SLOSHING OF WATER IN TORUS
PRESSURE-SUPPRESSION POOL OF BOILING WATER REACTORS
UNDER EARTHQUAKE GROUND MOTIONS

A Report of an Analytical and Experimental Study
of Sloshing Response in Axisymmetric Tanks
Under Earthquake Ground Motions

by

M. Aslam
Associate Development Engineer
Department of Civil Engineering, University of California
Berkeley, California 94720

W. G. Godden
Professor of Civil Engineering
University of California
Berkeley, California 94720

and

D. T. Scalise
Engineering Sciences Department Head
Lawrence Berkeley Laboratory, University of California
Berkeley, California 94720

NOTICE

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Department of Energy, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

August 1978

Fig

CONTENTS

LIST OF SYMBOLS.	v
ABSTRACT	vi
1. INTRODUCTION.	1-1
1.1 Objective.	1-1
1.2 Scope of the Investigation	1-2
2. TORUS TANK MODEL TESTS AND COMPARISON WITH APPROXIMATE ANALYSIS	2-1
2.1 Introduction	2-1
2.2 Model Description and Instrumentation.	2-1
2.3 Test Procedure and Experimental Data	2-2
2.3.1 Tests on small shaking table	2-2
2.3.2 Tests under simulated earthquakes.	2-3
2.4 Comparison of Test Results with Approximate Theory	2-5
2.4.1 Comparison of natural sloshing frequencies	2-5
2.4.2 Comparison of sloshing displacements under steady state harmonic ground motion.	2-6
2.4.3 Comparison of sloshing displacements under simulated earthquake ground motions.	2-6
2.5 Dynamic Pressures.	2-7
2.6 Linearity Range.	2-8
2.7 Summary of Important Observations on Sloshing in Torus Tank Model	2-8
3. FINITE ELEMENT ANALYSIS OF EARTHQUAKE INDUCED SLOSHING IN AXISYMMETRIC TANKS	3-1
3.1 Introduction	3-1
3.2 Mathematical Formulation	3-2
3.2.1 Equations of motion.	3-2
3.2.2 Boundary conditions.	3-3
3.3 Finite Element Formulation	3-4
3.3.1 Derivation of finite element equations	3-4
3.3.2 Isoparametric formulation for axisymmetric tank under arbitrary horizontal ground motion	3-7
3.3.3 Free surface (mass) matrix for axisymmetric element.	3-12
3.3.4 Evaluation of fluid (stiffness) matrix for axisymmetric element	3-13
3.3.5 Load vector for axisymmetric element	3-14

3.4	Numerical Solution of Finite Element Equations	3-15
3.4.1	The Newmark-Wilson algorithm for linear step-by-step integration	3-17
3.5	Computer Program 'SLOSH2'	3-18
3.6	Sample Analyses and Comparison with Test Data from Annular and Torus Tank Models.	3-20
3.6.1	Annular tank model	3-20
3.6.2	Torus tank model	3-21
3.7	Sample Analysis of Prototype Torus Tank under El Centro 1940 Earthquake.	3-22
4.	CONCLUSIONS	4-1
	ACKNOWLEDGMENTS	A-1
	REFERENCES	R1
	APPENDIX A1	A1-1
	APPENDIX A2	A2-1

LIST OF SYMBOLS

a	Outer radius of tank
b	Inner radius of tank
R1	Liquid-solid interface boundary
B2	Free surface boundary
EB1	Element boundary B1
EB2	Element boundary B2
EV	Element Volume
<u>F</u>	Load vector (Solid-liquid interface matrix)
g	Acceleration of gravity
h	Water depth
<u>J</u>	Jacobian matrix
<u>K</u>	Fluid matrix (equivalent to stiffness matrix)
<u>M</u>	Free surface matrix (equivalent to mass matrix)
N	Shape functions or a number
p	Pressure
r	Local coordinate
s	Local coordinate or surface
t	Time
v_n	Normal velocity
X,Y,Z	Rectangular coordinates
ϕ	Velocity potential
$\dot{\phi}$	$d\phi/dt$
Δt	Time step
ρ	Mass density
δ	Free surface displacement or a parameter

ABSTRACT

This report presents an analytical and experimental investigation into the sloshing of water in torus tanks under horizontal earthquake ground motions. This study was motivated because of the use of torus tanks for pressure-suppression pools in Boiling Water Reactors. Such a pressure-suppression pool would typically have 80 ft and 140 ft inside and outside diameters, a 30 ft diameter section, and a water depth of 15 ft.

A general finite element analysis was developed for all axisymmetric tanks and a computer program was written to obtain time-history plots of sloshing displacements of water and dynamic pressures. Tests were carried out on a 1/60th scale model under sinusoidal as well as simulated earthquake ground motions. Tests and analytical results regarding natural frequencies, surface water displacements, and dynamic pressures were compared and a good agreement was found within the range of displacements studied. The computer program gave satisfactory results within a maximum range of sloshing displacements in the full-size prototype of 30 in. which is greater than the value obtained under the full intensity of the El Centro earthquake (N-S component 1940). The range of linear behavior was studied experimentally by subjecting the torus model to increasing intensities of the El Centro earthquake. The general computer program was also used for comparison with a previous study on the sloshing of water in annular tanks, and the previous annular tank solution was also used as an approximate solution in the torus tank

problem showing that sloshing response is not very sensitive to the precise cross-sectional geometry of an axisymmetric tank.

Tests were also conducted to study the effect of vertical ground motions on the dynamic response of the fluid. These showed a negligible effect on sloshing displacements.

KEY WORDS

Sloshing Response, Pressure-suppression pool, Torus tanks, Annular tanks, Earthquake response, Dynamic pressures, Finite element solution.

1. INTRODUCTION

1.1 Objective

This report presents the results of a study into the sloshing response of water in torus tanks under the action of earthquake-induced ground motions, and is a continuation of a previous investigation into the sloshing response of water in annular tanks [1].

Torus tanks are used as pressure-suppression pools in certain designs of boiling water reactors (BWR) (e.g., General Electric Mark I), and a knowledge of the dynamic response of the water and particularly of the resulting water surface elevations is important in evaluating the effectiveness of the system under earthquake conditions. A typical torus suppression pool as used in the GE Mark I reactor has an outside diameter of 140 ft and a section height of 30 ft. A 1/60th scale idealized model of such a pool is shown in Fig. 2-1a.

Rather than deriving a particular analytical solution to the torus tank problem, it was decided at the outset to undertake a general study into the sloshing of water in axisymmetric tanks; this would be applicable to the torus tank, the annular tank previously studied, and to all tanks with a constant section of revolution. References [1] through [16] are some of the previous studies related to the sloshing of fluids in tanks.

1.2 Scope of the Investigation

This study includes the testing of a 1/60th scale model of GE Mark 1 torus, the development of a general finite element theory for axisymmetric tanks, and a computer code to implement the finite element theory. A comparison of test results with an approximate analysis based on the annular tank theory is also given in Chapter 2.

The test model was constructed by cementing together wedge-shaped lengths of 6 in. lucite tubing as shown in Fig. 2-1. This was tested under harmonic and simulated earthquake ground motions. The quantities measured included the sloshing frequencies and free surface displacements.

The finite element equations (Chapter 3) were derived using the Galerkin principle and a linearized small displacement theory was used. The velocity potential ϕ was taken as the field variable and the sloshing displacements as well as impulsive pressures were derived from ϕ . The finite element equations were first derived for a general three dimensional sloshing problem under arbitrary ground motions, and then it was specialized to an axisymmetric tank subjected to horizontal earthquake ground motions only.

A computer code named 'SLOSH2' was developed to implement the finite element theory, and a comparison of sloshing displacements and pressures as predicted by the computer program was made with the test results from annular [1] and torus tanks. These comparisons show that the finite element program can successfully predict the sloshing displacements as well as impulsive pressures in an axisymmetric tank under horizontal ground motions within the range of linear behavior.

2. TORUS TANK MODEL TESTS AND COMPARISON WITH APPROXIMATE ANALYSIS

2.1 Introduction

The objective of the tests on the torus tank model was to obtain experimental data to compare with the results obtained from approximate analysis based on a previous study on annular tanks [1], and also to check the accuracy of the finite element solution developed and described in Chapter 3.

Test data on sloshing displacements was obtained for harmonic as well as simulated seismic-type ground motions. Details of the test procedure, experimental data, and comparison with results of approximate analysis based on annular tank theory are discussed in this chapter.

2.2 Model Description and Instrumentation

A simplified 1/60th scale model of a Mark 1 suppression pool was constructed as shown in Figs. 2-1a and 2-1b. The model was fabricated from 16 short lengths of 6 in. diameter clear plastic tubing cemented together to approximate complete torus. The internal details of the prototype, including the 'Headers' and 'Downcomers', were not reproduced in the model and not taken into account in the analysis. However, for reference purposes, the Header and Downcomer configuration is shown in Fig. 2-1c for a 1/30th scaled model designed for proof-tests by the reactor manufacturer. The mean diameter of the 1/60th scale model was 22 in. The normal operating water depth was 3 in.; that is, the water

surface at the section diameter, though water surface elevations above and below this level were also studied. The tank was mounted on a plywood base which was in turn prestressed to the shaking table.

The model was instrumented with one displacement gage located at a distance of $3/8$ in. from the inside wall. This gage was of the same type as used in the annular tank tests [1].

2.3 Test Procedure and Experimental Data

2.3.1 Tests on the small shaking table

Sloshing frequencies of the $1/60$ th scale model of the Mark 1 torus and the steady state sloshing response under sinusoidal table motions were measured using the 3×4 ft shaking table described in Ref. [1] Sec. 4.4. The test set-up is shown in Fig. 2-2 and is similar to that described in Ref. [1] Sec. 4.5, the only difference being that this time a spectrum analyzer was used to determine the sloshing frequencies.

The test procedure to determine the steady state sloshing response under sinusoidal motions was the same as described in Ref. [1] Sec. 4.5 and was measured at the gage location shown in Fig. 2-1a. To determine the sloshing frequencies, the frequency of the table motion was continuously changed and the water displacement signal was fed to the spectrum analyzer which was arranged to produce an averaged frequency spectrum. In this way the sloshing frequencies were read directly.

Tests on the small shaking table were not only conducted at the normal depth of 3 in., but also at depths of 2, 2.5, 3.5 and 4 in.

Test data regarding the sloshing frequencies and the wave heights are presented in Tables 2-1 and 2-2 respectively. Table 2-1 shows the mode number, the depth of water, and the measured as well as the analytical sloshing frequencies as approximated by using annular tank theory. Table 2-2 gives the steady-state response for harmonic ground motion, and tabulates the frequency of table motion, depth of water in the tank, amplitude of the table acceleration, and measured as well as the predicted values of the amplitudes of wave heights at the gage location. Two approximate solutions are given based on annular tank theory and are described in Sec. 2.4.

2.3.2 Tests under simulated earthquakes

Tests under simulated earthquake motion using the El Centro (1940) record were made at the University of California's Earthquake Simulator Laboratory. The test set-up of the 1/60th scale model of the Mark 1 torus is shown in Fig. 2-3 where the model is shown mounted on the 20 x 20 ft shaking table. Details of the shaking table facility and data acquisition system are given in Ref. [1] Chapter 5. The testing procedure was similar to that described in Ref. [1] Sec. 5.6. The prototype of the El Centro earthquake record was reduced by a factor of $\sqrt{60}$ to meet the similitude requirements of a 1/60th scale model.

In each test the quantities that were recorded, digitized and stored on magnetic tape included the horizontal and vertical table accelerations and displacements, and the water surface displacements in the model at the gage location shown in Fig. 2-1. The peak values of these quantities together, with the test numbers are shown in Table 2-3. The maximum and minimum water displacements in this table represents the

peak upward and downward displacements respectively. These tests were carried out both with and without the vertical component of the recorded ground motion. They were made at increasing amplitudes of ground motion in order to determine the range of linear response. Some selected results are plotted in Figs. 2-4 through 2-8. The center graph in Figs. 2-4 through 2-7 also shows a comparison of the measured displacement with approximate results from annular tank theory under horizontal ground motion only. Figure 2-8 shows that the vertical ground motion alone does not produce sloshing displacements as could be expected.

It should be remembered that a shaking table acts as a low-pass filter and will not fully reproduce motions at frequencies above the natural frequency of the system. The small time scale ($T_r = \sqrt{60} = 7.75$) used in this study resulted in the acceleration peaks associated with high frequency ground motion being filtered out by the table. Hence the intensity of the earthquake given in Table 2-3, and measured by peak table acceleration, cannot be compared directly with the intensity of the original El Centro record. However the sloshing response is produced mainly by the low frequency components of the ground motion and these are correctly reproduced. It is estimated that the simulated earthquake ground motion in Table 2-3 with a peak acceleration of 0.34 g is equivalent to approximately 2.0 times the actual intensity of the El Centro earthquake in the significant low frequency components.

(Note: The recorded peak acceleration in the actual 1940 El Centro earthquake is 0.32 g.)

2.4 Comparison of Test Results with Approximate Theory

Approximate analyses were carried out using the computer program 'SLOSH' which is based on the annular tank theory developed in Ref. [1]. For carrying out the analyses the measured table acceleration was used and the tank radii a and b were taken as the horizontal distances from the axis of symmetry to the outer and inner walls where the free water surfaces come in contact with the solid boundaries. Two approximations were then made for the water depth as follows:

(1) In the first approximation the depth of the water (h) in the torus tank solution was taken as the actual depth from the bottom of the torus to the free surface. This is called the 'Annular Tank' approximation in Tables 2-1 and 2-2.

(2) In the second approximation, the depth of water used in the annular solution was adjusted to make the volume of water in the annular tank (with the same values of a and b as the torus tank) equal to the actual volume in the torus. This is called the 'Equivalent Volume' approximation in Tables 2-1 and 2-2.

Using these two approximations, analyses were carried out using the SLOSH program and the comparison of test and computer results for the sloshing frequencies and displacements is given.

2.4.1 Comparison of natural sloshing frequencies

Table 2-1 shows the comparison between the approximate results based on annular tank theory and the test values. It can be seen that for the first four modes, and within the range of water depth considered, the two approximations give very similar results. This could be expected

as in an annular tank over this range the natural frequencies are not very sensitive to water depth. Also, the approximate solutions compare well with the measured torus values in the 2 in. to 4 in. depth range indicating that frequencies are not very sensitive to the actual shape of cross-section of the tank.

2.4.2 Comparison of sloshing displacements under steady state harmonic ground motion

Table 2-2 gives a comparison of the measured and computed results from approximate annular tank theory for water depths ranging from 2.5 in. to 3.5 in. under steady state sinusoidal table motions varying in frequency from 1.50 Hz to 2.55 Hz. It can be observed in Table 2-2 that the annular tank theory gives satisfactory results for this range of water depth and the 'Equivalent Volume' approximation gives better results in most cases. It was also found that when the torus tank is nearly empty or nearly full (water depths less than 2 in. or greater than 4 in. in the model) as could be expected the approximate theory does not give satisfactory results and should not be used.

2.4.3 Comparison of sloshing displacements under simulated earthquake ground motions

Figures 2-4 through 2-7 show the time-history plots of the water surface displacements at a distance of $3/8$ " from the inside wall under increasing intensities of the simulated El Centro earthquake motion (horizontal component only). The depth of water in the torus tank was 3 in. in all these tests and the comparison of measured and approximate analytical displacements is given in the middle plot in each case. In Fig. 2-4

the analysis was carried out using approximation (1) (i.e., depth of water in the annular solution was taken as 3 in.). It can be seen that the measured and computed results differ by about 30% in displacement amplitude, and the measured frequency of response oscillation is somewhat lower. In Fig. 2-5 the 'Equivalent Volume' approximation was used ($h = 2.36$ in.) for the same ground motion and it can be seen that the agreement between the test and approximate analysis is better as regards displacement amplitude.

Figures 2-6 and 2-7 show a comparison at higher intensities of the ground motion where the analysis in both cases was done using the 'Equivalent Volume' annular tank approximation. It can be seen that even at these relatively large displacements the approximate theory, where a modified depth based on an equivalent volume is used in the annular tank solution, gives reasonably satisfactory results. This approximate theory should however be used with caution especially when the water depth is outside the range of that used in these tests.

2.5 Dynamic Pressures

The dynamic pressures were not measured on account of their small values and therefore no comparison is available between measured and analytical results. It is however anticipated that the annular tank theory should not be expected to give satisfactory results for the dynamic impulsive pressures in a torus tank and should not be used for this purpose. If the pressures are required in the torus, the Finite Element Method described in the next chapter should be used.

2.6 Linearity Range

The range of linear behavior of sloshing response was tested experimentally by subjecting the model to increasing intensities of the simulated El Centro earthquake (horizontal component only), and measuring the maximum (upward) and minimum (downward) peak water surface displacements. The displacements associated with this ground motion are primarily in the first radial mode, and hence the following comments on linearity are primarily related to displacements in this mode. The results are plotted in Fig. 2-9 and are tabulated in Table 2-3, together with some tests which included the vertical component of ground motion.

For displacements less than 0.1 in. in the model, linearity holds within 1%. But as the amplitude of displacement increases, nonlinearity also increases and becomes approximately 10% at displacements in the order of 0.4 in. estimated on the basis of both upward and downward displacements. For practical purposes it may be assumed that the linear theory gives satisfactory results as long as the displacements are less than 0.5 in. in the model (or 30 in. in the prototype Mark I suppression pool). It may be seen in Fig. 2-6 that the modified annular tank theory (Equivalent Volume approximation) gives quite satisfactory results although the displacements are of the order of 0.5 in.

2.7 Summary of Important Observations on Sloshing in Torus Tank Model

1) The overall sloshing behavior of water in the torus tank model studied in this report was very similar to that in the annular tank described in Ref. [1].

2) The effect of vertical ground motion on the sloshing displacements is negligible in the linear range (compare Test No. 220378.2 with Test No. 220378.10 in Table 2-3) but becomes more significant in the non-linear range (compare Test No. 220378.6 with Test No. 220378.13 in Table 2-3).

3) Modified annular tank theory gives satisfactory results for sloshing displacements in the torus tank. However this approximate theory cannot be applied for the determination of dynamic pressures.

4) A non-linearity of approximately 10% could be expected at displacements of the order of 0.4 in. in the model (or 24 in. in the prototype). Linear theory gives satisfactory results even under strong ground motions such as the El Centro earthquake of 1940.

TABLE 2-1. Natural sloshing frequencies of small scale model (1/60-scale model).

Mode No.	Depth of Water (in.)	Natural Frequencies (Hz)		
		Test	Theory	
			Annular Tank	Equivalent Volume
1	3	0.45	0.49	0.44
2	3	2.15	2.18	2.12
3	3	3.02	3.20	3.19
4	3	3.95	3.93	3.92
1	4	0.55	0.56	0.52
2	4	2.37	2.34	2.31
3	4	3.15	3.31	3.30
4	4	4.15	4.04	4.04
1	2	0.35	0.40	0.34
2	2	2.00	2.13	1.91
3	2	3.20	3.27	3.16
4	2	3.92	4.04	4.00

TABLE 2-2. Sloshing response of water in torus tank under sinusoidal ground acceleration (1/60th scale model) a = 14", b = 8".

Frequency (Hz)	Depth (in.)	Table Acceleration (g)	Displacement at Inner Wall (in.)		
			Test	Theory	
				(Equivalent Volume)	Annular Tank
1.5	3.0	.0109	.069	.069	.0685
1.8	3.0	.00392	.040		
1.8	3.0	.00785	.079		
1.8	3.0	.0118	.119		
1.8	3.0	.0157	.157	.157	.139
1.8	3.0	.0196	.179		
1.9	3.0	.00438	.058		
1.9	3.0	.00875	.116		
1.9	3.0	.0131	.179		
1.9	3.0	.0175	.240	.240	.190
1.9	3.0	.0218	.271		
2.4	3.0	.0242	.246	.205	.319
1.50	2.5	.0118	.083	.078	.073
1.50	2.5	.0233	.165		
1.80	2.5	.0163	.216	.210	.148
1.80	2.5	.0326	.444		
1.80	2.5	.0245	.330		
1.50	3.5	.0118	.072		.074
1.80	3.5	.0163	.150	.140	.132
1.80	3.5	.0326	.310		
2.55	3.5	.0325	.252	.226	.275

TABLE 2-3. Extreme values in torus tank model tests under simulated El Centro 1940 earthquake (time scale ($= \sqrt{60} = 7.75$, depth of water = 3 inches).

