file will be skipped either forward or backward depending on the sign of zero.

Examples:

a. Tape 4 is positioned within or at the end of file 2 and we wish to get to file 5. Then J = 3.

   A = FILE(4, 3)

b. Tape 4 is positioned at the beginning of file 12 and file 16 is desired. Then J = 4.

   A = FILE(4, 4)

c. Tape 4 is positioned anywhere within file 7 and file 4 is desired. Then J = -4.

   A = FILE(4, -4)

d. Tape 4 is positioned anywhere within file 7 and the beginning of file 7 is desired. Then J = -1.

   A = FILE(4, -1)

   Even if the tape is at the beginning of a file, you must count that file in computing the value of J. In the last example, if tape 4 is at the beginning of file 7 and a J of -1 is given, the next file to be read or written would still be file 7.

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR;

```
$IBMAP FILE.Q                                                          N5250000
       TITLE                                                           N5250010
       LBL     FILE,BEGIN,                                             N5250020
*************************************************************************N5250030
*                                                                      N5250040
*      ROUTINE TO POSITION A FORTRAN TAPE AT THE BEGINNING OF A FILE.  N5250050
*                                                                      N5250060
*      CALL    FILE(UNIT,COUNT)                                        N5250070
*                                                                      N5250080
*      UNIT    = FORTRAN TAPE NUMBER. (INTEGER)                        N5250090
*              UNIT MUST BE A 729 TAPE.                                N5250100
*              UNIT MUST BE A SCRATCH OR RESERVE TAPE.                 N5250110
*                                                                      N5250120
*      COUNT   = NUMBER OF FILES TO BE SKIPPED OR BACKSPACED,          N5250130
*              INCLUDING THE CURRENT FILE. (INTEGER)                   N5250140
*              COUNT GT 0 MEANS SKIP.                                  N5250150
*              COUNT LT 0 MEANS BACKSPACE.                             N5250160
*              COUNT EQ 0 MEANS NO OPERATION.                          N5250170
*                                                                      N5250180
*      ORIGINAL PROGRAMMING BY E. LESTER, 892 G.O., 7/10/63            N5250190
*      REVISED BY H. HARNED, I.B.M.                                    N5250200
*      REWRITTEN BY D. H. ADAMS, 200-312 S.I.D., 8/13/65               N5250210
*                                                                      N5250220
*************************************************************************N5250230
       TTL     PARAMETER DEFINITIONS                                   N5250240
UNIT   SET     1+2                  (ARG) FORTRAN UNIT                 N5250250
COUNT  SET     2+2                  (ARG) FILE COUNT (+ OR -)          N5250260
       SPACE                                                           N5250270
UCW3   SET     2                    UCB WORD 3                         N5250280
FCW3   SET     2                    FCB WORD 3                         N5250290
FCW4   SET     3                    FCB WORD 4                         N5250300
FCW6   SET     5                    FCB WORD 6                         N5250310
FCW10  SET     9                    FCB WORD 10                        N5250320
BSF    SET     6                    BACKSPACE FILE                     N5250330
LENGTH SET     12                   FCB LENGTH                         N5250340
       SPACE                                                           N5250350
UCB    SET     4                    (XR) UNIT CONTROL BLOCK            N5250360
FCB    SET     1                    (XR) FILE CONTROL BLOCK            N5250370
NFILES SET     1                    (XR) FILE COUNT                    N5250380
LFB    SET     4                    (XR) LENGTH OF FILE BLOCK          N5250390
       SPACE                                                           N5250400
MONUNI LBOOL   200000               MONITOR UNIT                       N5250410
INT    LBOOL   400000               INTERNAL FILE                      N5250420
```

```
NON729 BOOL    600000          DEVICE NOT 729 TAPE          N5250430
       SPACE                                               N5250440
.CLOSE EQU     .CLSFQ          AVOID IOCS FOR ALTIO         N5250450
       SPACE                                               N5250460
SYSUNI OCT     140             SYSTEM UNIT TABLE ORIGIN     N5250470
       TTL     INITIALIZATION                               N5250480
FILE   SAVE    1,I                                          N5250490
       NZT*    COUNT,4         =FILE COUNT                  N5250500
       RETURN  FILE            RETURN IF ZERO               N5250510
       CLA*    UNIT,4          =FORTRAN UNIT                N5250520
       TSX     .DECVA,4        CONVERT TO BCD               N5250530
       XCL                                                  N5250540
       ANA     =07777          MODULO 100                   N5250550
       SLW     NUMBER          STORE FOR COMPARE            N5250560
       CAL     .LFBL                                        N5250570
       PAC     ,FCB            =FILE BLOCK ORIGIN           N5250580
       PDX     ,LFB            =FILE BLOCK LENGTH           N5250590
*------                        SEARCH FCB'S FOR UNIT        N5250600
SEARCH CAL     FCW10,FCB       =FILE NAME, 'UNIT**'         N5250610
       ANA     =07777          EXTRACT **                   N5250620
       LAS     NUMBER          COMPARE WITH UNIT            N5250630
       TRA     *+2                                          N5250640
       TRA     FOUND                                        N5250650
       TNX     NOFILE,LFB,LENGTH  IS THERE ANOTHER FILE     N5250660
       TXI     SEARCH,FCB,-LENGTH YES, EXAMINE IT           N5250670
*------                                                     N5250680
FOUND  SCA     CLOSE,FCB       STORE LOC(FILE)              N5250690
       SCA     NDSEL,FCB                                    N5250700
       PCA     ,FCB                                         N5250710
       SUB     SYSUNI          DECEIVE .LXSEL               N5250720
       STA     LXSEL                                        N5250730
       TTL     HOUSEKEEPING                                 N5250740
       TSX     .CLOSE,4        CLOSE FILE                   N5250750
CLOSE  MON     **                                           N5250760
       SPACE                                                N5250770
       LDI     FCW3,FCB        IS UNIT                      N5250780
       BFT     MONUNI          MONITOR CONTROLLED           N5250790
       TRA     ERROR           YES                          N5250800
       SPACE                                                N5250810
       LDI     FCW4,FCB        IS FILE                      N5250820
       BFT     INT             INTERNAL                     N5250830
       TRA     ERROR           YES                          N5250840
       SPACE                                                N5250850
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

```
        LDI     FCW6,FCB            IS UNIT                          N5250860
        BFT     NON729              A 729 TAPE                       N5250870
        TRA     FRROR               NO                               N5250880
        SPACF                                                        N5250890
        LXA     LK.DR,4                                              N5250900
        CLA*    COUNT,4             =FILE COUNT                      N5250910
        PAX     ,NFILES                                              N5250920
        TPL     SKIP                FORWARD IF PLUS, ELSE ...        N5250930
        TTL     PERFORM RFQUESTED ACTION                             N5250940
        TSX     .NDSFL,4            BACKSPACE 1 FILE                 N5250950
NDSEL   PZE     **,,BSF                                              N5250960
        TIX     *-2,NFILES,1        COUNT DOWN                       N5250970
        SPACE                                                        N5250980
        CLA*    NDSEL               FXAMINF TAPE POSITION --         N5250990
        PAC     ,UCB                IF NOT AT LOAD POINT             N5251000
        NZT     UCW3,UCB            THEN READ PAST EOF               N5251010
        RETURN  FILE                FLSE RETURN                      N5251020
        SPACE                                                        N5251030
SKIP    CALL    .LXSEL(LXSEL)       SKIP 1 FILE                      N5251040
        TIX     SKIP,NFILES,1       COUNT DOWN                       N5251050
        RETURN  FILF                                                 N5251060
        SPACE   4                                                    N5251070
LXSEL   PON     **,,*+1                                              N5251080
        IORPN   ,,0                                                  N5251090
        TCH     *-1                                                  N5251100
        TTL     ERROR HANDLING                                       N5251110
NOFILE  CALL    .FXEM.(CODEXX)      FILE SEARCH FAILED               N5251120
        RETURN  FILE                                                 N5251130
CODEXX  DEC     151                                                  N5251140
        IOCD    FAIL,,4                                              N5251150
        IOCD    IGNORF,,3                                            N5251160
        SPACF                                                        N5251170
ERROR   CAL     FCW10,FCB           =FILE NAME                       N5251180
        SLW     NAME                                                 N5251190
        CALL    .FXEM.(CODEYY)      ILLEGAL FILE                     N5251200
        RETURN  FILE                                                 N5251210
CODEYY  DEC     152                                                  N5251220
        IOCD    BAD,,5                                               N5251230
        IOCD    IGNORE,,3                                            N5251240
        SPACF                                                        N5251250
FAIL    BCI     1,0UNIT                                              N5251260
NUMBER  BCI     1,******            UNIT (BCD)                       N5251270
        BCI     2, UNDEFINED                                         N5251280
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

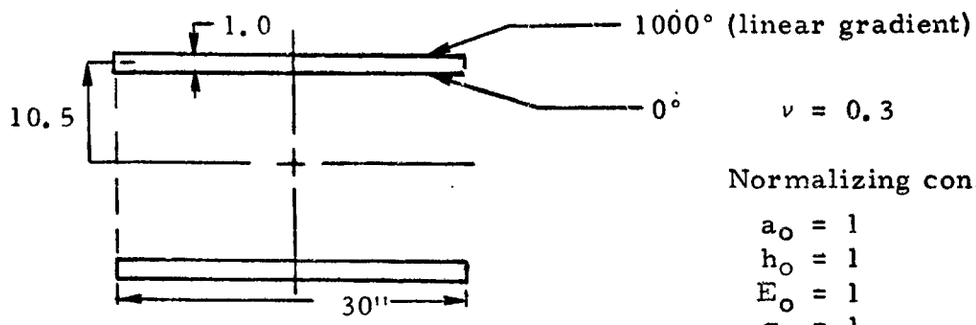
"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

```
BAD     BCI     4,0FILF SKIP ILLEGAL FOR                          N5251290
NAME    BCI     1,******          FILE NAME                       N5251300
IGNORE  BCI     3,0RFQUEST IGNORED.                               N5251310
        TTL                                                       N5251320
LK.DR   LDIR                                                      N5251330
        END                                                       N5251340
```

3.6 SAMPLE PROBLEM 1

3.6.1 Problem Description and Set-Up

A free cylinder under thermal loading is considered. The geometry and loading is shown in the following figure



$\nu = 0.3$

Normalizing constants

$a_o = 1$
$h_o = 1$
$E_o = 1$
$\sigma_o = 1$

FREE ENDS
No pressure load
Axisymmetric NF = 0
Stiffness Parameters

BBB = 2.4725×10^7 (in plane)
DDD = 1.8887×10^6 (bending)

Thermal Load

ENTT = 9.375×10^4 (in plane)
EMTT = 1.0045×10^4 (bending)

Stress Output Parameters

EI1 = 3×10^7 EI2 = 1.5×10^7

DN1 = -0.4167 DN2 = 0.5833

POIS1 = 0.3 POIS2 = 0.3

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

DECK NO PROGRAMMER DATE 1 2 JOB NO.

NUMBER	IDENTIFICATION	DESCRIPTION	DO NOT KEY PUNCH
			Title Cards
			CYL
			INDER UNDER
			THERMAL LØAD
	1		
			ARS JØURNAL,
			JANUARY 196
2			
	2		
			RADKØWSKI
			, DAVIS, BØL
			DUC
	3		
1			GDA, General Data
1.		AO	
1.		HO	
1.		EO	
1.		SIGO	
0.3	10	PØI	

| DECK NO. | | |

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

3.6.3 Functional Subprogram Used.

```
$IBFTC BFCN                                                    00000001
C                                          STIFFNESS FUNCTION  00000009
C                                                              00000019
      FUNCTION BBB( PFL,ZTA )                                  00000020
C                                                              00000021
      BBB  =  2.4725274F7                                      00000100
      RETURN                                                   00000110
C                                                              00000119
      ENTRY  DDD ( ZTA )                                       00000120
      BBB  =  1.8887362F6                                      00000200
      RETURN                                                   00000210
      END                                                      00000999
```

```
$IBFTC TFCN                                                    00000001
C                                          TEMPERATURE FUNCTION 00000009
C                                                              00000019
      FUNCTION  ENTT( PFL,ZTA )                                00000020
C                                                              00000021
      ENTT  =  9.375E+4                                        00000100
      RETURN                                                   00000110
C                                                              00000119
      ENTRY     EMTT( PFL,ZTA )                                00000120
      ENTT  =  1.0045F4                                        00000200
      RETURN                                                   00000210
      END                                                      00000999
```

```
$IBFTC FCN                                                          00000001
C                                                SPRING FUNCTION    00000009
C                                                                   00000019
      FUNCTION  DKK3( PFL,ZTA )                                     00000020
C                                                                   00000021
      DKK3  =  0.0                                                  00000100
      RETURN                                                        00000110
C                                                                   00000119
      END                                                           00000999
```

```
$IBFTC PFCN                                                         00000001
C                                            PRESSURE FUNCTION      00000009
C                                                                   00000019
      FUNCTION  PPPN( PFL,ZTA )                                     00000020
C                                                                   00000021
      PPPN  =  0.                                                   00000100
      RETURN                                                        00000110
C                                                                   00000119
      ENTRY     PPPH( PFL,ZTA )                                     00000120
      PPPN  =  0.                                                   00000200
      RETURN                                                        00000210
C                                                                   00000219
      ENTRY     PPPF( PFL,ZTA )                                     00000220
      PPPN  =  0.                                                   00000300
      RETURN                                                        00000310
      END                                                           00000999
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR;

```
$IBFTC STR1FN                                                           00000001
C          1ST SURFACE ** ELASTIC MODULUS, DISTANCE, POISSON'S RATIO    00000009
C                                                                       00000019
      FUNCTION  EI1 ( PEL,ZTA )                                         00000020
C                                                                       00000021
      EI1    =  3.0E+7                                                  00000100
      RETURN                                                            00000110
C                                                                       00000119
      ENTRY     DN1 ( PEL,ZTA )                                         00000120
      EI1    = -0.41666667                                              00000200
      RETURN                                                            00000210
C                                                                       00000219
      ENTRY    POIS1( PEL,ZTA )                                         00000220
      EI1    =  0.3                                                     00000300
      RETURN                                                            00000310
      END                                                               00000999
```

```
$IBFTC STR2FN                                                           00000001
C          2ND SURFACE ** ELASTIC MODULUS, DISTANCE, POISSON'S RATIO    00000009
C                                                                       00000019
      FUNCTION  EI2 ( PEL,ZTA )                                         00000020
C                                                                       00000021
      EI2    =  1.5E+7                                                  00000100
      RETURN                                                            00000110
C                                                                       00000119
      ENTRY     DN2 ( PEL,ZTA )                                         00000120
      EI2    =  0.58333333                                              00000200
      RETURN                                                            00000210
C                                                                       00000219
      ENTRY    POIS2( PEL,ZTA )                                         00000220
      EI2    =  0.3                                                     00000300
      RETURN                                                            00000310
      END                                                               00000999
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

```
$IBFTC TTF1                                                          00000001
C          1ST SURFACE ** COEF. OF THERMAL EXPANSION, TEMPERATURE    00000009
C                                                                    00000019
      FUNCTION  TMP1( PEL,ZTA )                                      00000020
C                                                                    00000021
      TMP1  =  0.0                                                   00000100
      RETURN                                                         00000110
C                                                                    00000119
      ENTRY     ALF1( PEL,ZTA )                                      00000120
      TMP1  =  10.0E-6                                               00000200
      RETURN                                                         00000210
      END                                                            00000999
```

```
$IBFTC TTF2                                                          00000001
C          2ND SURFACE ** COEF. OF THERMAL EXPANSION, TEMPERATURE    00000009
C                                                                    00000019
      FUNCTION  TMP2( PEL,ZTA )                                      00000020
C                                                                    00000021
      TMP2  =  1000.                                                 00000100
      RETURN                                                         00000110
C                                                                    00000119
      ENTRY     ALF2( PEL,. A )                                      00000120
      TMP2  =  5.0E-6                                                00000200
      RETURN                                                         00000210
      END                                                            00000999
```

3.6.4 Output

CYLINDER UNDER THERMAL LOAD

ARS JOURNAL, JANUARY 1962

RADKOWSKI, DAVIS, BOLDUC

GENERAL DATA

AO	=	1.0000E 00	HO	=	1.0000E 00	EO	=	1.0000E 00	SIGO	=	1.0000E 00
POI	=	3.0000E-01	ENF	=	1.0000E 00	BCIT	=	4.0000E 00	BCIB	=	1.0000E 01
PFLAG	=	-1.0000E 00									

PRINT ANGLES (THT) -

0.0000E-39 -1.0000E 10 -1.0000E 10 -1.0000E 10 -1.0000E 10 -1.0000E 10

FOURIER HARMONIC PRINTS (FPRNT) -

1.0000E 00 -1.0000E 10 -1.0000E 10 -1.0000E 10

BOUNDARY MATRICES

TOP BOUNDARY **

OMEGA

1.0000000E 00	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	1.0000000E 00	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	1.0000000E 00	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	1.0000000E 00	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	1.0000000E 00

LAMBDA

0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39

L -

0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39

BOTTOM BOUNDARY **

OMEGA

0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	1.0000000E 00	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39

LAMBDA

1.0000000E 00	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	1.0000000E 00	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	1.0000000E 00	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	1.0000000E 00

L -

0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39

GEOMETRY DATA FOR CONE - CYLINDER

NUMBER OF STATIONS - 21

RAI = 1.0500E 01 AXL = 1.5000E 01 ANX = 0.0000E-39

DEL = 7.5000E-01

I	R(I)	XI(I)	W(THETA)	W(XI)	RHOX(I)	GAMA(I)
1	1.0500000E 01	0.0000000E-39	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
2	1.0500000E 01	7.5000000E-01	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
3	1.0500000E 01	1.5000000E 00	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
4	1.0500000E 01	2.2500000E 00	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
5	1.0500000E 01	3.0000000E 00	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
6	1.0500000E 01	3.7500000E 00	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
7	1.0500000E 01	4.5000000E 00	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
8	1.0500000E 01	5.2500000E 00	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
9	1.0500000E 01	6.0000000E 00	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
10	1.0500000E 01	6.7500000E 00	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
11	1.0500000E 01	7.5000000E 00	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
12	1.0500000E 01	8.2500000E 00	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
13	1.0500000E 01	9.0000000E 00	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
14	1.0500000E 01	9.7500000E 00	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
15	1.0500000E 01	1.0500000E 01	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
16	1.0500000E 01	1.1250000E 01	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
17	1.0500000E 01	1.2000000E 01	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
18	1.0500000E 01	1.2750000E 01	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
19	1.0500000E 01	1.3500000E 01	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
20	1.0500000E 01	1.4250000E 01	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39
21	1.0500000E 01	1.5000000E 01	9.5238095E-02	0.0000000E-39	1.0500000E 01	0.0000000E-39

CHECKPRINT OF COEFC BLOCKS STORED IN GROUPS OF (2*ENF) PER MERIDIONAL STATION. THE COEFFICIENTS PRINTED ARE SCB1, SCD1, SCNTSI, SCMTSI, DCK3 (IN THAT ORDER) AND ARE SEPARATED FROM EACH OTHER BY THE THREE ..... FIELDS.

FOR STATION 1 THROUGH STATION 21

2.4725267E 07	-4.7055554E 00	2.4725267E 07	-4.7055554E 00	2.4725267E 07	-4.7055554E 00
2.4725267E 07	-4.7055554E 00	2.4725267E 07	-4.7055554E 00	2.4725267E 07	-4.7055554E 00
2.4725267E 07	-4.7055554E 00	2.4725267E 07	-4.7055554E 00	2.4725267E 07	-4.7055554E 00
2.4725267E 07	-4.7055554E 00	2.4725267E 07	-4.7055554E 00	2.4725267E 07	-4.7055554E 00
2.4725267E 07	-4.7055554E 00	2.4725267E 07	-4.7055554E 00	2.4725267E 07	-4.7055554E 00
2.4725267E 07	-4.7055554E 00	2.4725267E 07	-4.7055554E 00	2.4725267E 07	-4.7055554E 00
2.4725267E 07	-4.7055554E 00	2.4725267E 07	-4.7055554E 00	2.4725267E 07	-4.7055554E 00
.....	.....	.....	1.8887359E 06	-2.9236110E-01	1.8887359E 06
-2.9236110E-01	1.8887359E 06	-2.9236110E-01	1.8887359E 06	-2.9236110E-01	1.8887359E 06
-2.9236110E-01	1.8887359E 06	-2.9236110E-01	1.8887359E 06	-2.9236110E-01	1.8887359E 06
-2.9236110E-01	1.8887359E 06	-2.9236110E-01	1.8887359E 06	-2.9236110E-01	1.8887359E 06
-2.9236110E-01	1.8887359E 06	-2.9236110E-01	1.8887359E 06	-2.9236110E-01	1.8887359E 06
-2.9236110E-01	1.8887359E 06	-2.9236110E-01	1.8887359E 06	-2.9236110E-01	1.8887359E 06
-2.9236110E-01	1.8887359E 06	-2.9236110E-01	1.8887359E 06	-2.9236110E-01	1.8887359E 06
-2.9236110E-01	1.8887359E 06	.....	.....	.....	.....
9.3749997E 04	-1.8120659E-02	9.3749997E 04	-1.8120659E-02	9.3749997E 04	-1.8120659E-02
9.3749997E 04	-1.8120659E-02	9.3749997E 04	-1.8120659E-02	9.3749997E 04	-1.8120659E-02
9.3749997E 04	-1.8120659E-02	9.3749997E 04	-1.8120659E-02	9.3749997E 04	-1.8120659E-02
9.3749997E 04	-1.8120659E-02	9.3749997E 04	-1.8120659E-02	9.3749997E 04	-1.8120659E-02
9.3749997E 04	-1.8120659E-02	9.3749997E 04	-1.8120659E-02	9.3749997E 04	-1.8120659E-02
9.3749997E 04	-1.8120659E-02	9.3749997E 04	-1.8120659E-02	9.3749997E 04	-1.8120659E-02
9.3749997E 04	-1.8120659E-02	9.3749997E 04	-1.8120659E-02	9.3749997E 04	-1.8120659E-02
.....	.....	.....	1.0045000E 04	-2.0589192E-03	1.0045000E 04
-2.0589192E-03	1.0045000E 04	-2.0589192E-03	1.0045000E 04	-2.0589192E-03	1.0045000E 04
-2.0589192E-03	1.0045000E 04	-2.0589192E-03	1.0045000E 04	-2.0589192E-03	1.0045000E 04
-2.0589192E-03	1.0045000E 04	-2.0589192E-03	1.0045000E 04	-2.0589192E-03	1.0045000E 04
-2.0589192E-03	1.0045000E 04	-2.0589192E-03	1.0045000E 04	-2.0589192E-03	1.0045000E 04
-2.0589192E-03	1.0045000E 04	-2.0589192E-03	1.0045000E 04	-2.0589192E-03	1.0045000E 04
-2.0589192E-03	1.0045000E 04	-2.0589192E-03	1.0045000E 04	-2.0589192E-03	1.0045000E 04
.....	.....	.....	.....	.....	.....
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
:	:	:	:	:	:

148

CHECKPRINT OF LOAD COEF. STORED IN GROUPS OF ALL ENF PER MERIDIONAL STATION. THE COEFFICIENTS ARE PFE, PTH, PN (IN THAT ORDER) AND ARE SEPARATED FROM EACH OTHER BY ****** FIELDS.

FOR STATION 1 THROUGH STATION 21

0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	.....	.....	.....
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	.....	.....	.....
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39			

'Z' MATRIX FOR STATION 21

0.0000000E-39	0.0000000E-39	3.0273973E-02	2.9521609E-10	0.0000000E-39

'Z' MATRIX FOR STATION 20

-2.2108287E-03	0.0000000E-39	3.0281600E-02	2.2427470E-05	0.0000000E-39

'Z' MATRIX FOR STATION 19

-4.4213307E-03	0.0000000E-39	3.0304480E-02	5.0239731E-05	0.0000000E-39

'Z' MATRIX FOR STATION 18

-6.6311037E-03	0.0000000E-39	3.0349646E-02	9.0617861E-05	0.0000000E-39

'Z' MATRIX FOR STATION 17

-8.8395379E-03	0.0000000E-39	3.0429458E-02	1.4980632E-04	0.0000000E-39

'Z' MATRIX FOR STATION 16

-1.1045710E-02	0.0000000E-39	3.0560775E-02	2.32[illegible]686E-04	0.0000000E-39

'Z' MATRIX FOR STATION 15

-1.3248305E-02	0.0000000E-39	3.0763433E-02	3.3978652E-04	0.0000000E-39

'Z' MATRIX FOR STATION 14

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

'Z' MATRIX FOR STATION 6

-3.2645028E-02	0.0000000E-39	3.4107951E-02	-1.5989654E-03	0.0000000E-39

'Z' MATRIX FOR STATION 5

-3.4790510E-02	0.0000000E-39	3.2547584E-02	-3.3380269E-03	0.0000000E-39

'Z' MATRIX FOR STATION 4

-3.6985673E-02	0.0000000E-39	2.9471267E-02	-5.7134319E-03	0.0000000E-39

'Z' MATRIX FOR STATION 3

-3.9268604E-02	0.0000000E-39	2.4355890E-02	-8.7390917E-03	0.0000000E-39

'Z' MATRIX FOR STATION 2

-4.1688553E-02	-0.0000000E-39	1.6683303E-02	-1.2315728E-02	-0.0000000E-39

'Z' MATRIX FOR STATION 1

-4.4304717E-02	0.0000000E-39	6.0422107E-03	-1.6167123E-02	-0.0000000E-39

STATION 21

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
0.0000000E-39	0.0000000E-39	3.0273973E-02	2.9521609E-10	0.0000000E-39

STATION 20

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
-2.2108287E-03	0.0000000E-39	3.0281600E-02	2.2427470E-05	0.0000000E-39

STATION 19

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
-4.4213307E-03	0.0000000E-39	3.0304480E-02	5.0239731E-05	0.0000000E-39
.	.	.	.	.
.	.	.	.	.

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

```
                .                .                .                .                .
                .                .                .                .                .

THETA  =    0.00
 ENF   =    0.0
          U                 V                 W                PHI(XI)          PHI(THETA)
   -3.9268604E-02    0.0000000E-39    2.4355890E-02    -8.7390917E-03    0.0000000E-39

STATION     2

THETA  =    0.00
 ENF   =    0.0
          U                 V                 W                PHI(XI)          PHI(THETA)
   -4.1688553E-02   -0.0000000E-39    1.6683303E-02    -1.2315728E-02   -0.0000000E-39

STATION     1

THETA  =    0.00
 ENF   =    0.0
          U                 V                 W                PHI(XI)          PHI(THETA)
   -4.4304717E-02    0.0000000E-39    6.0422107E-03    -1.6167123E-02    -0.0000000E-39
```

STATION 21

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
5.2645898E 02	-5.9428223E 02	0.0000000E-39	-1.4750057E-03

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
-1.0094696E 04	-1.0059907E 04	0.0000000E-39	-0.0000000E-39

STRESSES FOR LAYER 1

SIG(XI)	SIG(THETA)	SIG(XI,THETA)	SIG(XI,ETA)	SIG(THETA,ETA)
1.2606342E 05	1.2431609E 05	-0.0000000E-39	-1.9666753E-03	-0.0000000E-39

STRESSES FOR LAYER 2

SIG(XI)	SIG(THETA)	SIG(XI,THETA)	SIG(XI,ETA)	SIG(THETA,ETA)
-4.4544874E 04	-4.5114929E 04	0.0000000E-39	-9.8333765E-04	-0.0000000E-39

STATION 20

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
5.2107422E 02	-5.7955566E 02	0.0000000E-39	1.8084047E 01

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
-1.0108257E 04	-1.0063975E 04	0.0000000E-39	-0.0000000E-39

STRESSES FOR LAYER 1

SIG(XI)	SIG(THETA)	SIG(XI,THETA)	SIG(XI,ETA)	SIG(THETA,ETA)
1.2615487E 05	1.2436532E 05	-0.0000000E-39	2.4112076E 01	-0.0000000E-39
.	.	.	.	.

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

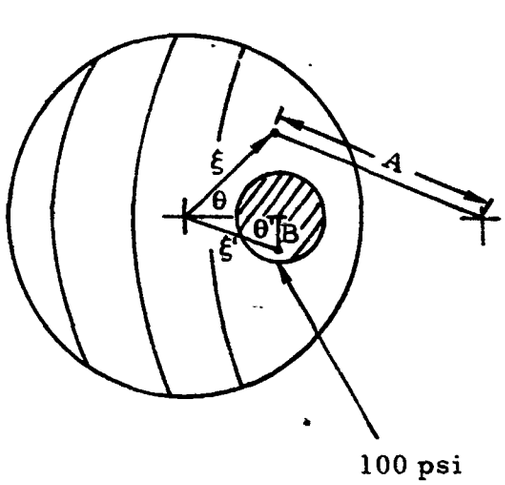
"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

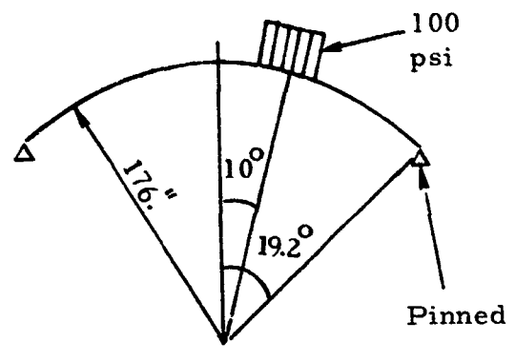
```
          .              .              .              .
          .              .              .              .
STATION     2
      N(XI)          N(THETA)       N(XI,THETA)        Q(XI)
  1.0485137E 03   -2.9560514E 04   0.0000000E-39   -9.2257202E 02

      M(XI)          M(THETA)       M(XI,THETA)      Q(THETA)
 -6.9193945E 02   -7.2390801E 03   0.0000000E-39   0.0000000E-39

STRESSES FOR LAYER    1
      SIG(XI)        SIG(THETA)     SIG(XI,THETA)    SIG(XI,ETA)    SIG(THETA,ETA)
  5.8375792E 04    6.5179318E 04   -0.0000000E-39   -1.2300967E 03   0.0000000E-39

STRESSES FOR LAYER    2
      SIG(XI)        SIG(THETA)     SIG(XI,THETA)    SIG(XI,ETA)    SIG(THETA,ETA)
  3.6717552E 03   -5.0065183E 04   0.0000000E-39   -6.1504833E 02   0.0000000E-39

STATION     1
      N(XI)          N(THETA)       N(XI,THETA)        Q(XI)
 -7.6171875E-02   -5.2677420E 04   0.0000000E-39   3.6267831E-02

      M(XI)          M(THETA)       M(XI,THETA)      Q(THETA)
 -1.1962891E-02   -7.0315019E 03   0.0000000E-39   -0.0000000E-39

STRESSES FOR LAYER    1
      SIG(XI)        SIG(THETA)     SIG(XI,THETA)    SIG(XI,ETA)    SIG(THETA,ETA)
  5.2945470E 04    3.2847100E 04   -0.0000000E-39   4.8357133E-02   -0.0000000E-39

STRESSES FOR LAYER    2
      SIG(XI)        SIG(THETA)     SIG(XI,THETA)    SIG(XI,ETA)    SIG(THETA,ETA)
  6.4952361E 03   -6.4419699E 04   0.0000000E-39   2.4178567E-02   -0.0000000E-39
```

3.7 SAMPLE PROBLEM 2

3.7.1 Problem Description

An Apollo-like shell pinned at the edge is considered.





Normalizing constants

$a_o = 1$
$h_o = 1$
$E_o = 1$
$\sigma_o = 1$

No Thermal Loads

Stiffness Parameter

$A = \xi^2 + 5625 - 2(75)\xi, \cos\theta$

if $A<4425$, $BBB = 3.296 \times 10^6$
$DDD = 3.296 \times 10^6$

if $4425<A<7225$, $BBB = 1.9779 \times 10^6$
$DDD = 1.9779 \times 10^6$

if $7225<A<11025$, $BBB = 1.386 \times 10^6$
$DDD = 1.386 \times 10^6$

if $A>11025$, $BBB = 0.79116 \times 10^6$
$DDD = 0.79116 \times 10^6$

Pressure parameters

Normal pressure definition

$$B = \xi^2 + (30.492)^2 - 2(30.492)\,\xi \cos\theta$$

if $B < 400$ PPPN = 100.0

if $B \geq 400$ PPPN = 0.0

also circumferential and meridional pressures definition

PPPH = PPPF = 0.0

Stress output parameters

$EI1 = EI2 = 29.5 \times 10^6$ Young's Modulus of stress output location

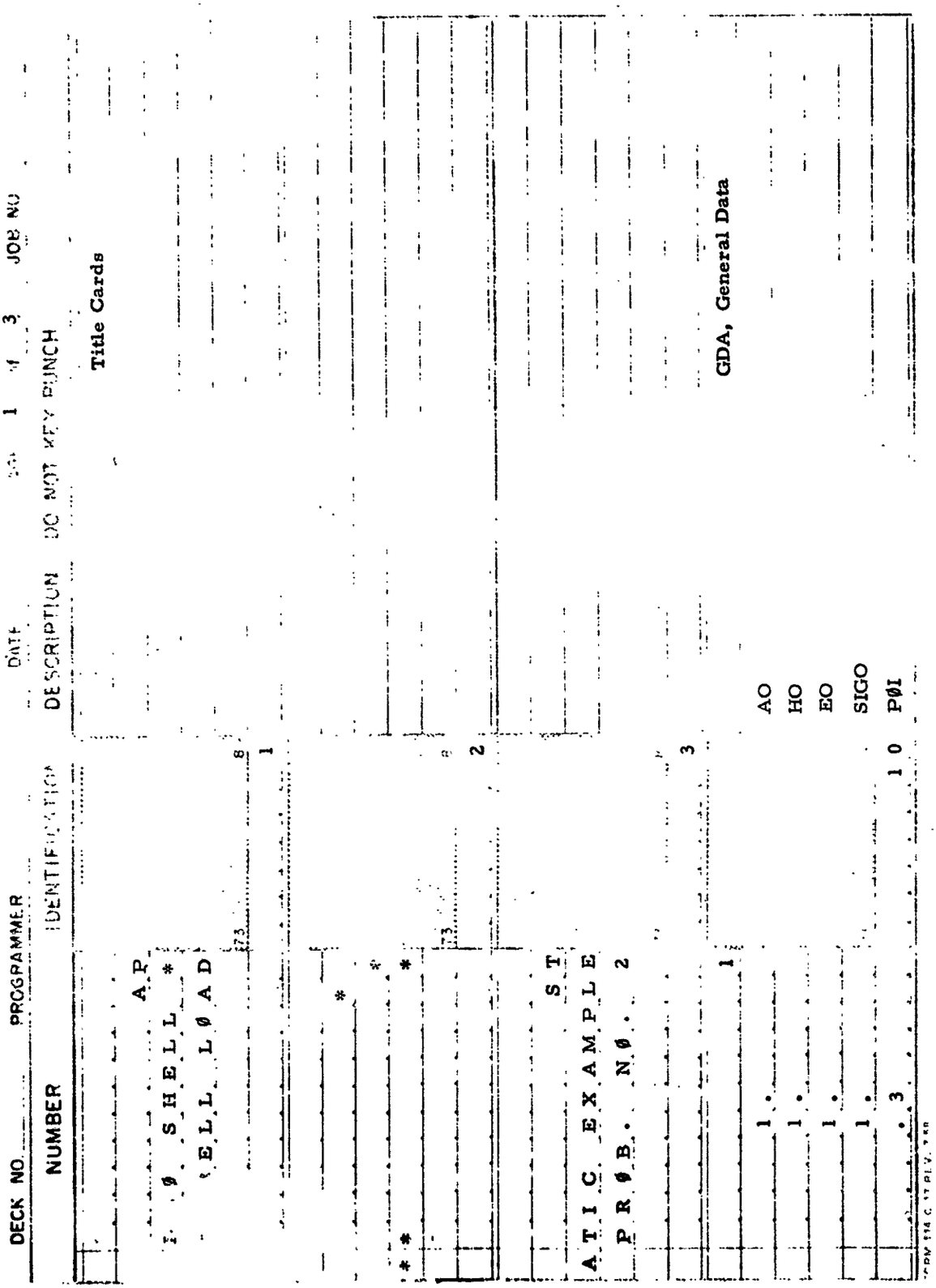
Distance to neutral surface

$$A = \xi^2 + 5625. - 2(75)\,\xi \cos\theta$$

if $A < 4425$	DN1 = -1.025,	DN2 = +1.025
if $4425 < A < 7225$,	DN1 = -1.005,	DN2 = +1.005
if $7225 < A < 11025$,	DN1 = -0.995,	DN2 = +0.995
if $A > 11025$,	DN1 = -0.987,	DN2 = +0.987

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

3.7.2 Data Sheets



DECK NO. _____ PROGRAMMER _____ DATE _____ PAGE 2 of 3 JOB NO. _____

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
6		
6.		ENF (No. of Fourier harmonics)
1.		BCIT (Closed apex)
2.		BCIB (Pinned)
-1.	73	PFLAG
	11	
12		
0.		THT (Thetas to print)
45.		
90.		
135.	73 80	
180.	12	
18		
6.		FPRNT (Fourier harmonics to print)
	73 80	
	13	
1		GMDA, Geometry Data
2.		GIN (Sphere)
26.		EN (No. of stations)
176.	73 80	RC
	20	RØFF

FORM 114-C-17 REV. 7-58

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

DECK NO. _____ PROGRAMMER _____ DATE _____ PAGE 3 OF 3 JOB NO. _____

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
6		
1 9 · 2		PHIO
		PHIN
	73 80	
	2 1	
	73 80	
	73 80	
	73 80	

FORM 114 C-17 REV. 7 58

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

3.7.3 Function Subprograms Used

```
$IBFTC BFCN                                                    APOLLO SHELL
C
      FUNCTION  BBB (PEL,ZTA)
C
      A  =  PEL * PEL + 5625. - 150. * PEL * COS( ZTA )
C
      IF (A .LE. 4225.)  GO TO 20
      IF (A .LE. 7225.)  GO TO 30
      IF (A .LE. 11025.) GO TO 35
C
      K  =  1
   10 BBB  =  0.79116F6
      GO  TO  100
C
   20 K  =  2
   25 BBB  =  3.296F6
      GO  TO  100
C
   30 K  =  3
      BBB  =  1.9779F6
      GO  TO  100
C
   35 K  =  4
   40 BBB  =  1.386F6
      GO  TO  100
C
      ENTRY  DDD (ZTA)
C
      IF (K - 2) 10, 25, 50
C
   50 IF (K .GT. 3)  GO TO 40
C
      BBB  =  1.9779F6
  100 RETURN
C
      END
```

```
$IBFTC TFCN                                                    00000001
C                                          TEMPERATURE FUNCTION 00000009
C                                                               00000019
      FUNCTION  ENTT( PEL,ZTA )                                 00000020
C                                                               00000021
      ENTT  =  0.                                               00000100
      RETURN                                                    00000110
C                                                               00000119
      ENTRY     EMTT( PEL,ZTA )                                 00000120
      ENTT  =  0.                                               00000200
      RETURN                                                    00000210
      END                                                       00000999
```

```
$IBFTC KFCN                                                    00000001
C                                          SPRING FUNCTION      00000009
C                                                               00000019
      FUNCTION  DKR3( PEL,ZTA )                                 00000020
C                                                               00000021
      DKR3  =  0.0                                              00000100
      RETURN                                                    00000110
C                                                               00000119
      END                                                       00000999
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

```
$IBFTC PFCN                                                          00000001
C                                                   BELL  LOAD       00000010
                                                                     00000019
      FUNCTION  PPPN (PFL,ZTA)                                       00000020
C                                                                    00000021
      A  =  PFL*PFL + 30.492*30.492 - 2.0 * 30.492 * PFL * COS( ZTA )  00000030
C                                                                    00000039
      PPPN  =  0.0                                                   00000040
      IF(A .LE. 400.0)  PPPN = 100.0                                 00000050
      RETURN                                                         00000060
C                                                                    00000069
      ENTRY  PPPH (PFL,ZTA)                                          00000100
      PPPN  =  0.0                                                   00000110
      RETURN                                                         00000120
C                                                                    00000129
      ENTRY  PPPF (PFL,ZTA)                                          00000130
      PPPN  =  0.0                                                   00000140
      RETURN                                                         00000150
C                                                                    00000199
      END                                                            00000200
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR;

```
$IBFTC STR1FN                                                          00000001
C       1ST SURFACE ** ELASTIC MODULUS, DISTANCE, POISSON'S RATIO      00000009
C                                                  APOLLO  SHELL       00000010
C                                                                      00000019
      FUNCTION  EI1 ( PEL,ZTA )                                        00000020
C                                                                      00000021
      EI1    =  29.5E6                                                 00000100
      RETURN                                                           00000110
C                                                                      00000119
      ENTRY     DN1 ( PEL,ZTA )                                        00000120
C                                                                      00000121
      A  =  PEL * PEL + 5625. - 150. * PEL * COS( ZTA )                00000125
C                                                                      00000129
      IF (A .LE. 4225.)  GO TO 20                                      00000130
      IF (A .LE. 7225.)  GO TO 30                                      00000132
      IF (A .LE. 11025.) GO TO 40                                      00000134
C                                                                      00000139
      EI1    =  -.987                                                  00000140
      GO  TO  100                                                      00000145
C                                                                      00000149
   20 EI1    =  -1.025                                                 00000150
      GO  TO  100                                                      00000155
C                                                                      00000159
   30 EI1    =  -1.005                                                 00000160
      GO  TO  100                                                      00000165
C                                                                      00000169
   40 EI1    =  -.995                                                  00000170
  100 RETURN                                                           00000210
C                                                                      00000219
      ENTRY     POIST( PEL,ZTA )                                       00000220
      EI1    =  0.3                                                    00000300
      RETURN                                                           00000310
      END                                                              00000999
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

```
$IBFTC STR2FN
C         2ND SURFACE ** ELASTIC MODULUS, DISTANCE, POISSON'S RATIO
C                                                    APOLLO. SHELL
C
      FUNCTION  EI2 ( PFL,ZTA )
C
      EI2    =   29.5E6
      RETURN
C
      ENTRY      DN2 ( PFL,ZTA )
C
      A  =  PEL * PFL + 5625. - 150. * PFL * COS( ZTA )
C
      IF (A .LE. 4225.)  GO TO 20
      IF (A .LE. 7225.)  GO TO 30
      IF (A .LE. 11025.) GO TO 40
C
      EI2  =    .987
      GO  TO  100
C
   20 EI2  =    1.025
      GO  TO  100
C
   30 EI2  =    1.005
      GO  TO  100
C
   40 EI2  =    .995
  100 RETURN
C
      ENTRY    POIS2( PFL,ZTA )
      EI2    =   0.3
      RETURN
      END
```

```
$IBFTC TTF1                                                          00000001
C          1ST SURFACE ** COEF. OF THERMAL EXPANSION, TEMPERATURE    00000009
C                                                                    00000019
      FUNCTION  TMP1( PFL,ZTA )                                      00000020
C                                                                    00000021
C                                                                    00000119
      ENTRY     ALF1( PFL,ZTA )                                      00000120
      TMP1  =  0.0                                                   00000200
      RETURN                                                         00000210
      END                                                            00000999
```

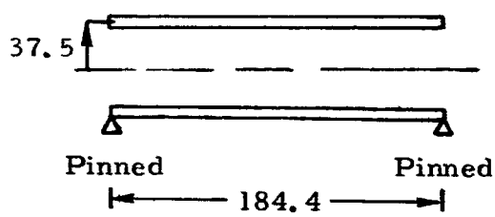
```
$IBFTC TTF2                                                          00000001
C          2ND SURFACE ** COEF. OF THERMAL EXPANSION, TEMPERATURE    00000009
C                                                                    00000019
      FUNCTION  TMP2( PFL,ZTA )                                      00000020
C                                                                    00000021
C                                                                    00000119
      ENTRY     ALF2( PFL,ZTA )                                      00000120
      TMP2  =  0.0                                                   00000200
      RETURN                                                         00000210
      END                                                            00000999
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

3.8 SAMPLE PROBLEM 3

3.8.1 Problem Description

A uniform cylinder under semisinusoidal time dependent loading is considered.