Test No	Peak Table Acceleration (g)		Peak Table Displacement (inches)		Water Displacement (inches)	
	Horizontal	Vertical	Horizontal	Vertical	Maximum (up)	Minimum (down)
220378.2	0.115	0.0	0.048	0.0	0.246	0.231
220378.3	0.164	0.0	0.073	0.0	0.409	0.346
220378.4	0.237	0.0	0.100	0.0	0.614	0.458
220378.5	0.284	0.0	0.125	0.0	0.826	0.563
220378.6	0.338	0.0	0.149	0.0	0.999	0.667
220378.7	0.049	0.0	0.023	0.0	0.109	0.111
220378.8	0.0	0.083	0.0	0.05	0.003	0.006
220378.9	0.0	0.260	0.0	0.160	0.014	0.012
220378.10	0.107	0.067	0.046	0.036	0.249	0.245
220378.11	0.170	0.103	0.075	0.063	0.485	0.378
220378.12	0.232	0.163	0.100	0.098	0.758	0.552
220378.13	0.358	0.217	0.151	0.136	1.26	0.924
230378.1	0.477	0.0	0.201	0.0	1.22	0.921

XBL 789-2279

FIG.2-1a 1/60 SCALE MODEL OF MARK I PRESSURE SUPPRESSION POOL

XBB 785-5492

FIG.2-1b CONSTRUCTION OF 1/60 SCALE MODEL OF MARK I TORUS
PRESSURE SUPPRESSION POOL

11.5" ID, 0.5" WALL

PLASTIC TUBING, 16 SECTIONS

XBL 789-2280

FIG. 2-1c 1/30 SCALE MODEL OF MARK I SUPPRESSION POOL
TORUS WITH HEADERS AND DOWNCOMERS

CBB 779-8988

FIG.2-2 1/60 SCALE MODEL OF TORUS TANK ON SMALL SHAKING TABLE

XBB 785-5344

FIG.2-3 1/60 SCALE MODEL ON 20ft X 20ft SHAKING TABLE

FIG. 2-4 SLOSHING RESPONSE OF WATER IN A TORUS TANK (INNER RADIUS = 8 IN., OUTER RADIUS = 14 IN., DEPTH OF WATER = 3 IN.) UNDER SIMULATED EL CENTRO 1940 EARTHQUAKE, TIME SCALE = 7.7, TEST NO. 220378.2. PEAK SHAKING TABLE ACCELERATION = 0.115g HORIZONTAL, 0.0g VERTICAL

XBL 789-11309

FIG. 2-5 SLOSHING RESPONSE OF WATER IN A TORUS TANK (INNER RADIUS = 8 IN., OUTER RADIUS = 14 IN., DEPTH OF WATER = 3 IN.) UNDER SIMULATED EL CENTRO 1940 EARTHQUAKE, TIME SCALE = 7.7, TEST NO. 220378.2. PEAK SHAKING TABLE ACCELERATION = 0.115g HORIZONTAL, 0.0g VERTICAL.

FIG. 2-6 SLOSHING RESPONSE OF WATER IN A TORUS TANK (INNER RADIUS = 8 IN., OUTER RADIUS = 14 IN., DEPTH OF WATER = 3 IN.) UNDER SIMULATED EL CENTRO 1940 EARTHQUAKE, TIME SCALE = 7.7, TEST NO. 220378.4. PEAK SHAKING TABLE ACCELERATION = 0.237g HORIZONTAL, 0.0g VERTICAL.

FIG. 2-7
SLOSHING RESPONSE OF WATER IN A TORUS TANK (INNER RADIUS = 8 IN., OUTER RADIUS = 14 IN., DEPTH OF WATER = 3 IN.) UNDER SIMULATED EL CENTRO 1940 EARTHQUAKE. TIME SCALE = 7.7. TEST NO. 220378.6. PEAK SHAKING TABLE ACCELERATION = 0.338g HORIZONTAL, 0.0g VERTICAL.

FIG. 2-8 SLOSHING RESPONSE OF WATER IN A TORUS TANK (INNER RADIUS = 8 IN., OUTER RADIUS = 14 IN., DEPTH OF WATER = 3 IN.) UNDER SIMULATED EL CENTRO 1940 EARTHQUAKE, TIME SCALE = 7.7, TEST NO. 220378.9. PEAK SHAKING TABLE ACCELERATION = 0.0g HORIZONTAL, 0.260g VERTICAL.

FIG. 2-9 VARIATION OF PEAK WATER DISPLACEMENTS WITH INTENSITY OF SIMULATED EL CENTRO 1940 EARTHQUAKE GROUND MOTION, TIME SCALE $\sqrt{60}$

XBL 789-2276

3. FINITE ELEMENT ANALYSIS OF EARTHQUAKE INDUCED SLOSHING IN AXISYMMETRIC TANKS

3.1 Introduction

The finite element analysis [18] has become a powerful tool in solving complex engineering problems. Since the finite element method is completely general, it was decided that instead of looking for a closed form solution to predict the sloshing displacements and hydrodynamic pressures due to earthquake ground motions in a torus tank, the finite element method would be a better alternative in that it would be more general and thus can be applied to tank shapes other than cylindrical and toroidal.

Previous work on the finite element analysis of sloshing in tanks was done by Edwards [19] in which the shell theory was used for the prediction of seismic stresses and displacements in a cylindrical tank filled with liquid, but the sloshing was not considered in this analysis.

Finite element analysis for liquid sloshing problems by Luck [20] gives only the mode shapes and frequencies in an elastic container. His analysis is based on the variational principle suggested by Tong [21].

In this investigation our main concern is to study the sloshing effects in pressure-suppression pools of boiling water reactors, namely the Mark I torus and the Mark III annular suppression pools. Such structures can be considered as effectively rigid for the sloshing problem and thus the coupled effect of water-structure interaction is neglected in this analysis. Also the nonlinear sloshing problem has been linearized [13] for this analysis.

The finite element equations were first derived for a completely general three dimensional problem and then were specialized to an axisymmetric tank subjected to arbitrary horizontal ground motions. The finite element equations were derived using the Galerkin principle [22]. More background information on Galerkin method may be found in References [18] and [23-26].

3.2 Mathematical Formulation:

3.2.1 Equation of motion

Consider a tank of arbitrary shape with rigid walls filled with a liquid whose free surface area is B_2 as shown in Fig. 3-1. B_1 represents the surface area of liquid in contact with the solid boundary of the container. V is the volume of the liquid and δ is the surface water displacement with respect to the undisturbed liquid surface. Using the same assumptions as in Ref. [1] Sec. 2.2, the velocity potential ϕ exists at every point in V and must satisfy the Laplace equation which in rectangular coordinates can be written as:

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} + \frac{\partial^2 \phi}{\partial z^2} = 0 \quad (3-1)$$

where $\phi = \phi(x, y, z, t)$.

Equation (3-1) will be solved by the finite element method subject to the appropriate time dependent boundary conditions as specified below.

3.2.2 Boundary conditions

Let $v_n(t)$ be the velocity of the tank wall along its outward normal to the boundary at any point, then:

$$\frac{\partial \phi}{\partial n} = v_n(t) \quad \text{on B1} \quad (3-2)$$

where n is the outward normal to the solid boundary and v_n is a function of time t .

It can also be shown that a liquid particle on the free surface B2 must satisfy the following two conditions [13]

$$\frac{\partial \phi}{\partial x} \frac{\partial \delta}{\partial x} - \frac{\partial \phi}{\partial z} + \frac{\partial \phi}{\partial y} \frac{\partial \delta}{\partial y} + \frac{\partial \delta}{\partial t} = 0 \quad (3-3)$$

and

$$g\delta + \frac{\partial \phi}{\partial t} + \frac{1}{2} \left[\left(\frac{\partial \phi}{\partial x} \right)^2 + \left(\frac{\partial \phi}{\partial y} \right)^2 + \left(\frac{\partial \phi}{\partial z} \right)^2 \right] = 0 \quad (3-4)$$

where g is the acceleration of gravity.

Equations (3-3) and (3-4) which represent the non-linear free surface boundary conditions can be simplified and combined into one boundary condition by neglecting higher order terms and eliminating δ . This single linearized boundary condition [13] can be written as follows.

$$\frac{1}{g} \frac{\partial^2 \phi}{\partial t^2} + \frac{\partial \phi}{\partial z} = 0 \quad \text{on B2.} \quad (3-5)$$

3.3 Finite Element Formulation

3.3.1 Derivation of finite element equations

In the finite element analysis, the continuum is divided into discrete elements or subregions which are interconnected at a finite number of points called nodes. The necessary formulation follows the Galerkin principle where we let the unknown field variable ϕ , throughout the solution domain, be approximated as

$$\phi = \sum_{j=1}^N N_j(x, y, z) \phi_j(t) \quad (3-6)$$

in which N_j are the shape functions defined piecewise, element by element, and $\phi_j(t)$ are the time dependent nodal values of the field variable i.e., the velocity potential in this case. In the summation process an appropriate function for the particular point in space must be used. The N nodal values ϕ_j are obtained by solving a set of N simultaneous equations each derived by equating the boundary and interior residuals calculated by multiplying with a weighting function and integrating over the domain. In the Galerkin approach, the shape functions are taken as the weighting functions and for a typical node i substituting Eq. (3-6) into Eqs. (3-1), (3-2) and (3-5) and equating the weighted and integrated interior and boundary residual, we have:

$$\begin{aligned} \int_v N_i \left\{ \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \right\} \sum_{j=1}^N N_j \phi_j dv = \int_{B1} N_i \frac{\partial}{\partial n} \sum_{j=1}^N N_j \phi_j ds \\ - \int_{B1} N_i v_n ds + \int_{B2} \left\{ \frac{N_i}{g} \frac{\partial^2}{\partial t^2} \sum_{j=1}^N N_j \phi_j + N_i \frac{\partial}{\partial z} \sum_{j=1}^N N_j \phi_j \right\} ds \end{aligned} \quad (3-7)$$

in which $\int dv$ and $\int ds$ represent the integrals over the volume and appropriate surfaces respectively. Consider the first term on the left hand side of Eq. (3-7) and write it in the following form

$$\int_v N_i \sum_1^N \frac{\partial^2 N_j}{\partial x^2} \phi_j dv = \int_v \frac{\partial}{\partial x} \left(N_i \sum_1^N \frac{\partial N_j}{\partial x} \right) \phi_j dv - \int_v \frac{\partial N_i}{\partial x} \sum_1^N \frac{\partial N_j}{\partial x} \phi_j dv. \quad (3-8)$$

Applying the Divergence theorem on the first integral on the right hand side of Eq. (3-8), we can rewrite it in the following form.

$$\int_v N_i \sum_1^N \frac{\partial^2 N_j}{\partial x^2} \phi_j dv = \int_B N_i \sum_1^N \frac{\partial N_j}{\partial x} \ell_x \phi_j ds - \int_v \frac{\partial N_i}{\partial x} \sum_1^N \frac{\partial N_j}{\partial x} \phi_j dv \quad (3-9)$$

in which $B = B_1 + B_2$ and ℓ_x is the direction cosine in the x-direction of the outward normal n . Similar expression can be written for the second and third terms in Eq. (3-7). Substituting Eq. (3-9) and similar expressions into the right hand side of Eq. (3-7), we get

$$\begin{aligned} & \int_B N_i \left(\sum_1^N \frac{\partial N_j}{\partial x} \ell_x \phi_j + \sum_1^N \frac{\partial N_j}{\partial y} \ell_y \phi_j + \sum_1^N \frac{\partial N_j}{\partial z} \ell_z \phi_j \right) ds \\ & - \int_v \left(\frac{\partial N_i}{\partial x} \sum_1^N \frac{\partial N_j}{\partial x} \phi_j + \frac{\partial N_i}{\partial y} \sum_1^N \frac{\partial N_j}{\partial y} \phi_j + \frac{\partial N_i}{\partial z} \sum_1^N \frac{\partial N_j}{\partial z} \phi_j \right) dv \quad (3-10) \\ & = \int_{B_1} N_i \sum_1^N \frac{\partial N_j}{\partial n} \phi_j ds - \int_{B_1} N_i v n ds + \frac{1}{g} \int_{B_2} N_i \sum_1^N N_j \ddot{\phi} ds + \int_{B_2} N_i \sum_1^N \frac{\partial N_j}{\partial z} \phi_j ds \end{aligned}$$

in which $\ddot{\phi} = d^2\phi/dt^2$.

The boundary integral on the left hand side of Eq. (3-10) can be substituted by

$$\int_B N_i \sum_1^N \frac{\partial N_j}{\partial n} \phi_j ds \quad (3-11)$$

or by

$$\int_{B1} N_i \sum_1^N \frac{\partial N_j}{\partial n} \phi_j ds + \int_{B2} N_i \sum_1^N \frac{\partial N_j}{\partial n} \phi_j ds. \quad (3-12)$$

Replacing the boundary integrals of left hand side of Eq. (3-10) with Eq. (3-12) and using the approximation (small slopes)

$$\frac{\partial N_i}{\partial z} \approx \frac{\partial N_i}{\partial n} \quad \text{on } B2, \text{ Eq. (3-10)}$$

can be simplified to the following form.

$$\begin{aligned} \int_v \left[\frac{\partial N_i}{\partial x} \sum_1^N \frac{\partial N_j}{\partial x} \phi_j + \frac{\partial N_i}{\partial y} \sum_1^N \frac{\partial N_j}{\partial y} \phi_j + \frac{\partial N_i}{\partial z} \sum_1^N \frac{\partial N_j}{\partial z} \phi_j \right] dv \\ + \frac{1}{g} \int_{B2} N_i \sum_1^N N_j \ddot{\phi}_j ds = \int_{B1} N_i v_n ds \end{aligned} \quad (3-13)$$

or

$$\underline{\underline{M}} \ddot{\underline{\underline{\phi}}} + \underline{\underline{K}} \underline{\underline{\phi}} = \underline{\underline{F}} \quad (3-14)$$

in which the elements of $\underline{\underline{M}}$, $\underline{\underline{K}}$ and $\underline{\underline{F}}$ are given by

$$M_{ij} = \frac{1}{g} \sum \int_{EB2} N_i N_j ds \quad (3-15)$$

$$K_{ij} = \sum \int_{EV} \left(\frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} + \frac{\partial N_i}{\partial z} \frac{\partial N_j}{\partial z} \right) dv \quad (3-16)$$

$$F_i = \sum \int_{EB1} N_i v_n ds \quad (3-17)$$

where summation for M_{ij} covers only the elements on the free surface boundary and the integral is carried out on the free surface of each element EB2. Summation for K_{ij} covers the contribution of each element and EV is the element region. EB1 refers only to the elements which lie on the solid boundary, B1, and the loading term thus is associated with the elements that lie on the tank wall boundary.

The free surface matrix $\underline{M}$ and the fluid matrix $\underline{K}$ are comparable to the mass and stiffness matrices respectively used in structural dynamics. It is interesting to note that the free surface matrix $\underline{M}$ gets the contribution only from the free surface elements. $\underline{M}$ and $\underline{K}$ are symmetric matrices and Eq. (3-14) is a set of second order linear differential equations which can be solved either by direct integration or by mode superposition.

3.3.2 Isoparametric formulation for axisymmetric tank under arbitrary horizontal ground motions

The following analysis will be restricted to rigid tanks which are symmetrical about the z - axis and are subjected to arbitrary horizontal ground motions alone. Since Eq. (3-13) involves only the first derivatives of shape functions, a 4-node quadrilateral element with linear interpolation functions will satisfy the convergence requirements. However, the more recently developed 4-to-8 variable node isoparametric element [27] has greater flexibility in accommodating the curved boundaries and is convenient for numerical integration. Therefore a variable 4-to-8 node, 2-dimensional isoparametric element will be used in the present formulation. Such an element can be used to model an axisymmetric problem or a two dimensional problem such as sloshing in a rectangular tank.

Figure 3-2(a) shows such a 4-to-8 variable node element lying in the x - z plane where z is the axis of symmetry. Any of the mid-side nodes 5 through 8 may or may not be present and can be eliminated if desired. Such a curvilinear 4-to-8 node element can be obtained by using an isoparametric mapping from a bi-unit square which has a local r - s coordinate system as shown in Fig. 3-2(b). The local coordinates r and s vary between -1 and +1. The nodes 1 through 4 are the corner nodes and the nodes 5 through 8 are the mid-side nodes corresponding to Fig. 3-2(a). Node 1 has the coordinates (1,1). The mapping between the local coordinate system (r,s) and the global (x,z) coordinate system must be unique in order to carry out the transformations properly. The coordinate transformation between the bi-unit square and the curvilinear element is given by;

$$x_m(r,s) = \sum_{i=1}^8 h_i(r,s) x_{im} \quad (3-18)$$

$$z_m(r,s) = \sum_{i=1}^8 h_i(r,s) z_{im} \quad (3-19)$$

in which (x_{im}, z_{im}) are the global coordinates of node i in element m and h_i are the interpolation functions in local coordinates corresponding to node m . The interpolation functions h_i for any element m are defined as follows:

$$\begin{aligned}
h_1 &= \frac{1}{4} (1+r) (1+s) - \frac{1}{2} h_5 - \frac{1}{2} h_8 \\
h_2 &= \frac{1}{4} (1-r) (1+s) - \frac{1}{2} h_5 - \frac{1}{2} h_6 \\
h_3 &= \frac{1}{4} (1-r) (1-s) - \frac{1}{2} h_6 - \frac{1}{2} h_7 \\
h_4 &= \frac{1}{4} (1+r) (1-s) - \frac{1}{2} h_7 - \frac{1}{2} h_8 \\
h_5 &= \frac{1}{2} (1-r^2) (1+s) \\
h_6 &= \frac{1}{2} (1-r) (1-s^2) \\
h_7 &= \frac{1}{2} (1-r^2) (1-s) \\
h_8 &= \frac{1}{2} (1+r) (1-s^2).
\end{aligned} \tag{3-20}$$

Since a horizontal ground motion will excite only the antisymmetric modes (in the linearized case) of sloshing in an axisymmetric tank, we can approximate the distribution of the velocity potential within an element m in terms of the velocity potential at node 1 through 8 and the interpolation functions given by Eq. (3-20)

$$\phi_m(x, s, \theta, t) = \sum_{n=1}^{\infty} \sum_{i=1}^8 h_i(x, s) \cos n\theta \phi_{im}(t) \tag{3-21}$$

in which ϕ_{im} is the value of the velocity potential at node i of element m .

It is obvious that if this shape function given by Eq. (3-21) is used in Eq. (3-17) to calculate the loading vector, the integral between the limits 0 and 2π will be non-zero only when $n = 1$ in the case of horizontal ground motion only, because v_n in such a case varies as a function

of $\cos\theta$ and the $\int_0^{2\pi} \cos\theta \cdot \cos n\theta \, d\theta = 0$ for any $n \neq 1$. Therefore Eq. (3-21) can be written as

$$\phi_m(r, s, \theta, t) = \sum_{i=1}^8 h_i(r, s) \cos\theta \cdot \phi_{im}(t) \quad (3-22)$$

and

$$\frac{\partial}{\partial r} \phi_m(r, s, \theta, t) = \sum_{i=1}^8 \frac{\partial}{\partial r} h_i \cos\theta \cdot \phi_{im}(t) \quad (3-23)$$

$$\frac{\partial}{\partial s} \phi_m(r, s, \theta, t) = \sum_{i=1}^8 \frac{\partial}{\partial s} h_i \cos\theta \cdot \phi_{im}(t) \quad (3-24)$$

or in matrix form, the above two equations can be written as

$$\begin{Bmatrix} \frac{\partial \phi}{\partial r} \\ \frac{\partial \phi}{\partial s} \end{Bmatrix}_m = \cos\theta \begin{bmatrix} p_m(r, s) \\ \frac{2 \times 8}{2 \times 8} \end{bmatrix} \begin{bmatrix} \phi_m(t) \\ 8 \times 1 \end{bmatrix} \quad (3-25)$$

in which $p_m(r, s)$ contains the derivatives of interpolation functions derived from Eq. (3-20). Using the chain rule of differentiation, we can relate the global derivatives to the local derivatives.

$$\begin{Bmatrix} \frac{\partial \phi}{\partial r} \\ \frac{\partial \phi}{\partial s} \end{Bmatrix}_m = \underbrace{\begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial z}{\partial r} \\ \frac{\partial x}{\partial s} & \frac{\partial z}{\partial s} \end{bmatrix}}_{\text{Jacobian matrix}} \begin{Bmatrix} \frac{\partial \phi}{\partial x} \\ \frac{\partial \phi}{\partial z} \end{Bmatrix}_m \quad (3-26)$$

by inverting

$$\begin{bmatrix} \frac{\partial \phi}{\partial x} \\ \frac{\partial \phi}{\partial z} \end{bmatrix}_m = \underline{J}_m^{-1} \begin{bmatrix} \frac{\partial \phi}{\partial r} \\ \frac{\partial \phi}{\partial s} \end{bmatrix}_m \quad (3-27)$$

in which $\underline{J}_m^{-1}$ is the inverse of the Jacobian matrix in Eq. (3-26). Substituting for $\partial\phi/\partial r$ and for $\partial\phi/\partial s$ from Eq. (3-25) into Eq. (3-27), we obtain the relationship between the global derivatives and nodal values of ϕ .

$$\begin{bmatrix} \frac{\partial \phi}{\partial x} \\ \frac{\partial \phi}{\partial z} \end{bmatrix}_m = \underline{J}_m^{-1} \times \frac{p_m}{2 \times 2} (r, s) \frac{\phi_m(t)}{8 \times 1} \cdot \cos \theta \quad (3-28)$$

or

$$\begin{bmatrix} \frac{\partial \phi}{\partial x} \\ \frac{\partial \phi}{\partial z} \end{bmatrix}_m = \underline{B}_m \phi_m(t) \cos \theta \quad (3-29)$$

$$\text{in which } \underline{B}_m = \underline{J}_m^{-1} \underline{p}_m. \quad (3-30)$$

Thus

$$\begin{Bmatrix} \frac{\partial N}{\partial x} \\ \frac{\partial N}{\partial y} \end{Bmatrix}_m = \underline{B}_m \cos \theta \quad (3-31)$$

3.3.3 Free surface (mass) matrix for axisymmetric element

The complete free surface matrix for the system is formed by direct summation of individual element matrices i.e.,

$$\underline{M} = \frac{1}{g} \sum_{m=1}^n \underline{M}_m \quad (3-32)$$

where n is the total number of free surface elements, and the element matrix $\underline{M}_m$ is a 2×2 matrix given by

$$M_m = \int_{EB2} N_m^T N_m ds. \quad (3-33)$$

In case of axisymmetric case the surface integral can be transformed to a line integral. Consider a free surface element with nodes i and j (Fig. 3-3) and let

$$\begin{aligned} r &= \text{local coordinates for free surface boundary element} \\ R(r) &= \text{radius to any point } r \text{ between node } i \text{ and } j \\ L &= \text{length of the elements} \\ L &= \sqrt{(x_i - x_j)^2 + (z_i - z_j)^2}. \end{aligned} \quad (3-34)$$

Assuming that x -axis coincides with the horizontal plane, we can write the transformation.

$$R(r) = \begin{bmatrix} 1 - r/L & r/L \end{bmatrix} \begin{Bmatrix} x_i \\ x_j \end{Bmatrix}. \quad (3-35)$$

$$ds = R(r) dr d\theta \quad (3-36)$$

where θ is defined in Fig. 3-2

$$ds = \begin{bmatrix} 1 - r/L & r/L \end{bmatrix} \begin{Bmatrix} x_i \\ x_j \end{Bmatrix} dr d\theta. \quad (3-38)$$

Thus

$$\underline{N} = \begin{bmatrix} 1 - r/L & r/L \end{bmatrix} \cos\theta. \quad (3-39)$$

Therefore

$$\underline{M}_{nn} = \int_0^{2\pi} \int_0^L \begin{Bmatrix} 1-r/L \\ r/L \end{Bmatrix} \begin{bmatrix} (1-r/L) & r/L \end{bmatrix} R(r) \cos^2\theta \, dr d\theta. \quad (3-40)$$

$$\underline{M}_{m} = \pi \int_0^L \begin{Bmatrix} 1-r/L \\ r/L \end{Bmatrix} \begin{bmatrix} (1-r/L) & r/L \end{bmatrix} R(r) \, dr d\theta. \quad (3-41)$$

Evaluation of the above matrix gives

$$M_{ii} = \frac{\pi L}{4} \left[x_i + \frac{x_j}{3} \right] \quad (3-42)$$

$$M_{jj} = \frac{\pi L}{4} \left[\frac{x_i}{3} + x_j \right] \quad (3-43)$$

$$M_{ij} = \frac{\pi L}{12} \left[x_i + x_j \right] \quad (3-44)$$

3.3.4 Evaluation of fluid (stiffness) matrix for axisymmetric element

The complete system fluid matrix is formed by direct summation of element matrices

$$\underline{K} = \sum_{m=1}^n \underline{K}_m \quad (3-45)$$

where n is the number of liquid elements and the element matrix $\underline{K}_m$ is obtained by using Eq. (3-31) and Eq. (3-16).

$$\underline{K}_m = \int_{EV} \cos^2\theta \, \underline{B}_m^T \underline{B}_m \, dv \quad (3-46)$$

In case of an axisymmetric element, $dv = R \, d\theta dr$, the volume integral then becomes

$$\underline{K}_m = \int_0^{2\pi} \int_{A_m} \cos^2 \theta \ R \ \underline{B}_m^T \ \underline{B}_m \ dA d\theta \quad (3-47)$$

where R is the radius of any point and A is the area.

In the natural coordinate system $dA = |J_m| \ dsdr$ where $|J_m|$ is the determinant of the Jacobian matrix, therefore

$$\underline{K}_m = \pi \int_{-1}^1 \int_{-1}^1 R \ \underline{B}_m^T \ \underline{B}_m \ |J_m| \ dsdr. \quad (3-48)$$

The above integral can be evaluated numerically using Gaussian quadrature as follows [18]

$$\underline{K}_m = \sum_{j=1}^N \sum_{i=1}^N H_i \ H_j \ \underline{f} \ (r_i, s_i) \quad (3-49)$$

where N refers to the order of integration; H_i and H_j are the weighting factors and

$$\underline{f} \ (r_i, s_i) = \underline{B}_m^T \ (r_i, s_i) \ \underline{B}_m \ (r_i, s_i) |J_m \ (r_i, s_i)| \ R \ (r_i, s_i) \quad (3-49)$$

3.3.5 Load vector for axisymmetric element

The loading vector $\underline{F}$ for the complete system is the sum of the contribution of individual elements and using Eq. (3-17) we can write

$$\underline{F} = \sum_{m=1}^n \int_{EB1} N_m^T \ v_n \ ds = \sum_{m=1}^n \underline{F}_m \quad (3-50)$$

where

$$\underline{F}_m = \int_{EB1} N_m^T \ v_n \ ds \quad (3-51)$$

where summation is over all the n elements which are at the liquid-solid interface.

Consider a typical liquid-solid boundary element with nodes i and j and let r be a local coordinate as shown in Fig. 3-4. Let $R(r)$ be the radius at any point r and L be the length of the element where

$$L = \sqrt{(x_i - x_j)^2 + (z_i - z_j)^2}.$$

$$R(r) = \begin{bmatrix} (1-r/L) & r/L \end{bmatrix} \begin{Bmatrix} x_i \\ x_j \end{Bmatrix}. \quad (3-52)$$

$$N = \begin{bmatrix} (1-r/L) & r/L \end{bmatrix} \cos \theta \quad (3-53)$$

$$ds = R(r) dr d\theta.$$

If v_x = horizontal ground velocity in the x -direction then

$$v_n = v_x \cos \theta \cdot \cos \psi \quad \text{where} \quad \psi = \tan^{-1} \left(\frac{x_i - x_j}{z_i - z_j} \right); \text{ then}$$

$$F_m = v_x \int_0^L \int_0^{2\pi} \begin{bmatrix} (1-r/L) & r/L \end{bmatrix} \begin{Bmatrix} x_i \\ x_j \end{Bmatrix} \cos^2 \theta \cdot \cos \psi d\theta dr \quad (3-54)$$

integrating we get the following element load vector

$$F_m = \frac{\pi L v_x \cos \psi}{6} \begin{Bmatrix} 2x_i + x_j \\ x_i + 2x_j \end{Bmatrix} = \begin{Bmatrix} F_i \\ F_j \end{Bmatrix}. \quad (3-55)$$

The contribution to the loading matrix comes from those elements which lie at the liquid-solid interface.

3.4 Numerical Solution of Finite Element Equations

The discretization of the continuum into finite elements and the assemblage of free surface, liquid and loading element matrices results in a set of linear, coupled, second order ordinary differential equations. Since these equations are linear they can be uncoupled by an orthogonal transformation and the solution can be obtained using mode superposition [28, 29]

or they can be solved by direct step-by-step integration. In this study Newmark's step by step integration method [30,29] which is based on the following expressions was used.

$$\ddot{\phi}_{t+\Delta t} = \ddot{\phi}_t + \Delta t (1-\delta) \ddot{\phi}_t + \Delta t \delta \ddot{\phi}_{t+\Delta t} \quad (3-56)$$

$$\phi_{t+\Delta t} = \phi_t + \Delta t \dot{\phi}_t + \Delta t^2 \left(\frac{1}{2} - \alpha\right) \ddot{\phi}_t + \Delta t^2 \alpha \ddot{\phi}_{t+\Delta t} \quad (3-57)$$

in which Δt is step size. α and δ are parameters which are selected to produce the desired stability and accuracy. In all the sample analyses carried out in this investigation, Newmark's constant-average-acceleration method ($\delta=1/2$ and $\alpha = 1/4$) was used, which is an unconditionally stable method without numerical damping.

This is an implicit method and satisfies the equilibrium equations at time $t + \Delta t$, i.e.,

$$\underline{M} \ddot{\phi}_{t+\Delta t} + \underline{K} \phi_{t+\Delta t} = \underline{F}_{t+\Delta t} \quad (3-58)$$

The above three equations can be combined into a step-by-step algorithm which involves the solution of a set of equations at each time step of the form.

$$\underline{K}^* \phi_{t+\Delta t} = \underline{F}^* \quad (3-59)$$

In this analysis $\underline{K}^*$ is independent of time and is formed and triangularized only once. To make the numerical algorithm more general, the option of combining the Wilson Theta method [31] with the Newmark method was incorporated in the computer program. The Wilson Theta method was first applied to Newmark's linear accelerator method in order to improve stability and to damp out high frequency oscillations which often develop in step by step integration. A summary of the Newmark-Wilson algorithm used in the computer program is given below.

3.4.1 The Newmark-Wilson algorithm for linear step-by-step integration

INITIAL CALCULATIONS

1. Initialize ϕ_0 and $\dot{\phi}_0$ (taken to be zero) (ϕ is a vector)
2. Form the free surface and fluid matrices ($\underline{M}$, $\underline{K}$)
3. Specify algorithm parameters α , δ and θ
4. Calculate integration constants.

$$\begin{aligned} \tau &= \theta \Delta t & a_3 &= \frac{1}{2\alpha} - 1 & a_7 &= \Delta t \delta \\ a_0 &= \frac{1}{\alpha \tau^2} & a_4 &= \frac{\delta}{\alpha} - 1 & a_8 &= \Delta t^2 \left(\frac{1}{2} - \alpha \right) \\ a_1 &= \frac{\delta}{\alpha \tau} & a_5 &= \frac{\tau}{2} (\delta/\alpha - 2) & a_9 &= \alpha \Delta t^2 \\ a_2 &= \frac{1}{\alpha \tau} & a_6 &= \Delta t (1 - \delta) \end{aligned}$$

5. Form $\underline{K}^* = \underline{K} + a_0 \underline{M}$
6. Triangularize $\underline{K}^*$: $\underline{K}^* = \underline{LDL}^T$

FOR EACH TIME STEP

1. Calculate the effective load vector $\underline{F}^*$ at time $t + \tau$

$$\underline{F}^* = \underline{F}_{t+\tau} + \underline{M} (a_0 \phi_t + a_2 \dot{\phi}_t + a_3 \ddot{\phi}_t).$$
2. Solve for velocity potential ϕ at $t + \tau$:

$$\underline{LDL}^T \phi_{t+\tau} = \underline{F}^*$$
3. Calculate the velocity potential ϕ and its derivatives at time $t + \Delta t$:

$$\begin{aligned} \ddot{\phi}_{t+\tau} &= a_0 (\phi_{t+\tau} - \phi_t) - a_2 \dot{\phi}_t - a_3 \ddot{\phi}_t \\ \ddot{\phi}_{t+\Delta t} &= \ddot{\phi}_t + \frac{1}{\theta} (\ddot{\phi}_{t+\tau} - \ddot{\phi}_t) \\ \dot{\phi}_{t+\Delta t} &= \dot{\phi}_t + a_6 \ddot{\phi}_t + a_7 \ddot{\phi}_{t+\Delta t} \\ \phi_{t+\Delta t} &= \phi_t + \Delta t \dot{\phi}_t + a_8 \ddot{\phi}_t + a_9 \ddot{\phi}_{t+\Delta t} \end{aligned}$$

4. Determine the sloshing displacements and hydrodynamic impulsive pressures at time $t + \Delta t$

$$\text{Sloshing displacement } \delta_{t+\Delta t} = -\frac{1}{g} \dot{\phi}_{t+\Delta t}$$

$$\text{Impulsive pressure } p_{t+\Delta t} = -\rho \ddot{\phi}_{t+\Delta t}$$

3.5 Computer Program 'SLOSH2'

The program 'SLOSH2' developed to implement the finite element theory of the sloshing phenomenon in axisymmetric tanks is coded in standard FORTRAN IV language. The basic set up of SLOSH2 is the same as that of the computer program DOT [32] because of the fact that the finite element formulation of the sloshing problem is similar in certain respects to that of the heat conduction equations.

The earthquake input can either be as an accelerogram or a displacement-time history, digitized in the appropriate format. The program derives the velocity-time history by integration or differentiation depending upon the type of ground motion input. The earthquake input must be properly adjusted for base-line correction such that at the end of earthquake as acceleration goes to zero, the ground velocity and displacement also go to zero.

The 'effective' equilibrium equations (Eq. (3-59)) are solved using the linear equation solver COLSOL [33]. This subroutine processes only those elements which are within the skyline of $\underline{K}^*$, thus minimizing the storage requirements as well as the number of operations. This subroutine is based on Gauss elimination and requires a symmetrical positive-definite system of equations.

A compact storage scheme is used in the computer program whereby a one-dimensional array is used to store only those elements of K^* which are within its skyline. In the actual implementation of the computer program a lumped mass parameter system was used which not only simplifies the analysis considerably, but also minimizes the storage requirements. The three finite element groups namely, the free surface elements, the fluid elements and the liquid-solid interface elements, are processed in blocks and then stored on the disc for later use in order to increase the maximum capacity of the program.

It should be noted that the free surface elements and the liquid-solid interface elements which contribute to $\underline{M}$ and $\underline{F}$ respectively, are only two node elements for the axisymmetric case, whereas the liquid continuum itself is discretized by two-dimensional 4-to-8 node isoparametric elements which contribute to the fluid matrix $\underline{K}$.

In this computer program, a variable dimension is used for dynamic allocation of primary storage into a single array in bank common. The lower primary storage locations are used for the storage of each block of element group data which is read in from the secondary storage (disc) as required during the solution phase. The user has to supply the maximum estimated number of storage locations required to store any individual element group in the lowest primary storage. Usually this number is determined by the number of liquid elements that form the $\underline{K}$ matrix and not the free surface or liquid-solid interface elements. An estimation of the C.P.U. time required to run the program on CDC 6400 will be indicated in the next article. A user's manual and Fortran listing for SLOSH2 are given in Appendix A1 and A2 respectively.

3.6 Sample Analyses and Comparison with Test Data from Annular and Torus Tanks

Figures 3-5 and 3-6 show the finite element mesh layout for the 8 ft diameter annular tank and the 28 inches diameter torus tank models respectively which were studied in Ref. [1] and in Chapter 2 of this report. The annular tank chosen for this analysis is a simplified 1/15th scale model of pressure-suppression pool of Boiling Water Reactor GE Mark III (see Ref. [1] Figs. 5-3 and 5-7a) whereas the torus tank represents the 1/60th scale model (see Fig. 2-1) of the GE Mark I pressure-suppression pool. These analyses were carried out to check the accuracy of the finite element model against precise test data. A finer mesh size has been used near the free water surface because that is where the maximum sloshing displacements occur.

3.6.1 Annular tank

In Fig. 3-5 the x-axis is taken at the bottom of the tank and z-axis as the axis of symmetry of the annular tank. There are a total of 25, 10 and 17 elements in group Nos. 1, 2 and 3 respectively. Group No.1 contains 4-to-8 node elements whereas group Nos. 2 and 3 contain the free surface elements and liquid-solid interface elements respectively with two nodes each. There are a total of 52 nodes in this mesh. The finite element analysis was done using the digitized accelerogram recorded in Test No. 211276.1 of Ref. [1] Chapter 5 and a comparison of measured and predicted results is given in Figs. 3-7 and 3-8.

Figure 3-7 shows a comparison of the sloshing displacements at node 2 between the finite element solution and the test data under the

same ground motion. The results are shown for the first six seconds and it can be seen that there is close agreement between the test and finite element results.

Figure 3-8 shows a comparison of impulsive pressures between the measured and finite element results, and it can be seen again that the agreement between the two is excellent with test results consistently 5-10% higher compared with the finite element solution indicating a possible error of calibration of pressure gage and some error due to electrical noise. This comparison is given at node 52 (Fig. 3.5).

3.6.2 Torus tank

The finite element mesh layout for the torus tank model (Fig. 3-6) has a total of 24, 12 and 24 elements in group Nos. 1, 2 and 3 respectively. The z-axis is taken as the axis of symmetry of the torus and the x-axis at the bottom of the tank. The total number of nodes in this case is 73.

The earthquake input used for the finite element analysis was the recorded shaking table displacement instead of acceleration because the displacements gave zero velocity and acceleration on differentiation at the end of the earthquakes whereas the integration of the accelerogram usually gives finite amount of velocity and displacement without a base line correction.

In comparing the finite element solution with the test data for the torus tank it should be remembered that the two systems are not exactly the same: the finite element solution is for a perfectly round tank, whereas the test tank was made of a set of 16 straight segments as shown in Fig. 2-1. Figures 3-9, 3-10 and 3-11 show the comparison of sloshing displacements at node No.2 for increasing intensity of ground motion. In Figs. 3-9 and 3-10 the agreement between the two results during the first half

of the earthquake is excellent and for the rest of the earthquake quite satisfactory. In Fig. 3-11, although the sloshing response is relatively large, the linear finite element analysis still gives satisfactory results for practical purposes. The hydrodynamic pressures in this case were not measured on account of their relatively small magnitude in this small scale model.

The C.P.U. time required by the CDC 6400 to obtain the time-history response in the torus tank model analysis (Fig. 3-9) with the mesh size shown in Fig. 3-6 was 126 seconds.

3.7 Sample Analysis of Mark I Prototype Torus Tank Under El Centro 1940 Earthquake

Figure 3-12 shows the sloshing response in the prototype of Mark I torus under the full intensity of the 1940 El Centro earthquake (N-S component). The mesh layout for this analysis was similar to that shown in Fig. 3-6 except the overall dimensions which in this case are 60 times larger. The sloshing displacement shown in Fig. 3-12 is at node #2 with a maximum value of 24.3".

FIG. 3-1 TANK OF ARBITRARY SHAPE FILLED WITH LIQUID

XBL 789-2282

(a) TWO-DIMENSIONAL ELEMENT IN GLOBAL X-Z SYSTEM

(b) BI-UNIT SQUARE IN LOCAL r - s SYSTEM

FIG.3-2 TWO-DIMENSIONAL MAPPING OF AN ISOPARAMETRIC ELEMENT

FIG.3-3 FREE SURFACE ELEMENT (AXISYMMETRIC CASE)

FIG.3-4 BOUNDARY ELEMENT AT LIQUID-SOLID INTERFACE
(AXISYMMETRIC CASE)

XBL 789-2277

FIG. 3-5 FINITE ELEMENT MESH FOR ANNULAR TANK MODEL

XBL 789-2281

FIG. 3-6 FINITE ELEMENT MESH LAYOUT FOR TORUS TANK MODEL

XBL 789-2278

FIG. 3-7 COMPARISON OF SLOSHING DISPLACEMENTS BETWEEN THE TEST AND FINITE ELEMENT RESULTS AT NODE #2 (TEST #211276.1) IN ANNULAR TANK MODEL (INNER RADIUS = 33.2 IN., OUTER RADIUS = 48.0 IN., DEPTH OF WATER = 16 IN.) UNDER SIMULATED EL CENTRO 1940 EARTHQUAKE, TIME SCALE = $\sqrt{15} = 3.9$, PEAK SHAKING TABLE ACCELERATION = 0.24g HORIZONTAL, 0.0g VERTICAL.

FIG. 3-8 COMPARISON OF IMPULSIVE PRESSURES BETWEEN THE TEST AND FINITE ELEMENT RESULTS AT NODE #52 (TEST #211276.1) IN ANNULAR TANK MODEL.

FIG. 3-9 COMPARISON OF SLOSHING DISPLACEMENTS BETWEEN THE TEST AND FINITE ELEMENT RESULTS AT NODE #2 (TEST #220378.2) IN TORUS TANK MODEL (INNER RADIUS = 8 IN., OUTER RADIUS = 14 IN., DEPTH OF WATER = 3 IN.)

XBL 789-11315

FIG. 3-10 COMPARISON OF SLOSHING DISPLACEMENTS BETWEEN THE TEST AND FINITE ELEMENT RESULTS AT NODL #2 (TEST #220378.3) IN TORUS TANK MODEL (INNER RADIUS = 8 IN., OUTER RADIUS = 14 IN., DEPTH OF WATER = 3 IN.)

XBL 789-11317

FIG. 3-11 COMPARISON OF SLOSHING DISPLACEMENTS BETWEEN THE TEST AND FINITE ELEMENT RESULTS AT NODE #2 (TEST #220378.4) IN TORUS TANK MODEL (INNER RADIUS = 8 IN., OUTER RADIUS = 14 IN., DEPTH OF WATER = 3 IN.)

XBL 789-11318

FIG. 3-12 SLOSHING DISPLACEMENTS IN PROTOTYPE TORUS TANK AT NODE #2 UNDER ELCENTRO 1940 EARTHQUAKE (INNER RADIUS = 40 FT., OUTER RADIUS = 70 FT., DEPTH OF WATER = 15 FT.)

4. CONCLUSIONS

(1) A comparison of data from shaking table tests on annular and torus tanks confirms that the finite element analysis presented in this report can successfully predict hydrodynamic pressures and free surface displacement in rigid axisymmetric tanks under strong motion earthquakes. The finite element program is applicable not only to axisymmetric tanks but may also have possible application to offshore structures under seismic conditions.

(2) As sloshing response is not very sensitive to the precise geometry of the tank section, a modified annular tank solution gives satisfactory results in predicting the sloshing frequencies and displacements in torus tanks under horizontal ground motions.

(3) The validity of the theory developed herein is independent of the size of the model that was analyzed and tested. In the 1/60-scale torus model the sloshing response is produced mainly by the low frequency components of the reference earthquake ground motion, and these are correctly reproduced on the 20-foot shaking table.

ACKNOWLEDGMENTS

This investigation was sponsored by the United States Nuclear Regulatory Commission and was done with support from the United States Department of Energy.

REFERENCES

1. M. Aslam, W. G. Godden, and D. T. Scalise, "Sloshing of Water in Annular Pressure-Suppression Pool of Boiling Water Reactors Under Earthquake Ground Motions", Lawrence Berkeley Laboratory, University of California, Berkeley, Ca. LBL-6754, August 1977.
2. E. W. Graham and A. M. Rodriguez, "The Characteristics of Fuel Motion which Affect Airplane Dynamics", Journal of Applied Mechanics, Vol. 19, No. 3, September 1952.
3. J. Miles, "On the Sloshing of Liquid in a Flexible Tank", ASME, Paper No. 57-A-12, 1957.
4. J. Miles, "On the Sloshing of a Liquid in a Cylindrical Tank", Ramo-Wooldridge AM-6-5, CM-TR-18, 1956.
5. B. Budiansky, "Sloshing of Liquids in Circular Canals and Spherical Tanks", Lockheed Aircraft Corp. LMSD-5151, December 1958.
6. H. F. Bauer, "Fluidoscillations in a Circular Cylindrical Tank", American Boilers Manufacturers Association (ABMA) Report DA-TR-1-58.
7. H. F. Bauer, "Fluidoscillations in a Cylindrical Tank with Damping", ABMA Report DA-TR-4-58.
8. H. F. Bauer, "Fluidoscillations in a Circular Cylindrical Tank Due to Bending of Tank Walls", ABMA Report DA-3-58.
9. H. F. Bauer, "Damped Fluidoscillations in a Circular Cylindrical Tank Due to Bending Walls", ABMA Report DA-9-58.