$\nu = 1/6$
$a_o = 1$
$h_o = 1$
$E_o = 1$
$\sigma_o = 1$

Stiffness Parameters

$BBB = 12.0925 \times 10^6$

$DDD = 16.56 \times 10^4$

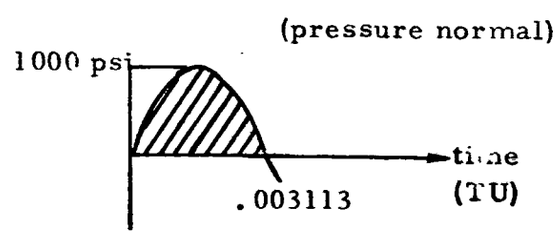
Mass properties $DMM1 = 3.49 \times 10^{-3}$

No thermal loading

No elastic foundation

No external damping

Pressure loading



if TU < .003113

PPPN = 1000 Sin (1003 TU)

if TU ≥ .003113

PPPN = 0.0
PPPH = PPPF = 0.0

DECK NO. _____ PROGRAMMER _____ DATE _____ PAGE 1 2 JOB NO. _____

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
		Title Cards
TRANSIENT RE		
SPØNSE ØF A		
SIMPLY SUPPØ		
RTED CYLINDE	73 80	
R	1	
TØ A UNIFØRM		
SINUSØIDAL		
PULSE		
* SHEN	73 80	
G	2	
* *		
*		
	73 80	
	3	
1		GDA, General Data
1.		AO
1.		HO
1.		EO
1.		SIGO
.1667	10	PØI

FORM 114-C-17 REV. 7-58

DECK NO. _____ PROGRAMMER _____ DATE _____ PAGE 2 of 2 JOB NO _____

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
6		
1.		ENF
2.		BCIT
2.		BCIB
-1.	73 80	PFLAG
3.	11	CEXT
-11		
5. -4		DELT
	73 80	
	12	
-1		GMDA, Geometry Data
1.		GIN (cylinder)
11.		EN
37.5	73	RAI
184.4	20	AXL ANX = 0.
	73 80	

FORM 114-C-17 REV. 7 58

```
$IBFTC DBBDD
C                                                    STIFFNESS FUNCTION
C
      FUNCTION BBB( PFL,ZTA )
C
      BBB  =  12.0925E+6
      RETURN
C
      ENTRY  DDD ( ZTA )
      BBB  =  16.56E+4
      RETURN
      END
```

```
00000001
00000009
00000019
00000020
00000021
00000100
00000110
00000119
00000120
00000200
00000210
00000999
```

```
$IBFTC DTMP
C                                                    TEMPERATURE FUNCTION
C
      FUNCTION  FNTT( PFL,ZTA )
C
C
      ENTRY     EMTT( PFL,ZTA )
      ENTT  =  0.
      RETURN
      END
```

```
00000001
00000009
00000019
00000020
00000021
00000119
00000120
00000200
00000210
00000999
```

```
$IBFTC DKDMP
C                                   SPRING AND DAMPING FUNCTION
C
      FUNCTION  DKK1( PEL,ZTA )
C
      ENTRY     DKK2( PEL,ZTA )
C
      ENTRY     DKK3( PEL,ZTA )
C
      ENTRY     DMP1( PEL,ZTA )
C
      ENTRY     DMP2( PEL,ZTA )
C
      ENTRY     DMP3( PEL,ZTA )
      DKK1  =  0.
      RETURN
      END
```

```
$IBFTC DMASS
C                                   MASS FUNCTION
C
      FUNCTION  DMM1( PEL,ZTA )
C
      DMM1 = 3.4900886E-3
      RETURN
C
      ENTRY     DMM4( PEL,ZTA )
      DMM1  =  0.
      RETURN
      END
```

```
$IBFTC DPRSS                                                    PRESSURE FUNCTION   00000001
C                                                                                   00000009
C                                                                                   00000019
      FUNCTION  PPPN( PEL,ZTA,TU)                                                   00000020
C                                                                                   00000021
      IF(TU .LT. 3.113E-3) GO TO 5                                                  00000030
      PPPN  =  0.                                                                   00000040
      GO  TO  10                                                                    00000050
    5 PPPN  =  1000. * SIN(1003. * TU)                                              00000100
   10 RETURN                                                                        00000110
C                                                                                   00000119
      ENTRY     PPPH( PEL,ZTA,TU )                                                  00000120
      PPPN  =  0.                                                                   00000200
      RETURN                                                                        00000210
C                                                                                   00000219
      ENTRY     PPPF( PEL,ZTA,TU )                                                  00000220
      PPPN  =  0.                                                                   00000300
      RETURN                                                                        00000310
      END                                                                           00000999
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

3.8.4 Output

```
TRANSIENT RESPONSE OF SIMPLY SUPPORTED CYLINDER TO UNIFORM
SINUSOIDAL PULSE                    * * *   SHENG
                                    * * *
```

GENERAL DATA

A0	=	1.0000E 00	H0	=	1.0000E 00	E0	=	1.0000E 00	SIG0	=	1.0000E 00
P0I	=	1.6670E-01	ENF	=	1.0000E 00	BCIT	=	2.0000E 00	BCIB	=	2.0000E 00
PFLAG	=	-1.0000E 00	CEXT	=	3.0000E 00	DELT	=	5.0000E-04			

PRINT ANGLES (THT) -

0.0000E-39 -1.0000E 10 -1.0000E 10 -1.0000E 10 -1.0000E 10 -1.0000E 10

FOURIER HARMONIC PRINTS (FPRNT) -

1.0000E 00 -1.0000E 10 -1.0000E 10 -1.0000E 10

BOUNDARY MATRICES

TOP BOUNDARY **

OMEGA

0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	1.0000000E 00	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	1.0000000E 00

LAMBDA

1.0000000E 00	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	1.0000000E 00	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	1.0000000E 00	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39

L -

0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39

BOTTOM BOUNDARY **

OMEGA

0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	1.0000000E 00	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	1.0000000E 00

LAMBDA

1.0000000E 00	0.0000000E-39	0.0000000E-39	0.000[illegible]-39	0.0000000E-39
0.0000000E-39	1.0000000E 00	0.0000000E-39	0.00[illegible]E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	1.0000000E 00	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39

L -

0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

GEOMETRY DATA FOR CONE - CYLINDER

NUMBER OF STATIONS - 11

RA1 = 3.7500E 01 AXL = 1.8440E 02 ANX = 0.0000E-39

DEL = 1.8440E 01

I	R(I)	XI(I)	W(THETA)	W(XI)	RHOX(I)	GAMA(I)
1	3.7500000E 01	0.0000000E-39	2.6666667E-02	0.0000000E-39	3.7500000E 01	0.0000000E-39
2	3.7500000E 01	1.8440000E 01	2.6666667E-02	0.0000000E-39	3.7500000E 01	0.0000000E-39
3	3.7500000E 01	3.6880000E 01	2.6666667E-02	0.0000000E-39	3.7500000E 01	0.0000000E-39
4	3.7500000E 01	5.5319999E 01	2.6666667E-02	0.0000000E-39	3.7500000E 01	0.0000000E-39
5	3.7500000E 01	7.3759998E 01	2.6666667E-02	0.0000000E-39	3.7500000E 01	0.0000000E-39
6	3.7500000E 01	9.2199998E 01	2.6666667E-02	0.0000000E-39	3.7500000E 01	0.0000000E-39
7	3.7500000E 01	1.1064000E 02	2.6666667E-02	0.0000000E-39	3.7500000E 01	0.0000000E-39
8	3.7500000E 01	1.2908000E 02	2.6666667E-02	0.0000000E-39	3.7500000E 01	0.0000000E-39
9	3.7500000E 01	1.4751999E 02	2.6666667E-02	0.0000000E-39	3.7500000E 01	0.0000000E-39
10	3.7500000E 01	1.6595999E 02	2.6666667E-02	0.0000000E-39	3.7500000E 01	0.0000000E-39
11	3.7500000E 01	1.8439999E 02	2.6666667E-02	0.0000000E-39	3.7500000E 01	0.0000000E-39

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

CHECKPRINT OF COEFC BLOCKS STORED IN GROUPS OF (2*ENF) PER MERIDIONAL STATION. THE COEFFICIENTS PRINTED ARE SCB1, SCD1, SCMTSI, SCMTSI, DCM1, DCM4, DCC1, DCC2, DCC3, DCK1, DCK2, DCK3 (IN THAT ORDER) AND ARE SEPARATED FROM EACH OTHER BY THE THREE ..... FIELDS.

FOR STATION 1 THROUGH STATION 11

1.2092497E 07	-2.2277777E 00	1.2092497E 07	-2.2277777E 00	1.2092497E 07	-2.2277777E 00
1.2092497E 07	-2.2277777E 00	1.2092497E 07	-2.2277777E 00	1.2092497E 07	-2.2277777E 00
1.2092497E 07	-2.2277777E 00	1.2092497E 07	-2.2277777E 00	1.2092497E 07	-2.2277777E 00
1.2092497E 07	-2.2277777E 00	1.2092497E 07	-2.2277777E 00	.....	.....
.....	1.6559999E 05	-2.9296874E-02	1.6559999E 05	-2.9296874E-02	1.6559999E 05
-2.9296874E-02	1.6559999E 05	-2.9296874E-02	1.6559999E 05	-2.9296874E-02	1.6559999E 05
-2.9296874E-02	1.6559999E 05	-2.9296874E-02	1.6559999E 05	-2.9296874E-02	1.6559999E 05
-2.9296874E-02	1.6559999E 05	-2.9296874E-02	1.6559999E 05	-2.9296874E-02	.....
.....	.....	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
.....	.....	.....	0.0000000E-39	-0.0000000E-39	0.0000000E-39
-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39
-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39
-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39
-0.0000000E-39	.....	.....	.....	3.4900880E-03	-6.1312067E-10
3.4900880E-03	-6.1312067E-10	3.4900880E-03	-6.1312067E-10	3.4900880E-03	-6.1312067E-10
3.4900880E-03	-6.1312067E-10	3.4900880E-03	-6.1312067E-10	3.4900880E-03	-6.1312067E-10
3.4900880E-03	-6.1312067E-10	3.4900880E-03	-6.1312067E-10	3.4900880E-03	-6.1312067E-10
3.4900880E-03	-6.1312067E-10	.....	.....	.....	0.0000000E-39
-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39
-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39
-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39
-0.0000000E-39	0.0000000E-39	-0.0000000E-39	.....	.....	.....
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	.....	.....
.....	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39
-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39
-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	.....
.....	.....	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
.....	.....	.....	0.0000000E-39	-0.0000000E-39	0.0000000E-39
-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39
-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39
-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39
-0.0000000E-39	.....	.....	.....	0.0000000E-39	-0.0000000E-39
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39
0.0000000E-39	-0.0000000E-39	.....	.....	.....	0.0000000E-39
-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39
-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39
-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39	-0.0000000E-39	0.0000000E-39
-0.0000000E-39	0.0000000E-39	-0.0000000E-39			

CHECKPRINT OF LOAD COEF. STORED IN GROUPS OF ALL ENF PER MERIDIONAL STATION. THE COEFFICIENTS ARE PFF, PTH, PN (IN THAT ORDER) AND ARE SEPARATED FROM EACH OTHER BY ..... FIELDS.

FOR STATION 1 THROUGH STATION 11

0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	.....
.....	.....	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	.....	.....	.....	4.8074125E 02	4.8074125E 02
4.8074125E 02	4.8074125E 02	4.8074125E 02	4.8074125E 02	4.8074125E 02	4.8074125E 02
4.8074125E 02	4.8074125E 02	4.8074125E 02			

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR;

"Z" MATRIX FOR STATION 11

0.0000000E-39	0.0000000E-39	0.0000000E-39	1.8837273E-04	0.0000000E-39

"Z" MATRIX FOR STATION 10

-5.4073139E-05	-0.0000000E-39	4.7842581E-03	1.4433218E-04	-0.0000000E-39

"Z" MATRIX FOR STATION 9

-1.7326856E-05	-0.0000000E-39	5.3232849E-03	1.2210523E-05	-0.0000000E-39

"Z" MATRIX FOR STATION 8

-2.9512694E-06	-0.0000000E-39	5.2341661E-03	-3.2122134E-06	-0.0000000E-39

"Z" MATRIX FOR STATION 7

-3.8498865E-07	-0.0000000E-39	5.2047550E-03	-8.4332169E-07	-0.0000000E-39

"Z" MATRIX FOR STATION 6

-9.0949470E-13	-0.0000000E-39	5.2030697E-03	-1.8189894E-12	-0.0000000E-39

"Z" MATRIX FOR STATION 5

3.8498592E-07	-0.0000000E-39	5.2047549E-03	8.4332896E-07	-0.0000000E-39

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

"Z" MATRIX FOR STATION 4

2.9512657E-06	-0.0000000E-39	5.2341662E-03	3.2122225E-06	-0.0000000E-39

"Z" MATRIX FOR STATION 3

1.7326853E-05	-0.0000000E-39	5.3232850E-03	-1.2210519E-05	-0.0000000E-39

"Z" MATRIX FOR STATION 2

5.4073140E-05	-0.0000000E-39	4.7842582E-03	-1.4433218E-04	-0.0000000E-39

"Z" MATRIX FOR STATION 1

-8.5192534E-05	-0.0000000E-39	-6.9652297E-06	1.7655379E-03	0.0000000E-39

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

STATION 11

THETA = 0.00

FNF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
0.0000000E-39	0.0000000E-39	0.0000000E-39	1.8837273E-04	0.0000000E-39

STATION 10

THETA = 0.00

FNF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
-5.4073139E-05	-0.0000000E-39	4.7842581E-03	1.4433218E-04	-0.0000000E-39

STATION 9

THETA = 0.00

FNF = 0.0

U	V	W	PHI(XI)	PHI(THETA)

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

-1.7326856E-05	-0.0000000E-39	5.3232849E-03	1.2210523E-05	-0.0000000E-39

STATION 8

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
-2.9512694E-06	-0.0000000E-39	5.2341661E-03	-3.2122134E-06	-0.0000000E-39

STATION 7

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
-3.8498865E-07	-0.0000000E-39	5.2047550E-03	-8.4332169E-07	-0.0000000E-39

STATION 6

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
-9.0949470E-13	-0.0000000E-39	5.2030697E-03	-1.8189894E-12	-0.0000000E-39

STATION 5

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
3.8498592E-07	-0.0000000E-39	5.2047549E-03	8.4332896E-07	-0.0000000E-39

STATION 4

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
2.9512657E-06	-0.0000000E-39	5.2341662E-03	3.2122225E-06	-0.0000000E-39

STATION 3

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
1.7376853E-05	-0.0000000E-39	5.3232850E-03	-1.2210519E-05	-0.0000000E-39

STATION 2

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
5.4073140E-05	-0.0000000E-39	4.7842582E-03	-1.4433218E-04	-0.0000000E-39

STATION 1

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
-8.5192534E-05	-0.0000000E-39	-6.9652297E-06	1.7655379E-03	0.0000000E-39

STATION 11

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
6.5236389E 01	1.0875239E 01	0.0000000E-39	-9.3807093E 02

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
-8.1676947E-08	-1.3615547E-08	0.0000000E-39	-0.0000000E-39

STATION 10

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
2.6285987E 02	1.5437102E 03	0.0000000E-39	-4.2890719E-02

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
7.9101019E-01	1.3186140E-01	0.0000000E-39	0.0000000E-39

STATION 9

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
2.6939186E 02	1.7137870E 03	0.0000000E-39	5.6839940E-02

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
6.6250940E-01	1.1044032E-01	0.0000000E-39	0.0000000E-39

STATION 8

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
2.7580846E 02	1.6869174E 03	0.0000000E-39	8.6695042E-03

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)

5.8614865E-02	9.7710979E-03	0.0000000E-39	0.0000000E-39

STATION 7

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
2.7381481E 02	1.6781980E 03	0.0000000E-39	-7.2522718E-04

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
-1.4423594E-02	-2.4044130E-03	0.0000000E-39	0.0000000E-39

STATION 6

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
2.7943943E 02	1.6777738E 03	0.0000000E-39	6.7393227E-06

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
-7.5734623E-03	-1.2624962E-03	0.0000000E-39	0.0000000E-39

STATION 5

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
2.7881480E 02	1.6781980E 03	0.0000000E-39	7.5393842E-04

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
-1.4423651E-02	-2.4044226E-03	0.0000000E-39	0.0000000E-39

STATION 4

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
2.7580848E 02	1.6869174E 03	0.0000000E-39	-8.6554708E-03

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
5.8614882E-02	9.7711007E-03	0.0000000E-39	0.0000000E-39

STATION 3

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
2.6935936E 02	1.7136088E 03	0.0000000E-39	-5.6823896E-02

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
6.6250943E-01	1.1044032E-01	0.0000000E-39	0.0000000E-39

STATION 2

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
2.9079345E 02	1.5483667E 03	0.0000000E-39	9.9445981E-01

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
-7.9825127E 00	-1.3306849E 00	0.0000000E-39	0.0000000E-39

STATION 1

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
1.4866468E 02	2.2598750E 01	0.0000000E-39	1.0785388E 04

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
-2.6320569E 01	-4.3876387E 00	-0.0000000E-39	0.0000000E-39

CHECKPRINT OF LOAD COEF. STORED IN GROUPS OF ALL ENF PER MERIDIONAL STATION. THE COEFFICIENTS ARE PFE, PTH, PN (IN THAT ORDER) AND ARE SEPARATED FROM EACH OTHER BY ..... FIELDS.

FOR STATION 1 THROUGH STATION 11

0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	.....
.....	.....	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	.....	.....	.....	8.4308790E 02	8.4308790E 02
8.4308790E 02	8.4308790E 02	8.4308790E 02	8.4308790E 02	8.4308790E 02	8.4308790E 02
8.4308790E 02	8.4308790E 02	8.4308790E 02			

'Z' MATRIX FOR STATION 11

0.0000000E-39	0.0000000E-39	0.0000000E-39	1.1038860E-03	0.0000000E-39

'Z' MATRIX FOR STATION 10

-6.3454623E-04	0.0000000E-39	2.5893420E-02	8.6705199E-04	0.0000000E-39

'Z' MATRIX FOR STATION 9

-3.7170339E-04	0.0000000E-39	3.1978566E-02	1.5654976E-04	0.0000000E-39

'Z' MATRIX FOR STATION 8

-1.3692552E-04	0.0000000E-39	3.1665081E-02	-2.2956723E-05	0.0000000E-39

'Z' MATRIX FOR STATION 7

-3.9743107E-05	0.0000000E-39	3.1131263E-02	-1.7755876E-05	0.0000000E-39

'Z' MATRIX FOR STATION 6

-0.0000000E-39	0.0000000E-39	3.1010200E-02	2.1827873E-11	0.0000000E-39

'Z' MATRIX FOR STATION 5

3.9743143E-05	0.0000000E-39	3.1131264E-02	1.7755898E-05	0.0000000E-39

```
'Z' MATRIX FOR STATION     4
     1.3692557E-04    0.0000000E-39    3.1665081E-02    2.2956723E-05    0.0000000E-39

'Z' MATRIX FOR STATION     3
     3.7170340E-04    0.0000000E-39    3.1978566E-02   -1.5654974E-04    0.0000000E-39

'Z' MATRIX FOR STATION     2
     6.3454624E-04   -0.0000000E-39    2.5893420E-02   -8.6705197E-04   -0.0000000E-39

'Z' MATRIX FOR STATION     1
    -2.1984192E-11    0.0000000E-39    1.1268639E-09   -1.1038859E-03   -0.0000000E-39
```

STATION 11

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
0.0000000E-39	0.0000000E-39	0.0000000E-39	1.1038860E-03	0.0000000E-39

STATION 10

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
-6.3454623E-04	0.0000000E-39	2.5893420E-02	8.6705199E-04	0.0000000E-39

STATION 9

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)

```
-3.7170339E-04    0.0000000E-39    3.1978566E-02    1.5654976E-04    0.0000000E-39


STATION    8

THETA  =    0.00
 ENF   =    0.0
              U                V                W              PHI(XI)          PHI(THETA)
      -1.3692552E-04    0.0000000E-39    3.1665081E-02   -2.2956723E-05    0.0000000E-39


STATION    7

THETA  =    0.00
 ENF   =    0.0
              U                V                W              PHI(XI)          PHI(THETA)
      -3.9743107E-05    0.0000000E-39    3.1131263E-02   -1.7755876E-05    0.0000000E-39


STATION    6

THETA  =    0.00
 ENF   =    0.0
```

U	V	W	PHI(XI)	PHI(THETA)
-0.0000000E-39	0.0000000E-39	3.1010200E-02	2.1827873E-11	0.0000000E-39

STATION 5

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
3.9743143E-05	0.0000000E-39	3.1131264E-02	1.7755898E-05	0.0000000E-39

STATION 4

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
1.3692557E-04	0.0000000E-39	3.1665081E-02	2.2956723E-05	0.0000000E-39

STATION 3

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
3.7170340E-04	0.0000000E-39	3.1978566E-02	-1.5654974E-04	0.0000000E-39

STATION 2

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
6.3454624E-04	-0.0000000E-39	2.5893420E-02	-8.6705197E-04	-0.0000000E-39

STATION 1

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
-2.1984192E-11	0.0000000E-39	1.1268639E-09	-1.1038859E-03	-0.0000000E-39

STATION 11

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
7.1036246E 02	1.1841742E 02	0.0000000E-39	-4.2191703E 03

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
-5.5540324E-07	-9.2585720E-08	0.0000000E-39	-0.0000000E-39

STATION 10

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
1.5137821E 03	8.3700780E 03	0.0000000E-39	-2.3069341E-01

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
4.2537650E 00	7.0910262E-01	0.0000000E-39	-0.0000000E-39

STATION 9

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
1.5558496E 03	1.0284818E 04	0.0000000E-39	2.5867319E-01

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
3.9963506E 00	6.6619164E-01	0.0000000E-39	-0.0000000E-39

STATION 8

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
1.5933162E 03	1.0192784E 04	0.0000000E-39	8.9983445E-02

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)

7.8267372E-01	1.3047171E-01	0.0000000E-39	-0.0000000E-39

STATION 7

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
1.6285702E 03	1.0031306E 04	0.0000000E-39	6.1323189E-03

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
-1.0308124E-01	-1.7183643E-02	0.0000000E-39	-0.0000000E-39

STATION 6

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
1.6408960E 03	9.9954070E 03	0.0000000E-39	4.6360177E-05

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
-1.5945630E-01	-2.6581365E-02	0.0000000E-39	-0.0000000E-39

STATION 5

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
1.6285702E 03	1.0031307E 04	0.0000000E-39	-6.0223427E-03

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
-1.0308104E-01	-1.7183610E-02	0.0000000E-39	-0.0000000E-39

STATION 4

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
1.5933164E 03	1.0192785E 04	0.0000000E-39	-8.9919292E-02

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
7.8267372E-01	1.3047171E-01	0.0000000E-39	-0.0000000E-39

STATION 3

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
1.5556607E 03	1.0283748E 04	0.0000000E-39	-2.5856462E-01

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
3.9963505E 00	6.6619163E-01	0.0000000E-39	-0.0000000E-39

STATION 2

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
1.5137822E 03	8.3700781E 03	0.0000000E-39	2.3058344E-01

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
4.2537646E 00	7.0910256E-01	0.0000000E-39	0.0000000E-39

STATION 1

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
7.1036217E 02	1.184174[illegible] 02	0.0000000E-39	4.2191705E 03

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
-1.4375143E-06	-2.3963362E-07	0.0000000E-39	-0.0000000E-39

CHECKPRINT OF LOAD COEF. STORED IN GROUPS OF ALL FNF PER MERIDIONAL STATION. THE COEFFICIENTS ARE PFE, PTH, PN (IN THAT ORDER) AND ARE SEPARATED FROM EACH OTHER BY ..... FIELDS.

FOR STATION 1 THROUGH STATION 11

0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	.....
.....	.....	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
0.0000000E-39	.....	.....	.....	9.9780282E 02	9.9780282E 02
9.9780282E 02	9.9780282E 02	9.9780282E 02	9.9780282E 02	9.9780282E 02	9.9780282E 02
9.9780282E 02	9.9780282E 02	9.9780282E 02			

'Z' MATRIX FOR STATION 11

0.0000000E-39	0.0000000E-39	0.0000000E-39	2.7384748E-03	0.0000000E-39

'Z' MATRIX FOR STATION 10

-2.1800071E-03	0.0000000E-39	6.0952131E-02	2.1914625E-03	0.0000000E-39

'Z' MATRIX FOR STATION 9

-1.5983047E-03	0.0000000E-39	8.0825035E-02	5.5042525E-04	0.0000000E-39

'Z' MATRIX FOR STATION 8

-7.3723757E-04	0.0000000E-39	8.1248075E-02	-4.1839783E-05	0.0000000E-39

'Z' MATRIX FOR STATION 7

-2.5419031E-04	0.0000000E-39	7.9279975E-02	-7.0573849E-05	0.0000000E-39

'Z' MATRIX FOR STATION 6

-0.0000000E-39	0.0000000E-39	7.8644956E-02	1.1641532E-10	0.0000000E-39

'Z' MATRIX FOR STATION 5

2.5419041E-04	0.0000000E-39	7.9279978E-02	7.0573937E-05	0.0000000E-39

'Z' MATRIX FOR STATION 4

7.3723778E-04	0.0000000E-39	8.1248075E-02	4.1839696E-05	0.0000000E-39

'Z' MATRIX FOR STATION 3

1.5983048E-03	0.0000000E-39	8.0825035E-02	-5.5042517E-04	0.0000000E-39

'Z' MATRIX FOR STATION 2

2.1800072E-03	-0.0000000E-39	6.0952132E-02	-2.1914624E-03	-0.0000000E-39

'Z' MATRIX FOR STATION 1

-4.9936943E-11	0.0000000E-39	4.3257842E-09	-2.7384744E-03	-0.0000000E-39

STATION 11

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
0.0000000E-39	0.0000000E-39	0.0000000E-39	2.7384748E-03	0.0000000E-39

STATION 10

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
-2.1800071E-03	0.0000000E-39	6.0952131E-02	2.1914625E-03	0.0000000E-39

STATION 9

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)

-1.5983047E-03	0.0000000E-39	8.0825035E-02	5.5042525E-04	0.0000000E-39

STATION 8

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
-7.3723757E-04	0.0000000E-39	8.1248075E-02	-4.1839783E-05	0.0000000E-39

STATION 7

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
-2.5419031E-04	0.0000000E-39	7.9279975E-02	-7.0573849E-05	0.0000000E-39

STATION 6

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
-0.0000000E-39	0.0000000E-39	7.8644956E-02	1.1641532E-10	0.0000000E-39

STATION 5

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
2.5419041E-04	0.0000000E-39	7.9279978E-02	7.0573937E-05	0.0000000E-39

STATION 4

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
7.3723778E-04	0.0000000E-39	8.1248075E-02	4.1839696E-05	0.0000000E-39

STATION 3

THETA = 0.00

ENF = 0.0

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

U	V	W	PHI(XI)	PHI(THETA)
1.5983048E-03	0.0000000E-39	8.0825035E-02	-5.5042517E-04	0.0000000E-39

STATION 2

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
2.1800072E-03	-0.0000000E-39	6.0952132E-02	-2.1914624E-03	-0.0000000E-39

STATION 1

THETA = 0.00

ENF = 0.0

U	V	W	PHI(XI)	PHI(THETA)
-4.9936943E-11	0.0000000E-39	4.3257842E-09	-2.7384744E-03	-0.0000000E-39

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

STATION 11

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
2.3351250E 03	3.8926534E 02	0.0000000E-39	-8.4685251E 03

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
-1.6335390E-06	-2.7231094E-07	0.0000000E-39	-0.0000000E-39

STATION 10

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
3.8005566E 03	1.9742384E 04	0.0000000E-39	-5.3272472E-01

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
9.8248624E 00	1.6378045E 00	0.0000000E-39	-0.0000000E-39

STATION 9

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
3.8716973E 03	2.5984509E 04	0.0000000E-39	5.1087612E-01

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)
1.0028058E 01	1.6716772E 00	0.0000000E-39	-0.0000000E-39

STATION 8

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
3.92[illegible]56E 03	2.61263178E 04	0.0000000E-39	2.7448915E-01

M(XI)	M(THETA)	M(XI,THETA)	Q(THETA)

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

2.7884335E 00	4.6483187E-01	0.0000000E-39	-0.0000000E-39

STATION 7

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
4.0199776E 03	2.5524842E 04	0.0000000E-39	4.8591152E-02
M(XI)	**M(THETA)**	**M(XI,THETA)**	**Q(THETA)**
-1.8787109E-01	-3.1318111E-02	0.0000000E-39	-0.0000000E-39

STATION 6

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
4.0608813E 03	2.5332579E 04	0.0000000E-39	2.0484360E-04
M(XI)	**M(THETA)**	**M(XI,THETA)**	**Q(THETA)**
-6.3378710E-01	-1.0565231E-01	0.0000000E-39	-0.0000000E-39

STATION 5

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
4.0199776E 03	2.5524843E 04	0.0000000E-39	-4.8151247E-02
M(XI)	**M(THETA)**	**M(XI,THETA)**	**Q(THETA)**
-1.8786965E-01	-3.1317871E-02	0.0000000E-39	-0.0000000E-39

STATION 4

N(XI)	N(THETA)	N(XI,THETA)	Q(XI)
3.9267862E 03	2.6126318E 04	0.0000000E-39	-2.7442041E-01

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

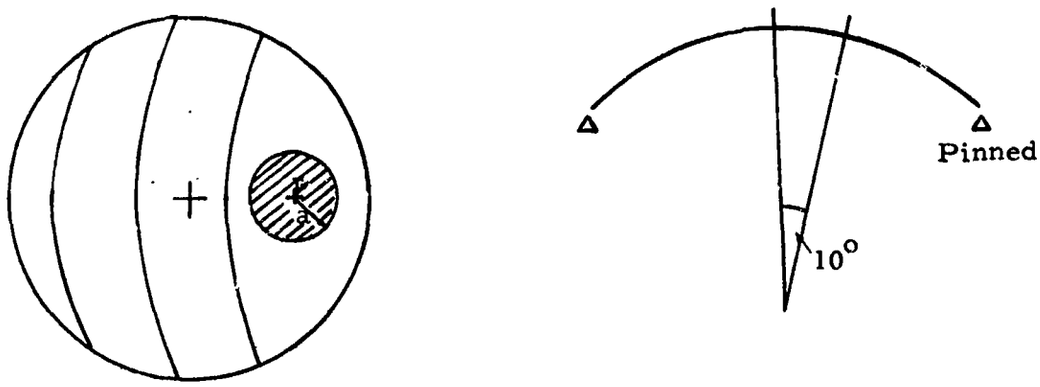
```
        M(XI)            M(THETA)         M(XI,THETA)        Q(THETA)
     2.7884336E 00    4.6483188E-01    0.0000000E-39    -0.0000000E-39

STATION     3

        N(XI)            N(THETA)         N(XI,THETA)        Q(XI)
     3.8712259E 03    2.5981806E 04    0.0000000E-39    -5.1056799E-01

        M(XI)            M(THETA)         M(XI,THETA)        Q(THETA)
     1.0028057E 01    1.6716771E 00    0.0000000E-39    -0.0000000E-39

STATION     2

        N(XI)            N(THETA)         N(XI,THETA)        Q(XI)
     3.8005567E 03    1.9742384E 04    0.0000000E-39     5.3257809E-01

        M(XI)            M(THETA)         M(XI,THETA)        Q(THETA)
     9.8248610E 00    1.6378043E 00    0.0000000E-39     0.0000000E-39

STATION     1

        N(XI)            N(THETA)         N(XI,THETA)        Q(XI)
     2.3351237E 03    3.8926515E 02    0.0000000E-39     8.4685262E 03

        M(XI)            M(THETA)         M(XI,THETA)        Q(THETA)
    -4.7045922E-06   -7.8425551E-07    0.0000000E-39    -0.0000000E-39


END EXECUTION     32                                   00-09-20
D06961FURUIKE     8302517-11   980            00-09-22   03/03/77
JOB TOTALS=     **PRINTING=   228 LINES **PUNCHING=     7 CARDS    **CRT=     00000-35MM   00000-9INCH
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

3.9 SAMPLE PROBLEM 4

3.9.1 Problem Description

The Apollo-like shell of sample problem 1, Section 3.7 is now subjected to hydrodynamic pressures compacted from a rigid shell solution.



All the input parameters are the same as sample problem 1 except for the subroutine describing pressures.

V_o	= 360 in./s	Initial impact velocity
R_c	= 175.6 in.	Radius of curvature
ρ_w	= 62.5/17280 #/in.^{3}	Fluid density
W_T	= 10,000	Weight of body

inside wetted region

$$PPPN = \frac{\sqrt{2}\,\rho_w V_o^{3/2} R_c^{1/2}\left[1 - \gamma(2 - 3r^2/a^2)\right]}{\pi t^{1/2}(1 - r^2/a^2)^{1/2}(1 + \gamma)^2}$$

where

$$t = \frac{1}{R_c V_o}\left(\frac{3W_T}{4\sqrt{2}\,\rho_w}\right)^{2/3}, \qquad \gamma = \frac{8\sqrt{2}\,\rho_w\left(R_c V_o t\right)^{3/2}}{3W_T}$$

outside wetted region

$$PPPN = 0.0$$

also

$$PPPH = PPPF = 0.0$$

The mass properties of the Apollo-like shell.

$$DMM1 = 1.4648 \times 10^{-3}(t_f) + 4.7443 \times 10^{-5}$$

where

$$A = \xi^2 + 5625 - 2(75)\xi \cos\theta$$

$A < 4225$ $\quad t_f = 0.05$

$4225 < A < 7225$ $\quad t_f = 0.03$

$7225 < A < 11025$ $\quad t_f = 0.02$

$A > 11025$ $\quad t_f = 0.012$

DECK NO. _____ PROGRAMMER _____ DATE _____ PAGE 1 2 JOB NO. _____

NUMBER	IDENTIFICATION	DESCRIPTION	DO NOT KEY PUNCH
A PØLLØ SHELL			Title Cards
* UNSYMMETR			
I C RIGID BØD			
Y HYDRØDYNAM			
I CS PRESSURE			
*	1		
TEN DEGRE			
E ØFFSET CAS			
E			
	73 80		
	2		
REFEREN			
C E * FØURTH			
Q UARTERLY RE			
PØRT	73 80		
	3		
1			GDA, General Data
1		A0	
1		H0	
1		E0	
1	73 80	SIG0	
3	10	PØI	

FORM 114-C-17 REV. 7-58

DECK NO. _____ PROGRAMMER _____ DATE _____ PAGE 2 of 2 JOB NO. _____

NUMBER	IDENTIFICATION	DESCRIPTION DO NOT KEY PUNCH
6		GDA, continued
1.		ENF (No. of Fourier harmonics)
1.		BCIT (Closed apex)
2.		BCIB (Pinned)
-1.	73 8	PFLAG
6.	11	CEXT (No. of time cycles)
-11		
5. -4		DELT (Time increment)
	73 80	
	12	THT & FPRNT are set by the program.
1		GMDA, Geometry Data
2.		GIN (Sphere)
26.		EN (No. of stations)
176.	73 80	RC
	20	RØFF
-7		
19.2		PHIN
		(PHIO = 0.0)
	73 80	
	21	

FORM 114-C-17 REV. 7-58

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR;

3.9.3 Function Subprogram Used

```
$IBFTC DBEOD                                                            00000001
C                                                   APOLLO  SHELL       00000010
C                                                                       00000011
      FUNCTION  BBB (PFL,ZTA)                                           00000015
C                                                                       00000019
      A  =  PFL * PFL + 5625. - 150. * PFL * COS( ZTA )                 00000020
C                                                                       00000029
      IF (A .LE. 4225.)  GO TO 20                                       00000030
      IF (A .LE. 7225.)  GO TO 30                                       00000040
      IF (A .LE. 11025.) GO TO 35                                       00000050
C                                                                       00000059
      K  =  1                                                           00000060
   10 BBB  =  0.79116E6                                                 00000070
      GO  TO  100                                                       00000080
C                                                                       00000089
   20 K  =  2                                                           00000090
   25 BBB  =  3.296E6                                                   00000100
      GO  TO  100                                                       00000110
C                                                                       00000119
   30 K  =  3                                                           00000120
      BBB  =  1.9779E6                                                  00000130
      GO  TO  100                                                       00000140
C                                                                       00000149
   35 K  =                                                              00000150
   40 BBB  =  1.386E6                                                   00000160
      GO  TO  100                                                       00000170
C                                                                       00000179
      ENTRY  DDD (ZTA)                                                  00000180
C                                                                       00000189
      IF (K - 2) 10, 25, 50                                             00000190
C                                                                       00000199
   50 IF (K .GT. 3)  GO TO 40                                           00000200
C                                                                       00000209
      BBB  =  1.9779E6                                                  00000210
  100 RETURN                                                            00000999
C                                                                       00001009
      END                                                               00001010
```

```
$IBFTC DTMP                                                    TEMPERATURE FUNCTION
C
C
      FUNCTION  FNTT( PEL,ZTA )
C
C
      ENTRY     EMTT( PEL,ZTA )
      FNTT  =  0.
      RETURN
      END
```

```
$IBFTC DKDMP                                                   SPRING AND DAMPING FUNCTION
C
C
      FUNCTION  DKK1( PEL,ZTA )
C
      ENTRY     DKK2( PEL,ZTA )
C
      ENTRY     DKK3( PEL,ZTA )
C
      ENTRY     DMP1( PEL,ZTA )
C
      ENTRY     DMP2( PEL,ZTA )
C
      ENTRY     DMP3( PEL,ZTA )
      DKK1  =  0.
      RETURN
      END
```

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR;"

```
$IBFTC DMASS                                                            00000001
C                                                   MASS FUNCTION       00000009
C                                                 APOLLO  SHELL         00000010
C                                                                       00000019
      FUNCTION  DMM1( PEL,ZTA )                                         00000020
C                                                                       00000021
      A  =  PEL * PEL + 5625. - 150. * PEL * COS( ZTA )                 00000125
C                                                                       00000129
      IF (A .LE. 4225.)  GO TO 20                                       00000130
      IF (A .LE. 7225.)  GO TO 30                                       00000132
      IF (A .LE. 11025.) GO TO 40                                       00000134
C                                                                       00000139
      TF  =  .012                                                       00000140
      GO  TO  100                                                       00000145
C                                                                       00000149
   20 TF  =  .05                                                        00000150
      GO  TO  100                                                       00000155
C                                                                       00000159
   30 TF  =  .03                                                        00000160
      GO  TO  100                                                       00000165
                                                                        00000169
   40 TF  =  .02                                                        00000170
  100 DMM1  =  1.4648033E-3 * TF + 4.7443145E-5                         00000180
      RETURN                                                            00000210
C                                                                       00000219
      ENTRY     DMM4( PEL,ZTA )                                         00000220
      DMM1  =  0.                                                       00000300
      RETURN                                                            00000310
      END                                                               00000999
```

```
$IBFTC DPRSS
C          PRESSURE FUNCTION  **  RIGID BODY HYDRODYNAMICS PRESSURE **
C
      FUNCTION PPPN (PFL,ZTA,TU)
C
      COMMON    GDA(25), DFL
C
C                 UNSYMMETRIC PRESSURE CASE * 10 DEGREE OFFSET *
C                 CONSTANTS SET TO APOLLO IMPACT CASE
C
C     VIN     INITIAL IMPACT VELOCITY, IN /SEC
      VIN  =  360.0
C     R       RADIUS OF CURVATURE
      R  =  175.6
C     RHO     FLUID DENSITY, LBS /CU IN
      RHO  =  62.5 /1728.0
C     WT      WEIGHT OF BODY, LBS
      WT  =  10000.0
C
      RMAX  =  SQRT (2.0 * TU * R * VIN)
      A3  =  1.3333 * RHO * RMAX **3 /WT
      BAF  =  A3 + 1.0
      A1  =  2.0 * R * VIN **2 * RHO /(3.1415927 * RMAX * 386.4 *BAF**2)
      A  =  2.0 * RHO * RMAX **3 /WT
      B  =  A1 * RMAX * (1.0 - 4.0 * A /3.0)
      SII  =  RMAX - DFL
      SI  =  SQRT (DFL **2 + 929.762 - 60.984 * PFL * COS(ZTA))
      ROC  =  SI /RMAX
C
      IF(SI .GT. RMAX) GO TO 40
C
      IF(SI .LT. RMAX .AND. SI .GT. SII) GO TO 30
C
   20 PPPN  =  -A1 *(1.0 - A3 *(2.0 - 3.0 * ROC **2)) /SQRT(1.0 -ROC**2)
      GO  TO  50
C
C                        WETTED BOUNDARY, PRESSURE AVERAGED OVER DEL
C
   30 RORM  =  SII /RMAX
      PPPN  =  -B *((1.0 + A) * 3.1415927 /2.0 - ((1.0 + A) * ARSIN
     1  (RORM) - A * RORM * SQRT(1.0 - RORM **2))) /DEL
      GO  TO  50
C
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

```
00000260
00000269
00000270
00000279
00000280
00000290
00000300
00000309
00000310
00000320
00000330
00000500
   40 PPPN   =   0.0
C
   50 RETURN
C
      ENTRY    PPPH (PFL,ZTA,TU)
      PPPN   =   0.0
      RETURN
C
      ENTRY    PPPF (PFL,ZTA,TU)
      PPPN   =   0.0
      RETURN
      END
```

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR;

3.10 ERROR INDICATIONS, PITFALLS, RECOMMENDATIONS

Several of the error indications resulting from improper data input have already been discussed. To reiterate, they were:

1. A bad index on a DECRD card (Section 3.5.1)

2. Omission of the negative sign on the last card of a data array (Section 3.5.1)

3. Omission of some or all of the title cards (Section 3.3.1).

One should be very careful to check the output from the program to see that it corresponds to the input that he entered. Better yet, an independent check of input data may prevent a wasted run on the digital computer.

The amount of data entered by the DECRD routine is relatively small but such things as sign convention, angle measurements, and compatibility of units are common pitfalls.

Many of the needed parameters are supplied to the program by use of FUNCTION subprograms. Before writing such subprograms, Section 3.4, 3.6.3, 3.7.3, 3.8.3 and 3.9.3 should be studied. A check list is given below.

1. Are the FUNCTION and ENTRY names spelled the same as those given in the table of Section 3.4?

2. Does the number of arguments agree with the number in the calling program? Note that ENTRY DDD has just one argument, (ZTA), and the pressure functions for the Dynamics version require the added argument TU.

3. Has the FUNCTION name been used to return the function value to the calling program even when reference was made to an ENTRY name?

4. Does the FUNCTION subprogram contain at least one RETURN statement and an END statement?

5. If additional data were needed to define the function have the data cards been included with the data deck?

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR;

3.11 PROGRAM LISTINGS FOR STATIC VERSION

3.11a Main Program

```
$IBFTC MAIN
C          APOLLO WATER IMPACT ANALYSIS                    **          **
C
C  REFERENCES ** AIAA JOURNAL, VOL. 1, NO. 8, AUG. 1963, PG.1833FF AND
C                VOL. 2, NO. 3, MARCH 1964, PG. 590FF
C              ** SID 65-1633 REPORT, APOLLO WATER IMPACT ANALYSIS, NOV.
C                23, 1965, PG. 81FF
C
C     NOMENCLATURE
C          AO        REFERENCE LENGTH (IN)
C          HO        REFERENCE THICKNESS (IN)
C          EO        REFERENCE YOUNG MODULUS (PSI)
C          SIGO      REFERENCE STRESS (PSI)
C          POI       POISSON RATIO
C          ENF       NO. OF FOURIER COMPONENTS
C          BCIT      TOP BOUNDARY CONDITION FLAG
C          BCIB      BOTTOM BOUNDARY CONDITION FLAG.
C                  = 1.0     CLOSED APEX
C                  = 2.0     PINNED
C                  = 3.0     CLAMPED
C                  = 4.0     FREE
C                  = 5.0     ROLLER W
C                  = 6.0     ROLLER U
C                  = 10.0    SPECIAL BOUNDARY READ WITH GEOMETRY DATA
C          PFLAG     DATA PRINT INDICATOR
C                  = 0.0     PRINTS GENERAL DATA AND BND. MATRICES
C                  = 1.0     PRINTS ABOVE INFORMATION AND INPUT DATA
C                            FOR A PARTICULAR GEOMETRY, AND THE PARA-
C                            METERS COMPUTED IN  GEOM
C                  =-1.0     PRINTS ALL OF THE ABOVE INFORMATION AND
C                            STIFFNESS COEFFICIENTS, ELASTIC COEF.,
C                            THERMAL LOAD AND MOMENT AND PRESSURES
C          THT       CIRCUMFERENTIAL ANGLE THETA (DEGREES)  5 MAX
C          FPRNT     FOURIER COMPONENT PRINT VALUES. 3 PERMITTED
C
C     TAPE USAGE -
C          TAPE  3 -      THREE FILES.
C                         FIRST FILE  - FOURIER COEFFICIENTS FOR LOAD.
C                         SECOND FILE - INITIAL BOUNDARY MATRICES
C                                       GENERATED IN 'RSLT'.
C                         THIRD FILE  - MATRICES FROM STATION NO. 1.
C
C          TAPE  4        TWO FILES.
```

```
C                         FIRST FILE  - A RECORD OF EACH REGION STUDIED 00000310
C                                       IN 'GEOMETRY'.                  00000311
C                                       A LAST RECORD CONTAINING A 'P'  00000330
C                                       AND AN 'X' GENERATED FOR        00000331
C                                       STATION (N-2).  THIS RECORD IS  00000332
C                                       WRITTEN IN FORMTX AND BACKSPACED.00000333
C                         SECOND FILE - A 'Z' MATRIX FOR EACH STATION.  00000360
C                                                                       00000369
C          TAPE  9 -      ONE FILE.                                     00000370
C                                       STIFFNESS COEFFICIENTS GENERATED00000380
C                                       IN 'DATLYR'.                    00000390
C                                                                       00000399
C          TAPE 10 -      ONE FILE.                                     00000400
C                                       EACH RECORD CONTAINS 'P' AND 'X'00000410
C                                       FOR EACH STATION.  LAST RECORD  00000411
C                                       CONTAINS ONLY R, S, AAO, AND    00000412
C                                       L1 FOR LAST STATION.            00000413
C                                                                       00000420
      DIMENSION   BCD(36), THT(6), FPRNT(3)                             00000430
C                                                                       00000450
      EQUIVALENCE (GDA( 1),   AO),                (GDA( 2),   HO),      00000460
     1            (GDA( 3),   EO),                (GDA( 4), SIGO),      00000470
     2            (GDA( 5),  POI),                (GDA( 6),  ENF),      00000480
     3            (GDA( 7), BCIT),                (GDA( 8), BCIB),      00000481
     4            (GDA( 9),PFLAG),                                      00000482
     5                                            (GDA(12),  THT),      00000483
     6            (GDA(18),FPRNT)                                       00000484
C                                                                       00000500
      COMMON  GDA(25), DEL, N, NNF, NTH,                                00000510
     1 BMTX(160)                                                        00000520
C                                                                       00000530
    5 REWIND  3                                                         00000540
      REWIND  10                                                        00000545
      REWIND  4                                                         00000550
      REWIND  9                                                         00000560
C                                 READ AND PRINT 3 CARDS OF TITLE DATA  00000561
      READ (5,7)  BCD                                                   00000562
    7 FORMAT( 12A6 )                                                    00000563
      WRITE (6,8) BCD                                                   00000564
    8 FORMAT(1H1 /(18X, 12A6 //) )                                      00000565
C                                                                       00000569
      DO 10 L = 1,11                                                    00000570
      GDA(L+11)  =  -1.E+10                                             00000575
```


"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR"

```
   10 GDA(L) = 0.0                                                      00000580
C                                                                       00000590
C                                  READ GENERAL DATA                    00000599
   20 CALL DECRD (GDA)                                                  00000600
      IF(THT(1) .EQ. -1.E+10)  THT(1)  =  0.0                           00000602
      IF(FPRNT(1) .EQ. -1.E+10)  FPRNT(1)  =  ENF                       00000603
C                                                                       00000610
C                                     DATA  LINK                        00000619
      CALL DATLNK                                                       00000620
C                                                                       00000680
C                                FORM 'P' AND 'X' MATRICES              00000689
      CALL PANDX                                                        00000690
      REWIND  9                                                         00000700
C                                                                       00000710
C                                       SOLUTIONS                       00000719
      CALL ZMTRX                                                        00000720
C                                                                       00000730
C                              SUMS AND INTERNAL LOADS                  00000739
      CALL SUMS                                                         00000740
C                                                                       00000750
      GO  TO  5                                                         00000760
C                                                                       00009999
      END                                                               00010000
```

"REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR."

3.11b DATLNK Subroutine (DLNK)

```
$IBFTC DLNK                                                          00000001
C      SUB-EXECUTIVE ROUTINE FOR GEOM, DATLYR AND DATLD TO PERMIT ONE 00000010
C         COPY OF CODIMA, ENTERP AND DINTRP IN THE DECK SET-UP       00000011
C                                                                    00000019
       SUBROUTINE DATLNK                                             00000020
C                                                                    00000099
       COMMON   GDA(25), DEL, N, NNF, NTH, BMTX(160)                 00000100
C                                                                    00000198
C                                  SET UP R, RHOX, GAMA, WTH, WFE, XSI 00000199
       CALL  GEOM                                                    00000200
       BACKSPACE  4                                                  00000205
C                                                                    00000218
C                                        SET UP MATERIAL PROPERTIES  00000219
       CALL  DATLYR                                                  00000220
       REWIND  9                                                     00000225
C                                                                    00000238
C                                        SET UP FOURIER LOAD COEFF.  00000239
       CALL  DATLD                                                   00000240
C                                                                    00000999
       RETURN                                                        00001000
C                                                                    00001009
       END                                                           00001010
```

```
$IBFTC GMTRY
C     * *  APOLLO WATER IMPACT STUDY    * *     GEOMETRY COMPUTATION **
C
      SUBROUTINE GEOM
C
C        GMI    -  GEOMETRY INDICATOR                  VIZ.     GIN
C               =  1.0  - CONE - CYLINDER                        **
C               =  2.0  - SPHERE - TOROID                        **
C               =  3.0  - ARBITRARY FUNCTIONS                    **
C               =  4.0  - GENERAL DISCRETE POINTS                **
C                 **                                             **
C        INPUTS -     GMI = 1.0
C                 RA1    =   RADIUS AT STATION 1
C                 AXL    =   AXIAL SURFACE LENGTH
C                 ANX    =   ANGLE - GENERATOR AND AXIS OF REVOLUTION
C                 **                                             **
C                      GMI = 2.0
C                 RC   =  RADIUS OF CURV
C                 ROFF   =  OFFSET DISTA  TO CENTER OF CURVATURE
C                 PHIO   = INITIAL  OPENING ANGLE FROM VERTICAL AXIS
C                 PHIN   =  FINAL ANGLE
C                 **                                             **
C                     GMI  = 4.0  (- 4.0 DISCRETE ARCLENGTHS )
C                 EM     =  NUMBER OF RIPTS GIVEN
C                 RIPT   =  DISCRETE RADII
C                 XIPT   =  DISCRETE XI'S (OR ARCLENGTHS)
C                 RCURV  =  MERIDIONAL RADII OF CURVATURE
C                 RCURZ  =  CIRCUMFERENTIAL RADII OF CURVATURE
C                 **                                             **
C
      DIMENSION WFW(16), WFP(4), STAP(4), STAW(16) , RRJ(12), RCRV(100),
     1RCURZ(100), RCURV(100), RR(100), RIPT(100),XIPT(100), SARB(100),
     2SURF(100), XJ(11), RJ(11), DLR(10), R(100), XSI(100), WTHX(100),
     3WFEX(100), RHOX(100), GAMA(100), WFEPX(100), RCRZ(100), GMDA(442)
C
      DIMENSION BNDTX(17), BNDTB(17), THT(6), FPRNT(4)
      DIMENSION  OMGO(5,5),  OMGN(5,5), ALMO(5,5), ALMN(5,5)
      DIMENSION  ALMI(5,5), OMGI(5,5)
      REAL  LLO(5),  LLLN(5)
C
      COMMON  GDA(25), DEL, N, NNF, NTH,
     1 BMTX(160)
C
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      EQUIVALENCE  (BMTX(  1), OMG0),                (BMTX( 26), OMGN),
     1             (BMTX( 51), ALM0),                (BMTX( 76), ALMN),
     2             (BMTX(101), OMG1),                (BMTX(126), ALM1),
     3             (BMTX(151),  LL0),                (BMTX(156), LLLN)
C
      EQUIVALENCE (GDA( 1),    A0),              (GDA( 2),    H0),
     1            (GDA( 3),    E0),              (GDA( 4), SIG0),
     2            (GDA( 5),   P0I),              (GDA( 6),  ENF),
     3            (GDA( 7), BCIT),               (GDA( 8), BCIB),
     4            (GDA( 9),PFLAG),
     5                                           (GDA(12),  THT),
     6            (GDA(18),FPRN))
C
      EQUIVALENCE  (GMDA(   1),        GIN),  (GMDA(   2),        FN),
     1             (GMDA(   3),      BLANK),  (GMDA(   4),     RA1,RC),
     2             (GMDA(   5),   AXL,ROFF),  (GMDA(   6),   ANX,PHI0),
     3             (GMDA(   7),      PHIN),  (GMDA(   8),        EM),
     4             (GMDA(   9),      RIPT),  (GMDA(109),       XIPT),
     5             (GMDA(209),      RCURV),  (GMDA(309),      RCURZ),
     6             (GMDA(409),      BNDTX),  (GMDA(426),      BNDTB)
C
      DO  2  I = 1,442
    2 GMDA(I) = 0.0
C
      CALL DECRD (GMDA)
C
      DO  3 I = 1,160
    3 BMTX(I) = 0.0
C
      IF (BCIT .NE. 1.0) GO TO 5
C                                          CLOSED APEX
      OMG0(3,3) = 1.0
      OMG1 (1,1) = 1.0
      OMG1 (1,2) = -1.0
      OMG1 (5,4) = -1.0
      OMG1 (5,5) = 1.0
      ALM0(1,1) = 1.0
      ALM0(2,2) = 1.0
      ALM0(4,4) = 1.0
      ALM0(5,5) = 1.0
      ALM1 (2,1) = 1.0
      ALM1 (2,2) = 1.0
      ALM1(3,3) = 1.0
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      ALM1(4,4) = 1.0                                                   00000830
      ALMI (4,5) = 1.0                                                  00000840
      GO TO 21                                                          00000850
C                                                                       00000859
    5 IF (BCIT .NE. 10.) GO TO 10                                       00000860
C                                                   SPECIAL BOUNDARY    00000870
      OMG0(1,1) = BNDTX( 1)                                             00000880
      OMG0(1,2) = BNDTX( 2)                                             00000890
      OMG0(2,2) = BNDTX( 3)                                             00000900
      OMG0(3,3) = BNDTX( 4)                                             00000910
      OMG0(4,4) = BNDTX( 5)                                             00000920
      OMG0 (5,4) = BNDTX( 6)                                            00000930
      OMG0(5,5) = BNDTX( 7)                                             00000940
      ALMO(1,1) = BNDTX( 8)                                             00000950
      ALMO(2,2) = BNDTX( 9)                                             00000960
      ALMO(3,3) = BNDTX(10)                                             00000970
      ALMO(4,4) = BNDTX(11)                                             00000980
      ALMO(5,5) = BNDTX(12)                                             00000990
      DO  8 IJ = 1,5                                                    00001000
    8 LLO(IJ) = BNDTX(IJ+12)                                            00001010
      GO  TO  21                                                        00001020
C                                                                       00001029
   10 IBMX  =  BCIT                                                     00001030
      TO  =  1                                                          00001040
      IL  =  51                                                         00001050
C                                                   CHOOSE BOUNDARY     00001060
   11 GO TO (21,12,13,14,15,16), IBMX                                   00001070
C                                                   PINNED              00001080
   12 BMTX(IO + 18)  =  1.                                              00001090
      BMTX(IO + 24)  =  1.                                              00001100
      BMTX(IL)  =  1.                                                   00001110
      BMTX(IL + 6)  =  1.                                               00001120
      BMTX(IL + 12)  =  1.                                              00001130
      GO  TO  20                                                        00001140
C                                                   CLAMPED             00001150
   13 BMTX(IL)  =  1.                                                   00001160
      BMTX(IL + 6)  =  1.                                               00001170
      BMTX(IL + 12)  =  1.                                              00001180
      BMTX(IL + 18)  =  1.                                              00001190
      BMTX(IL + 24)  =  1.                                              00001200
      GO  TO  20                                                        00001210
C                                                   FREE                00001230
   14 BMTX(IO)  =  1.                                                   00001240
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR.

```
      BMTX(IO + 6)   =  1.
      BMTX(IO + 12)  =  1.
      BMTX(IO + 18)  =  1.
      BMTX(IO + 24)  =  1.
      GO  TO  20
C                                                  ROLLER  W
   15 BMTX(IO + 12)  =  1.
      BMTX(IO + 18)  =  1.
      BMTX(IO + 24)  =  1.
      BMTX(IL)  =  1.
      BMTX(IL + 6)  =  1.
      GO  TO  20
C                                                  ROLLER  U
   16 BMTX(IO)  =  1.
      BMTX(IO + 18)  =  1.
      BMTX(IO + 24)  =  1.
      BMTX(IL + 6)  =  1.
      BMTX(IL + 12)  =  1.
C
   20 IF(IO .NE. 1)  GO TO 28
C
   21 IF (BCIB .NE. 1.0) GO TO 23
C                                                  CLOSED APEX
      OMGN(3,3) = 1.0
      OMG1 (1,1) = 1.0
      OMG1 (1,2) = -1.0
      OMG1 (5,4) = -1.0
      OMG1 (5,5) = 1.0
      ALMN(1,1)  = 1.0
      ALMN(2,2)  = 1.0
      ALMN(4,4)  = 1.0
      ALMN(5,5)  = 1.0
      ALM1 (2,1) = 1.0
      ALM1 (2,2) = 1.0
      ALM1 (3,3) = 1.0
      ALM1(4,4) = 1.0
      ALM1 (4,5) = 1.0
      GO TO 28
C
   23 IF (BCIB .NE. 10.) GO TO 25
C                                                  SPECIAL BOUNDARY
      OMGN(1,1) = BNDTB( 1)
      OMGN(1,2) = BNDTB( 2)
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      OMGN(2,2) = BNDTB( 3)
      OMGN(3,3) = BNDTB( 4)
      OMGN(4,4) = BNDTB( 5)
      OMGN (5,4) = BNDTB( 6)
      OMGN(5,5) = BNDTB(7)
      ALMN(1,1) = BNDTB(8)
      ALMN(2,2) = BNDTB( 9)
      ALMN(3,3) = BNDTB(10)
      ALMN(4,4) = BNDTB(11)
      ALMN(5,5) = BNDTB(12)
      DO 24 IJ = 1,5
   24 LLLN(IJ) = BNDTB(IJ+12)
      GO TO 28
C
   25 IBMX  =  BCIB
      IC  =  26
      IL  =  76
      GO TO 11
C
   28 WRITE (6,1000) A , HO, EO, SIGO, POI, ENF, BCIT, BCIB, PFLAG,
     1  THT, EPRNT,
     2  OMGO, ALMO, LLO, OMGN, ALMN, LLLN
C
      N = EN
      GMI = ABS(GIN)
      NN = N - 1
      IF (GMI - 2.0) 30, 35, 50
C
C                 CONE - CYLINDER                          **
C
   30 IF (PFLAG .NE. 1.0) WRITE(6,31)          N, RA1, AXL, ANX
C
   31 FORMAT (1H1,39X,33HGEOMETRY DATA FOR CONE - CYLINDER //35X,22HNU
     1ER OF STATIONS =  ,I4//6X,7HRA1    =,1PE13.4,7X,7HAXL    =,E13.4,7X,
     2 7HANX    =,E13.4)
C
      DEL =  AXL/(EN - 1.0)
      SINFI  = SIND(ANX)
      COSFI  = COSD(ANX)
      XSI(1) = 0.0
      WTMX(1) = A0 * COSFI/RA1
      WEFX(1) = 0.0
      RHOX(1) = RA1/A0
```

```
      R(1) = RA1                                                        00002030
      DO 32 I = 2,N                                                     00002040
      R(I)   = R(I-1) + DEL * SINFI                                     00002050
      XSI(I) = XSI(I-1) + DEL * COSFI                                   00002060
      WTHX(I) = A0 * COSFI/R(I)                                         00002070
      WFEX(I) = 0.0                                                     00002080
   32 RHOX(I) = R(I)/A0                                                 00002090
      GO TO 95                                                          00002100
C                                                                       00002110
C                 SPHERE - TOROID                              **       00002120
C                                                                       00002130
   35 IF (PFLAG .NE. 0.0)  WRITE (6,37)  N, RC, ROFF, PHIO, PHIN        00002140
C                                                                       00002150
   37 FORMAT (1H1,39X,33HGEOMETRY DATA FOR SPHERE - TOROID //35X,22HNUMB 00002160
     1ER OF STATIONS -  ,I4//6X,7HRC     =,1PE13.4,7X,7HROFF   =,E13.4,7X, 00002170
     2 7HPHIO   =,E13.4,7X,7HPHIN   =,E13.4 )                           00002180
C                                                                       00002190
      ANGSP = PHIN - PHIO                                               00002200
      DEL =  ANGSP/(EN - 1.0)                                           00002210
      AM = 1.0                                                          00002220
      AMU = SIGN(AM,DEL)                                                00002230
      XSI(1) =  0.0                                                     00002240
      BPHI = PHIO                                                       00002250
      BSINP = SIND(PHIO)                                                00002260
      BCOSP = COSD(PHIO)                                                00002270
      R(1) = RC * BSINP + ROFF                                          00002280
      DO 40 I = 1,NN                                                    00002290
      APHI = BPHI + DEL                                                 00002300
      ASINP = SIND(APHI)                                                00002310
      ACOSP = COSD(APHI)                                                00002320
      R(I+1) = R(I) + RC * (ASINP - BSINP)                              00002330
      XSI(I+1) = XSI(I) + RC * (BCOSP - ACOSP)                          00002340
      WFEX(I) = A0/RC * AMU                                             00002350
      IF (ROFF .EQ. 0.0) GO TO 38                                       00002360
      WTHX(I) = A0 * BSINP/R(I)                                         00002370
      GO TO 39                                                          00002380
   38 WTHX(I) = WFEX(I)                                                 00002390
   39 RHOX(I) = R(I)/A0                                                 00002400
      BPHI = APHI                                                       00002410
      BSINP = ASINP                                                     00002420
      BCOSP = ACOSP                                                     00002430
   40 CONTINUE                                                          00002440
C                                                                       00002450
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      DEL = ABS(DEL)
      WFEX(N) = A0/RC*ABD
      IF (ROFF .EQ. 0.0) GO TO 45
      WTHX(N) = A0 * BSINP /R(N)
      GO TO 46
   45 WTHX(N) = WFEX(N)
      IF (BCIB .EQ. 1.0) GO TO 47
   46 RHOX(N) = R(N)/A0
      GO TO 48
   47 RHOX(N) = 0.0
   48 DEL = DEL * RC * 0.01745329
      GO TO 95
C
   50 IF (GMT .GT. 3.0) GO TO 60
C
      WRITE (6,55)
   55 FORMAT (1H1,43HARBITRARY FUNCTIONS CANNOT BE HANDLED, YET. )
C
      GO TO 140
C
C                    GENERAL DISCRETE POINTS                    **
C
   60 M  =  EM
      MM = M - 1
      MM2 = M- 2
C
      IF (PFLAG .NE. 0.0) WRITE(6,62)     N, (RIPT(I), XIPT(I), I = 1,M)
C
   62 FORMAT (1H1,39X,31HGEOMETRY DATA - DISCRETE POINTS //35X,22HNUMBER
     1 OF STATIONS -  ,I4//16X,1HR,16X,2HX1//(3X,1P2E20.7) )
C
      SARB(1) = 0.0
      IF(GIN .LT. 0.) GO TO 77
      DO 75 IL = 1,MM
      SURB = 0.0
      DLT = XIPT(IL+1) - XIPT(IL)
      K = 10
      AK = K
      DDL = DLT/AK
      KP1 = K + 1
      DO 65 JI = 1,KP1
      AJI = JI - 1
      XJ(JI) = XIPT(IL) + AJI * DDL
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
   65 CONTINUE
C
      CALL CODIMA (KP1,XJ, RRJ, XIPT, RIPT, M, 1.0)
      DO  69 I = 2,K
   69 RJ(I) =  (RRJ(I-1) + RRJ(I) + RRJ(I+1) )/3.0
      RJ(1) = RRJ(1)
      RJ(KP1) = RRJ(KP1)
C
      DO 70 JR = 1,K
      DLR(JR) = RJ(JR+1) - RJ(JR)
      DLS     =  SQRT(DLR(JR)**2 + DDL**2)
   70 SURB = SURB + DLS
      SARB(IL+1) = SARB(IL) + SURB
   75 CONTINUE
      GO TO 80
   77 DO 78 I = 1,M
   78 SARB(I) = XIPT(I)
   80 DEL = SARB(M) /(EN - 1.0)
C
      SURF(1) = 0.0
      DO 82 I = 1,NN
   82 SURF(I+1) = SURF(I) + DLL
      CALL CODIMA(N, SURF,RCRV, SARB, RCURV, M, 1.0)
      CALL CODIMA(N,SURF, RCRZ, SARB, RCURZ, M, 1.0)
C
      CALL CODIMA (N,SURF, R, SARB, RIPT, M, 1.0)
C
      CALL CODIMA (N, SURF, XSI, SARB, XIPT, M, -1.0)
      MLN = N - 2
      NSM = 1
   84 DO 85 I = 3,MLN
      RR(I) = (-3.*R(I-2) + 12.*R(I-1) + 17.*R(I) + 12.*R(I+1) - 3. *
     1 R(I+2) )/ 35.0
   85 CONTINUE
      RR(NN) = R(NN)
      RR(2) = R(2)
      RR(N) = R(N)
      RR(1) = R(1)
      IF (NSM .EQ. 25) GO TO 88
      NSM = NSM + 1
      DO  87 I = 1,N
   87 R(I) = RR(I)
      GO TO  84
```

```
   88 RHOX(1) = RR(1)/A0                                                00003330
      DELSQ = DEL * DEL                                                 00003340
      DO 89 I = 1,NN                                                    00003350
   89 RHOX(I+1) = RR(I+1) /A0                                           00003360
      GO TO 110                                                         00003370
C                                                                       00003380
C                    COMPUTE GAMA                              **       00003390
C                                                                       00003400
   95  DEL = DEL/ A0                                                    00003410
       DELSQ = DEL * DEL                                                00003420
       DO 105 I = 1,N                                                   00003430
      DENM = 12. * RHOX(I) * DEL                                        00003440
      DENMP= 2. * RHOX(I) * DEL                                         00003450
      IF(RHOX(I) .EQ. 0.)  GO TO 97                                     00003460
      IF (I .NE. 1) GO TO 98                                            00003470
      IF (BCIT .EQ. 1.0) GO TO 103                                      00003475
      GAMA(I) = (3.*(RHOX(I+1) - RHOX(I)) + RHOX(I+1) - RHOX(I+2))/DENMP00003480
      GO TO 105                                                         00003490
   97 GAMA(I)  =  0.0                                                   00003500
      GO  TO  105                                                       00003510
   98 IF (I .EQ. N) GO TO 100                                           00003520
      IF (I .EQ. 2) GO TO 99                                            00003530
      IF (I .EQ. N-1) GO TO 99                                          00003540
      GAMA(I) = (RHOX(I-2) - 8. *(RHOX(I-1) -      RHOX(I+1))- RHOX(I+2)00003550
     1 )/DENM                                                           00003560
      GO TO 105                                                         00003570
   99 GAMA(I) = (RHOX(I+1) - RHOX(I-1)) /DENMP                          00003580
      GO TO 105                                                         00003590
  100 IF (BCIB .EQ. 1.0)  GO TO 103                                     00003595
      GAMA(I) =(3.*(RHOX(I)-RHOX(I-1)) + RHOX(I-2) - RHOX(I-1)) /DENMP  00003600
      GO  TO  105                                                       00003602
  103 GAMA(I)  =  0.                                                    00003608
  105 CONTINUE                                                          00003610
       GO TO 123                                                        00003620
C                                                                       00003630
  110 DO 115 I = 1,N                                                    00003640
      WFEX(I) = A0/RCRV(I)                                              00003650
      WTHX(I) = A0/RCRZ(I)                                              00003660
      IF (RHOX(I) .EQ. 0.0) GO TO 113                                   00003670
      PRO = (RHOX(I) * WTHX(I))**2                                      00003680
      IF (PRO .GT. 1.0) GO TO 114                                       00003690
      GAMA(I) = SQRT(1.-PRO)/RHOX(I)                                    00003700
      GO TO 115                                                         00003710
```


REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
  113 GAMA(I) = 0.0                                                     00003720
      GO TO 115                                                         00003730
  114 GAMA(I) = 0.0                                                     00003740
  115 CONTINUE                                                          00003750
C                                                                       00003759
  123 DO 127 I = 1,N                                                    00003760
      WFDNM  =  2. * DFL                                                00003770
      IF (I .NE. 1) GO TO 124                                           00003780
      WFEPX(I) = (4. * WFEX(I+1) - 3. * WFEX(I) - WFEX(I+2)) /WFDNM     00003790
      GO TO 127                                                         00003800
  124 IF (I .EQ. N) GO TO 126                                           00003810
      WFEPX(I) = (WFEX(I+1) - WFEX(I-1)) /WFDNM                         00003820
      GO TO 127                                                         00003830
  126 WFEPX(I) = (WFEX(I-2) + 3. * WFEX(I) - 4. * WFEX(I-1)) /WFDNM     00003840
  127 CONTINUE                                                          00003850
C                                                                       00003859
      WRITE ( 4) THX, WFEX, GAMA, RHOX, WFEPX                           00003860
C                                                                       00003870
  129 IF (PFLAG .EQ. 0.0) GO TO 140                                     00003880
      WRITE (6,130) DFL, (I, R(I), XSI(I), ATHX(I), WFEX(I), RHOX(I),   00003890
     1 GAMA(I), I = 1,N)                                                00003900
C                                                                       00003910
  130 FORMAT (1H-, 5X, 7HDFL    =,1PE13.4 //                            00003920
     1              3X,1HI,9X,4HR(I),12X,5HXI(I),1 X,9H(THETA),11X,    00003921
     2 5HW(XI),11X,7HRHOX(I),1 X,7HGAMA(I) //(I4, 1P6E17.7) )           00003922
C                                                                       00003940
  140 RETURN                                                            00003950
C                                                                       00003959
 1000 FORMAT(1H1,47X,12HGENERAL DATA/// 3X,7HAO    =,1PE13.4, 8X,7HHO   00004120
     1 =,E13.4, 8X,7HEO    =,E13.4, 8X,7HGIG    =,E13.4// 3X,7HPOI    =, 00004121
     2 E13.4, 8X,7HFNF    =,E13.4, 8X,7HBCIT   =,E13.4, 8X,7HBCIB   =,  00004122
     3 E13.4//3X,7HPFLAG =, E13.4 //                                    00004123
     4       //64X,20HPRINT ANGLES (THT) - // 24X,6E13.4 //6X,33HFOURIER HO00004124
     5ARMONIC PRINTS (FPRNT) - // 24X, 4E13.4                           00004125
     6                                        ///1H-,45X,17HBOUNDAR00004126
     7Y MATRICES /6X,15HTOP BOUNDARY **// 6X,5HOMEGA /5(5E17.7)// 6X,  00004127
     8 6HLAMBDA /5(5E17.7)// 6X,3HL -/ 5E17.7 ///6X,18HBOTTOM BOUNDARY 00004128
     9** // 6X,5HOMEGA /5(5E17.7)// 6X,6HLAMBDA /5(5E17.7)// 6X,3HL -/ 00004129
     X 5E17.7  )                                                        00004130
      END                                                               00009999
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

3.11d DATLYR Subroutine (STIFF)

```
$IBFTC STIFF
C              FOURIER STIFFNESS COEFFICIENTS GENERATOR      * *
C
      SUBROUTINE DATLYR
C
C     *** STIFFNESS COEFF. ARE STORED IN SECTIONS OF ALL FOURIER
C         COMPONENTS /MERIDIONAL STATION, UP TO BLOCKS OF 200
C
      DIMENSION  COEFC(3000)
      DIMENSION  SCB1(200), SCB2(200), SCB3(200), SCG1(200), SCG2(200),
     1 SCG3( 200), SCD1(200), SCD2(200), SCD3(200), SCG13(200),
     2 SCNTST(200), SCNTTH(200), SCMTST(200), SCMTTH(200), DCK3(200)
C
      COMMON  GDA(51), DFL, N, NNF, NTH,
     1 BMTX(160)
C
C
      EQUIVALENCE (GDA( 1),   A0),                (GDA( 2),   H0),
     1            (GDA( 3),   E0),                (GDA( 4), SIG0),
     2            (GDA( 5),  POI),                (GDA( 6),  ENF),
     3            (GDA( 7), BCIT),                (GDA( 8), BCIB),
     4            (GDA( 9),PFLAG),
     5                                            (GDA(12),  THT),
     6            (GDA(18),FPRNT)
C
C
      EQUIVALENCE (COEFC(    1),  SCB1),          (COEFC( 201),  SCB2),
     1            (COEFC( 401),  SCB3),           (COEFC( 601),  SCG1),
     2            (COEFC( 801),  SCG2),           (COEFC(1001),  SCG3),
     3 (COEFC(1201),  SCD1), (COEFC(1401),  SCD2), (COEFC(1601),  SCD3),
     4 (COEFC(1801), SCG13), (COEFC(2001),SCNTST), (COEFC(2201),SCNTTH),
     5 (COEFC(2401),SCMTST), (COEFC(2601),SCMTTH), (COEFC(2801),  DCK3)
C
      IF(PFLAG .LT. 0.)  WRITE(6,10)
   10 FORMAT(1H1/6X, 96HCHECKPRINT OF COEFC BLOCKS STORED IN GROUPS OF
     12*ENF) PER MERIDIONAL STATION.  THE COEFFICIENTS /107H PRINTED ARE
     2 SCB1, SCD1, SCNTST, SCMTST, DCK3 (IN THAT ORDER) AND ARE SEPARATE
     3D FROM EACH OTHER BY THE      / 21H THREE ...... FIELDS.  /// )
C
      ENTH  =  90.
      H02 = H0 * H0
      SS3 = 2.*3.1415927
      NTH  = ENTH
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      NNF  =  FNF                                                       00000440
      NNF2 = 2 * NNF                                                    00000450
      IN = 1                                                            00000460
                                                                        00000470
C                 DETERMINE SIZE OF BLOCK                      **       00000480
C                                                                       00000490
      MNF  = N * NNF2                                                   00000500
   40 IF (MNF .GT. 200) GO TO 50                                        00000510
C                                                LESS THAN ONE BLOCK MORE 00000519
      LN = N                                                            00000520
      GO TO 60                                                          00000530
   50 IF (MNF .LT. 400) GO TO 55                                        00000540
C                                            AT LEAST 2 FULL BLOCKS MORE 00000549
      LN = 200/NNF2                                                     00000550
      GO TO 58                                                          00000560
C                                                                       00000570
C                            GREATER THAN ONE BLOCK, LESS THAN TWO      00000579
C                          AVOID HAVING ONLY TWO STATIONS IN A BLOCK.   00000580
C                                                                       00000590
   55 LN = 100/NNF2                                                     00000600
   58 LN = IN + LN - 1                                                  00000610
C                                                                       00000620
   60 DELTH = 3.14159277 ENTH                                           00000630
C                                                                       00000640
C                                            ZERO COEFFICIENT AREA      00000649
   63 DO 65 I = 1,2000                                                  00000650
   65 COEFF(I) = 0.0                                                    00000660
C                                                                       00000670
   70 DO 100 I = IN, LN                                                 00000680
C                                 ESTABLISH MERIDIONAL DISTANCE         00000689
      FI = I - 1                                                        00000700
      DEL = FI * DEI                                                    00000710
C                                                                       00000719
      K1 = NNF2 * (I - IN)                                              00000720
      DO 100 NN = 1,NNF2                                                00000730
      K2 = K1 + NN                                                      00000740
      FNN = NN - 1                                                      00000750
C                                                                       00000760
   75 DO 95 J = 1,NTH                                                   00000770
      FJ = J - 1                                                        00000800
      ZTA = FJ * DELTH + DELTH * 0.5                                    00000810
      IF (NN .EQ. 1) GO TO 90                                           00000820
C                                                                       00000830
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
C                                                   OTHER FOURIER COMPONENTS
   80 SCB1(K2) = SCB1(K2) + BBB(PEL,ZTA)      * COS (ENN *  ZTA)
      SCNTSI(K2) = SCNTSI(K2) + ENTI(PEL,ZTA) * COS(ENN * ZTA)
      SCMTSI(K2) = SCMTSI(K2) + EMTI(PEL,ZTA) * COS(ENN * ZTA)
      SCD1(K2) = SCD1(K2) + DDD(ZTA)          * COS (ENN *  ZTA)
      DCK3(K2) =DCK3(K2)+  DKK3(PEL,ZTA)  * COS(ENN* ZTA)
      GO TO 95
C                                                   FIRST FOURIER COMPONENT
   90 SCB1(K2) = SCB1(K2) + BBB(PEL,ZTA) / 2.0
      SCNTSI(K2) = SCNTSI(K2) + ENTI(PEL,ZTA) / 2.0
      SCMTSI(K2) = SCMTSI(K2) + EMTI(PEL,ZTA) / 2.0
      SCD1(K2) = SCD1(K2) + DDD(ZTA) / 2.0
      DCK3(K2) =DCK3(K2) + DKK3(PEL,ZTA)/2.0
   95 CONTINUE
C
C                                                   NORMALIZE
      S3 = SS3/(EO * HO)
      SCB1(K2) = S3 * DELTH * SCB1(K2)
      S4 = S3 * AO*AO
      DCK3(K2) = S4 * DELTH * DCK3(K2)
      S3 = SS3 /(SIGO * HO)
      SCNTSI(K2) = S3 * DELTH * SCNTSI(K2)
      S3 = SS3 * AO /(SIGO * HO2 * HO)
      SCMTSI(K2) = S3 * DELTH * SCMTSI(K2)
      S3 = SS3/(EO * HO2 * HO)
      SCD1(K2) = S3 * DELTH * SCD1(K2)
C
   98 SS2 = (1. - POI) / 2.
      SCG1B(K2) = SS2 * SCD1(K2)
      SCG1(K2) = SS2 * SCB1(K2)
      SCB2(K2) = SCB1(K2)
      SCB3(K2) = POI * SCB1(K2)
      SCG2(K2) = SS2 * SCB1(K2)
      SCG3(K2) = SCG1(K2)
      SCD2(K2) = SCD1(K2)
      SCD3(K2) = POI * SCD1(K2)
  100 CONTINUE
C
      WRITE ( 9) LN, IN, COFFC
      NX  =  (LN - IN + 1) * NNF2
      XX  =  GDA(25)
      IF(PFLAG .LT. 0.)
     XWRITE (6,105)  IN, LN, (SCB1(I), I=1,NX), XX, XX, XX, (SCD1(I),
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
     1 I=1,NX), XX, XX, XX, (CNDS1(I), I=1,NX), XX, XX, XX, (CCM1 I(I),
     2 I=1,NX), XX, XX, XX, (DCK3(I), I=1,NX)
  105 FORMAT(/10X,11HFOR STATION, I3,16H THRO CH  TION, I4///
     1       ( 1P6E17.7) )
C
      IF (LN .EQ. N) GO TO 110
      IN = LN + 1
      NCN = N - LN
      MNF = NCN * NMF2
      GO TO 40
  110 RETURN
C
      END
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

3.11e DATLD Subroutine (LOAD)

```
$IBFTC LOAD                                                          00000010
C                FOURIER LOAD COEFFICIENTS GENERATOR BY FUNCTIONS ONLY 00000020
C                                                                    00000030
      SUBROUTINE  DATLD                                              00000040
C                                                                    00000050
C                                                                    00000060
      DIMENSION  PFF(10,50), PTH(10,50), PN(10,50)                   00000340
C                                                                    00000430
      EQUIVALENCE (GDA( 1),   A0),            (GDA( 2),   H0),       00000440
     1            (GDA( 3),   E0),            (GDA( 4), SIG0),       00000441
     2            (GDA( 5),  POI),            (GDA( 6),  ENF),       00000442
     3            (GDA( 7), BCIT),            (GDA( 8), BCIB),       00000443
     4            (GDA( 9),PFLAG),                                   00000444
     5                                        (GDA(12),  THT),       00000445
     6            (GDA(18),FPRNT)                                    00000446
C                                                                    00000449
      COMMON  GDA(25), DFL, N, NNF, NTH,                             00000490
     1 BMTX(160)                                                     00000500
C                                                                    00000510
C                INITIALIZE *          *          *          *       00000520
C                                                                    00000530
      NAB = 0                                                        00000540
      SS3 = 2. /3.1415927                                            00000560
      S3 = SS3 * A0/(SIG0 * H0)                                      00000570
C                                                                    00000580
      DO 20 I = 1,50                                                 00000590
      DO 20 K = 1,10                                                 00000600
      PFF(K,I) = 0.0                                                 00000610
      PTH(K,I) = 0.0                                                 00000630
      PN(K,I)  = 0.0                                                 00000650
   20 CONTINUE                                                       00000670
C                                                                    00000680
      IF(PFLAG .LT. 0.)  WRITE(6,30)                                 00000730
   30 FORMAT(1H1/6X, 98HCHECKPRINT OF LOAD COEF. STORED IN GROUPS OF ALL00000740
     1 ENF PER MERIDIONAL STATION.  THE COEFFICIENTS ARE  / 81H PFF, PTH00000741
     2, PN (IN THAT ORDER) AND ARE SEPARATED FROM EACH OTHER BY ****** FOR00000742
     3IELDS.  ///)                                                   00000743
      NT = N                                                         00000770
C                                                                    00000790
C                                  SPECIAL  FUNCTIONS                00000800
   50 FT = 0.0                                                       00000810
      NI1 = 1                                                        00000820
      NI2 =  N                                                       00000830
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS

```
      FNTH  =  NTH                                                      00000840
      DELTH = 3.1415927 / FNTH                                          00000850
C                                                                       00000870
C                 EACH BLOCK OF LOADS CANNOT CONTAIN MORE THAN 50 STAS. 00000880
C                                                                       00000890
      IF (NI .GT. 50) NI = 50                                           00000900
   70 DO 100 I = 1,NI                                                   00000910
      FI = FI + 1.0                                                     00000920
      PEL = (FI - 1.0) * DEL                                            00000930
C                                                                       00000940
      DO 100 NN = 1,NNF                                                 00000940
      FNN = NN- 1                                                       00000950
C                                                                       00000970
      DO 95 J = 1,NTH                                                   00000980
      FJ = J - 1                                                        00000990
      ZTA = FJ * DELTH + DELTH * 0.5                                    00001010
      IF (NN .EQ. 1)  GO TO 90                                          00001010
C                                                                       00001020
      PTH(NN,I) = PTH(NN,I) + PPPH(PEL,ZTA) *  SIN(FNN * (ZTA-1.5707963)) 00001030
      PEF(NN,I) = PEF(NN,I)  + PPPF(PEL,ZTA) * COS(FNN * ZTA)           00001040
      PN(NN,I) = PN(NN,I) + PPPN(PEL,ZTA) * COS(FNN * ZTA)              00001060
      GO TO 95                                                          00001070
C                                                                       00001080
   90 PTH(NN,I) = PTH(NN,I) + PPPH(PEL,ZTA) /2.                         00001080
      PEF(NN,I) = PEF(NN,I) + PPPF(PEL,ZTA) /2.                         00001090
      PN(NN,I) = PN(NN,I) + PPPN(PEL,ZTA) / 2.                          00001100
   95 CONTINUE                                                          00001120
C                                                                       00001130
      PTH(NN,I) = S3 * DELTH * PTH(NN,I)                                00001140
      PEF(NN,I) = S3 * DELTH * PEF(NN,I)                                00001150
      PN(NN,I) = S3 * DELTH * PN(NN,I)                                  00001160
  100 CONTINUE                                                          00001170
C                                                                       00001180
      WRITE ( 3)  PEF, PTH, PN                                          00001190
  107 XX  =  GOA(25)                                                    00001220
      IF(N .GT. 50 .AND. NAF.EQ. 0)  NI2 = 50                           00001222
      IF(PFLAG .LT. 0.)                                                 00001229
     1WRITE (6,108) NI1, NI2, ((PEF(NN,I), NN=1,NNF), I=1,NI), XX, XX,  00001230
     1X, ((PTH(NN, I), NN=1,NNF), I=1,NI), XX, XX, XX, ((PN(NN,I), NN=1, 00001231
     2NNF), I=1,NI)                                                     00001232
  108 FORMAT(/10X,11HFOR STATION, I3,16H THROUGH STATION, I4///         00001235
     1       ( 1P6E17.7) )                                              00001236
C                                                                       00001238
```

```
      NAB = NAB + NI                                                    00001239
      IF (NAB .EQ. NI) GO TO 600                                        00001240
C                                                                       00001250
C                 MORE THAN 50 STATIONS TO STORE                  *     00001260
C                                                                       00001270
      NI = N - NI                                                       00001280
      NI1  =  51                                                        00001281
      NI2  =  N                                                         00001282
      DO 115 I = 1,50                                                   00001290
      DO 110 J = 1,10                                                   00001300
      PTH(J,I) = 0.0                                                    00001310
      PFE(J,I) = 0.0                                                    00001320
  110 PN(J,I) = 0.0                                                     00001330
  115 CONTINUE                                                          00001340
C                                                                       00001350
      GO TO 70                                                          00001360
C                                                                       00003000
  600 END FILE 3                                                        00003010
      REWIND 3                                                          00003020
C                                                                       00003030
      RETURN                                                            00003040
  620 FORMAT (1H-// (10X, 1P5E17.7))                                    00003050
      END                                                               00003060
```

```
$IBFTC PEANDX                                                          00000010
C          GENERATES THE 'P' AND 'X' MATRICES                   **     00000020
C              NOTE    RECURSION TERMS ARE ALSO GENERATED       **     00000030
C                                                                      00000040
      SUBROUTINE  PANDX                                                00000050
C                                                                      00000060
C     A0          REFERENCE LENGTH (IN)                              00000070
C     H0          REFERENCE THICKNESS (IN)                           00000080
C     E0          REFERENCE YOUNGS MODULUS  (PSI)                    00000090
C     SIG0        REFERENCE STRESS  (PSI)                            00000100
C     NNF         NO. OF TERMS IN THE FOURIER EXPANSION.             00000110
C     LAM         LAMBDA  =  H0 /A0                                  00000120
C     WFE         OMEGA (XI)                                         00000130
C     WTH         OMEGA (THETA)                                      00000140
C      ARRAYS USED                                                   00000150
C          COEFC   -   CONTAINS STIFF. COEFF. FROM DATLYR            00000160
C          P1SCB1  -   TEMPORARY STORAGE                             00000170
C          P2SCB1  -   FOR STIFFNESS COEFF. (B1)                     00000180
C      PANDX FLAGS                                                   00000190
C          IRTE .NE. 1   FIRST STATION  -  NOT FIRST BLOCK           00000200
C          IRTE .EQ. 1   LAST STATION - PREVIOUS BLOCK               00000210
C          IFGFG = 10   F, G, H, PE  HAVE NOT BEEN COMPLETED, YET.   00000220
C          IFGFG = 0    P AND X  HAVE BEEN COMPUTED FOR IRTE = 1.    00000230
C          IFGFG = 1    P AND X  COMPUTED.  INCREMENT STA. INDEX.    00000240
C          IRSFG = 10   R, S, LT,  HAVE NOT BEEN COMPLETED.          00000250
C          IRSFG = 1   INITIAL BOUNDARY COMPUTED.                    00000260
C          IRSFG = 0   TERMINAL BOUNDARY COMPUTED.                   00000270
C      COUNTERS  -                                                   00000280
C          MM   - ABSOLUTE STATION NUMBER.        *          *       00000290
C          I    - STATION NO. WITHIN EACH BLOCK OF STIFF. COEFF.     00000300
C          MMN  - STATION NO. WITHIN BLOCK OF LOADS.                 00000310
C          IFLG - STIFF. COEFF. BLOCK. *          *                  00000320
C                                                                    00000330
      EQUIVALENCE  (BMTX(  1), OMG0),            (BMTX( 26), OMGN),  00000340
     1             (BMTX( 51), ALM0),            (BMTX( 76), ALMN),  00000350
     2             (BMTX(101), OMG1),            (BMTX(126), ALM1),  00000360
     3             (BMTX(151),  LL0),            (BMTX(156), LLLN)   00000370
C                                                                    00000379
      EQUIVALENCE (GDA( 1),   A0),            (GDA( 2),   H0),       00000380
     1            (GDA( 3),   E0),            (GDA( 4), SIG0),       00000390
     2            (GDA( 5),  POI),            (GDA( 6),  ENF),       00000400
     3            (GDA( 7), BCIT),            (GDA( 8), BCIB),       00000410
     4            (GDA( 9),PFLAG),                                   00000411
```

```
     5                                        (GDA(12),  THT),          00000412
     6            (GDA(18),'PRNT)                                       00000413
C                                                                       00000419
      EQUIVALENCE (COEFC(   1),  SCB1),           (COEFC( 201),  SCB2), 00000420
     1            (COEFC( 401),  SCB3),           (COEFC( 601),  SCG1), 00000421
     2            (COEFC( 801),  SCG2),           (COEFC(1001),  SCG3), 00000422
     3 (COEFC(1201),  SCD1), (COEFC(1401),  SCD2), (COEFC(1601),  SCD3),00000423
     4 (COEFC(1801), SCG13), (COEFC(2001),SCNTSI), (COEFC(2201),SCNTTH),00000424
     5 (COEFC(2401),SCMTSI), (COEFC(2601),SCMTTH), (COEFC(2801),  DCK3) 00000425
C                                                                       00000500
      DIMENSION  WTHX(100), WFEX(100), RHOX(100), GAMA(100), WFEPX(100) 00000510
      DIMENSION  ALMO(5,5), ALMN(5,5), OMGO(5,5), OMGN(5,5)             00000520
      DIMENSION  ALM1(5,5), OMG1(5,5)                                   00000530
      REAL  LLO(5),  LLLN(5),                       MFE(10), MTP(10)    00000540
      DIMENSION  P1SCB1(20), P2SCB1(20), P1SCD1(20), P2SCD1(20),        00000550
     1  TFE(10), P1SG13(20), P2SG13(20), P1ENT(20), P2ENT(20),          00000560
     2  TTP(10), P1SCG1(20), P2SCG1(20), P1EMT(20), P2EMT(20),          00000570
     3  P2DCK3(20), PFE50(10), PTH50(10), PN50(10)                      00000571
      DIMENSION   PFF(10, 50),   PTH(10, 50),   PN(10, 50)              00000580
      DIMENSION  P(50,50), X(50)                                        00000600
C                                                                       00000610
      DIMENSION  COEFC(3000)                                            00000620
      DIMENSION  SCB1(200), SCB2(200), SCB3(200), SCG1(200), SCG2(200), 00000630
     1 SCG3( 200), SCD1(200), SCD2(200), SCD3(200), SCG13(200),         00000631
     2 SCNTSI(200), SCNTTH(200), SCMTSI(200), SCMTTH(200), DCK3(200)    00000632
C                                                                       00000670
      COMMON  GDA(25), DEL, N, NNF, NTH,                                00000680
     1 BMTX(160)                                                        00000690
C                                                                       00000699
      REAL   LAM, LAMO2, LAM2O2, LAM2O4, LAM2O8, LAM2, KORO, KORO2,     00000700
     1          MFFTK, MTHTK, MFETP,     LLI                           00000710
C                                                                       00000719
      COMMON /PXCMN/      MM, LLN, I, IFLG, LN, IRSFG, IFGFG,           00000720
     1 BB1KJ, BB2KJ, BB3KJ, GG1KJ, GG2KJ, GG3KJ, DD1KJ, DD2KJ, DD3KJ,   00000721
     2 GG13KJ, TFETK, TTHTK, MFETK, MTHTK, BB1P, BB3P, GG1P, GG2P,      00000722
     3 DD1P, DD3P, GG13P, TFETP, MFETP, OMG(5,5), ALM(5,5), LLI(5),     00000723
     4 WTH, WFE, GAM, RHO, WFEP, LAM, LAM2, LAMO2, LAM2O2, LAM2O4,      00000724
     5 LAM2O8, GAM2, WTMWF, WTMWF2, WFWT, PTHK, PNK, PFEK, P, X,        00000725
     6 DCK1KJ, DCK2KJ, DCK3KJ                                           00000726
C                                                                       00000789
      COMMON /BLOCK1/  EMM2(50,50)                                      00000790
      COMMON /BLOCK2/  EMM3(50)                                         00000800
C                                                                       00000810
```

```
C
      IF(PFLAG .EQ. -1.E+10)  WRITE (6,340)
C                 READ GEOMETRY PARAMETERS FROM TAPE 4, FIRST FILE. (100)
C                 THIS TAPE IS WRITTEN IN GEOM AND BACKSPACED IN DATLNK
      READ  ( 4) WTHX, WFEX, GAMA, RHOX, WFEPX
C
C                 INITIALIZE *          *          *          *
      N2 = 2 * NNF
      MMN = 0
      IFLG = 0
C                 READ PRESSURES (10,50) FROM TAPE 3, FIRST FILE.
C                 TAPE 3 WRITTEN AND REWOUND IN DATLD SUBROUTINE
      READ ( 3)  PFE, PTH, PN
C
C                 READ STIFFNESS COEFF (200) FROM TAPE 9. THIS TAPE IS
C                 WRITTEN IN DATLYR AND REWOUND IN DATLNK AND MAIN
C                 IN  -  INITIAL STATION IN BLOCK
C                 LN  -  LAST STATION IN BLOCK
   10 READ  ( 9) LN, IN, COE.C
C
      IRTE = 1
      MM = IN - 1
      IFLG = IFLG + 1
      LLN = LN - IN + 1
      I = 0
      KK = NNF
      DEL2 = 2. * DEL
C
   20 I = I + 1
      MM = MM + 1
      MMN = MMN + 1
      IF (MM .NE. 51)  GO TO 23
      DO  21      IX = 1,KK
      PFE50(IX)  =  PFE(IX,50)
      PTH50(IX)  =  PTH(IX,50)
   21 PN50(IX)   =  PN(IX,50)
      READ ( 3)  PFE, PTH, PN
      MMN = 1
      REWIND 3
C
   23 DO 9998 J = 1,KK
C
C                 ESTABLISH BOUNDARY CONDITIONS ARRAY.        *
```

```
C
      IF (MM .NE. 1) GO TO 33
      IF (J .EQ. 1) GO TO 31
      IF (BCIT .NE. 1.0) GO TO 31
      IF (J .GT. 2) GO TO 29
C                                          CLOSED APEX, 2ND HARMONIC
   27 DO 28 KN = 1,5
      LLI(KN) = 0.0
      DO 28 JJ = 1,5
      OMG(KN,JJ) = OMG1(KN,JJ)
   28 ALM(KN,JJ) = ALM1(KN,JJ)
      GO TO 37
C                                       CLOSED APEX, .GT. 2ND HARMONIC
   29 DO 30 KN = 1,5
      DO 30 JJ = 1,5
      OMG(KN,JJ) = 0.0
      ALM(KN,JJ) = 0.0
      IF (KN .EQ. JJ) ALM(KN,JJ) = 1.0
   30 LLI(KN) = 0.0
      GO TO 37
C                                     1ST FOURIER COMPONENT, 1ST STA.
   31 DO 32 KN = 1,5
      LLI(KN) = LLO(KN)
      DO 32 JJ = 1,5
      OMG(KN,JJ) = OMGO(KN,JJ)
   32 ALM(KN,JJ) = ALMO(KN,JJ)
      GO TO 37
C
   33 IF (MM .NE. N) GO TO 37
      IF (J .EQ. 1) GO TO 34
      IF (BCIB .NE. 1.0) GO TO 34
      IF (J .GT. 2) GO TO 29
      GO TO 27
   34 DO 35 KN = 1,5
      LLI(KN) = LLLN(KN)
      DO 35 JJ = 1,5
      OMG(KN,JJ) = OMGN(KN,JJ)
   35 ALM(KN,JJ) = ALMN(KN,JJ)
C
   37 DO 9998  K = 1,KK
      IF (I .NE. 1) GO TO 85
      IF (IFLG .NE. 1) GO TO 98
C
```

```
C                 RECURSION FORMS        *          *          *
C
   38 JMK = J - K
      KMJA = 1 + IABS(K - J)
      KPJ = K + J - 1
      TERM1 = 1.0
      IF (JMK .EQ. 0) TERM1 = 0.0
      TERM2 = 1.0
      IF (K .EQ. 1) TERM2 = 0.0
      FCTR = 1. - TERM1 + TERM2
      FCTR2 = 1. - 2. * TERM2
      IF(K+J .GT. 10) FCTR2 = 0.
C
      K1 = N2 * I + KPJ
      K2 = N2 * (I-1) + KPJ
      K3 = N2 * (I+1) + KPJ
      K4 = N2 * I + KMJA
      K5 = N2 * (I-1) + KMJA
      K6 = N2 * (I+1) + KMJA
      K7 = N2 * (I-2) + KPJ
      K8 = N2 * (I-2) + KMJA
      K9 = N2 * (I-3) + KPJ
      K10 = N2 * (I-3) + KMJA
C
      IF (MM .NE. 1) GO TO 55
C
C                 FORWARD DIFFERENCE FOR FIRST STATION.        *
      IF(J .NE. 1) GO TO 39
      TTP(K) = (4. * SCNTSI(K1) - 3. * SCNTSI(K2) - SCNTSI(K3)) /DEL2
      MTP(K) = (4. * SCMTSI(K1) - 3. * SCMTSI(K2) - SCMTSI(K3)) /DEL2
      TFE(K)  =  SCNTSI(K2)
      MFE(K)  =  SCMTSI(K2)
C
   39 BP1KPJ = (4. * SCB1(K1) - 3. * SCB1(K2) - SCB1(K3)) /DEL2
      DP1KPJ = (4. * SCD1(K1) - 3. * SCD1(K2) - SCD1(K3)) /DEL2
      G13KPJ = (4. * SCG13(K1) - 3. * SCG13(K2) - SCG13(K3)) /DEL2
      GP1KPJ = (4. * SCG1(K1) - 3. * SCG1(K2) - SCG1(K3)) /DEL2
C
      BP1KMJ = (4. * SCB1(K4) - 3. * SCB1(K5) - SCB1(K6)) /DEL2
      DP1KMJ = (4. * SCD1(K4) - 3. * SCD1(K5) - SCD1(K6)) /DEL2
      G13KMJ = (4. * SCG13(K4) - 3. * SCG13(K5) - SCG13(K6)) /DEL2
      GP1KMJ = (4. * SCG1(K4) - 3. * SCG1(K5) - SCG1(K6)) /DEL2
C
```

```
C                 DERIVATIVE OF STIFFNESS RECURSIONS.          *
C
   40 BB1P       =  0.5 *  (BP1KPJ + FCTR * BP1KMJ)
      DD1P    =     0.5 * (DP1KPJ + FCTR * DP1KMJ)
      GG13P = 0.5 * (FCTR * G13KMJ + FCTR2 * G13KPJ)
      GG1P = 0.5 * (FCTR * GP1KMJ + FCTR2 * GP1KPJ)
C
C                 STIFFNESS RECURSION   *          *          *
C
   50 BB1KJ  = 0.5 * (SCB1(K2) + FCTR * SCB1(K5))
      DD1KJ  = 0.5 * (SCD1(K2) + FCTR * SCD1(K5))
      GG13KJ = 0.5 * (FCTR * SCG13(K5) + FCTR2 * SCG13(K2))
      GG1KJ = 0.5 * (FCTR * SCG1(K5) + FCTR2 * SCG1(K2))
      DCK3KJ = 0.5 * (DCK3(K2) + FCTR * DCK3(K5))
      GO TO 150
C
   55 IF (MM .EQ. N) GO TO 65
      IF(J .NE. 1) GO TO 57
      TTP(K) = (SCNTSI(K1) - SCNTSI(K7)) /DEL2
      MTP(K) = (SCMTSI(K1) - SCMTSI(K7)) /DEL2
      TFE(K)  =  SCNTSI(K2)
      MFE(K)  =  SCMTSI(K2)
   57 BP1KPJ = (SCB1(K1) - SCB1(K7)) /DEL2
      DP1KPJ = (SCD1(K1) - SCD1(K7)) /DEL2
      G13KPJ = (SCG13(K1) - SCG13(K7)) /DEL2
      GP1KPJ = (SCG1(K1) - SCG1(K7)) /DEL2
C
C
   60 BP1KMJ = (SCB1(K4) - SCB1(K8)) /DEL2
      DP1KMJ = (SCD1(K4) - SCD1(K8)) /DEL2
      G13KMJ = (SCG13(K4) - SCG13(K8)) /DEL2
      GP1KMJ = (SCG1(K4) - SCG1(K8)) /DEL2
      GO TO 40
C
C                 BACKWARD DIFFERENCE FOR LAST STATION.        *
   65 IF(J .NE. 1) GO TO 70
      TTP(K) = (SCNTSI(K9) + 3. * SCNTSI(K2) - 4. * SCNTSI(K7)) /DEL2
      MTP(K) = (SCMTSI(K9) + 3. * SCMTSI(K2) - 4. * SCMTSI(K7)) /DEL2
      TFE(K)  =  SCNTSI(K2)
      MFE(K)  =  SCMTSI(K2)
C
   70 BP1KPJ = (SCB1(K9) + 3. * SCB1(K2) - 4. * SCB1(K7)) /DEL2
      DP1KPJ = (SCD1(K9) + 3. * SCD1(K2) - 4. * SCD1(K7)) /DEL2
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR.