10. K. Kachigan, "Forced Oscillations of a Fluid in a Cylindrical Tank", Convair Report ZU-7-046, October 1956.
11. A. F. Schmitt, "Forced Oscillations of a Fluid in a Cylindrical Tank Undergoing Both Translation and Rotation", Convair Report Zu-7-069, October 1956.
12. H. F. Bauer, "The Dynamic Behavior of Liquids in Moving Containers", National Aeronautics and Space Administration, Washington, D.C., 1966, edited by H. Norman Abramson.
13. J. J. Stoker, "Water Waves", Interscience Publishers, Inc., New York, 1957.
14. L. S. Jacobson and R. S. Ayre, "Hydrodynamic Experiments with Rigid Cylindrical Tanks Subjected to Transient Motions", Bull. Seism. Soc. Am., Vol. 41, 1952.
15. P. C. Jennings, G. W. Housner, and N. C. Tsai, "Simulated Earthquake Motion", California Institute of Technology, Pasadena, California, April, 1968.
16. D. P. Clough, "Experimental Evaluation of Seismic Design Methods for Broad Cylindrical Tanks", Ph.D. Dissertation, University of California, Berkeley, 1976.
17. A. S. Veletsos and J. Y. Yang, "Dynamics of Fixed-Base Liquid Storage Tank", U. S., Japan Seminar on Lifelines, 1976.
18. O. C. Zienkiewicz, "The Finite Element Method in Engineering Science", McGraw-Hill, London 1978.

19. N. W. Edwards, "A Procedure for Dynamic Analysis of Thin-Walled Cylindrical Liquid Storage Tanks Subjected to Lateral Ground Motions", Ph.D. Dissertation, University of Michigan, 1969.
20. C. H. Luk, "Finite Element Analysis for Liquid Sloshing Problems", Office of Aerospace Research, U. S., Air Force Report No. 69-1504 TR, 1969.
21. P. Tong and T. Pain, "A Variational Principle and Convergence of a Finite Element Model Based on Stress Distributions", AFOSR TR 68-0384 (M.I.T. ASRL Tr 144-1), June 1968.
22. S. Crandall, "Engineering Analysis", McGraw-Hill, New York, 1965.
23. B. A. Finlayson, "The Method of Weighted Residuals", Applied Mechanics Reviews, Vol. 19 No. 9, September 1966.
24. S. G. Hutton, "Finite Element Method-Galerkin Approach", Journal of Engineering Mechanics Division, Vol. 97 No. EM5, October 1971.
25. O. C. Zienkiewicz and C. J. Parekh, "Transient Field Problems: Two-Dimensional and Three-Dimensional Analysis by Isoparametric Finite Elements", International Journal of Numerical Methods in Engineering, Vol. 2, No. 1, June 1970.
26. G. P. Pinder, "Application of Galerkin Procedure to Aquifer Problems", ASCE Journal of Water Resources, Vol. 8 No. 1, February 1972.

27. I. Ergatoudis, B. M. Irons and O. C. Zienkiewicz, "Curved Iso-parametric Quadrilateral Elements for Finite Element Analysis", International Journal of Solids and Structures, Vol. 4, 1968, pp. 31-42.
28. R. W. Clough and J. Penzien, "Dynamics of Structures", McGraw-Hill Book Company, New York, N.Y., 1975.
29. K. J. Bathe and E. L. Wilson, "Numerical Methods in Finite Element Analysis", Prentice-Hall, Inc., Englewood Cliffs, N.J. 1976.
30. N. M. Newmark, "A Method of Computation for Structural Dynamics", ASCE, Journal of Engineering Mechanics Division, Vol. 85, 1959, pp. 67-94.
31. E. L. Wilson, "Computer Analysis Language for the Static and Dynamic Analysis of Structural Systems", Department of Civil Engineering, University of California, Berkeley, California, Report No. UC SESM 77-2. January 1977.
32. R. M. Polivka and E. L. Wilson, "Finite Element Analysis of Non-Linear Heat Transfer Problems", Department of Civil Engineering, University of California, Berkeley, California. Report No. UC SESM 76-2, June 1976.
33. E. L. Wilson, K. J. Bathe and W. P. Doherty, "Direct Solution of Large Systems of Linear Equations", Computers and Structures, Vol. 4, No. 2, pp. 363-372, March 1974.

APPENDIX A1

SLOSH2 USER'S MANUAL

SLOSH2: A linear finite element program which determines the sloshing displacements and impulsive pressures in axisymmetric rigid tanks under arbitrary horizontal ground motions.

Developed by: Mohammad Aslam
Department of Civil Engineering
University of California, Berkeley

CONTENTS

A1.1	Program Description
A1.2	Program Capacity
A1.3	Input Data
	I. Problem Initiation and Title
	II. Master Control Card
	III. Nodal Coordinates
	IV. Solution Time and Step Size
	V. Earthquake Input
	VI. Element Data
	Type 1: Two Dimensional Finite Elements
	Type 2: Free Surface Elements
	Type 3: Solid-Liquid Interface Elements
	VII. New Problem Data
	VIII. Termination Card
A1.4	Output

A1.1 Program Description

The computer program SLOSH2 has been developed to predict sloshing displacements and impulsive pressures in a liquid filled axisymmetric container subjected to only horizontal ground motion. The tank is assumed to be rigid and fixed at the base. The program requires the following three types of elements.

1. Two dimension 4-to-8 node axisymmetric elements idealizing the liquid.
2. Two node free surface elements.
3. Two node elements representing the liquid-solid interface.

A1.2 Program Capacity

The program uses a variable dimensioning in order to make an optimum use of high speed storage. Element group data is stored block wise on the disc. The program capacity can be varied through two Fortran statements in the main program.

```
COMMON (n)
MTOT = n
```

The total memory n required can be estimated by the following formula.

$$n = M + 2 * NPTM + 10 * NUMNP$$

in which

```
M          = NELL * (4 * MXNODS - 2)
NELL       = Number of elements in group 1
MXNODS     = Maximum nodes in any element of group 1
NPTM       = Number of points of earthquake input
NUMNP      = Total number of node points
```

A1.3 Program Input Data

The following format should be followed for the necessary input data.

I. Problem Initiation and Title (A5, 3X, 18A4) - one card

<u>Columns</u>	<u>Variable</u>	<u>Description</u>
1 - 5	MODE	Punch the word 'START'
9 - 80	HED	Title of the problem

II. Master Control Card (4I5) - one card

<u>Columns</u>	<u>Variable</u>	<u>Description</u>
1 - 5	NUMNP	Total number of nodes
6 - 10	NEG	Number of element groups
11 - 15	NUMEST	Estimated number of storage locations required (M ₁) for element group 1. Zero or blank: defaults to 3000
16 - 20	MODEX	Execution mode. Specify (a) Zero: data check only (b) 1: execution

III. Nodal Coordinates (I5, 5X, 2F 10.0, I5)

As many cards as needed to generate total number of nodes

NUMNP and their coordinates

<u>Columns</u>	<u>Variable</u>	<u>Description</u>
1 - 5	N	Node number See Note 1
11 - 20	X(N)	X coordinate
21 - 30	Y(N)	Z coordinate
35 - 35	KN	Node number difference between successive generated nodes (given on first card in a sequence). Specify. Zero: No generation. See Note 2.

Note:

- (1) Node cards may not be in numerical order. Eventually, however, all nodes must be identified.

- (2) The mesh generation parameter KN must appear on the first card of a series of nodal points to be generated. The intermediate nodes to be generated between nodes (say N1 and N2) will be located at equal intervals along the straight line joining the two nodes. KN is the nodal increment to be added to previous node number. The node difference N2-N1 must be exactly divisible by KN.

IV. Solution Time and Step Size (2I5, 3F10.0, 3I5)

<u>Columns</u>	<u>Variable</u>	<u>Description</u>
1 - 5	NDT	Number of solution time steps. Specify Zero: defaults to 1 step
6 - 15	DT	Step size
16 - 20	NP:INT	Time interval for printout of nodal displacements and pressures expressed as a multiple of the integration time step. Specify Zero: defaults to 1

V. Earthquake Input

A. Control Information (2I5) - one card

<u>Columns</u>	<u>Variable</u>	<u>Description</u>
1 - 5	NBCE	Number of ground input components (use 1)
6 - 10	NPTM	Maximum number of points to describe the earthquake input. See Note 1.

Note:

- (1) NPTM is the number of $[f(t), t]$ pairs used to define the earthquake ground motion which could be either an acceleration or displacement-time history record. At least two points are required to describe the input.

B. Earthquake Input Data

For one component of ground motion (horizontal in this case) a control card followed by as many cards as needed to define the earthquake.

1. Control Card (2I5, F10.0, I5) - First card

<u>Columns</u>	<u>Variable</u>	<u>Description</u>
1 - 5	NC	Function number. Specify equal to 1 in this case
6 - 10	NPTS(NC)	Number of time points used to describe the earthquake input (GE.2 and EQ. to NPTM)
11 - 20	FOM	Multiplication factor for conversion to right units. See Note 1
21 - 25	INPUT	Specify 1 - If acceleration is ground input 2 - If displacement is the given ground input.

2. [f(t), t] Earthquake Data (8F10.0)

As many cards as needed to define NPTS (NC) pairs of points

[TFN(NC,I), FN(NC,I)]; four pairs per card. See Note 2.

<u>Columns</u>	<u>Variable</u>	<u>Description</u>
1 - 10	TFN(NC,1)	Time at point 1 : t_1
11 - 20	FN(NC,1)	Acceleration or displacement value at point 1 : $f(t_1)$
21 - 30	TFN(NC,2)	t_2
31 - 40	FN(NC,2)	$f(t_2)$
41 - 50	TFN(NC,3)	t_3
51 - 60	FN(NC,3)	$f(t_3)$
61 - 70	TFN(NC,4)	t_4
71 - 80	FN(NC,4)	$f(t_4)$
<u>Next card(s)</u>	-	as many as needed to define the earthquake input.

Note:

- (1) Factor of multiplication if necessary to make the units of ground input compatible with the units of the tank dimensions. This option is available only if input is in the form of accelerometer. In case of displacement history the units must be in inches, seconds and pounds.
- (2) Time values at successive points are assumed to increase in magnitude. Values of ground input other than $TFN(NC,I)$ are calculated within the program using a linear interpolation.

VI. Element Data

Elements are divided into three groups (NEG). An element group is a series of elements of a particular type.

The elements in a particular group must be numbered sequentially starting with the number of the first element as specified on the element group control card.

Following are the three types of element groups used in this program.

Type 1 - Two Dimensional Finite Elements

These are 4-to-8 node axisymmetric isoparametric elements which lie in the global X-Z plane and are used to model the liquid continuum. Z-axis has been taken as the axis of revolution for the axisymmetric tank.

Type 2 - Free Surface Elements

These are 2 node axisymmetric elements which have been used to represent the free surface of the liquid. These

elements lie in the X-Z global plane where γ is the axis of revolution of the tank. These elements contribute to mass matrix.

Type 3 - Liquid-Solid Interface Elements

These are two node axisymmetric elements and lie in the global X-Z plane. These elements contribute to the loading vector.

Type 1 - Two Dimensional Finite Elements

A. Control Information (6I5) - one card.

<u>Columns</u>	<u>Variable</u>	<u>Description</u>
1 - 5	NGR	Element group indicator. Punch the number "1".
6 - 10	NEL1	Number of elements in group 1.
11 - 15	MFST	Element number of the first element in this. See Note 1.
16 - 20	ITYP2D	Element type code. Specify Zero: axisymmetric
21 - 25	MXNODS	Maximum number of nodes used to describe any one element. Specify Zero: defaults to 4. (GE.4 and LE.8)
26 - 30	NINT	Numerical integration order to be used in Gaussian quadrature. Specify Zero: defaults to 2 (GE.2 and LE.4) See Note 2.

Note:

- (1) Element numbers in any group may not start from 1 if MFST is specified.
- (2) For rectangular elements, an integration order of 2 is sufficient. For non rectangular elements a higher order should be used.

B. Element Data (11I5)

As many data cards as are needed in order to generate the element data for the elements (NEL1) in this group.

<u>Columns</u>	<u>Variable</u>	<u>Description</u>
1 - 5	M	Element number See Note 1
6 - 10	NOD(1)	Global node number of element node 1.
11 - 15	NOD(2)	Global node number of element node 2.
16 - 20	NOD(3)	Global node number of element node 3.
21 - 25	NOD(4)	Global node number of element node 4.
26 - 30	NOD(5)	Global node number of element node 5.
31 - 35	NOD(6)	Global node number of element node 6.
36 - 40	NOD(7)	Global node number of element node 7.
41 - 45	NOD(8)	Global node number of element node 8. See Note 2
46 - 50	IEL	Number of nodes in the element. Zero: defaults to MXNODS
51 - 55	KG	Node number increment for element generation (given on 1st card in a sequence) Zero: defaults to 1 See Note 3

Note:

- (1) Elements must be input in increasing sequence, with MFST being the 1st element. Cards for the first and last element must be included.
- (2) If an element has less than 8 nodes (i.e., IEL.LT.8), input a zero or blank corresponding to the missing node location. For example, for a 6 node element with nodes 6 and 8 missing, the element node number array would be NOD(I) - [X X X X X 0 X 0] where X entries represent the global node numbers.

- (3) The node generation parameter KG must appear on the first element card of a sequence and is used to determine the node numbers for a group of missing elements in that sequence. If M is the first element of the sequence and the elements $[M+1, M+2, \dots, M+J]$ are the missing J elements, then the node numbers of the successive J elements are automatically incremented by the value KG given for the element M. Only the nonzero node numbers appearing on the M-th element card are incremented in this automatic generation. In the printout of the element data, generated elements are marked with an asterisk.

Type 2 - Free Surface Boundary Elements

A. Control Information (4I5) - one card.

<u>Columns</u>	<u>Variable</u>	<u>Description</u>
1 - 5	NGR	Element group number. Punch the number '2'.
6 - 10	NEL2	Number of elements in group 2
11 - 15	MFST	Number of the first element in group 2 (need not start with 1)
16 - 20	ITYP	Element type. Specify Zero: axisymmetric this is the only option available.

B. Element Data (4I5)

As many cards as needed to generate NEL2 elements.

<u>Columns</u>	<u>Variable</u>	<u>Description</u>
1 - 5	M	Element number See Note (1)
6 - 10	NOD(1)	Global node number of element node I
11 - 15	NOD(2)	Global node number of element node J
16 - 20	KG	Node number increment for element generation. Zero: defaults to 1. See Note (2)

Note:

- (1) All elements must be input in ascending numerical order, starting with element number MFST. Cards for the first and last element must be included.
- (2) The node generation parameter KG must be given on the first element card prior to a group of missing elements. In the print out of the element data, generated elements are pre-fixed by an asterisk.

Type 3 - Liquid-Solid Interface ElementsA. Control Information (4I5) - one card.

<u>Columns</u>	<u>Variable</u>	<u>Description</u>
1 - 5	NGR	Element group number Punch the number '3'
6 - 10	NEL3	Number of elements in group 3
11 - 15	MFST	Number of first element in group 3
16 - 20	ITYP	Element type : Specify Zero: axisymmetric (the only option available)

B. Element Data (3I5, 5X, F10.0)

As many cards as the number of elements NEL3 in group 3

<u>Columns</u>	<u>Variable</u>	<u>Description</u>
1 - 5	M	Element number
6 - 10	NOD(1)	Global node number of element node I
11 - 15	NOD(2)	Global node number of element node J
21 - 30	COSS	X-direction cosine of the outward normal to the element.

VII. New Problem Data

A new problem may now be solved by adding data starting with Section I. Any number of problems can be solved in one run.

VIII. Termination Card (A4) - one card

<u>Columns</u>	<u>Variable</u>	<u>Description</u>
1 - 4	MODE	Punch the word 'STOP'.

A1.4 Output

Output includes the nodal displacements and impulsive pressures. Displacements are meaningful only for the free surface nodes.

A2-1

APPENDIX A2

FORTRAN LISTING OF SLOSH2

```

*****
PROGRAM SLOSH2(INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,      SLOSH  1
+   TAPE1,TAPE2,TAPE3=PUNCH)                                     SLOSH  2
*****                                                             SLOSH  3
SLOSH2----A FINITE ELEMENT PROGRAM TO DETERMINE THE SLOSHING      SLOSH  4
RESPONSE UNDER EARTHQUAKE GROUND MOTIONS IN AN                  SLOSH  5
AXI-SYMMETRIC RIGID TANK                                          SLOSH  6
DEVELOPED BY--- MOHAMMAD ASLAM, DEPARTMENT OF CIVIL              SLOSH  7
ENGINEERING, UNIVERSITY OF CALIFORNIA,BERKELEY                   SLOSH  8
AUGUST 1978                                                        SLOSH  9
*****                                                             SLOSH 10
COMMON /CNTRL1/ NUMNP,NEG,MODEX,NPAR(10),NG,KBC                  SLOSH 11
COMMON /CNTRL2/ KST,NDT,DT,TSTART,TAMB,NPRINT,NTSREF,TIME,KP     SLOSH 12
COMMON /DIM / N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15 SLOSH 13
COMMON /ELSTOR/ NUNEST,MIDEST,MAXEST                             SLOSH 14
COMMON /JUNK / HED(18),MTOT,NLINE                                SLOSH 15
COMMON /NBC / NNBC,NBCF,NPTM                                     SLOSH 16
COMMON /WORK / WORK(200)                                         SLOSH 17
COMMON/CONST /A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,THETA,DELTA,ALPHA,P1, SLOSH 18
+RO                                                                SLOSH 19
DIMENSION SD(2000),PB(2000)                                       SLOSH 20
COMMON A(10000)                                                    SLOSH 21
                                                                SLOSH 22
MTOT = 10000                                                       SLOSH 23
200 MAXEST = 0                                                      SLOSH 24
                                                                SLOSH 25
*****                                                             SLOSH 26
INPUT PHASE                                                         SLOSH 27
*****                                                             SLOSH 28
PROGRAM MASTER CONTROL DATA                                       SLOSH 29
CALL DOTI                                                           SLOSH 30
INPUT ELEMENT INFORMATION                                           SLOSH 31
                                                                SLOSH 32
CALL ELCAL                                                         SLOSH 33
                                                                SLOSH 34
*****                                                             SLOSH 35
SOLUTION PHASE                                                      SLOSH 36
*****                                                             SLOSH 37
*****                                                             SLOSH 38
*****                                                             SLOSH 39
*****                                                             SLOSH 40
*****                                                             SLOSH 41
*****                                                             SLOSH 42
BLANK COMMON STORAGE ALLOCATION                                     SLOSH 43
*****                                                             SLOSH 44
      ARRAY -----DESCRIPTION----- DIMENSION                SLOSH 45
N1   TFN      TIME VALUES AT POINTS      NPTM*NBCE            SLOSH 46
N2   FN       FUNCTION VALUES AT POINTS   NPTM*NBCE            SLOSH 47
N3   NPTS     NUMBER OF FUNCTION INPUT POINTS NBCE              SLOSH 48
N4   TD       FIRST DERIVATIVE OF VEL POT. NUMNP                SLOSH 49
N5   TDD      2ND DERIVATIVE OF VEL. POT.  NUMNP                SLOSH 50
N6   TTAU     NUMNP                SLOSH 51
N7   P        PRESSURE(DYNAMIC IMPULSIVE) NUMNP                SLOSH 52
N8   T        VELOCITY POTENTIAL          NUMNP                SLOSH 53
N9   MAXA     ADDRESSES OF XK DIAGONAL ELEMENTS NUMNP+1        SLOSH 54
N10  XK       EFFECTIVE STIFFNESS MATRIX   NUK                 SLOSH 55
N11  D        LOADING VECTOR              NUMNP                SLOSH 56
N12  C        MASS MATRIX                 NUMNP                SLOSH 57
N13  E        WATER DISPLACEMENTS AT SURFACE NUMNP

```

C	N14	TT	VELOCITY POTENTIAL AT (T-DT)	NUMNP	SLOSH 58
C					SLOSH 59
C					SLOSH 60
			CALL ADRSK (A(N11),A(N12),NUMNP,NLK,MB)		SLOSH 61
C					SLOSH 62
C			SHIFT STORAGE TO ELIMINATE NODAL COORDINATE DATA		SLOSH 63
C					SLOSH 64
	5	I = 1 + MAXEST			SLOSH 65
		N12M = N12 - 1			SLOSH 66
		DO 10 J=N3,N12M			SLOSH 67
		A(I) = A(J)			SLOSH 68
	10	I = I + 1			SLOSH 69
C					SLOSH 70
		N1 = 1 + MAXEST			SLOSH 71
		N2 = N1 + NPTT*NBCF			SLOSH 72
		N3 = N2 + NPTT*NBCF			SLOSH 73
		N4 = N3 + NBCF			SLOSH 74
		N5=N4 + NUMNP			SLOSH 75
		N6=N5 + NUMNP			SLOSH 76
		N7=N6 + NUMNP			SLOSH 77
		N8=N7 + NUMNP			SLOSH 78
		N9 = N8 + NUMNP			SLOSH 79
		N10 = N9 + NUMNP + 1			SLOSH 80
		N11 = N10 + NLK			SLOSH 81
		N12 = N11 + NUMNP			SLOSH 82
		N13 = N12 + NUMNP			SLOSH 83
		N14 = N13 + NUMNP			SLOSH 84
		N15 = N14 + NUMNP			SLOSH 85
		IF(N15.GT.MTOT) CALL ERROR (N15-MTOT)			SLOSH 86
C					SLOSH 87
		IF(MODEX.EQ.0) GO TO 200			SLOSH 88
C					SLOSH 89
C			INITIALIZE STIFFNESS MATRIX (XX) AND LOADING VECTOR 0		SLOSH 90
C					SLOSH 91
		N12M = N12 - 1			SLOSH 92
		DO 15 I=N10,N12M			SLOSH 93
	15	A(I) = 0.0			SLOSH 94
C					SLOSH 95
C			INITIALIZE VELOCITY POTENTIAL VECTOR AT AT TT(0)=T(0)		SLOSH 96
C					SLOSH 97
		DO 20 I=1,NUMNP			SLOSH 98
		IT = N8 + I - 1			SLOSH 99
		ITT = N14 + I - 1			SLOSH100
	20	A(ITT) = A(IT)			SLOSH101
C					SLOSH102
C			INITIALIZE THE TIME STEP COUNTER		SLOSH103
C					SLOSH104
	22	KSTEP = 0			SLOSH105
		TIME=0.			SLOSH106
C					SLOSH107
C			INITIALIZE MASS MATRIX (LUMPED MASS SYSTEM)		SLOSH108
C					SLOSH109
		N13M = N13 - 1			SLOSH110
		DO 25 I=N12,N13M			SLOSH111
	25	A(I) = 0.0			SLOSH112
C					SLOSH113
C			CALCULATE CONSTANTS OF INTEGRATION		SLOSH114
C					SLOSH115
		P1=3.141592654			SLOSH116
		G=386.18			SLOSH117
		RO=0.00003351			SLOSH118
		THETA=1.0			SLOSH119
					SLOSH120

	DELTA=0.50	SLOSH121
	ALPHA=0.25	SLOSH122
	TAU=THETA*DT	SLOSH123
	A0=1.0/(ALPHA*TAU*TAU)	SLOSH124
	A1=DELTA/(ALPHA*TAU)	SLOSH125
	A2=1.0/(ALPHA*TAU)	SLOSH126
	A3=1.0/(2.*ALPHA)-1.	SLOSH127
	A4=DELTA/ALPHA-1.	SLOSH128
	A5=TAU*(DELTA/ALPHA-2.0)/2.	SLOSH129
	A6=DT*(1.-DELTA)	SLOSH130
	A7=DT*DELTA	SLOSH131
	A8=DT*DT*(0.5-ALPHA)	SLOSH132
	A9=ALPHA*DT*DT	SLOSH133
C	ASSEMBLE THE EFFECTIVE SYSTEM STIFFNESS MATRIX(K*)	SLOSH134
C		SLOSH135
C	30 CALL ASSEM*	SLOSH136
C		SLOSH137
C	FORM THE EFFECTIVE K AND CALL IT KK	SLOSH138
C		SLOSH139
C	CALL KSTAR(A(N9),A(N10),A(N12))	SLOSH140
C		SLOSH141
C	INITIALIZE VELOCITY POTENTIAL AND ITS DERIVATIVES	SLOSH142
	NGM=NG-1	SLOSH143
	DO 36 I=N4,NG	SLOSH144
36	A(I)=0.	SLOSH145
C	TRIANGULARIZE THE EFFECTIVE CONDUCTIVITY MATRIX. (K*)	SLOSH146
C		SLOSH147
	40 KTR = 0	SLOSH148
	CALL COLSOL (A(N10),A(N11),A(N9),NUMNP,MB,NLK,KTR)	SLOSH149
		SLOSH150
C	*****	SLOSH151
C	TIME MARCHING LOOP	SLOSH152
C	INITIALIZE Q	SLOSH153
	N12M=N12-1	SLOSH154
	DO 44 I=N11,N12M	SLOSH155
44	A(I)=0.	SLOSH156
	TX=THETA*DT	SLOSH157
	*****	SLOSH158
C		SLOSH159
C		SLOSH160
	100 KSTEP = KSTEP + 1	SLOSH161
	TTH=TIME+TX	SLOSH162
	TIME = TIME + DT	SLOSH163
C		SLOSH164
C	FORM THE LOAD VECTOR	SLOSH165
C		SLOSH166
75	CALL FORMDC(TTH)	SLOSH167
C		SLOSH168
C	COMPUTE EFFECTIVE LOAD VECTOR	SLOSH169
		SLOSH170
C	CALL QEFF(A(N11),A(N12),A(N8),A(N4),A(N5),NUMNP)	SLOSH171
		SLOSH172
C	UPDATE (T*) VECTOR	SLOSH173
C		SLOSH174
	DO 82 I=1,NUMNP	SLOSH175
	IT = N8 + I - 1	SLOSH176
	ITT = N14 + I - 1	SLOSH177
82	A(ITT) = A(IT)	SLOSH178
C		SLOSH179
C	SOLVE THE EQUILIBRIUM EQUATIONS FOR VELOCITY POTENTIAL	SLOSH180
		SLOSH181
	84 KTR = ?	SLOSH182
		SLOSH183

C	CALL COLSOL (A(N10),A(N11),A(N9),NUMNP,MB,NLK,KTR)	SLOSH184
C	Q-VECTOR IS NOW T-VECTOR. SET T(1)=Q(1) AND Q(1)=0.	SLOSH185
C	DO 85 I=1,NUMNP	SLOSH186
	IT = N8 + I - 1	SLOSH187
	IO = N11 + I - 1	SLOSH188
	A(IT) = A(IO)	SLOSH189
85	A(IO) = 0.0	SLOSH190
C		SLOSH191
C	CALCULATE VEL. POTENTIAL AND ITS DERIVATIVE AT TIME+DT	SLOSH192
C	CALL CALCU(A(N8),A(N14),A(N4),A(N5),A(N7),A(N13),NUMNP)	SLOSH193
C	PRINT AND/OR PUNCH THE NODAL DISPLACEMENTS AND PRESSURES.	SLOSH194
C	IF REQUESTED, AT THIS TIME STEP	SLOSH195
C	K = MOD(KSTEP,NPRINT)	SLOSH196
	IF(K.NE.0) GO TO 90	SLOSH197
	CALL OUT(A(N13),NUMNP,TIME,KSTEP)	SLOSH198
	CALL OUT(A(N7),NUMNP,TIME,KSTEP)	SLOSH199
90	IF(KP.EQ.0) GO TO 92	SLOSH200
	L = MOD(KSTEP,KP)	SLOSH201
	IF(L.NE.0) GO TO 92	SLOSH202
	CALL PTEMP (A(N8),TIME,NUMNP)	SLOSH203
C		SLOSH204
92	CONTINUE	SLOSH205
	SD(1)=0.	SLOSH206
	NN=KSTEP+1	SLOSH207
C	CHECK FOR FINAL TIME STEP	SLOSH208
C		SLOSH209
	IF(KSTEP.LT.NNT) GO TO 100	SLOSH210
	GO TO 200	SLOSH211
C		SLOSH212
	END	SLOSH213
	SUBROUTINE DOT1	SLOSH214
C		SLOSH215
	COMMON /CNTRL1/ NUMNP,NEG,MODEX,NPAR(10),NG,KBC	SLOSH216
	COMMON /CNTRL2/ KST,NNT,DT,TSTART,TAMB,NPRINT,NTSREF,TIME,KP	DOT1 1
	COMMON /DIM / N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	DOT1 2
	COMMON /ELSTOR/ NUMEST,MIDEST,MAXEST	DOT1 3
	COMMON /JUNK / HET(19),HTOT,HLINE	DOT1 4
	COMMON /NBC / NNBC,NBCF,NPTH	DOT1 5
	COMMON /K(1)	DOT1 6
	COMMON /MOD(2)	DOT1 7
	DATA MOD/SHSTART,SHSTOP /	DOT1 8
C		DOT1 9
	*****	DOT1 10
	READ CONTROL INFORMATION	DOT1 11
	*****	DOT1 12
10	READ (5,1000) MODE,MCD	DOT1 13
	IF(MODE.EQ.MOD(2)) STOP	DOT1 14
	IF(MODE.EQ.MOD(1)) GO TO 20	DOT1 15
	WRITE(6,3000)	DOT1 16
	GO TO 10	DOT1 17
C		DOT1 18
20	READ (5,1001) NUMNP,NEG,NUMEST,MODEX	DOT1 19
C		DOT1 20
	IF(NUMEST.EQ.0) NUMEST = 4000	DOT1 21
	IF(NUMNP.GT.0) GO TO 30	DOT1 22
	WRITE(6,3001)	DOT1 23
	STOP	DOT1 24
30	IF(NEG.GT.0) GO TO 40	DOT1 25
		DOT1 26
		DOT1 27
		DOT1 28
		DOT1 29

```

WRITE(6,3002)
STOP
40 CALL TITLE (HED)
WRITE(6,2000) NUMNP,NIG,NUMEST,MODEX
NLINE = 17

*****

BLANK COMMON STORAGE ALLOCATION

N1 ARRAY -----DESCRIPTION----- DIMENSION
N2 X MODAL X-COORDINATES NUMNP
N3 Y MODAL Y-COORDINATES NUMNP
N4 TFM TIME VALUES AT POINTS NPTM*IBCF
N5 FN FUNCTION VALUES AT POINTS NPTM*IBCF
N6 NPTS NUMBER OF FUNCTION INPUT POINTS NBCF
N7 TD FIRST DERIVATIVE OF VEL. POT. NUMNP
N8 TDD 2ND DERIVATIVE OF VEL. POT. NUMNP
N9 TTAU NUMNP
N10 P PRESSURE(DYNAMIC IMPULSIVE) NUMNP
N11 T VELOCITY POTENTIAL NUMNP
N12 MAXA ADDRESSES OF (XX) DIAGONAL ELTS. NUMNP+1
N13 MHT ACTIVE COLUMN HEIGHTS NUMNP

*****

=====
READ MODAL POINT COORDINATE DATA
=====

N1 = 1 + NUMEST
N2 = N1 + NUMNP
N3 = N2 + NUMNP
IF(N3.GT.MTOT) CALL ERROR (N3-MTOT)

CALL COORD (A(N1),A(N2),NUMNP)

=====
READ SOLUTION TIME AND GROUND ACCELERATION
=====

READ(5,1002)NDT,DT,NPRINT,KP
KST=0 $ TSTART=0.
TAMB=0. $ NTS.EF=0

IF(NDT.EQ.0) NDT = 1
IF(NPRINT.EQ.0) NPRINT = 1
CALL TITLE (HED)
WRITE(6,2001) NDT,DT,NPRINT,KP
NLINE = 22

=====
READ THE EARTHQUAKE ACCELEROGRAMS
=====

READ (5,1003) NBCF,NPTM,NNBC
WRITE(6,2002) NBCF,NPTM
NLINE = NLINE + 9
N4 = N3 + NPTM*NBCF
N5 = N4 + NPTM*NBCF
N6 = N5 + NBCF

```

	IF(N6.GT.MTOT) CALL ERROR (N6-MTOT)	DOTI	94
C		DOTI	95
	CALL FUNC (A(N3),A(N4),A(N5),NP(M))	DOTI	96
C		DOTI	101
68	CALL TITLE (HED)	DOTI	102
NLINE = 10		DOTI	104
70	N7=N6+NUMNP	DOTI	105
	N8=N7+NUMNP	DOTI	106
	N9=N8+NUMNP	DOTI	107
	N10=N9+NUMNP	DOTI	108
	IF(N10.GT.MTOT) CALL ERROR (N10-MTOT)	DOTI	112
		DOTI	115
	*****	DOTI	116
	INITIALISE VEL. POTENTIAL	DOTI	117
	*****	DOTI	118
80	N11 = N10 + NUMNP	DOTI	119
	IF(N11.GT.MTOT) CALL ERROR (N11-MTOT)	DOTI	120
		DOTI	121
	CALL INITIAL (A(N10),TAMB,NUM P)	DOTI	122
		DOTI	123
	FORMAT STATEMENTS	DOTI	124
		DOTI	125
1000	FORMAT(A5,SX,1BA4)	DOTI	126
1001	FORMAT(4I5)	DOTI	127
1002	FORMAT(15,F10.0,2I5)	DOTI	128
1003	FORMAT(3I5)	DOTI	129
2000	FORMAT(20(IH*)/20H CONTROL INFORMATION/20(IH*))///	DOTI	130
1	34H NUMBER OF NODAL POINTS = 15//	DOTI	131
2	34H NUMBER OF ELEMENT GROUPS ... = 15//	DOTI	132
3	34H MAX. ELEMENT GROUP STORAGE .. = 15//	DOTI	133
4	34H SOLUTION MODE = 15//	DOTI	134
4	20H EQ. 0. DATA CHECK/SX,16HEV. 1. EXECUTION///	DOTI	135
2001	FORMAT(50(IH*)/30H SOLUTION TIME AND PRINT.PUNCH	DOTI	136
1	12H INFORMATION/50(IH*))///	DOTI	137
5	48H NUMBER OF SOLUTION TIME STEPS = 15//	DOTI	138
6	43H SOLUTION TIME STEP INCREMENT = F10.4//	DOTI	139
9	48H OUTPUT PRINT INTERVAL = 15//	DOTI	140
8	48H OUTPUT PUNCH INTERVAL = 15///	DOTI	141
2002	FORMAT(25(IH*)/25H TIME DEPENDENT FUNCTIONS/25(IH*))///	DOTI	142
1	48H NUMBER OF TIME DEPENDENT FUNCTIONS = 15//	DOTI	143
2	48H MAXIMUM NUMBER OF (F(T),T) POINTS = 15///	DOTI	144
3000	FORMAT(//51H **ERROR** PROBLEM DECL. MUST BEGIN WITH START CARD)	DOTI	145
3001	FORMAT(//43H **ERROR** NO. OF NODAL POINTS MUST BE .GT. ZERO)	DOTI	146
3002	FORMAT(//51H **ERROR** NO. OF ELEMENT GROUPS MUST BE .GT. ZERO)	DOTI	147
		DOTI	148
	RETURN	DOTI	149
	END	DOTI	150
	SUBROUTINE COORD (X,Y,NUMNP)	DOTI	151
		DOTI	152
		DOTI	153
		DOTI	154
		DOTI	155
		DOTI	156
		DOTI	157
		DOTI	158
		DOTI	159
		DOTI	160
		COORD	1
		COORD	2
	*****	COORD	3
	THIS ROUTINE READS AND GENERATES THE GLOBAL NODAL POINT	COORD	4
	COORDINATE DATA FOR 4- TO 8-NODE ISOPARAMETRIC ELEMENTS	COORD	5
	*****	COORD	6
		COORD	7
	DIMENSION D(1,DY(1))	COORD	8
	COMMON /JUM / HED(18),MTOT,NLINE	COORD	9
		COORD	10
	PEAK OF GENERATE NODAL POINT DATA	COORD	11
		COORD	12
	WRITE(6,2000)	COORD	13
	WRITE(6,2001)	COORD	14
	NLINE = NLINE + 12	COORD	15

	HOLD = 0	COOR	16
C		COOR	17
	10 READ (5,1000) N,X(N),Y(N),KN,JPR	COOR	18
	IF(N.EQ.1) IPR=JPR	COOR	19
	IF(NLINE.LT.55) GO TO 15	COOR	20
	CALL TITLE (HED)	COOR	21
	WRITE(6,2001)	COOR	22
	NLINE = 10	COOR	23
C		COOR	24
	15 WRITE(6,2002) N,X(N),Y(N),KN	COOR	25
	NLINE = NLINE + 1	COOR	26
	IF(NOLD.EQ.0) GO TO 30	COOR	27
C		COOR	28
C	CHECK IF GENERATION IS REQUIRED	COOR	29
C		COOR	30
	IF(KNOLD.EQ.0) GO TO 30	COOR	31
	NUM = (N-NOLD)/KNOLD	COOR	32
	NUMN = NUM - 1	COOR	33
	RNUM = NUM	COOR	34
	DX = (X(N)-X(NOLD))/RNUM	COOR	35
	DY = (Y(N)-Y(NOLD))/RNUM	COOR	36
	K = NOLD	COOR	37
	DO 20 J=1,NUMN	COOR	38
	KK = K	COOR	39
	K = K + KNOLD	COOR	40
	X(K) = X(KK) + DX	COOR	41
	Y(K) = Y(KK) + DY	COOR	42
C		COOR	43
	30 NOLD = N	COOR	44
	KNOLD= KN	COOR	45
	IF(N.NE.NUMNP) GO TO 10	COOR	46
C		COOR	47
	IF(IPR.EQ.1) GO TO 200	COOR	48
C		COOR	49
C	PRINT ALL NODAL POINT DATA	COOR	50
C		COOR	51
	CALL TITLE (HED)	COOR	52
	WRITE(6,2003)	COOR	53
	NLINE = 9	COOR	54
	NROW = NUMNP/3 + 1	COOR	55
	NR = 0	COOR	56
C		COOR	57
	DO 100 I=1,NUMNP,3	COOR	58
	NR = NR + 1	COOR	59
	IP = I + 2	COOR	60
	IF(NR.EQ.NROW) IP = NUMNP	COOR	61
	IF(NLINE.LT.55) GO TO 50	COOR	62
	CALL TITLE (HED)	COOR	63
	WRITE(6,2003)	COOR	64
	NLINE = 9	COOR	65
	50 WRITE(6,2004) (N,X(N),Y(N),N=I,IP)	COOR	66
	100 NLINE = NLINE + 1	COOR	67
C		COOR	68
C	FORMAT STATEMENTS	COOR	69
C		COOR	70
	1000 FORMAT(15,5X,2F10.0,15,11)	COOR	71
	2000 FORMAT(20(11H)/20H NODAL POINT COORDINATE DATA/20(1H*))	COOR	72
	2001 FORMAT(19(1H+)/19H A. INPUT NODE DATA/19(1H*))	COOR	73
	1 4X,4HNODE,5X,7HX-COORD,5X,7HY-COORD,5X,4HDIFF	COOR	74
	2002 FORMAT(3X,15,2F12.3,3X,15)	COOR	75
	2003 FORMAT(23(1H+)/23H B. GENERATED NODE DATA/23(1H*))	COOR	76
	1 3(4X,4HNODE,5X,7HX-COORD,5X,7HY-COORD,5X)	COOR	77
	2004 FORMAT(3(3X,15,2F12.3,5X))	COOR	78

C	200 RETURN	COORD	79
	END	COORD	80
	SUBROUTINE FUNC (TFN,FN,NPTS,NPTM1)	COORD	81
C		FUNC	1
C	*****	FUNC	2
C	*****	FUNC	3
C	*****	FUNC	4
C	*****	FUNC	5
	DIMENSION TFN(NPTM1,1),FN(NPTM1,1),NPTS(1)	FUNC	6
	COMMON /CNTRL2/ KST,NDT,DT,TSTART,TAMB,NPRINT,NTSREF,TIME,KP	FUNC	7
	COMMON /JUNK / HED(18),MTOT,NLINE	FUNC	8
	COMMON /NBC / NNBC,NBCF,NPTM1	FUNC	9
	COMMON /WORK / FORM(4),WORK(196)	FUNC	10
C		FUNC	11
	WRITE(6,2001)	FUNC	12
	NLINE = NLINE + 3	FUNC	13
C		FUNC	14
	DO 100 LL=1,NBCF	FUNC	15
	READ (5,1000) NC,NPTS(NC),FOM,INPUT	FUNC	16
	WRITE(6,2002) NC,NPTS(NC),FOM	FUNC	17
	NLINE = NLINE + 1	FUNC	18
	IF(NPTS(NC).GE.2.AND.NPTS(NC).LE.NPTM1) GO TO 20	FUNC	19
	WRITE(6,3000)	FUNC	20
	STOP	FUNC	21
C		FUNC	22
C	READ TIME FUNCTION VERSUS TIME TABLE	FUNC	23
C		FUNC	24
	20 NT = NPTS(NC)	FUNC	25
	READ (5,1001) (TFN(K,NC),FN(K,NC),K=1,NT)	FUNC	26
C		FUNC	27
C	CHECK THAT TIME POINTS ARE IN INCREASING ORDER	FUNC	28
C		FUNC	29
	TCLO = -1.	FUNC	30
	DO 30 K=1,NT	FUNC	31
	IF(TFN(K,NC).GT.TOLD) GO TO 30	FUNC	32
	WRITE(6,3000)	FUNC	33
	WRITE(6,3001)	FUNC	34
	STOP	FUNC	35
	30 TOLD = TFN(K,NC)	FUNC	36
C		FUNC	37
C		FUNC	45
	50 DO 70 K=1,NT	FUNC	46
	IF(NLINE.LT.55) GO TO 60	FUNC	47
	CALL TITLE (HED)	FUNC	48
	WRITE(6,2000)	FUNC	49
	WRITE(6,2001)	FUNC	50
	NLINE = 10	FUNC	51
	60 WRITE(6,2003) K,TFN(K,NC),FN(K,NC)	FUNC	52
	70 NLINE = NLINE + 1	FUNC	53
C		FUNC	54
C		FUNC	55
C	INTEGRATE THE ACCELEROGRAM TO OBTAIN THE VELOCITY	FUNC	55
	F1=FN(1,NC)	FUNC	56
	FN(1,NC)=0.	FUNC	57
	IF(INPUT.EQ.2) GO TO 55	FUNC	58
	DO 80 Y=2,NT	FUNC	59
	L=K-1	FUNC	60
	F11=F1(K,NC)	FUNC	61
	FN(K,NC)=FN(L,NC)+(F1+F11)*(TFN(K,NC)-TFN(L,NC))*FOM/2.	FUNC	62
	F1=F11	FUNC	63
	WRITE(6,01)Y,TFN(K,NC),FN(K,NC)	FUNC	64
81	FORMAT(60X,15,2F15,5)	FUNC	65
		FUNC	66

```

88  CONTINUE                                FUNC-67
    GO TO 100                              FUNC-68
C                                     FUNC-68
    DIFFERENTIATE DISPLACEMENT TO GET VELOCITY
C                                     FUNC-69
55  DO 110 I=2,NT                          FUNC-70
    J=I-1                                  FUNC-71
    IF(I.EQ.NT) GO TO 111                 FUNC-72
    K=I+1                                  FUNC-73
    A=TFN(I,NC)-TFN(J,NC)                 FUNC-74
    B=TFN(K,NC)-TFN(I,NC)                 FUNC-75
    C=FN(I,NC)                            FUNC-76
    D=(C-F)/A                             FUNC-77
    E=(FN(K,NC)-C)/B                      FUNC-78
    FN(I,NC)=(D+E)/2.                     FUNC-79
    FI=C                                   FUNC-80
    WRITE(6,81)I,TFN(I,NC),FN(I,NC)      FUNC-81
110  CONTINUE                             FUNC-82
111  FN(NT,NC)=(FN(NT,NC)-C)/(TFN(NT,NC)-TFN(J,NC))
    WRITE(6,81)K,TFN(K,NC),FN(K,NC)      FUNC-83
100  CONTINUE                             FUNC-84
C                                     FUNC-85
C                                     FUNC-55
C                                     FUNC-56
C                                     FUNC-57
C                                     FUNC-58
C                                     FUNC-59
    FORMAT STATEMENTS                     FUNC-60
1000  FORMAT(215,F10.0,I5)               FUNC-61
1001  FORMAT(8F10.0)                     FUNC-62
2000  FORMAT(25(1H*)/25H TIME DEPENDENT FUNCTIONS/25(1H*)//)
2001  FORMAT(4X,8HFUNCTION,4X,9HNUMBER OF,6X,10HTIME POINTS,4X,4HTIME,
1      4X,8HFUNCTION/5X,6HNUMBER,4X,11HTIME POINTS,7X,5HNUMBER,
2      5X,5HVALUE,5X,5HVALUE//)
2002  FORMAT(4X,15,7X,15,40X,*MULTIPLICATION FACTOR*,F10.4)
2003  FORMAT(31X,16,F12.3,F12.6)
3000  FORMAT(//49H **ERROR** (NPTS) MUST BE .GE. 2 AND .LE. (NPTM))
3001  FORMAT(//52H **ERROR** BC FUNCTION TIME P NTS ARE OUT OF ORDER)
C                                     FUNC-67
    RETURN                                FUNC-68
    END                                  FUNC-69
    SUBROUTINE INITIAL (T,TAMB,NUMNP)    FUNC-70
C                                     FUNC-71
C                                     INIT 1
C                                     INIT 2
C                                     INIT 3
    INITIALISE V.P. (VELOCITY POTENTIAL) UNIT-4
C                                     INIT 5
C                                     INIT 6
C                                     INIT 7
    DIMENSION T(NUMNP)                   INIT 8
C                                     INIT 9
    ILON=0                               INIT-10
    TAMB=0.                              INIT-12
    DO 100 I=1,NUMNP                     INIT-13
100  T(I) = TAMB                          INIT 35
300  RETURN                              INIT 36
    END                                  ELCL 1
    SUBROUTINE ELCAL                      ELCL 2
C                                     ELCL 3
C                                     ELCL 4
    COMMON /CNTRL1/ NUMNP,NEG,MODEX,NPAR(10),HG,KBC
    COMMON /DIM / N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15
    COMMON /ELSTOR/ NUMEST,MIDEST,MAYEST
    COMMON /JUNK / HED(18),MTOT,NLINE
    COMMON /WORK / NST(10),WORK(190)
    COMMON A(1)
    DIMENSION LABFL(2,2)
    DATA LABEL,SHAXISYM,CMETRIC,6HP L A ,6HN A R /
C                                     ELCL 10
C                                     ELCL 11