```
      G13KPJ = (SCG13(K9) + 3. * SCG13(K2) - 4. * SCG13(K7)) /DEL2
      GP1KPJ = (SCG1(K9) + 3. * SCG1(K2) - 4. * SCG1(K7)) /DEL2
C
      BP1KMJ = (SCB1(K10) + 3. * SCB1(K5) - 4. * SCB1(K8)) /DEL2
      DP1KMJ = (SCD1(K10) + 3. * SCD1(K5) - 4. * SCD1(K8)) /DEL2
      G13KMJ = (SCG13(K10) + 3. * SCG13(K5) - 4.*SCG13(K8)) /DEL2
      GP1KMJ = (SCG1(K10) + 3. * SCG1(K5) - 4. * SCG1(K8)) /DEL2
      GO TO 40
C
C
   85 IF (LN .EQ. N) GO TO 38
      IF (I .NE. 3) GO TO 93
C                                                AT ARBITRARY STATION 3
C                SAVE COEFF. OF LAST TWO STAS. OF THIS BLOCK.  TO BE
C                USED TO COMPUTE DERIVATIVES ON NEXT BLOCK.  *
C
      KK2 = 2 * KK
      K11 = N2 * (LLN-2)
      K12 = N2 * (LLN-1)
      DO 90 IK = 1, KK2
      K13 = K11 + IK
      K14 = K12 + IK
      P1ENT(IK) = SCNTSI(K13)
      P2ENT(IK) = SCNTSI(K14)
      P1SCB1(IK) = SCB1(K13)
      P2SCB1(IK) = SCB1(K14)
      P1SCD1(IK) = SCD1(K13)
      P2SCD1(IK) = SCD1(K14)
      P1SG13(IK) = SCG13(K13)
      P2SG13(IK) = SCG13(K14)
      P1SCG1(IK) = SCG1(K13)
      P2DCK3(IK) = DCK3(K14)
      P2SCG1(IK) = SCG1(K14)
      P1EMT(IK) = SCMTSI(K13)
      P2EMT(IK) = SCMTSI(K14)
   90 CONTINUE
C                                          IS COEFF BLOCK COMPLETED
   93 IF (I .EQ. LLN) GO TO 10
C                                                     NO
      GO TO 38
C
C
   98 JMK  = J - K
```

```
      KMJA = 1 + IABS(K - J)                                            00002570
      KPJ = K + J - 1                                                   00002580
      TERM1 = 1.0                                                       00002590
      IF (JMK .EQ. 0) TERM1 = 0.0                                       00002600
      TERM2 = 1.0                                                       00002610
      IF (K .EQ. 1) TERM2 = 0.0                                         00002620
      FCTR = 1. - TERM1 + TERM2                                         00002630
      FCTR2 = 1. - 2. * TERM2                                           00002640
      IF(K+J .GT. 10) FCTR2 = 0.                                        00002645
C                                                                       00002650
      K1 = N2 * I + KPJ                                                 00002660
      K2 = N2 * (I-1) + KPJ                                             00002670
      K4 = N2 * I + KMJA                                                00002680
      K5 = N2 * (I-1) + KMJA                                            00002690
C                                                                       00002700
C         IRTF .NE. 1  - FIRST STATION IN A SECTION (NOT REGION).       00002701
C         IRTE .EQ. 1  - LAST STATION IN PREVIOUS SECTION.              00002702
C                                                                       00002703
  100 IF (IRTE .NE. 1) GO TO 110                                        00002710
C                                                                       00002719
C                 LAST STATION - PREVIOUS BLOCK                         00002720
      IF (K .NE. 1 .OR. J .NE. 1) GO TO 102                             00002730
      MM = MM - 1                                                       00002740
      MMN = MMN - 1                                                     00002750
  102 IF(J .NE. 1) GO TO 103                                            00002752
      TTP(K) = (SCNTSI(K2) - P1ENT(KPJ)) /DEL2                          00002753
      MTP(K) = (SCMTSI(K2) - P1EMT(KPJ)) /DEL2                          00002754
      TFE(K)  =  P2ENT(KPJ)                                             00002755
      MFE(K)  =  P2EMT(KPJ)                                             00002756
  103 BP1KPJ = (SCB1(K2) - P1SCB1(KPJ)) /DEL2                           00002760
      DP1KPJ = (SCD1(K2) - P1SCD1(KPJ)) /DEL2                           00002770
      G13KPJ = (SCG13(K2) - P1SG13(KPJ)) /DEL2                          00002780
      GP1KPJ = (SCG1(K2) - P1SCG1(KPJ)) /DEL2                           00002790
C                                                                       00002800
  105 BP1KMJ = (SCB1(K5) - P2SCB1(KMJA)) /DEL2                          00002810
      DP1KMJ = (SCD1(K5) - P2SCD1(KMJA)) /DEL2                          00002820
      G13KMJ = (SCG13(K5) - P2SG13(KMJA)) /DEL2                         00002830
      GP1KMJ = (SCG1(K5) - P2SCG1(KMJA)) /DEL2                          00002840
      GO TO 120                                                         00002850
C                                                                       00002860
C                 FIRST STATION - NOT FIRST BLOCK                       00002900
  110 IF (K .NE. 1 .OR. J .NE. 1) GO TO 111                             00002910
      MM = MM + 1                                                       00002920
```

```
      MMN = MMN + 1                                                     00002930
  111 IF(J .NE. 1) GO TO 112                                            00002932
      TTP(K) = (SCNTSI(K1) - P2ENT(KPJ)) /DEL2                          00002933
      MTP(K) = (SCMTSI(K1) - P2EMT(KPJ)) /DEL2                          00002934
      TFE(K)  =  SCNTSI(K2)                                             00002935
      MFE(K)  =  SCNTSI(K2)                                             00002936
  112 BP1KPJ = (SCB1(K1) - P2SCB1(KPJ)) /DEL2                           00002940
      DP1KPJ = (SCD1(K1) - P2SCD1(KPJ)) /DEL2                           00002950
      G13KPJ = (SCG13(K1) - P2SG13(KPJ)) /DEL2                          00002960
      GP1KPJ = (SCG1(K1) - P2SCG1(KPJ)) /DEL2                           00002970
C                                                                       00002980
      BP1KMJ = (SCB1(K4) - P2SCB1(KMJA)) /DEL2                          00002990
      DP1KMJ = (SCD1(K4) - P2SCD1(KMJA)) /DEL2                          00003000
      G13KMJ = (SCG13(K4) - P2SG13(KMJA)) /DEL2                         00003010
      GP1KMJ = (SCG1(K4) - P2SCG1(KMJA)) /DEL2                          00003020
C                                                                       00003030
  120 BB1P    =  0.5   * (BP1KPJ +  FCTR *  BP1KMJ)                     00003040
      DD1P      = 0.5 * (DP1KPJ + FCTR * DP1KMJ)                        00003050
      GG13P = 0.5 * (FCTR * G13KMJ + FCTR2 * G13KPJ)                    00003060
      GG1P = 0.5 * (FCTR * GP1KMJ + FCTR2 * GP1KPJ)                     00003070
C                                                                       00003080
  125 IF (IRTF .NE. 1) GO TO 135                                        00003090
      TERM11 = P2SCB1(KPJ)                                              00003100
      TERM21 = P2SCD1(KPJ)                                              00003120
  130 BB1KJ    = 0.5 * (TERM11 + FCTR * P2SCB1(KMJA))                   00003130
      DD1KJ    = 0.5 * (TERM21 + FCTR * P2SCD1(KMJA))                   00003140
      GG13KJ = 0.5 * (FCTR * P2SG13(KMJA) + FCTR2 * P2SG13(KPJ))        00003150
      GG1KJ = 0.5 * (FCTR * P2SCG1(KMJA) + FCTR2 * P2SCG1(KPJ))         00003160
      DCK3KJ = 0.5 * (P2DCK3(KPJ) + FCTR * P2DCK3(KMJA))                00003166
      GO TO 150                                                         00003170
C                                                                       00003180
  135 BB1KJ  = 0.5 * (SCB1(K2) + FCTR * SCB1(K5))                       00003190
      DD1KJ  = 0.5 * (SCD1(K2) + FCTR * SCD1(K5))                       00003200
      GG13KJ = 0.5 * (FCTR * SCG13(K5) + FCTR2 * SCG13(K2))             00003210
      GG1KJ = 0.5 * (FCTR * SCG1(K5) + FCTR2 * SCG1(K2))                00003220
      DCK3KJ = 0.5 * (DCK3(K2) + FCTR * DCK3(K5))                       00003226
C                                                                       00003230
  150 BB2KJ   =   BB1KJ                                                 00003250
      BB3KJ  =  POI  *  BB1KJ                                           00003260
      SS2  =  (1. - POI) / 2.                                           00003270
      GG2KJ = SS2 * BB1KJ                                               00003280
      GG3KJ  =  GG1KJ                                                   00003290
      BB3P   =   POI  *  BB1P                                           00003300
```

```
      GG3P  = GG1P                                                      00003310
      GG2P  = SS2 * BB1P                                                00003320
      DD2KJ    =   DD1KJ                                                00003400
      DD3KJ  =  POI   *  DD1KJ                                          00003410
      DD3P  =  POI * DD1P                                               00003420
      DCK1KJ  =  0.                                                     00003422
      DCK2KJ  =  0.                                                     00003424
      TFETK  =  TFE(K)                                                  00003430
      TTHTK  =  TFE(K)                                                  00003440
      MFETK  =  MFE(K)                                                  00003450
      MTHTK  =  MFE(K)                                                  00003460
      TFETP  =  TTP(K)                                                  00003470
      MFETP  =  MTP(K)                                                  00003480
      IF(MMN .EQ. 0) GO TO 153                                          00003482
      PFEK  = PFE(K,MMN)                                                00003490
      PTHK = PTH(K,MMN)                                                 00003500
      PNK = PN(K,MMN)                                                   00003510
      GO  TO  154                                                       00003511
C                                                                       00003512
  153 PFEK  =  PFE50(K)                                                 00003513
      PTHK  =  PTH50(K)                                                 00003514
      PNK   =  PN50(K)                                                  00003515
C                                                                       00003520
C            P AND X FORMULATION                                  **    00003530
C                                                                       00003540
C                                                                       00003550
  154 WTH = WTHX(MM)                                                    00003560
      WFE = WFEX(MM)                                                    00003570
      GAM = GAMA(MM)                                                    00003580
      RHO = RHOX(MM)                                                    00003590
      WFEP = WFEPX(MM)                                                  00003600
      LAM  =  HO /AO                                                    00003610
      LAM2  =  LAM * LAM                                                00003620
      LAMO2  =  LAM /2.                                                 00003630
      LAM202  =  LAM * LAMO2                                            00003640
      LAM204  =  LAMO2 * LAMO2                                          00003650
      LAM208  =  LAM204 /2.                                             00003660
      GAM2  =  GAM **2                                                  00003670
      WTMWF  =  WTH - WFE                                               00003680
      WTMWF2  =  WTMWF **2                                              00003690
      WFWT  =  WFE * WTH                                                00003700
C                                                                       00003710
      IF (IFLG .NE. 1) GO TO 235                                        00003720
```

```
      IF (I .NE. 1) GO TO 235                                           00003730
C                                                                       00003740
C                                          SET UP BOUNDARY MATRICES     00003749
  155 CALL RSLT (J,K,KK)                                                00003750
      IF (IRSFG .GT. 1)  GO TO 9998                                     00003760
      IF (IRSFG .EQ. 1)  GO TO 20                                       00003770
      RETURN                                                            00003780
C                                                                       00003790
  235 IF (I .NE. LLN)  GO TO 237                                        00003800
      IF (LN .EQ. N) GO TO 155                                          00003810
C                                                                       00003818
C                                     SET UP MATRICES IN EQUILIBRIUM EQ. 00003819
  237 CALL FGHPF (J,K,KK,IRTE)                                          00003820
      IF (IFGFG .GT. 1) GO TO 9998                                      00003830
      IF (IFGFG .EQ. 1) GO TO 20                                        00003840
      GO TO 23                                                          00003850
 9998 CONTINUE                                                          00003880
C                                                                       00003890
  340 FORMAT(1H-,14HENTERED PANDX. )                                    00003900
      RETURN                                                            00003910
      END                                                               00003920
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

3.11g RSLT Subroutine (BNDRY)

```
$IBFTC BNDRY
C
      SUBROUTINE RSLT (J,K,KK)
C
C        COMPUTES THE BOUNDARY MATRICES FOR 'P' AND 'X' MATRICES.
C
      DIMENSION    R(50,50), AA(50,50), S(50,50), LT(50)
      DIMENSION  P(50,50), X(50), ZX(50)
C
      COMMON  GDA(25), DEL, N, NNF, NIB,
     1 BMTX(160)
C
      COMMON /PXCMN/      HX, LLN, I, IELG, LM, IRSEC, IEGEG,
     1 BB1KJ, BB2KJ, BB3KJ, GG1KJ, GG2KJ, GG3KJ, DD1KJ, DD2KJ, DD3KJ,
     2 GG13KJ, TFETK, TTHTK, MFETK, MTHTK, DELF, BB3P, GG1P, GG2P,
     3 DD1P, DD3P, GG13P, TFETP, MFETP, OMG(5,5), AL(5,5), LLT(5),
     4 WTH, WEE, GAM, RHO, WFEP, LAM, LAM2, LAM02, LAM202, LAM204,
     5 LAM208, GAM2, WTMWF, WTMWF2, WFWT, PTHK, PNK, PFLK, P, X,
     6 DCK1KJ, DCK2KJ, DCK3KJ
C
      COMMON /BLOCK1/  EMM2(50,50)
      COMMON /BLOCK2/  EMM3(50)
C
      EQUIVALENCE (GDA( 1),   A0),              (GDA( 2),   H0),
     1            (GDA( 3),   L0),              (GDA( 4), SIG0),
     2            (GDA( 5),  POT),              (GDA( 6),  ENF),
     3            (GDA( 7), BCIT),              (GDA( 8), BCIB),
     4            (GDA( 9),PFLAG),
     5                                          (GDA(12),  THT),
     6            (GDA(18),PPRNT)
C
      REAL  LAM, LAM02, LAM202, LAM204, LAM208, LAM2, KJR0, KJR02,
     1          FFTK, TTHTK, FFTP, LT, LLT
C     R, S, LT  MATRICES IN BOUNDARY EQUATIONS  (44, 45) SID 65-1653
C
      KK5  =  5 * NNF
      KJR0  =  0.
      IF (K .NE. 1)  GO TO 164
      IF (J .NE. 1)  GO TO 164
C
C                ZERO ARRAYS FOR ALL MATRICES.     *          *
C
      DO 163 KN = 1,KK5
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      L1(KN) = 0.0                                                      0.00036
      DO 162 JJ = 1,KP1                                                 0.00379
      S(KN,JJ)  = 0.0                                                   0.00388
      R(KN,JJ)  = 0.0                                                   0.00390
  162 CONTINUE                                                          0.00400
  163 CONTINUE                                                          0.00410
C                                                                       0.00419
  164 IF(RHO .EQ. 0.)  GO TO 165                                        0.00420
      KORO = FLOAT(K-1) / RHO                                           0.00425
  165 KORO2  =  KORO * KORO                                             0.00434
      R(5*K-4,5*J-4)= BB1KJ * G4G(1,1)                                  0.00440
      R(5*K-3,5*J-3)= GG1KJ /2. + LAM2DB * RTMWF2 * GG13KJ              0.00462
      R(5*K-3,5*J) =  LAM2D4 * WTMWF  * GG13KJ                          0.00463
      R(5*K-2,5*J-2)= GG2KJ * G4G(3,3)                                  0.00490
      R(5*K-1,5*J-1)=  DD1KJ                                            0.00510
      R(5*K,5*J-3) =  ((GG13KJ * RTMWF /2.) * 0.5) * G4G(5,5)           0.00527
      R(5*K,5*J)=  GG13KJ  *  0.5 * G4G(5,5)                            0.00534
      R(5*K-4,5*J-3)  =  R(5*K-4,5*J-4)  *  G4G(1,2)                    0.00550
      R(5*K-4,5*J)  =  R(5*K-3,5*J)  *  G4G(1,2)                        0.00555
      R(5*K-3,5*J-1)  =  R(5*K-1,5*J-1) * G4G(5,4)                      0.00550
C                                                                       0.00589
      S(5*K-4,5*J-4)=  GA1 *  BB3KJ                                     0.00600
      S(5*K-4,5*J-3)=  KORO  *  BB3KJ                                   0.00606
      S(5*K-4,5*J-2)=  WFE  *  BB1KJ  +  WTH  *  BB2KJ                  0.00610
      S(5*K-3,5*J-4)=  LAM2D3  *  RTMWF2  *  GG13KJ  -  GG1KJ /2.       0.00640
      S(5*K-3,5*J-3) = GA1 * S(5*K-2,5*J-4)                             0.00650
      S(5*K-3,5*J)  =  -GA1 * S(5*K-2,5*J)                              0.00656
      S(5*K-3,5*J-4)  =  KORO  * S(5*K-1,5*J-4)                         0.00671
      S(5*K-3,5*J-1)  =  -KORO * S(5*K-1,5*J)                           0.00685
      S(5*K-2,5*J-4)=  WFE  *  GG2KJ * G4G(3,3)                         0.00696
      S(5*K-2,5*J-1)=  GG2KJ * G4G(3,3)                                 0.00717
      S(5*K-1,5*J-1)=  GA1  *  DD1KJ                                    0.00742
      S(5*K-1,5*J)=  S1  *  DD1KJ                                       0.00757
      S(5*K,5*J-3)  =  GG13KJ * RTMWF  /4.                              0.00775
      S(5*K,5*J-2) = GA1 * S(5*K,5*J-4)                                 0.00771
      S(5*K,5*J-4)  =  KORO * S(5*K,5*J-4)                              0.00780
      S(5*K,5*J-1)=  - KORO  *  GG13KJ  /2.                             0.00791
      S(5*K,5*J)=  - GA1  *  GG13KJ /2.                                 0.00808
      R(5*K-4,5*J-3)  =  R(5*K-4,5*J-4)  * G4G(2,2)                     0.00811
      R(5*K-3,5*J)  =  R(5*K-3,5*J)  * G4G(2,2)                         0.00832
      R(5*K-1,5*J-1)  =  R(5*K-1,5*J-1) * G4G(4,4)                      0.00841
C                                                                       0.00850
      S(5*K-4,5*J-4) = S(5*K-4,5*J-4) * G4G(1,1) + S(5*K-3,5*J-4)  *    0.00860
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR.

```
     1 OMG(1,2)
      S(5*K-4,5*J-3) = S(5*K-4,5*J-3) * OMG(1,1) + S(5*K-3,5*J-3) *
     1OMG(1,2)
      S(5*K-4,5*J-2) = S(5*K-4,5*J-2) * OMG(1,1)
      S(5*K-4,5*J-1) =  S(5*K-3,5*J-1) * OMG(1,2)
      S(5*K-4, 5*J)  =  S(5*K-3, 5*J) * OMG(1,2)
      S(5*K-3,5*J-3)=S(5*K-3,5*J-3) * OMG(2,2)
      S(5*K-3,5*J) = S(5*K-3,5*J) * OMG(2,2)
      S(5*K-3,5*J-4) = S(5*K-3,5*J-4) * OMG(2,2)
      S(5*K-3,5*J-1) = S(5*K-3,5*J-1) * OMG(2,2)
      S(5*K,5*J-4) = S(5*K,5*J-4) * OMG (5,5)
      S(5*K,5*J-3) = S(5*K,5*J-3) * OMG(5,5)
      S(5*K,5*J-1) = S(5*K,5*J-1) * OMG(5,5) + S(5*K-1,5*J-1)  *
     1 OMG(5,4)
      S(5*K-1,5*J-1) = S(5*K-1,5*J-1) * OMG(4,4)
      S(5*K,5*J)  = S(5*K,5*J) * OMG(5,5) + S(5*K-1,5*J) * OMG(5,4)
      S(5*K-1,5*J) = S(5*K-1,5*J) * OMG(4,4)
C
      LT(5*K-4)    =   TFETK   *  OMG(1,1) + LLI(1)
      LT(5*K-3)   =  LLI(2)
      LT(5*K-2) = LLI(3)
      LT(5*K-1)   =    TFETK    *   OMG(4,4) + LLI(4)
      LT(5*K)  = LLI(5)
C
      IF (K .NE. J) GO TO 166
      S(5*K-2,5*J-2) = ALM(3,3)
      S(5*K-4,5*J-4) = S(5*K-4,5*J-4) + ALM(1,1)
      S(5*K-3,5*J-3) = S(5*K-3,5*J-3) + ALM(2,2)
      S(5*K-3,5*J-4) = S(5*K-3,5*J-4) + ALM(2,1)
      S(5*K-1,5*J-1) = S(5*K-1,5*J-1) + ALM(4,4)
      S(5*K-1,5*J) = S(5*K-1,5*J) + ALM(4,5)
      S(5*K,5*J) = S(5*K,5*J) + ALM(5,5)
C
  166 IF (K .NE. KK) GO TO 167
      IF (J .EQ.KK)  GO TO 169
  167 IRSFG = 10
      RETURN
C
  169 IF (IFLG .GT. 1) GO  TO 305
      IF (I .GT. 1) GO TO 305
      A  =  FILE (3,1)
C
C                INITIAL BOUNDARY      *          *          *
```

```
C
      DO 175 K = 1,KK5
      DO 175 J = 1,KK5
  175 R(K,J)  =  (2./DEL) * R(K,J)
C
C                 SAVE  AO   *          *          *
      WRITE ( 3)  R
      DO 176 K = 1,KK5
      DO 176 J = 1,KK5
  176 AAO(K,J) =  -.25 * R(K,J)
C
C                 SAVE  AO   *          *          *          *
      WRITE ( 3) AAO
      DO 177 K = 1,KK5
      DO 177 J = 1,KK5
  177 EM42(K,J) = 3. * AAO(K,J)
      CALL MAD (KK5,KK5, S, EM42, S )
C
C                 SAVE  CO   *          *          *          *
      WRITE ( 3) S
C
C                 SAVE  GO   *          *          *          *
      WRITE ( 3) LT
  190 END FILE  3
      IRSEG = 1
  200 IF(PFLAG .EQ. -1.E+10)  WRITE(6,310)
      RETURN
C
C                 TERMINAL BOUNDARY     *          *          *
C
  305 DO 306 K = 1,KK5
      DO 306 J = 1,KK5
  306 R(K,J) =  (3./(2.*DEL)) * R(K,J)
      CALL MAD (KK5,KK5,  S, R, AA  )
      DO 307 K = 1,KK5
      DO 307 J = 1,KK5
      R(K,J) =  R(K,J) /3.
  307 S(K,J) =  -4. * R(K,J)
C
C                 SAVE  CN, DN, AN, GN (IN THAT ORDER)
C                 TO BE USED FOR  Z(N)
      WRITE (10) R, S, AAO, LT
      BACKSPACE 10

      BACKSPACE 10
C
      IRSEG = 0
      GO  TO  200
  310 FORMAT (1H-,11HLEFT BNDRY, )
      END
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR.

3.11h FGHPE Subroutine (FGMTX)

```
$IBFTC FGMTX
C
      SUBROUTINE FGHPE (J,K,KK,IRTF)
C
      DIMENSION          S(50,5 ), T(5 ,5 ), G(5 ,5 ), H(5 ,50),
     1 PE(50), P(50,50), X(5 )
C
      COMMON  GDA(25), DEL, N, NMF, NTH,
     1 BMTX(160)
C
      COMMON /PXCMN/      MM, LIN, I, IF     LN, IRSEG, IFGEG,
     1 BB1KJ, BB2KJ, BB3KJ, GG1KJ, GG2KJ, GG3KJ, DD1KJ, DD2KJ, DD3KJ,
     2 GG13KJ, TFETK, TTHTK, MFETK, MTHTK, BB1P, BB3P, GG1P, GG2P,
     3 DD1P, DD3P, GG13P, TFETP, MFETP, OMG(5,5), ALB(5,5), LLI(5),
     4 WTH, WFE, GAM, RHO, WFEP, LAM, LAM2, LAM02, LAM202, LAM204,
     5 LAM208, GAM2, WTMWF, WTMWF2, WFAT, PTHK, PNK, PFEK, P, X,
     6 DCK1KJ, DCK2KJ, DCK3KJ
C
      COMMON /BLOCK1/  EMM2(50,50)
      COMMON /BLOCK2/  EMM3(50)

      EQUIVALENCE (GDA( 1),    A0),              (GDA( 2),    H0),
     1            (GDA( 3),    L0),              (GDA( 4), SIG0),
     2            (GDA( 5),  POI),               (GDA( 6),  ENF),
     3            (GDA( 7), BCIT),               (GDA( 8), BCIB),
     4            (GDA( 9),PFLAG),
     5                                           (GDA(12),   THT),
     6            (GDA(18),FPRNT)
C
      REAL   LAM, LAM02, LAM202, LAM204, LAM208, LAM2, KORO, KORO2,
     1         MFETK, MTHTK, MFETP, LT(50), LLI
C
C      F, G, H, PE. MATRICES IN EQUILIBRIUM EQUATIONS (33) SID 65-1632
C
      KK5  =  5 * NNF
      IF (K .NE. 1)  GO TO 164
      IF (J .NE. 1)  GO TO 164
C                ZERO ARRAYS FOR ALL MATRICES.     *          *
      DO 163 KN = 1,KK5
      PE(KN)  = 0.0
      DO 162 JJ = 1,KK5
      F(KN,JJ)  =  0.0
      G(KN,JJ)  =  0.0
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      H(KN,JJ)  =  0.0                                                  00000415
  162 CONTINUE                                                          00000419
  163 CONTINUE                                                          00000420
C                                                                       00000429
  164 KORO = FLOAT(K-1) /RHO                                            00000430
  168 KORO2 = KORO * KORO                                               00000440
C                                                                       00000449
C     F,G,H,PE   MATRICES IN EQUILIBRIUM EQUATIONS (33)  SID 65-1633    00000450
C                                                                       00000451
      F(5*K-4,5*J-4)=  BP1KJ                                            00000460
      F(5*K-3,5*J-3)  =   GG1KJ /2. + LAM208 * WTMWF2 * GG13KJ          00000480
      F(5*K-3, 5*J)   =   LAM204 * WTMWF * GG13KJ                       00000500
      F(5*K-2,5*J-2)=  GG2KJ                                            00000510
      F(5*K-1,5*J-1)=  LAM2  *  DD1KJ                                   00000520
      F(5*K,5*J)=  LAM202  *  GG13KJ                                    00000530
C                                                                       00000539
      G(5*K-4,5*J-4)=  BB1P  +  GAM  *  BB1KJ                           00000554
      G(5*K-4,5*J-3)=  KORO  * ( BB3KJ  +  GG1KJ  /2.  -  LAM208  *     00000555
     1 WTMWF2 *  GG13KJ  )                                              00000556
      G(5*K-4,5*J-2)=  WFF  * ( BP1KJ  +  GG2KJ  ) + GTH  *  BB3KJ      00000570
      G(5*K-4, 5*J)  =  KORO  *( - LAM204 * WTMWF * GG13KJ)             00000590
      G(5*K-3,5*J-4)  = - G(5*K-4,5*J-3)                                00000610
      G(5*K-3,5*J-3)  =  GG1P /2. + LAM208 * ( WTMWF2 * GG13P - GG13KJ  00000620
     1* ( WTMWF2  *  GAM  + 2. *  WFEP  *  WTMWF ) ) +  GG1KJ  * GAM /2. 00000621
      G(5*K-3,5*J-1)  =  -KORO * ( LAM204 * WTMWF * GG13KJ )            00000650
      G(5*K-3,5*J)   =  LAM204 *(GG13P * WTMWF - GG13KJ * (2. * GAM  .  00000670
     1  * WTMWF + WFEP ) )                                              00000671
      G(5*K-2,5*J-4)  = - G(5*K-4,5*J-2)                                00000690
      G(5*K-2,5*J-2)=  GG2P  +  GAM  *  GG2KJ                           00000700
      G(5*K-2,5*J-1)  =  GG2KJ                                          00000710
      G(5*K-1,5*J-3)  = - G(5*K-3,5*J-1)                                00000730
  180 G(5*K-1,5*J-2)  =  - G(5*K-2,5*J-1)                               00000740
      G(5*K-1,5*J-1)= LAM2 * (  DD1P  +  GA   *  DD1KJ  )               00000750
      G(5*K-1,5*J)=  KORO * ( LAM2  *  DD3KJ  +  LAM202  *  GG13KJ  )   00000760
      G(5*K,5*J-4)  = - G(5*K-4,5*J)                                    00000770
      G(5*K, 5*J-3)  =  LAM204 *(GG13P * WTMWF + GG13KJ * (2. * GAM *   00000780
     1   WTMWF - WFEP )  )                                              00000781
      G(5*K,5*J-1)  = - G(5*K-1,5*J)                                    00000800
  183 G(5*K,5*J)= LAM202 * ( GG13P  +  GAM  *  GG13KJ  )                00000810
C                                                                       00000818
      H(5*K-4,5*J-4)= GAM * BB3P  - WFWT  *  BB3KJ  - GAM2  *  BB2KJ -  00000820
     1  KORO2 *(GG1KJ /2. + LAM208 * WTMWF2 * GG13KJ) - WFF **2 * GG2KJ 00000821
     2  -  DCK1KJ                                                      00000822
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      H(5*K-4,5*J-3)=  KORO * ( BB3P  -  GAM  *  BB2KJ  -  GAM /2. *
     1  GG1KJ - GAM * WTMWF2 * LAM2O8 * GG13KJ )
      H(5*K-4,5*J-2)=  WFE *  BB1P  +  WTH * ( BB3P  -  GAM *  BB2KJ  )
     1 +  BB1KJ  * (  WFEP  +  GAM  *  WFF  )
  184 H(5*K-4,5*J-1) =  WFE * GG2KJ + KORO2 *(LAM2O4 * WTMWF * GG13KJ)
      H(5*K-4, 5*J)  =  KORO * GAM * LAM2O4 * WTMWF * GG13KJ
      H(5*K-3,5*J-4)= KORO * ( LAM2O8 * (  WTMWF2 * GG13P  - WTMWF  *
     1  GG13KJ  * ( 3. * GAM * WTMWF -  2. *  WFEP ) ) - GG1P /2. + GAM
     2  * ( -GG1KJ /2. - BB2KJ ) )
      H(5*K-3,5*J-3)  =  (WFWT - GAM2) /2. * GG1KJ - GAM /2. * GG1P
     1  - KORO2 * BB2KJ - WTH **2 * GG3KJ + LAM2O8 * ( GAM * WTMWF2 *
     2   GG13P - GG13KJ * ( WTMWF2 *(WFWT + 3.*GAM2) + 2. * WTMWF
     3   * GAM * WFEP ) ) - DCK2KJ
  190 H(5*K-3,5*J-2)=- KORO * ( WFE *  BB3KJ + WTH * (  BB2KJ  + GG3KJ))
      H(5*K-3,5*J-1)= KORO * ( LAM2O4 * ( GAM * ( 2. * WTMWF  +  WFEP )*
     1  GG13KJ  - WTMWF * GG13P ))
      H(5*K-3, 5*J) =  WTH * GG3KJ + LAM2O4 * WTMWF *
     2 ( GG13KJ  * ( WFWT + 2. * GAM2 ) - GAM *  GG13P  ) + GAM * LAM2O4
     3  *  WFEP  *  GG13KJ
  193 H(5*K-2,5*J-4)=-  GG2KJ * ( WFEP  + GAM * WFF ) - WFF * ( GAM *
     1 BB3KJ  +  GG2P ) - WTH *  GAM  * BB2KJ
      H(5*K-2,5*J-3) = H(5*K-3,5*J-2)
      H(5*K-2,5*J-2)=- KORO2 *  GG3KJ - WFE **2  *  BB1KJ -  WTH * ( 2.*
     1 WFE *  BB3KJ + WTH  *  BB2KJ  ) - DCK3KJ
      H(5*K-2,5*J-1)= GG2P +  GAM *    GG2KJ
      H(5*K-2,5*J)= KORO *   GG3KJ
      H(5*K-1,5*J-4) = H(5*K-4,5*J-1)
      H(5*K-1,5*J-3) =  KORO * LAM2O4 * GAM * WTMWF * GG13KJ
  195 H(5*K-1,5*J-1)= LAM2 * ( GAM * DD3P  - WFWT *  DD3KJ  - KORO2 /2.
     1 *  GG13KJ - GAM2 *  DD2KJ  ) -   GG2KJ
      H(5*K-1,5*J)= LAM2 * KORO * (  DD3P - GAM * ( GG13KJ  /2. +
     1  DD2KJ  )  )
      H(5*K,5*J-4)= KORO *   LAM2O4 * ( WTMWF *  GG13P - WFEP *  GG13KJ)
      H(5*K, 5*J-3) =  LAM2O4 * (GAM * WTMWF * GG13P - GG13KJ * (GAM *
     1   WFEP + WFWT * WTMWF ) ) + WTH * GG3KJ
      H(5*K,5*J-2) = H(5*K-2,5*J)
  198 H(5*K,5*J-1)=- KORO * ( LAM2O2 * ( GG13P + GAM *  GG13KJ ) + LAM2
     1 * GAM *  DD2KJ  )
      H(5*K,5*J)= LAM2O2 * ( ( WFWT -  GAM2 ) *  GG13KJ - GAM * GG13P )
     1 - KORO2 * LAM2 * DD2KJ  -  GG3KJ
C
      PE(5*K-4)  =  - PFEK + TFETP + GAM * ( TFETK - TTHTK  )
      PE(5*K-3) =  - PTHK    -   TTHTK  *  KORO
```

```
      PE(5*K-2)  =  - PNK  -  WFC * D(IK  -  WTH * TTHTK
      PE(5*K-1)  =  LAM2  *  (WFETP  +  GAM * (WFETK - ZTHTK) )
      PE(5*K)  =  - KORG * LAM2 * WTHTK
C
      IF (K .NE. KK) GO TO 203
      IF (J .EQ. KK)  GO TO 208
  203 IFGFG = 10
      GO  TO  261
C
C
C                 FORM MATRICES  A, B, C, F(PE)     *              *
C
  208 DO 210 K=1,KK5
      DO 210 J = 1,KK5
  210 F(K,J) =  (2./DEL) * F(K,J)
      CALL MAD (KK5,KK5, F, G, EMM2 )
      CALL MSU (KK5,KK5, F, G,  G )
      DO 212 K = 1,KK5
      PE(K)  =  2. * DEL * PE(K)
      DO 212 J = 1,KK5
      H(K,J) =  2. * DEL * H(K,J)
  212 F(K,J) = - 2. * F(K,J)
      CALL MAD (KK5,KK5, F,  H,  H )
      DO 214 K = 1,KK5
      DO 214 J = 1,KK5
  214 F(K,J) = EMM2(K,J)
C
C        COMPUTE P AND X                                      **
      IF (MM .NE. 2) GO TO 240
C
      CALL INVNNM (F, 50, KK5, IERR )
C
      IF (IERR .LE. 0) GO TO 320
C
C                 WRITE ON TAPE  -  TO BE USED WHEN FORMING  Z(1)
C                 ORDER  -      A, B, C, G
C
      WRITE ( 3) F, H, G, PE
      BACKSPACE  3
      BACKSPACE  3
C
C                 STATION NO. 2  -  TOP BOUNDARY    *              *
C
```

*BACKSPACE 3 WHEN NOT USING ALTIØ

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
C                 READ MATRICES GENERATED IN 'RSLT'.          *
C
C
C                 READ  GO   *          *          *          *
C
      READ ( 3) LT
      BACKSPACE 3
      BACKSPACE 3
C
C                 READ  CO   *          *          *          *
C
      READ ( 3) S
      CALL MMY (KK5,KK5,KK5,  S,  G, EMM2 )
      CALL MMY (KK5,KK5,KK5, EMM2,  F,  G )
      BACKSPACE 3
      BACKSPACE 3
C
C                 READ  AO   *          *          *          *
C
      READ ( 3) S
      CALL MSU (KK5, KK5,  S, G, F )
      CALL MMY (KK5,KK5,KK5, EMM2,  H,  G )
      BACKSPACE 3
      BACKSPACE 3
C
C                 READ  BO   *          *          *          *
C
      READ ( 3) S
      CALL MSU (KK5,KK5, S, G, S )
      CALL MMY (KK5,KK5,  1, EMM2, PF, EMM3 )
      CALL MSU (KK5,  1, LT, EMM3, LT )
      CALL INVNNM (S, 50, KK5,IERR )
      IF (IERR .LE. 0)  GO TO 320
C
      CALL MMY (KK5, KK5, KK5, S, F, P )
      CALL MMY (KK5,KK5,  1, S, LT, X )
      BACKSPACE  3
      BACKSPACE  3
C
  225 WRITE (10) I, P, X
  230 IFGFG  =  1
      GO  TO  261
C
```

*BACKSPACE 3 WHEN NOT USING ALFIØ

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
C
  240 CALL MMY (KK5,KK5,KK5, G, P, [illegible] )
      CALL  SU (KK5,KK5, H, E[illegible]2, H )
      CALL INVM[illegible] (H, 5[illegible], KK5, IERR )
      IF (IERR .LE. [illegible]) GO TO 320
  248 CALL MMY (KK5, KK5, I , G, X, [illegible] )
      CALL M[illegible]U (KK5,  I, PE, EM43, PF )
      CALL MMY (KK5,KK5,KK5, H, F, [illegible] )
      CALL MMY (KK5,KK5,  I, H, PE, X [illegible]
C
      WRITE (10) I, P, X
      IF (I .NE. 1) GO TO 250
      IF (IRTE .NE. 1) GO TO 250
      IRTE = 0
      IEGEG = 0
      IF (MM .EQ. N-2) GO TO 250
      RETURN
  250 IF (MM .NE. N-2) GO TO 230
      IEGEG = 1
C
  260 WRITE ( 4) P, X
      BACKSPACE  4
  261 IF(PFLAG .EQ. -1.[illegible]+1[illegible]) GRI[illegible] (6,262)
  262 FORMAT (1H-,11HLEFT EGMIX, )
      RETURN
  320 WRITE (6,325)  I
  325 FORMAT (1H-,9X,16HSINGULAR MATRIX,,9X,6HI = ,I4)
      CALL EXIT
      STOP
      END
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

3.11i ZMTRX Subroutine (SOLTN)

```
$IBFTC SOLTN
C          SOLUTION MATRICES
C
      SUBROUTINE  ZMTRX
C
C
      DIMENSION  PX(50,50),  XX(50), ZX(50), AAI1(50,50), BBI1(50,50),
     1 CCI1(50,50), GGI1(50), Z(50)
      DIMENSION ZX2(50),              BBO(50,50), GGO(50)
C
      EQUIVALENCE  (GDA( 9), PFLAG)
C
      COMMON /BLK1/  EMM2(50,50)
      COMMON /BLK2/  EMM3(50)
      COMMON  GDA(25), DEL, N, NNF, NIH,
     1 BMTX(160)
C
      A = FILE (3,2)
C
      DO 15 I = 1,50
   15 Z(I) = 0.0
C
C                READ MATRICES P AND X FOR STA. (N-2)
C
      READ ( 4) BBO, GGO
      END FILE 4
C
      KK5  =  5 * NNF
      DO 100 NN = 1,N
      MN = N - NN + 1
      IF (NN .NE. 1) GO TO 20
C
C                READ MATRICES P AND X FOR STA. (N-1).
C
      READ (10) I, PX, XX
C
C                READ MATRICES A, B, C, E(PE) GENERATED FOR STA. (N)
C
      READ (10) AAI1, BBI1, CCI1, GGI1
      CALL MMMY(KK5,KK5,KK5, AAI1, BBO, EMM2 )
      CALL MMSU(KK5, KK5, EMM2, BBI1, BBI1 )
      CALL MMMY(KK5, KK5,  1, AAI1, GGO, EMM3 )
      CALL MMSU(KK5,   1, GGI1, EMM3, GGI1 )
```

```
      CALL MMMY(KK5, KK5,  1, BBI1, XX, EMM3 )
      CALL MMAD(KK5,  1, GGI1, EMM3, GGI1 )
      CALL MMMY(KK5, KK5, KK5, BBI1, PX, EMM2 )
      CALL MMAD(KK5, KK5, CCI1, EMM2, CCI1 )
      CALL INVMS  (CCI1, 50, KK5, IERR )
      IF (IERR .LE. 0) GO TO 120
      CALL MMMY(KK5, KK5,  1, CCI1, GGI1, ZX )
C
      WRITE ( 4) MN, ZX
C
      WRITE (6, 130) MN,  (ZX(IX), IX=1,KK5)
      GO TO 100
C
   20 IF (NN .EQ. N) GO TO 40
      BACKSPACE 10
      BACKSPACE 10
      READ  (10) I, PX, XX
C
      CALL MMMY (KK5, KK5,  1, PX, ZX, Z )
      CALL MMSU (KK5, 1, XX, Z, Z )
C
      IF (NN .NE. N-2) GO TO 30
      DO 25 K = 1,KK5
   25 ZX2(K) = Z(K)
   30 DO 35 K = 1,KK5
   35 ZX(K) = Z(K)
C
      WRITE ( 4) MN, ZX
C
      WRITE (6, 140) MN,  (ZX(IX), IX=1,KK5)
      GO TO 100
C
C
C              READ MATRICES A, B, C, E(PE) GENERATED FOR STA. (2)
C
   40 READ ( 3) AAI1, BBI1, CCI1, GGI1
      CALL MMMY(KK5,KK5,  1, BBI1, ZX, PX )
      CALL MMMY(KK5,KK5,  1, AAI1, ZX2, ZX )
      CALL MMSU(KK5,  1, GGI1, ZX, GGI1 )
      CALL MMSU(KK5,  1, GGI1, PX, GGI1 )
      CALL MMMY(KK5,KK5,  1, CCI1, GGI1, Z )
C
      WRITE (6, 140) MN, (Z(IX), IX=1,KK5)
```


REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
C                                                                       00000819
      WRITE ( 4) MN, Z                                                  00000820
  100 CONTINUE                                                          00000830
C                                                                       00000840
      IF(PFLAG .EQ. -1.E+10)  WRITE(6,109)                              00000842
  109 FORMAT (1H-, 10HLEFT SOLTN )                                      00000843
  110 RETURN                                                            00000850
C                                                                       00000859
  120 WRITE (6,125)                                                     00000860
      CALL EXIT                                                         00000870
      STOP                                                              00000880
  125 FORMAT (1H-,10X,45HERROR -  INVERSION OF SINGULAR MATRIX IN 'Z'. )00000890
C                                                                       00000900
  130 FORMAT (1H1,10X,24H'Z' MATRIX FOR STATION  , I4// (12X,1P5E17.7)) 00000910
  140 FORMAT (/// 10X,24H'Z' MATRIX FOR STATION  , I4// (12X,1P5E17.7)) 00000915
C                                                                       00000920
      END                                                               00000930
```


REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
$IBFTC SUSM                                                             00000010
C              SOLUTION SUMMATION     W/ STRESSES                 **    00000020
C                                                                       00000030
      SUBROUTINE SUMS                                                   00000040
C                                                                       00000041
C   NOMENCLATURE                                                        00000042
C     THT          HORIZONTAL ANGLES (DEGREES)  5 MAXIMUM               00000043
C     FPRNT        FOURIER COMPONENT PRINT VALUES OF DEFLECTIONS AND    00000044
C                  ROTATIONS    (3 POSSIBLE)                            00000045
C                                                                       00000050
      DIMENSION    Z(50), THT( 6), FPRNT( 3)                            00000060
      DIMENSION   ULZ(100,5), VLZ(100,5), WLZ(100,5), PHSLZ(100,5),     00000070
     1            PHTLZ(100,5), UL(100,5), VL(100,5), WL(100,5),        00000080
     2            PHSL(100,5), PHTL(100,5), WFFX(100), WTHX(100),       00000090
     3            GAMA(100), RHOX(100), WFFPX(100)                      00000100
C                                                                       00000110
      DIMENSION   B1(100,5), B2(100,5), B3(100,5), D1(100,5), D2(100,5), 00000150
     1            D3(100,5), G1(100,5), G2(100,5), G3(100,5), G13(100,5),00000151
     2            ENTSI(100,5), ENTTH(100,5), EMTSI(100,5), EMTTH(100,5) 00000152
C                                                                       00000160
      DIMENSION   ENFE( 5), FNTH( 5), ENFT( 5), QFE( 5), EMFE( 5),      00000170
     1            EMTH( 5), EMFT( 5), QTH( 5)                           00000180
C                                                                       00000190
      DIMENSION   SIGSI1(5), SIGTH1(5), TAUST1(5), TAUSG1(5),           00000200
     1 TAUTG1(5), SIGSI2(5), SIGTH2(5), TAUST2(5), TAUSG2(5),           00000210
     2 TAUTG2(5)                                                        00000220
C                                                                       00000230
      DIMENSION   COFFC(3000)                                           00000240
      DIMENSION   SCB1(200), SCB2(200), SCB3(200), SCG1(200), SCG2(200), 00000250
     1 SCG3( 200), SCD1(200), SCD2(200), SCD3(200), SCG13(200),         00000260
     2 SCNTSI(200), SCNTTH(200), SCMTSI(200), SCMTTH(200)               00000270
C                                                                       00000390
      EQUIVALENCE (GDA( 1),   AO),              (GDA( 2),   HO),        00000400
     1            (GDA( 3),   EO),              (GDA( 4), SIGO),        00000410
     2            (GDA( 5),  POI),              (GDA( 6),  ENF),        00000420
     3            (GDA( 7), BCIT),              (GDA( 8), BCIB),        00000430
     4            (GDA( 9),PFLAG),                                      00000431
     5                                          (GDA(12),  THT),        00000432
     6            (GDA(18),FPRNT)                                       00000433
C                                                                       00000440
      EQUIVALENCE (COEFC(   1),  SCB1),         (COEFC( 201),  SCB2),   00000450
     1            (COEFC( 401),  SCB3),         (COEFC( 601),  SCG1),   00000460
     2            (COEFC( 801),  SCG2),         (COEFC(1001),  SCG3),   00000470
```

```
     3 (COEFC(1201),  SCD1), (COEFC(1401),  SCD2), (COEFC(1601),  SCD3),00000480
     4 (COEFC(1801), SCG13), (COEFC(2001),SCNTSI), (COEFC(2201),SCNTTH),00000490
     5 (COEFC(2401),SCMTSI), (COEFC(2601),SCMTTH)                       00000500
C                                                                       00000510
      COMMON  GDA(25), DEL, N, NNF, NTH, BMTX(160)                      00000520
C                                                                       00000530
      REWIND   4                                                        00000540
C                                                                       00000600
      A = FILF (4,1)                                                    00000610
      SS1 = SIGO/FO                                                     00000620
      SS2 = SS1 * AO                                                    00000630
C                                                                       00000639
      DO 200 I = 1,N                                                    00000640
      IF(I .EQ. 1)  WRITE( 6,246)                                       00000645
      II = N - I + 1                                                    00000650
C                                                                       00000660
      READ ( 4) MN, Z                                                   00000670
C                                                                       00000680
      WRITE (6,290) II                                                  00000690
C                                                                       00000700
      DO 35 IT = 1,5                                                    00000710
      B1(I,IT)  =  0.0                                                  00000720
      D1(I,IT)  =  0.0                                                  00000722
      ENTSI(I,IT)  =  0.0                                               00000724
      EMTSI(I,IT)  =  0.0                                               00000726
      ULZ(I,IT) = 0.0                                                   00000730
      VLZ(I,IT) = 0.0                                                   00000740
      WLZ(I,IT) = 0.0                                                   00000750
      PHSLZ(I,IT) = 0.0                                                 00000760
   35 PHTLZ(I,IT) = 0.0                                                 00000770
C                                                                       00000780
      IT = 1                                                            00000790
   50 U = 0.0                                                           00000800
      V = 0.0                                                           00000810
      W = 0.0                                                           00000820
      PHS = 0.0                                                         00000830
      PHT = 0.0                                                         00000840
      IX = 1                                                            00000850
      THETA = THT(IT)                                                   00000860
C                                                                       00000870
      DO 150 NF = 1,NNF                                                 00000880
      ANF = NF - 1                                                      00000890
      NT = (NF - 1) * 5                                                 00000900
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      NU = NT + 1                                                       00000910
      NV = NT + 2                                                       00000920
      NW = NU + 2                                                       00000930
      NFS = NV + 2                                                      00000940
      NFT = NW + 2                                                      00000950
      FCT = COSD (ANF * THETA)                                          00000960
      FCT2 = FCT * SS2                                                  00000970
      FCT1 = FCT * SS1                                                  00000980
      FCT3 = SIND (ANF * THETA)                                         00000990
      FCT4 = SS2 * FCT3                                                 00001000
      FCT5 = SS1 * FCT3                                                 00001010
C                                                                       00001020
      U = U + Z(NU) * FCT2                                              00001030
      V = V + Z(NV) * FCT4                                              00001040
      W = W + Z(NW) * FCT2                                              00001050
      PHS = PHS + Z(NFS) * FCT1                                         00001060
      PHT = PHT + Z(NFT) * FCT5                                         00001070
C                                                                       00001080
C                                   DERIVATIVES WITH RESPECT TO THETA   00001089
      ULZ(I,IT) = ULZ(I,IT) - Z(NU) * FCT4 * ANF                        00001090
      VLZ(I,IT) = VLZ(I,IT) + Z(NV) * FCT2 * ANF                        00001100
      WLZ(I,IT) = WLZ(I,IT) - Z(NW) * FCT4* ANF                         00001110
      PHSLZ(I,IT) = PHSLZ(I,IT ) - Z(NFS) * FCT5 * ANF                  00001120
      PHTLZ(I,IT) = PHTLZ(I,IT) + Z(NFT) * FCT1 * ANF                   00001130
C                                                                       00001140
      BNF = ANF + 1.                                                    00001150
      IF (BNF .NE. FPRNT(IX)) GO TO 150                                 00001160
C                                                                       00001170
      WRITE (6,300) THETA, ANF, U, V, W, PHS, PHT                       00001180
      IX = IX + 1                                                       00001190
  150 CONTINUE                                                          00001200
C                                                                       00001210
C                                           DISPLACEMENTS               00001229
      UL(I,IT) = U                                                      00001230
      VL(I,IT) = V                                                      00001240
      WL(I,IT) = W                                                      00001250
      PHSL(I,IT) = PHS                                                  00001260
      PHTL(I,IT) = PHT                                                  00001270
C                                                                       00001280
      IF (THT(IT+1) .LT. 0.0) GO TO 200                                 00001290
      IT = IT + 1                                                       00001300
      GO TO 50                                                          00001310
  200 CONTINUE                                                          00001320
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
C
      REWIND 4
      IIT = IT
      S1 = FO * HO
      S2 = S1 * HO * HO
      S3 = (1. - POI)/2.
      S4 = SIGO * HO
      S5 = AO /(S4 * HO **2)
      II = 0
      NF2  =  2 * NNF
C
  210 READ ( 9) LN, IN, COEFC
      LLN = LN - IN + 1
      DO 240 I = 1,LLN
      II = II + 1
      K1 = NF2 * (I-1)
C
      DO 230 IT = 1,IIT
      THETA = THT(IT)
      DO 225 NF = 1,NF2
      ANF = NF - 1
      K2 = K1 + NF
      B1(II,IT) = B1(II,IT) + SCB1(K2) * COSD(ANF * THETA) * S1
      D1(II,IT) = D1(II,IT) + SCD1(K2) * COSD(ANF * THETA) * S2
      ENTSI(II,IT) = ENTSI(II,IT) + SCNTSI(K2) * COSD(ANF * THETA) * S4
      EMTSI(II,IT) = EMTSI(II,IT) + SCMTSI(K2) * COSD(ANF * THETA) * S5
  225 CONTINUE
      B11 = B1(II,IT)
      D11 = D1(II,IT)
      B2(II,IT) = B11
      B3(II,IT) = POI * B11
      D2(II,IT) = D11
      D3(II,IT) = POI * D11
      G1(II,IT) = S3 * B11
      G2(II,IT) = S3 * B11
      G3(II,IT) = B11 * S3
      G13(II,IT) = S3 * D11
      ENTTH(II,IT) = ENTSI(II,IT)
      EMTTH(II,IT) = EMTSI(II,IT)
  230 CONTINUE
  240 CONTINUE
C
      IF (LN .EQ. N) GO TO 245
```


REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      GO TO 210                                                         00001750
C                                                                       00001790
  245 READ ( 4) WTHX, WFEX, GAMA, RHOX, WFEPX                           00001800
C                                                                       00001810
      K1  =  1                                                          00001820
      K2  =  2                                                          00001821
      DNM = -2. * DEL                                                   00001840
C                                                                       00001860
      DO 275 I = 1,N                                                    00001870
      IF(I .EQ. 1)  WRITE( 6,246)                                       00001872
  246 FORMAT( 1H1 )                                                     00001874
      L = N - I + 1                                                     00001880
      H = L                                                             00001890
      PEL  =  (H - 1.) * DEL                                            00001895
      WFF = WFEX(L)                                                     00001900
      WTH = WTHX(L)                                                     00001910
      GAM = GAMA(L)                                                     00001920
      RHO = RHOX(L)                                                     00001930
      DRHO =  0.                                                        00001940
      IF(RHO .NE. 0.)  DRHO = 1. /RHO                                   00001941
      DAO  =  1./AO                                                     00001950
C                                                                       00001959
      DO 265 IT = 1,IIT                                                 00001960
      THETA = THT(IT)                                                   00001970
      ZTA  =  THETA * .017453293                                        00001972
C                                   DERIVATIVES WITH RESPECT TO PSI     00001979
      IF (I .NE. 1) G  TO 250                                           00001980
C                                              LAST STATION             00001989
      ULP = (4. * UL( 1,IT) - 3. * UL(I,IT) - UL(I+2,IT) )/ DNM         00001990
      VLP = (4. * VL( 1,IT) - 3. * VL(I,IT) - VL(I+2,IT) )/ DNM         00002000
      WLP = (4. * WL(I+1,IT) - 3. * WL(I,IT) - WL(I+2,IT) )/ DNM        00002010
      PHSLP = (4. * PHSL(I+1,IT) - 3. * PHSL(I,IT) - PHSL(I+2,IT) )/DNM 00002020
      PHTLP = (4. * PHTL(I+1,IT) - 3. * PHTL(I,IT) - PHTL(I+2,IT) )/DNM 00002030
      GO TO 260                                                         00002040
  250 IF (I .EQ. N) GO TO 255                                           00002050
C                                   (NOTE - 'I' IS COUNTING BACKWARDS)  00002059
      ULP = (UL(I+1,IT) - UL(I-1,IT) ) /DNM                             00002060
      VLP = (VL(I+1,IT) - VL(I-1,IT) ) /DNM                             00002070
      WLP = (WL(I+1,IT) - WL(I-1,IT) ) /DNM                             00002080
      PHSLP = (PHSL(I+1,IT) - PHSL(I-1,IT) ) /DNM                       00002090
      PHTLP = (PHTL(I+1,IT) - PHTL(I-1,IT) ) /DNM                       00002100
      GO TO 260                                                         00002110
C                                              FIRST STATION            00002119
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
  255 ULP = (UL(I-2,IT) + 3. * UL(I,IT) - 4. * UL(I-1,IT) ) /DNM
      VLP = (VL(I-2,IT) + 3. * VL(I,IT) - 4. * VL(I-1,IT) ) /DNM
      WLP = (WL(I-2,IT) + 3. * WL(I,IT) - 4. * WL(I-1,IT) ) /DNM
      PHSLP = (PHSL(I-2,IT) + 3. * PHSL(I,IT) - 4. * PHSL(I-1,IT) )/DNM
      PHTLP = (PHTL(I-2,IT) + 3. * PHTL(I,IT) - 4. * PHTL(I-1,IT) )/DNM
C
C                 COMPUTATION OF STRAINS ** (STR 139, PG.17) **
C
  260 ESI = DA0 * (ULP + WFF * WL(I,IT))
      ETH = DA0 * (DRHO * VLZ(I,IT) + GAM * UL(I,IT) + WTH * WL(I,IT))
      ESTH = DA0/2. * (DRHO * ULZ(I,IT) + VLP - GAM * VL(I,IT) )
      GSI = PHSL(I,IT) - DA0 * (WFF * UL(I,IT) - WLP )
      GTH = PHTL(I,IT) - DA0 * (WTH * VL(I,IT) - DRHO * WLZ(I,IT) )
      EKSI = DA0 * PHSLP
      EKTH = DA0 * (DRHO * PHTLZ(I,IT) + GAM * PHSL(I,IT) )
      EKSTH = DA0/2. * (DRHO * PHSLZ(I,IT) + PHTLP - GAM * PHTL(I,IT) +
     1 DA0/2.* (WFE-WTH)*(DRHO* ULZ(I,IT) - VLP-GAM*VL(I,IT)))
      IF(PFLAG .EQ. -10.)  WRITE(6,261) L, THETA, ESI, ETH, ESTH, GSI,
     1                                  GTH, EKSI, EKTH, EKSTH
  261 FORMAT(/// 10X, 34HCHECKPRINT OF STRAINS AT STATION  ,I4,  5X,
     1  7HTHETA  ,1PE12.4// 8F13.4 )
C
C                 COMPUTATION OF INTERNAL FORCES ** (STR 139, PG.12) **
C
      ENFE(IT)  =  B1(L,IT) * ESI + B3(L,IT) * ETH - ENTSI(L,IT)
      ENTH(IT)  =  B3(L,IT) * ESI + B2(L,IT) * ETH - ENTTH(L,IT)
      ENFT(IT)  =  G1(L,IT) * ESTH
      QFE(IT)   = G2(L,IT) * GSI
      EMFE(IT)  =  D1(L,IT) * EKSI + D3(L,IT) * EKTH - EMTSI(L,IT)
      EMTH(IT)  =  D3(L,IT) * EKSI + D2(L,IT) * EKTH - EMTTH(L,IT)
      EMFT(IT)  =  G13(L,IT) * EKSTH
      QTH(IT)    = G3(L,IT) * GTH
C
C                 COMPUTATION OF STRESSES ** (STR 139, PG.11) **
C
C                                                INNER SURFACE
      OPE   =  FI1 (PEL,ZTA)
      IF (OPE .EQ. 0.) GO TO 265
      OPD   =  DN1 (PEL,ZTA)
      OPA   =  ALF1(PEL,ZTA)
      OPTT  =  TMP1(PEL,ZTA)
      POI1  =  POIS1 (PEL, ZTA)
      FTR = OPE /(1.-POI1*POI1)
```


REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      TERM = (OPA + POI1 * OPA) * OPTT                                  00002520
      TER = OPE/(2.*(1.+POI1) )                                         00002530
C                                                                       00002539
  262 SIGSI1(IT) = FTR * (ESI + POI1*ETH + OPD*(EKSI+POI1*EKTH) - TERM) 00002540
      SIGTH1(IT) = FTR * (ETH + POI1*ESI + OPD*(EKTH+POI1*EKSI) - TERM) 00002550
      TAUST1(IT) = TER * (ESTH + OPD * EKSTH)                           00002560
      TAUSG1(IT) = TER * GSI                                            00002570
      TAUTG1(IT) = TER * GTH                                            00002580
C                                                  SELECTED SURFACE     00002589
      OPD   =  DN2 (PEL,ZTA)                                            00002590
      OPE   =  EI2 (PEL,ZTA)                                            00002592
      OPA   =  ALF2(PEL,ZTA)                                            00002594
      OPTT  =  TMP2(PEL,ZTA)                                            00002596
      POI2  =  POIS2 (PEL, ZTA)                                         00002598
      FTR = OPE/(1.-POI2*POI2)                                          00002630
      TERM = (OPA + POI2 * OPA) * OPTT                                  00002640
      TER = OPE/(2.*(1.+POI2) )                                         00002650
C                                                                       00002659
  264 SIGSI2(IT) = FTR * (ESI + POI2*ETH + OPD*(EKSI+POI2*EKTH) - TERM) 00002660
      SIGTH2(IT) = FTR * (ETH + POI2*ESI + OPD*(EKTH+POI2*EKSI) - TERM) 00002670
      TAUST2(IT) = TER * (ESTH + OPD * EKSTH)                           00002680
      TAUSG2(IT) = TER * GSI                                            00002690
      TAUTG2(IT) = TER * GTH                                            00002700
  265 CONTINUE                                                          00002710
C                                                                       00002720
      WRITE (6,268) L, (ENFE(J), ENTH(J), ENFT(J), QFE(J), J = 1,IIT)   00002730
      WRITE (6,270)  (EMFE(J), EMTH(J), EMFT(J), QTH(J), J = 1,IIT)     00002740
      IF (OPE .EQ. 0.) GO TO 275                                        00002745
      WRITE (6,272) K1, (SIGSI1(J), SIGTH1(J), TAUST1(J), TAUSG1(J),    00002750
     1 TAUTG1(J), J = 1,IIT)                                            00002760
      WRITE (6,272) K2, (SIGSI2(J), SIGTH2(J), TAUST2(J), TAUSG2(J),    00002770
     1 TAUTG2(J), J = 1,IIT)                                            00002780
C                                                                       00002790
  268 FORMAT (1H-/11X,9HSTATION  ,I4// 19X, 5HN(XI),11X, 8HN(THETA), 7X,00002800
     1 11HN(XI,THETA),10X, 5HQ(XI) /(12X, 1P4E17.7) )                   00002810
C                                                                       00002820
  270 FORMAT (1H-,18X, 5HM(XI),11X, 8HM(THETA), 7X,11HM(XI,THETA), 8X,  00002830
     1 8HQ(THETA) / (12X, 1P4E17.7) )                                   00002840
C                                                                       00002849
  272 FORMAT (1H-/16X,18HSTRESSES FOR LAYER ,I4// 19X, 7HSIG(XI), 8X,   00002850
     1 10HSIG(THETA),6X,13HSIG(XI,THETA),5X,11HSIG(XI,ETA),5X,          00002860
     2 14HSIG(THETA,ETA) /(12X, 1P5E17.7) )                             00002870
C                                                                       00002880
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
  275 CONTINUE                                                          00002890
C                                                                       00002900
      IF(PFLAG .EQ. -1.E+10)  WRITE(6,280)                              00002902
  280 FORMAT (1H-, 9HLEFT SUMS )                                        00002904
      RETURN                                                            00002910
C                                                                       00002919
  290 FORMAT (////10X, 9HSTATION   ,I4)                                 00002920
  300 FORMAT (/// 10X, 9HTHETA  = ,F7.2//11X, 9HENF     = ,F5.1//21X,1HU,00002930
     1 16X,1HV,16X,1HW,12X, 7HPHI(XI), 9X,10HPHI(THETA) /(12X,1P5E17.7)100002940
C                                                                       00002950
      END                                                               00002960
```

```
$IBFTC EXEC                                                          00000010
C          APOLLO WATER IMPACT ANALYSIS              **          **   00000020
C                                                                    00000030
C  REFERENCES ** AIAA JOURNAL, VOL. 1, NO. 8, AUG. 1963, PG.1833FF AND 00000040
C                VOL. 2, NO. 3, MARCH 1964, PG. 590FF                00000050
C             ** SID 65-1633 REPORT, APOLLO WATER IMPACT ANALYSIS, NOV. 00000060
C                23, 1965, PG. 81FF                                  00000070
C                                                                    00000080
C     NOMENCLATURE                                                   00000090
C           A0        REFERENCE LENGTH (IN)                          00000100
C           H0        REFERENCE THICKNESS (IN)                       00000110
C           E0        REFERENCE YOUNG MODULUS (PSI)                  00000120
C           SIG0      REFERENCE STRESS (PSI)                         00000130
C           POI       POISSON RATIO                                  00000140
C           ENF       NO. OF FOURIER COMPONENTS                      00000150
C           BCIT      TOP BOUNDARY CONDITION FLAG                    00000180
C           BCIB      BOTTOM BOUNDARY CONDITION FLAG.                00000181
C                 = 1.0     CLOSED APEX                              00000182
C                 = 2.0     PINNED                                   00000183
C                 = 3.0     CLAMPED                                  00000184
C                 = 4.0     FREE                                     00000185
C                 = 5.0     ROLLER W                                 00000186
C                 = 6.0     ROLLER U                                 00000187
C                 = 10.0    SPECIAL BOUNDARY READ WITH GEOMETRY DATA 00000195
C           DELT      TIME INCREMENT                                 00000200
C           CEXT      NUMBER OF TIME CYCLES                          00000210
C           PFLAG     DATA PRINT INDICATOR                           00000220
C                 = 0.0     PRINTS GENERAL DATA AND BND. MATRICES    00000221
C                 = 1.0     PRINTS ABOVE INFORMATION AND INPUT DATA  00000222
C                           FOR A PARTICULAR GEOMETRY, AND THE PARA- 00000223
C                           METERS COMPUTED IN  GEOM                 00000224
C                =-1.0      PRINTS ALL OF THE ABOVE INFORMATION AND  00000225
C                           STIFFNESS COEFFICIENTS, ELASTIC COEF.,   00000226
C                           THERMAL LOAD AND MOMENT AND PRESSURES    00000227
C           THT       CIRCUMFERENTIAL ANGLE THETA (DEGREES)  5 MAX   00000230
C           FPRNT     FOURIER COMPONENT PRINT VALUES. 3 PERMITTED    00000231
C                                                                    00000239
C                                                                    00000520
      DIMENSION   BCD(96), THT(6), FPRNT(3)                          00000525
C                                                                    00000529
      EQUIVALENCE (GDA( 1),   A0),                (GDA( 2),   H0),   00000530
     1            (GDA( 3),   E0),                (GDA( 4), SIG0),   00000531
     2            (GDA( 5),  POI),                (GDA( 6),  ENF),   00000532
```

```
     3            (GDA( 7), BCIT),                    (GDA( 8), BCIB),      00000533
     4            (GDA( 9),PFLAG),                    (GDA(10), CEXT),      00000534
     5            (GDA(11), DELT),                    (GDA(12),  THT),      00000535
     6            (GDA(18),FPRNT)                                           00000536
C                                                                           00000580
      COMMON  GDA(25), DEL, N, NNF, NTH,                                    00000590
     1 BMTX(160), TCTR, KZTW, KZTR, KPTW, KPTR                              00000591
C                                                                           00000608
C                                   READ AND PRINT 3 CARDS OF TITLE DATA    00000609
    5 READ (5,7)  BCD                                                       00000610
    7 FORMAT( 12A6 )                                                        00000611
      WRITE (6,8) BCD                                                       00000612
    8 FORMAT(1H1 /(18X, 12A6 //) )                                          00000613
C                                                                           00000619
      DO 10 L = 1,11                                                        00000620
      GDA(L+11)  =  -1.E+10                                                 00000625
   10 GDA(L) = 0.0                                                          00000630
      KZTR  =  8                                                            00000640
      KZTW  =  11                                                           00000650
      KPTR  =  10                                                           00000660
      KPTW  =  12                                                           00000670
C                                       READ GENERAL DATA                   00000680
      CALL DECRD (GDA)                                                      00000690
C                                                                           00000699
      IEXT  =  CEXT                                                         00000700
      IF(THT(1) .EQ. -1.E+10)  THT(1)  =  0.0                               00000702
      IF(FPRNT(1) .EQ. -1.E+10)  FPRNT(1)  =  ENF                           00000703
C                                                                           00000709
      DO    100    I = 1,IEXT                                               00000710
C                                         REWIND  ALL  TAPES                00000749
      REWIND  3                                                             00000750
      REWIND  4                                                             00000755
      REWIND  8                                                             00000760
      RFWIND  9                                                             00000770
      REWIND  10                                                            00000780
      REWIND  11                                                            00000790
      REWIND  12                                                            00000800
      REWIND  13                                                            00000810
C                                         SET  TIME  COUNTER                00000819
      TCTR  =  I                                                            00000820
C                                               DATA  LINK                  00000849
      CALL DATLNK (I)                                                       00000850
C                                          FORM 'P' AND 'X' MATRICES        00000919
```

```
      CALL PANDX                                                        00000920
      END FILE 3                                                        00000930
      REWIND  3                                                         00000940
      IF (TCTR .EQ. 2.0)  BACKSPACE 10                                  00000950
C                                                   SOLUTIONS           00000960
      CALL ZMTRX                                                        00000970
C                                         SUMS AND INTERNAL LOADS       00000980
      CALL SUMS                                                         00000985
C                                                                       00000989
  100 CONTINUE                                                          00000990
C                                                                       00001000
      GO  TO  5                                                         00001010
      END                                                               00001030
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS PO

3.12b DATLNK Subroutine (DLKDY)

```
$IBFTC DLKDY
C     SUB-EXECUTIVE ROUTINE FOR GEOM, DATLYR AND DATLD TO PERMIT ONE
C        COPY OF CODIMA, ENTERP AND DINTRP IN THE DECK SET-UP
C
      SUBROUTINE DATLNK (I)
C
      COMMON  GDA(25), DEL, N, NNF, NTH,
     1 BMTX(160), TCTR, KZTW, KZTR, KPTW, KPTR
C
      IF (I .NE. 1) GO TO 50
C                                   SET UP R, RHOX, GAMA, WTH, WFE, XSI
      CALL  GEOM
C
C                                   SET UP MATERIAL PROPERTIES
      CALL  DATLYR
      REWIND  9
C
C                                   SET UP FOURIER LOAD COEFF.
   50 CALL  DATLD
C
      RETURN
C
      END
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

3.12c GEOM Subroutine (GMYD)

```
$IBFTC GMYD                                                            00000010
C       * *  APOLLO WATER IMPACT STUDY    * *     GEOMETRY COMPUTATION ** 00000020
C                                                                      00000030
      SUBROUTINE GEOM                                                  00000040
C                                                                      00000050
C        GMI    -  GEOMETRY INDICATOR              VIZ.    GIN         00000060
C               =  1.0   - CONE - CYLINDER                  **         00000070
C               =  2.0   - SPHERE - TOROID                  **         00000080
C               =  3.0   - ARBITRARY FUNCTIONS              **         00000090
C               =  4.0   - GENERAL DISCRETE POINTS          **         00000100
C                 **                                        **         00000110
C        INPUTS -     GMI = 1.0                                        00000120
C                 RA1    =   RADIUS AT STATION 1                       00000130
C                 AXL    =   AXIAL SURFACE LENGTH                      00000140
C                 ANX    =   ANGLE - GENERATOR AND AXIS OF REVOLUTION  00000150
C                 **                                        **         00000160
C                      GMI = 2.0                                       00000170
C               RC   =  RADIUS OF CURVATURE                            00000180
C               ROFF   =  OFFSET DISTANCE TO CENTER OF CURVATURE       00000190
C               PHIO   = INITIAL  OPENING ANGLE FROM VERTICAL AXIS     00000200
C               PHIN   =  FINAL ANGLE                                  00000210
C                 **                                        **         00000220
C                    GMI   = 4.0  (- 4.0 DISCRETE ARCLENGTHS )         00000230
C               EM     =  NUMBER OF RIPTS GIVEN                        00000240
C               RIPT   =  DISCRETE RADII                               00000250
C               XIPT   =  DISCRETE XI'S (OR ARCLENGTHS)                00000260
C               RCURV  =  MERIDIONAL RADII OF CURVATURE                00000270
C               RCURZ  =  CIRCUMFERENTIAL RADII OF CURVATURE           00000280
C                 **                                        **         00000290
C                                                                      00000300
      DIMENSION WFW(16), WFP(4), STAP(4), STAW(16) , RRJ(12), RCRV(100),00000310
     1RCURZ(100), RCURV(100), RR(100), RIPT(100),XIPT(100), SARB(100),  00000320
     2SURF(100), XJ(11), RJ(11), DLR(10), R(100), XSI(100), WTHX(100),  00000330
     3WFEX(100), RHOX(100), GAMA(100), WFEPX(100), RCRZ(100), GMDA(442) 00000340
C                                                                      00000350
      DIMENSION BNDTX(17), BNDTB(17), THT(6), FPRNT(4)                 00000360
      DIMENSION  OMG0(5,5),  OMGN(5,5), ALM0(5,5), ALMN(5,5)           00000370
      DIMENSION  ALM1(5,5), OMG1(5,5)                                  00000380
      REAL  LL0(5),  LLLN(5)                                           00000390
C                                                                      00000400
      COMMON  GDA(25), DEL, N, NNF, NTH,                               00000410
     1 BMTX(160), TCTR, KZTW, KZTR, KPTW, KPTR                         00000420
C                                                                      00000430
```

```
      EQUIVALENCE  (BMTX(  1), OMG0),                (BMTX( 26), OMGN),
     1             (BMTX( 51), ALM0),                (BMTX( 76), ALMN),
     2             (BMTX(101), OMG1),                (BMTX(126), ALM1),
     3             (BMTX(151),  LL0),                (BMTX(156), LLLN)
C
      EQUIVALENCE (GDA( 1),    A0),              (GDA( 2),    H0),
     1            (GDA( 3),    E0),              (GDA( 4), SIG0),
     2            (GDA( 5),  POI),               (GDA( 6),  ENF),
     3            (GDA( 7), BCIT),               (GDA( 8), BCIB),
     4            (GDA( 9),PFLAG),               (GDA(10), CEXT),
     5            (GDA(11), DFLT),               (GDA(12),  THT),
     6            (GDA(18),FPRNT)
C
      EQUIVALENCE  (GMDA(  1),        GIN),  (GMDA(  2),        FN),
     1             (GMDA(  3),      BLANK),  (GMDA(  4),     RA1,RC),
     2             (GMDA(  5),   AXL,ROFF),  (GMDA(  6),   ANX,PHI0),
     3             (GMDA(  7),       PHIN),  (GMDA(  8),        FM),
     4             (GMDA(  9),       RIPT),  (GMDA(109),       XIPT),
     5             (GMDA(209),      RCURV),  (GMDA(309),     RCURZ),
     6             (GMDA(409),      BNDTX),  (GMDA(426),     BNDTB)
C
      DO  2  I = 1,442
    2 GMDA(I) = 0.0
C
      CALL DECRD (GMDA)
C
      DO  3 I = 1,160
    3 BMTX(I) = 0.0
C
      IF (BCIT .NE. 1.0) GO TO 5
C                                          CLOSED APEX
      OMG0(3,3) = 1.0
      OMG1 (1,1) = 1.0
      OMG1 (1,2) = -1.0
      OMG1 (5,4) = -1.0
      OMG1 (5,5) = 1.0
      ALM0(1,1) = 1.0
      ALM0(2,2) = 1.0
      ALM0(4,4) = 1.0
      ALM0(5,5) = 1.0
      ALM1 (2,1) = 1.0
      ALM1 (2,2) = 1.0
      ALM1(3,3) = 1.0
```


REPRODUCIBILITY OF THE ORIGINAL PAGE IS POO

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POO

```
      ALM1(4,4) = 1.0                                                   00000830
      ALM1 (4,5) = 1.0                                                  00000840
      GO TO 21                                                          00000850
C                                                                       00000860
    5 IF (BCIT .NE. 10.) GO TO 10                                       00000870
C                                                   SPECIAL BOUNDARY    00000880
      OMGO(1,1) = BNDTX( 1)                                             00000890
      OMGO(1,2) = BNDTX( 2)                                             00000900
      OMGO(2,2) = BNDTX( 3)                                             00000910
      OMGO(3,3) = BNDTX( 4)                                             00000920
      OMGO(4,4) = BNDTX( 5)                                             00000930
      OMGO (5,4) = BNDTX( 6)                                            00000940
      OMGO(5,5) = BNDTX( 7)                                             00000950
      ALMO(1,1) = BNDTX( 8)                                             00000960
      ALMO(2,2) = BNDTX( 9)                                             00000970
      ALMO(3,3) = BNDTX(10)                                             00000980
      ALMO(4,4) = BNDTX(11)                                             00000990
      ALMO(5,5) = BNDTX(12)                                             00001000
      DO  8 IJ = 1,5                                                    00001010
    8 LLO(IJ) = BNDTX(IJ+12)                                            00001020
      GO  TO  21                                                        00001030
C                                                                       00001040
   10 IBMX  =  BCIT                                                     00001050
      IO  =  1                                                          00001060
      IL  =  51                                                         00001070
C                                                   CHOOSE BOUNDARY     00001080
   11 GO TO (21,12,13,14,15,16), IBMX                                   00001090
C                                                   PINNED              00001100
   12 BMTX(IO + 18)  =  1.                                              00001110
      BMTX(IO + 24)  =  1.                                              00001120
      BMTX(IL)  =  1.                                                   00001130
      BMTX(IL + 6)  =  1.                                               00001140
      BMTX(IL + 12)  =  1.                                              00001150
      GO  TO  20                                                        00001160
C                                                   CLAMPED             00001170
   13 BMTX(IL)  =  1.                                                   00001180
      BMTX(IL + 6)  =  1.                                               00001190
      BMTX(IL + 12)  =  1.                                              00001200
      BMTX(IL + 18)  =  1.                                              00001210
      BMTX(IL + 24)  =  1.                                              00001230
      GO  TO  20                                                        00001240
C                                                   FREE                00001250
   14 BMTX(IO)  =  1.                                                   00001260
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      BMTX(IO + 6)  =  1.                                          00001270
      BMTX(IO + 12)  =  1.                                         00001280
      BMTX(IO + 18)  =  1.                                         00001290
      BMTX(IO + 24)  =  1.                                         00001300
      GO  TO  20                                                   00001310
C                                                   ROLLER  W      00001320
   15 BMTX(IO + 12)  =  1.                                         00001330
      BMTX(IO + 18)  =  1.                                         00001340
      BMTX(IO + 24)  =  1.                                         00001350
      BMTX(IL)  =  1.                                              00001360
      BMTX(IL + 6)  =  1.                                          00001370
      GO  TO  20                                                   00001380
C                                                   ROLLER  U      00001390
   16 BMTX(IO)  =  1.                                              00001400
      BMTX(IO + 18)  =  1.                                         00001410
      BMTX(IO + 24)  =  1.                                         00001420
      BMTX(IL + 6)  =  1.                                          00001430
      BMTX(IL + 12)  =  1.                                         00001440
C                                                                  00001450
   20 IF(IO .NE. 1)  GO TO 28                                      00001460
C                                                                  00001470
   21 IF (BCIB .NE. 1.0) GO TO 23                                  00001480
C                                                   CLOSED APEX    00001490
      OMGN(3,3) = 1.0                                              00001500
      OMG1 (1,1) = 1.0                                             00001510
      OMG1 (1,2) = -1.0                                            00001520
      OMG1 (5,4) = -1.0                                            00001530
      OMG1 (5,5) = 1.0                                             00001540
      ALMN(1,1)  = 1.0                                             00001550
      ALMN(2,2)  = 1.0                                             00001560
      ALMN(4,4)  = 1.0                                             00001570
      ALMN(5,5)  = 1.0                                             00001580
      ALM1 (2,1) = 1.0                                             00001590
      ALM1 (2,2) = 1.0                                             00001600
      ALM1 (3,3) = 1.0                                             00001610
      ALM1(4,4) = 1.0                                              00001620
      ALM1 (4,5) = 1.0                                             00001630
      GO TO 28                                                     00001640
C                                                                  00001650
   23 IF (BCIB .NE. 10.) GO TO 25                                  00001660
C                                                   SPECIAL BOUNDARY 00001670
      OMGN(1,1) = BNDTB( 1)                                        00001680
      OMGN(1,2) = BNDTB( 2)                                        00001690
```

```
      OMGN(2,2) = BNDTB( 3)                                             00001700
      OMGN(3,3) = BNDTB( 4)                                             00001710
      OMGN(4,4) = BNDTB( 5)                                             00001720
      OMGN (5,4) = BNDTB( 6)                                            00001730
      OMGN(5,5) = BNDTB(7)                                              00001740
      ALMN(1,1) = BNDTB(8)                                              00001750
      ALMN(2,2) = BNDTB( 9)                                             00001760
      ALMN(3,3) = BNDTB(10)                                             00001770
      ALMN(4,4) = BNDTB(11)                                             00001780
      ALMN(5,5) = BNDTB(12)                                             00001790
      DO 24 IJ = 1,5                                                    00001800
   24 LLLN(IJ) = BNDTB(IJ+12)                                           00001810
      GO  TO  28                                                        00001820
C                                                                       00001830
   25 IBMX  =  BCIB                                                     00001840
      IO  =  26                                                         00001850
      IL  =  76                                                         00001860
      GO  TO  11                                                        00001870
C                                                                       00001880
   28 WRITE (6,1000) A0, H0, E0, SIG0, POI, ENF, BCIT, BCIB, PFLAG,     00001890
     1  CEXT, DELT, THT, FPRNT,                                         00001891
     2  OMG0, ALM0, LL0, OMGN, ALMN, LLLN                               00001892
C                                                                       00001910
      N = EN                                                            00001920
      GMI = ABS(GIN)                                                    00001930
      NN = N - 1                                                        00001940
      IF (GMI - 2.0) 30, 35, 50                                         00001950
C                                                                       00001960
C                    CONE - CYLINDER                          **        00001970
C                                                                       00001980
   30 IF (PFLAG .NE. 0.0) WRITE(6,31)          N, RA1, AXL, ANX         00001990
C                                                                       00002000
   31 FORMAT (1H1,39X,33HGEOMETRY DATA FOR CONE - CYLINDER //35X,22HNUMB00002010
     1ER OF STATIONS -  ,I4//8X,7HRA1    =,1PE13.4,7X,7HAXL    =,E13.4,7X,00002020
     2 7HANX    =,E13.4)                                                00002030
C                                                                       00002040
      DEL =  AXL/(EN - 1.0)                                             00002050
      SINFI  = SIND(ANX)                                                00002060
      COSFI  = COSD(ANX)                                                00002070
      XSI(1) = 0.0                                                      00002080
      WTHX(1) = A0 * COSFI/RA1                                          00002090
      WFEX(1) = 0.0                                                     00002100
      RHOX(1) = RA1/A0                                                  00002110
```


REPRODUCIBILITY OF THE ORIGINAL PAGE IS

```
      R(1) = RA1                                                        00002120
      DO 32 I = 2,N                                                     00002130
      R(I)  = R(I-1) + DEL * SINFI                                      00002140
      XSI(I) = XSI(I-1) + DEL * COSFI                                   00002150
      WTHX(I) = A0 * COSFI/R(I)                                         00002160
      WFEX(I) = 0.0                                                     00002170
   32 RHOX(I) = R(I)/A0                                                 00002180
      GO TO 95                                                          00002190
C                                                                       00002200
C                    SPHERE - TOROID                                 **  00002210
C                                                                       00002220
   35 IF (PFLAG .NE. 0.0)  WRITE (6,37)  N, RC, ROFF, PHI0, PHIN        00002230
C                                                                       00002240
   37 FORMAT (1H1,39X,33HGEOMETRY DATA FOR SPHERE - TOROID //35X,22HNUMB00002250
     1ER OF STATIONS -  ,I4//6X,7HRC     =,1PE13.4,7X,7HROFF  =,E13.4,7X,00002260
     2 7HPHI0  =,E13.4,7X,7HPHIN  =,E13.4 )                             00002270
C                                                                       00002280
      ANGSP = PHIN - PHI0                                               00002290
      DEL =  ANGSP/(FN -1.0)                                            00002300
      AM = 1.0                                                          00002310
      AMU = SIGN(AM,DEL)                                                00002320
      XSI(1) =  0.0                                                     00002330
      BPHI = PHI0                                                       00002340
      BSINP = SIND(PHI0)                                                00002350
      BCOSP = COSD(PHI0)                                                00002360
      R(1) = RC * BSINP + ROFF                                          00002370
      DO 40 I = 1,NN                                                    00002380
      APHI = BPHI + DEL                                                 00002390
      ASINP = SIND(APHI)                                                00002400
      ACOSP = COSD(APHI)                                                00002410
      R(I+1) = R(I) + RC * (ASINP - BSINP)                              00002420
      XSI(I+1) = XSI(I) + RC * (BCOSP - ACOSP)                          00002430
      WFEX(I) = A0/RC * AMU                                             00002440
      IF (ROFF .EQ. 0.0) GO TO 38                                       00002450
      WTHX(I) = A0 * BSINP/R(I)                                         00002460
      GO TO 39                                                          00002470
   38 WTHX(I) = WFEX(I)                                                 00002480
   39 RHOX(I) = R(I)/A0                                                 00002490
      BPHI = APHI                                                       00002500
      BSINP = ASINP                                                     00002510
      BCOSP = ACOSP                                                     00002520
   40 CONTINUE                                                          00002530
C                                                                       00002540
```

```
      DFL = ABS(DFL)                                                    00002550
      WFFX(N) = A0/RC*AMU                                               00002560
      IF (ROFF .EQ. 0.0) GO TO 45                                       00002570
      WTHX(N) = A0 * RSINP /R(N)                                        00002580
      GO TO 46                                                          00002590
   45 WTHX(N) = WFEX(N)                                                 00002600
      IF (BCIB .EQ. 1.0) GO TO 47                                       00002610
   46 RHOX(N) = R(N)/A0                                                 00002620
      GO TO 48                                                          00002630
   47 RHOX(N) = 0.0                                                     00002640
   48 DFL = DFL * RC * 0.01745329                                       00002650
      GO TO 95                                                          00002660
C                                                                       00002670
   50 IF (GMI .GT. 3.0) GO TO 60                                        00002680
C                                                                       00002690
      WRITE (6,55)                                                      00002700
   55 FORMAT (1H1,43HARBITRARY FUNCTIONS CANNOT BE HANDLED, YET. )      00002710
C                                                                       00002720
      GO TO 140                                                         00002730
C                                                                       00002740
C                    GENERAL DISCRETE POINTS                    **      00002750
C                                                                       00002760
   60 M  =  EM                                                          00002770
      MM = M - 1                                                        00002780
      MM2 = M- 2                                                        00002790
C                                                                       00002800
      IF (PFLAG .NE. 0.0) WRITE(6,62)     N, (YIPT(I), XIPT(I), I = 1,M)00002810
C                                                                       00002820
   62 FORMAT (1H1,39X,31HGEOMETRY DATA - DISCRETE POINTS //35X,22HNUMBER00002830
     1 OF STATIONS -   ,I4//16X,1HR,16X,2HXI//(3X,1P2E20.7) )           00002840
C                                                                       00002850
      SARB(1) = 0.0                                                     00002860
      IF(GIN .LT. 0.) GO TO 77                                          00002870
      DO 75 IL = 1,MM                                                   00002880
      SURB = 0.0                                                        00002890
      DLT = XIPT(IL+1) - XIPT(IL)                                       00002900
      K = 10                                                            00002910
      AK = K                                                            00002920
      DDL = DLT/AK                                                      00002930
      KP1 = K + 1                                                       00002940
      DO 65 JI = 1,KP1                                                  00002950
      AJI = JI - 1                                                      00002960
      XJ(JI) = XIPT(IL) + AJI * DDL                                     00002970
```

```
   65 CONTINUE                                                          00002980
C                                                                       00002990
      CALL CODIMA (KP1,XJ, RRJ, XIPT, RIPT, M, 1.0)                     00003000
      DO  69 I = 2,K                                                    00003010
   69 RJ(I) =  (RRJ(I-1) + RRJ(I) + RRJ(I+1) )/3.0                      00003020
      RJ(1) = RRJ(1)                                                    00003030
      RJ(KP1) = RRJ(KP1)                                                00003040
C                                                                       00003050
      DO 70 JR = 1,K                                                    00003060
      DLR(JR) = RJ(JR+1) - RJ(JR)                                       00003070
      DLS     =  SQRT(DLR(JR)**2 + DDL**2)                              00003080
   70 SURB = SURB + DLS                                                 00003090
      SARB(IL+1) = SARB(IL) + SURB                                      00003100
   75 CONTINUE                                                          00003120
      GO TO 80                                                          00003130
   77 DO 78 I = 1,M                                                     00003140
   78 SARB(I) = XIPT(I)                                                 00003150
   80 DEL = SARB(M) /(FN - 1.0)                                         00003160
C                                                                       00003170
      SURF(1) = 0.0                                                     00003180
      DO 82 I = 1,NN                                                    00003190
   82 SURF(I+1) = SURF(I) + DEL                                         00003200
      CALL CODIMA(N, SURF,RCRV, SARB, RCURV, M, 1.0)                    00003210
      CALL CODIMA(N,SURF, RCRZ, SARB, RCURZ, M, 1.0)                    00003220
C                                                                       00003230
      CALL CODIMA (N,SURF, R, SARB, RIPT, M, 1.0)                       00003240
C                                                                       00003250
      CALL CODIMA (N, SURF, XSI, SARB, XIPT, M, -1.0)                   00003260
      MLN = N - 2                                                       00003270
      NSM = 1                                                           00003280
   84 DO 85 I = 3,MLN                                                   00003290
      RR(I) = (-3.*R(I-2) + 12.*R(I-1) + 17.*R(I) + 12.*R(I+1) - 3. *   00003300
     1 R(I+2) )/ 35.0                                                   00003310
   85 CONTINUE                                                          00003320
      RR(NN) = R(NN)                                                    00003330
      RR(2) = R(2)                                                      00003340
      RR(N) = R(N)                                                      00003350
      RR(1) = R(1)                                                      00003360
      IF (NSM .EQ. 25) GO TO 88                                         00003370
      NSM = NSM + 1                                                     00003380
      DO  87 I = 1,N                                                    00003390
   87 R(I) = RR(I)                                                      00003400
      GO TO  84                                                         00003410
```

```
   88 RHOX(1) = RR(1)/A0                                                00003420
      DELSQ = DEL * DEL                                                 00003430
      DO 89 I = 1,NN                                                    00003440
   89 RHOX(I+1) = RR(I+1) /A0                                           00003450
      GO TO 110                                                         00003460
C                                                                       00003470
C                    COMPUTE GAMA                                  **   00003480
C                                                                       00003490
   95  DEL = DEL/ A0                                                    00003500
       DELSQ = DEL * DEL                                                00003510
       DO 105 I = 1,N                                                   00003520
      DENM = 12. * RHOX(I) * DEL                                        00003530
      DENMP= 2. * RHOX(I) * DEL                                         00003540
      IF(RHOX(I, .EQ. 0.)  GO TO 97                                     00003550
      IF (I .NE. 1) GO TO 98                                            00003560
      IF (BCIT .EQ. 1.0) GO TO 103                                      00003570
      GAMA(I) = (3.*(RHOX(I+1) - RHOX(I)) + RHOX(I+1) - RHOX(I+2))/DENMP00003580
      GO TO 105                                                         00003590
   97 GAMA(I)  =  0.0                                                   00003600
      GO  TO  105                                                       00003610
   98 IF (I .EQ. N) GO TO 100                                           00003620
      IF (I .EQ. 2) GO TO 99                                            00003630
      IF (I .EQ. N-1) GO TO 99                                          00003640
      GAMA(I) = (RHOX(I-2) - 8. *(RHOX(I-1) -      RHOX(I+1))- RHOX(I+2)00003650
     1 )/DENM                                                           00003660
      GO TO 105                                                         00003670
   99 GAMA(I) = (RHOX(I+1) - RHOX(I-1)) /DENMP                          00003680
      GO TO 105                                                         00003690
  100 IF (BCIB .EQ. 1.0)  GO TO 103                                     00003700
      GAMA(I) =(3.*(RHOX(I)-RHOX(I-1)) + RHOX(I-2) - RHOX(I-1)) /DENMP  00003710
      GO  TO  105                                                       00003720
  103 GAMA(I)  =  0.                                                    00003730
  105 CONTINUE                                                          00003740
       GO TO 123                                                        00003750
C                                                                       00003760
  110 DO 115 I = 1,N                                                    00003770
      WFEX(I) = A0/RCRV(I)                                              00003780
      WTHX(I) = A0/RCRZ(I)                                              00003790
      IF (RHOX(I) .EQ. 0.0) GO TO 113                                   00003800
      PRO = (RHOX(I) * WTHX(I))**2                                      00003810
      IF (PRO .GT. 1.0) GO TO 114                                       00003820
      GAMA(I) = SQRT(1.-PRO)/RHOX(I)                                    00003830
      GO TO 115                                                         00003840
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
  113 GAMA(I) = 0.0                                                     00003850
      GO TO 115                                                         00003860
  114 GAMA(I) = 0.0                                                     00003870
  115 CONTINUE                                                          00003880
C                                                                       00003890
  123 DO 127 I = 1,N                                                    00003900
      WFDNM  =  2. * DFL                                                00003910
      IF (I .NE. 1) GO TO 124                                           00003920
      WFEPX(I) = (4. * WFEX(I+1) - 3. * WFEX(I) - WFEX(I+2)) /WFDNM     00003930
      GO TO 127                                                         00003940
  124 IF (I .EQ. N) GO TO 126                                           00003950
      WFEPX(I) = (WFEX(I+1) - WFEX(I-1)) /WFDNM                         00003960
      GO TO 127                                                         00003970
  126 WFEPX(I) = (WFEX(I-2) + 3. * WFEX(I) - 4. * WFEX(I-1)) /WFDNM     00003980
  127 CONTINUE                                                          00003990
C                                                                       00003999
      WRITE ( 3) WTHX, WFEX, GAMA, RHOX, WFEPX                          00004000
C                                                                       00004010
  129 IF (PFLAG .EQ. 0.0) GO TO 140                                     00004020
      WRITE (6,130) DEL, (I, R(I), XSI(I), WTHX(I), WFEX(I), RHOX(I),   00004030
     1 GAMA(I), I = 1,N)                                                00004050
C                                                                       00004060
  130 FORMAT (1H-, 5X, 7HDFL    =,1PE13.4 //                            00004070
     1            3X,1HI,9X,4HR(I),12X,5HXI(I),10X,8HW(THETA),11X,      00004071
     2 5HW(XI),11X,7HRHOX(I),10X,7HGAMA(I) //(I4, 1P6E17.7) )           00004072
C                                                                       00004090
  140 RETURN                                                            00004100
C                                                                       00004110
 1000 FORMAT(1H1,47X,12HGENERAL DATA/// 3X,7HA0     =,1PE13.4, 8X,7HH0  00004120
     1 =,E13.4, 8X,7HE0     =,E13.4, 8X,7HSIG0   =,E13.4// 3X,7HPOI   =, 00004121
     2 E13.4, 8X,7HENF    =,E13.4, 8X,7HBCIT   =,E13.4, 8X,7HBCIB   =,  00004122
     3 E13.4//3X,7HPFLAG =,E13.4, 8X,7HCFXT   =,E13.4, 8X,7HDFLT   =,   00004123
     4 E13.4//6X,20HPRINT ANGLES (THT) - // 24X,6E13.4 //6X,33HFOURIER H00004124
     5ARMONIC PRINTS (FPRNT) - // 24X, 4E13.4                           00004125
     6                                          ///1H-,45X,17HBOUNDAR00004126
     7Y MATRICES /6X,15HTOP BOUNDARY **// 6X,5HOMEGA /5(5E17.7)// 6X,  00004127
     8 6HLAMBDA /5(5E17.7)// 6X,3HL -/ 5E17.7////6X,18HBOTTOM BOUNDARY 00004128
     9** // 6X,5HOMEGA /5(5E17.7)// 6X,6HLAMBDA /5(5E17.7)// 6X,3HL -/00004129
     X 5E17.7  )                                                        00004130
      END                                                               00009999
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS PO