```

```

C *****
C THIS ROUTINE CALLS THE APPROPRIATE ELEMENT ROUTINES FOR READING, ELCL 12
C GENERATING AND STORING THE ELEMENT DATA ELCL 13
C ELCL 14
C ELCL 15
C TAPE ALLOCATION ELCL 16
C TAPE 1 - STORES ELEMENT GROUP DATA ELCL 17
C ***** ELCL 18
C TWO DIMENSIONAL FINITE ELEMENTS ELCL 19
C ELCL 20
C   NPAR(1) = 1 ELCL 21
C   NPAR(2) = NUMBER OF TWO DIMENSIONAL ELEMENTS (NEL1) ELCL 22
C   NPAR(3) = NUMBER OF FIRST ELEMENT IN THIS GROUP (MFST) ELCL 23
C   NPAR(4) = ELEMENT TYPE CODE (ITYP2D) ELCL 24
C             EQ.0, AXISYMMETRIC ELCL 25
C   NPAR(5) = MAXIMUM NUMBER OF NODES (MNODS) ELCL 27
C   NPAR(6) = NUMERICAL INTEGRATION ORDER (NINT) ELCL 28
C ***** ELCL 32
C FREE SURFACE ELEMENTS ELCL 33
C ELCL 34
C   NPAR(1) = 2 ELCL 35
C   NPAR(2) = NUMBER OF FREE SURFACE ELEMENTS (NEL2) ELCL 36
C   NPAR(3) = NUMBER OF FIRST ELEMENT IN THIS GROUP (MFST) ELCL 37
C   NPAR(4) = ELEMENT TYPE CODE (ITYP) ELCL 38
C             EQ.0, AXISYMMETRIC FREE SURFACE ELEMENT ELCL 39
C ***** ELCL 42
C SOLID BOUNDARY ELEMENTS ELCL 43
C ELCL 44
C   NPAR(1) = 3 ELCL 45
C   NPAR(2) = NUMBER OF SOLID BOUNDARY ELEMENTS (NEL3) ELCL 46
C   NPAR(3) = NUMBER OF FIRST ELEMENT IN THIS GROUP (MFST) ELCL 47
C   NPAR(4) = ELEMENT TYPE CODE (ITYP) ELCL 47
C             EQ.0, AXISYMMETRIC SOLID-LIQUID BOUNDARY ELEMENTS ELCL 47
C ***** ELCL 48
C ZERO ACTIVE COLUMN HEIGHT ARRAY (MHT) ELCL 49
C ELCL 50
C   M12 = M11 + MHTMP + 1 ELCL 51
C   M13 = M12 + MHTMP ELCL 52
C   IF(M13.GT.MTOT) CALL ERROR (M13-MTOT) ELCL 53
C ELCL 54
C   DO 5 I=M12,M13 ELCL 55
C 5 A(I) = 0.0 ELCL 56
C ELCL 57
C   REWIND 1 ELCL 58
C ELCL 59
C   LOOP OVER ALL ELEMENT GROUPS ELCL 60
C ELCL 61
C   DO 100 NG=1,NEG ELCL 62
C   CALL TITLE (HED) ELCL 63
C   WRITE(6,2000) NG ELCL 64
C   NLINE = 7 ELCL 65
C ELCL 66
C   READ (5,1000) NPAR ELCL 67
C ELCL 68
C   NGR = NPAR(1) ELCL 69
C ELCL 70
C   GO TO (1,2,3) NGR ELCL 71
C ELCL 72
C ----- ELCL 73
C ELEMENT GROUP 1 ELCL 74
C ----- ELCL 75
C ELCL 76
C 1 IF(NPAR(2).GT.0) GO TO 10 ELCL 77
C ELCL 78

```

WRITE(6,3000)	ELCL 79
STOP	ELCL 80
10 IF(NPAR(6).LE.4) GO TO 20	ELCL 81
WRITE(6,3001)	ELCL 82
STOP	ELCL 83
20 IF(NPAR(3).EQ.0) NPAR(3) = 1	ELCL 84
IF(NPAR(5).EQ.0) NPAR(5) = 4	ELCL 85
IF(NPAR(6).EQ.0) NPAR(6) = 2	ELCL 86
IF(NPAR(7).EQ.0) NPAR(7) = 1	ELCL 87
IF(NPAR(8).EQ.0) NPAR(8) = 1	ELCL 88
IT = NPAR(4) + 1	ELCL 89
C	ELCL 90
WRITE(6,2001) NGR,(LABEL(1,IT),I=1,2),NPAR(2),NPAR(3),NPAR(5),	ELCL 91
1 NPAR(6)	ELCL-92
CALL ADK51	ELCL 93
GO TO 50	ELCL 94
C	ELCL 95
-----	ELCL 96
ELEMENT GROUP 2	ELCL 97
-----	ELCL 98
C	ELCL 99
2 IF(NPAR(2).GT.0) GO TO 30	ELCL 100
WRITE(6,3000)	ELCL 101
STOP	ELCL 102
30 IF(NPAR(5).EQ.0) NPAR(5) = 1	ELCL-103
IT = NPAR(4) + 1	ELCL 105
C	ELCL 106
WRITE(6,2002) NGR,(LABEL(1,IT),I=1,2),NPAR(2),NPAR(3)	ELCL-107
CALL ADK52	ELCL 108
GO TO 50	ELCL 109
C	ELCL 110
-----	ELCL 111
ELEMENT GROUP 3	ELCL 112
-----	ELCL 113
C	ELCL 114
3 IF(NPAR(2).GT.0) GO TO 40	ELCL 115
WRITE(6,3000)	ELCL 116
STOP	ELCL 117
40 IF(NPAR(3).EQ.0) NPAR(3) = 1	ELCL 118
C	ELCL 119
I = NPAR(4) + 1	ELCL-107
WRITE(6,2002) NGR,(LABEL(1,IT),I=1,2),NPAR(2),NPAR(3),NPAR(5)	ELCL 107
CALL ADK53	ELCL 122
C	ELCL 123
50 IF(MIDEST.GT.MAXEST) MAXEST = MIDEST	ELCL 124
IF(MIDEST.LE.NUMEST) GO TO 60	ELCL 125
GO TO 100	ELCL 126
C	ELCL 127
STORE ALL ELEMENT GROUP INFORMATION ONTO TAPE 1	ELCL 128
C	ELCL 129
60 WRITE(1) MIDEST,NPAR,NST,(A(I),I=1,MIDEST)	ELCL 130
C	ELCL 131
100 CONTINUE	ELCL 132
C	ELCL 133
IF(MAXEST.LE.NUMEST) GO TO 300	ELCL 134
WRITE(6,3002) MAXEST	ELCL 135
STOP	ELCL 136
C	ELCL 137
FORMAT STATEMENTS	ELCL 138
C	ELCL 139
1000 FORMAT(10I5)	ELCL 140
2000 FORMAT(23(1H*)/20H ELEMENT DATA, GROUP,13/23(1H*)//)	ELCL 141
3001 FORMAT(26H ELEMENT GROUP INDICATOR =13,18H TWO DIMENSIONAL ,2A6,	ELCL 142

```

1      10H ELEMENTS)///
2      38H  NUMBER OF ELEMENTS ..... = 15/
3      38H  NO. OF FIRST ELEMENT IN GROUP .. = 15/
4      38H  MAX. NO. OF NODES PER ELEMENT .. = 15/
5      38H  NUMERICAL INTEGRATION ORDER .... = 15/)
2002 FORMAT(26H ELEMENT GROUP INDICATOR =13.
1      2H (.205.36H) FREE SURFACE ELEMENTS
2      38H  NUMBER OF ELEMENTS ..... = 15/
3      38H  NO. OF FIRST ELEMENT IN GROUP .. = 15/)
3000 FORMAT(//51H **ERROR** NO. OF ELTS. IN GROUP MUST BE .GT. ZERO)
3001 FORMAT(//53H **ERROR** NO. OF INTEGRATION PTS. MUST BE .LE. FOUR)
3002 FORMAT(//41H **ERROR** (NUMEST) MUST BE INCREASED TO 16)
C
300 RETURN
END
SUBROUTINE ADRS1
C
COMMON /CNTRL1/ NUMNP,NEG,MODEX,NPAR(10),NG,KBC
COMMON /DIM / N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15
COMMON /ELSTOR/ NU,NUEST,MIDEST,MAXEST
COMMON /WORK / M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,WORK(190)
COMMON A(1)
*****
B L A N K   C O M M O N   S T O R A G E   A L L O C A T I O N
M1      APPRAY -----DESCRIPTION----- DIMENSION
M2      LM      ELEMENT CONNECTIVITY ARRAY      MNODES*NEL1
M3      XY      ELEMENT COORDINATES              2*MNODES*NEL1
M4      IELT    NO. OF NODES DESCRIBING ELEMENT  NEL1
M5      NOD5    MIDSIDE NODES LOCATION ARRAY     NDSDIM*NEL1
*****
NEL1 = NPAR(2)
MNODS = NPAR(5)
NDM = 2*MNODES
NDSDIM = MNODS-4
M1 = 1
M2 = M1 + MNODS*NEL1
M3 = M2 + NDM*NEL1
M4 = M3 + NEL1
M5 = M4 + NDSDIM*NEL1
NLAST=M5-1
WRITE(6,2000) NLAST
MIDEST = NLAST
IF(NDSDIM.EQ.0) NDSDIM = 1
CALL ELGR1 (A(N1),A(N2),A(N12),A(M1),A(M2),A(M3),A(M4),MNODS,NDM,
NDSDIM)
2000 FORMAT(38H  LENGTH OF ELEMENT INFORMATION .. = 15///
C
RETURN
END
SUBROUTINE ELGR1 (X,Y,MHT,LM,XY,IELT,NOD5,MNODES,NDM,NDSDIM)
C
*****
INPUT INFORMATION FOR 4- TO 8-NODE ISOPARAMETRIC ELEMENTS
*****

```

C	DIMENSION X(1),Y(1),MHT(1),LM(MMNODS,1),XY(NDM,1),IELT(1),	ELG1	7
	1 NODS(NDSDIM,1)	ELG1	8
	COMMON /CNTRL1/ NUMNP,NEG,MODEX,NPAR(10),NG,KBC	ELG1	9
	COMMON /JUNK / HED(10),MTOT,NLINE	ELG1	11
	COMMON /WORK / DUM(10),NOD(8),NODM(8),NODSM(8),WORK(166)	ELG1	12
	DIMENSION AST(2)	ELG1	13
	DATA AST/2H .2H */	ELG1	14
C		ELG1	15
	NEL1 = NPAR(2)	ELG1	16
	MFST = NPAR(3)	ELG1	17
C		ELG1	18
C	=====	ELG1	19
C	=====	ELG1	20
C		ELG1	22
	CALL TITLE (HED)	ELG1	23
C		ELG1	24
C	=====	ELG1	50
C	READ AND GENERATE ELEMENT INFORMATION	ELG1	51
C	=====	ELG1	52
	CALL TITLE (HED)	ELG1	53
	WRITE(6,2003) NG	ELG1	54
	WRITE(6,2004) (I,I=1,8)	ELG1	55
	NLINE = 10	ELG1	56
	N = 1	ELG1	57
	INEM = MFST	ELG1	58
	NLAST = MFST + NEL1 - 1	ELG1	59
C		ELG1	60
	100 READ (5,1002) M,NOD,IEL,KG	ELG1	61
C		ELG1	62
	IF(MTYP.EQ.0) MTYP = 1	ELG1	63
	IF(IEL.EQ.0) IEL = MNODS	ELG1	64
	IF(KG.EQ.0) KG = 1	ELG1	65
	IF(MMNODS.GE.IEL) GO TO 110	ELG1	66
	WRITE(6,3002) M	ELG1	67
	STOP	ELG1	68
C		ELG1	69
	110 IF(M-INEM) 200,120,200	ELG1	70
C		ELG1	71
C	SAVE ELEMENT INFORMATION FOR GENERATION OF ADDITIONAL ELEMENTS	ELG1	72
C		ELG1	73
	120 DO 130 I=1,8	ELG1	74
	130 NUDM(I)=NOD(I)	ELG1	75
	IF(IEL.EQ.4) GO TO 150	ELG1	76
	II=0	ELG1	77
	DO 140 I=5,8	ELG1	78
	NN=NOD(I)	ELG1	79
	IF(NN.EQ.0) GO TO 140	ELG1	80
	II=II+1	ELG1	81
	NODSM(II)=I	ELG1	82
	140 CONTINUE	ELG1	83
C		ELG1	84
	150 IELM = IEL	ELG1	85
	KKK = KG	ELG1	86
	ASTT = AST(1)	ELG1	87
C		ELG1	88
C	STORE PERMANENT ELEMENT INFORMATION	ELG1	89
C		ELG1	90
	200 I2 = 0	ELG1	91
	DO 210 I=1,IELM	ELG1	92
	IF(I.LE.4) GO TO 210	ELG1	93
	JJ = NODSM(I-4)	ELG1	94
		ELG1	95
		ELG1	96
		ELG1	97

	II = NODM(JJ)	ELG1	98
	GO TO 220	ELG1	99
210	II = NODM(I)	ELG1	100
220	LM(I,N) = II	ELG1	101
	I2 = I2 + 2	ELG1	102
	XY(I2-1,N) = X(II)	ELG1	103
230	XY(I2 ,N) = Y(II)	ELG1	104
C		ELG1	105
	IELT(N) = IELM	ELG1	106
	IF(IELM.EQ.4) GO TO 250	ELG1	108
	NN=IELM - 4	ELG1	109
	DO 240 I=1,NN	ELG1	110
240	NOD5(I,N)=NOD5M(I)	ELG1	111
C		ELG1	112
C	UPDATE COLUMN HEIGHTS AND BANDWIDTH	ELG1	113
C		ELG1	114
250	CALL COLHT (MHT,IELM,LM(1,N))	ELG1	115
C		ELG1	116
	IF(NLINE.LT.55) GO TO 260	ELG1	117
	CALL TITLE (HED)	ELG1	118
	WRITE(6,2003) NG	ELG1	119
	WRITE(6,2004) (I,I=1,8)	ELG1	120
	NLINE = 10	ELG1	121
C		ELG1	122
260	WRITE(6,2005) ASTT,IMEM,NODM,IELM	ELG1	123
	NLINE = NLINE + 1	ELG1	124
	IF(IMEM.EQ.NLAST) GO TO 300	ELG1	125
C		ELG1	126
	N = N + 1	ELG1	127
	IMEM = IMEM + 1	ELG1	128
C		ELG1	129
C	CHECK IF ELEMENT DATA IS TO BE STORED FOR CURRENT ELEMENT	ELG1	130
C		ELG1	131
	IF(IMEM.EQ.M) GO TO 120	ELG1	132
C		ELG1	133
C	GENERATE NODE NUMBERS FOR NEXT ELEMENT	ELG1	134
C		ELG1	135
	DO 270 I=1,8	ELG1	136
	IF(NODM(I).EQ.0) GO TO 270	ELG1	137
	NODM(I)=NODM(I)+KKK	ELG1	138
270	CONTINUE	ELG1	139
C		ELG1	140
C	CHECK IF NEXT ELEMENT CARD IS TO BE READ	ELG1	141
C		ELG1	142
	ASTT=AST(2)	ELG1	143
	IF(IMEM.GT.M) GO TO 100	ELG1	144
C		ELG1	145
C	GENERATE INFORMATION FOR NEXT ELEMENT	ELG1	146
C		ELG1	147
	GO TO 200	ELG1	148
C		ELG1	149
280	WRITE(6,3003) M	ELG1	150
	STOP	ELG1	151
C		ELG1	152
C	FORMAT STATEMENTS	ELG1	153
C		ELG1	154
1002	FORMAT(1115)	ELG1	155
2002	FORMAT(30(1H*)/27H ELEMENT INFORMATION, GROUP,13/30(1H*)//)	ELG1	156
2004	FORMAT(4X,4HELT,3X,10(1H-),12HNODE NUMBERS,10(1H-),3X,5H	ELG1	159
1	6HNO, OF/5X,3HNO,3X,8(3X,1),4X,3HNG,5X,5HNODES/)	ELG1	160
2005	FORMAT(A2,15,4X,814,10X,15)	ELG1	171
3002	FORMAT(//10H **ELEMENT,15,34H EXCEEDS MAXIMUM NUMBER OF NODES**)	ELG1	174
3003	FORMAT(//26H **ERROR** ELEMENT CARD =15,16H OUT OF SEQUENCE)	ELG1	175

```

C 300 RETURN
C   END
C   SUBROUTINE ADRS2
C
C   COMMON /CNTRL1/ NUMNP,NEG,MODEX,NPAR(10),NG,KBC
C   COMMON /DIM / N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15
C   COMMON /ELSTOR/ NUMEST,MIDEST,MAXEST
C   COMMON /WORK / M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,WORK(193)
C   COMMON A(1)
C
C   *****
C   BLANK COMMON STORAGE ALLOCATION
C
C   M1  ARRAY  -----DESCRIPTION-----  DIMENSION
C   M2  LM     ELEMENT CONNECTIVITY ARRAY  2*NEL2
C   M3  XX     ELEMENT X-COORDINATES        2*NEL2
C   M4  CL     ELEMENT LENGTHS              NEL2
C
C   *****
C   NEL2 = NPAR(2)
C
C   M1 = 1
C   M2 = M1 + 2*NEL2
C   M3 = M2 + 2*NEL2
C   M4 = M3 + NEL2
C   NLAST=M4-1
C
C   WRITE(6,2000) NLAST
C   MIDEST = NLAST
C
C   CALL ELGR2 (A(N1),A(N2),A(M1),A(M2),A(M3))
C
C 2000 FORMAT(38H      LENGTH OF ELEMENT INFORMATION .. = 15///)
C
C   RETURN
C   END
C   SUBROUTINE ELGR2 (X,Y,LM,XX,CL)
C
C   *****
C   INPUT INFORMATION FOR FREE SURFACE ELEMENTS
C   *****
C
C   DIMENSION X(1),Y(1),LM(2,1),XX(2,1),CL(1)
C
C   COMMON /CNTRL1/ NUMNP,NEG,MODEX,NPAR(10),NG,KBC
C   COMMON /JUNK / HED(10),MTOT,NLINE
C   COMMON /NBC / NNBC,NBCF,NPTM
C   COMMON /WORK / DUM(10),NOD(2),NODM(2),WORK(186)
C   DIMENSION AST(2)
C   DATA AST/2H ,2H */
C
C   NEL2 = NPAR(2)
C   MFST = NPAR(3)
C
C   *****
C   KBC = 0
C
C   *****
C   READ AND GENERATE ELEMENT INFORMATION
C   *****