3.12d DATLYR Subroutine (STFDY)

```
$IBFTC STFDY
C              FOURIER STIFFNESS COEFFICIENTS GENERATOR ** FUNCTIONS
C
      SUBROUTINE DATLYR
C
C     ***  STIFFNESS COEFF. ARE STORED IN SECTIONS OF ALL FOURIER
C          COMPONENTS /MERIDIONAL STATION, UP TO BLOCKS OF 100
C
      DIMENSION  COEFC(2200)
      DIMENSION  SCB1(100), SCB2(100), SCB3(100), SCG1(100), SCG2(100),
     1 SCG3(100), SCD1(100), SCD2(100), SCD3(100), SCG13(100),
     2 SCNTSI(100), SCNTTH(100), SCMTSI(100), SCMTTH(100), DCM1(100),
     3 DCM4(100), DCC1(100), DCC2(100), DCC3(100), DCK1(100), DCK2(100),
     4 DCK3(100)
C
      COMMON  GDA(25), DEL, N, NNF, NTH,
     1 BMTX(160), TCTR, KZTW, KZTR, KPTW, KPTR
C
C
      EQUIVALENCE (GDA( 1),   A0),            (GDA( 2),   H0),
     1            (GDA( 3),   E0),            (GDA( 4), SIG0),
     2            (GDA( 5),  POI),            (GDA( 6),  ENF),
     3            (GDA( 7), BCIT),            (GDA( 8), BCIB),
     4            (GDA( 9),PFLAG),            (GDA(10), CEXT),
     5            (GDA(11), DELT),            (GDA(12),  THT),
     6            (GDA(18),FPRNT)
C
      EQUIVALENCE (COEFC(   1),  SCB1),          (COEFC( 101),  SCB2),
     1            (COEFC( 201),  SCB3),          (COEFC( 301),  SCG1),
     2            (COEFC( 401),  SCG2),          (COEFC( 501),  SCG3),
     3            (COEFC( 601),  SCD1),          (COEFC( 701),  SCD2),
     4            (COEFC( 801),  SCD3),
     5 (COEFC( 901), SCG13), (COEFC(1001),SCNTSI), (COEFC(1101),SCNTTH),
     6 (COEFC(1201),SCMTSI), (COEFC(1301),SCMTTH), (COEFC(1401),  DCM1),
     7 (COEFC(1501),  DCM4), (COEFC(1601),  DCC1), (COEFC(1701),  DCC2),
     8 (COEFC(1801),  DCC3), (COEFC(1901),  DCK1), (COEFC(2001),  DCK2),
     9 (COEFC(2101),  DCK3)
C
C
      IF(PFLAG .LT. 0.)  WRITE(6,10)
   10 FORMAT(1H1/6X, 96HCHECKPRINT OF COEFC BLOCKS STORED IN GROUPS OF (
     12*ENF) PER MERIDIONAL STATION.  THE COEFFICIENTS /103H PRINTED ARE
     2 SCB1, SCD1, SCNTSI, SCMTSI, DCM1, DCM4, DCC1, DCC2, DCC3, DCK1, D
```

```
     3CK2, DCK3 (IN THAT ORDER) / 62H AND ARE SEPARATED FROM EACH OTHER 00000435
     4BY THE THREE ...... FIELDS.  ///)                                 00000436
      ENTH  =  90.                                                      00000440
      H02 = H0 * H0                                                     00000450
      SS3 = 2./3.1415927                                                00000460
      NTH  = ENTH                                                       00000470
      ENNF2 = ENF + ENF                                                 00000480
      NNF  =  ENF                                                       00000490
      NNF2  = 2 * NNF                                                   00000500
      IN = I                                                            00000510
C                DETERMINE SIZE OF BLOCK                      **        00000518
C                                                                       00000519
      MNF  = N * NNF2                                                   00000520
   40 IF (MNF .GT. 100) GO TO 50                                        00000530
C                                        LESS THAN ONE BLOCK MORE       00000539
      LN = N                                                            00000540
      GO TO 60                                                          00000550
   50 IF (MNF .LT. 200) GO TO 55                                        00000560
C                                      AT LEAST 2 FULL BLOCKS MORE      00000569
      LN = 100/NNF2                                                     00000570
      GO TO 58                                                          00000580
C                                                                       00000584
C                            GREATER THAN ONE BLOCK, LESS THAN TWO      00000585
C                         AVOID HAVING ONLY TWO STATIONS IN A BLOCK.    00000586
C                                                                       00000587
   55 LN = 50/NNF2                                                      00000590
   58 LN = IN + LN - 1                                                  00000600
C                                                                       00000620
   60 DFLTH = 3.1415927/ ENTH                                           00000630
C                                                                       00000640
C                                      ZERO COEFFICIENT AREA            00000649
   63 DO  65  I = 1,2200                                                00000650
   65 COEFC(I) = 0.0                                                    00000660
C                                                                       00000670
   70 DO 100 I = IN, LN                                                 00000680
C                                ESTABLISH MERIDIONAL DISTANCE          00000689
      EI = I - 1                                                        00000690
      PEL = EI * DEL                                                    00000700
      K1 = NNF2 * (I - IN)                                              00000710
C                                                                       00000719
      DO 100  NN = 1,NNF2                                               00000720
      K2 = K1 + NN                                                      00000730
      ENN = NN - 1                                                      00000740
```


REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POO

```
C                                                                       00000770
   75 DO 95 J = 1,NTH                                                   00000780
      EJ = J - 1                                                        00000790
      ZTA = EJ * DELTH + DELTH * 0.5                                    00000800
      IF (NN .EQ. 1) GO TO 90                                           00000810
C                                                                       00000820
C                                          OTHER FOURIER COMPONENTS     00000839
   80 SCB1(K2) = SCB1(K2) + BBB(PEL,ZTA)     * COS (ENN *  ZTA)         00000840
      SCD1(K2) = SCD1(K2) + DDD(ZTA)         * COS (ENN *  ZTA)         00000845
      SCNTSI(K2) = SCNTSI(K2) + ENTT(PEL,ZTA) * COS(ENN * ZTA)          00000850
      SCMTSI(K2) = SCMTSI(K2) + EMTT(PEL,ZTA) * COS(ENN * ZTA)          00000855
      DCM1(K2) =DCM1(K2)+  DMM1(PEL,ZTA)  * COS(ENN* ZTA)               00000860
      DCM4(K2) =DCM4(K2)+  DMM4(PEL,ZTA)  * COS(ENN* ZTA)               00000870
      DCK1(K2) =DCK1(K2)+  DKK1(PEL,ZTA)  * COS(ENN* ZTA)               00000880
      DCC1(K2) =DCC1(K2)+  DMP1(PEL,ZTA)  * COS(ENN* ZTA)               00000885
      DCK2(K2) =DCK2(K2)+  DKK2(PEL,ZTA)  * COS(ENN* ZTA)               00000890
      DCC2(K2) =DCC2(K2)+  DMP2(PEL,ZTA)  * COS(ENN* ZTA)               00000895
      DCK3(K2) =DCK3(K2)+  DKK3(PEL,ZTA)  * COS(ENN* ZTA)               00000900
      DCC3(K2) =DCC3(K2)+  DMP3(PEL,ZTA)  * COS(ENN* ZTA)               00000905
      GO TO 95                                                          00000910
C                                          FIRST FOURIER COMPONENT      00000919
   90 SCB1(K2) = SCB1(K2) + BBB(PEL,ZTA) / 2.0                          00000920
      SCD1(K2) = SCD1(K2) + DDD(ZTA) / 2.0                              00000925
      SCNTSI(K2) = SCNTSI(K2) + ENTT(PEL,ZTA) / 2.0                     00000930
      SCMTSI(K2) = SCMTSI(K2) + EMTT(PEL,ZTA) / 2.0                     00000935
      DCM1(K2) =DCM1(K2) + DMM1(PEL,ZTA)/2.0                            00000940
      DCM4(K2) =DCM4(K2) + DMM4(PEL,ZTA)/2.0                            00000950
      DCK1(K2) =DCK1(K2) + DKK1(PEL,ZTA)/2.0                            00000960
      DCC1(K2) =DCC1(K2) + DMP1(PEL,ZTA)/2.0                            00000965
      DCK2(K2) =DCK2(K2) + DKK2(PEL,ZTA)/2.0                            00000970
      DCC2(K2) =DCC2(K2) + DMP2(PEL,ZTA)/2.0                            00000975
      DCK3(K2) =DCK3(K2) + DKK3(PEL,ZTA)/2.0                            00000980
      DCC3(K2) =DCC3(K2) + DMP3(PEL,ZTA)/2.0                            00000985
   95 CONTINUE                                                          00000990
C                                                                       00001000
C                                                   NORMALIZE           00001009
      S3 = SS3/(E0 * H0)                                                00001010
      SCB1(K2) = S3 * DELTH * SCB1(K2)                                  00001020
      S4 = S3 * A0*A0                                                   00001030
      DCM1(K2) = S4* DELTH * DCM1(K2)                                   00001040
      DCM4(K2) = S3 * DELTH * DCM4(K2)                                  00001050
      DCK1(K2) = S4 * DELTH * DCK1(K2)                                  00001060
      DCC1(K2) = S4 * DELTH * DCC1(K2)                                  00001065
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      DCK2(K2) = S4 * DELTH * DCK2(K2)                                  00001070
      DCC2(K2) = S4 * DELTH * DCC2(K2)                                  00001075
      DCK3(K2) = S4 * DELTH * DCK3(K2)                                  00001080
      DCC3(K2) = S4 * DELTH * DCC3(K2)                                  00001083
      S3 = SS3 /(SIGO * HO)                                             00001085
      SCNTSI(K2) = S3 * DELTH * SCNTSI(K2)                              00001090
      S3 = SS3 * AO /(SIGO * HO2 * HO)                                  00001093
      SCMTSI(K2) = S3 * DELTH * SCMTSI(K2)                              00001096
      S3 = SS3/(EO * HO2 * HO)                                          00001100
      SCD1(K2) = S3 * DELTH * SCD1(K2)                                  00001110
C                                                                       00001120
   98 SS2 = (1. - POI) / 2.                                             00001130
      SCG13(K2) = SS2 * SCD1(K2)                                        00001140
      SCG1(K2) = SS2 * SCB1(K2)                                         00001150
      SCB2(K2) = SCB1(K2)                                               00001160
      SCB3(K2) = POI * SCB1(K2)                                         00001170
      SCG2(K2) = SS2 * SCB1(K2)                                         00001180
      SCG3(K2) = SCG1(K2)                                               00001190
      SCD2(K2) = SCD1(K2)                                               00001200
      SCD3(K2) = POI * SCD1(K2)                                         00001210
  100 CONTINUE                                                          00001280
C                                                                       00001290
      WRITE ( 9) LN, IN, COEFC                                          00001300
      NX  =  (LN - IN + 1) * NNF2                                       00001305
      XX  =  GDA(25)                                                    00001306
      IF(PFLAG .LT. 0.)                                                 00001309
     XWRITE (6,105)  IN, LN, (SCB1(I), I=1,NX), XX, XX, XX, (SCD1(I),   00001310
     1 I=1,NX), XX, XX, XX, (SCNTSI(I), I=1,NX), XX, XX, XX, (SCMTSI(I),00001311
     2 I=1,NX), XX, XX, XX, (DCM1(I), I=1,NX), XX, XX, XX, (DCM4(I), I  00001312
     3 = 1,NX), XX, XX, XX, (DCC1(I), I=1,NX), XX, XX, XX, (DCC2(I), I  00001313
     4 = 1,NX), XX, XX, XX, (DCC3(I), I=1,NX), XX, XX, XX, (DCK1(I), I  00001314
     5 = 1,NX), XX, XX, XX, (DCK2(I), I=1,NX), XX, XX, XX, (DCK3(I), I  00001315
     6 = 1,NX)                                                          00001316
  105 FORMAT(/10X,11HFOR STATION, I3,16H THROUGH STATION, I4///         00001320
     1       ( 1P6E17.7) )                                              00001321
C                                                                       00001330
      IF (LN .EQ. N) GO TO 110                                          00001340
      IN = LN + 1                                                       00001350
      NCN = N - LN                                                      00001360
      MNF = NCN * NNF2                                                  00001370
      GO TO 40                                                          00001380
  110 RETURN                                                            00001390
C                                                                       00001400

      END                                                               00001410
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

3.12e DATLD Subroutine (DLDY)

```
$IBFTC DLDY                                                            00000010
C              FOURIER LOAD COEFFICIENTS GENERATOR BY FUNCTIONS ONLY   00000020
C                                                                      00000030
      SUBROUTINE  DATLD                                                00000040
C                                                                      00000050
C                                                                      00000060
      DIMENSION  PFE(10,50), PTH(10,50), PN(10,50)                     00000340
      EQUIVALENCE (GDA( 1),   A0),            (GDA( 2),   H0),         00000440
     1            (GDA( 3),   E0),            (GDA( 4), SIG0),         00000450
     2            (GDA( 5),  POI),            (GDA( 6),  ENF),         00000460
     3            (GDA( 7), BCIT),            (GDA( 8), BCIB),         00000470
     4            (GDA( 9),PFLAG),            (GDA(10), CEXT),         00000480
     5            (GDA(11), DELT),            (GDA(12),  THT),         00000481
     6            (GDA(18),FPRNT)                                      00000482
C                                                                      00000490
      COMMON  GDA(25), DFL, N, NNF, NTH,                               00000500
     1 BMTX(160), TCTR, KZTW, KZTR, KPTW, KPTR                         00000510
C                                                                      00000520
C              INITIALIZE *          *          *          *           00000525
      NAB = 0                                                          00000530
      SS3 = 2. /3.1415927                                              00000550
      S3 = SS3 * A0/(SIG0 * H0)                                        00000560
      NUN = 2                                                          00000570
      IF (N .GT. 50)  NUN = 3                                          00000580
      IF (TCTR .GT. 1.0)  AF = FILE (9,NUN)                            00000590
C                                                                      00000600
      DO 20 I = 1,50                                                   00000610
      DO 20 K = 1,10                                                   00000620
      PFE(K,I) = 0.0                                                   00000630
      PTH(K,I) = 0.0                                                   00000650
      PN(K,I)  = 0.0                                                   00000670
   20 CONTINUE                                                         00000690
C                                                                      00000720
      IF(PFLAG .LT. 0.)  WRITE(6,30)                                   00000730
   30 FORMAT(1H1/6X, 98HCHECKPRINT OF LOAD COEF. STORED IN GROUPS OF ALL00000740
     1 ENF PER MERIDIONAL STATION.  THE COEFFICIENTS ARE  / 81H PFE, PTH00000741
     2, PN (IN THAT ORDER) AND ARE SEPARATED FROM EACH OTHER BY ,,,,,,, F00000742
     3IELDS.  ///)                                                     00000743
      NI = N                                                           00000770
C                                                                      00000790
C                                  SPECIAL  FUNCTIONS                  00000795
C                                                                      00000800
   50 EI = 0.0                                                         00000810
```

```
      NI1 = 1                                                           00000820
      NI2  =  N                                                         00000830
      ENTH  =  NTH                                                      00000840
      DELTH = 3.1415927 / ENTH                                          00000850
C                                                                       00000870
C                 EACH BLOCK OF LOADS CANNOT CONTAIN MORE THAN 50 STAS. 00000873
C                                                                       00000876
      IF (NI .GT. 50) NI = 50                                           00000880
   70 DO 100 I = 1,NI                                                   00000890
      EI = EI + 1.0                                                     00000900
      PEL = (EI - 1.0) * DEL                                            00000910
C                                                                       00000919
      DO 100 NN = 1,NNF                                                 00000920
      ENN = NN- 1                                                       00000930
      TU = DELT * TCTR                                                  00000940
C                                                                       00000960
      DO 95 J = 1,NTH                                                   00000970
      EJ = J - 1                                                        00000980
      ZTA = EJ * DELTH + DELTH * 0.5                                    00000990
      IF (NN .EQ. 1)  GO TO 90                                          00001000
C                                                                       00001010
      PTH(NN,I) = PTH(NN,I)+PPPH(PEL,ZTA,TU)* SIN(ENN * (ZTA-1.5707963))00001030
      PFE(NN,I) = PFE(NN,I)  +PPPF(PEL,ZTA,TU)*COS(ENN * ZTA)           00001040
      PN(NN,I) = PN(NN,I) + PPPN(PEL,ZTA,TU)* COS(ENN * ZTA)            00001050
      GO TO 95                                                          00001060
C                                                                       00001069
   90 PTH(NN,I) = PTH(NN,I) + PPPH(PEL,ZTA,TU) /2.                      00001080
      PFE(NN,I) = PFE(NN,I) + PPPF(PEL,ZTA,TU) /2.                      00001090
      PN(NN,I)= PN(NN,I) + PPPN(PEL,ZTA,TU)/2.0                         00001100
   95 CONTINUE                                                          00001110
C                                                                       00001120
      PTH(NN,I) = S3 * DELTH * PTH(NN,I)                                00001130
      PFE(NN,I) = S3 * DELTH * PFE(NN,I)                                00001140
      PN(NN,I) = S3 * DELTH * PN(NN,I)                                  00001150
  100 CONTINUE                                                          00001160
C                                                                       00001170
      IF (TCTR .EQ. 1.0) GO TO 105                                      00001180
      WRITE ( 9) PFE, PTH, PN                                           00001190
      GO TO 107                                                         00001200
  105 WRITE ( 3) PFE, PTH, PN                                           00001210
  107 XX  =  GDA(25)                                                    00001220
      IF(N .GT. 50 .AND. NAB.EQ. 0)  NI2 = 50                           00001222
      IF(PFLAG .LT. 0.)                                                 00001229
```


REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR.

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      XWRITE (6,108) NI1, NI2, ((PFE(NN,I), NN=1,NNF), I=1,NI), XX, XX, XX 00001230
      IX, ((PTH(NN, I), NN=1,NNF), I=1,NI), XX, XX, XX, ((PN(NN,I), NN=1,00001231
      2NNF), I=1,NI)                                                    00001232
  108 FORMAT(/10X,11HFOR STATION, I3,16H THROUGH STATION, I4///         00001235
      1       ( 1P6E17.7) )                                             00001236
C                                                                       00001240
      NAB = NAB + NI                                                    00001250
      IF (NAB .EQ. N) GO TO 600                                         00001260
C                                                                       00001264
C                 MORE THAN 50 STATIONS TO STORE                  *     00001265
C                                                                       00001266
      NI = N - NI                                                       00001270
      NI1  =  51                                                        00001281
      NI2  =  N                                                         00001282
      DO 115 I = 1,50                                                   00001290
      DO 110 J = 1,10                                                   00001300
      PTH(J,I) = 0.0                                                    00001310
      PFE(J,I) = 0.0                                                    00001320
  110 PN(J,I) = 0.0                                                     00001330
  115 CONTINUE                                                          00001340
C                                                                       00001350
      GO TO 70                                                          00001360
C                                                                       00001370
  600 IF (TCTR .NE. 1.0) GO TO 610                                      00002820
      REWIND 3                                                          00002830
      RETURN                                                            00002840
C                                                                       00002850
  610 IF (N .GT. 50) BACKSPACE  9                                       00002860
      BACKSPACE  9                                                      00002870
      RETURN                                                            00002880
C                                                                       00002890
      END                                                               00002910
```

```
$IBFTC PXDYN                                                            00000010
C          GENERATES THE 'P' AND 'X' MATRICES                    **     00000020
C              NOTE    RECURSION TERMS ARE ALSO GENERATED        **     00000030
C                                                                       00000040
      SUBROUTINE  PANDX                                                 00000050
C                                                                       00000060
C     AO          REFERENCE LENGTH (IN)                                 00000070
C     HO          REFERENCE THICKNESS (IN)                              00000080
C     EO          REFERENCE YOUNGS MODULUS  (PSI)                       00000090
C     SIGO        REFERENCE STRESS  (PSI)                               00000100
C     NMF         NO. OF TERMS IN THE FOURIER EXPANSION.                00000110
C     LAM         LAMBDA  =  HO /AO                                     00000120
C     WFE         OMEGA (XI)                                            00000130
C     WTH         OMEGA (THETA)                                         00000140
C      ARRAYS USED                                                      00000150
C          COEFC    -   CONTAINS STIFFNESS COEFF CALCULATED IN DATLYR   00000160
C          P1SCB1   -   TEMP STORAG   OR STIFFNESS COEFF AT LAST TWO    00000170
C          P2SCB1   -                 TIONS OF THE BLOCK                00000171
C      PANDX FLAGS                                                      00000190
C          JPATH = 0    PATH IND. THRU ENF LOOP WHEN F, G, H INCOMPLETE 00000196
C          JPATH = 1    PATH IND. THRU ENF LOOP TO SKIP FGHPE AND BND.  00000197
C          KPATH = 0    PATH IND. THRU ENF LOOP WHEN L, M, N INCOMPLETE 00000198
C          KPATH = 1    PATH IND. THRU ENF LOOP TO SKIP DYLMN AND BND.  00000199
C          IRTE .NE. 1   FIRST STATION  -  NOT FIRST BLOCK              00000200
C          IRTE .EQ. 1   LAST STATION - PREVIOUS BLOCK                  00000210
C          IFGFG = 10   F, G, H, PE  HAVE NOT BEEN COMPLETED, YET.      00000220
C          IFGFG = 0    P AND X  HAVE BEEN COMPUTED FOR IRTE = 1.       00000230
C          IFGFG = 1    P AND X  COMPUTED.  INCREMENT STA. INDEX.       00000240
C          IRSFG = 10   R, S, LT,  HAVE NOT BEEN COMPLETED.             00000250
C          IRSFG = 1   INITIAL BOUNDARY COMPUTED.                       00000260
C          IRSFG = 0   TERMINAL BOUNDARY COMPUTED.                      00000270
C          IFDYN = 10   AL, AM, AN HAVE NOT BEEN COMPLETED.             00000280
C          IFDYN = 1    AK, AM, AN COMPUTED.                            00000290
C          IFMX  = 10   PE NOT COMPLETED.                               00000300
C          IFMX  =  1   PE COMPUTED.                                    00000310
C      COUNTERS  -                                                      00000312
C          MM    - ABSOLUTE STATION NUMBER.          *          *      00000313
C          I     - STATION NO. WITHIN EACH BLOCK OF STIFF. COEFF.       00000314
C          MMN   - STATION NO. WITHIN BLOCK OF LOADS.                   00000315
C          IFLG - STIFF. COEFF. BLOCK. *             *                 00000316
C                                                                       00000320
      EQUIVALENCE  (BMTX(  1), OMGO),              (BMTX( 26), OMGN),   00000330
     1             (BMTX( 51), ALMO),              (BMTX( 76), ALMN),   00000340
```

```
     2              (BMTX(101), OMG1),              (BMTX(126), ALM1),     00000350
     3              (BMTX(151),  LL0),              (BMTX(156), LLLN)      00000360
C                                                                         00000370
      EQUIVALENCE (GDA( 1),   A0),              (GDA( 2),   H0),          00000380
     1            (GDA( 3),   E0),              (GDA( 4), SIG0),          00000390
     2            (GDA( 5),  POI),              (GDA( 6),  ENF),          00000400
     3            (GDA( 7), BCIT),              (GDA( 8), BCIB),          00000410
     4            (GDA( 9),PFLAG),              (GDA(10), CEXTL),         00000420
     5            (GDA(11), DELT),              (GDA(12),  THT),          00000421
     6            (GDA(18),FPRNT)                                         00000422
C                                                                         00000440
      DIMENSION  WTHX(100), WFEX(100), RHOX(100), GAMA(100), WFEPX(100) 00000450
      DIMENSION  ALM0(5,5), ALMN(5,5), OMG0(5,5), OMGN(5,5)              00000460
      DIMENSION  ALM1(5,5), OMG1(5,5)                                    00000470
      REAL  LL0(5),  LLLN(5),                        MFE(10), MTP(10)    00000480
      DIMENSION  P1SCB1(20), P2SCB1(20), P1SCD1(20), P2SCD1(20),         00000490
     1  TFE(10), P1SG13(20), P2SG13(20), P1ENT(20), P2ENT(20), PN50(10),00000491
     2  TTP(10), P1SCG1(20), P2SCG1(20), P1FMT(20), P2EMT(20),           00000492
     3  P2DCK1(20), P2DCK2(20), P2DCK3(20), P2DCM1(20), P2DCM4(20),      00000493
     4  P2DCC1(20), P2DCC2(20), P2DCC3(20), PFE50(10), PTH50(10)         00000494
      DIMENSION   PFE(10, 50),  PTH(10, 50),   PN(10, 50)                00000540
      DIMENSION  P(50,50), X(50)                                         00000570
C                                                                         00000580
      DIMENSION  COEFC(2200)                                             00000590
      DIMENSION  SCB1(100), SCB2(100), SCB3(100), SCG1(100), SCG2(100), 00000600
     1 SCG3(100), SCD1(100), SCD2(100), SCD3(100), SCG13(100),           00000610
     2 SCNTSI(100), SCNTTH(100), SCMTSI(100), SCMTTH(100), DCM1(100),    00000620
     3 DCM4(100), DCC1(100), DCC2(100), DCC3(100), DCK1(100), DCK2(100),00000630
     4 DCK3(100)                                                         00000640
C                                                                         00000649
      EQUIVALENCE (COEFC(   1),  SCB1),          (COEFC( 101),  SCB2), 00000650
     1            (COEFC( 201),  SCB3),          (COEFC( 301),  SCG1), 00000660
     2            (COEFC( 401),  SCG2),          (COEFC( 501),  SCG3), 00000670
     3            (COEFC( 601),  SCD1),          (COEFC( 701),  SCD2), 00000680
     4            (COEFC( 801),  SCD3),                                 00000690
     5 (COEFC( 901), SCG13), (COEFC(1001),SCNTSI), (COEFC(1101),SCNTTH),00000700
     6 (COEFC(1201),SCMTSI), (COEFC(1301),SCMTTH), (COEFC(1401),  DCM1),00000710
     7 (COEFC(1501),  DCM4), (COEFC(1601),  DCC1), (COEFC(1701),  DCC2),00000720
     8 (COEFC(1801),  DCC3), (COEFC(1901),  DCK1), (COEFC(2001),  DCK2),00000730
     9 (COEFC(2101),  DCK3)                                              00000740
C                                                                         00000760
      COMMON  GDA(25), DEL, N, NNF, NTH,                                 00000770
     1 BMTX(160), TCTR, KZTW, KZTR, KPTW, KPTR                           00000780
```

```
C
      REAL   LAM, LAMO2, LAM2O2, LAM2O4, LAM2O8, LAM2, KORO, KORO2,
     1         MFETK, MTHTK, MFETP,     LLI
C
      COMMON /PXCMN/     MM, LLN, I, IFLG, LN, IRSFG, IFGFG,
     1 BB1KJ, BB2KJ, BB3KJ, GG1KJ, GG2KJ, GG3KJ, DD1KJ, DD2KJ, DD3KJ,
     2 GG13KJ, TFETK, TTHTK, MFETK, MTHTK, BB1P, BB3P, GG1P, GG2P,
     3 DD1P, DD3P, GG13P, TFETP, MFETP, OMG(5,5), ALM(5,5), LLI(5),
     4 WTH, WFE, GAM, RHO, WFEP, LAM, LAM2, LAMO2, LAM2O2, LAM2O4,
     5 LAM2O8, GAM2, WTMWF, WTMWF2, WFWT, PTHK, PNK, PFEK, P, X,
     6 DCM1KJ, DCM2KJ, DCM3KJ, DCM4KJ, DCM5KJ, DCC1KJ, DCC2KJ, DCC3KJ,
     7 DCK1KJ, DCK2KJ, DCK3KJ
      COMMON /BLOCK1/  EMM2 (50,50)
      COMMON /BLOCK2/  EMM3 (50)
C
      IF(PFLAG .EQ. -1.E+10)  WRITE (6,340)
C                                          READ GEOMETRY PARAMETERS (100)
      READ  ( 3) WTHX, WFEX, GAMA, RHOX, WFEPX
      N2 = 2 * NNF
C
      IN = 1
      MMN = 0
      IFLG = 0
C
      IF (TCTR .EQ. 1.0) GO TO 8
      READ ( 9) PFE, PTH, PN
C                                                READ PRESSURES (10, 50)
      REWIND  9
      AF = FILE (9,1)
      GO TO 15
C
C                                     READ PRESSURES (10,50) TIME CYCLE 1
    8 READ ( 3) PFE, PTH, PN
C
C                                          READ STIFFNESS COEFF (100)
C                                       IN - INITIAL STATION IN BLOCK
C                                       LN - LAST STATION IN BLOCK
   10 READ  ( 9) LN, IN, COEFC
C
   15 IRTE = 1
      MM = IN - 1
      IFLG = IFLG + 1
      LLN = LN - IN + 1
```

```
      I = 0                                                             00001110
      KK = NNF                                                          00001120
      DEL2 = 2. * DEL                                                   00001130
C                                                                       00001140
   20 I = I + 1                                                         00001150
      MM = MM + 1                                                       00001160
      MMN = MMN + 1                                                     00001170
      IF (TCTR .GT. 1.0 .AND. MM .EQ. 1) GO TO 20                       00001180
      IF (TCTR .GT. 1.0 .AND. MM .EQ. N)  RETURN                        00001190
      JPATH = 0                                                         00001200
      KPATH = 0                                                         00001210
      IF (MM .NE. 51)  GO TO 23                                         00001230
      DO  19      IX = 1,KK                                             00001232
      PFE50(IX)   =  PFE(IX,50)                                         00001234
      PTH50(IX)   =  PTH(IX,50)                                         00001236
   19 PN50(IX)    =  PN(IX,50)                                          00001238
      IF (TCTR .EQ. 1.0) GO TO 21                                       00001240
      AF = FILE(9,2)                                                    00001250
      READ ( 9) PFE, PTH, PN                                            00001260
      READ ( 9) PFE, PTH, PN                                            00001270
      REWIND  9                                                         00001280
      AF = FILE(9,2)                                                    00001290
      GO TO 22                                                          00001300
   21 READ ( 3) PFE, PTH, PN                                            00001310
   22 MMN = 1                                                           00001320
C                                                                       00001329
   23 DO 9998 J = 1,KK                                                  00001330
      IF (TCTR .GT. 1.0) GO TO 37                                       00001340
      IF (JPATH .EQ. 1) GO TO 37                                        00001350
      IF (KPATH .EQ. 1) GO TO 37                                        00001360
C                                                                       00001365
C                ESTABLISH BOUNDARY CONDITIONS ARRAY.                   00001366
C                                                                       00001367
      IF (MM .NE. 1) GO TO 33                                           00001370
      IF (J .EQ. 1) GO TO 31                                            00001380
      IF (BCIT .NE. 1.0) GO TO 31                                       00001390
      IF (J .GT. 2) GO TO 29                                            00001400
C                                        CLOSED APEX, 2ND HARMONIC      00001409
   27 DO 28 KN = 1,5                                                    00001410
      DO 28 JJ = 1,5                                                    00001420
      OMG(KN,JJ) = OMG1(KN,JJ)                                          00001430
      ALM(KN,JJ) = ALM1(KN,JJ)                                          00001440
   28 LLI(KN) = 0.0                                                     00001450
```

```
      GO TO 37                                                          00001460
C                                       CLOSED APEX, .GT. 2ND HARMONIC  00001469
   29 DO 30 KN = 1,5                                                    00001470
      DO 30 JJ = 1,5                                                    00001480
      OMG(KN,JJ) = 0.0                                                  00001490
      ALM(KN,JJ) = 0.0                                                  00001500
      IF (KN .EQ. JJ) ALM(KN,JJ) = 1.0                                  00001510
   30 LLI(KN) = 0.0                                                     00001520
      GO TO 37                                                          00001530
C                                   1ST FOURIER COMPONENT, 1ST STA.     00001539
   31 DO 32 KN = 1,5                                                    00001540
      DO 32 JJ = 1,5                                                    00001550
      OMG(KN,JJ) = OMGO(KN,JJ)                                          00001560
      ALM(KN,JJ) = ALMO(KN,JJ)                                          00001570
   32 LLI(KN) = LLO(KN)                                                 00001580
      GO TO 37                                                          00001590
C                                                                       00001599
   33 IF (MM .NE. N) GO TO 37                                           00001600
C                                                   LAST STATION        00001609
      IF (J .EQ. 1) GO TO 34                                            00001610
      IF (BCIB .NE. 1.0) GO TO 34                                       00001620
      IF (J .GT. 2) GO TO 29                                            00001630
      GO TO 27                                                          00001640
C                                   1ST FOURIER COMPONENT, NTH STA.     00001649
   34 DO 35 KN = 1,5                                                    00001650
      DO 35 JJ = 1,5                                                    00001660
      OMG(KN,JJ) = OMGN(KN,JJ)                                          00001670
      ALM(KN,JJ) = ALMN(KN,JJ)                                          00001680
   35 LLI(KN) = LLLN(KN)                                                00001690
C                                                                       00001699
   37 DO 9998  K = 1,KK                                                 00001700
      IF (TCTR .GT. 1.0) GO TO 152                                      00001710
      IF (KPATH .EQ. 1) GO TO 152                                       00001720
      IF (I .NE. 1) GO TO 85                                            00001730
      IF (IFLG .NE. 1) GO TO 98                                         00001740
C                                                                       00001750
C                RECURSION FORMS        *          *          *        00001751
C                                                                       00001752
   38 JMK = J - K                                                       00001760
      KMJA = 1 + IABS(K - J)                                            00001770
      KPJ = K + J - 1                                                   00001780
      TERM1 = 1.0                                                       00001790
      IF (JMK .EQ. 0) TERM1 = 0.0                                       00001800
```

```
      TERM2 = 1.0                                                       00001810
      IF (K .EQ. 1) TERM2 = 0.0                                         00001820
      FCTR = 1. - TERM1 + TERM2                                         00001830
      FCTR2 = 1. - 2. * TERM2                                           00001840
C                                                                       00001850
      K2 = N2 * (I-1) + KPJ                                             00001860
      K5 = N2 * (I-1) + KMJA                                            00001870
      IF (JPATH .EQ. 1) GO TO 50                                        00001880
      K1 = N2 * I + KPJ                                                 00001890
      K3 = N2 * (I+1) + KPJ                                             00001900
      K4 = N2 * I + KMJA                                                00001910
      K6 = N2 * (I+1) + KMJA                                            00001920
      K7 = N2 * (I-2) + KPJ                                             00001930
      K8 = N2 * (I-2) + KMJA                                            00001940
      K9 = N2 * (I-3) + KPJ                                             00001950
      K10 = N2 * (I-3) + KMJA                                           00001960
C                                                                       00001970
      IF (MM .NE. 1) GO TO 55                                           00001980
C                                                                       00001988
C                 FORWARD DIFFERENCE FOR FIRST STATION.                 00001989
      IF(J .NE. 1) GO TO 39                                             00001990
      TTP(K) = (4. * SCNTSI(K1) - 3. * SCNTSI(K2) - SCNTSI(K3)) /DEL2   00001991
      MTP(K) = (4. * SCMTSI(K1) - 3. * SCMTSI(K2) - SCMTSI(K3)) /DEL2   00001992
      TFE(K)  =  SCNTSI(K2)                                             00001993
      MFE(K)  =  SCMTSI(K2)                                             00001994
C                                                                       00001995
   39 BP1KPJ = (4. * SCB1(K1) - 3. * SCB1(K2) - SCB1(K3)) /DEL2         00002000
      DP1KPJ = (4. * SCD1(K1) - 3. * SCD1(K2) - SCD1(K3)) /DEL2         00002010
      G13KPJ = (4. * SCG13(K1) - 3. * SCG13(K2) - SCG13(K3)) /DEL2      00002020
      GP1KPJ = (4. * SCG1(K1) - 3. * SCG1(K2) - SCG1(K3)) /DEL2         00002040
C                                                                       00002050
      BP1KMJ = (4. * SCB1(K4) - 3. * SCB1(K5) - SCB1(K6)) /DEL2         00002060
      DP1KMJ = (4. * SCD1(K4) - 3. * SCD1(K5) - SCD1(K6)) /DEL2         00002080
      G13KMJ = (4. * SCG13(K4) - 3. * SCG13(K5) - SCG13(K6)) /DEL2      00002090
      GP1KMJ = (4. * SCG1(K4) - 3. * SCG1(K5) - SCG1(K6)) /DEL2         00002110
C                                                                       00002120
C                 DERIVATIVE OF STIFFNESS RECURSIONS.                   00002121
C                                                                       00002122
   40 BB1P      =  0.5 *  (BP1KPJ + FCTR * BP1KMJ)                      00002130
      DD1P    =    0.5 * (DP1KPJ + FCTR * DP1KMJ)                      00002140
      GG13P = 0.5 * (FCTR * G13KMJ + FCTR2 * G13KPJ)                    00002150
      GG1P = 0.5 * (FCTR * GP1KMJ + FCTR2 * GP1KPJ)                     00002160
C                                                                       00002170
```

```
C                 STIFFNESS RECURSION    *          *          *
C
   50 BB1KJ  = 0.5 * (SCB1(K2) + FCTR * SCB1(K5))
      DD1KJ  = 0.5 * (SCD1(K2) + FCTR * SCD1(K5))
      DCM1KJ = 0.5 * ( DCM1(K2) + FCTR * DCM1(K5))
      DCM2KJ = 0.5 * ( FCTR * DCM1(K5) + FCTR2 * DCM1(K2))
      DCM4KJ = 0.5 * ( DCM4(K2) + FCTR * DCM4(K5))
      DCM5KJ = 0.5 * ( FCTR * DCM4(K5) + FCTR2 * DCM4(K2))
      DCK1KJ = 0.5 * (DCK1(K2)+ FCTR * DCK1(K5))
      DCK2KJ = 0.5 * ( FCTR * DCK2(K5) + FCTR2 * DCK2(K2))
      DCK3KJ = 0.5 * (DCK3(K2) + FCTR * DCK3(K5))
      GG13KJ = 0.5 * (FCTR * SCG13(K5) + FCTR2 * SCG13(K2))
      GG1KJ = 0.5 * (FCTR * SCG1(K5) + FCTR2 * SCG1(K2))
      GO TO 150
C
   55 IF (MM .EQ. N) GO TO 65
      IF(J .NE. 1) GO TO 57
      TTP(K) = (SCNTSI(K1) - SCNTSI(K7)) /DEL2
      MTP(K) = (SCMTSI(K1) - SCMTSI(K7)) /DEL2
      TFE(K)  =  SCNTSI(K2)
      MFE(K)  =  SCMTSI(K2)
   57 BP1KPJ = (SCB1(K1) - SCB1(K7)) /DEL2
      DP1KPJ = (SCD1(K1) - SCD1(K7)) /DEL2
      G13KPJ = (SCG13(K1) - SCG13(K7)) /DEL2
      GP1KPJ = (SCG1(K1) - SCG1(K7)) /DEL2
C
   60 BP1KMJ = (SCB1(K4) - SCB1(K8)) /DEL2
      DP1KMJ = (SCD1(K4) - SCD1(K8)) /DEL2
      G13KMJ = (SCG13(K4) - SCG13(K8)) /DEL2
      GP1KMJ = (SCG1(K4) - SCG1(K8)) /DEL2
      GO TO 40
C
C                 BACKWARD DIFFERENCE FOR LAST STATION.      *
   65 IF(J .NE. 1) GO TO 70
      TTP(K) = (SCNTSI(K9) + 3. * SCNTSI(K2) - 4. * SCNTSI(K7)) /DEL2
      MTP(K) = (SCMTSI(K9) + 3. * SCMTSI(K2) - 4. * SCMTSI(K7)) /DEL2
      TFE(K)  =  SCNTSI(K2)
      MFE(K)  =  SCMTSI(K2)
C
   70 BP1KPJ = (SCB1(K9) + 3. * SCB1(K2) - 4. * SCB1(K7)) /DEL2
      DP1KPJ = (SCD1(K9) + 3. * SCD1(K2) - 4. * SCD1(K7)) /DEL2
      G13KPJ = (SCG13(K9) + 3. * SCG13(K2) - 4. * SCG13(K7)) /DEL2
      GP1KPJ = (SCG1(K9) + 3. * SCG1(K2) - 4. * SCG1(K7)) /DEL2
```

```
C
      BP1KMJ = (SCB1(K10) + 3. * SCB1(K5) - 4. * SCB1(K8)) /DEL2
      DP1KMJ = (SCD1(K10) + 3. * SCD1(K5) - 4. * SCD1(K8)) /DEL2
      G13KMJ = (SCG13(K10) + 3. * SCG13(K5) - 4.*SCG13(K8)) /DEL2
      GP1KMJ = (SCG1(K10) + 3. * SCG1(K5) - 4. * SCG1(K8)) /DEL2
      GO TO 40
C
C
   85 IF (LN .EQ. N) GO TO 38
      IF (I .NE. 3) GO TO 93
C                                          AT ARBITRARY STATION 3
C                 SAVE COEFF. OF LAST TWO STAS. OF THIS BLOCK.  TO BE
C                 USED TO COMPUTE DERIVATIVES ON NEXT BLOCK.  *
C
      KK2 = 2 * KK
      K11 = N2 * (LLN-2)
      K12 = N2 * (LLN-1)
      DO 90 IK = 1, KK2
      K13 = K11 + IK
      K14 = K12 + IK
      P1SCB1(IK) = SCB1(K13)
      P2SCB1(IK) = SCB1(K14)
      P1SCD1(IK) = SCD1(K13)
      P2SCD1(IK) = SCD1(K14)
      P1SG13(IK) = SCG13(K13)
      P2SG13(IK) = SCG13(K14)
      P1SCG1(IK) = SCG1(K13)
      P2SCG1(IK) = SCG1(K14)
      P2DCM1(IK) = DCM1(K14)
      P2DCM4(IK) = DCM4(K14)
      P2DCK1(IK) = DCK1(K14)
      P2DCK2(IK) = DCK2(K14)
      P2DCK3(IK) = DCK3(K14)
      P1ENT(IK) = SCNTSI(K13)
      P2ENT(IK) = SCNTSI(K14)
      P1EMT(IK) = SCMTSI(K13)
      P2EMT(IK) = SCMTSI(K14)
   90 CONTINUE
C                                          IS COEFF BLOCK COMPLETED
   93 IF (I .EQ. LLN) GO TO 10
C                                                   NO
      GO TO 38
C
```

```
C
   98 JMK   = J - K
      KMJA = 1 + IABS(K - J)
      KPJ = K + J - 1
      TERM1 = 1.0
      IF (JMK .EQ. 0) TERM1 = 0.0
      TERM2 = 1.0
      IF (K .EQ. 1) TERM2 = 0.0
      FCTR = 1. - TERM1 + TERM2
      FCTR2 = 1. - 2. * TERM2
C
      K1 = N2 * I + KPJ
      K2 = N2 * (I-1) + KPJ
      K4 = N2 * I + KMJA
      K5 = N2 * (I-1) + KMJA
C
      IF (JPATH .EQ. 1) GO TO 125
C
C         IRTE .NE. 1  - FIRST STATION IN A SECTION (NOT REGION).
C         IRTE .EQ. 1  - LAST STATION IN PREVIOUS SECTION.
C
  100 IF (IRTE .NE. 1) GO TO 110
C                LAST STATION - PREVIOUS BLOCK
      IF (K .NE. 1 .OR. J .NE. 1) GO TO 102
      MM = MM - 1
      MMN = MMN - 1
  102 IF(J .NE. 1) GO TO 103
      TTP(K) = (SCNTSI(K2) - P1ENT(KPJ)) /DEL2
      MTP(K) = (SCMTSI(K2) - P1EMT(KPJ)) /DEL2
      TFE(K)  =  P2ENT(KPJ)
      MFE(K)  =  P2EMT(KPJ)
  103 BP1KPJ = (SCB1(K2) - P1SCB1(KPJ)) /DEL2
      DP1KPJ = (SCD1(K2) - P1SCD1(KPJ)) /DEL2
      G13KPJ = (SCG13(K2) - P1SG13(KPJ)) /DEL2
      GP1KPJ = (SCG1(K2) - P1SCG1(KPJ)) /DEL2
C
  105 BP1KMJ = (SCB1(K5) - P2SCB1(KMJA)) /DEL2
      DP1KMJ = (SCD1(K5) - P2SCD1(KMJA)) /DEL2
      G13KMJ = (SCG13(K5) - P2SG13(KMJA)) /DFL2
      GP1KMJ = (SCG1(K5) - P2SCG1(KMJA)) /DEL2
      GO TO 120
C                FIRST STATION - NOT FIRST BLOCK
  110 IF (K .NE. 1 .OR. J .NE. 1) GO TO 111
```

```
      MM = MM + 1                                                       00003210
      MMN = MMN + 1                                                     00003220
  111 IF(J .NE. 1) GO TO 112                                            00003221
      TTP(K) = (SCNTSI(K1) - P2ENT(KPJ)) /DEL2                          00003222
      MTP(K) = (SCMTSI(K1) - P2EMT(KPJ)) /DEL2                          00003223
      TFE(K)  =  SCNTSI(K2)                                             00003224
      MFE(K)  =  SCNTSI(K2)                                             00003225
  112 BP1KPJ = (SCB1(K1) - P2SCB1(KPJ)) /DEL2                           00003230
      DP1KPJ = (SCD1(K1) - P2SCD1(KPJ)) /DEL2                           00003240
      G13KPJ = (SCG13(K1) - P2SG13(KPJ)) /DEL2                          00003250
      GP1KPJ = (SCG1(K1) - P2SCG1(KPJ)) /DEL2                           00003260
C                                                                       00003270
      BP1KMJ = (SCB1(K4) - P2SCB1(KMJA)) /DEL2                          00003280
      DP1KMJ = (SCD1(K4) - P2SCD1(KMJA)) /DEL2                          00003290
      G13KMJ = (SCG13(K4) - P2SG13(KMJA)) /DEL2                         00003300
      GP1KMJ = (SCG1(K4) - P2SCG1(KMJA)) /DEL2                          00003310
C                                                                       00003320
  120 BB1P    =  0.5   * (BP1KPJ +  FCTR *  BP1KMJ)                     00003330
      DD1P      = 0.5 * (DP1KPJ + FCTR * DP1KMJ)                        00003340
      GG13P = 0.5 * (FCTR * G13KMJ + FCTR2 * G13KPJ)                    00003350
      GG1P = 0.5 * (FCTR * GP1KMJ + FCTR2 * GP1KPJ)                     00003360
C                                                                       00003370
  125 IF (IRTE .NE. 1) GO TO 135                                        00003380
      TERM11 = P2SCB1(KPJ)                                              00003390
      TERM21 = P2SCD1(KPJ)                                              00003400
  130 BB1KJ    = 0.5 * (TERM11 + FCTR * P2SCB1(KMJA))                   00003410
      DD1KJ    = 0.5 * (TERM21 + FCTR * P2SCD1(KMJA))                   00003420
      DCM1KJ = 0.5 * (P2DCM1(KPJ) + FCTR * P2DCM1(KMJA))                00003430
      DCM2KJ = 0.5 * (FCTR * P2DCM1(KMJA) + FCTR2 * P2DCM1(KPJ))        00003440
      DCM4KJ = 0.5 * (P2DCM4(KPJ) + FCTR * P2DCM4(KMJA))                00003450
      DCM5KJ = 0.5 * (FCTR * P2DCM4(KMJA) + FCTR2 * P2DCM4(KPJ))        00003460
      DCK1KJ = 0.5 * (P2DCK1(KPJ) + FCTR * P2DCK1(KMJA))                00003470
      DCK2KJ = 0.5 * (FCTR * P2DCK2(KMJA)+ FCTR2 * P2DCK2(KPJ))         00003480
      DCK3KJ = 0.5 * (P2DCK3(KPJ) + FCTR * P2DCK3(KMJA))                00003490
      GG13KJ = 0.5 * (FCTR * P2SG13(KMJA) + FCTR2 * P2SG13(KPJ))        00003500
      GG1KJ = 0.5 * (FCTR * P2SCG1(KMJA) + FCTR2 * P2SCG1(KPJ))         00003510
      GO TO 150                                                         00003520
C                                                                       00003530
  135 BB1KJ  = 0.5 * (SCB1(K2) + FCTR * SCB1(K5))                       00003540
      DD1KJ  = 0.5 * (SCD1(K2) + FCTR * SCD1(K5))                       00003550
      DCM1KJ = 0.5 * ( DCM1(K2) + FCTR * DCM1(K5))                      00003560
      DCM2KJ = 0.5 * ( FCTR * DCM1(K5) + FCTR2 * DCM1(K2))              00003570
      DCM4KJ = 0.5 * ( DCM4(K2) + FCTR * DCM4(K5))                      00003580
```

```
      DCM5KJ = 0.5 * ( FCTR * DCM4(K5) + FCTR2 * DCM4(K2))
      DCK1KJ = 0.5 * ( DCK1(K2) + FCTR * DCK1(K5))
      DCK2KJ = 0.5 * ( FCTR * DCK2(K5) + FCTR2 * DCK2(K2))
      DCK3KJ = 0.5 * (DCK3(K2) + FCTR * DCK3(K5))
      GG13KJ = 0.5 * (FCTR * SCG13(K5) + FCTR2 * SCG13(K2))
      GG1KJ = 0.5 * (FCTR * SCG1(K5) + FCTR2 * SCG1(K2))
C
  150 BB2KJ   =   BB1KJ
      BB3KJ  =  POI  *  BB1KJ
      SS2  =  (1. - POI) / 2.
      GG2KJ = SS2 * BB1KJ
      GG3KJ  =  GG1KJ
      BB3P   =   POI  *  BB1P
      GG3P  = GG1P
      GG2P  = SS2 * BB1P
      DD2KJ     =   DD1KJ
      DD3KJ  =  POI   *  DD1KJ
      DD3P  =  POI * DD1P
  152 TFETK  =  TFE(K)
      TTHTK  =  TFE(K)
      MFETK  =  MFE(K)
      MTHTK  =  MFE(K)
      TFETP  =  TTP(K)
      MFETP  =  MTP(K)
      IF(MMN .EQ. 0) GO TO 153
      PFEK  = PFE(K,MMN)
      PTHK = PTH(K,MMN)
      PNK = PN(K,MMN)
      GO  TO  154
C
  153 PFEK  =  PFE50(K)
      PTHK  =  PTH50(K)
      PNK   =  PN50(K)
C
C            P AND X FORMULATION
C
C
  150 WTH = WTHX(MM)
      WFE = WFEX(MM)
      GAM = GAMA(MM)
      RHO = RHOX(MM)
      WFEP = WFEPX(MM)
      LAM  =  HO /AO
```

```
      LAM2  =  LAM * LAM                                                00003970
      IF (TCTR .GT. 1.0) GO TO 450                                      00003980
      IF (KPATH .EQ. 1) GO TO 450                                       00003990
      LAMO2  =  LAM /2.                                                 00004000
      LAM2O2  =  LAM * LAMO2                                            00004010
      LAM2O4  =  LAMO2 * LAMO2                                          00004020
      LAM2O8  =  LAM2O4 /2.                                             00004030
      GAM2  =  GAM **2                                                  00004050
      WTMWF  =  WTH - WFE                                               00004060
      WTMWF2  =  WTMWF **2                                              00004070
      WFWT  =  WFE * WTH                                                00004080
C                                                                       00004090
      IF (JPATH .EQ. 1) GO TO 400                                       00004100
      IF (IFLG .NE. 1) GO TO 235                                        00004110
      IF (I .NE. 1) GO TO 235                                           00004120
C                                                                       00004130
C                                          SET UP BOUNDARY MATRICES     00004139
  155 CALL RSLT (J,K,KK)                                                00004140
      IF (IRSFG .GT. 1)  GO TO 9998                                     00004150
      IF (IRSFG .EQ. 1)  GO TO 20                                       00004160
      RETURN                                                            00004170
C                                                                       00004180
  235 IF (I .NE. LLN)  GO TO 237                                        00004190
      IF (LN .EQ. N) GO TO 155                                          00004200
C                                                                       00004208
C                                     SET UP MATRICES IN EQUILIBRIUM EQ. 00004209
  237 CALL FGHPE (J,K,KK,IRTE)                                          00004210
      IF (IFGFG .GT. 1) GO TO 9998                                      00004220
      JPATH = 1                                                         00004230
      GO TO 23                                                          00004240
C                                                                       00004268
C                                   SET UP L, M, N MATRICES USED IN G** 00004269
  400 CALL DYLMN (J,K,KK,IFDYN)                                         00004270
      IF (IFDYN .GT. 1) GO TO 9998                                      00004280
      KPATH = 1                                                         00004290
      GO TO 23                                                          00004300
C                                                                       00004308
C                                       FORM G**, D**, P AND X MATRICES 00004309
  450 CALL GMTX(J,K,KK,IFMX)                                            00004310
      IF (IFMX .GT. 1) GO TO 9998                                       00004320
      IF (TCTR .GT. 1.0) GO TO 20                                       00004330
      IF (IFGFG .EQ. 1) GO TO 20                                        00004340
      IRTE = 0                                                          00004350
      JPATH = 0                                                         00004360
      KPATH = 0                                                         00004370
      GO TO 23                                                          00004380
 9998 CONTINUE                                                          00004390
C                                                                       00004400
  340 FORMAT(1H-,14HENTERED PANDX. )                                    00004410
      RETURN                                                            00004420
      END                                                               00004430
```

```
$IBFTC BNDD
C
      SUBROUTINE RSLT (J,K,KK)
C
C        COMPUTES THE BOUNDARY MATRICES FOR 'P AND X' MATRICES.
C
      DIMENSION    R(50,50), AA0(50,50), S(50,50), LT(50)
      DIMENSION  P(50,50), X(50), ZX(50)
C
      COMMON  GDA(25), DFL, N, NNF, NTH,
     1 BMTX(160), TCTR, KZTR, KZIR, KPTR, KPTR
C
      COMMON /PXCMN/      MN, LLN, I, IFLG, LN, IRFG, IFGFG,
     1 BB1KJ, BB2KJ, BB3KJ, GG1KJ, GG2KJ, GG3KJ, DD1KJ, DD2KJ, DD3KJ,
     2 GG13KJ, TFFIK, TTHIK, MFEIK, MTHIK, BB1P, BB3P, GG1P, GG2P,
     3 DD1P, DD3P, GG13P, TFEIP, MFEIP, OMG(5,5), ALM(5,5), LLI(5),
     4 WTH, WFF, GAM, RHO, WFFP, LAM, LAM2, LAM02, LAM202, LAM204,
     5 LAM208, GAM2, WTMWF, WTMWF2, WFWI, PIHK, PNK, PFFK, P, X,
     6 DCM1KJ, DCM2KJ, DCM3KJ, DCM4KJ, DCM5KJ, DCC1KJ, DCC2KJ, DCC3KJ,
     7 DCK1KJ, DCK2KJ, DCK3KJ
C
      COMMON /BLOCK1/  EMM2 (50,50)
      COMMON /BLOCK2/  EMM3 (50)
C
      EQUIVALENCE (GDA( 1),   A0),              (GDA( 2),   H0),
     1            (GDA( 3),   F0),              (GDA( 4), SIGO),
     2            (GDA( 5),  POI),              (GDA( 6),  ENF),
     3            (GDA( 7), BCIT),              (GDA( 8), BCIB),
     4            (GDA( 9),PFLAG),              (GDA(10), CFXT),
     5            (GDA(11), DFLT),              (GDA(12),  THT),
     6            (GDA(18),FPRNT)
C
      REAL   LAM, LAM02, LAM202, LAM204, LAM208, LAM2, KORO, KORO2,
     1          MFEIK, MTHIK, MFEIP, LT, LLI
C     R, S, LT  MATRICES IN BOUNDARY EQUATIONS  (44, 45) GID 65-1633
C
      KK5  =  5 * NNF
      IF (K .NE. 1)  GO TO 164
      IF (J .NE. 1)  GO TO 164
C                                                ZERO MATRIX ARRAYS
      DO 163 KN = 1,KK5
      LT(KN) = 0.0
      DO 162 JJ = 1,KK5
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      S(KN,JJ)  = 0.0
      R(KN,JJ)  = 0.0
  162 CONTINUE
  163 CONTINUE
C
  164 KORO = FLOAT(K-1) / RHO
      KORO2  =  KORO * KORO
      R(5*K-4,5*J-4)= BB1KJ * OMG(1,1)
      R(5*K-3,5*J-3)= GG1KJ /2. + LAM2OB * WTMWF2 * GG13KJ
      R(5*K-3,5*J) =  LAM2O4 * WTMWF * GG13KJ
      R(5*K-2,5*J-2)= GG2KJ * OMG(3,3)
      R(5*K-1,5*J-1)=  DD1KJ
      R(5*K, 5*J-3) =  ((GG13KJ * WTMWF /2.) * 0.5) * OMG(5,5)
      R(5*K,5*J)=  GG13KJ  *  0.5 * OMG(5,5)
      R(5*K-4,5*J-3)  =  R(5*K-3,5*J-3)  *  OMG(1,2)
      R(5*K-4,5*J)  =  R(5*K-3,5*J)  *  OMG(1,2)
      R(5*K,5*J-1)  =  R(5*K-1,5*J-1) * OMG(5,4)
C
      S(5*K-4,5*J-4)=  GAM *  BB3KJ
      S(5*K-4,5*J-3)=  KORO  *  BB3KJ
      S(5*K-4,5*J-2)=  WFE  *  BB1KJ  +  WTH  *  BB3KJ
      S(5*K-3,5*J-4)=  LAM2O8  *  WTMWF2  *  GG13KJ  -  GG1KJ /2.
      S(5*K-3,5*J-3) = GAM * S(5*K-3,5*J-4)
      S(5*K-3,5*J)  =  -GAM * R(5*K-3,5*J)
      S(5*K-3,5*J-4)  =  KORO * S(5*K-3,5*J-4)
      S(5*K-3,5*J-1)  =  -KORO * R(5*K-3,5*J)
      S(5*K-2,5*J-4)= - WFE  *  GG2KJ * OMG(3,3)
      S(5*K-2,5*J-1)=  GG2KJ * OMG(3,3)
      S(5*K-1,5*J-1)=  GAM  *  DD3KJ
      S(5*K-1,5*J)=  KORO  *  DD3KJ
      S(5*K, 5*J-4)  =  GG13KJ * WTMWF  /4.
      S(5*K,5*J-3) = GAM * S(5*K, 5*J-4)
      S(5*K,5*J-4)  =  KORO * S(5*K,5*J-4)
      S(5*K,5*J-1)=  - KORO  *  GG13KJ  /2.
      S(5*K,5*J)=  - GAM  *  GG13KJ /2.
      R(5*K-3,5*J-3) =  R(5*K-3,5*J-3) * OMG(2,2)
      R(5*K-3,5*J) =  R(5*K-3,5*J)  * OMG(2,2)
      R(5*K-1,5*J-1) =  R(5*K-1,5*J-1) * OMG(4,4)
C
      S(5*K-4,5*J-4) = S(5*K-4,5*J-4) * OMG(1,1) + S(5*K-3,5*J-4)  *
     1 OMG(1,2)
      S(5*K-4,5*J-3) = S(5*K-4,5*J-3) * OMG(1,1) + S(5*K-3,5*J-3) *
     1OMG(1,2)
```

```
      S(5*K-4,5*J-2) = S(5*K-4,5*J-2) * OMG(1,1)
      S(5*K-4,5*J-1) =  S(5*K-3,5*J-1) * OMG(1,2)
      S(5*K-4, 5*J)  =  S(5*K-3, 5*J) * OMG(1,2)
      S(5*K-3,5*J-3)=S(5*K-2,5*J-3) * OMG(2,2)
      S(5*K-3,5*J) = S(5*K-3,5*J) * OMG(2,2)
      S(5*K-3,5*J-4) = S(5*K-3,5*J-4) * OMG(2,2)
      S(5*K-3,5*J-1) = S(5*K-3,5*J-1) * OMG(2,2)
      S(5*K,5*J-4) = S(5*K,5*J-4) * OMG (5,5)
      S(5*K,5*J-3) = S(5*K,5*J-3) * OMG(5,5)
      S(5*K,5*J-1) = S(5*K,5*J-1) * OMG(5,5) + S(5*K-1,5*J-1)  *
     1 OMG(5,4)
      S(5*K-1,5*J-1) = S(5*K-1,5*J-1) * OMG(4,4)
      S(5*K,5*J)  = S(5*K,5*J) * OMG(5,5) + S(5*K-1,5*J) * OMG(5,4)
      S(5*K-1,5*J) = S(5*K-1,5*J) * OMG(4,4)
C
      LT(5*K-4)    =   IFE1K   *  OMG(1,1) + LLI(1)
      LT(5*K-3)   =  LLI(2)
      LT(5*K-2) = LLI(3)
      LT(5*K-1)   =    MFE1K    *  OMG(4,4) + LLI(4)
      LT(5*K)  = LLI(5)
C
      IF (K .NE. J) GO TO 166
      S(5*K-2,5*J-2) = ALM(3,3)
      S(5*K-4,5*J-4) = S(5*K-4,5*J-4) + ALM(1,1)
      S(5*K-3,5*J-3) = S(5*K-3,5*J-3) + ALM(2,2)
      S(5*K-3,5*J-4) = S(5*K-3,5*J-4) + ALM(2,1)
      S(5*K-1,5*J-1) = S(5*K-1,5*J-1) + ALM(4,4)
      S(5*K-1,5*J) = S(5*K-1,5*J) + ALM(4,5)
      S(5*K,5*J) = S(5*K,5*J) + ALM(5,5)
C
  166 IF (K .NE. KK) GO TO 167
      IF (J .EQ.KK)  GO TO 169
  167 IRSEG = 10
      GO  TO  180
C
  169 IF (IFLG .GT. 1) GO  TO 305
      IF (I .GT. 1) GO TO 305
C
C                                                INITIAL BOUNDARY
      DO 175 K = 1,KK5
      DO 175 J = 1,KK5
  175 R(K,J)  =  (2./LFL) * R(K,J)
C                                                SAVE  B0
```

```
      WRITE (11) R                                                      00001390
      DO 176 K = 1,KK5                                                  00001400
      DO 176 J = 1,KK5                                                  00001410
  176 AA0(K,J) =  -.25 * R(K,J)                                         00001420
C                                                   SAVE  A0            00001429
      WRITE (11) AA0.                                                   00001430
      DO 177 K = 1,KK5                                                  00001440
      DO 177 J = 1,KK5                                                  00001450
  177 FMM2(K,J) = 3. * AA0(K,J)                                         00001460
      CALL MAD (KK5,KK5, S, FMM2, S )                                   00001470
C                                                   SAVE  C0            00001479
      WRITE (11)  S                                                     00001480
C                                                   SAVE  G0            00001489
      WRITE (11) LI                                                     00001490
      END FILE 11                                                       00001500
      IRSEG = 1                                                         00001510
  180 IF(PFLAG .EQ. -1.E+10)  WRITE(6,310)                              00001520
      RETURN                                                            00001530
C                                                                       00001540
C                                                   TERMINAL BOUNDARY   00001550
  305 DO 306 K = 1,KK5                                                  00001560
      DO 306 J = 1,KK5                                                  00001570
  306 R(K,J) =  (3./(2.*DEL))  * R(K,J)                                 00001580
      CALL MAD (KK5,KK5,  S, R, AA0 )                                   00001590
      DO 307 K = 1,KK5                                                  00001600
      DO 307 J = 1,KK5                                                  00001610
      R(K,J) =  R(K,J) /3.                                              00001620
  307 S(K,J) =  -4. * R(K,J)                                            00001630
C                                                                       00001640
C                                   SAVE CN, BN, AN, GN  (IN THAT ORDER) 00001649
      WRITE (10) R, S, AA0, LI                                          00001650
      BACKSPACE 10                                                      00001660
C                                                                       00001670
      IRSEG = 0                                                         00001680
      REWIND  13                                                        00001690
      END FILE  3                                                       00001700
      END FILE  9                                                       00001710
      REWIND  11                                                        00001720
C                                   MOVE B0, A0, C0, G0 FROM TP 11 TO 3 00001728
C                                           BY SEPARATE RECORDS         00001729
      READ (11) P                                                       00001730
      WRITE  ( 3) P                                                     00001740
      READ  (11)  P                                                     00001750
```

```
      WRITE ( 3) P
      READ (11) P
      WRITE ( 3) P
      READ (11) X
      WRITE ( 3) X
      AF = FILE (11,1)
      END FILE 3
C                                   MOVE CN, BN, AN, GN FROM TAPE 10 TO 3
      READ (10) P, EMM2, AAO, X
      WRITE ( 3) P, EMM2, AAO, X
C                                   MOVE A1, B1, C1, G1 FROM TAPE 11 TO 3
      READ (11) P, EMM2, AAO, X
      WRITE ( 3) P, EMM2, AAO, X
      BACKSPACE  10
      BACKSPACE  10
      AF = FILE (11, 1)
C                                   MOVE P(N-2), X(N-2) - TP 11 TO 3
      READ (11) P, X
      WRITE ( 3) P, X
C
C                                   MOVE L, M, N - TP 13 TO 9 (2 FILES POSSIBLE)
C
      NM2 = (N - 2) * 3
      DO 170 I = 1,NM2
      IF (I .EQ. 147)  END FILE 9
      READ (13) P
      WRITE ( 9)  P
  170 CONTINUE
      END FILE  9
      GO  TO  180
  310 FORMAT (1H-,11HLEFT BNDRY. )
      END
```

```
$IBFTC FGHDY
C
      SUBROUTINE FGHPE (J,K,KK,IPTF)
C
      DIMENSION          S(5,5), F(5,5), G(5,5), H(5,5)
      DIMENSION  P(50,50), X(50)
C
      COMMON  GDA(25), DEL, N, NNF, NTH,
     1 BMTX(160), ICTR, K2IR, K2IR, KPIR, KPTR
C
      COMMON /PXCMN/      MM, LLN, I, IFLG, LN, IPSEG, IFGEG,
     1 BB1KJ, BB2KJ, BB3KJ, GG1KJ, GG2KJ, GG3KJ, DD1KJ, DD2KJ, DD3KJ,
     2 GG13KJ, TEETK, THHK, MEETK, MTHTK, BB1P, BB3P, GG1P, GG2P,
     3 DD1P, DD3P, GG13P, TEETP, MEETP, OMG(5,5), ALM(5,5), LLI(5),
     4 WTH, WEE, GAM, RHO, WEEP, LAM, LAM2, LAMO2, LAM2O2, LAM2O4,
     5 LAM2OB, GAM2, WTMWE, WTMWE2, WEWT, PTHK, PNK, PEEK, P, X,
     6 DCM1KJ, DCM2KJ, DCM3KJ, DCM4KJ, DCM5KJ, DCC1KJ, DCC2KJ, DCC3KJ,
     7 DCK1KJ, DCK2KJ, DCK3KJ
C
      COMMON /BLOCK1/  EMM2 (50,50)
      COMMON /BLOCK2/  EMM3 (50)
C
      EQUIVALENCE (GDA( 1),    A0),              (GDA( 2),    H0),
     1            (GDA( 3),    E0),              (GDA( 4), SIG0),
     2            (GDA( 5),   P0I),              (GDA( 6),   ENF),
     3            (GDA( 7), BCIT),               (GDA( 8), BCIB),
     4            (GDA( 9),PFLAG),               (GDA(10), CEXT),
     5            (GDA(11), DELT),               (GDA(12),   THT),
     6            (GDA(18),FPRNT)
C
      REAL  LAM, LAMO2, LAM2O2, LAM2O4, LAM2OB, LAM2, KORO, KORO2,
     1          MEETK, MTHTK, MEETP, LT(50), LLI
C
C      F, G, H, PE  MATRICES IN EQUILIBRIUM EQUATIONS (33) SID 65-1633
C
      KK5  =  5 * NNF
      IF (K .NE. 1)  GO TO 164
      IF (J .NE. 1)  GO TO 164
C                                          ZERO MATRIX ARRAYS
      DO 163 KN = 1,KK5
      DO 162 JJ = 1,KK5
      F(KN,JJ)  =  0.0
      G(KN,JJ)  =  0.0
```

```
      H(KN,JJ)   =  0.0                                                 00000377
  162 CONTINUE                                                          00000380
  163 CONTINUE                                                          00000390
C                                                                       00000399
  164 KORO = FLOAT(K-1) /RHO                                            00000400
  168 KORO2 = KORO * KORO                                               00000410
C                                                                       00000419
C     F,G,H,PF   MATRICES IN EQUILIBRIUM EQUATIONS (33)  SID 65-1632    00000420
C                                                                       00000421
      DELT2 = DELT * DELT                                               00000430
      F(5*K-4,5*J-4)=  BB1KJ                                            00000440
      F(5*K-3,5*J-3) = GG1KJ/2. + LAM2O8 * WTMWF2* GG13KJ               00000450
      F(5*K-3,5*J) = LAM2O4 * WTMWF * GG13KJ                            00000460
      F(5*K-2,5*J-2)=  GG2KJ                                            00000470
      F(5*K-1,5*J-1)=  LAM2  *  DD1KJ                                   00000480
      F(5*K,5*J-3) = F(5*K-3,5*J)                                       00000490
      F(5*K,5*J)=  LAM2O2  *  GG13KJ                                    00000500
C                                                                       00000509
      G(5*K-4,5*J-4)=  BB1P  +  GAM  *  BB1KJ                           00000510
      G(5*K-4,5*J-3)=  KORO  * ( BB3KJ  +  GG1KJ  /2.  -  LAM2O8  *     00000520
     1 WTMWF2 *  GG13KJ  )                                              00000530
      G(5*K-4,5*J-2)=  WFE  * ( BB1KJ  +  GG2KJ  ) + WTH  *  BB3KJ      00000540
      G(5*K-4,5*J) = -KORO * LAM2O4 * WTMWF * GG13KJ                    00000550
      G(5*K-3,5*J-4) = - G(5*K-4,5*J-3)                                 00000560
      G(5*K-3,5*J-3) = GG1P/2. + LAM2O8 * (WTMWF2 * GG13P - GG13KJ      00000570
     2* ( WTMWF2  *  GAM  + 2. *  WFEP  *  WTMWF ) ) +  GG1KJ  * GAM /2.00000580
      G(5*K-3,5*J-1) = -KORO * LAM2O4 * WTMWF * GG13KJ                  00000590
      G(5*K-3,5*J) = LAM2O4 *                                           00000600
     1(  GG13P  *  WTMWF  -  GG13KJ  * (2. *  GAM  * WTMWF  + WFEP  ) ) 00000610
      G(5*K-2,5*J-4) = - G(5*K-4,5*J-2)                                 00000620
      G(5*K-2,5*J-2)=  GG2P  +  GAM  *  GG2KJ                           00000630
      G(5*K-2,5*J-1) = GG2KJ                                            00000640
      G(5*K-1,5*J-3) = - G(5*K-3,5*J-1)                                 00000650
  180 G(5*K-1,5*J-2) = - G(5*K-2,5*J-1)                                 00000660
      G(5*K-1,5*J-1)=  LAM2 * (  DD1P  +  GAM  *  DD1KJ  )              00000670
      G(5*K-1,5*J)=  KORO * ( LAM2  *  DD3KJ  +  LAM2O2  *  GG13KJ  )   00000680
      G(5*K,5*J-4) = - G(5*K-4,5*J)                                     00000690
      G(5*K,5*J-3) = LAM2O4 * (                                         00000700
     1 GG13P  * WTMWF +  GG13KJ  * ( 2. * GAM * WTMWF  - WFEP )  )      00000710
      G(5*K,5*J-1) = - G(5*K-1,5*J)                                     00000720
  183 G(5*K,5*J)= LAM2O2 * ( GG13P  +  GAM  *  GG13KJ  )                00000730
C                                                                       00000739
      H(5*K-4,5*J-4)=GAM * BB3P - WFWT * BB3KJ - GAM2 * BB2KJ - KORO2 * 00000740
```

```
     1(GG1KJ/2.0+ LAM2OB* WTMWF2*GG13KJ) -WFF**2 * GG2KJ - DCK1KJ
     2 2.0 * DCM1KJ / DELT2 - 11.0 * DCC1KJ / ( 6.0 * DELT )
      H(5*K-4,5*J-3)=  KORO * ( BB3P  -  GAM  *  BB2KJ  -  GAM /2. *
     1 GG1KJ - GAM * WTMWF2 * LAM2OB * GG13KJ )
      H(5*K-4,5*J-2)=  WFF *  BB1P  +  WTH * ( BB3P  -  GAM *  BB2KJ  )
     1 +  BB1KJ  * (  WEFP  +  GAM  *  WFF  )
      H(5*K-4,5*J-1) = WFF * GG2KJ + KORO2 * (LAM2O4 * WTMWF * GG13KJ )
      H(5*K-4,5*J) = KORO * GAM * LAM2O4 * WTMWF * GG13KJ
      H(5*K-3,5*J-4)= KORO * ( LAM2OB * (  WTMWF2 * GG13P  - WTMWF  *
     1  GG13KJ  * ( 3. * GAM * WTMWF -  2. *  WEFP ) ) - GG1P /2. + GAM
     2 *   (-GG1KJ/2. - BB2KJ ) )
      H(5*K-3,5*J-3)=(WFWT - GAM2)/2.* GG1KJ - GAM/2. * GG1P - KORO2*
     1 BB2KJ - WTH **2 * GG3KJ + LAM2OB *( GAM* WTMWF2 * GG13P -GG13KJ
     2 * (WTMWF2 * (WFWT + 2.*GAM2)+2.* WTMWF* GAM*WEFP)) -DCK2KJ
     3       - 2.0 * DCM2KJ / DELT2 - 11.0 * DCC2KJ / ( 6.0 * DELT )
  190 H(5*K-3,5*J-2)=- KORO * ( WFF *  BB3KJ + WTH * (  BB2KJ  + GG3KJ))
      H(5*K-3,5*J-1)= KORO * ( LAM2O4 * ( GAM * ( 2. * WTMWF  +  WEFP )*
     1 GG13KJ - WTMWF * GG13P) )
      H(5*K-3,5*J) = WTH * GG  J +  LAM2O4 * WTMWF *
     2 ( GG13KJ  * ( WFWT + 2. * GAM2 ) - GAM *  GG13P  ) + GAM * LAM2O4
     3  *  WEFP  *  GG13KJ
  193 H(5*K-2,5*J-4)=-  GG2KJ * ( WEFP  + GAM * WFF ) - WFF * ( GAM *
     1 BB3KJ  +  GG2P ) - WTH *  GAM  * BB2KJ
      H(5*K-2,5*J-3) = H(5*K-3,5*J-2)
      H(5*K-2,5*J-2)=-KORO2* GG3KJ - WFF**2* BB1KJ -WTH* (2.*WFF*
     1 BB3KJ + WTH * BB2KJ ) - DCK3KJ - 2.0 * DCM1KJ/ DELT2 - 11.0
     2 * DCC3KJ / ( 6.0 * DELT )
      H(5*K-2,5*J-1) = GG2P + GAM * GG2KJ
      H(5*K-2,5*J) = KORO * GG3KJ
      H(5*K-1,5*J-4) = H(5*K-4,5*J-1)
      H(5*K-1,5*J-3) = KORO * LAM2O4 * GAM * WTMWF * GG13KJ
  195 H(5*K-1,5*J-1)=LAM2 *(GAM *DD3P- WFWT* DD3KJ - KORO2/2.*GG13KJ -
     1GAM2* DD2KJ) - GG2KJ                -2.0 * DCM4KJ / DELT2
      H(5*K-1,5*J)= LAM2 * KORO * (  DD3P - GAM * ( GG13KJ  /2. +
     1  DD2KJ  )  )
      H(5*K,5*J-4)= KORO * ( LAM2O4 * ( WTMWF *  GG13P - WEFP *  GG13KJ
     1 ) )
      H(5*K,5*J-3) = LAM2O4 * (GAM * WTMWF * GG13P - GG13KJ * (GAM *WEFP
     1 + WFWT * WTMWF) ) + WTH * GG3KJ
      H(5*K,5*J-2) = H(5*K-2,5*J)
  198 H(5*K,5*J-1)=- KORO * ( LAM2O2 * ( GG13P + GAM *  GG13KJ ) + LAM2
     1 * GAM *  DD2KJ  )
      H(5*K  ,5*J ) = LAM2O2 * ((WFWT  - GAM2)*GG13KJ -GAM* GG13P)
```

```
     1 -FCR02 * LAM2* DD2KJ - GG2KJ                    -2. * DCK5KJ / DELT2
C                                                    BE FORMED IN GMTX
C
      IF (K .NE. KK) GO TO 203
      IF (J .EQ. KK)  GO TO 208
  203 IEGFG = 10
      GO  TO  251
C
C                                                  FORM A, B, C MATRICES
C
  208 DO 210 K=1,KK5
      DO 210 J = 1,KK5
      F(K,J) = (2./DELT) * F(K,J) * DELT
      G(K,J) = DELT * G(K,J)
  210 H(K,J) = DELT * H(K,J)
      CALL MAD (KK5,KK5, F, G, FMM2 )
      CALL MSUB(KK5,KK5, F, G,  G )
      DO 212 K = 1,KK5
      DO 212 J = 1,KK5
      H(K,J) =  2. * DELT * H(K,J)
  212 F(K,J) = - 2. * F(K,J)
      CALL MAD (KK5,KK5, F,  H,  H )
      DO 214 K = 1,KK5
      DO 214 J = 1,KK5
  214 F(K,J) = FMM2(K,J)
C
C
      IF (MM .NE. 2) GO TO 260
C
      CALL INVNNM (G, 50, KK5, IERR )
      IF (IERR .LE. 0) GO TO 320
C
C                        SAVE A1, B1, C1 - TO BE USED IN FORMING Z(0)
C
      WRITE (11) F, H, G
      BACKSPACE 11
      BACKSPACE 11
      WRITE (12) F, H, G
  250 IEGFG = 1
  251 IF(PFLAG .EQ. -1.E+10)  WRITE(6,252)
  252 FORMAT (1H-,11HLEFT EGMTX. )
      RETURN
C
```


REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
  260 WRITE (12) F, H, G
      IF (I .NE. 1) GO TO 250
      IF (IRTF .NE. 1) GO TO 250
      IFGFG = 0
      GO  TO  251
  320 WRITE (6,325)  I
  325 FORMAT (1H-,9X,16HSINGULAR MATRIX,,9X,4HI = ,I4)
      CALL EXIT
      STOP
      END
```


REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

3.12i DYLMN Subroutine (LMNDY)

```
$IBFTC LMNDY
C
C
      SUBROUTINE  DYLMN(J,K,KP,IFLYM)
C
C
      DIMENSION  AL(50,50), AM(50,50), ANM(50,50)
      DIMENSION  P(50,50), Y(50)
      EQUIVALENCE (GDA( 1),    A0),              (GDA( 2),    H0),
     1            (GDA( 3),    F0),              (GDA( 4),   FG0),
     2            (GDA( 5),  POT),               (GDA( 6),   TNF),
     3            (GDA( 7), BCTI),               (GDA( 8),  BCTB),
     4            (GDA( 9),PFLAG),               (GDA(10), CFLT),
     5            (GDA(11), DFLT),               (GDA(12),  THT),
     6            (GDA(18),FPKNT)
C
      COMMON  GDA(25), DEL, N, NNF, NCH,
     1 RMTX(160), ICTR
C
      COMMON /RXCMN/    NM, ILN, I, IFLG, LN, IRSLG, IFGLG,
     1 BB1KJ, BB2KJ, BB3KJ, GG1KJ, GG2KJ, GG3KJ, DD1KJ, DD2KJ, DD3KJ,
     2 GG13KJ, TEETK, TTHTK, MEETK, MTHTK, BB1P, BB3P, GG1P, GG2P,
     3 DD1P, DD3P, GG13P, TEETP, MEETP, OMG(5,5), ALM(5,5), LLT(5),
     4 WTH, WEE, GAM, RHO, WEEP, LAM, LAM2, LAM02, LAM202, LAM204,
     5 LAM208, GAM2, WTMWE, WTMWE2, PFLT, PTHK, PNK, PEEK, P, X,
     6 DCM1KJ, DCM2KJ, DCM3KJ, DCM4KJ, DCM5KJ, DCC1KJ, DCC2KJ, DCC3KJ,
     7 DCK1KJ, DCK2KJ, DCK3KJ
C
      COMMON /BLOCK1/  EMM2 (50,50)
      COMMON /BLOCK2/  EMM3 (50)
C
      REAL   LAM, LAM02, LAM202, LAM204, LAM208, LAM2, KUR0, KUR02,
     1          MEETK, MTHTK, MEETP,      LLT
C
C
      KK5= 5 * NNF
      IF (K .NE. 1) GO TO 164
      IF (J .NE. 1) GO TO 164
C                                          ZERO MATRIX ARRAYS
      DO 163 KN = 1,KK5
      DO 162 JJ = 1,KK5
      AL(KN,JJ) = 0.0
      AM(KN,JJ) = 0.0
```

```
      AN(KN,JJ) = 0.0
  162 CONTINUE
  163 CONTINUE
C
  164 DELD = DEL/DELT
      AL(5*K-4,5*J-4) = -10. * DELD * DCM1KJ - 6.0*DEL*DCC1KJ
      AL(5*K-3,5*J-3) = -10.*DELD * DCM2KJ- 6.0*DEL*DCC2KJ
      AL(5*K-2,5*J-2) = -10.0 * DELD * DCM1KJ - 6.0 * DEL * DCC3KJ
      AL(5*K-1,5*J-1) = -10.0 * DELD * DCM4KJ - 6.0 * DEL * DCC3KJ
      AL(5*K,5*J) = -10. * DELD * DCM5KJ
C
      AM(5*K-4,5*J-4) = 8.0* DELD * DCM1KJ + 3.0 * DEL*DCC1KJ
      AM(5*K-3,5*J-3) = 8.0*DELD * DCM2KJ+ 3.0* DEL*DCC2KJ
      AM(5*K-2,5*J-2)  = 8.0 * DELD * DCM1KJ + 3.0 * DEL * DCC3KJ
      AM(5*K-1,5*J-1) = 8.0 * DELD * DCM4KJ
      AM(5*K,5*J) = 8.0* DELD * DCM5KJ
C
      AN(5*K-4,5*J-4) = -2.0* DELD * DCM1KJ - 2.0/3.0*DEL*DCC1KJ
      AN(5*K-3,5*J-3) = -2.0 * DELD * DCM2KJ - 2.0/3.0 * DEL* DCC2KJ
      AN(5*K-2,5*J-2) = -2.0 * DELD * DCM1KJ - 2.0 / 3.0 * DEL * DCC3KJ
      AN(5*K-1,5*J-1) = -2.0 * DELD * DCM4KJ
      AN(5*K,5*J) = -2.0 * DELD * DCM5KJ
C
      IF (K .NE. KK) GO TO 225
      IF (J .EQ. KK) GO TO 235
  225 IFDYN = 10
      GO  TO  249
C                                MATRICES ARE COMPLETE - SAVE ON TP 13
C
C
  235 IFDYN = 1
      WRITE (13) AL
      WRITE (13) AM
      WRITE (13) AN
      BACKSPACE  13
      BACKSPACE  13
      BACKSPACE  13
  249 IF(IPFLAG .EQ. -1.E+10) WRITE(6,250)
  250 FORMAT (1H-, 9HLEFT LMN. )
      RETURN
      END
```


REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

3.12j GMTX Subroutine (GDYN)

```
$IBFTC GDYN
C                                        FORM G**, B**, P AND X MATRICES
C
      SUBROUTINE  GMTX(J,K,KK,IFMX)
C
C
      DIMENSION  PE(50), AL(50,50), FMM4(50), A(50,50), B(50,50),
     1           C(50,50), P(50,50), X(50)
      DIMENSION  CONDI(150), ZP(50), ZPP(50), ZPPP(50)
C
      EQUIVALENCE (CONDI(  1),  ZP),(CONDI( 51), ZPP),(CONDI(101),ZPPP)
C
      EQUIVALENCE (GDA( 1),   A0),              (GDA( 2),   H0),
     1            (GDA( 3),   E0),              (GDA( 4), SIG0),
     2            (GDA( 5),  POI),              (GDA( 6),  ENF),
     3            (GDA( 7), BCIT),              (GDA( 8), BCIB),
     4            (GDA( 9),PFLAG),              (GDA(10), CEXT),
     5            (GDA(11), DELT),              (GDA(12),  THT),
     6            (GDA(18),FPRNT)
C
      COMMON  GDA(25),  DEL, N, NNF, NTH,
     1 BMTX(160), TCTR, KZTW, KZTR, KPTW, KPTR
C
      COMMON /PXCMN/     MM, LLN, I, IFLG, LN, IRSFG, IFGFG,
     1 BB1KJ, BB2KJ, BB3KJ, GG1KJ, GG2KJ, GG3KJ, DD1KJ, DD2KJ, DD3KJ,
     2 GG13KJ, TFETK, TTHTK, MFETK, MTHTK, BB1P, BB3P, GG1P, GG2P,
     3 DD1P, DD3P, GG13P, TFETP, MFETP, OMG(5,5), ALM(5,5), LLI(5),
     4 WTH, WFE, GAM, RHO, WFEP, LAM, LAM2, LAM02, LAM202, LAM204,
     5 LAM208, GAM2, WTMWF, WTMWF2, WFWT, PTHK, PNK, PFEK, P, X,
     6 DCM1KJ, DCM2KJ, DCM3KJ, DCM4KJ, DCM5KJ, DCC1KJ, DCC2KJ, DCC3KJ,
     7 DCK1KJ, DCK2KJ, DCK3KJ
C
      COMMON /BLOCK1/  EMM2 (50,50)
      COMMON /BLOCK2/  EMM3(50)
C
      REAL   LAM, LAM02, LAM202, LAM204, LAM208, LAM2, KORO, KORO2,
     1         MFETK, MTHTK, MFETP, LT, LLI
C
      KK5 = 5 * NNF
      IF (K .NE. 1) GO TO 20
      IF (J .NE. 1) GO TO 30
      DO 15  KN = 1,KK5
      PE(KN) = 0.0
```

```
   15 CONTINUE                                                          00000390
   20 IF(ITCTR .EQ. 1.) GO TO 22                                        00000392
      READ (4)   TMP1, TMP2, TMP3, TMP4, TMP5                           00000394
      GO  TO  28                                                        00000396
C                                                                       00000399
   22 KORO = FLOAT(K-1) /RHO                                            00000400
      TMP1  =  TFETP + GAM *(TFETK - TTHTK)                             00000401
      TMP2  =  TTHTK * KORO                                             00000402
      TMP3  =  WFE * TFETK + WTH * TTHTK                                00000403
      TMP4  =  LAM2 *(MFETP + GAM *(MFETK - MTHTK) )                    00000404
      TMP5  =  KORO * LAM2 * MTHTK                                      00000405
      WRITE (4)  TMP1, TMP2, TMP3, TMP4, TMP5                           00000408
C                                                                       00000410
C                PE MATRIX                                              00000420
C                                                                       00000430
   28 PE(5*K-4)  =  - PFEK + TMP1                                       00000440
      PE(5*K-3) =  - PTHK    -   TMP2                                  00000450
      PE(5*K-2)  =  - PNK  -  TMP3                                      00000460
      PE(5*K-1)  =  TMP4                                                00000470
      PE(5*K)  =  - TMP5                                                00000480
   30 IF (K .NE. KK) GO TO 35                                           00000490
      IF (J .EQ. KK) GO TO 40                                           00000500
   35 IFMX = 10                                                         00000510
      GO  TO  519                                                       00000520
C                                                                       00000530
   40 DO 50 KN = 1,KK5                                                  00000540
      PE(KN) = 2. * DEL * DELT * PE(KN)                                 00000550
   50 CONTINUE                                                          00000560
C                                                                       00000570
      IF (ITCTR .GT. 1.0 .AND. MM .EQ. 2) GO TO 70                      00000590
      IF (ITCTR .GT. 1.0) GO TO 60                                      00000600
      DO 55 KN = 1,150                                                  00000610
   55 CONDI(KN) = 0.0                                                   00000620
      WRITE ( 8) ZP, ZPP, ZPPP                                          00000640
      GO TO 85                                                          00000650
C                                                                       00000658
   60 IF (KZTR .EQ. 8) GO TO 66                                         00000660
      BACKSPACE  11                                                     00000670
      BACKSPACE  11                                                     00000680
      GO TO 80                                                          00000690
C                                                                       00000698
   66 BACKSPACE  8                                                      00000700
      BACKSPACE  8                                                      00000710
```


REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      GO TO 80                                                          00000720
C                                                                       00000728
   70 AF = FILE (KZTR, 1)                                               00000730
      IF (KZTR .EQ. 8) GO TO 76                                         00000740
      BACKSPACE  11                                                     00000750
      GO TO 80                                                          00000760
C                                                                       00000768
   76 BACKSPACE  8                                                      00000770
C                                                                       00000778
   80 READ (KZTR) ZP, ZPP, ZPPP                                         00000780
C                                                                       00000790
   85 DELT2 = DELT * DELT                                               00000810
      IF (TCTR - 2.0) 90,212,215                                        00000820
   90 DO 95 K = 1,KK5                                                   00000830
      EMM3(K) = 2. * ZP(K)                                              00000840
      EMM4(K) = DELT2 * ZPP(K)                                          00000850
      ZPP(K) = EMM3(K) + EMM4(K)                                        00000860
      EMM4(K) = 6. * EMM4(K)                                            00000870
      ZPPP(K) = 6. * DELT * ZPPP(K)                                     00000880
      EMM3(K) = 9. * ZP(K)                                              00000890
      ZPPP(K) = EMM4(K) + ZPPP(K) + EMM3(K)                             00000900
   95 CONTINUE                                                          00000910
C                                                                       00000920
C                                                   READ L MATRIX       00000928
  150 READ (13) AL                                                      00000930
      CALL MMY (KK5,KK5,  1, AL, ZP, EMM3)                              00000940
C                                                   READ M MATRIX       00000948
      READ (13) EMM2                                                    00000950
      CALL MMY (KK5,KK5,  1, EMM2, ZPP, EMM4)                           00000960
      CALL MAD (KK5,  1, EMM4, EMM3, EMM3)                              00000970
C                                                                       00000980
      DO 170 K = 1,KK5                                                  00000990
      DO 170 J = 1,KK5                                                  00001000
      EMM2(K,J) = 0.33333333 * EMM2(K,J)                                00001010
      AL(K,J) = AL(K,J)/6.                                              00001020
  170 CONTINUE                                                          00001030
      CALL MAD (KK5,KK5, AL, EMM2, AL)                                  00001040
C                                                   READ N MATRIX       00001048
  180 READ (13) EMM2                                                    00001050
      CALL MMY (KK5,KK5,  1, EMM2, ZPPP, EMM4)                          00001060
      DO 190 K = 1,KK5                                                  00001070
      EMM3(K) = EMM3(K) + EMM4(K)                                       00001080
      PE(K) = PE(K) + EMM3(K)                                           00001090
```

```
  190 CONTINUE
C
  208 BACKSPACE  12
  209 DO 210 K = 1,KK5
      DO 210 J = 1,KK5
  210 EMM2(K,J) = 3.5 * EMM2(K,J)
      CALL MSU (KK5,KK5, EMM2, AL, AL)
      GO TO 225
C
C
  212 DO 213 K = 1,KK5
      ZPPP(K) = DELT2 * ZPPP(K)
      EMM3(K) = 2. * ZPP(K)
      ZP(K) = 8. * ZP(K)
      ZPPP(K) = ZPPP(K) + EMM3(K) + ZP(K)
      ZPP(K) = ZPP(K) + ZP(K)/3.0
      ZP(K) = ZP(K) /6.0
  213 CONTINUE
C
  215 READ ( 9) AL
      CALL MMY (KK5,KK5,  1, AL, ZP, EMM3)
      READ ( 9) EMM2
      CALL MMY (KK5,KK5,  1, EMM2, ZPP, EMM4)
      CALL MAD (KK5,  1, EMM4, EMM3, EMM3)
      READ ( 9) EMM2
      CALL MMY (KK5,KK5,  1, EMM2, ZPPP, EMM4)
      DO 220 K = 1,KK5
      EMM3(K) = EMM3(K) + EMM4(K)
      PE(K) = PE(K) + EMM3(K)
  220 CONTINUE
C
C                 G DBLSTAR HAS BEEN GENERATED AND STORED '' PE
C
C
  225 IF (TCTR .LE. 2.0)  GO TO 230
      READ (13) A, B, C
      GO TO 250
C
  230 READ (12) A, B, C
C                                                  COMPUTE  B**
      IF (TCTR .GT. 1.0) GO TO 250
      CALL MAD (KK5,KK5, B, AL, B)
C
```

```
C                                          P AND X ARE READY TO BE CREATED   00001528
  250 IF (MM .NE. 2) GO TO 400                                               00001530
C                                                                            00001540
C                                                STATION NO. 2 - TOP BOUNDARY 00001548
      IF (TCTR - 2.0) 265,260,300                                            00001550
  260 AF = FILE (3,2)                                                        00001560*
      BACKSPACE  3                                                           00001570
      READ ( 3) EMM4                                                         00001580
      BACKSPACE 3                                                            00001590
      BACKSPACE 3                                                            00001600
      READ ( 3) AL                                                           00001610
      BACKSPACE 3                                                            00001620
      BACKSPACE 3                                                            00001630
C                                                                            00001640
      CALL MMY (KK5,KK5,KK5, AL, C, EMM2)                                    00001650
      CALL MMY (KK5, KK5, KK5, EMM2, A, AL)                                  00001660
C                                                                            00001668
      READ ( 3)  P                                                           00001670
      BACKSPACE 3                                                            00001680
      BACKSPACE 3                                                            00001690
      CALL MSU (KK5, KK5, P, AL, AL)                                         00001700
      CALL MMY (KK5, KK5,KK5, EMM2, B, P)                                    00001710
C                                                                            00001720
      READ ( 3)  B                                                           00001730
      CALL MSU (KK5, KK5, B, P, B)                                           00001740
      CALL MMY (KK5, KK5,  1, EMM2, PE, EMM3)                                00001750
      CALL MSU (KK5,  1, EMM4, EMM3, EMM4)                                   00001760
      CALL INVNNM (B, 50, KK5, IERR)                                         00001770
      IF (IERR .LE. 0)  GO TO 320                                            00001780
C                                                                            00001789
C                                              INVERSE REPLACES B ON TAPE    00001798
      WRITE (13) A, B, C                                                     00001800
C                                                                            00001810
      CALL MMY (KK5, KK5, KK5, B, AL, P)                                     00001820
      CALL MMY (KK5, KK5,  1, B, EMM4, X)                                    00001830
      GO TO 500                                                              00001840
C                                                                            00001850
C                                                          READ  GO          00001858**
  265 READ (11) EMM4                                                         00001860
      BACKSPACE 11                                                           00001870
      BACKSPACE 11                                                           00001880
C                                                          READ  CO          00001888
      READ (11) AL                                                           00001890
```

*BACKSPACE 3 WHEN NOT USING ALTIØ
**265 BACKSPACE 11 - REMOVE STATEMENT NO. 265 FROM CARD 1860

```
      BACKSPACE 11                                                      00001900
      BACKSPACE 11                                                      00001910
      CALL MMY (KK5,KK5,KK5, AL, C, EMM2)                               00001920
      CALL MMY (KK5,KK5,KK5, EMM2, A, C)                                00001930
C                                                   READ  A0            00001938
      READ (11) AL                                                      00001940
      BACKSPACE 11                                                      00001950
      BACKSPACE 11                                                      00001960
      CALL MSU (KK5,KK5, AL, C, A)                                      00001970
      CALL MMY (KK5,KK5,KK5, EMM2, B, C)                                00001980
C                                                   READ  B0            00001988
      READ (11) AL                                                      00001990
  270 CALL MSU(KK5,KK5, AL, C, AL)                                      00002000
      CALL MMY (KK5,KK5,  1, EMM2, PE, EMM3)                            00002010
      CALL MSU (KK5,  1, EMM4, EMM3, EMM4)                              00002020
      CALL INVNNM (AL, 50, KK5, IERR)                                   00002030
      IF (IERR .LE. 0)  GO TO 320                                       00002040
  275 CALL MMY (KK5,KK5,KK5, AL, A, P)                                  00002050
      CALL MMY (KK5,KK5,  1, AL, EMM4, X)                               00002060
      GO TO 500                                                         00002070
C                                                                       00002080
  300 READ (KPTR) MM, P, X                                              00002090
C                                                                       00002098
      AF = FILE (3,2)                                                   00002100
      BACKSPACE  3                                                      00002110*
      READ ( 3) EMM4                                                    00002120
      BACKSPACE 3                                                       00002130
      BACKSPACE 3                                                       00002140
      READ ( 3) AL                                                      00002150
      BACKSPACE 3                                                       00002160
      BACKSPACE 3                                                       00002170
C                                                                       00002180
      CALL MMY (KK5,KK5,KK5, AL, C, EMM2)                               00002190
      CALL MMY (KK5,KK5,  1, EMM2, PE, EMM3)                            00002210
      CALL MSU (KK5,  1, EMM4, EMM3, EMM4)                              00002220
      CALL MMY (KK5,KK5,  1, B,  EMM4, X)                               00002240
      GO TO 500                                                         00002250
C                                                                       00002260
C                                         * *  COMPUTE P AND X  **      00002270
C                                                                       00002280
  400 IF (TCTR .GT. 2.0) GO TO 450                                      00002290
      CALL MMY (KK5,KK5,KK5, C, P, EMM2)                                00002300
      CALL MSU (KK5,KK5, B, EMM2, B)                                    00002310
```

*BACKSPACE 3 WHEN NOT USING ALTIØ

```
      CALL INVNNM (B, 50, KK5, IERR)                                    00002320
      IF (IERR .LE. 0) GO TO 320                                        00002330
C                                                                       00002340
      IF (TCTR .EQ. 1.0) GO TO 450                                      00002350
      WRITE (13) A, B, C                                                00002360
  450 CALL MMY(KK5,KK5,  1, C, X, EMM3)                                 00002370
      IF (TCTR .LE. 2.0) GO TO 460                                      00002380
      READ (KPTR) MM, P, X                                              00002390
  460 CALL MSU (KK5,  1, PE, EMM3, PE)                                  00002400
      CALL MMY (KK5,KK5,  1, B, PE, X)                                  00002410
C                                                                       00002420
      IF (TCTR .GT. 2.0) GO TO 480                                      00002430
      CALL MMY (KK5,KK5,KK5, B, A, P)                                   00002440
C                                                                       00002450
  480 IF (MM .NE. N-2) GO TO 500                                        00002460
      IF (TCTR .GT. 1.0) GO TO 500                                      00002470
      WRITE (11) P, X                                                   00002480
      GO TO 510                                                         00002490
C                                                                       00002498
  500 IF (MM .NE. 2) GO TO 510                                          00002500
      IF (TCTR .GT. 1.0) GO TO 504                                      00002510
      AF = FILE(11,1)                                                   00002520
      READ (11) A, B, C                                                 00002530
      BACKSPACE  11                                                     00002540
      WRITE (11) A, B, C, PE                                            00002550
      END FILE  11                                                      00002560
      GO TO 510                                                         00002570
C                                                                       00002578
  504 AF = FILE (3,1)                                                   00002580
      READ ( 3) A, B, C, EMM3                                           00002590
      READ ( 3) A, B, C, EMM3                                           00002600
      BACKSPACE  3                                                      00002610
      WRITE ( 3) A, B, C, PE                                            00002620
C                                                                       00002628
  510 IF (TCTR .GT. 2.0) GO TO 514                                      00002630
      WRITE (10) MM, P, X                                               00002640
      GO TO 515                                                         00002650
C                                                                       00002659
  514 WRITE (KPTW) MM, P, X                                             00002660
      IF (MM .EQ. N-1) GO TO 540                                        00002670
  515 IFMX = 1                                                          00002680
  519 IF(PFLAG .EQ. -1.E+10)  WRITE(6,520)                              00002690
  520 FORMAT (1H-,10HLEFT GDYN. )                                       00002700
```

```
      RETURN                                                            00002715
C                                                                       00002716
  540 IF (KPTW .EQ. 12) GO TO 550                                       00002720
      BACKSPACE  10                                                     00002730
      BACKSPACE  10                                                     00002740
      GO TO 515                                                         00002750
C                                                                       00002759
  550 BACKSPACE  12                                                     00002760
      BACKSPACE  12                                                     00002770
      GO TO 515                                                         00002780
C                                                                       00002789
  320 WRITE (6,325) MM                                                  00002790
  325 FORMAT (1H-,9X,16HSINGULAR MATRIX.,9X,4HMM= ,I4)                  00002800
      CALL EXIT                                                         00002805
      STOP                                                              00002806
C                                                                       00002820
      END                                                               00002830
```


REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

3.12k ZMTRX Subroutine (SLND)

```
$INSERT SLND
C
C          SOLUTION MATRICES
C
      SUBROUTINE  ZMTRX
C
C
      DIMENSION  PX(50,5 ),  X(5 ), X(5 ), AZ(50 ,5 ), BB(50 ,5 ),
     1 CCI(50,50), GGI(5 ), Z(5 )
      DIMENSION ZX2(50),          BB2(5 ,5 ), GG2(5 )
      DIMENSION  ZP(5 ), ZPP(5 ), ZPPP(5 ),              (1 )
C
      EQUIVALENCE  (GDA( 9),PELAG),              (GDA(11),DELT)
C
      COMMON /BLK1/  EMM2    (5 ,5 )
      COMMON /BLK2/  EMM3    (5 )
      COMMON  GDA(25), DEL, N, NNF, NTH,
     1 BMIX(16 ), ICIR, KZ  , KZIR, KPI , KPIR
C
      IF (KZTR .EQ. 8) GO TO   1
      BACKSPACE  11
      GO TO   2
    1 BACKSPACE  8
C
    2 DO  4  I = 1,50
    4 Z(I) = 0.0
C
      IF (ICIR .GT. 1.0)  GO TO 5
      AF = FILF (3,3)
      REWIND  11
      BACKSPACE  3
C                                  READ P AND X MATRICES FOR STA. (N-2)
      READ ( 3) BB0, GG0
      END FILE 3
      GO TO 10
C
    5 AF = FILF (3,2)
      IF (ICIR .GT. 2.0) GO TO 8
      BACKSPACE  10
      READ (10) I, BB0, GG0
      GO TO 10
    8 READ (KPTW) I, BB0, GG0
C
```

*BACKSPACE 3 WHEN NOT USING AIMI∅

```
   10 KK5 = 5 * KK1
C
      DO 100 NN = 1,N
      MN = N - NN + 1
C
      IF (NN .NE. 1) GO TO 20
C                                                    LAST STATION
      IF (TCTR .GT. 2.0) GO TO 12
C
      READ (10) I, PX, XX
      GO TO 13
C
   12 READ (KPTW) I, PX, XX
C
   13 IF (TCTR .GT. 1.0) GO TO 14
      READ (10) AAI1, BBI1, CCI1, GGI1
      GO TO 16
C
   14 READ ( 3) AAI1, BBI1, CCI1, GGI1
      AF = FILE (3,1)
C
   16 CALL MMMY (KK5,KK5,KK5, AAI1, BB.., EMM2 )
      CALL MMSU(KK5, KK5, EMM2, BBI1, BBI1 )
      CALL MMMY(KK5, KK5,  1, AAI1, GG.., EMM3 )
      CALL MMSU(KK5,   1, GGI1, EMM3, GGI1 )
      CALL MMMY(KK5, KK5,  1, BBI1, XX, EMM3 )
      CALL MMAD(KK5,  1, GGI1, EMM3, GGI1 )
      CALL MMMY(KK5, KK5, KK5, BBI1, PX, EMM2 )
      CALL MMAD(KK5, KK5, CCI1, EMM2, CCI1 )
      CALL INVMS  (CCI1, 5 , KK5, IERR )
      IF (IERR .LE. 0) GO TO 120
      CALL MMMY(KK5, KK5,  1, CCI1, GGI1, ZX )
C
      WRITE ( 3) MN, ZX
C
      WRITE (6,130)  MN, (ZX(KP), KP=1,KK5)
      GO TO 100
C
   20 IF (NN .EQ. N) GO TO 40
      IF (TCTR .GT. 2.0) GO TO 22
      IF (TCTR .EQ. 2.0 .AND. NN .EQ. 2)  GO TO  21
      BACKSPACE 10
   21 BACKSPACE  10
```


REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
C
      READ  (10) 1, PX, XX
C
      GO TO 25
C
   22 IF (KPTW .EQ. 10) GO TO  23
      IF (NN .NE. 2)  BACKSPACE 12
      BACKSPACE  12
      GO TO  24
C
   23 IF (NN .NE. 2)  BACKSPACE 10
      BACKSPACE  10
   24 READ (KPTW) 1, PX, XX
   25 READ (KZTR) ZP, ZPP, ZPPP
C
      IF (TCT2 .NE. 1.0) GO TO 27
      IF (KZTR .EQ. 8) GO TO  26
      BACKSPACE 11
      BACKSPACE 11
      GO TO 27
C
   26 BACKSPACE  8
      BACKSPACE  8
   27 CALL MMMY (KK5,KK5,  1, PX, ZY, Z)
      CALL MMSU (KK5, 1, XX, Z, Z )
C
      IF (NN .NE. N-2) GO TO 30
      DO 28 K = 1,KK5
   28 ZX2(K) = Z(K)
C
   30 DO 35 K = 1,KK5
   35 ZX(K) = Z(K)
C
      WRITE  (KZTW)  ZX, ZP, ZPP
C
      WRITE ( 3) MN, ZX
C
      WRITE (6, 140) MN,  (ZX(IX), IX=1,KK5)
C
      GO TO 100
C                                                    FIRST STATION
   40 END FILE  3
      REWIND  3
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      AF = FILE (3,2)
      READ ( 3) AAII, BBII, CCII, GGII
C                                          A, B, C-1, G FOR 2N, 5TA,
      READ ( 3) AAII, BBII, CCII, GGII
      AF = FILE (3,2)
C
      CALL MMMY(KK5,KK5,  1, BBII, ZX, PX )
      CALL MMMY(KK5,KK5,  1, AAII, ZX2, ZX )
      CALL MMSU(KK5,  1, GGII, ZX, GGII )
      CALL MMSU(KK5,  1, GGII, PX, GGII )
      CALL MMMY(KK5,KK5,  1, CCII, GGII, Z )
C
      WRITE (6, 140) MN, (Z(IX), IX=1,KK5)
C                                          ZJ IN FILE BY ITSELF
      WRITE ( 3) MN, Z
  100 CONTINUE
C
C                                          COMPUTE VELOCITIES
      DO 102 I = 1,10
  102 W(I) = 0.0
C
      REWIND  3
      REWIND KZTR
      REWIND KZTW
      DELT2 = DELT * DELT
C
      DO 108 I = 1,N
      IF (I .EQ. 1 .OR. I .EQ. N) GO TO 108
C
      READ (KZTW)  ZP,ZPP,ZPPP
C
      DO 104 NF = I,NNF
      NT = (NF - 1) * 5
      NW = NT + 3
      IF (TCTR .NE. 1.0) GO TO 103
      DO = ZPP(NW)
      ZM1 = ZPPP(NW) *DELT2 + 2. * ZPP(NW) - ZP(NW)
      W(NF) = (3.*ZP(NW) - 4.*DO + ZM1) / (2.*DELT)
      GO TO 104
  103 W(NF) = (3.*ZP(NW) - 4.*ZPP(NW) + ZPPP(NW)) / (2.*DELT)
  104 CONTINUE
C
  106 WRITE (KZTR)  W
```

```
C
  100 CONTINUE
C
      END FILE KZTW
C
      S1 = KZTR
      KZTR = KZTW
      KZTW = S1
      S1 = KPTR
      KPTR = KPTW
      KPTW = S1
      IF(PFLAG .EQ. -1.E+10)  WRITE(6,109)
  109 FORMAT (1H-, 10HLEFT SOLIN )
C
  110 RETURN
C
  120 WRITE (6,125)
      CALL EXIT
      STOP
  125 FORMAT (1H-,10X,45HERROR -  INVERSION OF SINGULAR MATRIX IN 'Z', 1
C
  130 FORMAT (1H1,10X,24H'Z' MATRIX FOR STATION  , I4// (12X,1P5E17.7))
  140 FORMAT (/// 10X,24H'Z' MATRIX FOR STATION  , I4// (12X,1P5E17.7))
C
      END
```


REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
$IBFTC SUMD                                                             00000010
C              SOLUTION SUMMATION   W/O STRESSES              **        00000020
C                                                                       00000030
      SUBROUTINE SUMS                                                   00000040
C                                                                       00000041
C   NOMENCLATURE                                                        00000042
C     THT         HORIZONTAL ANGLES (DEGREES)  5 MAXIMUM                00000043
C     FPRNT       FOURIER COMPONENT PRINT VALUES OF DEFLECTIONS AND     00000044
C                 ROTATIONS   (3 POSSIBLE)                              00000045
C                                                                       00000050
      DIMENSION   Z(50), THT( 6), FPRNT( 3)                             00000060
      DIMENSION  ULZ(100,5), VLZ(100,5), WLZ(100,5), PHSLZ(100,5),      00000070
     1           PHTLZ(100,5), UL(100,5), VL(100,5), WL(100,5),         00000080
     2           PHSL(100,5), PHTL(100,5), WFEX(100), WTHX(100),        00000090
     3           GAMA(100), RHOX(100), WFEPX(100)                       00000100
C                                                                       00000110
      DIMENSION  B1(100,5), B2(100,5), B3(100,5), D1(100,5), D2(100,5), 00000150
     1           D3(100,5), G1(100,5), G2(100,5), G3(100,5), G13(100,5),00000151
     2           ENTSI(100,5), ENTTH(100,5), EMTSI(100,5), EMTTH(100,5) 00000152
C                                                                       00000160
      DIMENSION  ENFE( 5), ENTH( 5), ENFT( 5), QFE( 5), EMFE( 5),       00000170
     1           EMTH( 5), EMFT( 5), QTH( 5)                            00000180
C                                                                       00000190
      DIMENSION  COEFC(2200)                                            00000240
      DIMENSION  SCB1(100), SCB2(100), SCB3(100), SCG1(100), SCG2(100), 00000250
     1 SCG3(100), SCD1(100), SCD2(100), SCD3(100), SCG13(100),          00000251
     2 SCNTSI(100), SCNTTH(100), SCMTSI(100), SCMTTH(100), DCM1(100),   00000252
     3 DCM4(100), DCC1(100), DCC2(100), DCC3(100), DCK1(100), DCK2(100),00000253
     4 DCK3(100)                                                        00000254
C                                                                       00000309
      EQUIVALENCE (COEFC(   1),  SCB1),          (COEFC( 101),  SCB2), 00000310
     1            (COEFC( 201),  SCB3),          (COEFC( 301),  SCG1), 00000311
     2            (COEFC( 401),  SCG2),          (COEFC( 501),  SCG3), 00000312
     3            (COEFC( 601),  SCD1),          (COEFC( 701),  SCD2), 00000313
     4            (COEFC( 801),  SCD3),                                00000314
     5 (COEFC( 901), SCG13), (COEFC(1001),SCNTSI), (COEFC(1101),SCNTTH),00000315
     6 (COEFC(1201),SCMTSI), (COEFC(1301),SCMTTH), (COEFC(1401),  DCM1),00000316
     7 (COEFC(1501),  DCM4), (COEFC(1601),  DCC1), (COEFC(1701),  DCC2),00000317
     8 (COEFC(1801),  DCC3), (COEFC(1901),  DCK1), (COEFC(2001),  DCK2),00000318
     9 (COEFC(2101),  DCK3)                                             00000319
C                                                                       00000390
      EQUIVALENCE (GDA( 1),   A0),            (GDA( 2),   H0),          00000400
     1            (GDA( 3),   E0),            (GDA( 4), SIG0),          00000410
```

```
     2            (GDA( 5),  POI),              (GDA( 6),  ENF),          00000420
     3            (GDA( 7), BCIT),              (GDA( 8), BCIB),          00000430
     4            (GDA( 9),PFLAG),              (GDA(10), CEXT),          00000431
     5            (GDA(11), DELT),              (GDA(12),  THT),          00000432
     6            (GDA(18),FPRNT)                                         00000433
C                                                                         00000510
      COMMON  GDA(25), DEL, N, NNF, NTH,                                  00000520
     1 BMTX(160), TCTR, KZTW, KZTR, KPTW, KPTR                            00000521
C                                                                         00000530
      REWIND   9                                                          00000540
C                                                                         00000600
      AF = FILE (3,3)                                                     00000610
      SS1 = SIGO/EO                                                       00000620
      SS2 = SS1 * AO                                                      00000630
C                                                                         00000639
      DO 200 I = 1,N                                                      00000640
      IF(I .EQ. 1)  WRITE( 6,246)                                         00000645
      II = N - I + 1                                                      00000650
      IF (I .EQ. N) AF = FILE (3,1)                                       00000655
C                                                                         00000660
      READ ( 3) MN, Z                                                     00000670
C                                                                         00000680
      WRITE (6,290) II                                                    00000690
C                                                                         00000700
      DO 35 IT = 1,5                                                      00000710
      B1(I,IT)  =  0.0                                                    00000720
      D1(I,IT)  =  0.0                                                    00000722
      ENTSI(I,IT)  =  0.0                                                 00000724
      EMTSI(I,IT)  =  0.0                                                 00000726
      ULZ(I,IT) = 0.0                                                     00000730
      VLZ(I,IT) = 0.0                                                     00000740
      WLZ(I,IT) = 0.0                                                     00000750
      PNSLZ(I,IT) = 0.0                                                   00000760
   35 PMTLZ(I,IT) = 0.0                                                   00000770
C                                                                         00000780
      IT = 1                                                              00000790
   50 U = 0.0                                                             00000800
      V = 0.0                                                             00000810
      W = 0.0                                                             00000820
      PHS = 0.0                                                           00000830
      PHT = 0.0                                                           00000840
      IX = 1                                                              00000850
      THETA = THT(IT)                                                     00000860
```

```
C
      DO 150 NF = 1,NNF
      ANF = NF - 1
      NT = (NF - 1) * 5
      NU = NT + 1
      NV = NT + 2
      NW = NU + 2
      NFS = NV + 2
      NFT = NW + 2
      FCT = COSD (ANF * THETA)
      FCT2 = FCT * SS2
      FCT1 = FCT * SS1
      FCT3 = SIND (ANF * THETA)
      FCT4 = SS2 * FCT3
      FCT5 = SS1 * FCT3
C
      U = U + Z(NU) * FCT2
      V = V + Z(NV) * FCT4
      W = W + Z(NW) * FCT2
      PHS = PHS + Z(NFS) * FCT1
      PHT = PHT + Z(NFT) * FCT5
C
C                                    DERIVATIVES WITH RESPECT TO THETA
      ULZ(I,IT) = ULZ(I,IT) - Z(NU) * FCT4 * ANF
      VLZ(I,IT) = VLZ(I,IT) + Z(NV) * FCT2 * ANF
      WLZ(I,IT) = WLZ(I,IT) - Z(NW) * FCT4* ANF
      PHSLZ(I,IT) = PHSLZ(I,IT ) - Z(NFS) * FCT5 * ANF
      PHTLZ(I,IT) = PHTLZ(I,IT) + Z(NFT) * FCT1 * ANF
C
      BNF = ANF + 1.
      IF (BNF .NE. FPRNT(IX)) GO TO 150
C
      WRITE (6,300) THETA, ANF, U, V, W, PHS, PHT
      IX = IX + 1
  150 CONTINUE
C
C                                                DISPLACEMENTS
      UL(I,IT) = U
      VL(I,IT) = V
      WL(I,IT) = W
      PHSL(I,IT) = PHS
      PHTL(I,IT) = PHT
C
```

```
      IF (THT(IT+1) .LT. 0.0) GO TO 200
      IT = IT + 1
      GO TO 50
  200 CONTINUE
C
      REWIND 3
      IIT = IT
      S1 = E0 * H0
      S2 = S1 * H0 * H0
      S3 = (1. - POI)/2.
      S4 = SIG0 * H0
      S5 = A0 /(S4 * H0 **2)
      II = 0
      NF2  =  2 * NNF
C
  210 READ ( 9) LN, IN, COEFC
      LLN = LN - IN + 1
      DO 240 I = 1,LLN
      II = II + 1
      K1 = NF2 * (I-1)
C
      DO 230 IT = 1,IIT
      THETA = THT(IT)
      DO 225 NF = 1,NF2
      ANF = NF - 1
      K2 = K1 + NF
      B1(II,IT) = B1(II,IT) + SCB1(K2) * COSD(ANF * THETA) * S1
      D1(II,IT) = D1(II,IT) + SCD1(K2) * COSD(ANF * THETA) * S2
      ENTSI(II,IT) = ENTSI(II,IT) + SCNTSI(K2) * COSD(ANF * THETA) * S4
      EMTSI(II,IT) = EMTSI(II,IT) + SCMTSI(K2) * COSD(ANF * THETA) * S5
  225 CONTINUE
      B11 = B1(II,IT)
      D11 = D1(II,IT)
      B2(II,IT) = B11
      B3(II,IT) = POI * B11
      D2(II,IT) = D11
      D3(II,IT) = POI * D11
      G1(II,IT) = S3 * B11
      G2(II,IT) = S3 * B11
      G3(II,IT) = B11 * S3
      G13(II,IT) = S3 * D11
      ENTTH(II,IT) = ENTSI(II,IT)
      EMTTH(II,IT) = EMTSI(II,IT)
```

```
  230 CONTINUE
  240 CONTINUE
C
      IF (LN .EQ. N) GO TO 245
      GO TO 210
C
  245 READ ( 3) WTHX, WFEX, GAMA, RHOX, WFEPX
C
      DNM = -2. * DEL
C
      DO 275 I = 1,N
      IF(I .EQ. 1)  WRITE( 6,246)
  246 FORMAT( 1H1 )
      L = N - I + 1
      H = L
      WFE = WFEX(L)
      WTH = WTHX(L)
      GAM = GAMA(L)
      RHO = RHOX(L)
      DRHO =  0.
      IF(RHO .NE. 0.)  DRHO = 1. /RHO
      DA0  =  1./A0
      DO 265 IT = 1,IIT
      THETA = THT(IT)
C                                        DERIVATIVES WITH RESPECT TO PSI
      IF (I .NE. 1) GO TO 250
C                                                LAST STATION
      ULP = (4. * UL(I+1,IT) - 3. * UL(I,IT) - UL(I+2,IT) )/ DNM
      VLP = (4. * VL(I+1,IT) - 3. * VL(I,IT) - VL(I+2,IT) )/ DNM
      WLP = (4. * WL(I+1,IT) - 3. * WL(I,IT) - WL(I+2,IT) )/ DNM
      PHSLP = (4. * PHSL(I+1,IT) - 3. * PHSL(I,IT) - PHSL(I+2,IT) )/DNM
      PHTLP = (4. * PHTL(I+1,IT) - 3. * PHTL(I,IT) - PHTL(I+2,IT) )/DNM
      GO TO 260
  250 IF (I .EQ. N) GO TO 255
C                                       (NOTE - 'I' IS COUNTING BACKWARDS)
      ULP = (UL(I+1,IT) - UL(I-1,IT) ) /DNM
      VLP = (VL(I+1,IT) - VL(I-1,IT) ) /DNM
      WLP = (WL(I+1,IT) - WL(I-1,IT) ) /DNM
      PHSLP = (PHSL(I+1,IT) - PHSL(I-1,IT) ) /DNM
      PHTLP = (PHTL(I+1,IT) - PHTL(I-1,IT) ) /DNM
      GO TO 260
C                                                FIRST STATION
  255 ULP = (UL(I-2,IT) + 3. * UL(I,IT) - 4. * UL(I-1,IT) ) /DNM
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
      VLP = (VL(I-2,IT) + 3. * VL(I,IT) - 4. * VL(I-1,IT) ) /DNM
      WLP = (WL(I-2,IT) + 3. * WL(I,IT) - 4. * WL(I-1,IT) ) /DNM
      PHSLP = (PHSL(I-2,IT) + 3. * PHSL(I,IT) - 4. * PHSL(I-1,IT) )/DNM
      PHTLP = (PHTL(I-2,IT) + 3. * PHTL(I,IT) - 4. * PHTL(I-1,IT) )/DNM
C
C                 COMPUTATION OF STRAINS ** (STR 139, PG.17) **
C
  240 ESI = DA0 * (ULP + WFE * WL(I,IT))
      ETH = DA0 * (DRHO * VLZ(I,IT) + GAM * UL(I,IT) + WTH * WL(I,IT))
      ESTH = DA0/2. * (DRHO * ULZ(I,IT) + VLP - GAM * VL(I,IT) )
      GSI = PHSL(I,IT) - DA0 * (WFE * UL(I,IT) - WLP )
      GTH = PHTL(I,IT) - DA0 * (WTH * VL(I,IT) - DRHO * WLZ(I,IT) )
      EKSI = DA0 * PHSLP
      EKTH = DA0 * (DRHO * PHTLZ(I,IT) + GAM * PHSL(I,IT) )
      EKSTH = DA0/2. * (DRHO * PHSLZ(I,IT) + PHTLP - GAM * PHTL(I,IT) +
     1 DA0/2.* (WFE-WTH)*(DRHO* ULZ(I,IT) - VLP-GAM*VL(I,IT)))
      IF(PFLAG .EQ. -10.)  WRITE(6,261) L, THETA, ESI, ETH, ESTH, GSI,
     1                                  GTH, EKSI, EKTH, EKSTH
  261 FORMAT(/// 10X, 34HCHECKPRINT OF STRAINS AT STATION  ,I4,  5X,
     1  7HTHETA  ,1PE12.4// 8E13.4 )
C
C                 COMPUTATION OF INTERNAL FORCES ** (STR 139, PG.12) **
C
      ENFE(IT)  =  B1(L,IT) * ESI + B3(L,IT) * ETH - ENTSI(L,IT)
      ENTH(IT)  =  B3(L,IT) * ESI + B2(L,IT) * ETH - ENTTH(L,IT)
      ENFT(IT)  =  G1(L,IT) * ESTH
      QFE(IT)   = G2(L,IT) * GSI
      EMFE(IT)  =  D1(L,IT) * EKSI + D3(L,IT) * EKTH - EMTSI(L,IT)
      EMTH(IT)  =  D3(L,IT) * EKSI + D2(L,IT) * EKTH - EMTTH(L,IT)
      EMFT(IT)  =  G13(L,IT) * EKSTH
      QTH(IT)    = G3(L,IT) * GTH
  265 CONTINUE
C
      WRITE (6,268) L, (ENFE(J), ENTH(J), ENFT(J), QFE(J), J = 1,IIT)
      WRITE (6,270)  (EMFE(J), EMTH(J), EMFT(J), QTH(J), J = 1,IIT)
C
  268 FORMAT (1H-/11X,9HSTATION  ,I4// 19X, 5HN(XI),11X, 8HN(THETA), 7X,
     1 11HN(XI,THETA),10X, 5HQ(XI) /(12X, 1P4E17.7) )
C
  270 FORMAT (1H-,18X, 5HM(XI),11X, 8HM(THETA), 7X,11HM(XI,THETA), 8X,
     1 8HQ(THETA) / (12X, 1P4E17.7) )
C
  275 CONTINUE
```

```
C                                                                       00002900
      IF(PFLAG .EQ. -1.F+10)  WRITE(6,280)                              00002902
  280 FORMAT (1H-, 9HLEFT SUMS )                                        00002906
      RETURN                                                            00002910
  290 FORMAT (////10X, 9HSTATION   ,I4)                                 00002920
  300 FORMAT (/// 10X, 9HTHETA  = ,F7.2//11X, 9HENF     = ,F5.1//21X,1HU,00002930
     1 16X,1HV,16X,1HW,12X, 7HPHI(XI), 9X,10HPHI(THETA) /(12X,1P5E17.7))00002960
C                                                                       00002950
      END                                                               00002960
```

```
                    * MEMORY MAP *

SYSTEM                                  00000 THRU  03765
FILE BLOCK ORIGIN                       03766
FILES        1.     UNIT06              (NO BUFF POOL ATTACHED)
             2.     UNIT01              (NO BUFF POOL ATTACHED)
             3.     UNIT02              (NO BUFF POOL ATTACHED)
             4.     UNIT03              (NO BUFF POOL ATTACHED)
             5.     UNIT04              (NO BUFF POOL ATTACHED)
             6.     UNIT05              (NO BUFF POOL ATTACHED)
             7.     UNIT07              (NO BUFF POOL ATTACHED)
             8.     UNIT08              (NO BUFF POOL ATTACHED)
             9.     UNIT09              (NO BUFF POOL ATTACHED)
            10.     UNIT10              (NO BUFF POOL ATTACHED)
            11.     UNIT11              (NO BUFF POOL ATTACHED)
            12.     UNIT12              (NO BUFF POOL ATTACHED)
            13.     UNIT13              (NO BUFF POOL ATTACHED)
            14.     UNIT14              (NO BUFF POOL ATTACHED)
            15.     UNIT15              (NO BUFF POOL ATTACHED)
            16.     UNIT16              (NO BUFF POOL ATTACHED)
PRE-EXECUTION INITIALIZATION            04266
CALL ON OBJECT PROGRAM                  04275
OBJECT PROGRAM                          04302 THRU  76606

LINK  DECK  ORIGIN      CONTROL SECTIONS  (/NAME/=NON 0 LENGTH, (LOC)=DELETED, *=NOT REFERENCED

  0  EXEC    04302     ///     /(76706)    EVEN    04303     ......  04577 *
     CECRD   04613      DECRD  (04613)
     NOFPT   04732     /FPSIGQ/ 04732      EVEN    04731
     .FPTRP  04733      EVEN    04743      FP.UND  05002 *   FP.OVR  05030 *   FP.DIV  05041 *
                        FP.ODD  05050 *    .FPOUT  05060 *   .FPARG  05070 *  /NOOVRQ/ 05121 *
                       /NOUNDQ/ 05122 *   /NODIVQ/ 05123 *  /FPSIGQ/(04732)    FP.LOC  05124 *
                        OVFLOW  05125 *
     .LINK   05132     /.LDT  / 05132     /.LRECT/ 05146    /.LVEC / 05174
     .LXCON  05222      .LXSTR  05222      .LXSTP  05227     .LXOUT  05324 *   .LXERR  05336
                        (BEXIT  05342 *    .LXRTN  05342     .LXCAL  05346     .OBCLS  05534 *
                        .LXARG  05655 *    .LO     05674    /TDUMP(/ 05676 *   .CLSE   05702 *
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

		.LFBL	05703	.LUNB	05704 *	.DFOUT	05705	CTES..	05707 *
		SC.SWT	05710	.OPNFQ	05711 *	.CLSFQ	05712	.WRTFQ	05713 *
		.REDFQ	05714 *	CNTL..	05715				
.LOVRY	05720	.LOVRY	(05720)	.LDT	(05132)	.LRECT	(05146)	.LVEC	(05174)
.LXSL	06472	.LXSEL	06472	.LXSCL	06473	.LXTST	06476	.LXOVL	06544
		.LXRCT	06555 *	.LXIND	06714	.LXDIS	06722	.LXFLG	06723
		.LTCH	06730						
.ERAS.	06750	E.1	06750	E.2	06751	F.3	06752	E.4	06753
.XCC.	06754	CC.1	06754	CC.2	06755	CC.3	06756	CC..	06757
XIT	06760	EXIT	06760	.EXIT.	06760				
FXEM	06761	.FXEM.	06761	.FXOUT	07314 *	.FXARG	07322 *	/.OPTW./	07376 *
FCNV	07410	.FCON.	07410	/HWDSQ /	07440 *	/NOHSHQ/	07442 *	.FCNV.	07444
		.ENDFS	07463	.CNVSW	07465	.FDX1	07471	.FDX2	07472
		.DBC	07474	.DBC10	07632	.DBC20	07660	.DDSW	07670
		.DDFIX	07677	.FIXSW	07705	.DDBC	07762	.DDRS1	10226
		.DDRS2	10230 *	.D1	10233	.D2	10235	.FERR2	10322 *
		.ANPT	10356	.DNPT	10373	.LNTP	10456	.AOUT	10525
		.DFLT	10544	.FLT	10701	.DEXPN	10772	.FXD	10773
		.HOUT	11136	.INTG	11207	.LOUT	11327	.DOUT	11350
		.XCF	11402	.TEST	12126	.KOUNT	12131	.LIST	12134
		.DONE	12145	.OUTBF	12212	EVEN	12211	.BUF	12242
		.QSTO	12243	.WIDTH	12244	.GAIN	12245	.GAIN1	12246
		.FBDBF	12256	EVEN	12267	.DDDFL	12303	.DDFLG	12304
		.MQD	12305	.PEX	12306	.FEXP	12307	.DIG	12310
FIOB	12326	.FIOB.	12326	.FCNT	12431	.FBLT.	12527	.FBDT.	12547
		.FRLR.	12573	.FRLR.	(12573)	.FWLR.	12637	.FWLR.	(12637)
		.FBIB.	12677	.FRITE	12771				
FIOH	12777	.FIOH.	12777	.FFIL.	13564	.FRTN.	13611		
FIOS.	14007	..FIOS	14007	..FIOC	14067	..FSEL	14124	..FBCK	14142
		..FICK	14144	..FCHK	14253				
FRDD.	14504	..FRDD	14504						
FWRD.	14531	..FWRD	14531						
FBCD.	14557	..FBCD	14557	..FBCW	14561	..FBCB	14572		
FRDB.	14622	..FRDB	14622						
FWRB.	14651	..FWRB	14651						
FBIN.	15322	..FBID	15322	..FBIW	15324	..FBIB	15337		
FEFT.	15741	..FEFT	15741						

	FBST.	15760	..FBST	15760							
	FRWT.	16042	..FRWT	16042							
	FVIO.	16070	..FVIO	16070							
	UNIT06	16157	..UN06	16157							
	UN01	16160	.UN01.	16160							
	UN02	16161	.UN02.	16161							
	UN03	16162	.UN03.	16162							
	UN04	16163	.UN04.	16163							
	UN05	16164	.UN05.	16164							
	UN07	16165	.UN07.	16165							
	UN08	16166	.UN08.	16166							
	UN09	16167	.UN09.	16167							
	UN10	16170	.UN10.	16170							
	UN11	16171	.UN11.	16171							
	UN12	16172	.UN12.	16172							
	UN13	16173	.UN13.	16173							
	UN14	16174	.UN14.	16174							
	UN15	16175	.UN15.	16175							
	UN16	16176	.UN16.	16176							
	FSCD	16177	COSD	16177	SIND	16201					
	FSCN	16230	COS	16230	SIN	16231					
	FSQR	16424	SQRT	16424							
	FSLDI	16477	.FSLI.	16515	.FSDI.	16523 *					
	FSLBI	16534	.FBLI.	16552	.FBDI.	16560 *					
	FSLI	16572	.SLI.	16572	.SLI1.	16577	.SDI.	16605	.SDI1.	16613	
	FSLDO	16626	.FSLO.	16644	.FSDO.	16652 *					
	FSLBO	16663	.FBLO.	16701	.FBDO.	16707 *					
	FSLO	16721	.SLO.	16721	.SLO2.	16727	.SDO.	16734	.SDO2.	16743	
	FILE.Q	16755	FILE	16756							
	//	76706									
1	DLKDY	17117	///	/(176706)	DATLNK	17150					
	DNTP	17164	EVFN	17165	DINTRP	17451 *					
	CF3P	17510	EVEN	17511	CODIMA	20524					
	ENTP	20652	EVEN	20653	ENTERP	21041					
2	GHYD	21105	///	/(176706)	GEOM	27137					

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

3.13 PROGRAM NOMENCLATURE

Ⓐ

A0	a	Reference length
A0 (50, 50)	A_o	Boundary matrix, see Equation (1.46), Section 1.8
A1 (50, 50)		Another designation for A0 (50, 50)
ALF1		ENTRY to function subprogram TMP1, used to define the coefficient of thermal expansion for the stress calculations of layer one
ALF2		As ALF1, for layer two
ALM0 (5, 5)	$\bar{\Lambda}_o$	Boundary displacement matrix at the first meridional station, Equation (1.36), 0th harmonic
ALM1 (5, 5)		As ALM0 (5, 5) for the 1st harmonic
ALMN (5, 5)	$\bar{\Lambda}_N$	As ALM0 (5, 5) for the terminal boundary
ANX		Angle between the generator and axis of revolution; cone-cylinder option of GEØM.
AXL		Axial surface length

Ⓑ

B**		Modified B matrix for dynamic response, Equation (1.56), Section 1.9
B0 (50, 50)	B_o	Boundary matrix; see Equation (1.46), Section 1.8
B1 (50, 50)		Same as B0 (50, 50)
BBB		FUNCTIØN subprogram to define membrane stiffness, Equation (1.28).

BCD (36)		Three title cards read in executive program
BCIB		Boundary condition indicator for the terminal boundary; entered with the general data, GDA. See Section 2.8.
BCIT		As BCIB, for the initial boundary
BFCN		Deck name for the static version of subprogram, BBB
BMTX (160)		CØMMØN array which includes Ω, Λ and ℓ boundary matrices for the 0th and 1st harmonics
BN (50, 50)	B_N	Terminal boundary matrix; Equation (1.46), Section 1.8
BNDTB (17)		Special boundary array for station N read with the geometry data, GMDA. See Sections 1.8, 2.8.
BNDTX (17)		As BNDTB, for station one
C		
C0, C1 (50, 50)	C_o	Boundary matrices; see Equation (1.46), Section 1.8.
CEXT		Number of time cycles desired. Read with GDA data, Section 3.3.2.
CN (50, 50)	C_N	Terminal boundary matrix, Equation (1.46), Section 1.8.
CØDIMA		Parabolic curve fitting subroutine, see Section 3.5.4.
CØEFC		Stiffness coefficient array set up in the DATLYR subroutine.
D		
DATLD		Subroutine used to set up pressure loads

DATLNK		Sub-executive subroutine which monitors GEØM, DATLYR and DATLD
DATLYR		Subroutine which sets up the stiffness coefficient array, CØEFC
DBBDD		Deck name for the Dynamics version of subprogram BBB
DCC1, 2, 3	D_m	External damping coefficients found in the CØEFC array
DCK1, 2, 3	K_m	Spring coefficients, in CØEFC
DCM1, 4	M_m	Mass coefficients of translation and rotation, in CØEFC
DDD		ENTRY to function subprogram BBB, used to define the bending stiffness coefficients of the CØEFC array
DECRD		Data read subroutine; see Section 3.5.1
DEL	Δ	Interval size between meridional stations
DELT	δ	Time increment (seconds), read with GDA data. See Section 3.3.2.
DELTH		Interval size between circumferential stations. Constant = 2°.
DKDMP		Deck name for the Dynamics deck which sets up spring and damping coefficients of the array CØEFC
DKK1, 2, 3		FUNCTIØN and ENTRY points to the subprogram which sets up spring coefficient
DMASS		Deck name for Dynamics deck which forms the mass coefficients
DMM1, 4		FUNCTIØN and ENTRY points of deck DMASS

DMP1, 2, 3		ENTRY points to deck DKDMP to set up the translational damping coefficients
DN1		ENTRY point to function subprogram, EI1. Sets up distance from neutral axis of the first layer used in stress calculations.
DN2		As DN1 for the second stresses
DPRSS		Deck name for the Dynamics version of PPPN which supplies the pressure loadings
DTMP		Deck name for the Dynamics version of ENTT which gives the temperature load and moment
DYLMN		Subroutine subprogram in the Dynamics deck which sets up the L, M and N matrices of Equation (1.51), Section 1.9
E		
E0	E_o	Reference Young's Modulus
EI1, 2		FUNCTIØN subprograms used to define the moduli of elasticity for the two layers at which stresses are desired
EM		Number of radii entered for the discrete point geometry option
EMTT		ENTRY point to subprogram ENTT; used to set up temperature moments
EN		Number of meridional stations
ENF		Number of Fourier harmonics where the first one is the 0th one
ENTH		Number of circumferential stations, internally set at ninety

ENTT		FUNCTIØN subprogram which sets the value of the temperature load, t_T, Equation (1.15), Section 1.6
Ⓕ		
F(50, 50)	F	Matrix of equilibrium Equation (1.34), Section 1.7
FGHPE		Subroutine which forms the F, G and H matrices of Equation (1.34). In the Statics version the force matrix PE is also set up here.
FPRNT		Fourier component print values, read with GDA data. See Section 3.3.2.
Ⓖ		
G (50, 50)	G	Matrix of equilibrium Equation (1.34), Section 1.7
G**	$g^*_{i,j}$	Modified g matrix for dynamic response, Equation (1.56), Section 1.9
GAMA	γ	ρ'/ρ
GDA		General data array, read by the executive program
GEØM		Geometry subroutine
GIN		Geometry indicator read with GMDA data; see Section 2.5
GMDA		The geometry data array; see Section 3.3.3
GMI		$\lvert GIN \rvert$
GMTX		Dynamics subroutine subprogram which modifies the B and g matrices and forms the P and X matrices of Equation (1.60), Section 1.10

H (50, 50)	H	Equilibrium matrix of Equation (1.34), Section 1.7
H0	h_o	Reference thickness
Ⓘ		
I		Station number within each block of stiffness coefficients, CØEFC, in PANDX subroutine
IFDYN		PANDX flag which determines when L, M and N matrices of the Dynamics routine have been completed
IFGFG		PANDX flag to determine entry into the FGHPE subroutine
IFLG		Stiffness coefficient block indicator; used in PANDX
IFMX		PANDX flag which shows whether the PE (force) matrix is complete
IRSFG		PANDX flag for boundary computations
IRTE		PANDX indicator for first or last station within a coefficient block
Ⓙ		
JPATH		Path indicator through the Fourier harmonic loop of PANDX; used to skip setting up the boundary conditions and the FGHPE subroutine
Ⓚ		
KFCN		Deck name for the Statics version of subprogram DKK1
KPATH		Path indicator through the Fourier harmonic loop of PANDX; used to skip boundary conditions and the DYLMN subroutine

KPTR, KPTW		Variable tape numbers 10 or 12 used in the Dynamics version for storing P and X matrices by subroutine GMTX
KZTR, KZTW		Variable tape number 8 or 11 used in GMTX for preserving the three previous solution matrices (Z) at each time interval
Ⓛ		
L (50, 50)	L_i	Matrix used in the Dynamics version to modify the g matrix, Equation (1.51), Section 1.9
LL0 (5)	ℓ_o	Initial boundary, ℓ matrix; Equation (1.38), Section 1.7
LLLN (5)	ℓ_N	Terminal boundary, ℓ matrix
Ⓜ		
M (50, 50)	M_i	As L (50, 50)
M (THETA)	M_θ	Bending moment per unit length in the circumferential direction
M (XI)	M_ξ	Bending moment per unit length in the meridional direction
M (XI, THETA)	$\overline{M}_{\xi\theta}$	Bending moment; shear
MFE		Current temperature moments at a given meridional station and for k harmonics
MM		Absolute station number in PANDX subroutine
MMN		Station number within block of loads in PANDX
MTP		First derivative of the temperature moments

Ⓝ

N		Fixed point form of EN, number of meridional stations
N (50, 50)	N_i	As L (50, 50)
N (THETA)	N_θ	Membrane force, circumferential
N (XI)	N_ξ	Membrane force, meridional
N (XI, THETA)	$\bar{N}_{\xi\theta}$	Membrane shear force
NNF	n	Fixed point form of ENF, number of Fourier harmonics
NTH		Fixed point form of ENTH, number of circumferential stations = 90

Ø

ØMG0 (5, 5)	Ω_1	Boundary force matrix for the 0th harmonic; Equation (1.36), Section 1.7
ØMG1 (5, 5)	Ω_2	As ØMG0 (5, 5) for the 1st harmonic
ØMGN (5, 5)	Ω_3	As ØMG0 (5, 5) for the terminal boundary

Ⓟ

P (50, 50)	P	Matrix of Equation (1.60), Section 1.10
PANDX		Sub-executive subroutine which directs the formation of the various matrices needed in computing the P and X matrices of Equation (1.60), Section 1.10. This subroutine calls the RSLT, FGHPE, DYLMN and GMTX subroutines, the latter two in the Dynamics version only.
PE (50)	p	Force matrix of Equation (1.34), Section 1.7

PEL		Meridional distance to a station from the initial station
PFCN		Deck name for the Statics version of subroutine PPPN
PFE	p_ξ	Fourier component for load in the meridional direction
PFLAG		Print flag, read with GDA, general data. See Section 3.3.2.
PHI0	ϕ_o	Initial opening angle from vertical axis for sphere or toroid
PHIN	ϕ_N	Final opening angle from vertical axis for sphere or toroid
PHI (THETA)	Φ_θ	Rotation in the circumferential direction
PHI (XI)	Φ_ξ	Rotation in the meridional direction
PHS		Program name for PHI (XI)
PHT		Program name for PHI (THETA)
PN	p_ζ	Fourier component for load in the normal direction
PØI		Poisson's ratio
PØIS1		ENTRY point to function subprogram, EI1. Sets up Poisson's ratio for use in the stress calculations for the first layer.
PØIS2		As PØIS1 for the second stresses
PPPF		ENTRY point to function subprogram, PPPN. Defines the meridional pressure.
PPPH		As PPPF for circumferential pressure

PPPN		FUNCTIØN subprogram for pressure loadings. See Section 1.5 for sign convention.
PTH	p_θ	Fourier component for load in the circumferential direction
Q		
Q (THETA)	Q_θ	Transverse force per unit length in the circumferential direction
Q (XI)	Q_ξ	Transverse force per unit length in the meridional direction
R		
R	r	Normal distance from axis to shell
R (50, 50)	R	Boundary matrix of Equation (1.46), Section 1.8
RA1		Radius of cone or cylinder at station 1
RC		Radius of curvature of sphere or toroid
RCURV		Input values of meridional radius of curvature
RCURZ		Input values of circumferential radius of curvature
RHØX	ρ	R/A0
RIPT		Discrete radii for general shell shape
RØFF		Offset distance of center of curvature from axis of revolution, for toroids
RSLT		Subroutine which computes the boundary matrices for the P and X matrices

Ⓢ

S (50, 50)	S	Boundary matrix of Equation (1.46), Section 1.8
SCB1, 2, 3	B_m	Membrane stiffness coefficients
SCD1, 2, 3	D_m	Bending stiffness coefficients
SCG1, 2, 3	G_m	Shear stiffness coefficients
SCG13	G_{13}	Shear twist stiffness coefficient
SCMTSI	$M^T_{\xi(n)}$	Meridional thermal moment coefficients
SCMTTH	$M^T_{\theta(n)}$	Circumferential thermal moment coefficients
SCNTSI	$t^T_{\xi(n)}$	Meridional thermal load coefficients
SCNTTH	$t^T_{\theta(n)}$	Circumferential thermal load coefficients
SIG0	σ_o	Reference stress
SIG (THETA)	σ_θ	Circumferential stress
SIG (THETA, ETA)	$\tau_{\theta\zeta}$	Circumferential transverse shear stress
SIG (XI)	σ_ξ	Meridional stress
SIG (XI. ETA)	$\tau_{\xi\zeta}$	Meridional transverse shear stress
SIG (XI, THETA)	$\sigma_{\xi\theta}$	In-plane shear stress
SUMS		Subroutine which does the Fourier summing for the deflections and rotations, computes the internal loads and, in the Static version, stresses

Ⓣ

TCTR		Time cycle number

TFCN		Deck name for the Statics version of function subprogram, ENTT
TFE		Current temperature load at a given meridional station and for k harmonics
THT		Circumferential angle THETA (degrees) at which print-outs of results are desired. (5 permitted)
TMP1, 2		FUNCTIØN subprograms used to define the temperature for the two layers at which stresses are desired
TTF1, 2		Deck names for function subprograms, TMP1 and TMP2, respectively
TTP		First derivative of the temperature loads
TU	t	Current time
Ⓤ		
U	u_ξ	Meridional displacement; see Figure 1.2
Ⓥ		
V	u_θ	Circumferential displacement; see Figure 1.2
Ⓦ		
W	w	Normal displacement; see Figure 1.2
W (THETA)	ω_θ	Circumferential curvature, print heading
W (XI)	ω_ξ	Meridional curvature, print heading
WDØT	$\frac{\partial w}{\partial t}$	Velocities
WFEPX	$\frac{\partial \omega_\xi}{\partial \xi}$	First derivative of meridional curvatures

WFEX	ω_ξ	Program name for W (XI)
WTHX	ω_θ	Program name for W (THETA)
⊗		
X (50)	x	Matrix of Equation (1.62), Section 1.10
XI	ξ	Meridional distances to stations computed in GEOM
XIPT		Discrete XI distances or arc lengths array entered with GMDA
Ⓩ		
Z (50)	z	Solution matrix, Equation (1.38), Section 1.7
ZMTRX		Subroutine which solves for the Z matrix using the P and X matrices
ZP		Solutions for previous time cycle
ZPP, ZPPP		Solution matrices for the second and third previous cycles
ZTA	ζ	Circumferential distance to a station
ZX (50)		Another name for Z (50)