```


C	N = 1	ELG2 48
	IMEM = MFST	ELG2 49
	NLAST = MFST + NEL2 - 1	ELG2 50
	CALL TITLE (HED)	ELG2 51
	WRITE(6,2003) NG	ELG2 52
	WRITE(6,2004)	ELG2 53
	NLINE = 10	ELG2 54
C		ELG2 55
100	READ (5,1003) M,NOD,KG	ELG2 56
C		ELG2 57
	IF(KG .EQ.0) KG = 1	ELG2 58
	II = NOD(1)	ELG2 59
	JJ = NOD(2)	ELG2 60
C		ELG2 61
	IF(M-IMEM) 200,120,200	ELG2 62
C		ELG2 63
		ELG2 64
C		ELG2 65
	SAVE ELEMENT INFORMATION FOR GENERATION OF ADDITIONAL ELEMENTS	ELG2 66
C		ELG2 67
120	NODM(1) = II	ELG2 68
	NODM(2) = JJ	ELG2 69
	KKK = KG	ELG2 70
	ASTT = AST(1)	ELG2 71
	XL = SQRT((X(JJ)-X(II))*2 + (Y(JJ)-Y(II))*2)	ELG2 72
C		ELG2 73
	STORE PERMANENT ELEMENT INFORMATION	ELG2 74
C		ELG2 75
		ELG2 76
200	DO 230 I=1,2	ELG2 77
	IJ = NODM(I)	ELG2 78
	LM(I,N) = IJ	ELG2 79
230	XX(I,N) = X(IJ)	ELG2 80
C		ELG2 81
	CL(N) = XL	ELG2 82
C		ELG2 83
	IF(NLINE.LT.50) GO TO 250	ELG2 84
	CALL TITLE (HED)	ELG2 85
	WRITE(6,2003) NG	ELG2 86
	WRITE(6,2004)	ELG2 87
	NLINE = 10	ELG2 88
C		ELG2 89
		ELG2 90
250	WRITE(6,2005) ASTT,IMEM,NODM	ELG2 91
	NLINE = NLINE + 1	ELG2 92
	IF(IMEM.EQ.NLAST) GO TO 300	ELG2 93
C		ELG2 94
	N = N + 1	ELG2 95
	IMEM = IMEM + 1	ELG2 96
C		ELG2 97
	CHECK IF ELEMENT DATA IS TO BE STORED FOR CURRENT ELEMENT	ELG2 98
C		ELG2 99
	IF(IMEM.EQ.N) GO TO 120	ELG2 100
C		ELG2 101
	GENERATE NODE NUMBERS FOR NEXT ELEMENT	ELG2 102
C		ELG2 103
	DO 270 I=1,2	ELG2 104
270	NODM(I) = NODM(1) + KKK	ELG2 105
C		ELG2 106
	CHECK IF NEXT ELEMENT CARD IS TO BE READ	ELG2 107
C		ELG2 108
	ASTT = AST(2)	ELG2 109
	IF(IMEM.GT.N) GO TO 100	ELG2 110
C		ELG2 111
	GENERATE INFORMATION FOR NEXT ELEMENT	ELG2 112
C		ELG2 113

```

GO TO 200
200 WRITE(6,3002) M
STOP

FORMAT STATEMENTS

1003 FORMAT(4I5)
2003 FORMAT(30(1H*)/27H ELEMENT INFORMATION, GROUP,13/30(1H*)//)
2004 FORMAT(4X,4HELT,4X,6H1-NODE,4X,6H1-NODE,4X,6H /5X,
1 3HNO,25X,3HNO,/)
2005 FORMAT(A2,15.5X,15.5X,15)
3002 FORMAT(/26H **ERROR** ELEMENT CARD =15,16H OUT OF SEQUENCE)

300 RETURN
END
SUBROUTINE ADR53

COMMON /CNTRL1/ NUMNP,NEG,MODEX,NPAR(10),NG,K9C
COMMON /ELSTOR/ NUHEST,MIDEST,MAXEST
COMMON /DIM / N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15,ADR52
COMMON /WORK / M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,WORK(190)
COMMON A(1)

*****

BLANK COMMON STORAGE ALLOCATION

      ARRAY -----DESCRIPTION----- DIMENSION
M1 LM ELEMENT LOCATION ARRAY NEL3 ADR53 12
M2 XX ELEMENT X-COORDINATES 2*NEL3 ADR53 15
M3 CL ELEMENT LENGTHS NEL3 ADR53 16
M4 SINS SINE OF ANGLE S1 NEL3 ADR53 17
M5 COSS COSINE OF ANGLE S1 NEL3 ADR53 17
*****
NEL3 = NPAR(2)

M1 = 1
M2=M1+2*NEL3
M3=M2+2*NEL3
M4=M3+NEL3
M5=M4+NEL3
M6=M5+NEL3
NLA5T=M6-1
WRITE(6,2000) NLA5T
MIDEST = NLA5T
CALL ELGR3 (A(N1),A(N2),A(M1),A(M2),A(M3),A(M4),A(M5))
2000 FORMAT(3BH LENGTH OF ELEMENT INFORMATION .. = 15///)
RETURN
END
SUBROUTINE ELGR3(X,Y,LM,XX,CL,SINS,COSS)

*****
INPUT INFORMATION FOR SOLID BOUNDARY ELEMENT
*****

DIMENSION X(1),Y(1),LM(2,1),XX(2,1),CL(1),SINS(1),COSS(1)

COMMON /CNTRL1/ NUMNP,NEG,MODEX,NPAR(10),NG,K9C
COMMON /JUNK / HED(18),MTOT,NLINE
COMMON /NBC / NBBC,NBCF,NPTM

ELG2 10
ELG2 11
ELG2 116
ELG2 117
ELG2 118
ELG2 119
ELG2 120
ELG2 121
ELG2 133
ELG2 134
ELG2 135
ELG2 136
ELG2 139
ELG2 140
ELG2 141
ELG2 142
ADR53 1
ADR53 2
ADR53 3
ADR53 4
ADR53 4
ADR53 6
ADR53 7
ADR53 8
ADR53 9
ADR53 10
ADR53 11
ADR53 12
ADR53 13
ADR53 14
ADR53 15
ADR53 16
ADR53 17
ADR53 17
ADR53 20
ADR53 21
ADR53 22
ADR53 23
ADR53 24
ADR53 25
ADR53 26
ADR53 27
ADR53 28
ADR53 28
ADR53 29
ADR53 29
ADR53 30
ADR53 31
ADR53 32
ADR53 33
ADR53 34
ADR53 42
ADR53 43
ELG3 1
ELG2 2
ELG2 3
ELG3 4
ELG2 5
ELG2 6
ELG2 7
ELG3 7
ELG2 9
ELG2 10
ELG2 11
ELG2 12

```

	COMMON /WORK / DUM(18),NOD(2),NODM(2),WORK(186)	ELG2 13
	DIMENSION AST(2)	ELG2 14
	DATA AST/2H ,2H */	ELG2 15
C	NEL3=NPAR(2)	ELG2 16
	MFST = NPAR(3)	ELG2 17
C		ELG2 18
C	-----	ELG2 19
	KBC = 0	ELG2 20
C		ELG2 24
C	-----	ELG2 44
C	READ AND GENERATE ELEMENT INFORMATION	ELG2 45
C	-----	ELG2 46
	N = 1	ELG2 47
	INEM = MFST	ELG2 48
	NLAST=MFST+NEL3-1	ELG2 49
	CALL TITLE (HED)	ELG2 50
	WRITE(6,2003) NG	ELG2 51
	WRITE(6,2004)	ELG2 52
	NLINE = 10	ELG2 53
		ELG2 54
C		ELG2 55
100	READ(5,1003)M,NOD,KG,CS,SS	ELG2 56
C		ELG2 57
	IF(KG .EQ.0) KG = 1	ELG2 58
	II = NOD(1)	ELG2 60
	JJ = NOD(2)	ELG2 61
C		ELG2 62
	IF(M-INEM) 200,120,200	ELG2 63
C		ELG2 64
C	SAVE ELEMENT INFORMATION FOR GENERATION OF ADDITIONAL ELEMENTS	ELG2 65
C		ELG2 66
120	NODM(1) = II	ELG2 67
	NODM(2) = JJ	ELG2 68
	YD=ABS(Y(II)-Y(JJ))	ELG2 69
	IF(YD.GT.0.00000001) GO TO 5	ELG2 70
	SI=999999999.9	ELG2 71
	GO TO 6	ELG2 72
5	SI=(X(II)-X(JJ))/(Y(II)-Y(JJ))	ELG2 73
6	KKK=KG	ELG2 74
	ASTT = AST(1)	ELG2 75
	XL = SORT((X(JJ)-X(II))*2 + (Y(JJ)-Y(II))*2)	ELG2 76
C		ELG2 77
C	STORE PERMANENT ELEMENT INFORMATION	ELG2 78
C		ELG2 79
200	DO 230 I=1,2	ELG2 80
	IJ = NODM(I)	ELG2 81
	LI(I,N) = IJ	ELG2 82
230	XX(I,N) = X(IJ)	ELG2 83
C		ELG2 84
	CL(N) = XL	ELG2 85
	SI=ATAN(SI)	ELG2 86
	SINS(I)=SIN(SI)	ELG2 87
	COSS(I)=COS(SI)	ELG2 88
	COSS(I)=CS	ELG2 89
	SINS(I)=SS	ELG2 90
C		ELG2 84
	IF(NLINE.LT.55) GO TO 250	ELG2 85
	CALL TITLE (HED)	ELG2 86
	WRITE(6,2003) NG	ELG2 87
	WRITE(6,2004)	ELG2 88
	NLINE = 10	ELG2 89
250	WRITE(6,2005) ASTT,INEM,NODM	ELG2 90

```

      NLINE = NLINE + 1
      IF (IMEM.EQ.NLAST) GO TO 300
C
      N = N + 1
      IMEM = IMEM + 1
C
      CHECK IF ELEMENT DATA IS TO BE STORED FOR CURRENT ELEMENT
C
      IF (IMEM.EQ.M) GO TO 120
C
      GENERATE NODE NUMBERS FOR NEXT ELEMENT
C
      DO 270 I=1,2
270  NODM(I) = NODM(I) + KKK
C
      CHECK IF NEXT ELEMENT CARD IS TO BE READ
C
      ASTT = AST(2)
      IF (IMEM.GT.M) GO TO 100
C
      GENERATE INFORMATION FOR NEXT ELEMENT
C
      GO TO 200
C
280  WRITE(6,3002) M
      STOP
C
      FORMAT STATEMENTS
C
1003  FORMAT(4I5,2F10.0)
2003  FORMAT(30(1H*)/27H ELEMENT INFORMATION, GROUP,13/30(1H*)//)
2004  FORMAT(4X,4HELT.,4X,6HI-NODE,4X,6HJ-NODE,4X,5HDIREC/5X,
1      3HNO.,25X,6HCOSINE/)
2005  FORMAT(A2,I5,5X,I5,5X,I5)
3002  FORMAT(/26H **ERROR** ELEMENT CARD #15,16H OUT OF SEQUENCE)
C
300  RETURN
      END
      SUBROUTINE COLHT (MHT,ND,LM)
C
      DIMENSION LM(1),MHT(1)
C
      FIND SMALLEST GLOBAL NODE NUMBER (LS) FOR ELEMENT
C
      LS=100000
      DO 100 I=1,ND
      IF (LM(I)) 80,100,100
80  IF (LM(I)-LS) 90,100,100
90  LS=LM(I)
100  CONTINUE
C
      COMPUTE COLUMN HEIGHT ABOVE DIAGONAL (ME) AND CHECK IF MAXIMUM
C
      DO 200 I=1,ND
      II=LM(I)
      IF (II.EQ.0) GO TO 200
      ME=II - LS
      IF (ME.GT.MHT(II)) MHT(II)=ME
200  CONTINUE
C
      RETURN
      END
      SUBROUTINE ERROR (N)

```

```

ELG2  92
ELG2  93
ELG2  94
ELG2  95
ELG2  96
ELG2  97
ELG2  98
ELG2  99
ELG2 100
ELG2 101
ELG2 102
ELG2 103
ELG2 104
ELG2 105
ELG2 106
ELG2 107
ELG2 108
ELG2 109
ELG2 110
ELG2 111
ELG2 112
ELG2 113
ELG2 114
ELG2 115
ELG2 116
ELG2 117
ELG2 118
ELG2 119
ELG2 120
ELG2 121
ELG2 133
ELG1-121
ELG2-135
ELG2-136
ELG2 139
ELG2 140
ELG2 141
ELG2 142
CLHT  1
CLHT  2
CLHT  3
CLHT  4
CLHT  5
CLHT  6
CLHT  7
CLHT  8
CLHT  9
CLHT 10
CLHT 11
CLHT 12
CLHT 13
CLHT 14
CLHT 15
CLHT 16
CLHT 17
CLHT 18
CLHT 19
CLHT 20
CLHT 21
CLHT 22
CLHT 23
CLHT 24
ERR   1

```

C	WRITE(6,2000) N	ERR	2
2000	FORMAT(/31H **ERROR** STORAGE EXCEEDED BY 16)	ERR	3
	STOP	ERR	4
	END	ERR	5
	SUBROUTINE TITLE (HED)	ERR	6
C		TITL	1
	DIMENSION HED(18)	TITL	2
C		TITL	3
	THIS ROUTINE PRINTS THE TITLE CARD AT TOP OF OUTPUT PAGE	TITL	4
C		TITL	5
	WRITE(6,2000) HED	TITL	6
2000	FORMAT(1H1,18A4.39X,8HDOT 1976/)	TITL	7
	RETURN	TITL	8
	END	TITL	9
	SUBROUTINE ADRSK (MAXA,MHT,NUMNP,NLWK,MA)	TITL	10
C		ADSK	1
	*****	ADSK	2
C		ADSK	3
	TO CALCULATE ADDRESSES OF DIAGONAL ELEMENTS IN A	ADSK	4
C		ADSK	5
	BANDED MATRIX WHOSE COLUMN HEIGHTS ARE KNOWN.	ADSK	6
C		ADSK	7
	MA = MAXIMUM BAND WIDTH	ADSK	8
C		ADSK	9
	MHT = ACTIVE COLUMN HEIGHTS ABOVE DIAGONAL	ADSK	10
C		ADSK	11
	MAXA = ADDRESSES OF DIAGONAL ELEMENTS	ADSK	12
C		ADSK	13
	NLWK = MAXIMUM STORAGE REQUIRED	ADSK	14
C		ADSK	15
	*****	ADSK	16
C		ADSK	17
	DIMENSION MAXA(1),MHT(1)	ADSK	18
C		ADSK	19
	MAXA(1) = 1	ADSK	20
	MAXA(2) = 2	ADSK	21
	MA = 0	ADSK	22
	IF (NUMNP.EQ.1) GO TO 100	ADSK	23
	DO 10 I=2,NUMNP	ADSK	24
	IF (MHT(I).GT.MA) MA = MHT(I)	ADSK	25
10	MAXA(I+1) = MAXA(I) + MHT(I) + 1	ADSK	26
100	MA = MA + 1	ADSK	27
	NLWK = MAXA(NUMNP+1) - 1	ADSK	28
C		ADSK	29
	RETURN	ADSK	30
	END	ADSK	31
	SUBROUTINE ASSEM	ASME	1
C		ASME	2
	*****	ASME	3
C		ASME	4
	ASSEMBLE THE EFFECTIVE SYSTEM STIFFNESS MATRIX (K*)	ASME	5
C		ASME	6
	*****	ASME	7
C		ASME	8
	COMMON /CNTRL/ NUMNP,NEG,MODEX,NPAR(10),HC,KBC	ASME	9
	COMMON /DIM / N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	ASME	10
	COMMON /WORK / M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,WORK(19)	ASME	11
	COMMON /A(1)	ASME	12
	DIMENSION HST(10)	ASME	13
	EQUIVALENCE (HST(1),M1)	ASME	14
C		ASME	15
	REWIND 1	ASME	16
C		ASME	17
	REWIND 2	ASME	18
C		ASME	19
	LOOP OVER ALL ELEMENT GROUPS	ASME	20
C		ASME	21
	DO 100 HG=1,NEG	ASME	22
C		ASME	23
	READ (1) MIDEST,NPAR,NST,(A(I),I=1,MIDEST)	ASME	24
C		ASME	25

C	NGR = NPAR(1)	ASMK	23
C	GO TO (1,2,3) NGR	ASMK	24
C	-----	ASMK	25
C	ELEMENT GROUP 1	ASMK	26
C	-----	ASMK	27
C	1 MNODE = NPAR(5)	ASMK	28
C	NDM = 2*MNODES	ASMK	29
C	NDSDIM = MNODES-4	ASMK	30
C	IF (NDSDIM.EQ.0) NDSDIM = 1	ASMK	31
C	CALL COND1 (A(M1),A(M2),A(M3),A(M4),A(M5),1,MNODES,NDM,NDSDIM)	ASMK	32
C	GO TO 100	ASMK	33
C	-----	ASMK	34
C	ELEMENT GROUP 2	ASMK	35
C	-----	ASMK	36
C	2 CALL COND2 (A(M1),A(M2),A(M3))	ASMK	37
C	GO TO 100	ASMK	38
C	-----	ASMK	39
C	ELEMENT GROUP 3	ASMK	40
C	-----	ASMK	41
C	3 CONTINUE	ASMK	42
C	100 CONTINUE	ASMK	43
C	-----	ASMK	44
C	RETURN	ASMK	45
C	END	ASMK	46
C	SUBROUTINE COND1 (LM,XY,IELT,NODES,T,MNODES,NDM,NDSDIM)	ASMK	47
C	CON1 = 1	ASMK	48
C	CON1 = 3	ASMK	49
C	*****	ASMK	50
C	FORM THE EFFECTIVE SYSTEM STIFFNESS MATRIX (K*)	CON1	51
C	*****	CON1	52
C	*****	CON1	53
C	DIMENSION LM(MNODES,1),XY(NDM,1),IELT(1),NODES(MNODES,1),T(1)	CON1	54
C	COMMON /CHTFL1/ HURFIP,NEG,MODEX,NPAR(10),NG,NBC	CON1	55
C	COMMON /CHTFL2/ KST,NDT,DT,TSTART,TAB,NPRINT,NTSPEF,T	CON1	56
C	COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	CON1	57
C	COMMON /TODIM/ NEL,NODES,NTYPE,MNODES	CON1	58
C	COMMON /WORK/ DUM(10),SK(64),SC(8),HF(8),THOD(8),WOR=100	CON1	59
C	COMMON A(1)	CON1	60
C	NEL1 = NPAR(2)	CON1	61
C	DO 100 N=1,NEL1	CON1	62
C	NEL = N	CON1	63
C	NODES = IELT(N)	CON1	64
C	MNODES = NODES - 4	CON1	65
C	NDOF = NODES*MNODES	CON1	66
C	ZERO ELEMENT STIFFNESS MATRIX SK(NODES,NODES)	CON1	67
C	DO 10 I=1,NDOF	CON1	68
C	10 SK(I) = 0.0	CON1	69
C	ZERO ELEMENT MASS VECTOR SC(NODES)	CON1	70
C	DO 40 I=1,NODES	CON1	71
C	40 SC(I) = 0.0	CON1	72
C		CON1	73
C		CON1	74
C		CON1	75

```

C      FORM ELEMENT STIFFNESS MATRIX USING GAUSS QUADRATURE          CON1- 52
C      CALL FORMK1 (SK,SC,XY(1,N),NOD5(1,N),NOD5)                  CON1- 53
C      CALL FORMK1 (SK,SC,XY(1,N),NOD5(1,N),NOD5)                  CON1- 54
C      ASSEMBLE EFFECTIVE ELEMENT STIFFNESS MATRIX (SK*)           CON1- 59
C      INTO EFFECTIVE STRUCTURAL STIFFNESS MATRIX (K*)              CON1- 61
C      70 CALL ADDBAN (A(N10),A(N9),SK,LM(1,N),NOD5)                CON1- 62
C      100 CONTINUE                                                  CON1- 63
C      RETURN                                                         CON1- 73
C      END                                                            CON1- 74
C      SUBROUTINE FORMK1 (SK,SC,XY,NOD5,NOD5)                       CON1- 75
C      FMK1- 1
C      FMK1- 3
C      FMK1- 4
C      FMK1- 5
C      FMK1- 6
C      FMK1- 7
C      FMK1- 8
C      FMK1- 9
C      FMK1- 11
C      FMK1- 12
C      FMK1- 13
C      FMK1- 14
C      FMK1- 15
C      FMK1- 16
C      FMK1- 17
C      FMK1- 18
C      FMK1- 19
C      FMK1- 20
C      FMK1- 21
C      FMK1- 22
C      FMK1- 23
C      FMK1- 24
C      FMK1- 25
C      FMK1- 27
C      FMK1- 28
C      FMK1- 29
C      FMK1- 30
C      FMK1- 31
C      FMK1- 32
C      FMK1- 33
C      FMK1- 34
C      FMK1- 35
C      FMK1- 36
C      FMK1- 37
C      FMK1- 38
C      FMK1- 39
C      FMK1- 40
C      FMK1- 41
C      FMK1- 42
C      FMK1- 43
C      FMK1- 44
C      FMK1- 45
C      FMK1- 46
C      FMK1- 47
C      FMK1- 48
C      FMK1- 49
C      FMK1- 50
C      FMK1- 77
C      FMK1- 78
C      FMK1- 79
C      FMK1- 80

```

 ISOPARAMETRIC FORMULATION OF ELEMENT STIFFNESS MATRIX
 FOR AXI-SYMMETRIC ELEMENT

 DIMENSION SK(NOD5,NOD5),SC(NOD5),XY(1,NOD5(1))
 COMMON /CNTRL1/ NUMNP,NEG,MODEX,NPAR(10),NG,KBC
 COMMON /CNTRL2/ KST,NDT,DT,TSTART,TAMB,NPRINT,NTSREF,TIME,KP
 COMMON /WORK / DUM(98),H(8),P(2,8),B(2,8),XJ(2,2),EK(3),WORK(55)
 DIMENSION XG(4,4),WGT(4,4)
 DATA XG / 0., 0., 0., 0.,
 1 -.5773502691896, .5773502691896, 0., 0.,
 2 -.7745966692415, .0000000000000, .7745966692415, 0.,
 3 -.8611363115941, -.3399810435849, .3399810435849, .8611363115941/
 DATA WGT / 2.000, 0., 0., 0.,
 1 1.0000000000000, 1.0000000000000, 0., 0.,
 2 .5555555555556, .8888888888889, .5555555555556, 0.,
 3 .3476540451375, .6521451548625, .6521451548625, .3476540451375/
 ITYP2D = NPAR(4)
 NINT = NPAR(6)
 LOOP OVER ALL INTEGRATION POINTS
 VOL = 0.0
 DO 100 LX=1,NINT
 R = XG(LX,NINT)
 DO 100 LY=1,NINT
 S = XG(LY,NINT)
 WT = WGT(LX,NINT)*WGT(LY,NINT)
 FIND INTERPOLATION FUNCTIONS (H) AND THEIR DERIVATIVES (P).
 FIND JACOBIAN (XJ) AND ITS DETERMINANT (DETJ).
 CALL SHAPE1 (R,S,XY,H,P,NOD5,XJ,DETJ)
 EVALUATE JACOBIAN INVERSE (XJI) AND GLOBAL DERIVATIVE OPERATOR (B)
 AT EACH INTEGRATION POINT (R,S) WITHIN THE ELEMENT
 CALL DERIV1 (XJ,H,P,B,XJ,DETJ,RAD,ITYP2D)
 FAC = WT*RAD*DETJ
 VOL = VOL + FAC
 FORM SK = B(TRANPOSE)*EK*Q FOR INTEGRATION POINT (R,S)
 DO 50 I=1,NOD5
 BT1 = B(1,I)

```

      BT2 = B(2,1)
      DO 50 J=1,NODS
50    SK(I,J)=SK(I,J)+(BT1*B(1,J)+BT2*B(2,J))*FAC
      C
      100 CONTINUE
      C
      NODM = NODS - 1
      DO 200 I=1,NODM
      II = I + 1
      DO 200 J=II,NODS
200    SK(J,I) = SK(I,J)
      C
      600 RETURN
      END
      SUBROUTINE SHAPE1 (R,S,XY,H,P,NODS,XJ,DETJ)
      SHP1 1
      SHP1 2
      SHP1 3
      1. TO FIND INTERPOLATION FUNCTIONS ( H )
      SHP1 4
      AND DERIVATIVES ( P ) CORRESPONDING TO THE NODAL POINTS
      SHP1 5
      OF A 4- TO 8-NODE TWO DIMENSIONAL ISOPARAMETRIC ELEMENT
      SHP1 6
      2. TO FIND JACOBIAN ( XJ ) AND ITS DETERMINANT ( DETJ )
      SHP1 7
      SHP1 8
      SHP1 9
      SHP1 10
      SHP1 11
      SHP1 12
      SHP1 13
      SHP1 14
      SHP1 15
      SHP1 16
      SHP1 17
      SHP1 18
      SHP1 19
      SHP1 20
      SHP1 21
      SHP1 22
      SHP1 23
      SHP1 24
      SHP1 25
      SHP1 26
      SHP1 27
      SHP1 28
      SHP1 29
      SHP1 30
      SHP1 31
      SHP1 32
      SHP1 33
      SHP1 34
      SHP1 35
      SHP1 36
      SHP1 37
      SHP1 38
      SHP1 39
      SHP1 40
      SHP1 41
      SHP1 42
      SHP1 43
      SHP1 44
      SHP1 45
      SHP1 46
      SHP1 47
      SHP1 48
      SHP1 49
      *****
      NODE NUMBERING CONVENTION
      2          5          1
      0 . . . . . 0 . . . . . 0
      .          .          .
      .          5          .
      .          .          .
      6 0          . . . R          0 8
      .          .          .
      .          .          .
      0 . . . . . 0 . . . . . 0
      3          7          4
      *****
      DIMENSION XY(2,1),H(1),P(2,1),NODS(1),XJ(2,2)
      COMMON /TODIM / NEL,NODS,MTYPE,NND5
      DIMENSION IPERM(4)
      DATA IPERM/2,3,4,1/
      C
      INTERPOLATION FUNCTIONS (4-NODE ELEMENT)
      RP = 1.0 + R
      SP = 1.0 + S
      KM = 1.0 - R
      SM = 1.0 - S
      R2 = 1.0 - R*R
      S2 = 1.0 - S*S
      C
      H(1) = 0.25*RP*SP
      H(2) = 0.25*RM*SM
      H(3) = 0.25*RM*SM

```


C	H(4) = 0.25*RP*SM	SHP1	56
C	LOCAL DERIVATIVES OF INTERPOLATION FUNCTIONS (4-NODE ELEMENT)	SHP1	57
C	P(1,1) = 0.25*SP	SHP1	58
C	P(1,2) = -P(1,1)	SHP1	59
C	P(1,3) = -0.25*SM	SHP1	60
C	P(1,4) = -P(1,3)	SHP1	61
C	P(2,1) = 0.25*RP	SHP1	62
C	P(2,2) = 0.25*RM	SHP1	63
C	P(2,3) = -P(2,2)	SHP1	64
C	P(2,4) = -P(2,1)	SHP1	65
C	INTERPOLATION FUNCTIONS AND LOCAL DERIVATIVES FOR MIDDLE NODES	SHP1	66
C	IF(NODS.EQ.4) GO TO 50	SHP1	67
C	I = 0	SHP1	68
C	2 I = I + 1	SHP1	69
C	IF (I.GT.NND5) GO TO 40	SHP1	70
C	NN = NOD5(I) - 4	SHP1	71
C	GO TO (5,6,7,8) NN	SHP1	72
C	5 H(5) = 0.50*R2*SP	SHP1	73
C	P(1,5) = -R*SP	SHP1	74
C	P(2,5) = 0.50*R2	SHP1	75
C	GO TO 2	SHP1	76
C	6 H(6) = 0.50*RM*SM	SHP1	77
C	P(1,6) = -0.50*S2	SHP1	78
C	P(2,6) = -RM*S	SHP1	79
C	GO TO 2	SHP1	80
C	7 H(7) = 0.50*R2*SM	SHP1	81
C	P(1,7) = -R*SM	SHP1	82
C	P(2,7) = -0.50*R2	SHP1	83
C	GO TO 2	SHP1	84
C	8 H(8) = 0.50*RP*S2	SHP1	85
C	P(1,8) = 0.50*S2	SHP1	86
C	P(2,8) = -RP*S	SHP1	87
C	GO TO 2	SHP1	88
C	MODIFY INTERPOLATION FUNCTIONS H(1) TO H(4) AND LOCAL DERIVATIVES	SHP1	89
C	40 IH = 0	SHP1	90
C	41 IH = IH + 1	SHP1	91
C	IF (IH.GT.NND5) GO TO 50	SHP1	92
C	IH = NOD5(IH)	SHP1	93
C	I1 = IH - 4	SHP1	94
C	I2 = IPERH(I1)	SHP1	95
C	H(I1) = H(I1) - 0.5*H(IH)	SHP1	96
C	H(I2) = H(I2) - 0.5*H(IH)	SHP1	97
C	H(IH+4) = H(IH)	SHP1	98
C	DO 45 J=1,2	SHP1	99
C	P(J,I1) = P(J,I1) - 0.5*P(J,IH)	SHP1	100
C	P(J,I2) = P(J,I2) - 0.5*P(J,IH)	SHP1	101
C	45 P(J,IH+4) = P(J,IH)	SHP1	102
C	GO TO 41	SHP1	103
C	EVALUATE THE JACOBIAN MATRIX AT POINT (R,S)	SHP1	104
C	50 DO 100 I=1,2	SHP1	105
C	DO 100 J=1,2	SHP1	106
C	SUM = 0.0	SHP1	107
C	DO 90 K=1,NODS	SHP1	108
C		SHP1	109
C		SHP1	110
C		SHP1	111
C		SHP1	112

```

      90 SUM = SUM + P(I,K)* XY(J,K)
      100 XJ(I,J) = SUM
      COMPUTE THE DETERMINANT OF THE JACOBIAN MATRIX AT POINT (R,S)
      DETJ = XJ(1,1)* XJ(2,2) - XJ(2,1)* XJ(1,2)
      DUM = ABS(DE TJ)
      IF(DUM.GT.1.0E-8) GO TO 500
      WRITE (6,3000) NEL
      STOP
C 3000 FORMAT(//49H **ERROR** ZERO JACOBIAN DETERMINANT FOR ELEMENT,15)
C 500 RETURN
      END
      SUBROUTINE DERIV1 (XY,H,P,B,XJ,DETJ,RAD,ITYP2D)
      DIMENSION XY(2,1),H(1),P(2,1),B(2,1),XJ(2,2)
      COMMON /TODIM / NEL,NODS,MTYPE,NNDS
      COMMON /WORK / DUM(145),XJ(2,2),WORK(51)
      COMPUTE INVERSE OF THE JACOBIAN MATRIX
      DETJI = 1.0/DETJ
      XJ1(1,1) = XJ(2,2)* DETJI
      XJ1(1,2) = -XJ(1,2)* DETJI
      XJ1(2,1) = -XJ(2,1)* DETJI
      XJ1(2,2) = XJ(1,1)* DETJI
      EVALUATE GLOBAL DERIVATIVE OPERATOR ( B-MATRIX )
      DO 10 K=1,NODS
      B(1,K) = XJ1(1,1)*P(1,K) + XJ1(1,2)*P(2,K)
      10 B(2,K) = XJ1(2,1)*P(1,K) + XJ1(2,2)*P(2,K)
      RAD = 1.0
      IF(ITYP2D.NE.0) GO TO 500
      COMPUTE THE RADIUS AT POINT (R,S) FOR AXISYMMETRIC SOLID
      RAD = 0.0
      DO 50 K=1,NODS
      50 RAD = RAD + H(K)* XY(1,K)
      IF(RAD.GT.1.0E-8) GO TO 500
      WRITE (6,3000) NEL
      STOP
C 3000 FORMAT(//50H **ERROR** ZERO RADIUS ENCOUNTERED IN ELEMENT NO.,15)
C 500 RETURN
      END
      SUBROUTINE ADDBAN (A,MAXA,S,LM,NDOF)
      ASSEMBLE ELEMENT STIFFNESS INTO COMPACTED GLOBAL STIFFNESS

```


	B=B + C*A(KK)	COLS	49
220	A(KK)=C	COLS	50
	A(KN)=A(KN) - B	COLS	51
C		COLS	52
	IF (A(KN)) 222,224,226	COLS	53
222	WRITE (6,3000) N	COLS	54
	STOP	COLS	55
224	WRITE (6,3001) N	COLS	56
	STOP	COLS	57
226	NR=MIN0(MA1,NN-N)	COLS	58
	IF (NR) 200,200,228	COLS	59
228	NN=KU - KL + 1	COLS	60
C		COLS	61
	DO 240 J=1,NR	COLS	62
	MJ=MAXA(N+J) + J	COLS	63
	MNJ=MAXA(N+J+1) - MJ - 1	COLS	64
	IF (MNJ) 240,240,230	COLS	65
230	ND=MIN0(MN,MNJ)	COLS	66
	C=0.	COLS	67
	KU=KN + ND	COLS	68
	IC=MJ - KN	COLS	69
	DO 300 KK=KL,KU	COLS	70
300	C=C + A(KK)*A(KK+IC)	COLS	71
	A(KN+IC)=A(KN+IC) - C	COLS	72
240	CONTINUE	COLS	73
C		COLS	74
200	CONTINUE	COLS	75
	IF(KKK.EQ.0) RETURN	COLS	76
C		COLS	77
C	=====	COLS	78
C	FORWARD REDUCTION	COLS	79
C	=====	COLS	80
C		COLS	81
700	DO 400 N=2,NN	COLS	82
	KL=MAXA(N) + 1	COLS	83
	KU=MAXA(N+1) - 1	COLS	84
	IF (KU-KL) 400,410,410	COLS	85
410	K=N	COLS	86
	C=0.	COLS	87
	DO 420 KK=KL,KU	COLS	88
	K=K - 1	COLS	89
420	C=C + A(KK)*V(K)	COLS	90
	V(N)=V(N) - C	COLS	91
400	CONTINUE	COLS	92
	GO TO 800	COLS	93
C		COLS	94
C	=====	COLS	95
C	BACK SUBSTITUTION	COLS	96
C	=====	COLS	97
C		COLS	98
800	DO 480 N=1,NN	COLS	99
	K=MAXA(N)	COLS	100
480	V(N)=V(N)/A(K)	COLS	101
	IF (NN.EQ.1) RETURN	COLS	102
	H=NN	COLS	103
	DO 500 L=2,NN	COLS	104
	PL=MAXA(H) + 1	COLS	105
	KU=MAXA(H+1) - 1	COLS	106
	IF (KU-PL) 500,510,510	COLS	107
510	K=H	COLS	108
	DO 520 KK=PL,KU	COLS	109
	K=K - 1	COLS	110
520	V(K)=V(K) - A(KK)*V(N)	COLS	111

```

500 N=N - 1
      FORMAT STATEMENTS
3000 FORMAT(//45H **STOP** STIFFNESS NOT POSITIVE DEFINITE.../,
1      10X,27H NEGATIVE PIVOT IN POSITION 14)
3001 FORMAT(//33H **STOP** ZERO PIVOT IN POSITION 14)

      RETURN
      END
      SUBROUTINE FORMQC(TTH)
      *****
      CALCULATE THE LOADING VECTOR
      *****
      COMMON /CNTRL1/ NUHNP,NEG,MODEX,NPAR(10),NG,KBC
      COMMON /CNTRL2/ KST,NDT,DT,TSTART,TAMB,NPRINT,NTSREF,TIME,KP
      COMMON /DIM / M1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15
      COMMON /NBC / NNBC,NBCF,NPTM
      COMMON /WORK / M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,WORK(190)
      COMMON /R(1)
      DIMENSION NST(10)
      EQUIVALENCE (NST(1),M1)

      REWIND 1
      REWIND 2

      LOOP OVER ALL ELEMENT GROUPS

      DO 100 NG=1,NEG

      READ (1) MIDEST,NPAR,NST,(A(I),I=1,MIDEST)

      NGR = NPAR(1)

      IF(NGR.NE.3) GO TO 100
      CALL FLUX2 (A(N1),A(N2),A(N3),A(N4),A(N5),A(N1),A(N2),A(N3),A(N8),
1A(N1),A(N14),NPTM,TTH)
100 CONTINUE

      RETURN
      END
      SUBROUTINE FLUX2 (LM,XX,CL,SINS,COSS,TFN,FN,NPTS,T,Q,TT,NPTM,TTH) FLX2=
      COMMON/CONST /A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,THETA,DELTA,ALPHA,PI,FLX2=
      +,RO FLX2=
      *****
      CALCULATE THE LOADING VECTOR
      *****
      COMMON /CNTRL1/ NUHNP,NEG,MODEX,NPAR(10),NG,KBC
      COMMON /CNTRL2/ KST,NDT,DT,TSTART,TAMB,NPRINT,NTSREF,TIME,KP
      COMMON /NBC / NNBC,NBCF,IDENTITY
      DIMENSION LM(2,1),XX(2,1),CL(1),SINS(1),COSS(1),T(1),TFN(NPTM,1),FLX2=
      IF(NPTM,1),NPTS(1),Q(1),TT(1) FLX2=
      NEL3=HPAR(2) FLX2=
      ITP = NPAR(4) FLX2=
      DO 100 N=1,NEL3 FLX2=
      II = LM(1,N) FLX2=
      JJ = LM(2,N) FLX2=

```

```

      XL = CL(N)
      NPT=NPTS(1)
      NC=1
      CALL INTERP(TFN(1,NC),FN(1,NC),NPT,TTH,VX)
      IF(NBCF.NE.1) GO TO 21
      VZ=0.
      GO TO 22
21    NC=2
      NPT=NPTS(2)
      CALL INTERP(TFN(1,NC),FN(1,NC),NPT,TTH,VZ)
22    VN=VX*COSS(N)+VZ*SINS(N)
      C
      C
      25  XI = XX(1,N)
          XJ = XX(2,N)
          Q(I1)=Q(I1)+ VN*XL*(2.*XI+XJ)/6.0
          Q(JJ)=Q(JJ)+ VN*XL*(XI+2.*XJ)/6.0
100  CONTINUE
      C
      RETURN
      END
      SUBROUTINE INTERP (TFN,FN,NPT,TIME,VAL)
      C
      C *****
      C THIS ROUTINE INTERPOLATES A GIVEN TIME-DEPENDENT FUNCTION TO FIND
      C THE VALUE OF THE FUNCTION (VAL) AT A PARTICULAR TIME POINT (TIME)
      C *****
      C
      DIMENSION TFN(1),FN(1)
      C
      DO 10 N=1,NPT
          DTIME = TFN(N) - TIME
          IF(DTIME.GT.0.) GO TO 15
10    CONTINUE
      C
      15  DIFF = TFN(N) - TFN(N-1)
          VAL = FN(N) - (FN(N) - FN(N-1))*DTIME/DIFF
      C
      RETURN
      END
      SUBROUTINE QEFF (Q,C,T,TD,TDD,NUMNP)
      C
      C *****
      C FORM THE EFFECTIVE LOAD VECTOR
      C *****
      C
      COMMON/CONST /A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,THETA,DELTA,ALPHA,P1,G0EFF-
      +,RO
      DIMENSION Q(1),C(1),T(1),TD(1),TDD(1)
      C
      DO 10 I=1,NUMNP
          Q(I)=Q(I)+C(I)*(A0*T(I)+A2*TD(I)+A3*TDD(I))
10    CONTINUE
      C
      RETURN
      END
      SUBROUTINE PTEMP (T,TIME,NUMNP)
      C
      C *****
      C PUNCH THE NODAL DISPLACEMENTS
      C *****
      C
      DIMENSION T(1)
      C
      C

```

	WRITE(3,2000) TIME	PTEM	9
	NCARD = NUMNP/4 + 1	PTEM	10
	NC = 0	PTEM	11
C		PTEM	12
	DO 100 I=1,NUMNP,4	PTEM	13
	NC = NC + 1	PTEM	14
	IP = 1 + 3	PTEM	15
	IF(NC.EU.NCARD) IP = NUMNP	PTEM	16
100	WRITE(3,2001) (N,T(N),N=1,IP)	PTEM	17
C		PTEM	18
2000	FORMAT(35H NODAL POINT DISPLACEMENT AT TIME = F11.4)	PTEM	19
2001	FORMAT(4(15.5X,F10.3))	PTEM	20
C		PTEM	21
	RETURN	PTEM	22
	END	PTEM	23
	SUBROUTINE CALCUL(T,TT,TD,TDD,P,E,NUMNP)	CAL	1
	COMMON/CONST /A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,THETA,DELTA,ALPHA,PI,G	CAL	2
	+RO	CAL	3
	COMMON /CNTRL2/ KST,NDT,DT,TSTART,TAMB,NPRINT,NTSREF,TIME,KP	CAL	4
	DIMENSION T(1),TT(1),TD(1),TDD(1),P(1),E(1)	CAL	5
		CAL	6
C	T ABOVE HAS VALUES AT TIME + TAU	CAL	7
C	DO 1000 I=1,NUMNP	CAL	8
C	CALCULATE SECOND DERIVATIVE OF V. P. AT TIME+ TAU(T1)	CAL	9
C		CAL	10
	T1=A0*(T(1)-TT(1))-A2*TD(1)-A3*TDD(1)	CAL	11
C		CAL	13
C	FIND SECOND DERIVATIVE OF V.P. AT TIME+DT	CAL	14
C		CAL	14
	T2=TDD(1)	CAL	15
	TDD(1)=T2+(T1-T2)/THETA	CAL	16
C		CAL	17
C	FIND FIRST DERIVATIVE OF V.P. AT TIME+DT	CAL	18
C		CAL	19
	T3=TD(1)	CAL	20
	TD(1)=T3+A6*T2+A7*TDD(1)	CAL	21
C		CAL	22
C	FIND V.P. AT TIME+DT	CAL	23
C	T(1)=TT(1)+DT*T3+A8*T2+A9*TDD(1)	CAL	24
C		CAL	25
C	FIND PRESSURE(P) AND SURFACE DISPLACEMENT E	CAL	26
C		CAL	27
	P(1)=-RO*TD(1)	CAL	28
	E(1)=-TD(1)/G	CAL	29
1000	CONTINUE	CAL	30
300	RETURN	NPBC	92
	END	NPBC	93
	SUBROUTINE OUT (T,NUMNP,TIME,KSTEP)	OUT	1
C		OUT	2
C	*****	OUT	3
C	PRINT NOD. DISPLACEMENTS FOR *TIME*	OUT	4
C	*****	OUT	5
C		OUT	6
	DIMENSION T(1)	OUT	7
C		OUT	8
	WRITE(6,2000) KSTEP,TIME	OUT	9
	WRITE(6,2001) (N,T(N),N=1,NUMNP)	OUT	10
C		OUT	11
C	FORMAT STATEMENTS	OUT	12
C		OUT	13
2000	FORMAT(//28H DISPLACEMENT AT TIME STEP =15.2X,7H(TIME -E11.4,1H)/)	OUT	19
2001	FORMAT(6(16,E14.6))	OUT	15
C		OUT	16